



# SENSOR AND DATA FUSION

[intechweb.org](http://intechweb.org)

# SENSOR AND DATA FUSION



# **SENSOR AND DATA FUSION**

EDITED BY  
DR. IR. NADA MILISAVLJEVIĆ

***I-Tech***

Published by In-Teh

In-Teh is Croatian branch of I-Tech Education and Publishing KG, Vienna, Austria.

Abstracting and non-profit use of the material is permitted with credit to the source. Statements and opinions expressed in the chapters are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published articles. Publisher assumes no responsibility liability for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained inside. After this work has been published by the In-Teh, authors have the right to republish it, in whole or part, in any publication of which they are an author or editor, and the make other personal use of the work.

© 2009 In-teh

[www.in-teh.org](http://www.in-teh.org)

Additional copies can be obtained from:

[publication@ars-journal.com](mailto:publication@ars-journal.com)

First published February 2009

Printed in Croatia

p. cm.

ISBN 978-3-902613-52-3

1. Sensor and Data Fusion, Dr. ir. Nada Milisavljević

## Preface

Data fusion is a research area that is growing rapidly due to the fact that it provides means for combining pieces of information coming from different sources/sensors, resulting in ameliorated overall system performance (improved decision making, increased detection capabilities, diminished number of false alarms, improved reliability in various situations at hand) with respect to separate sensors/sources. Different data fusion methods have been developed in order to optimize the overall system output in a variety of applications for which data fusion might be useful: security (humanitarian, military), medical diagnosis, environmental monitoring, remote sensing, robotics... Generally speaking, there is no fusion approach that works better than the others, but depending mainly on the types (quality, quantity) of data, some approaches might be better suited to a particular problem than the others. Actually, the choice of the combination method which is best-suited for a particular application is related to types of sources/sensors, types of data (numerical, symbolic, knowledge-based; maps, legends, historical information...), amounts of data available etc. As a consequence, various data fusion techniques have been investigated for years, such as probability theory, fuzzy logic, possibility theory, evidence theory (Dempster-Shafer, belief functions). In addition, depending mainly on the types of data processing involved and on the problem itself, several levels of data fusion exist in general (e.g., pixel level, feature level, decision level, or object assessment, situation assessment, impact assessment). In different data fusion books and articles, we can also find various data fusion architectures, having three main classes: centralized, decentralized and hybrid. As a result of this variety of techniques, architectures, levels, etc., data fusion is able to bring solutions in various areas of diverse disciplines.

The goal of this book is to provide highlights of the current research in the field of data fusion. The book consists of twenty-five research papers, addressing various problems in areas such as: target tracking (including adaptive sensor management, data association and road obstacle tracking), obstacle detection for the railway traffic, real-time traffic state estimation, air traffic control, automotive applications (e.g., car safety and driver assistance), robotic systems, smoke detectors and home security, industrial instrumentation and process monitoring, remote sensing (vegetation indices, update of scarce high resolution images with time series of coarser images, land cover classification), medical imaging, anomaly detection and behavior prediction, environmental monitoring including forest fires and electromagnetic pollution, change detection, (distributed, wireless) sensor networks, etc. The list of possible applications is, actually, large, since most of the methodologies presented in the book can be adapted easily to a variety of problems and situations. The techniques involved cover a wide range of classical or novel methods, from different Bayesian-based approaches via Dempster-Shafer evidence theory and fuzzy logic to artificial neural networks and multi-agent based fusion methods. Depending mainly on the technique,

different strategies for including expert knowledge and other collateral sources of information in the fusion process are also investigated.

Altogether, the book aims to provide a valuable source of up-to-date data fusion methods, systems, applications and tools. As such, it may be useful to researchers, engineers, computer scientists, as well as to undergraduate and graduate students who are interested in the latest developments in the data fusion field.

The editor is thankful to the contributors for their precious work towards the realization of this book as well as to Dr. Vedran Kordić for his valuable help.

January 2009

Editor

**Dr. ir. Nada Milisavljević**

*Department of Communication, Information, Systems and Sensors  
Royal Military Academy,  
Brussels, Belgium*

## Contents

Preface	V
1. Advanced Sensor and Dynamics Models with an Application to Sensor Management <i>Wolfgang Koch</i>	001
2. Target Data Association Using a Fuzzy-Logic Based Approach <i>Stephen Stubberud and Kathleen Kramer</i>	035
3. Data Fusion Performance Evaluation for Dissimilar Sensors: Application to Road Obstacle Tracking <i>Blanc Christophe, Checchin Paul, Gidel Samuel and Trassoudaine Laurent</i>	057
4. IR Barrier Data Integration for Obstacle Detection <i>J. Jesús García, Jesús Ureña, Manuel Mazo and Álvaro Hernández</i>	071
5. A Model of Federated Evidence Fusion for Real-Time Traffic State Estimation <i>Qing-Jie Kong and Yuncai Liu</i>	089
6. Multi Sensor Data Fusion Architectures for Air Traffic Control Applications <i>Baud Olivier, Gomord Pierre, Honoré Nicolas, Ostorero Loïc, Taupin Olivier and Tubery Philippe</i>	103
7. Sensor Data Fusion in Automotive Applications <i>Panagiotis Lytrivis, George Thomaidis and Angelos Amditis</i>	123
8. Multisensor Data Fusion Strategies for Advanced Driver Assistance Systems <i>Mahdi Rezaei Ghahroudi and Reza Sabzevari</i>	141
9. Trajectory Generation and Object Tracking of Mobile Robot Using Multiple Image Fusion <i>TaeSeok Jin and Hideki Hashimoto</i>	167



---

10. Multisensory Data Fusion for Ubiquitous Robotics Services <i>Ren C. Luo and Ogst Chen</i>	177
11. Design of an Intelligent Housing System Using Sensor Data Fusion Approaches <i>Arezou_Moussavi Khalkhali, Behzad_ Moshiri, Hamid Reza_ Momeni</i>	191
12. Model-based Data Fusion in Industrial Process Instrumentation <i>Gerald Steiner</i>	201
13. Multi-Sensor Data Fusion in Presence of Uncertainty and Inconsistency in Data <i>Manish Kumar and Devendra P. Garg</i>	225
14. Updating Scarce High Resolution Images with Time Series of Coarser Images: a Bayesian Data Fusion Solution <i>Dominique Fasbender, Valérie Obsomer, Patrick Bogaert and Pierre Defourny</i>	245
15. Multi-Sensor & Temporal Data Fusion for Cloud-Free Vegetation Index Composites <i>Bijay Shrestha, Charles O'Hara and Preeti Mali</i>	263
16. Three Strategies for Fusion of Land Cover Classification Results of Polarimetric SAR Data <i>Nada Milisavljević, Isabelle Bloch, Vito Alberga and Giuseppe Satalino</i>	277
17. Multilevel Information Fusion: A Mixed Fuzzy Logic/Geometrical Approach with Applications in Brain Image Processing <i>Julien Montagner and Vincent Barra</i>	299
18. Anomaly Detection & Behavior Prediction: Higher-Level Fusion Based on Computational Neuroscientific Principles <i>Bradley J. Rhodes, Neil A. Bomberger, Majid Zandipour, Lauren H. Stolarz, Denis Garagic, James R. Dankert and Michael Seibert</i>	323
19. A Biologically Based Framework for Distributed Sensory Fusion and Data Processing <i>Ferro M. and Pioggia G.</i>	337
20. Agent Based Sensor and Data Fusion in Forest Fire Observer <i>Ljiljana Šerić, Darko Stipaničev and Maja Štula</i>	365

- 
21. A Sensor Data Fusion Procedure for Environmental Monitoring Applications by a Configurable Network of Smart Web-Sensors 379  
*Claudio De Capua and Rosario Morello*
22. Monitoring Changes in Operational Scenarios via Data Fusion in Sensor Networks 401  
*Papantoni-Kazakos, Dr. Titsa and Burrell, Dr. Anthony*
23. Elements of Sequential Detection with Applications to Sensor Networks 417  
*Stefano Marano and Vincenzo Matta*
24. Parameter Estimation Over Noisy Communication Channels in Distributed Sensor Networks 437  
*Thakshila Wimalajeewa, Sudharman K. Jayaweera and Carlos Mosquera*
25. Monte Carlo Methods for Node Self-Localization and Nonlinear Target Tracking in Wireless Sensor Networks 471  
*Joaquín Míguez, Luis Arnaiz and Antonio Artés-Rodríguez*



# Advanced Sensor and Dynamics Models with an Application to Sensor Management

Wolfgang Koch  
*German Defence Research Establishment (FGAN e.V.)*  
Germany

## 1. Introduction

The methods provided by sensor and data fusion [14] are important tools for fusing large sets of mutually complementary data and efficiently exploiting the sensor systems available. A challenging exploitation technology at the common interface between sensors, command & control systems, and the human decision makers involved, this technology plays a key role in applications with time-critical situations or in situations with a high decision risk, where human deficiencies are to be compensated by automatically or interactively working fusion techniques (compensating decreasing attention in routine situations, focusing the attention on anomalous or rare events, complementing limited memory, reaction, or combination capabilities of human beings). Besides the advantages of reducing the human work load in routine or mass tasks, data fusion from mutually complementary information sources can well produce qualitatively new knowledge that otherwise would remain unrevealed.

### *A. Providing Elements for Situation Pictures*

Sensor and data fusion provides 'information elements' for producing near real-time situation pictures, which electronically represent a complex and dynamically evolving overall scenario in the air, on the ground, at sea, or in an urban environment. The concrete operational requirements in a given application define the particular information sources to be fused. A careful analysis of the underlying requirements is thus essential for any fusion system design.

Information elements are extracted from currently received sensor data while taking into account the available context knowledge and pre-history. They typically provide answers to questions related to objects of interest such as: Do objects exist at all, and how many of them are in the sensors' fields of view? Where are they at what time? Where will they be in the future with what probability? How can their overall behavior be characterized? Are anomalies or hints about their possible intentions recognizable? What can be inferred about the classes the objects belong to or even their identities? Are there characteristic interrelations between individual objects? In which regions do they have their origin? What can be said about their possible destinations? Are object flows visible? Where are sources or sinks of traffic?

The sensor data to be fused can be inaccurate, incomplete, or ambiguous. Closely-spaced objects are often totally or partially unresolvable. Possibly, the measured object parameters

are false or corrupted by hostile measures. The context information is in many cases hard to be formalized or even contradictory. These deficiencies of the information to be fused are unavoidable in any real-world application. Therefore, the extraction of ‘information elements’ for situation pictures is by no means trivial.

### B. Aspects of Sensor and Data Fusion

Among the primary technical prerequisites for sensor data and information fusion are communication links with a sufficient bandwidth, small latency, and robustness against failure or jamming. Moreover, the transformation of the sensor data into a common coordinate system requires a precise space-time registration of the sensors, including their mutual alignment.

Figure 1 provides an overview of different aspects and their mutual interrelation. The sensors play a central role and can be located in different ways (collocated, distributed, mobile) producing measurements of the same or of a different type. Fusion of heterogeneous sensor data is of particular importance, such as the combination of kinematic measurements with measured attributes providing information on the classes to which objects belongs to. In the context of defense and security applications especially, the distinction between active and passive sensing is important since passive sensors enable covert surveillance, which does not reveal itself by emitting radiation. Multifunctional sensor systems offer additional operational modes, thus requiring more intelligent strategies of sensor management that provide feedback via control or correction commands to the process of information acquisition. By this the surveillance objectives can often be reached more efficiently. Context information is given, for example, by available knowledge on the sensor and object properties, which is often quantitatively described by statistical models. Context knowledge is also environmental information on roads or topographical occlusions (GIS: Geographical Information Systems). Seen from a different perspective, context information, such as road maps, can be extracted from real-time sensor data as well [27]. Militarily relevant context knowledge (e.g. doctrines, planning data, tactics) and human observer reports (HUMINT: Human Intelligence) is also important information in the fusion process [4]. The exploitation of context information of any kind can significantly improve the fusion system performance.

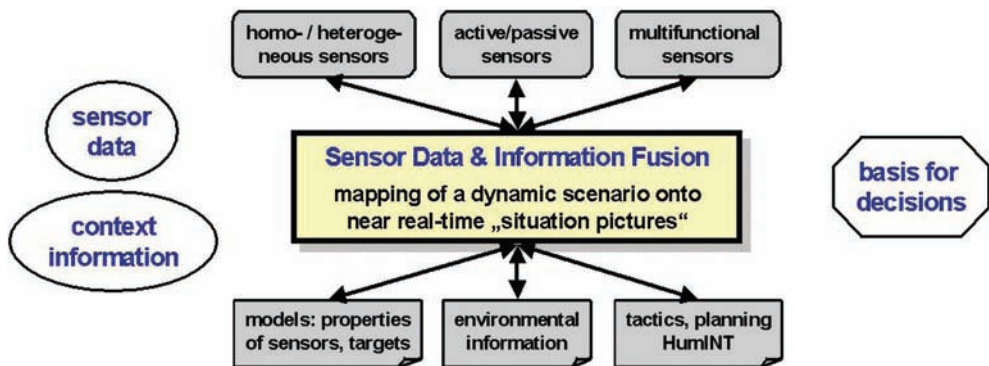


Fig. 1. Sensor data and information fusion for situation pictures: overview of characteristic aspects and their mutual interrelation.

The information elements required for producing a timely situation picture are provided by an integrative, spatio-temporal processing of the various pieces of information available, which in themselves often have only limited value for understanding the situation. Essentially, within the fusion process logical cross-references, inherent complementarity, and redundancy are exploited. More concretely speaking, the methods used are characterized by a stochastic approach (estimating relevant state quantities) and a more heuristically defined knowledgebased approach (imitating the actual human behavior when exploiting information). Besides the operational requirements, this more or less coherent methodology is the second building principle, which gives the field of sensor data and information fusion its characteristic shape.

### C. Overview of a Generic Tracking System

Among the fusion products, so-called 'tracks' are of particular importance. Tracks represent knowledge on relevant state quantities of individual objects, object groups such as convoys and formations, or even large object aggregations (e.g. march columns). The information obtained by 'tracking' [6], [2], [22] includes in particular the history of the objects. If possible, a one-to-one association between the objects/object groups and the tracks is to be established and has to be preserved as long as possible (track continuity). Quantitative measures describing the quality of this knowledge are important constituents of tracks. The achievable track quality, however, does not only depend on the sensor performance, but also on the operational conditions within the actually considered scenario and the available context knowledge.

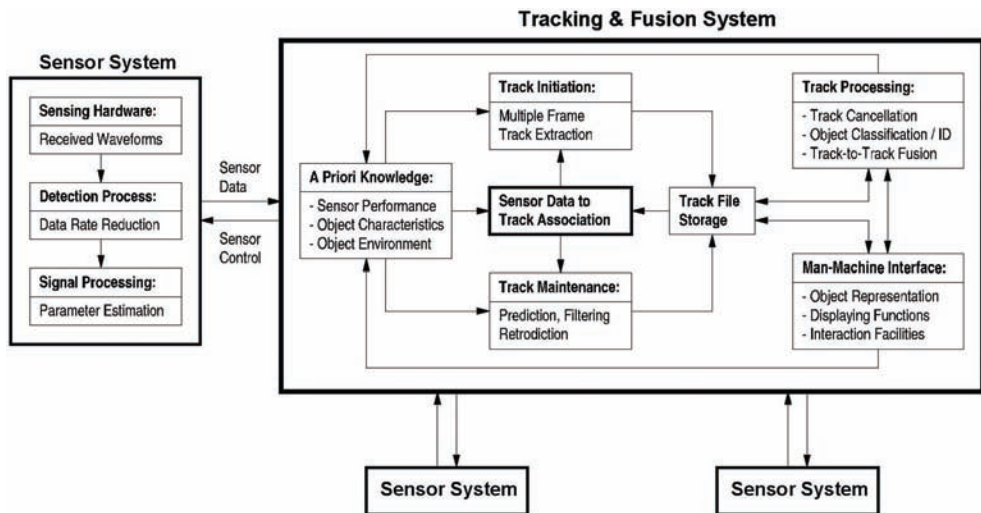


Fig. 2. Generic scheme of functional building blocks within a tracking/fusion system along with its relation to the sensors (centralized configuration, type IV according to O. Drummond).

Figure 2 shows a generic scheme of functional building blocks within a tracking/fusion system along with its relation to the underlying sensors. After passing a detection process, essentially working as a means of data rate reduction, the signal processing provides

estimates of parameters characterizing the waveforms received at the sensors' front ends (e.g. radar antennas). From these estimates sensor reports are created, i.e. measured quantities possibly related to objects of interest, which are the input for the tracking/fusion system. All sensor data that can be associated to existing tracks are used for track maintenance (using, e.g., prediction, filtering, and retrodiction). The remaining data are processed for initiating new tentative tracks (multiple frame track extraction). Association techniques thus play a key role in tracking/fusion applications. Context information in terms of statistical models (sensor performance, object characteristics, object environment) is a prerequisite to track maintenance and initiation. Track confirmation/termination, classification/identification, and fusion of tracks related to the same objects or object groups is part of the track processing. The scheme is completed by a man-machine interface with displaying and interaction functions. Context information can be updated or modified by direct human interaction or by the track processor itself, for example as a consequence of object classification or road map extraction. In the case of multifunctional sensors, feedback exists from the tracking system to the process of sensor data acquisition (sensor management).

#### *D. A Characteristic Application: Sensor Management*

Modern multifunctional agile-beam radar based on phased-array technology is an excellent example for a sensor system that requires sophisticated sensor management algorithms. This is particularly true for multiple object tracking tasks where such systems call for algorithms that efficiently exploit their degrees of freedom, which are variable over a wide range and may be chosen individually for each track. Of special interest are military air situations where both agile objects and objects significantly differing in their radar cross section must be taken into account. Unless properly handled, such situations can be highly allocation time- and energy-consuming. In this context, advanced sensor and dynamics models for combined tracking and sensor management are discussed, i.e. control of data innovation intervals, radar beam positioning, and transmitted energy management. By efficiently exploiting its limited resources, the total surveillance performance of the sensor system can be much improved.

Figure 3 shows a simplified scheme illustrating the information flow in tracking-driven phasedarray radar management. The starting point is the tracking system, which generates a request for new sensor information based on the current quality of an already established individual object track or on the requirement of initiating new tracks. We thus distinguish between track update and search requests, which enter into the priority management unit where its rank is evaluated based on the current threat or overload situation, for example, thus enabling graceful system degradation when necessary.

For each preparation of a radar system allocation, track-specific radar parameters must be set, such as the calculated radar revisit time and the corresponding radar beam position, range and Doppler-gates, or the type of the radar wave forms to be transmitted. Track search requests require the setting of appropriate revisit intervals, search sectors and patterns, and other radar parameters. In the dwell scheduling unit these preparations are transformed into antenna commands, by which the radar sensor is allocated and radar energy transmitted. The received echo signals pass a detection unit. If no detection occurs in the track maintenance mode, a local search procedure is initiated, new radar parameters are set, and a subsequent radar sensor allocation is started with as small a time delay as possible. This local search loop is repeated until either a valid detection is produced or the track is

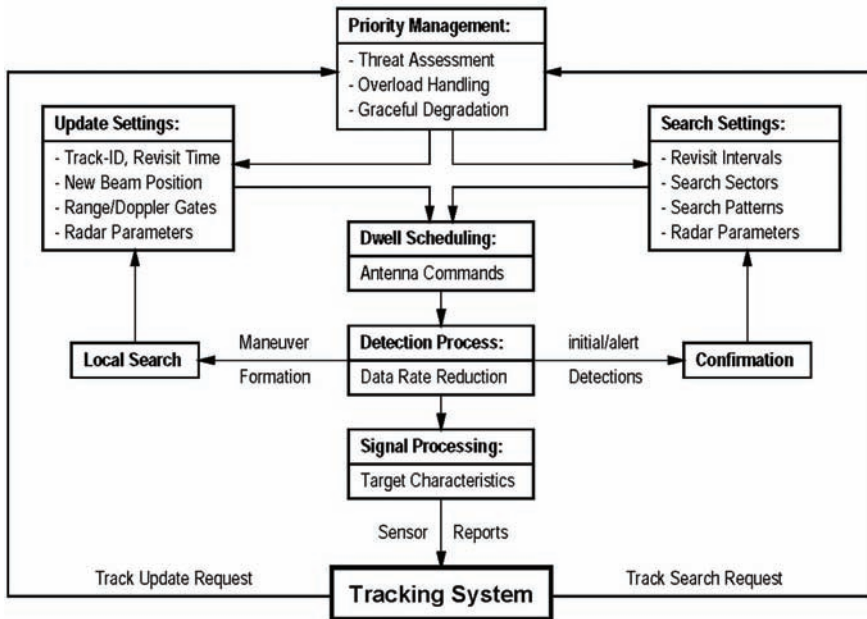


Fig. 3. Simplified scheme of the information flow in tracking-driven phased-array radar management.

canceled. While a new beam position according to a global or sector search pattern is calculated if no detection occurs in the track search mode, a tentative detection has to be confirmed before a new track is finally established. After a successful detection, the received signal passes the signal processing unit, where characteristic object parameters, such as object range, azimuth angle, radial velocity, and the object strength, are estimated being the input for the tracking system. This closes the data processing and sensor management loop. In military applications, distinct maneuvering phases often exist, since even agile objects do not maneuver permanently. Nevertheless, abrupt transitions to high- $g$  turns can well occur. Allocation time and energy savings are thus to be expected if adaptive dynamics models of the object dynamics are used. Besides their kinematic characteristics, the mean radar cross section (RCS) of the objects to be tracked is usually unknown and variable over a wide range. By processing of signal amplitude information, however, the energy spent for track maintenance can be adapted to the actual object strength. By this measure the total sensor load can also be significantly reduced.

Due to the locally confined object illumination by the pencil-beam of a phased-array radar, abrupt transitions into maneuvering flight phases are critical since, in contrast to more conventional track-while-scan radar, a periodic object illumination is no longer guaranteed. Any track reinitiation is thus highly allocation time- and energy-consuming and also locks the sensor for other tasks (e.g. weapon guidance or providing communications links). This calls for intelligent algorithms for beam positioning and local search [17], [24], [20] that are crucial to phased-array radar tracking.

For track-while-scan radar systems, Bayesian tracking techniques are well-established. They provide an iterative updating scheme for conditional probability densities of the object state,



given all sensor data and a priori information available. In those applications data acquisition and tracking are completely decoupled. For phased-array radar, however, the current signal-to-noise ratio of the object (i.e. the detection probability) strongly depends on the correct positioning of the pencil-beam, which is now taken into the responsibility of the tracking system. Sensor control and data processing are thus closely interrelated. This basically *local* character of the tracking process constitutes the principal difference between phased-array and track-while-scan applications from a tracking point of view. By using suitable sensor models, however, this fact can be incorporated into the Bayesian formalism. The potential of this approach is thus also available for phased-array radar. The more difficult problem of global optimization, taking successive allocations into account, is not addressed here.

## 2. Sensor and dynamics models in bayesian object tracking

Fusing data produced at different instants of time, i.e. the tracking problem, is typically characterized by uncertainty and ambiguities, which are inherent in the underlying scenario, the object dynamics, and the sensors used. The Bayesian approach provides a well-suited methodology for dealing with many of these phenomena. More concretely speaking, the Bayesian approach provides a processing scheme for dealing with uncertain information (of a particular type), which also allows to make ‘delayed’ decisions if a unique decision cannot be made in a particular data situation. Ambiguities can have different causes: Sensors may produce ambiguous data due to their limited resolution capabilities or due to phenomena such as Doppler blindness in MTI radar (MTI: Moving Target Indicator). Often the objects’ environment is a source of ambiguities itself (dense object situations, residual clutter, man-made noise, unwanted objects). A more indirect type of ambiguities arises from the objects’ behavior (e.g. qualitatively distinct maneuvering phases). Finally, the context knowledge to be exploited can imply problem-inherent ambiguities as well, such as intersections in road maps or ambiguous tactical rules describing the over-all object behavior. The general multiple-object, multiple-sensor tracking task, however, is highly complex and involves sophisticated combinatorial considerations that are beyond the scope of this chapter (see [5], [30] as an introduction). Nevertheless, in many applications, the tracking task can be partitioned into independent sub-problems of (much) less complexity.

According to this discussion, we proceed along the following lines.

- *Basis:* In the course of time, one or several sensors produce measurements of one or more objects of interest. The accumulated sensor data are an example of a ‘time series’. Each object is characterized by its current ‘state’, a vector typically consisting of the current object position, its velocity, and acceleration.
- *Objective:* Learn as much as possible about the individual object states at each time of interest by analyzing the ‘time series’ created by the sensor data.
- *Problem:* The sensor information is inaccurate, incomplete, and possibly even ambiguous. Moreover, the objects’ temporal evolution is usually not well-known.
- *Approach:* Interpret sensor measurements and object state vectors as random variables. Describe by probability density functions (pdfs) what is known about these random variables.
- *Solution:* Derive iteration formulae for calculating the probability density functions of the state variables and develop a mechanism for initiating the iteration. Derive state estimates from the pdfs along with appropriate quality measures.

### A. The Key-role of Bayes' Formula

At particular instants of time denoted by  $t_l$ ,  $l = 1, \dots, k$ , we consider the set  $Z_l = \{\mathbf{z}_l^1, \dots, \mathbf{z}_l^{n_l}\}$  of  $n_l$  measurements related to the object state  $\mathbf{x}_l$ . In case of multiple objects  $\mathbf{x}_l$  is the joint state. The corresponding time series up to and including  $t_k$  is recursively defined by  $\mathcal{Z}^k = \{Z_k, n_k, \mathcal{Z}^{k-1}\}$ . The central question of object tracking is: What can be known about the object states  $\mathbf{x}_l$  at time instants  $t_l$ , i.e. for the past ( $l < k$ ), at present ( $l = k$ ), and in the future ( $l > k$ ), by exploiting the sensor data collected in the times series  $\mathcal{Z}^k$ ? According to the approach previously sketched, the answer is given by the conditional probability density functions (pdf)  $p(\mathbf{x}_l | \mathcal{Z}^k)$  to be calculated iteratively as a consequence of Bayes' rule. For  $l = k$ , i.e. for object states at the current time  $t_k$ , we obtain:

$$p(\mathbf{x}_k | \mathcal{Z}^k) = \frac{p(Z_k, n_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathcal{Z}^{k-1})}{\int d\mathbf{x}_k p(Z_k, n_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathcal{Z}^{k-1})}. \quad (1)$$

In other words,  $p(\mathbf{x}_k | \mathcal{Z}^k)$  can be calculated from the pdfs  $p(\mathbf{x}_k | \mathcal{Z}^{k-1})$  and  $p(Z_k, n_k | \mathbf{x}_k)$ .  $p(\mathbf{x}_k | \mathcal{Z}^{k-1})$  describes, what is known on  $\mathbf{x}_k$  given all past sensor data  $\mathcal{Z}^{k-1}$ , i.e. a prediction. Obviously,  $p(Z_k, n_k | \mathbf{x}_k)$  needs to be known up to a constant factor only. Any function

$$\ell(\mathbf{x}_k; Z_k, n_k) \propto p(Z_k, n_k | \mathbf{x}_k) \quad (2)$$

produces the same result. Functions of this type are also called *likelihood functions* and describe what can be learned from the current sensor output  $Z_k, n_k$  about the object state  $\mathbf{x}_k$  at this time. This is the reason, why likelihood functions are often also called "sensor models", since they mathematically represent the sensor, its measurements and properties, in the data processing formalism. For well-separated objects, perfect detection, in absence of false returns, and for bias-free measurements of linear functions  $\mathbf{H}_k \mathbf{x}_k$  of the object state with a Gaussian, white noise measurement error characterized by a covariance matrix  $\mathbf{R}_k$ , the likelihood functions are proportional to a Gaussian:  $\ell(\mathbf{x}_k; \mathbf{z}_k, \mathbf{H}_k, \mathbf{R}_k) \propto \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k)$ .

### B. Prediction Update Step

The pdf  $p(\mathbf{x}_k | \mathcal{Z}^{k-1})$  in the Equation 1 is a prediction of the knowledge on the object state for the time  $t_k$  based on all the measurements received up to and including time  $t_{k-1}$ . By writing this pdf as a marginal density,  $p(\mathbf{x}_k | \mathcal{Z}^{k-1}) = \int d\mathbf{x}_{k-1} p(\mathbf{x}_k, \mathbf{x}_{k-1} | \mathcal{Z}^{k-1})$ , the object state  $\mathbf{x}_{k-1}$  at the previous time  $t_{k-1}$  comes into play yielding:

$$p(\mathbf{x}_k | \mathcal{Z}^{k-1}) = \int d\mathbf{x}_{k-1} \underbrace{p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathcal{Z}^{k-1})}_{\text{object dynamics}} \underbrace{p(\mathbf{x}_{k-1} | \mathcal{Z}^{k-1})}_{\text{previous posterior}}. \quad (3)$$

The state transition density  $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathcal{Z}^{k-1})$  is often called the "object dynamics model" and mathematically represents the kinematic object properties in the data processing formalism in the same way as the likelihood function represents the sensor(s).

1) *Gauss-Markov Dynamics*: A Gauss-Markov dynamics, defined by the transition density

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathcal{Z}^{k-1}) = \mathcal{N}(\mathbf{x}_k; \mathbf{F}_{k|k-1} \mathbf{x}_{k-1}, \mathbf{D}_{k|k-1}), \quad (4)$$

is characterized by the modeling parameters  $\mathbf{F}_{k|k-1}$  (evolution matrix), describing the deterministic part of the temporal evolution, and  $\mathbf{D}_{k|k-1}$  (dynamics covariance matrix), characterizing its stochastic part. If we additionally assume that the previous posterior is a Gaussian, given by

$$p(\mathbf{x}_{k-1} | \mathcal{Z}^{k-1}) = \mathcal{N}(\mathbf{x}_{k-1}; \mathbf{x}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}), \quad (5)$$

$p(\mathbf{x}_k | \mathcal{Z}^{k-1})$  is also a Gaussian:

$$p(\mathbf{x}_k | \mathcal{Z}^{k-1}) = \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}, \mathbf{P}_{k|k-1}) \quad (6)$$

with an expectation vector  $\mathbf{x}_{k|k-1}$  and a covariance matrix  $\mathbf{P}_{k|k-1}$  given by:

$$\mathbf{x}_{k|k-1} = \mathbf{F}_{k|k-1} \mathbf{x}_{k-1|k-1} \quad (7)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k|k-1} \mathbf{P}_{k-1|k-1} \mathbf{F}_{k|k-1}^\top + \mathbf{D}_{k|k-1}. \quad (8)$$

This directly results from a useful product formula for Gaussians<sup>1</sup>:

$$\mathcal{N}(\mathbf{z}; \mathbf{H}\mathbf{x}, \mathbf{R}) \mathcal{N}(\mathbf{x}; \mathbf{y}, \mathbf{P}) = \mathcal{N}(\mathbf{z}; \mathbf{H}\mathbf{y}, \mathbf{S}) \times \begin{cases} \mathcal{N}(\mathbf{x}; \mathbf{y} + \mathbf{W}\boldsymbol{\nu}, \mathbf{P} - \mathbf{W}\mathbf{S}\mathbf{W}^\top) \\ \mathcal{N}(\mathbf{x}; \mathbf{Q}(\mathbf{P}^{-1}\mathbf{x} + \mathbf{H}^\top \mathbf{R}^{-1}\mathbf{z}), \mathbf{Q}) \end{cases}, \quad (9)$$

where we used the abbreviations:

$$\boldsymbol{\nu} = \mathbf{z} - \mathbf{H}\mathbf{y}, \quad \mathbf{S} = \mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{R}, \quad \mathbf{W} = \mathbf{P}\mathbf{H}^\top \mathbf{S}^{-1}, \quad \mathbf{Q}^{-1} = \mathbf{P}^{-1} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}. \quad (10)$$

Note that after applying this formula the integration variable  $\mathbf{x}_{k-1}$  in the Equation 3 is no longer contained in the first Gaussian of the product. The integration becomes thus trivial as pdfs are normalized.

2) *IMM Dynamics Model*: In practical applications, it might be uncertain which dynamics model out of a set of possible alternatives is currently in effect. Such cases, e.g. objects characterized by different modes of dynamical behavior, can be handled by multiple dynamics models with a given probability of switching between them (IMM: Interacting Multiple Models, [2], [6] and the literature cited therein). The model transition probabilities are thus part of the modeling assumptions. More strictly speaking, suppose that  $r$  models are given and let  $j_k$  be denoting the dynamics model assumed to be in effect at time  $t_k$ , the statistical properties of systems with Markovian switching coefficients are summarized by the following equation:

$$p(x_k, j_k | x_{k-1}, j_{k-1}) = p(x_k | x_{k-1}, j_k) p(j_k | j_{k-1}) \quad (11)$$

---

<sup>1</sup>Sketch of proof: Interpret  $\mathcal{N}(\mathbf{z}; \mathbf{H}\mathbf{x}, \mathbf{R})\mathcal{N}(\mathbf{x}; \mathbf{y}, \mathbf{P})$  as a joint density  $p(\mathbf{z}, \mathbf{x}) = p(\mathbf{z} | \mathbf{x})p(\mathbf{x})$ . It can be written as a Gaussian, from which the marginal and conditional densities  $p(\mathbf{z})$ ,  $p(\mathbf{x} | \mathbf{z})$  can be derived. In the calculations make use of known formulae for the inverse of a partitioned matrix (see [2, p. 22], e.g.). From  $p(\mathbf{z}, \mathbf{x}) = p(\mathbf{x} | \mathbf{z})p(\mathbf{z})$  the formula results.

$$= p(j_k | j_{k-1}) \mathcal{N}(\mathbf{x}_k; \mathbf{F}_{k|k-1}^{j_k} \mathbf{x}_{k-1}, \mathbf{D}_{k|k-1}^{j_k}). \quad (12)$$

For  $r = 1$ , the previous linear-Gaussian model results as a limiting case. Fortunately, the tracking performance does not seem to critically depend on the particular choice of the model transition probabilities  $p(j_k | j_{k-1})$ , provided the number  $r$  of models involved is small [7].

Let us assume that the previous posterior is written as a Gaussian mixture,

$$p(\mathbf{x}_{k-1} | \mathcal{Z}^{k-1}) = \sum_{j_{k-1}=1}^r \cdots \sum_{j_{k-n}=1}^r p(\mathbf{x}_{k-1}, j_{k-1}, \dots, j_{k-n} | \mathcal{Z}^{k-1}) \quad (13)$$

$$= \sum_{\mathbf{j}_{k-1}} p(\mathbf{x}_{k-1}, \mathbf{j}_{k-1} | \mathcal{Z}^{k-1}) \quad (14)$$

$$= \sum_{\mathbf{j}_{k-1}} p(\mathbf{j}_{k-1} | \mathcal{Z}^{k-1}) \mathcal{N}(\mathbf{x}_{k-1}; \mathbf{x}_{k-1}^{\mathbf{j}_{k-1}}, \mathbf{P}_{k-1|k-1}^{\mathbf{j}_{k-1}}), \quad (15)$$

i.e. a weighted sum of individual Gaussians. The vector index  $\mathbf{j}_{k-1}$  is defined by  $\mathbf{j}_{k-1} = j_{k-1}, j_{k-2}, \dots, j_{k-n}$ , i.e. the mixture  $p(\mathbf{x}_{k-1} | \mathcal{Z}^{k-1})$  is given by  $r^n$  components, where  $n$  is a parameter. The case  $n = 1$  corresponds to the situation standard IMM prediction [2, p. ???ff]. With a previous posterior of this type, we obtain for the prediction update:

$$p(\mathbf{x}_k | \mathcal{Z}^{k-1}) = \sum_{j_k=1}^r \sum_{\mathbf{j}_{k-1}} \int d\mathbf{x}_{k-1} p(\mathbf{x}_k, j_k | \mathbf{x}_{k-1}, j_{k-1}) p(\mathbf{x}_{k-1}, \mathbf{j}_{k-1} | \mathcal{Z}^{k-1}) \quad (16)$$

$$= \sum_{j_k=1}^r \sum_{\mathbf{j}_{k-1}} p_{k|k-1}^{j_k, \mathbf{j}_{k-1}} \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{j_k, \mathbf{j}_{k-1}}, \mathbf{P}_{k|k-1}^{j_k, \mathbf{j}_{k-1}}) \quad (17)$$

$$= \sum_{\mathbf{j}_k} \sum_{j_{k-1}=1}^r p_{k|k-1}^{\mathbf{j}_k, j_{k-1}} \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{\mathbf{j}_k, j_{k-1}}, \mathbf{P}_{k|k-1}^{\mathbf{j}_k, j_{k-1}}) \quad (18)$$

with weighting factors  $p_{k|k-1}^{j_k, \mathbf{j}_{k-1}}$ , an expectation vector  $\mathbf{x}_{k|k-1}^{j_k, \mathbf{j}_{k-1}}$ , and a covariance matrix  $\mathbf{P}_{k|k-1}^{j_k, \mathbf{j}_{k-1}}$  given by:

$$p_{k|k-1}^{j_k, \mathbf{j}_{k-1}} = p(j_k | \mathbf{j}_{k-1}) p(\mathbf{j}_{k-1} | \mathcal{Z}^{k-1}) \quad (19)$$

$$\mathbf{x}_{k|k-1}^{j_k, \mathbf{j}_{k-1}} = \mathbf{F}_{k|k-1}^{j_k} \mathbf{x}_{k-1|k-1}^{\mathbf{j}_{k-1}} \quad (20)$$

$$\mathbf{P}_{k|k-1}^{j_k, \mathbf{j}_{k-1}} = \mathbf{F}_{k|k-1}^{j_k} \mathbf{x}_{k-1|k-1}^{\mathbf{j}_{k-1}} \mathbf{F}_{k|k-1}^{j_k \top} + \mathbf{D}_{k|k-1}^{j_k} \quad (21)$$

by exploiting the product formula (Equation 9). From these considerations follows that the number of mixture components is continuously increasing in each prediction update step.

Via moment matching [2, p. 56], the number of mixture components can be kept constant if the sum over  $j_{k-n}$  in Equation 18 is approximated by:

$$\sum_{j_{k-n}=1}^r p_{k|k-1}^{j_k, j_{k-1}} \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{j_k, j_{k-1}}, \mathbf{P}_{k|k-1}^{j_k, j_{k-1}}) \approx p(\mathbf{j}_k | \mathcal{Z}^{k-1}) \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{j_k}, \mathbf{P}_{k|k-1}^{j_k}), \quad (22)$$

with  $p(\mathbf{j}_k | \mathcal{Z}^{k-1})$ ,  $\mathbf{x}_{k|k-1}^{j_k}$ , and  $\mathbf{P}_{k|k-1}^{j_k}$  given by:

$$p_{k|k-1}^{j_k} = \sum_{j_{k-n}=1}^r p_{k|k-1}^{j_k, j_{k-1}} \quad (23)$$

$$\mathbf{x}_{k|k-1}^{j_k} = \frac{1}{p_{k|k-1}^{j_k}} \sum_{j_{k-n}=1}^r p_{k|k-1}^{j_k, j_{k-1}} \mathbf{x}_{k|k-1}^{j_k, j_{k-1}} \quad (24)$$

$$\mathbf{P}_{k|k-1}^{j_k} = \frac{1}{p_{k|k-1}^{j_k}} \sum_{j_{k-n}=1}^r p_{k|k-1}^{j_k, j_{k-1}} (\mathbf{P}_{k|k-1}^{j_k, j_{k-1}} + (\mathbf{x}_{k|k-1}^{j_k, j_{k-1}} - \mathbf{x}_{k|k-1}^{j_k})(\mathbf{x}_{k|k-1}^{j_k, j_{k-1}} - \mathbf{x}_{k|k-1}^{j_k})^\top) \quad (25)$$

yielding a Gaussian sum representation of  $p(\mathbf{x}_k | \mathcal{Z}^{k-1})$  with  $r^m$  mixture components.

### C. Filtering Update Step

According to previous considerations, the conditional pdf  $p(\mathbf{x}_k | \mathcal{Z}^k)$  can be calculated iteratively by combining the following pieces of evidence:  $p(\mathbf{x}_{k-1} | \mathcal{Z}^{k-1})$  (knowledge of the past),  $p(\mathbf{x}_k | \mathbf{x}_{k-1})$  (object dynamics),  $\ell(\mathbf{x}_k; \mathbf{z}_k, n_k)$  (measurements, sensor model).

1) *Standard Kalman Update Formulae*: In case of well-separated objects under ideal conditions, i.e. without false returns, assuming perfect detection, a single dynamics model, and Gaussian measurement errors, the well-known Kalman filtering results as a limiting case of this more general Bayesian approach. The Kalman filter is thus a simple straight-forward realization of Bayesian tracking. In this idealized situation, i.e. with:

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x}_k; \mathbf{F}_{k|k-1} \mathbf{x}_{k-1}, \mathbf{D}_{k|k-1}) \quad (26)$$

$$\ell(\mathbf{x}_k; \mathbf{z}_k) = \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k), \quad (27)$$

Equation 1 provides Gaussian pdfs,

$$p(\mathbf{x}_k | \mathcal{Z}^k) = \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k}, \mathbf{P}_{k|k}), \quad (28)$$

representing the available knowledge at each time  $t_k$ . According to the previous product formula (Equation 9), we obtain two equivalent versions of the Kalman update equations for  $\mathbf{x}_{k|k}$ ,  $\mathbf{P}_{k|k}$ :

$$\mathbf{x}_{k|k} = \begin{cases} \mathbf{x}_{k|k-1} + \mathbf{W}_{k|k-1}(\mathbf{z}_k - \mathbf{H}_k \mathbf{x}_{k|k-1}) \\ \mathbf{P}_{k|k}(\mathbf{P}_{k|k-1}^{-1} \mathbf{x}_{k|k-1} + \mathbf{H}_k^\top \mathbf{R}_k^{-1} \mathbf{z}_k) \end{cases} \quad (29)$$

$$\mathbf{P}_{k|k} = \begin{cases} \mathbf{P}_{k|k-1} - \mathbf{W}_{k|k-1} \mathbf{S}_{k|k-1} \mathbf{W}_{k|k-1}^\top \\ (\mathbf{P}_{k|k-1}^{-1} + \mathbf{H}_k^\top \mathbf{R}_k^{-1} \mathbf{H}_k)^{-1} \end{cases} \quad (30)$$

with the Kalman Gain Matrix  $\mathbf{W}_{k|k-1}$  and the Innovation Covariance Matrix  $\mathbf{S}_{k|k-1}$ , given by

$$\mathbf{W}_{k|k-1} = \mathbf{P}_{k|k-1} \mathbf{H}_{k|k-1}^\top \mathbf{S}_{k|k-1}^{-1}, \quad (31)$$

$$\mathbf{S}_{k|k-1} = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^\top + \mathbf{R}_k. \quad (32)$$

At time  $t_0$ , the pdf  $p(\mathbf{x}_0 | \mathcal{Z}^0) = \mathcal{N}(\mathbf{x}_0; \mathbf{x}_{0|0}, \mathbf{P}_{0|0})$  describes the initial knowledge on the object kinematics. As an example, let us consider state vectors  $\mathbf{x}_k = (\mathbf{r}_k^\top, \dot{\mathbf{r}}_k^\top)^\top$ , consisting of the object position and velocity, and position measurements  $\mathbf{z}_k$  with measurement error covariance matrices  $\mathbf{R}_k$ . Based on a first measurement  $\mathbf{z}_0$  and the context information that  $v_{\max}$  is a measure of the maximum object speed to be expected, a reasonable initiation is given by  $\mathbf{x}_{0|0} = (\mathbf{z}_0^\top, \mathbf{0}^\top)^\top$ ,  $\mathbf{P}_{0|0} = \text{diag}[\mathbf{R}_0, v_{\max}^2 \mathbf{1}]$ .

2) *More Sophisticated Sensor Models*: A very simple example illustrates, in which way negative sensor evidence, i.e. an expected but actually missing sensor measurement, is to be treated within the Bayesian formalism. Let us first exclude false measurements and assume that the objects of interest are detected with a constant detection probability  $P_D < 1$ . This problem is thus identical with the previously discussed Kalman filtering except that measurements are not at each time  $t_k$  available. In this case, the underlying sensor model, i.e. the likelihood function, has not only to describe the measurement process, characterized by the measurement matrix  $\mathbf{H}_k$  and the measurement error covariance matrix  $\mathbf{R}_k$ , but also the detection process, characterized by the detection probability  $P_D < 1$ . According to this discussion, there exist two possibilities: either the object was detected at time  $t_k$  (data interpretation hypothesis  $i_k = 1$ , or not (data interpretation hypothesis  $i_k = 0$ ). Under the assumption that the probabilities  $p(i_k = 1 | \mathbf{x}_k) = P_D$  and  $p(i_k = 0 | \mathbf{x}_k) = 1 - P_D$  do not depend on the object state  $\mathbf{x}_k$ , we obtain with  $\delta_{ij} = 0$  for  $i \neq j$  and  $\delta_{ij} = 1$  for  $i = j$  the following likelihood function:

$$p(Z_k, n_k | \mathbf{x}_k) = \sum_{i_k=0}^1 p(Z_k, n_k, i_k | \mathbf{x}_k) = \sum_{i_k=0}^1 p(Z_k, n_k | i_k, \mathbf{x}_k) p(i_k | \mathbf{x}_k) \quad (33)$$

$$= \delta_{0, n_k} (1 - P_D) + \delta_{1, n_k} P_D \mathcal{N}(\mathbf{z}_k; \mathbf{H} \mathbf{x}_k, \mathbf{R}) \quad (34)$$

$$=: \ell(\mathbf{x}_k; \mathbf{z}_k, \mathbf{H}_k, \mathbf{R}_k, P_D). \quad (35)$$

With  $p(\mathbf{x}_k | \mathcal{Z}^{k-1}) = \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}, \mathbf{P}_{k|k-1})$ , Equation 1 leads to the following conclusions:

1. For a positive sensor output ( $n_k = 1$ ) the measurement  $\mathbf{z}_k$  is processed via Kalman filtering resulting in  $p(\mathbf{x}_k | \mathcal{Z}^k) = \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k}, \mathbf{P}_{k|k})$  with  $\mathbf{x}_{k|k}$  and  $\mathbf{P}_{k|k}$  given by Equations 29 and 30.
2. For a negative sensor output ( $n_k = 0$ ) the likelihood function is given by the constant  $1 - P_D$ . This implies that prediction pdf is not modified in the filtering step:  $\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1}$

$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1}$ . According to the Kalman update equations this result can formally be interpreted as a processing of a *pseudo*-measurement with an infinitely large measurement error covariance matrix  $\mathbf{R}'_k$ , since in this case  $\mathbf{R}'_k^{-1}=0$ .

The Bayesian formalism and the sensor model (likelihood function) obviously define how a negative sensor output, i.e. a missing detection is to be processed.

In the case of well-separated objects in the presence of false returns and imperfect detection, the  $n_k$  sensor data  $Z_k$  are also not longer uniquely interpretable. Let  $i_k = 0$  denote the data interpretation hypothesis that the object has not been detected at time  $t_k$ , all sensor data being false returns, while  $i_k = i, i = 1, \dots, n_k$  represents the hypothesis that the object has been detected,  $\mathbf{z}_k^i \in Z_k$  being a object measurement, the remaining sensor data being false returns. Evidently,  $\{\mathbf{z}_k^i\}_{i=0}^{n_k}$  is a set of mutually exclusive and exhaustive data interpretations. Due to the total probability theorem, the corresponding likelihood function is thus given by:

$$\ell(\mathbf{x}_k; Z_k, n_k) = \sum_{i_k=0}^{n_k} p(Z_k, n_k | i_k, \mathbf{x}_k) p(i_k | \mathbf{x}_k) \quad (36)$$

$$= \sum_{i_k=0}^{n_k} p(Z_k | n_k, i_k, \mathbf{x}_k) p(n_k | i_k, \mathbf{x}_k) p(i_k | \mathbf{x}_k) \quad (37)$$

$$= |\text{FoV}|^{-n_k} p_F(n_k) (1 - P_D) \\ + |\text{FoV}|^{-(n_k-1)} p_F(n_k - 1) \left[ \frac{1}{n_k} P_D \right] \sum_{i_k=1}^{n_k} \mathcal{N}(\mathbf{z}_k^{i_k}; \mathbf{H}_k^{i_k} \mathbf{x}_k, \mathbf{R}_k^{i_k}) \quad (38)$$

$$\propto (1 - P_D) \rho_F + P_D \sum_{i_k=1}^{n_k} \mathcal{N}(\mathbf{z}_k^{i_k}; \mathbf{H}_k^{i_k} \mathbf{x}_k, \mathbf{R}_k^{i_k}), \quad (39)$$

where we assumed a constant detection probability  $P_D$  and false returns equally distributed in the field of view  $|\text{FoV}|$  and Poisson distributed in number; i.e. the probability of having  $n$  false returns is given by  $p_F(n) = \frac{\rho_F |\text{FoV}|^n}{n!} e^{-\rho_F |\text{FoV}|}$  with a spatial false return density  $\rho_F$  and  $|\text{FoV}|$  denoting the volume of the field of view. See [22] for a more detailed discussion. According to the Equation 1, this likelihood function implies that  $p(\mathbf{x}_k | \mathcal{Z}^k)$  becomes a Gaussian mixture, a weighted sum of Gaussians, whose parameters are obtained by exploiting the product formula (9).

#### D. Gaussian Mixtures and Multiple Hypothesis Tracking

In many applications, such as group target tracking with possibly unresolved measurements [26], in ground moving target tracking with STAP radar [21], or target tracking with a phasedarray radar in the presence of jamming [10], the sensor model is described by a likelihood function of the type  $\ell(\mathbf{x}_k; Z_k, n_k) \propto \sum_{i_k=0}^{n_k} p(Z_k, n_k | i_k, \mathbf{x}_k) p(i_k | \mathbf{x}_k)$  [20]. Such likelihood functions, which are essentially characterized by taking different data interpretation hypotheses  $i_k$  into account, are the basis for Multiple Hypothesis Tracking algorithms (MHT, see [5]). In this context, each mixture component of the pdfs that result from Bayes' Rule and a Gaussian mixture prediction,

$$p(\mathbf{x}_k | \mathcal{Z}^k) = \sum_{\mathbf{i}_k} p_{k|k}^{\mathbf{i}_k} \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k}^{\mathbf{i}_k}, \mathbf{P}_{k|k}^{\mathbf{i}_k}), \quad (40)$$

represents a track hypothesis, which is characterized by a sequence of data interpretation hypotheses  $\mathbf{i}_k = (i_k, i_{k-1}, \dots, \dots)$ , i.e. data interpretation history.

The structure of a Gaussian mixture for  $p(\mathbf{x}_k | \mathcal{Z}^k)$  also occurs if an IMM prediction  $p(\mathbf{x}_k | \mathcal{Z}^{k-1})$  (see previous subsection) is updated by using a Gaussian likelihood function  $\ell(\mathbf{x}_k; \mathbf{z}_k, \mathbf{H}_k, \mathbf{R}_k) = \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k)$  according to Equation 1 and the product formula (Equation 9):

$$p(\mathbf{x}_k | \mathcal{Z}^k) = \frac{\sum_{\mathbf{j}_k} p_{k|k-1}^{\mathbf{j}_k} \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k) \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{\mathbf{j}_k}, \mathbf{P}_{k|k-1}^{\mathbf{j}_k})}{\sum_{\mathbf{j}_k} p_{k|k-1}^{\mathbf{j}_k} \int d\mathbf{x}_k \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_k, \mathbf{R}_k) \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{\mathbf{j}_k}, \mathbf{P}_{k|k-1}^{\mathbf{j}_k})} \quad (41)$$

$$= \sum_{\mathbf{j}_k} p_{k|k}^{\mathbf{j}_k} \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k}^{\mathbf{j}_k}, \mathbf{P}_{k|k}^{\mathbf{j}_k}), \quad (42)$$

where the mixture parameters  $p_{k|k}^{\mathbf{j}_k}$ ,  $\mathbf{x}_{k|k}^{\mathbf{j}_k}$ , and  $\mathbf{P}_{k|k}^{\mathbf{j}_k}$  are given by:

$$p_{k|k}^{\mathbf{j}_k} = \frac{p_{k|k-1}^{\mathbf{j}_k} \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_{k|k-1}^{\mathbf{j}_k}, \mathbf{S}_{k|k-1}^{\mathbf{j}_k})}{\sum_{\mathbf{j}_k} p_{k|k-1}^{\mathbf{j}_k} \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k \mathbf{x}_{k|k-1}^{\mathbf{j}_k}, \mathbf{S}_{k|k-1}^{\mathbf{j}_k})} \quad (43)$$

$$\mathbf{x}_{k|k}^{\mathbf{j}_k} = \mathbf{x}_{k|k-1}^{\mathbf{j}_k} + \mathbf{W}_{k|k-1}^{\mathbf{j}_k} (\mathbf{z}_k - \mathbf{H}_k \mathbf{x}_{k|k-1}^{\mathbf{j}_k}) \quad (44)$$

$$\mathbf{P}_{k|k}^{\mathbf{j}_k} = \mathbf{P}_{k|k-1}^{\mathbf{j}_k} - \mathbf{W}_{k|k-1}^{\mathbf{j}_k} \mathbf{S}_{k|k-1}^{\mathbf{j}_k} \mathbf{W}_{k|k-1}^{\mathbf{j}_k \top} \quad (45)$$

with the standard Kalman Gain and Innovation Covariance matrices

$$\mathbf{W}_{k|k-1}^{\mathbf{j}_k} = \mathbf{P}_{k|k-1}^{\mathbf{j}_k} \mathbf{H}_k^\top \mathbf{S}_{k|k-1}^{\mathbf{j}_k}{}^{-1} \quad (46)$$

$$\mathbf{S}_{k|k-1}^{\mathbf{j}_k} = \mathbf{H}_k^\top \mathbf{P}_{k|k-1}^{\mathbf{j}_k} \mathbf{H}_k + \mathbf{R}_k \quad (47)$$

IMM filtering may thus be considered as a multiple ‘model hypotheses’ tracking method. Also combined IMM-MHT-approaches are discussed in the literature, e.g. [23]. See [34], [35] for an alternative treatment of the multiple hypothesis, multiple model tracking problem.

#### E. Summary and Realization Aspects

A Bayesian tracking algorithm is an iterative updating scheme for conditional probability density functions  $p(\mathbf{x}_i | \mathcal{Z}^k)$  representing all available knowledge on the kinematical state vectors  $\mathbf{x}_i$  of the objects to be tracked at discrete instants of time  $t_i$ . The pdfs are conditioned by both the sensor data  $\mathcal{Z}^k$  accumulated up to some time  $t_k$ , typically the current scan time, and by available context information, such as sensor characteristics, object dynamics, environments, topographical maps, tactical rules. Depending on the time  $t_i$  at which estimates for the state vectors  $\mathbf{x}_i$  are required, the related estimation process is referred to as prediction ( $t_i > t_k$ ) and filtering ( $t_i = t_k$ ). In the following the iterative calculation is illustrated schematically:



$$\begin{array}{lcl}
\text{prediction:} & p(\mathbf{x}_{k-1} | \mathcal{Z}^{k-1}) & \xrightarrow[\text{kinematic constraints}]{\text{dynamics model}} p(\mathbf{x}_k | \mathcal{Z}^{k-1}) \\
\text{filtering:} & p(\mathbf{x}_k | \mathcal{Z}^{k-1}) & \xrightarrow[\text{sensor model}]{\text{current sensor data}} p(\mathbf{x}_k | \mathcal{Z}^k)
\end{array} \tag{48}$$

Under more realistic conditions, the pdfs have the structure of finite mixtures, i.e. weighted sums of individual densities that assume particular data interpretations or model hypotheses to be true. This structure is a direct consequence of the uncertain origin of the sensor data and/or of the uncertainty related to the underlying system dynamics. Provided the densities  $p(\mathbf{x}_i | \mathcal{Z}^k)$  are calculated correctly, optimal estimators can be derived related to various risk functions adapted to the applications. See [19] for a generalization of this Bayesian scheme to extended objects and object clusters and to retrodiction [23].

Due to the uncertain origin of the sensor data, naively applied Bayesian tracking leads to memory explosion. The number of components in the mixture densities  $p(\mathbf{x}_k | \mathcal{Z}^k)$  exponentially grow at each step. Suboptimal approximation techniques are therefore inevitable in any practical realization. Fortunately, in many applications, the densities resulting from prediction and filtering are characterized by a finite number of modes that may be fluctuating and even large for a while, but does not explosively grow. This is the rationale for adaptive approximation methods that keep the number of mixture components under control without disturbing the density iteration too seriously [12], [32]. In other words, the densities can often be approximated by mixtures with (far) less components (e.g. by merging of similar and pruning of irrelevant mixture components). Provided the relevant features of the densities are preserved, the resulting suboptimal algorithms are expected to be close to optimal Bayesian filtering. For dealing with non-linearities ‘extended’ or ‘unscented’ Kalman filtering (EKF [2], UKF [14]) or particle filtering (PF [31]) can be used.

### 3. Example: tracking-driven phased-array radar management

Resource management for a multi-functional radar certainly depends on the particular application considered. We here discuss track maintenance for ground-based air surveillance while minimizing the allocation time and energy required. The track accuracy is important only insofar as stable tracks are guaranteed. Track initiation or implementation issues are not addressed here. To make the benefits of IMM modeling and amplitude information clearly visible, false detections (clutter, electronic counter measures), data association conflicts, or possibly unresolved measurements were excluded. Nevertheless, their impact might well be incorporated into the general Bayesian framework [16].

#### A. Sensor Modeling for Phased-array Radar

In phased-array radar tracking, additional sensor information can be acquired when needed. Before each “radar resource allocation” [7], certain radar parameters must be selected by the tracking system depending on the current lack of information. We here consider the *object revisit time*  $t_k$ , the current *beam position*  $\mathbf{b}_k$ , i.e. a unit vector pointing into the direction where radar energy is to be transmitted, and the transmitted *energy per dwell*  $e_k$ . Other radar parameters (detection threshold  $\lambda_D$ , radar beam width  $B$ ) are assumed to be constant. After processing the skin echo produced by the illuminated object, the resource allocation  $R_k$  at time  $t_k$  results in measurements of direction cosines of the object and the object range,  $\mathbf{z}_k =$

$(\bar{u}_k, \bar{v}_k, \bar{r}_k)$ , along with the signal amplitude  $a_k$ . A single dwell may be insufficient for object detection and subsequent fine localization. Let  $n_{B_k}$  denote the number of dwells needed for a successful detection and  $B_k = \{\mathbf{b}_k^i\}_{i=1}^{n_{B_k}}$  the set of the corresponding beam positions. Each radar allocation is thus characterized by the tuple  $R_k = (t_k, B_k, n_{B_k}, e_k, \mathbf{z}_k, a_k)$ . The sequence of successive allocations is denoted by  $\mathcal{R}^k = \{R_k, \mathcal{R}^{k-1}\}$ .

1) *Radar Cross Section Fluctuations*: The instantaneous radar cross section  $\sigma_k$  of realistic objects strongly depends on the radar frequency used and the current aspect angle. For this reason, statistical models are used for describing the backscattering properties of the objects. In many practical cases,  $\sigma_k$  is described by gamma-densities,

$$p(\sigma_k | \bar{\sigma}, m) = \mathcal{G}_m(\sigma_k; \bar{\sigma}, m) \quad (49)$$

$$= \frac{(m/\bar{\sigma})^m}{\Gamma(m)} \sigma_k^{m-1} e^{-\sigma_k m/\bar{\sigma}}. \quad (50)$$

In this equation  $\bar{\sigma}$  denotes the mean RCS of the object that is usually unknown, but constant in time and characteristic of a certain class of objects, while the parameter  $m$  denotes the number of "degrees of freedom". The individual samples  $\sigma_k$  are assumed to be statistically independent for subsequent dwells (guaranteed by frequency decorrelation, e.g.). The cases  $m = 1, 2$  are referred to as Swerling-I and -III fluctuations [11].

Let the instantaneous object signal  $\mathbf{v}_k = (v_1, v_2)$  with the two orthogonal signal components  $v_1$  and  $v_2$  be additively corrupted by Gaussian noise with variance  $\sigma_n^2$  according to the standard modeling assumptions [11]. Since the signal components are assumed to be statistically independent, the pdf of the resulting sensor signal  $\mathbf{s}_k = (s_1, s_2)$  is

$$p(\mathbf{s}_k | \mathbf{v}_k) = \mathcal{N}(s_1; v_1, \sigma_n^2) \mathcal{N}(s_2; v_2, \sigma_n^2). \quad (51)$$

The normalized scalar quantity  $a_k^2 = (s_1^2 + s_2^2)/2\sigma_n^2$ , derived from  $\mathbf{s}_k$ , is thus Rice-distributed [11]:  $p(a_k^2 | \text{sn}_k) = e^{-a_k^2 - \text{sn}_k} I_0(2a_k \sqrt{\text{sn}_k})$  with  $\text{sn}_k = (v_1^2 + v_2^2)/2\sigma_n^2$ . Hence,  $\text{sn}_k$  denotes the instantaneous signal-to-noise ratio of the object being proportional to the instantaneous radar cross section  $\sigma_k$ . The expectation value of  $a_k^2$  with respect to  $p(a_k^2 | \text{sn}_k)$  is given by  $\mathbb{E}[a_k^2] = 1 + \text{sn}_k$ . According to the normalization chosen, pure noise ( $\text{sn}_k = 0$ ) has thus unit power. Due to the RCS model previously discussed,  $\text{sn}_k$  is gamma-distributed with the mean SN:  $p(\text{sn}_k | \text{SN}) = \mathcal{G}_m(\text{sn}_k; \text{SN}, m)$ . The conditional density of  $a_k^2$  given SN is thus obtained by calculating:

$$p(a_k^2 | \text{SN}) = \int_0^\infty d\text{sn}_k p(a_k^2 | \text{sn}_k) p(\text{sn}_k | \text{SN}). \quad (52)$$

The integration can be carried out (see [1], e.g.) yielding:

$$p(a_k^2 | \text{SN}) = \left(\frac{m+\text{SN}}{m}\right)^{-m} e^{-ma_k^2/(m+\text{SN})} L_{m-1}\left(\frac{-a_k^2 \text{SN}}{m+\text{SN}}\right) \quad (53)$$

where  $L_{m-1}$  denotes the Laguerre polynomials. For Swerling-I/III these polynomials are given by:  $L_0(-x) = 1$ ,  $L_1(-x) = 1+x$ . Obviously,  $p(a_k^2 | \text{SN})$  can be interpreted as a gamma mixture with the expectation value  $\mathbb{E}[a_k^2] = 1 + \text{SN}$ .

2) *Mean Received Signal-to-Noise Ratio*: Any sensor model for phased-array radar tracking has to provide a functional relationship between the expected signal-to-noise ratio  $\text{SN}_k$  at the revisit time  $t_k$ , the sensor parameters considered (here: transmitted energy, beam position) and the relevant object parameters (mean RCS, object position). With a Gaussian beam form model [17], well proven in applications, the radar range equation (see [11], e.g.), we assume:

$$\text{SN}_k = \text{SN}_0 \left( \frac{\bar{\sigma}}{\sigma_0} \right) \left( \frac{e_k}{e_0} \right) \left( \frac{r_k}{r_0} \right)^{-4} e^{-2\Delta b_k} \quad (54)$$

$$\text{with } \Delta b_k = |\mathbf{d}_k - \mathbf{b}_k|^2 / B^2. \quad (55)$$

$r_k$  is the actual object range at time  $t_k$ , while  $\mathbf{d}_k = (u_k, v_k)^\top$  denotes the related direction cosines. With the beam position  $\mathbf{b}_k = (b_k^u, b_k^v)^\top$  and the (one-sided) beam width  $B$ ,  $\Delta b_k$  is a measure of relative beam positioning error. The radar parameter  $\text{SN}_0$  is the expected mean signal-to-noise ratio of a object with a standard mean cross section  $\bar{\sigma}_0$  at a reference range  $r_0$  that is directly ( $\Delta b_k = 0$ ) illuminated by the beam with the energy  $e_0$ . Due to the functional relationship stated in Equation 53, the signal strength  $a_k^2$  can be interpreted as a measurement of  $\bar{\sigma}$ .

3) *Detection and Measurement Process*: A detection is assumed if the received signal strength exceeds a certain detection threshold:  $a_k^2 > \lambda_D$ . For a given  $m$  in the fluctuation model (Equation 50), the detection probability  $P_D$  is a function of SN and  $\lambda_D$ :

$$P_D(\text{SN}, \lambda_D, m) = \int_{\lambda_D}^{\infty} da_k^2 p(a_k^2 | \text{SN}). \quad (56)$$

The false alarm probability  $P_F$  is analogously obtained:  $P_F(\lambda_D) = P_D(0, \lambda_D, m) = e^{-\lambda_D}$ . Integration results in explicit expressions for  $P_D$  [11]. For Swerling-I/III fluctuations, we obtain:

$$P_D^I(\text{SN}, \lambda_D) = e^{-\frac{\lambda_D}{1+\text{SN}}} = P_F^{\frac{1}{1+\text{SN}}} \quad (57)$$

$$P_D^{III}(\text{SN}, \lambda_D) = e^{-\frac{\lambda_D}{1+\text{SN}/2}} \left( 1 + \frac{(\text{SN}/2)\lambda_D}{(1+\text{SN}/2)^2} \right). \quad (58)$$

For object tracking  $a_k^2$  is available after a detection, i.e.  $a_k^2 > \lambda_D$ . We thus need the conditional density:

$$p(a_k^2 | a_k^2 > \lambda_D, \text{SN}, m) = \begin{cases} \frac{p(a_k^2 | \text{SN})}{P_D(\text{SN}, \lambda_D, m)} & \text{for } a_k^2 > \lambda_D \\ 0 & \text{else} \end{cases} \quad (59)$$

For strong objects we can assume  $\text{SN} \approx 1 + \text{SN} \approx \dots \approx m + \text{SN}$  and thus approximately obtain:  $p(a_k^2 | \text{SN}) \approx \left( \frac{\text{SN}}{m} \right)^{-m} e^{-ma_k^2/\text{SN}} L_{m-1}(-a_k^2)$ , which is similar to the expression in Equation 53. On the other hand, let the detection probability for  $m \neq 1$  be approximately given by:  $P_D(\text{SN}, \lambda_D, m) \approx P_D^I(\text{SN}, \lambda_D)$  (i.e. by a Swerling-I-model). We can therefore write:  $p(a_k^2 | a_k^2 > \lambda_D, \text{SN}, m) \approx S_m(a_k^2; \text{SN}, m)$  with:

$$\mathcal{S}_m(a_k^2; \text{SN}, m) = \begin{cases} \left(\frac{\text{SN}}{m}\right)^{-m} e^{-(ma_k^2 + \lambda_D)/\text{SN}} L_{m-1}(-a_k^2) & \text{for } a_k^2 > \lambda_D \\ 0 & \text{else.} \end{cases} \quad (60)$$

Let us furthermore assume that monopulse localization after detection result in bias-free measurements  $u'_k, v'_k$  of the direction cosines and range with Gaussian measurement errors. According to [11], the standard deviations  $\sigma_k^{u,v}$  depend on the beam width  $B$  and the instantaneous  $\text{sn}_k$  in the following manner:  $\sigma_k^{u,v} \propto B/\sqrt{\text{sn}_k} \approx B/\sqrt{a_k^2 - 1}$ . Since  $\text{sn}_k$  is unknown, in the last approximation  $a_k^2$  is used as a bias-free estimate of  $\text{sn}_k$  ( $\mathbb{E}[a_k^2] = 1 + \text{sn}_k$ ). The range error is assumed to be Gaussian with a constant standard deviation  $\sigma_r$ . Evidently, this model of the measurement process does not depend on the RCS fluctuation model.

### B. Bayesian Tracking Algorithms Revisited

According to the previous discussion, object tracking is an iterative updating scheme for conditional probability densities  $p(\mathbf{x}_k | \mathcal{R}^k)$  that describe the current object state  $\mathbf{x}_k$  given all available resource allocations  $\mathcal{R}^k$  and the underlying a priori information in terms of statistical models. The processing of each new measurement  $\mathbf{z}_k$  via Bayes' Rule establishes a recursive relation between the densities at two consecutive revisit times (a prediction step followed by filtering).

$$p(\mathbf{x}_k | \mathcal{R}^{k-1}) = \sum_{\mathbf{j}_k} p_{k|k-1}^{\mathbf{j}_k} \mathcal{N}(\mathbf{x}_k; \mathbf{x}_{k|k-1}^{\mathbf{j}_k}, \mathbf{P}_{k|k-1}^{\mathbf{j}_k}) \quad (61)$$

with  $\mathbf{j}_k = (j_k, \dots, j_{k-n+1})$  denoting a particular *model history*, i.e. a sequence of possible hypotheses regarding the object dynamics model from a certain observation at time  $t_{k-n+1}$  up to the most recent measurement at time  $t_k$  ("n scans back"). In the case of a single dynamics model ( $r = 1$ ), the prediction densities  $p(\mathbf{x}_k | \mathcal{R}^{k-1})$  are strictly given by Gaussians (standard Kalman prediction). For  $n = 1$ ,  $p(\mathbf{x}_k | \mathcal{R}^{k-1})$  is approximated by a mixture with  $r$  components according to the  $r$  dynamics models used. GPB2 and standard IMM algorithms are possible realizations of this scheme [3]. For standard IMM, the approximations are made after the prediction, but before the filtering step, while for GPB2 they are applied after the filtering step. Hence, GPB2 requires more computational effort. For details see [3].

2) *Processing of Signal Strength Information*: Let us treat the normalized mean RCS of the object,  $s_k = \bar{\sigma}_k / \bar{\sigma}_0$ , as an additional component of the state vector. Since the signal strength after a detection occurred may be viewed as a measurement of  $s_k$ , let us consider the augmented conditional density

$$p(\mathbf{x}_k, s_k | \mathcal{R}^k) = p(s_k | \mathbf{x}_k, \mathcal{R}^k) p(\mathbf{x}_k | \mathcal{R}^k). \quad (62)$$

The calculation of  $p(\mathbf{x}_k | \mathcal{R}^k)$  was discussed in section 2. For the remaining density  $p(s_k | \mathbf{x}_k, \mathcal{R}^k)$ , an application of Bayes' Rule yields up to a normalizing constant:

$$p(s_k | \mathbf{x}_k, a_k^2, \mathcal{R}^{k-1}) \propto \mathcal{S}_m(a_k^2; \text{SN}, m) p(s_k | x_k, R^{k-1}). \quad (63)$$

Let us furthermore assume that  $p(s_k | \mathbf{x}_k, \mathcal{R}^{k-1})$  are given by inverse gamma densities,

$$p(s_k | \mathbf{x}_k, \mathcal{R}^{k-1}) = \mathcal{I}_{\mu_{k|k-1}}(s_k; \hat{s}_{k|k-1}, \mu_{k|k-1}), \quad (64)$$

which are defined by:

$$\mathcal{I}_{\mu}(s; \hat{s}, \mu) = [((\mu - 1)\hat{s})^{\mu} / \Gamma(\mu)] s^{-\mu-1} e^{-\frac{(\mu-1)\hat{s}}{s}}, \quad (65)$$

where  $\hat{s}$  is the expectation of this density,  $\hat{s} = \mathbb{E}[s] > 0$ ,  $\mu$  a parameter  $\mu > 1$ . For  $\mu > 2$ , the related variance exists:  $\mathbb{V}[s] = \hat{s}^2 / (\mu - 2)$ . This class of densities is invariant under the successive application of Bayes Rule according to Equation 63, since up to normalization we obtain:

$$\mathcal{S}_m(a_k^2; \text{SN}, m) \mathcal{I}_{\mu_{k|k-1}}(s_k; \hat{s}_{k|k-1}, \mu_{k|k-1}) \quad (66)$$

$$\propto \alpha_k^{-m} s_k^{-\mu_{k|k-1}-m-1} \exp\left(-\frac{(\mu_{k|k-1}-1)\hat{s}_{k|k-1} + \frac{ma_k^2 + \lambda_D}{\alpha_k}}{s_k}\right) \quad (67)$$

$$\propto \mathcal{I}_{\mu_k}(s_k; \hat{s}_k, \mu_k), \quad (68)$$

where the parameters  $\alpha_k$ ,  $\hat{s}_k$  and  $\mu_k$  are given by:

$$\alpha_k = \text{SN}_0 \left(\frac{e_k}{e_0}\right) \left(\frac{r_k}{r_0}\right)^{-4} e^{-2\Delta b_k}, \quad (69)$$

$$\hat{s}_k = \frac{\mu_{k|k-1}-1}{\mu_{k|k-1}+m-1} \hat{s}_{k|k-1} + \frac{(ma_k^2 + \lambda_D)/\alpha_k}{\mu_{k|k-1}+m-1} \quad (70)$$

$$\mu_k = \mu_{k|k-1} + m. \quad (71)$$

With reference to  $s_k$  the density  $\mathcal{I}_{\mu_k}(s_k; \hat{s}_k, \mu_k)$  is correctly normalized. Evidently,  $\alpha_k$  depends on the object position ( $\alpha_k = \alpha_k(r_k, u_k, v_k)$ ). In order to preserve the factorization of  $p(\mathbf{x}_k, s_k | \mathcal{R}^k)$  in a normal mixture related to the kinematic properties of the object  $\mathbf{x}_k$  and an inverse gamma density related to its RCS  $s_k$ , we use the approximation:

$$\alpha_k \approx \left(\frac{e_k}{e_0}\right) \left(\frac{\hat{r}_k}{r_0}\right)^{-4} e^{-2\{(\hat{\mathbf{u}}_k - b_k^u)^2 + (\hat{\mathbf{v}}_k - b_k^v)^2\} / B^2}, \quad (72)$$

where  $\hat{\mathbf{r}}_k$ ,  $\hat{\mathbf{u}}_k$ ,  $\hat{\mathbf{v}}_k$  are the MMSE estimates for  $r_k$ ,  $u_k$  and  $v_k$  derived from  $p(\mathbf{x}_k | \mathcal{R}^k)$ . Hence,  $\alpha_k$  compensates both the estimated positioning error of the radar beam and the propagation loss due to the radar equation. Assuming  $s_k$  to be constant, we have  $\mathcal{I}_{\mu_{k|k}}(s_k; \hat{s}_k, \mu_k) = \mathcal{I}_{\mu_{k|k-1}}(s_k; \hat{s}_{k-1}, \mu_{k|k-1})$ . In principle, a dynamics model describing temporal changes of the radar cross section might be introduced.

### C. Adaptive Bayesian Sensor Management

The tracking results are essential for adaptive radar revisit time control, the selection of the transmitted radar energy, and the design of intelligent algorithms for local search.

1) *Adaptive Radar Revisit Time Control*: The time  $t_k$  when a radar allocation  $R_k$  should take place is determined by the current lack of information conveniently described [17] by the error covariance matrix  $\mathbf{P}_{k|k-1}$  of the predicted state estimate  $\mathbf{x}_{k|k-1}$ . Since  $p(\mathbf{x}_k | \mathcal{R}_{k-1})$  is a normal mixture,  $\mathbf{x}_{k|k-1}$  and  $\mathbf{P}_{k|k-1}$  are given by:

$$\mathbf{x}_{k|k-1} = \sum_{\mathbf{j}_k} p_{k|k-1}^{\mathbf{j}_k} \mathbf{x}_{k|k-1}^{\mathbf{j}_k} \quad (73)$$

$$\mathbf{P}_{k|k-1} = \sum_{\mathbf{j}_k} p_{k|k-1}^{\mathbf{j}_k} (\mathbf{P}_{k|k-1}^{\mathbf{j}_k} + (\mathbf{x}_{k|k-1}^{\mathbf{j}_k} - \mathbf{x}_{k|k-1})(\mathbf{x}_{k|k-1}^{\mathbf{j}_k} - \mathbf{x}_{k|k-1})^\top). \quad (74)$$

The covariance matrix  $\mathbf{P}_{k|k-1}^{\mathbf{j}_k}$  of the individual mixture components grow the faster in time the more often maneuvers are assumed in the corresponding model histories. This has an impact on the total covariance matrix  $\mathbf{P}_{k|k-1}$  according to the corresponding weighting factors  $p_{k|k-1}^{\mathbf{j}_k}$ . In addition,  $\mathbf{P}_{k|k-1}$  is "broadened" by the positively definite spread terms  $(\mathbf{x}_{k|k-1}^{\mathbf{j}_k} - \mathbf{x}_{k|k-1})(\mathbf{x}_{k|k-1}^{\mathbf{j}_k} - \mathbf{x}_{k|k-1})^\top$ . Obviously, the adaptive IMM modeling affects  $\mathbf{P}_{k|k-1}$  in a rather complicated way.

A scalar measure of the information deficit is provided, e.g., by the largest eigenvalue of the covariance matrix of the predicted object direction (in terms of  $u, v$ ). Let it be denoted by  $G_{k|k-1}$ . A track update is allocated when the  $G_{k|k-1}$  exceeds a predetermined proportion of the squared radar beam width  $B$ :

$$G_{k|k-1} > (v_0 B)^2. \quad (75)$$

The *relative track accuracy*  $v_0$  introduced by this criterion is a measure of the minimum track quality required and a parameter to be optimized. In many practical applications,  $v_0 = 0.3$  is a reasonable choice [17].

2) *Transmitted Radar Energy Selection*: In view of the tracking system, the sensor performance is mainly characterized by the signal-to-noise ratio that determines both, the detection probability and the measurement error. By suitably choosing the transmitted energy per dwell  $e_k$ , the expected signal-to-noise ratio  $\text{SN}_{k|k-1}$  can be kept constant during tracking. Besides  $v_0$ ,  $\text{SN}_{k|k-1}$  is an additional parameter subject to optimization. Since  $v_0$  may be viewed as a measure of the beam positioning error, the energy  $e_k$  at time  $t_k$  is defined by this condition (Equation 54):

$$\text{SN}_{k|k-1} \stackrel{!}{=} \text{const.} \quad (76)$$

$$\rightarrow \frac{e_k}{e_0} = \left( \frac{\text{SN}_0}{\text{SN}_{k|k-1}} \right) \left( \frac{\bar{\sigma}_0}{\bar{\sigma}} \right) \left( \frac{\hat{r}_{k|k-1}}{r_0} \right)^4 e^{2v_0^2}. \quad (77)$$

By this particular choice, the influence of the radar range equation is compensated (at least for a certain range interval). For the mean radar cross section  $\bar{\sigma}$  either a worst-case assumption or estimates from object amplitude information can be used. The track quality  $v_0$  also affects the transmitted energy. As a side effect of this choice, the standard deviations  $\sigma_k^{u,v}$  of the  $u, v$ -measurements are kept constant on an average.

3) *Bayesian Local Search Procedures*: Intelligent algorithms for beam positioning and local search are crucial for IMM-type phased-array tracking. Overly simple strategies may easily destroy the benefits of the adaptive dynamics model, because track loss immediately after a model switch can easily occur. To avoid this phenomenon, we adapt the optimal approach based on the predicted densities  $p(\mathbf{x}_k | \mathcal{R}^{k-1})$  proposed in [17] to IMM tracking [24].

1. The beam position  $\mathbf{b}_k^1$  of the first dwell at time  $t_k$  is simply given by the predicted direction  $\mathbf{d}_k|_{k-1}$  to be derived from the predicted density function  $p(\mathbf{x}_k | \mathcal{R}^{k-1})$ .
2. If no detection occurs in the first dwell, this very result provides useful information on the target. We thus have to calculate the conditional density of the target state given the event  $\neg D_k^1$ : 'no detection at time  $t_k$  in the direction  $\mathbf{b}_k^1$ '.
3. An application of Bayes' Rule directly yields:

$$p(\mathbf{d}_k | \neg D_k^1, \mathcal{R}^{k-1}) \propto (1 - P_D(\mathbf{d}_k; \mathbf{b}_k^1)) p(\mathbf{d}_k | \mathcal{R}^{k-1}) \quad (78)$$

up to a normalizing factor. In this expression, the detection probability  $P_D$  depends on the expected SN (Equation 54) and thus on the current beam and target position  $\mathbf{b}_k, \mathbf{d}_k$ .

4. The two dimensional density  $p(\mathbf{d}_k | \neg D_k^1, \mathcal{R}^{k-1})$  can easily be calculated on a grid. The beam position for the next dwell is then simply provided by its maximum.
5. This computational scheme for Bayesian local search is repeated until a detection occurs. Since the maximum of the densities  $p(\mathbf{d}_k | \neg D_k^1, \neg D_k^2, \dots, \mathcal{R}^{k-1})$  is searched, the computation of the normalization integral is not required. Numerically efficient realizations are possible.

Alternatively,  $p(\mathbf{d}_k | \neg D_k^1, \mathcal{R}^{k-1})$  might be used for calculating the expected SN in a certain direction  $\mathbf{b}_k$ :

$$\text{SN}(\mathbf{b}_k) = \int d\mathbf{d}_k \text{SN}(\mathbf{b}_k, \mathbf{d}_k) p(\mathbf{d}_k | \neg D_k^1, \mathcal{R}^{k-1}).$$

Searching the maximum of  $\text{SN}(\mathbf{b}_k)$  results in a different local search strategy. In the examples considered below, however, no significant performance improvements were observed. Nevertheless, there might be applications where the maximization of  $\text{SN}(\mathbf{b}_k)$  is advantageous (e.g. for track recovery in case of intermittent operating modes).

This local search scheme exploits 'negative' evidence, as also here the lack of an expected measurement carries information on the current target position. We here in particular observe a direct impact on adaptive sensor management. Again, the prerequisite for dealing with negative evidence is an adequate sensor performance model. As in the case of resolution phenomena (section 2), the processing of negative sensor evidence implies mixture densities with possibly *negative* mixture coefficients, i.e. not each mixture component has a direct probabilistic interpretation. As the mixture coefficients sum up to one, the overall density nevertheless has a well-defined probabilistic meaning.

Figure 4 illustrates this scheme of Bayesian local search for a particular example. In Figure 4a the predicted pdf  $p(\mathbf{d}_k | \mathcal{R}^{k-1})$ , a mixture density, is shown for some time  $t_k$ . The target is expected to be in the bright region with high probability, the true target position being indicated by a green dot. The blue dot denotes the beam position of the next dwell. The related detection probability is 26%. However, no detection occurred during the first dwell

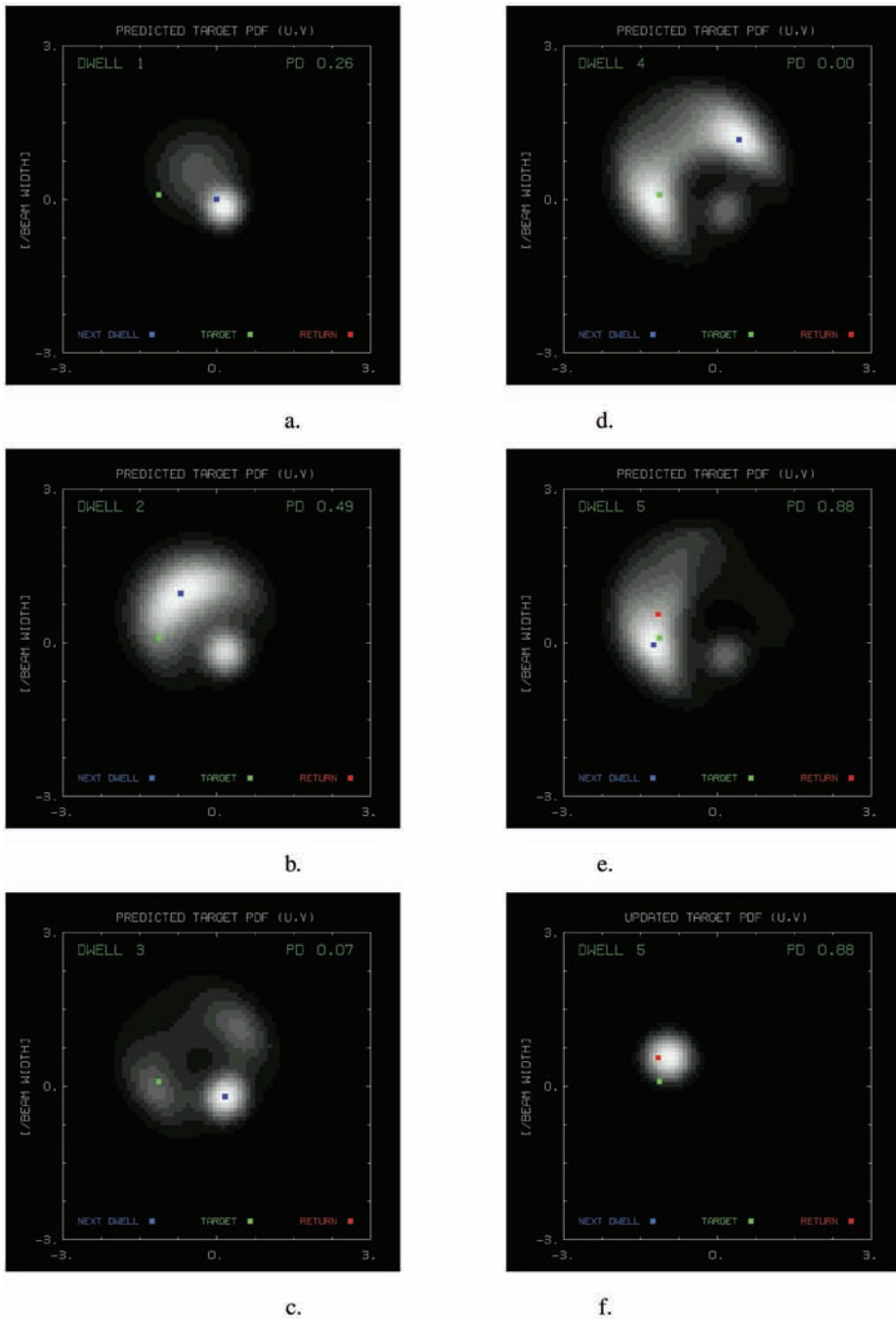


Fig. 4. Bayesian local search: five consecutive dwells



1. We thus calculate the conditional pdf  $p(\mathbf{d}_k | \neg D_k^1, \mathcal{R}^{k-1})$  given that event. As visible in Figure 4b, it differs significantly from  $p(\mathbf{d}_k | \mathcal{R}^{k-1})$ . The previous maximum decreased in height, while the global maximum is at a different location. Again no detection occurred; the resulting density  $p(\mathbf{d}_k | \neg D_k^1, \neg D_k^2, \mathcal{R}^{k-1})$  reflecting the two pieces of ‘negative’ evidence  $\neg D_k^1$  and  $\neg D_k^2$  is shown in Figure 4c. Now the search algorithm decides to look again near the position at dwell 1. Although wrong in this case, this does not seem to be unreasonable. In addition, two smaller local maxima appear that increase in size as in the next dwell also no detection occurred. According to Figure 4d the next decision is ambiguous. We finally obtain a decision which leads to success. The last picture shows the updated pdf (Figure 4f).

#### 4. Discussion of numerical simulation results

Simulation results provide hints as to what extent the total performance of multiple-object air surveillance by phased-array can be improved by using adaptive techniques for combined tracking and sensor control. The following four questions are addressed:

1. What resource savings (allocation time, energy) can be expected by using adaptive dynamics models?
2. How should the IMM dynamics modeling be designed (e.g. number of models, transition matrix)?
3. What energy savings can be expected by exploiting object amplitude information for sensor control?
4. Why is Bayesian local search important when adaptive dynamics models are used for revisit time control?

##### A. Discussion of Simulation Scenarios

In general we follow the parameter and threshold settings recommended in [17]. To exclude false alarms due to receiver noise, the false alarm probability is  $P_F = 10^{-4}$ . False returns due to clutter or ECM are not considered. The standard deviation of the measurement errors in object range is  $\sigma_r = 100$  m, while the the radar beam width is  $B = 1^\circ$ . We assume a minimum time interval of 20 ms between consecutive dwells on a particular object and statistically independent signal amplitudes (frequency decorrelation, e.g.). The reference range is set to  $r_0 = 80$  km.

1) *IMM Modeling Parameters:* Antenna coordinates (direction cosines, range) are used also for tracking; non-linearities introduced by these non-Cartesian coordinates are taken into account [16]. In each component  $u_k, v_k, r_k$  the state vector is given by position, speed, and acceleration. For the sake of simplicity, we consider a block diagonal system matrix defined by

$$\mathbf{F}_{k|k-1} = \begin{pmatrix} \mathbf{1} & \Delta t_k \mathbf{1} & \frac{1}{2} \Delta t_k^2 \mathbf{1} \\ \mathbf{0} & \mathbf{1} & \Delta t_k \mathbf{1} \\ \mathbf{0} & \mathbf{0} & e^{-\Delta t_k / \theta} \end{pmatrix} \quad (79)$$

$$\mathbf{D}_{k|k-1} = \Sigma^2 (1 - e^{-2\Delta t_k / \theta}) \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{pmatrix}, \quad (80)$$

with  $\Delta t_k = t_k - t_{k-1}$ . The maneuvering capability of the objects is thus characterized by two parameters: *maneuver correlation time*  $\theta$  and *acceleration width*  $\Sigma$ . For  $r = 2, 3$  we consider the parameter sets:

- $M_1$  (worst-case model):  $\Sigma_1 = 60 \text{ m/s}^2$ ,  $\theta_1 = 30 \text{ s}$
- $M_2$  (best-case model):  $\Sigma_2 = 1 \text{ m/s}^2$ ,  $\theta_2 = 10 \text{ s}$
- $M_3$  (medium-case model):  $\Sigma_3 = 30 \text{ m/s}^2$ ,  $\theta_3 = 30 \text{ s}$

The matrices of the model transition probabilities are given by:

$$(p_{ij})_{i=1,j=1}^2 = \begin{pmatrix} .8 & .1 \\ .2 & .9 \end{pmatrix}, \quad (p_{ij})_{i=1,j=1}^3 = \begin{pmatrix} .8 & .1 & .0 \\ .0 & .9 & .2 \\ .2 & .0 & .8 \end{pmatrix}. \quad (81)$$

We observed that the performance does not critically depend on the particular switching probabilities  $p_{ij}$  chosen. A detailed mismatch analysis, however, has not been performed. A track is considered to be lost if more than 50 dwells occur in the local search of if the beam positioning error  $\Delta b_k$  is greater than  $3B$ . We thus permit even a rather extensive local search that correspondingly burdens the total energy budget. In all simulations considered below (1000 runs) the relative frequency of track loss is less than 2%.

2) *Selected Benchmark Trajectories*: The horizontal projection of four standard benchmark trajectories (military cargo aircraft, medium bomber, fighter/attack aircraft, and anti-ship missile) is shown in Figure 5 along with representative kinematical characteristics such as acceleration (solid line), range (dashed), height (dotted), and speed (solid). They have been proposed in [8], [9] and cover a rather wide range of militarily relevant objects. The missile trajectory might serve to explore the performance limits of the algorithms. In principle, missiles can execute even stronger maneuvers. It is questionable, however, if for those objects and their individual missions the dynamics models discussed above remain applicable. All objects are tracked over a period of 180 s. The RCS fluctuations are described by a Swerling-III model. The mean cross sections significantly vary from object to object (4., 2., 1.2, .5 m<sup>2</sup>).

3) *Measures of Performance Considered*: The discussion is confined to a few intuitively clear and simple performance measures obtained by Monte-Carlo simulation (1000 runs). In general a single performance measure is not sufficient as there may exist applications where the transmitted energy is the limiting factor, while in a different scenario the number of radar allocations must be kept low.

The adaptivity becomes visible if the performance is evaluated as a function of the tracking time that can be compared with the kinematics of the individual trajectories (Figure 5). Here we used histograms with 100 cells. In particular we considered: the mean revisit intervals, the mean number of dwells for a successful update, the mean number of sensor allocations in total required for track maintenance, the mean energy spent for a successful allocation, the mean energy totally spent for track maintenance, and the mean RCS of the objects estimated during tracking.

Four tracking filters were compared: worst-case Kalman filter (KF), standard IMM filter with two or three models, respectively (S-IMM<sub>2,3</sub>), and IMM-MHT filtering with model histories of length  $n = 4$ . For IMM-MHT with  $n > 4$ , the performance characteristics change only slightly. We thus conclude that  $n = 4$  already provides a good approximation to optimal filtering (at least for the scenarios considered here). With reference to object amplitude information we considered three cases: 1) the object RCS  $\bar{\sigma}$  is known and used for energy management. 2) The mean RCS  $\bar{\sigma}$  is unknown and to be estimated during tracking. 3) A worst-case assumption is used for all objects ( $\bar{\sigma} = 0.5 \text{ m}^2$ ).

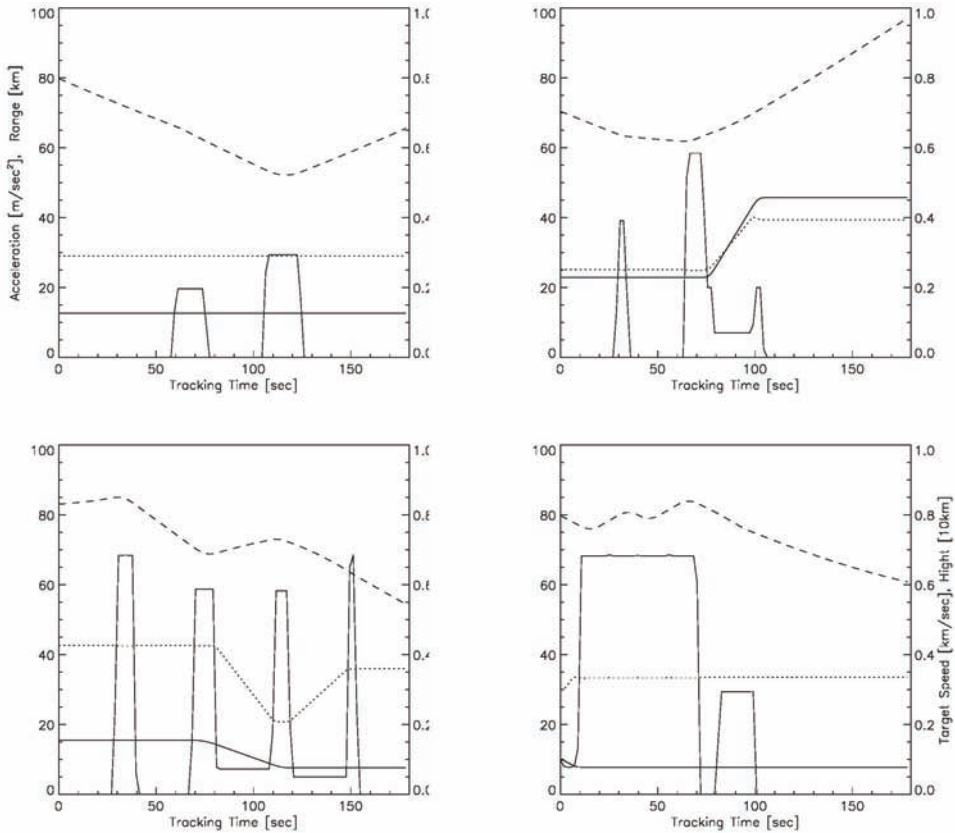


Fig. 5. Horizontal projections and kinematical quantities (military cargo aircraft, medium bomber, fighter/attack aircraft, and anti-ship missile)

### B. Remarks on IMM Modeling Design

Practically, the question arises how many models should be used in the IMM approach. In addition it must be clarified if each trajectory needs an individual modelling or if the same IMM modelling can be used without significant loss of performance. For the fighter scenario, a worst/best-case model should be appropriate at first sight. Trajectory 1 (Cargo Aircraft), however, shows that military objects can occur for which medium-case models are sufficient. To answer these questions, we used IMM with two ( $r = 2, M_1, M_2$ ) and three models ( $r = 3, M_1, M_2, M_3$ ), respectively, with  $v_0 = 0.3$ ,  $\text{SN}_{k|k-1} = 50$ . How these parameters affect the performance is discussed further below. Figure 6 shows the resulting mean revisit intervals for all trajectories. The kinematical object characteristics are clearly mirrored. We observed:

1. As expected, Kalman filtering ( $r = 1, M_1$ ) leads to constant revisit intervals that are comparable for all trajectories. This is no longer true for S-IMM. The resultant curves related to  $r = 2$  (solid) and  $r = 3$  (dashed) significantly differ from each other. The onset of maneuvers (Figure 5) strongly affects the mean update intervals and thus illustrates the adaptivity of the algorithm.

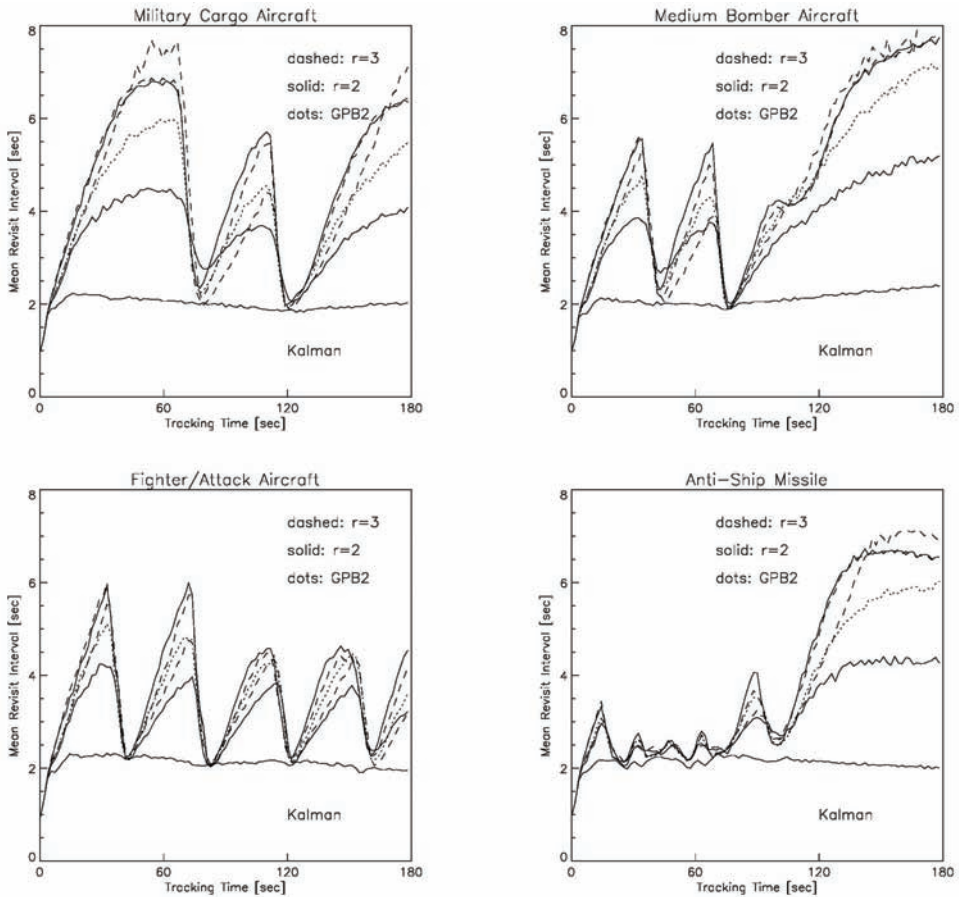


Fig. 6. Revisit intervals for different filters: military cargo aircraft, medium bomber, fighter/attack aircraft, and anti-ship missile

2. The difference between the cases  $r = 2, 3$  vanishes however, if IMM-MHT is used. If model histories are permitted (here  $n = 4$ ), it seems to be irrelevant if besides worst/bestcase assumption additional medium-case models are used. Even longer histories or further models ( $r > 3$ ) do not significantly improve the performance obtained with  $r = 2$  and  $n = 4$ . For a suitable (!) choice of the switching probabilities, the performance of S-IMM<sub>4</sub> approaches close to B-IMM<sub>2</sub>; for B-IMM<sub>4</sub> no improvement over B-IMM<sub>2</sub> was observed.
3. For the bomber and the fighter, S-IMM<sub>3</sub> ( $M_1, M_2, M_3$ ) outperforms S-IMM<sub>2</sub> ( $M_1, M_2$ ), in spite of the fact that for these trajectories only worst-case maneuvers occur and the medium-case model appeared to be unnecessary at first sight. The difference between  $r = 2$  and  $r = 3$ , however, is not as clear as for scenario 1 (cargo aircraft).
4. For the moderately maneuvering cargo aircraft, the question arises whether the performance can be improved by using a medium/best-case IMM modelling. We found

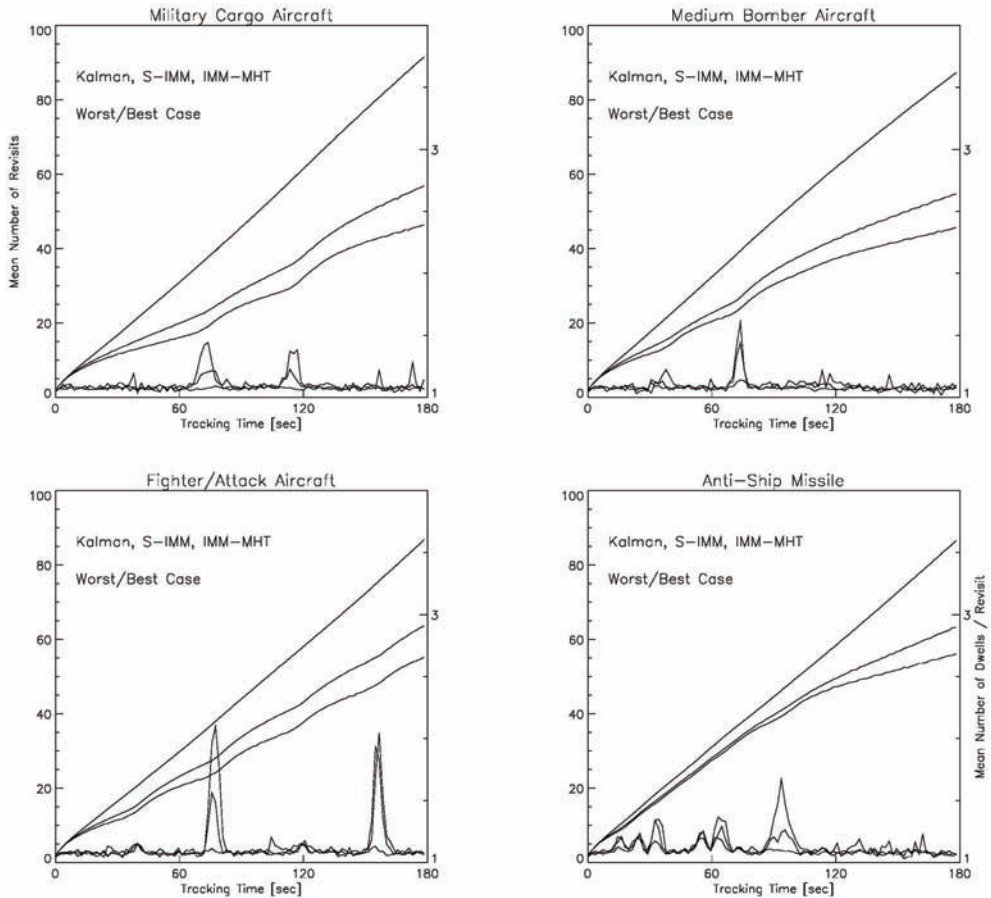


Fig. 7. Radar allocations for different filters: military cargo aircraft, medium bomber, fighter/attack aircraft, and anti-ship missile

that worst/best-case IMM-MHT and medium/best-case IMM-MHT differ, but not very much. This indicates that worst/best-case IMM-MHT has a more or less “universal” character, i.e. it does not critically depend of the scenario considered (at least within certain limits).

These observations indicate that the mixtures  $p(\mathbf{x}_k | \mathcal{R}^k)$  for  $n = 4$ ,  $r = 2$  have enough internal degrees of freedom to provide an adequate representation of the actual object behavior. Refined approximations by even more mixture components seem to be irrelevant for the trajectories considered. A rule of thumb: A worst/best-case analysis of the problem along with IMM-MHT seems to be sufficient to achieve a nearly optimal tracking performance. Obviously, for two dynamics models, reasonable and intuitive assumptions for the switching probabilities are easily obtained. IMM-MHT thus enables a more simplified dynamics modelling without significant loss of performance.

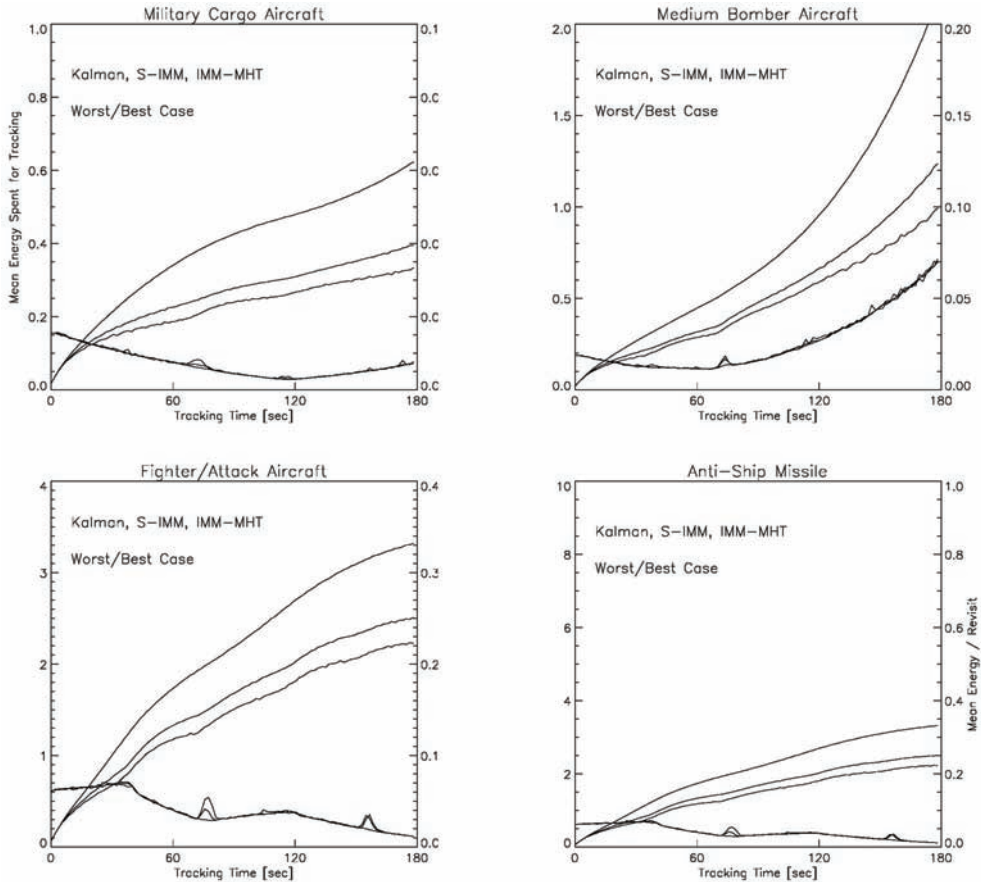


Fig. 8. Radar energy spent for different filters: military cargo aircraft, medium bomber, fighter/attack aircraft, and anti-ship missile

### C. Gain by IMM Modelling

To investigate the gain by adaptive dynamics models, let us for the present assume that the mean RCS of the object is known and used for energy management. Figure 7 shows the mean number of allocations required for track maintenance (KF, S-IMM<sub>2</sub>, B-IMM<sub>2</sub>). As expected, for KF the mean number of revisits linearly increases with increasing tracking time and is nearly the same for all trajectories. By adaptive dynamics modeling, however, the number of sensor allocations is reduced.

1. Compared with KF, IMM results in significant resource savings. There is an improvement by IMM-MHT over S-IMM; the difference, however, is less significant than between SIMM and KF. Besides simplified modeling assumptions, the practical use of IMM-MHT therefore consists in the exploration of the limiting bounds for performance improvements.
2. The largest gain is observed for the cargo aircraft and the bomber. In the case of the fighter aircraft, the allocations required are reduced by about 50 % compared with

worstcase Kalman filtering. Even during the 7 *g* weaving of the missile, some advantages of the IMM modeling can be observed.

Figure 7 shows the mean number of dwells per revisit. Up to peaks corresponding with the onset of maneuvers, it is constant and roughly equal for all filters and trajectories. The more adaptive the filter is, the higher the peaks are, i.e. the larger the revisit intervals can be during inertial flight. The peaks thus indicate that for abrupt maneuvers a local search might be required. This is the price to be paid for increased adaptivity. Evidently, intelligent algorithms for beam positioning and local search are essential for IMM phased-array tracking. These observations are consistent with Figure 8, which shows the mean energy spent for track maintenance (relative units). Besides the object maneuvers, these curves are influenced by the current object range (Figure 5, dotted line). In addition, the mean energy spent per revisit is displayed. Up to characteristic peaks, the energy per revisit is roughly the same for all tracking filters.

#### D. On the Quality of RCS Estimates

In a practical application, the mean RCS of the objects to be tracked is unknown and might be estimated from object amplitude information. In general, the estimators used should be at least approximately bias-free, the estimated error and the empirical error should be roughly identical, and the estimators should show a certain robustness against model mismatch. As indicated by Figure 9, the estimator previously proposed provides rather satisfying results for all trajectories. Using IMM-MHT for tracking, the recursion was initiated with  $\bar{\sigma} = .5 \text{ m}^2$  (worst-case assumption) and  $m_0 = 1.01$ .

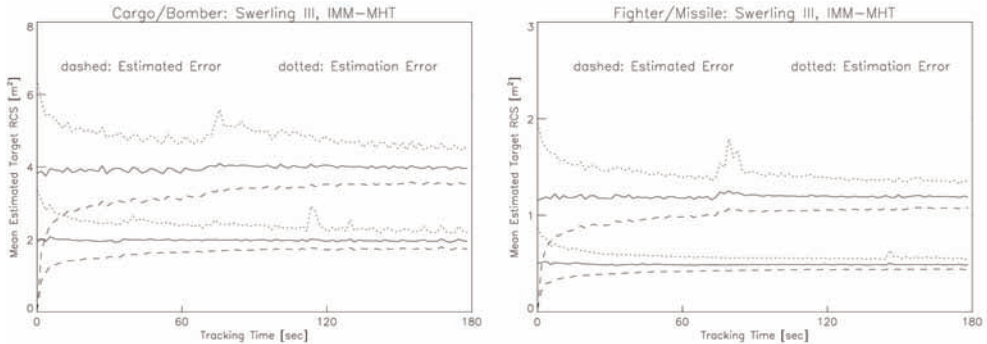


Fig. 9. On the quality of RCS estimates

The solid lines show the mean RCS estimates as a function of the tracking time. For all scenarios it is roughly constant and corresponds with the actual values (4, 2, 1.2, 0.5 m<sup>2</sup>). The dotted lines indicate the mean estimation error (available in the simulation). The curves show peaks that are related to the onset of maneuvers and the corresponding lack of track accuracy. The dashed lines denote the mean standard deviation calculated by the estimator itself. Tracking and RCS estimation are closely interrelated: Only when tracked over a certain period of time, are estimates reliable enough to distinguish between the object classes. A satisfying RCS estimation by signal processing only, i.e. without a temporal integration along the estimated trajectory does not seem to be possible. In this context, IMM retrodiction techniques [23] might be considered that can provide more accurate estimates of the trajectory and thus more accurate RCS estimates.

object type	processed	simulated	RCS [m <sup>2</sup> ]	estimated error	estimation error	energy
Bomber	III	III	1.96	0.22	0.28	0.37
	III	I	2.34	0.25	0.57	0.42
	I	III	1.77	0.28	0.33	0.39
	I	I	2.03	0.32	0.34	0.44
Fighter	III	III	1.19	0.13	0.16	0.82
	III	I	1.41	0.15	0.34	1.
	I	III	1.07	0.16	0.19	0.86
	I	I	1.22	0.18	0.19	0.99

Table I. Mismatch regarding the fluctuation model

#### E. RCS Model Mismatch

The backscattering properties of real objects are highly complex. A practicable method for estimating the RCS must thus show some robustness against model mismatch. To get a first hint, we generated in our simulation amplitude information according to both Swerling I and III being processed according to both modelling assumptions. The results for the four possible combinations are summarized in table I. Besides the quantities already shown in Figure 9, we also listed the total energy spent for tracking (relative units).

1. For matching models, the RCS estimates are nearly bias-free and more or less roughly consistent.
2. For Swerling III fluctuations, the estimates are more accurate than in case of Swerling I.
3. For Swerling I (no mismatch), more energy is spent than for Swerling III (keeping  $SN_k|_{k-1}$  constant).
4. If Swerling I amplitudes are processed according to Swerling III, the RCS is overestimated, consistency is lost.
5. It is underestimated if Swerling III amplitudes are processed according to Swerling I.
6. Mismatch does not greatly affect the performance (energy).

#### F. Adaptive Energy Management.

Finally we have to show up to what degree the radar energy to be spent can be reduced by estimating the RCS in comparison to worst-case assumptions. In Figure 9 the mean radar energy spent for track maintenance is displayed. The dotted lines refer to IMM-MHT tracking using the true RCS of the objects (as previously discussed). In a practical application, this cannot be realized; the resultant curves, however, may serve as a reference to discuss the performance of RCS-adaptive algorithms. The solid lines denote methods that exploit signal strength information for estimating the RCS (Worst-Case Kalman filter, IMM-MHT). Dashed lines indicate algorithms that use a worst-case assumption (here: 0.5 m<sup>2</sup>, missile) on the RCS (KF, IMM-MHT).

A comparison between sensor control by using the true RCS (not available in a real application) and methods exploiting recursive RCS estimates is of particular interest. The largest deviation is observed for scenario 1 ( $\bar{\sigma} = 4$  m<sup>2</sup>). This is to be expected, as the recursion was started with a worst-case assumption. The discrepancy between both curves, however, is not very significant in all four cases. Compared with IMM-MHT (Worst-Case



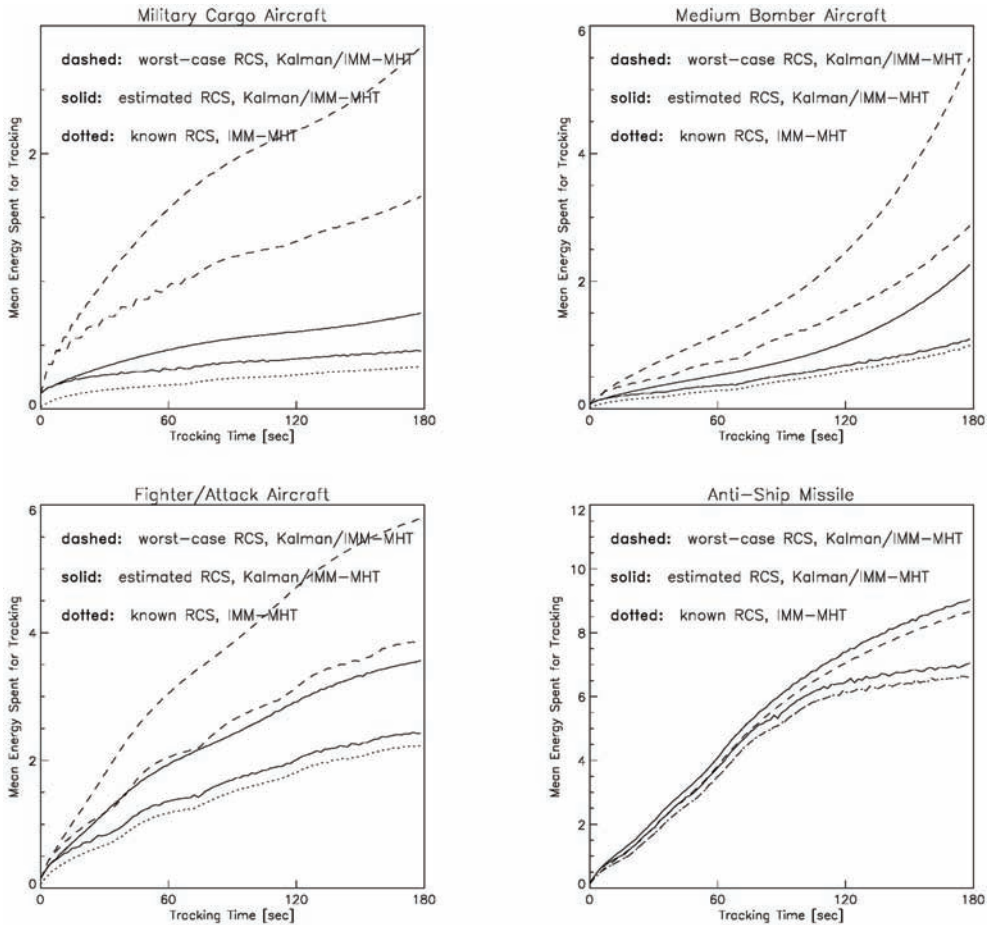


Fig. 10. Mean Number of Allocations for Different Filters and Trajectories

RCS) it can be neglected. The difference between sensor control with known and estimated RCS is roughly constant during tracking. We thus conclude that it is caused primarily in the initiation phase where not much signal strength information is yet available. As soon as reliable RCS estimates have been produced, the performance is practically identical. Figure 9 also shows how the resource savings due to adaptive dynamics models and RCS-adaptive energy management are related to each other.

In table II, scalar performance measures are summarized for all scenarios and processing methods: object revisit intervals ( $\Delta T$ ), sensor allocations required, energy spent for track maintenance (time averages taken over the tracking time). The last column shows the energy spent by the various methods relative to IMM-MHT with known RCS. Compared with IMM-MHT (Worst-Case RCS), the gain is: 3.8 (cargo aircraft), 2.6 (Bomber), 1.5 (Fighter), .9 (anti-ship missile). Hence, in the missile-scenario, where the worst-case assumption is correct, a small loss of performance must be taken into account.

object type	RCS	filter	$\Delta T$ [s]	revisits	energie	rel.
Cargo	worst case	Kalman	3.3	55.1	2828	8.5
		IMM-MHT	6.5	32.6	1664	5.0
	estimated	Kalman	2.0	90.3	750	2.2
		IMM-MHT	4.9	45.0	453	1.4
	known		4.9	46.4	334	1
	Bomber	worst case	Kalman	3.1	60.1	5488
IMM-MHT			6.4	34.0	2868	2.9
estimated		Kalman	2.2	85.5	2257	2.3
		IMM-MHT	5.1	44.2	1095	1.1
known			5.0	45.7	993	1
Fighter		worst case	Kalman	2.8	67.3	5786
	IMM-MHT		4.9	43.3	3882	1.7
	estimated	Kalman	2.2	85.6	3563	1.6
		IMM-MHT	3.9	54.0	2420	1.1
	known		3.8	55.1	2226	1
	Missile	worst case	Kalman	2.1	86.6	8657
IMM-MHT			4.1	56.1	6593	1
estimated		Kalman	2.2	85.4	9036	1.4
		IMM-MHT	4.2	55.4	7042	1.1
known			4.1	56.1	6593	1

Table II. Gain by rcs-adaptive energy control

## 5. Adaptive sensor management: summary of results

The gain by exploiting adaptive dynamics models and signal amplitude information is demonstrated by simulations with standard benchmark trajectories representative of typical military objects (cargo aircraft, medium bomber, fighter, and anti-ship missile) [8], [9]. Preliminary results were published in [25].

1. In the case of IMM-MHT, simple worst/best-case considerations seem to be sufficient for modelling the object dynamics. Medium-case models implying additional, a priori unknown parameters (e.g. transitions matrices) result in significant performance improvements only for standard IMM algorithms. IMM-MHT thus permits simplified, more *qualitative* models without significant loss of performance.

2. Compared with worst-case Kalman filtering, IMM results in considerable resource savings. The reduction with respect to the number of allocations required and the energy spent for track maintenance is roughly comparable and varies between 50 and 100% depending on the scenario considered. Essentially, the savings are due to longer revisit intervals on an average.
3. IMM-MHT improves on standard IMM algorithms. The difference, however, is less significant than between standard IMM and worst-case Kalman filtering. Besides simplified modelling assumptions, the practical use of IMM-MHT primarily consists in the exploration of the theoretical boundaries that limit the performance improvements achievable by adaptive dynamics models.
4. Due to abrupt maneuvers after a longer inertial flight, IMM-type tracking must necessarily be complemented by efficient Bayesian algorithms for adaptive beam positioning and local search. If used, however, tracking process remains highly stable, because all information on the possible dynamical behavior of the objects is taken into account.
5. By processing object amplitude information along the estimated trajectory, the a priori unknown RCS of the objects can (roughly) be estimated. The estimate is approximately bias-free; its variance corresponds with the empirical variance. It is closely related to the tracking process and might provide a contribution to object classification. Within certain limits, the method seems to be rather robust against model mismatch.
6. Compared with worst-case assumptions on the object RCS, significant energy savings can be obtained by exploiting amplitude information. Depending on the scenario considered, the gain is larger than the improvement achievable by adaptive dynamics models. The difference to algorithms that use of the correct object RCS (available in a simulation) is comparatively small and arises mainly in the initiation phase.

## 6. References

- [1] M. Abramowitz and I.A. Segun. *Handbook of Mathematical Functions*, Dover (1965).
- [2] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. Wiley & Sons, 2001.
- [3] Y. Bar-Shalom and X.-R. Li. *Estimation and Tracking: Principles, Techniques, Software*, Boston, MA: Artech House, 1993.
- [4] J. Biermann. Understanding Military Information Processing - An Approach to Support Intelligence in Defence and Security. In *NATO Advanced Research Workshop on Data Fusion Technologies in Harbour Protection*, Tallinn, Estonia, 2005.
- [5] S. Blackman. Multiple Hypothesis Methods. In *IEEE AES Magazine*, 19, 1, p. 41-52, 2004.
- [6] S. Blackman and R. Populi. *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.
- [7] R. J. Dempster, S. Blackman, S. H. Roszkowski, and D. M. Sasaki. IMM/MHT Solution to Radar and Multisensor Benchmark Tracking Problems. In *SPIE 3373, Signal and Data Processing of Small objects* (1998).
- [8] W. D. Blair, G.A. Watson, T. Kirubarajan, and Y. Bar-Shalom. Benchmark for Radar Resource Allocation and Tracking objects in the Presence of ECM. In *IEEE TAES* 35, No. 4 (1998).
- [9] W. D. Blair, G. A. Watson, and K. W. Kolb. Benchmark Problem for Beam Pointing Control of Phased-Array Radar in the Presence of False Alarms and ECM. In *Proceedings 1995 American Control Conference*, Seattle WA (1994).

- [10] W. Blanding, W. Koch, and U. Nickel. Tracking Through Jamming Using 'Negative' Information. In *Proc. of the 9<sup>th</sup> International Conference on Information Fusion FUSION 2006*, Florence, Italy, July 2006. To appear also in IEEE TAES.
- [11] Philip L. Bogler. Radar Principles with Applications to Tracking Systems. John Wiley & Sons (1990).
- [12] W. Fleskes and G. van Keuk. On Single Target Tracking in Dense Clutter Environment - Quantitative Results. In *Proc. of IEE International Radar Conference RADAR 1987*, pp. 130-134.
- [13] I. S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*, Academic Press (1979).
- [14] D. L. Hall and J. Llinas (Eds.). *Handbook of Multisensor Data Fusion*. CRC Press, 2001.
- [15] Young-Hun Jung and Sun-Mog Hong. Modelling and Parameter Optimization of Agile Beam Radar Tracking. In IEEE TAES, vol. 39, no. 1, p.13-33, Jan. 2003.
- [16] G. van Keuk. Multihypothesis Tracking with Electronically Scanned Radar". In *IEEE AES* 31, No. 3 (1995).
- [17] G. van Keuk and S. Blackman. On Phased-Array Tracking and Parameter Control. In *IEEE AES* 29, No. 1 (1993).
- [18] T. Kirubarajan, Y. Bar-Shalom, W. D. Blair, and G. A. Watson. IMMPDAF Solution to Benchmark for Radar Resource Allocation and Tracking Targets in the Presence of ECM", *IEEE AES* 35, No. 4 (1998).
- [19] W. Koch. A Bayesian Approach to Extended Object and Cluster Tracking using Random Matrices. In *IEEE TAES*, July 2008.
- [20] W. Koch. On Exploiting 'Negative' Sensor Evidence for object Tracking and Sensor Data Fusion. in *Information Fusion*, 8(1), 28-39, 2007.
- [21] W. Koch. Ground Target Tracking with STAP Radar: Selected Tracking Aspects. Chapter 15 in: R. Klemm (Ed.), *The Applications of Space-Time Adaptive Processing*, IEE Publishers, 2004.
- [22] W. Koch. Target Tracking. Chapter 8 in S. Stergiopoulos (Ed.), *Signal Processing Handbook*. CRC Press, 2001.
- [23] W. Koch. Fixed-Interval Retrodiction Approach to Bayesian IMM-MHT for Maneuvering Multiple Targets. In *IEEE TAES*, 36, No. 1, (2000).
- [24] W. Koch. On Adaptive Parameter Control for Phased-Array Tracking. Signal and Data Processing of Small Targets, SPIE Vol. 3809, pp. 444-455, Denver, USA, July 1999.
- [25] W. Koch and G. van Keuk. On Bayesian IMM-Tracking for Phased-Array Radar. In Proc. DGON/ITG International Radar Symposium, IRS '98, Conference Proceedings, p. 715-723, Mnchen (1998).
- [26] W. Koch and G. van Keuk. Multiple Hypothesis Track Maintenance with Possibly Unresolved Measurements. IEEE TAES, 33, No. 3 (1997).
- [27] W. Koch, J. Koller, and M. Ulmke. Ground object Tracking and Road Map Extraction. In *ISPRS Journal on Photogrammetry & Remote Sensing*, 61, 197-208, 2006.
- [28] B. F. La Scala and B. Moran. Optimal Target Tracking with Restless Bandits. In *Digital Signal Processing*, vol. 16, no. 5, p. 479-87, Sept. 2006.
- [29] B. F. La Scala, M. Rezaeian, and B. Moran. Optimal Adaptive Waveform Selection for Target Tracking. In Proc. of the 9<sup>th</sup> International Conference on Information Fusion, FUSION 2006, Florence, Italy, July 2006.

- 
- [30] R. P. S. Mahler. "Statistics 101" for Multisensor, Multitarget Data Fusion. In *IEEE AES Magazine*, 19, 1, p. 53-64, 2004.
  - [31] B. Ristic and N. Gordon. *Beyond Kalman Filtering*. Wiley, 2004.
  - [32] D. Salmond. Mixture Reduction Algorithms for Target Tracking in Clutter. In *SPIE*, vol. 1305, April 1990, pp. 434-445.
  - [33] P. W. Sarunic and R. J. Evans. Adaptive Update Rate Tracking using IMM Nearest Neighbour Algorithm Incorporating Rapid Re-Looks. In *IEE Proc.-Radar, Sonar, Navig.*, 144, No. 4 (1997)
  - [34] R. Streit and T. E. Luginbuhl. Probabilistic Multi-Hypothesis Tracking. Naval Undersea Warfare Center Dahlgren Division, *Research Report NUWC-NPT/10/428*.
  - [35] M. Wieneke and W. Koch. The PMHT: Solutions to Some of its Problems. In *Proc. of SPIE Signal and Data Processing of Small Targets*, San Diego, USA, August 2007. Also submitted to *ISIF Journal of Advances in Information Fusion*.

# Target Data Association Using a Fuzzy-Logic Based Approach

Stephen Stubberud<sup>1</sup> and Kathleen Kramer<sup>2</sup>

<sup>1</sup>Rockwell Collin,

<sup>2</sup>University of San Diego  
USA

## 1. Introduction

Level 1 data fusion is defined as object assessment (Hall & Linus, 2001). This process of estimation and prediction of an entity can be decomposed into a functional series of subprocesses as defined by the well known Bowman model (Steinberg, et al, 1999) that is depicted in Figure 1. While data fusion can apply to a host of applications, this chapter looks at the problem from a target tracking point of view.

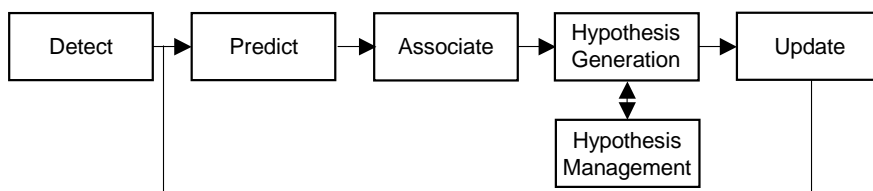


Fig. 1. Fusion Architecture Based upon the Bowman Model.

In tracking, the object, a potential target, can be considered to have two descriptive state vector elements: kinematics (e.g., position, velocity, etc.) and classification. These state vectors are generated based upon the information provided by a host of sensors over time. When a single target is present, this problem is straightforward. When multiple targets exist in the region of interest, however, the association subcomponent of Figure 1 becomes the key element of the process. Data association, sometimes referred to correlation, is the process of determining the correct target object to relate to each measurement. In Figure 2, multiple sensors along with a number of targets appear in the region interest. The reported kinematic measurements from all the sensors are shown in Figure 3. The association step of the fusion process determines which measurements are from the same target. This process is repeated each time a sensor reports measurements to the fusion system. In addition to issues with the number of targets and sensors reporting, each sensor has a degree of uncertainty associated with it. Instead of each measurement being expressed as a crisp point or even a line of bearing, uncertainty in the measurement creates a region of varying shape and size that depends on the type of sensor measurement and the sensor's accuracy is created. These regions, usually described as probability density functions, then interact with probability density functions that describe the target's kinematic track information.

To generate the associations between targets and measurements and between measurements of different sensors, association scoring routines have been developed. Since the measurements are described as random variables, as are the fused target tracks, the association algorithms are often based on a probabilistic measure. Most often, the association is based on the joint probability of the distance between the measurement and predicted location of the target based on the last known position and its velocity information:

$$F_z(z) = \int_a^b \int_c^d f_{xy}(x, y) dx dy \quad (1)$$

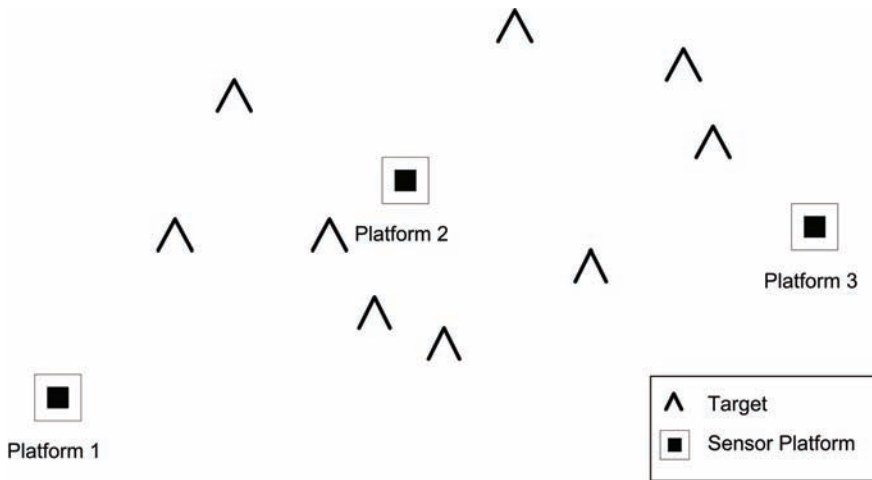


Fig. 2. Multiple sensor platforms in a region with multiple targets

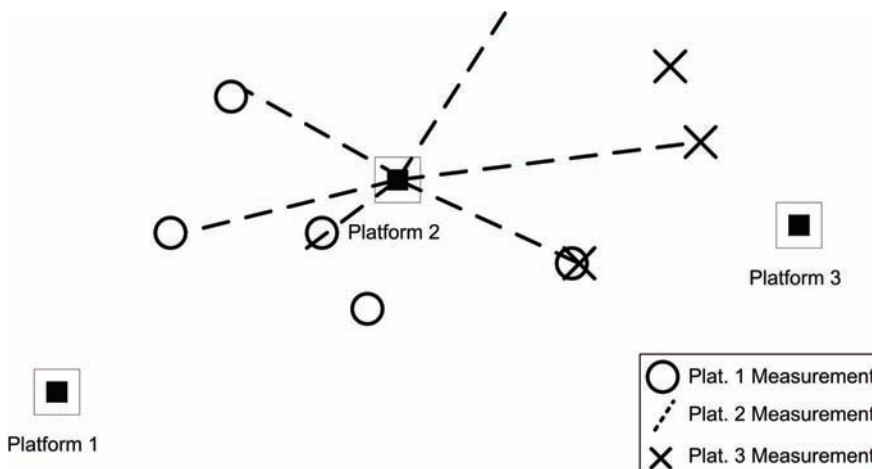


Fig. 3. Resulting reported kinematic measurements from platforms

where the bounds for the calculating the probability are based an estimate of the error in the target location prediction. Often, the track state estimate is computed using a Kalman filter (Blackman, 1986), (Blackman & Popoli, 1999), (Barshalom & Li, 1993). This implies that the measurements and the track states are Gaussian random vectors. When this assumption is used, a simplified distance measure, referred to has the Mahalanobis distance (Blackman & Popoli, 1999),

$$\chi^2 = (\mathbf{h}(\mathbf{x}_1) - \mathbf{z}_2)^T (\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}_2)^{-1} (\mathbf{h}(\mathbf{x}_1) - \mathbf{z}_2) \quad (2)$$

is used. The values  $\mathbf{R}$  and  $\mathbf{P}$  denote the measurement error covariance and the state error covariance, respectively. This weighted distance is computed in the measurement space. The coordinate transformation from the track state-space to the measurement space is through the output-coupling function

$$\mathbf{z}_1 = \mathbf{h}(\mathbf{x}_1) \quad (3)$$

and its associated Jacobian

$$\mathbf{H} = \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial x_1} & \frac{\partial \mathbf{h}}{\partial x_2} & \dots & \frac{\partial \mathbf{h}}{\partial x_n} \end{bmatrix}^T. \quad (4)$$

The distance is a unitless value and is a chi-squared random variable. As such, the Mahalanobis distance is sometimes referred to as the chi-squared metric. For some applications in tracking, the chi-squared metric is normalized between 0 and 1 as either a non-metric

$$e^{-\frac{(\mathbf{h}(\mathbf{x}_1) - \mathbf{z}_2)^T (\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}_2)^{-1} (\mathbf{h}(\mathbf{x}_1) - \mathbf{z}_2)}{2}}, \quad (5)$$

where 1 indicates perfect association and 0 indicates no association, or as a normalized metric

$$1 - e^{-\frac{(\mathbf{h}(\mathbf{x}_1) - \mathbf{z}_2)^T (\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}_2)^{-1} (\mathbf{h}(\mathbf{x}_1) - \mathbf{z}_2)}{2}}, \quad (6)$$

where the reverse is true.

Often, the probability association score of Eq. (1) can be considered computationally complex for real time calculations. One reason for this is that the density function can also be dependent on the reported mean value of the target location, i.e., near field effects. Also, each target track's density function is recalculated for each update with a new measurement. Eq. (1) does clearly allow for modeling of the sensor anomalies and blockages that can occur. The boundary could be quite complex. Piece-wise linear approximations can be used to reduce this complexity. On the other hand, the chi-squared metric (Eq. (2)), which is a standard association routine, is straightforward in its computation. The target tracks provided the Kalman filter already have the associate state error covariance. A sensor model provides the conversion. One drawback with this method is that the uncertainty of the measurement is assumed to be Gaussian. Some implementations use a Gaussian sum approach (Alspach & Sorenson, 1972) to approximate non-Gaussian distributions. This



requires multiple iterations of the scoring routine followed by a weighting computation. A new Gaussian sum must be computed if blockage occurs on the sensor. A problem that underlies both the probability association approach and the chi-squared metric approach is that the implementations often do not incorporate the underlying probability that is necessary for the algorithms to provide accurate results. Often, the equations are employed without regard to an understanding of the probability distributions or, in the case of blockages and anomalies, the redistributions of the probability.

Another technique for the data association problem is to use basic fuzzy logic to provide the scoring mechanism. Fuzzy logic can utilize a linguistic/rule-based interpretation and implementation of the problem. The technique that will be described in the succeeding sections of this chapter is based on two interpretation of the widely-used chi-squared metric: a mathematical interpretation and a graphical interpretation. Its foundations for the basic Gaussian measurement to Gaussian track problem are then expanded to the problems where one of the two elements of the association pair is Gaussian and the other is of uniform distribution. This is followed by the case where both track and measurement are uniformly distributed. The applications of both hard constraints or boundary conditions and soft constraints are developed.

## 2. Basic fuzzy logic

Fuzzy logic is simply the mapping from an input measurement space to an output measurement using linguistic variables. It gives us the ability to model imprecisions by incorporating qualitative components into a quantitative analysis. Fuzzification is the process of mapping a numerical value into linguistic variables and associated degrees of membership. Defuzzification in contrast is the process of taking a consequence fuzzy membership function and creating a crisp value.

To develop fuzzy logic for the correlation of data, two sets of fuzzy membership functions, antecedent and conclusion (consequent), are required. A membership function is a function that maps the elements of the set to a value from 0 to 1. The value to which the set is mapped is called the degree of membership. Each membership function for a fuzzy set relates to some knowledge base for that fuzzy set. For example, if the fuzzy set were temperature, the membership functions of hot, warm, and cold might be used. The range of temperatures would then be mapped into degrees of membership for each membership function. Figure 4 details these sets of membership functions. In Figure 4, the temperature 76 degrees has a degree of membership of 0.1 for cold, 0.6 for warm, and 0 for hot.

Figure 4 also demonstrates that all of the membership functions need not be the same shape or symmetric, although those characteristics are usually desirable to simplify the implementation. Another issue is the difference between inclusion of boundary points between the antecedent membership functions and the conclusion fuzzy sets. In antecedent sets, the extreme membership functions extend to infinity, as seen in Figure 5. Even if bounds exist on the antecedent fuzzy set, the extension will not have an adverse effect on the result. The conclusion function bounds are more important, however. As seen in Figure 6, the output bounds are exceeded. One of the most useful properties of fuzzy sets is that crisp values can be given a fuzzy representation. A crisp value in an antecedent fuzzy set can be represented as a simple discrete unit impulse. For the consequence membership functions, the crisp value is represented as the Dirac delta function.

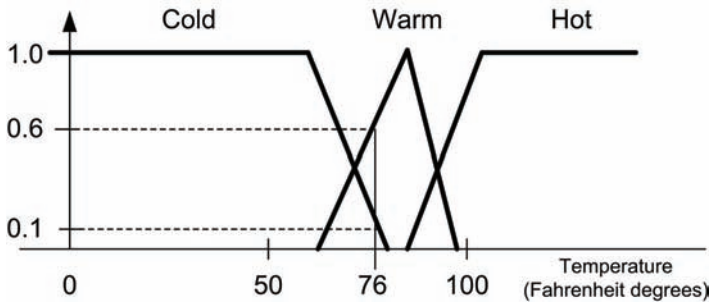


Fig. 4. Resulting reported kinematic measurements from platforms

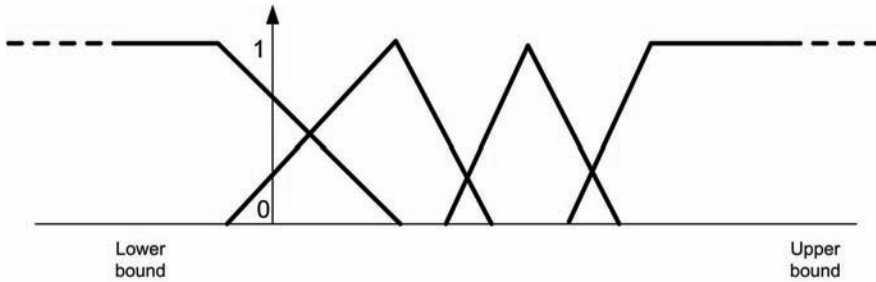


Fig. 5. Resulting reported kinematic measurements from platforms

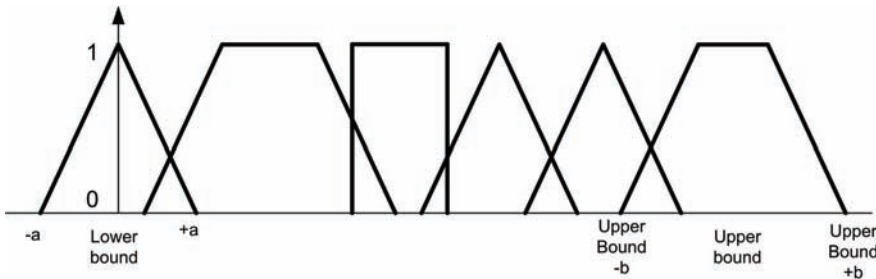


Fig. 6. Resulting reported kinematic measurements from platforms

If the function was defuzzified using the center of gravity method of Eq. (7), the bounds could not be achieved unless the center of the area of the extreme membership functions existed at the bounds. The position of the extreme bounds is determined by the defuzzification algorithm chosen. In the center of gravity technique,

$$output(t) = \frac{\sum_i center_i \cdot area_i}{\sum_i area_i} \tag{7}$$

where  $center_i$  is the center point of the  $i^{th}$  membership function and  $area_i$  is the area of the  $i^{th}$  membership function.

### 3. Basic fuzzy data association

There are a number of approaches to using fuzzy logic in the data association step for kinematic fusion. The approach developed here uses the chi-squared metric of Eq. (5) as the foundation. When this is done, the resulting association measure will be similar to the standard scoring algorithm when both the measurement and the track are Gaussian distributions. To create a fuzzy interpretation of the chi-squared metric, two interpretations of the association score are used: a mathematical interpretation and a graphical interpretation.

First, the chi-squared metric is considered a measure of the distance between two measurements: an estimated or predicted measurement and a reported measurement. The distance is weighted by the associated error covariances for each of these measurements. The covariance in the fuzzy logic association is used to scale the residuals or individual components of the Euclidian distance when they are mapped into the antecedent membership functions. This scaling relative to the context can be considered similar to the considerations of distances of golf as opposed to astronomy. The term *close* means in golf can mean a few inches while in astronomy it is considered in light years. Thus, the first fuzzy membership functions are developed based on the weighted residuals.

As seen in Figure 7, the parameters used to map the residual to the antecedent membership functions do not have fixed values. These parameters that define these membership functions are created using a fuzzy logic interpretation of the error covariance matrices and their related error ellipses. An error ellipse is an ellipse centered at the position mean of a measurement. The semi-major and semi-minor axes of the error ellipse are defined by the square root of the eigenvalues of the position covariance components of the error covariance matrices. The ellipse is oriented based on the eigenvectors. Using just the eigenvalues as the semi-axes creates a 1-sigma error ellipse. For this association approach, a 1.5-sigma error ellipse is used. An interpretation for the measurement covariance and an interpretation for the track covariance are both used to create and develop the parameter set, as seen in Figure 8. The parameter set is a weighted combination of the existing parameter set. This "layered" fuzzy approach is shown in Figure 9. By layering the fuzzy logic, two potential inputs to the association scoring inference engine are eliminated.

The second interpretation of the chi-squared metric is based graphically on the overlap of  $n$ -sigma ellipse of the track state and the measurement, as seen in an example in Figure 10. A percentage of overlap of the area of the measurement to the track covariance ellipse as well as the percentage of overlap area of the track to measurement covariance ellipse is calculated. The antecedent membership functions are shown in Figure 11. The two area measures along with the combined residual score form the three fuzzified inputs to the inference engine. A sample inference engine component for a residual of *medium* and the two overlap values is shown in Table 1.

In summary the five potential inputs have been reduced to three fuzzy inputs: the weighted residual, the percentage of measurement covariance overlap with the track covariance, and the percentage of the track covariance overlap with the measurement covariance. This is mapped through inference engines (Table 1) to an association score that is valued between 0 and 1 by defuzzifying the consequent resulting from Figure 12.

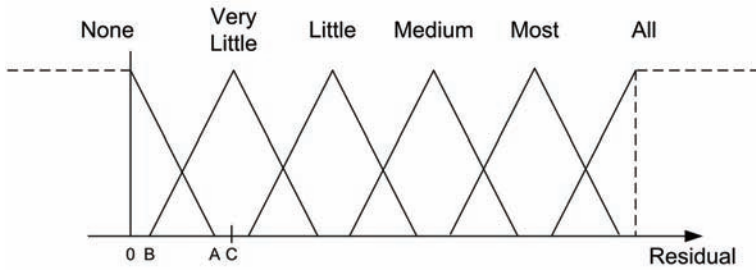


Fig. 7. Residual antecedent membership functions.

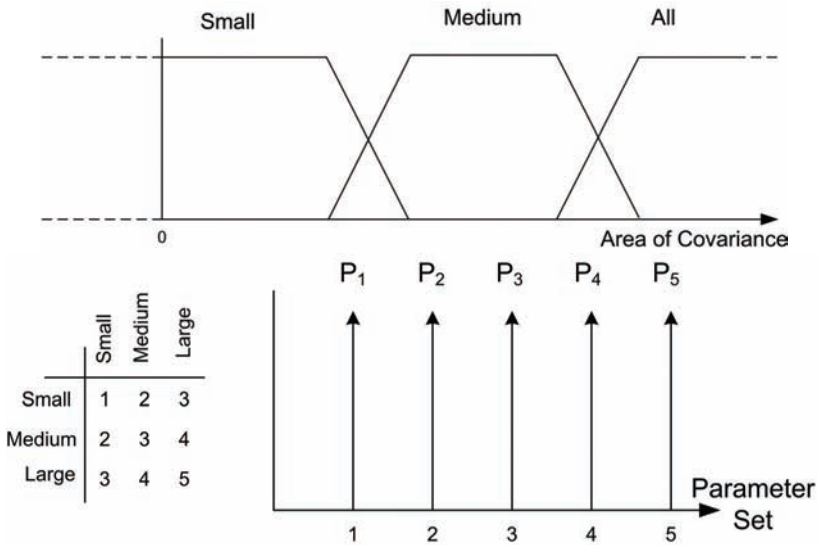


Fig. 8. Measurement and track covariances are used to determine a weighted parameter set through a fuzzy inference engine.

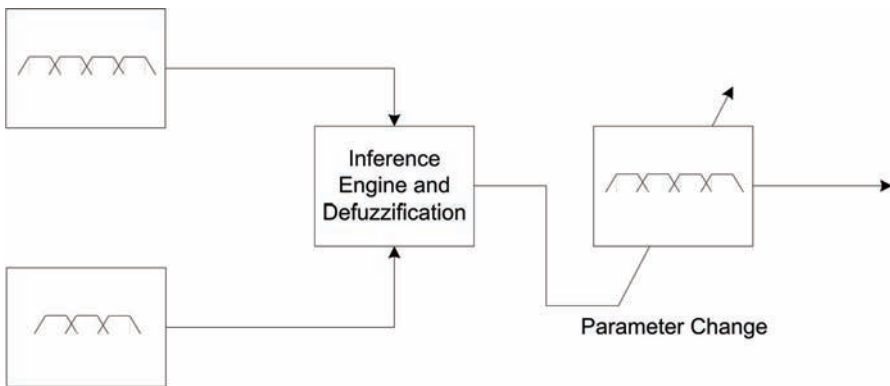


Fig. 9. A layered fuzzy system uses fuzzy logic to adjust the antecedent membership functions for another input. This reduces significantly the number of rules in the inference engines.

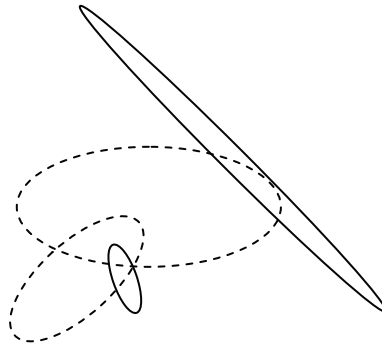


Fig. 10. Error ellipses approximate measurement and track Gaussian distributions. Here, two measurements (solid lines) overlap two tracks (dashed lines).

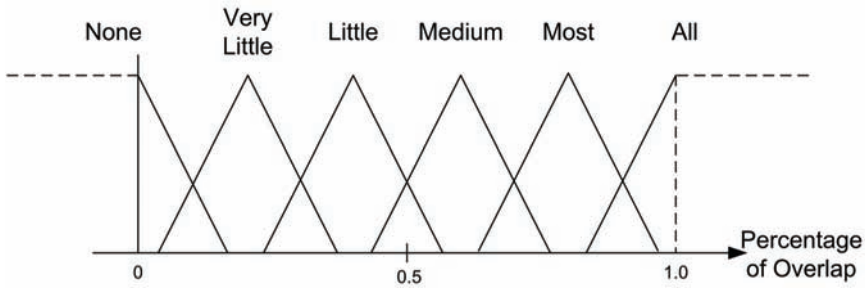


Fig. 11. Antecedent sets used to compute the association score for percentage of overlap of both track error ellipse and measurement error ellipse.

Medium	None	Very Low	Low	Med.	Most	All
None	Very Low	Very Low	Very Low	Very Low	Very Low	Very Low
Very Low	Very Low	Very Low	Very Low	Very Low	Low	Low
Low	Very Low	Very Low	Low	Low	Med	Med
Med.	Very Low	Very Low	Low	Med	Med	High
Most	Very Low	Low	Med	Med	High	Very High
All	Very Low	Low	Med	High	Very High	Very High

Table 1. Element of Inference Engine to Map to Association Score Consequence Functions for a Uniform Measurement to Gaussian Track

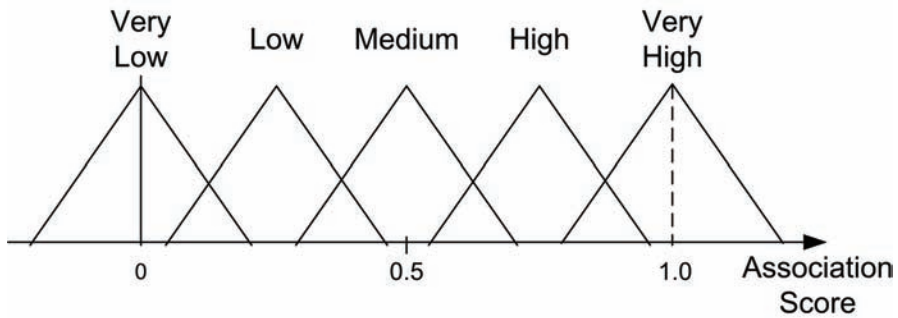


Fig. 12. Association score consequence membership functions.

### 3.1 Algorithm outline

The computation of the fuzzy data association score is performed using the following algorithm:

- Step 1. Compute the semi-major and semi-minor axes and the rotation of the error ellipse defined by the covariance matrices of the tracks and the measurements. The values are a result of the eigenvalues and the orthonormal eigenvectors. This will define a  $1$ -sigma error ellipse.
- Step 2. Compute the points of the octagons approximation of the  $n$ -sigma error ellipse for the measurements and the tracks.
- Step 3. For each track-measurement pair define the overlapping region of the error octagon approximations as set forth in the following steps:
  - Step 3a Determine the intersection points of each a measurement octagon with a track octagon. This is done by computing the intersections of the lines that define the line segments of the octagons sides.

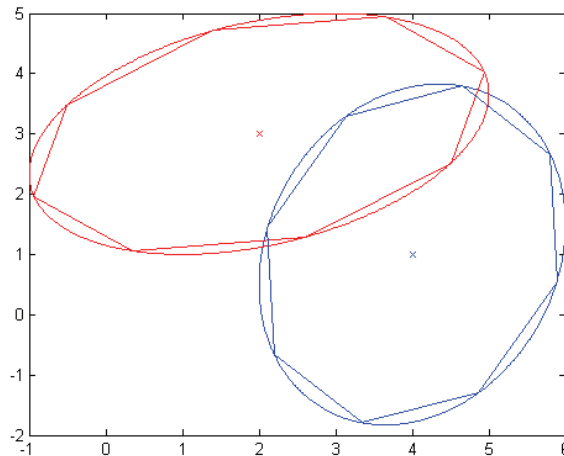


Fig. 13. Ellipses are approximated by octagons and the convex hull of their intersection is defined.

Step 3b Determine the vertices that may lie inside the other octagon such, as shown in Figure 13. This is done by determining taking each vertex separately of the measurement octagon placing it the track octagon and calculating if it creates a new convex hull. If a new convex hull is created then that vertex does not lie within the other octagon. The procedure is repeated by using the track vertices with the measurement octagon.

Step 4. Using the triangle area calculation of Heron's formula

$$area = \sqrt{s(s-a)(s-b)(s-c)},$$

where  $a$ ,  $b$ , and  $c$  are the vertices of the triangle and

$$s = \frac{1}{2}(a + b + c),$$

compute the area of the two octagons and the overlapping region as seen in Figure 14.

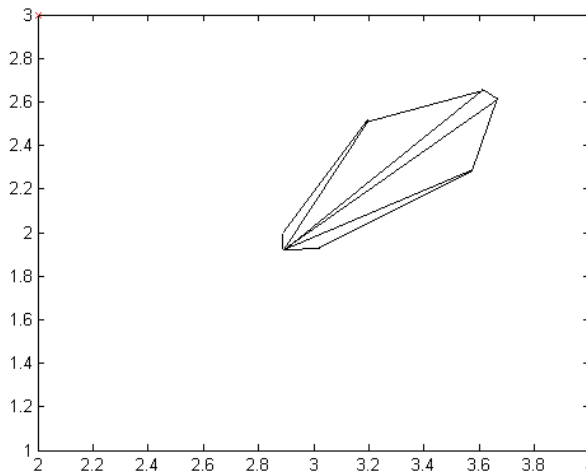


Fig. 14. The area of overlap between the two octagons is computed by dividing the polygon into triangles and applying Heron's formula.

Step 5. Compute the percentage of overlap for each covariance matrix.

Step 6. Using the layered fuzzy approach, which depends on the size of the covariances, define the degree of membership of the residual size. The covariances are used to generate the antecedent membership function parameters from an *a priori* set of parameters:

$$\mathbf{P}_1 = (a_1, b_1, c_1, \dots)$$

$$\mathbf{P}_2 = (a_2, b_2, c_2, \dots)$$

⋮

using a fuzzy logic approach.

- Step 7. Employing the residual antecedent function from Step 7 define the degree of membership of the residual.
- Step 8. Employing the antecedent membership functions, define the degree of memberships for the percentage of overlap for each of the octagons (track and measurement).
- Step 9. Employ the inference engines with the three inputs, residual and two percentages of overlap, and the consequence membership functions to compute the score.
- The functional flow of the algorithm is shown in Figure 15.

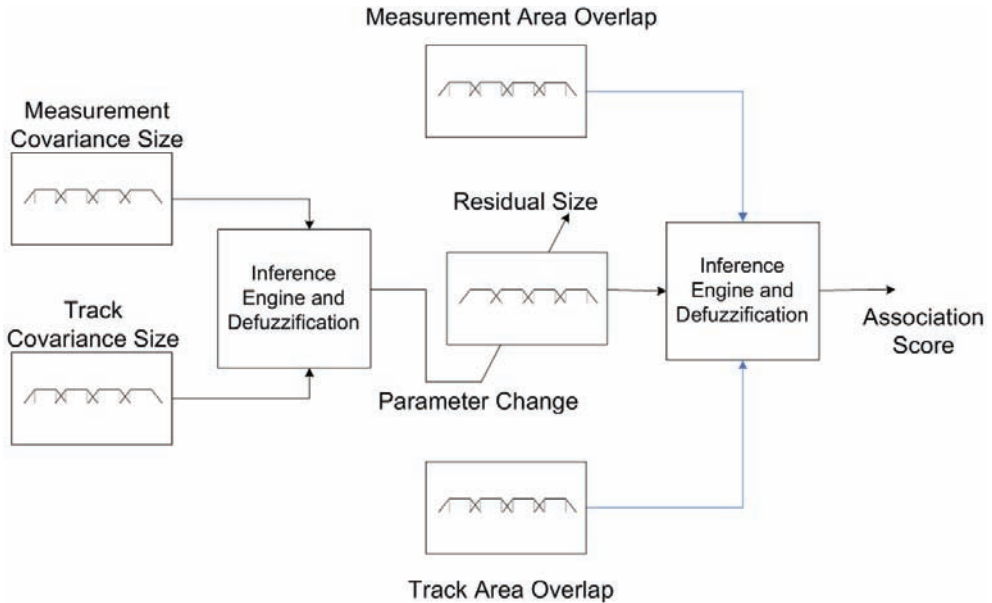


Fig. 15. To reduce inference engine complexity, a layering of fuzzy logic provides the functional approach to chi-squared metric emulation.

#### 4. Fuzzy data association with uniform measurements

One of the benefits of the fuzzy association approach is that it is not necessary that for either the measurements or the track to have a Gaussian distribution. With some minor modifications to the algorithm, the case of a uniform distribution, as happens with bearings-only measurements, can be associated without using a Gaussian approximation.

##### 4.1 Fuzzy association for a Gaussian track and a uniform measurement

The generation of the association score when the measurement is uniform can be performed in a manner quite similar to the algorithm outlined in Section 3.1. First, the semi-major and semi-minor axes and the rotation of the error ellipse defined by the covariance matrix of the track are computed in order to generate the 1-sigma error ellipse. The desired n-sigma error ellipse for the track is approximated by an octagon, while, for now, the uniform distribution is a rectangle. Use of more complex polygons can prove more accurate approximations to the uncertainty region. For each track-measurement pair, an overlapping region of the error octagon approximation and the uniform distribution is calculated. A pseudo-residual is



computed, based on the average distance from the Gaussian mean to the uniform distribution, as in Figure 16. The covariance size of the track generates the parameters for the distance antecedent membership function. Antecedent membership functions such as those in Figure 17 define the degree of membership values for the percentage of overlap for the track's octagon and the measurement distribution. These fuzzy values fire the inference engines using the three inputs - the residual and two percentages of overlap. Table 2 shows an inference engine when the residual value is fuzzified to *close*.

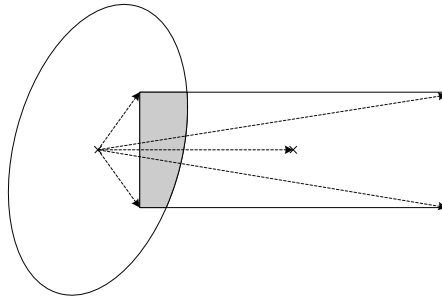


Fig. 16. Association of a Gaussian to a uniform distribution is dependent on both overlap and a modified residual.

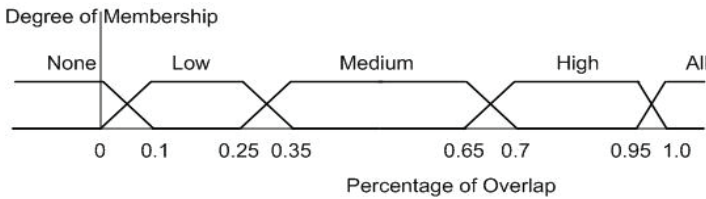


Fig. 17. Antecedent membership functions describe the percentage of overlap.

Relative Distance Close		Percentage of Gaussian Overlap					
		None	Very Low	Low	Med	High	All
Percentage of Uniform Overlap	None	Very Low	Very Low	Very Low	Very Low	Very Low	Very Low
	Very Low	Very Low	Very Low	Low	Low	Med Low	Med High
	Low	Very Low	Low	Low	Med Low	Med Low	High
	Med.	Low	Med Low	Med Low	Med High	High	High
	High	Med Low	Med High	Med High	High	High	Very High
	All	Med High	High	High	Very High	Very High	Very High

Table 2. Element of Inference Engine to Map to Association Score Consequence Functions for a Uniform Measurement to Gaussian Track

The resulting membership functions are defuzzified to generate the association score. The consequence membership functions used are similar to those depicted in Figure 18.

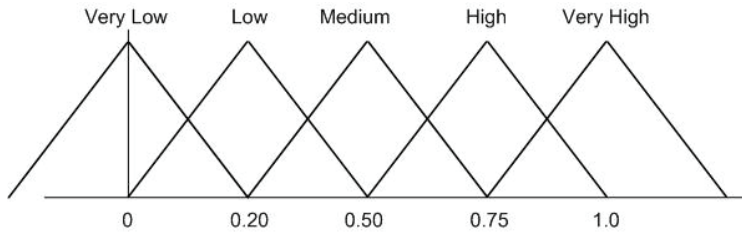


Fig. 18. Consequence membership functions used to develop the association score.

**4.2 Fuzzy association for a uniformly distributed track and a uniformly distributed measurement**

For the two uniform distributions case, the association problem degenerates into a simple problem of determining the area of overlap. First, the area of overlap for both the measurement and the track is computed. The percentage of overlap for each is then calculated. Fuzzification, using the antecedent membership functions similar to that of Figure 17, defines the degree of membership for the percentage of overlap for the track distribution and the measurement distribution. A single inference engine, such as shown in Table 3, maps the inputs to the consequence membership functions. The consequence membership functions shown in Figure 18 would also be suitable for this case. Defuzzifying the resulting membership function computes the resulting association score.

		Percentage of Measurement 1 Overlap					
		None	Very Low	Low	Med	High	All
Percentage of Measurement 2 Overlap	None	Very Low	Very Low	Low	Low	Med Low	Med Low
	Very Low	Very Low	Very Low	Low	Low	Med Low	Med Low
	Low	Low	Low	Low	Med High	Med High	Med High
	Med	Low	Med Low	Med High	Med High	Med High	High
	High	Med Low	Med Low	Med High	Med High	High	Very High
	All	Med Low	Med Low	Med High	High	Very High	Very High

Table 3. Inference Engine to Map to Association Score Consequence Functions for a Uniform Measurement to a Uniform Track

**5. Fuzzy data association in the presence of constraints**

The fuzzy association routine can be augmented to accommodate a variable scale target location region. As seen in Figure 19, information such as bathymetric data could influence

the location possibilities of a submarine or a ship (National Geophysical Data Center, 2007). Deep draft ships and submarines often avoid shallows that could endanger the vessel and or mission. Thus, various depths would weight the effectiveness of uncertainty differently based on the locations. So in Figure 20, the uncertainty ellipses of a track or measurement would be less pronounced near the shores and over the seamounts where operations would be less likely. Such measurements would better fit with littoral-active vessels. The channels would be much likely operations for the vessels in question. The incorporation of this new information is incorporated by creating a weighting to the component of the area of overlap in the fuzzy association routine. The weighting is generated using a fuzzy scoring technique. The added information improves the performance of the association routine based in a global sense.

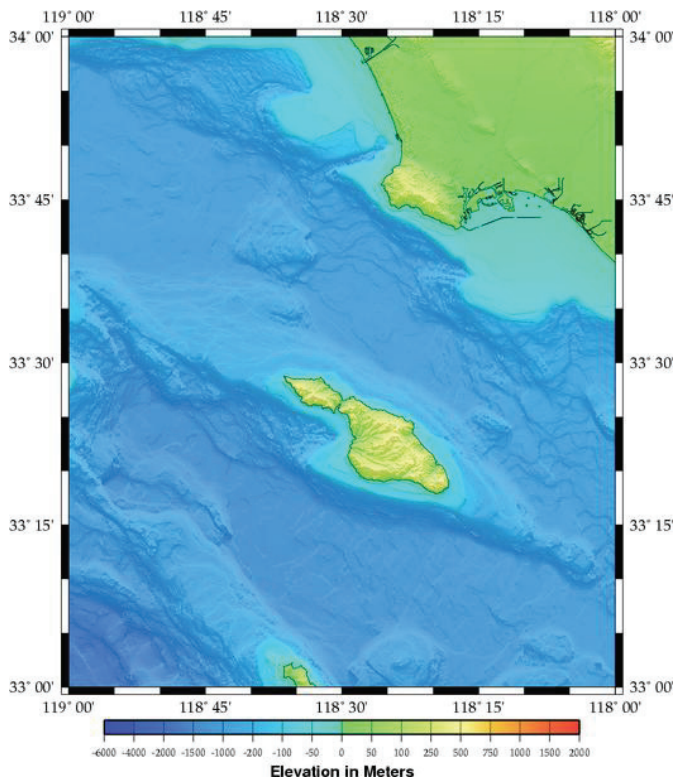


Fig. 19. Bathymetric data influences feasible track space of a vessel.

### 5.1 Fuzzy association with hard constraints

If sensor blockage or target prohibition is present, the covariance ellipses can be modified accordingly to incorporate the constraint. Figure ww+4a details the reduction of the error ellipse area based on sensor blockage of the sensor and prohibited terrain for the target. The new area considered for overlap is shown in Figure 20b. These are considered hard constraints in that the regions are removed. The remaining feasible regions can be mapped similar to the redistribution of the Dempster-Schafer concept (Dempster, 1967) or by a

simple reduction in area. The steps of the algorithm remain the same if area is removed due to sensor blockage or terrain prohibitions. Only the percentages of overlap change.

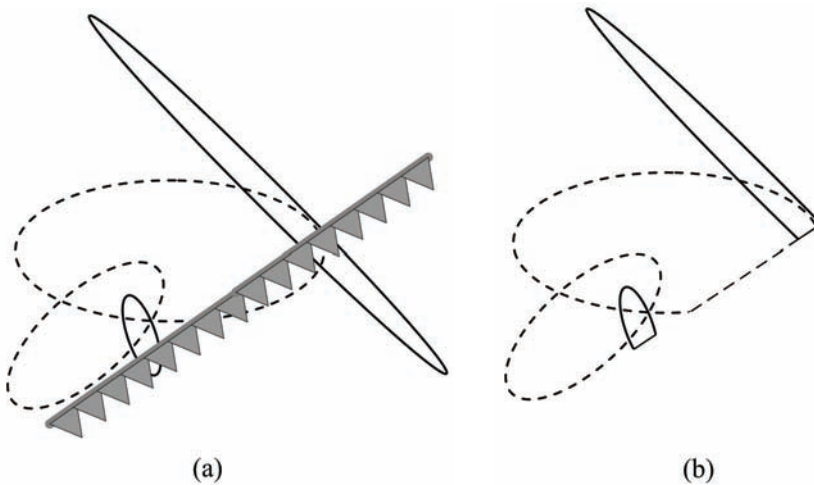


Fig. 20. Uncertainty error ellipses are affected by sensor blockage. a) Gaussian ellipses and blockage indicated; b) resulting alteration of ellipse.

### 5.2 Incorporation of soft constraints for fuzzy association – the penalty method

In Figure 21, shading is used to indicate four distinct regions where the target can operate with the measurement and the track uncertainty ellipses overlaying these. These regions might be used to reflect operational conditions, such as indicating the depth that a submarine can operate in the specific regions. Since stealth is the submarines primary

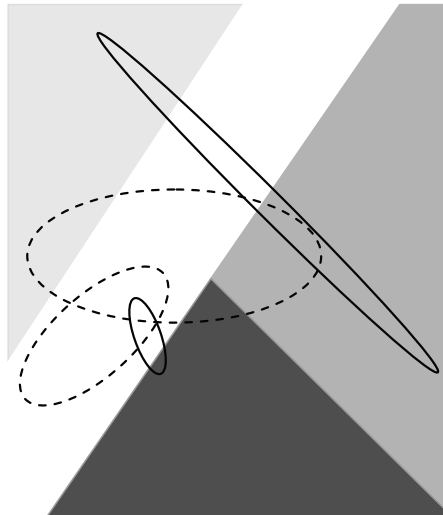


Fig. 21. Measurement and track ellipses span distinct regions of operation.

defense, operations in the shallows, where visual spotting becomes more likely, are less probable. So the different regions of overlap indicate that the area of overlap used by the fuzzy association algorithm should be reduced in a systematic way. The penalty method approach is based on the concept of discrete regions of target possibility or sensor capability. The approach proposed is to use a weighting scheme that maps the operational parameters into the environmental reports to create a weight. Since much of the information is heuristic in nature, the use of a fuzzy logic mapping for this weighting is proposed.

Figure 22 presents a chi-squared emulation with the added component of the penalty function. As shown, the weighting adapts the area of overlap using a fuzzy logic system. The complexity of the component varies significantly depending on the knowledge of the target type, the environment, and the sensor system.

As discussed in (Stimson, 1998) and (Hall & Linus, 2001), the sensor systems are often well understood with the physical capabilities well defined. The operational environment is somewhat less understood but often known to a good degree so that the mission can be successful. Finally, while the target capabilities may be well known, the actual classification of the target to use such data is not often available. A library of information can be used to generate membership functions *a priori* for each component of knowledge. The level of uncertainty associated with the knowledge component will be combined into the final score that will drive the adaptation weighting. This is shown in Figure 23.

The implementation of this penalty based routine is created by implementing the following algorithm:

- Step 1. Compute the semi-major and semi-minor axes and the rotation of the error ellipse defined by the covariance matrices of the tracks and the measurements. This is performed by generating the eigenvalues and eigenvectors of the covariance matrices. This will define a 1-sigma error ellipse.
- Step 2. Approximate the  $n$ -sigma error ellipses for both the measurement and the track by octagons. Compute the vertices of the octagons.
- Step 3. For each track-measurement pair, define the overlapping region of the error by using at least an octagon approximation to each area as set forth in the following steps:
  - Step 3a Determine the intersection points of each measurement octagon with a track octagon by computing the intersections of the lines that define the line segments of the sides of the octagon.
  - Step 3b Determine the vertices that may lie inside the other octagon by taking each vertex of the measurement octagon separately and calculating if it creates a new convex hull with the track octagon.  
If a new convex hull is created, then that vertex does not lie within the other octagon. The procedure is repeated by using each of the track vertices with the measurement octagon.
  - Step 3c Approximate the constraint boundary condition as exemplified in Figure 5 with a piece-wise linear approximation.
  - Step 3d Define the multiple irregular polygons for each operational region using the vertices defined in Step 3c and 3d.
- Step 4. Using the triangle area calculation of Heron's formula

$$area = \sqrt{s(s-a)(s-b)(s-c)},$$

where  $a$ ,  $b$ , and  $c$  are the vertices of the triangle and

$$s = \frac{1}{2}(a + b + c),$$

the area of the two octagons and the overlapping region is then computed by summing the areas of the triangles.

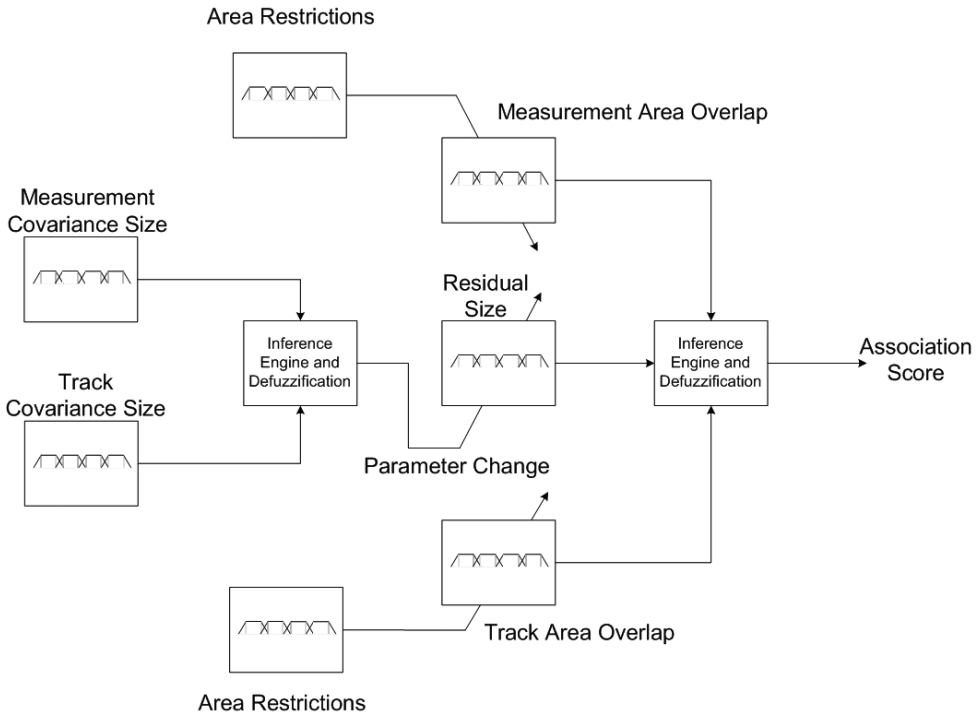


Fig. 22. Penalty function augments determination of overlap.

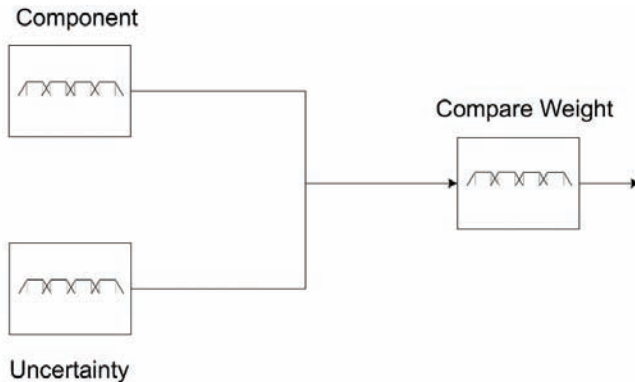


Fig. 23. Level of uncertainty combines into final score.

- Step 5. Compute the percentage of overlap for each covariance matrix using penalty method approach:
- Step 5a Using layered fuzzy logic as in Figure 7, the information from the different constraint components and their uncertainties create a component weight which then is fed as input to the fuzzy systems to create the area restriction weight for each region defined in Step 3d.
- Step 5b Compute the weighted area

$$area = \sum_{i=1}^{\# \text{ of regions}} w_i area_i$$

and incorporate that into percentage of overlap:

$$percent = \left( \frac{area}{\prod eig(\mathbf{R})} \right) \cdot 100$$

- Step 6. Using the layered fuzzy approach shown in Figure kk+1, which depends on the size of the covariances, define the degree of membership of the residual size. Antecedent membership functions provide the fuzzification of the error covariances. The inference engine then maps input to the consequence membership functions. Each parameter defines a set of parameters for each of the membership functions which provide the parameters that describe the antecedent membership functions for the residual.
- Step 7. Employing the antecedent membership functions for overlap, the degree of memberships for the percentage of overlap for each of the octagons (track and measurement) is defined.
- Step 8. Employing the inference engines with the three inputs, residual and two percentages of overlap as described above.
- Step 9. Defuzzify the resulting membership function to generate the association score. The consequence membership functions are used to compute the score.

## 6. Association example with constraints

To demonstrate the implementation of the association concept an example with constraints is shown. The implementation of this complex example is shown for the three cases: no constraint, a hard constraint, and soft constraints.

The typical passive fixed-sonar array provides a superb example for the use of the soft-constraint fuzzy association routine. Depending on the deployment of the line array and its final settling point on the sea bottom, the capabilities of sonar for different bearing angles varies. In Figure 24, a simulated example of an array's direct blast zone is shown. Because of line distortions as the result of nonlinear deployment and sea-bottom features which block sound propagation to some or all of the receivers, the acoustic capabilities change. The darker regions indicate less reception capabilities for the array. A fixed line array also has the problem that sign of the angle of the acoustic source from the sonar cannot always be determined. Thus, a mirror image of the true measurement is also generated.

Figure 25 shows a three target example for the sonar array defined in Figure 24 with the boundaries of the capability regions are overlaid. The measurement is a bearings-only

measurement with ambiguity. The associated uncertainty is defined by the solid ellipses. The track uncertainties are defined by the dashed lines. The uncertainty ellipses and track locations are defined in Table 4 for the three targets. The center of the direct blast convergence zone is considered to be origin of the region. As an ambiguous line of bearing, the measurement is approximated by a single Gaussian with its uncertainty ellipse defined in Table 5.

Figure 26 defines the antecedent membership functions based on the capabilities. Figure 27 defines the consequence membership functions to generate weighting functions for the areas of overlap.

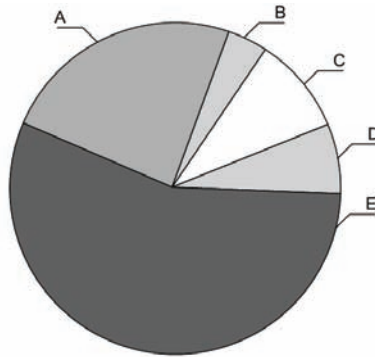


Fig. 24. Array has varying levels of reception (with darker regions indicating poorer reception) over direct blast zone.

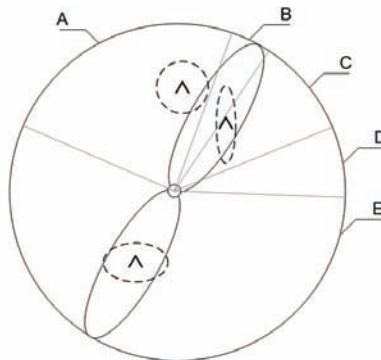


Fig. 25. Example with three targets in array region with varying levels of reception.

Track no.	x-coord (NM)	y-coord (NM)	Semi-major axis (NM)	Semi-minor axis (NM)	Angle (deg)
1	7.69	9.85	5.77	1.45	90
2	1.23	15.38	3.85	3.85	0
3	-5.69	-10.62	4.81	2.89	0

Table 4. Target track locations and uncertainties



Meas. no.	x-coord (NM)	y-coord (NM)	Semi-major axis (NM)	Semi-minor axis (NM)	Angle (deg)
1	6.31	10.92	12.5	3.85	60
2	-6.31	-10.92	12.5	3.85	240

Table 5. Measurement locations with an ambiguous detection

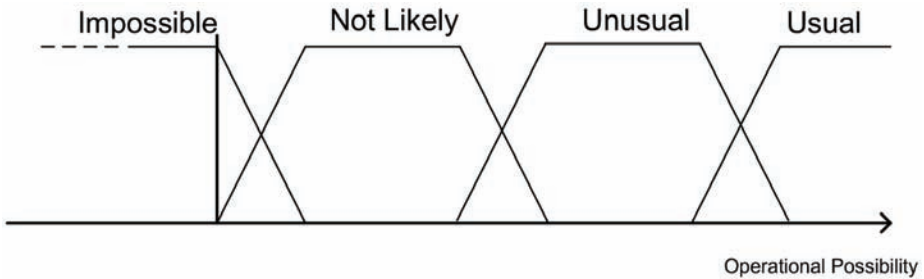


Fig. 26. Antecedent membership functions for possibility of a target in a region

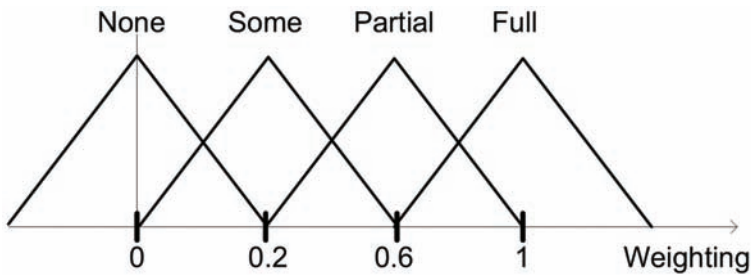


Fig. 27. Consequence membership functions for the weighting based on soft constraints.

The inference engine is defined as

<b>Track Existence</b>	<b>Impossible</b>	<b>Not Likely</b>	<b>Unusual</b>	<b>Usual</b>
	None	Some	Partial	Full

For the hard constraint case, the inference engine is defined as

<b>Track Existence</b>	<b>Impossible</b>	<b>Not Likely</b>	<b>Unusual</b>	<b>Usual</b>
	None	None	Full	Full

The results for the soft-constraint or penalty method approach for association are compared in Table 6. They are compared against those of the normalized chi-squared technique, the standard fuzzy association method, and the hard-constraint or barrier method.

	Chi-Squared	Fuzzy Association	Fuzzy Assoc. with Constraints	Fuzzy Assoc. with Penalty
Target 1	0.767	0.637	0.637	0.58
Target 2	0.024	0.083	0	0.008
Target 3	0.9252	0.841	0	0

Table 6. Association scores based on the normalized chi-squared and the three fuzzy association techniques

The results for this example case clearly indicate that use of constraints provides improved association scoring as the non-possible solutions are eliminated. The soft-constraint approach has the advantage over the use of hard-constraints in that improbable solutions still appear but with a lower score. This is of use in multi-hypothesis tracking systems and N-dimensional-assignment based systems.

## 8. References

- Hall, D. & Llinas, J. , (2001). *Handbook of Multisensor Data Fusion*, CRC Press, Boca Raton.
- Steinberg A. ; Bowman, C., & White, F. , (1999). "Revisions to the JDL Data Fusion Model", *Proceedings of the SPIE Sensor Fusion: Architectures, Algorithms, and Applications III*, pp 430-441, Orlando, FL, March, 1999.
- Blackman, S., (1986). *Multiple-Target Tracking with Radar Applications*, Artech House, Norwood, MA.
- Blackman, S. & Popoli, R. , (1999). *Design and Analysis of Modern Tracking Systems*, Artech House, Norwood, MA.
- Bar-Shalom, Y. & Li, X.-R., (1993). *Estimation and Tracking: Principles, Techniques, and Software*, Artech House Inc., Norwood, MA.
- Alspach, D. & Sorenson, H., (1972). "Nonlinear Bayesian Estimation Using Gaussian Sum Approximations," *IEEE Transactions on Automatic Control*, Vol. 17 , No. 4 , pp. 439 - 448, Aug., 1972.
- National Geophysical Data Center, Geophysical Search and Data Retrieval System, (2007).
- Bar-Shalom, Y. & Tse, E., (1975). "Tracking in a Cluttered Environment with Probabilistic Data Association," *Automatica*, 11, Sep., 1975, pp. 451-460.
- Priebe, R. & Jones, R., (1991). "Fuzzy Logic Approach to Multi-Target Tracking in Clutter," *Proc. of SPIE, Vol. 1482, Acquisition, Tracking, and Pointing V*, pp. 265-274, Orlando, FL, March, 1991.
- Lefebvre, E. & Helleur, C., (2002). "Use of Fuzzy Logic for Data Fusion in a Recognized Maritime Picture," *Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation (Proc. of the 3rd WSES International Conference on Fuzzy Sets and Fuzzy Systems)*, pp. 24-29, Interlaken, Switzerland, Feb, 2002.
- Dempster, A.P., (1967). "Upper and Lower Probabilities Induced by a Multivalued Mapping," *Annals of Mathematical Statistics*, 38, 1967, pp. 325 - 339.
- Stimson, G.W. , (1998). *Introduction to Airborne Radar: Second Edition*, Scitech Publishing, Inc., Mendham, New Jersey.

Stubberud, S.C. & Kramer, K.A., (2006). "Data Association for Multiple Sensor Types Using Fuzzy Logic," *IEEE Transactions on Instrumentation and Measurement*, Vol. 55, No. 6, pp. 2292-2303, Dec., 2006.

# Data Fusion Performance Evaluation for Dissimilar Sensors: Application to Road Obstacle Tracking

Blanc Christophe, Checchin Paul, Gidel Samuel and Trassoudaine Laurent  
*LASMEA - Blaise Pascal University  
France*

## 1. Introduction

Many science and engineering applications require hardware and software systems that acquire, process, and integrate information gathered by various knowledge sources. A typical example is that of an intelligent road vehicle using several sensors installed in its front bumper to perform a robust driving assistance under all operative conditions. For most of these systems, information made available by the knowledge sources is incomplete, inconsistent, or imprecise. A crucial element in achieving autonomy and efficiency for these systems is the availability of a mechanism that can model, fuse, and interpret the information for knowledge assimilation and decision making. The fused data reflects not only information generated by each knowledge source, but also information that cannot be inferred by either knowledge source acting alone.

This chapter deals with fusing data from sensors that provide information about kinematics characteristics of targets in a moving road scene. The sensors include radar and laser range finder. Information about the positions and velocities, in addition to errors associated with sensor readings is used to solve the target tracking problem. Target tracking with both single and multiple sensors involve this concept: if the track kinematic estimate errors are small, it is easy to locate the next target measurement and continue to update and/or refine the kinematic estimate. The correctly associated measurements provide a “restoring force” since they must correspond to the same underlying target. Loss of accuracy for any reason makes it more difficult to identify the measurements from the target of interest if other targets are present. The aim of the fusion center is to obtain an estimate of the target kinematic state vector and the accuracy of this state estimate quantified by its error covariance. Estimation fusion can be classified into three categories, depending on which information is available at the fusion center: centralized fusion if all measurements are available at the fusion center, decentralized fusion if local estimates are available at the fusion center and hybrid fusion if available information includes both unprocessed data from one sensor and processed data from the other one. In order to be robust, the best achievable performance has to be defined. It depends on the accuracy of the measurements, the sampling interval, and the scenario used. For a linear and Gaussian system, the Kalman filter estimation is optimal. However, in non-linear cases, one cannot conclude on the

optimality of an estimation system. Some comparisons using bounds are thus made to characterize performance limitations, and consequently, to determine whether imposed performance requirements are realistic or not. In time-invariant statistical models, a commonly used lower bound is the Cramer-Rao Lower Bound (CRLB), given by the inverse of the Fisher information matrix. An extension of the CRLB to random parameters was derived: Posterior CRLB (PCRLB). More recently, a simple and straightforward derivation of the PCRLB for the problem of discrete-time nonlinear filtering has been proposed. Many bound comparisons will be made according to fusion architectures, to the accuracy of measurements, to various sensor configurations, and to the scenarios used in order to select the most robust fusion system which has to be installed in our experimental vehicle (VELAC: LASMEA's experimental vehicle). This paper starts with a description of the motivation to use a combination of range-only measurements with Cartesian ones, and provides a mathematical formulation of the problem. Subsequently, the Posterior Cramer-Rao Lower Bounds (PCRLB) are derived and analyzed from the aspect of algorithm convergence. Section 4 presents the two proposed algorithms and their error performance comparison according to the theoretical bounds. Since the combination of range-only measurements with Cartesian ones can be formulated as a nonlinear filtering problem, the Extended Kalman Filter (EKF) and the Particle Filter (PF) are selected as approximation to the optimal recursive Bayesian solution of the nonlinear filtering problem. Finally, an experiment is made in order to evaluate the decentralized fusion performance.

## 2. Problem description

### 2.1 Background

Among the mobile robotic problems, obstacle detection and avoidance are the most important. Indeed, each mobile robot having to move in an unknown environment must be able to detect obstacles. J. Hancock's point of view (Hancock, 1999) is that the problem of obstacle avoidance will never be solved. Indeed, mobile robots are becoming more and more capable, and are evolving with increasing speed; these robots will thus need to observe farther and higher areas. Our proposed method for road obstacle detection and tracking combines two dissimilar sensor measurements to achieve a robust performance. It uses a laser based 3D-sensor (Laser Mirror Scanner LMS-Z210-60 from Riegler) which measures range and angles, combined with a radar sensor which delivers range and range rate. For the laser sensor, since target motion is best described in Cartesian coordinates but measurements are available in sensor coordinates, a commonly used method is to convert measurements from sensor to Cartesian coordinates (Li & Jilkov, 2001). Thus, we combine Cartesian target coordinates  $x$  and  $y$  with radar range  $r$  and radar range rate  $\dot{r}$  target measurements. Given the characteristics of the radar sensors (Blanc et al., 2004) we can affirm that radar data are complementary with all the other data. Indeed, the radar is insensitive to atmospheric conditions, thus it is judicious, even essential, to use such a sensor for obstacle detection in road environment. These sensors are installed in VELAC (LASMEA's Experimental Vehicle), see Fig. 1.

*Lidar sensor* : the 3D-Laser Mirror Scanner LMS-Z210-60 is a surface imaging system based upon accurate distance measurement by means of electro-optical range measurement and a two axis beam scanning mechanism. The range finder system is based upon the principle of time-of-flight measurement of short laser pulses in the infrared wavelength region. Many methods for time-of-flight's calculation are described in (Hancock, 1999). The task of the



Fig. 1. LASMEA's experimental vehicle exteroceptive sensors.

scanner mechanism is to direct the laser beam for range measurement in an accurately defined position. The 3D images are configurable. In our approach 20 lines x 103 pixels images at nearly 2 Hz are used (see Fig.2). The line scan mechanism (rotating polygonal four facets mirrors) provides a scan angle range about  $60^\circ$  fixed at a speed of 5 lines/s up to maximum 90 lines/s with an angle step width included between  $0.072^\circ$  and  $0.36^\circ$  and a readout accuracy of  $0.036^\circ$ . The frame scanner mechanism which is slower ( $1^\circ/s$  up to max  $20^\circ/s$ ) than the line scan relies on rotating the optical head together with the fast line scan. This is accomplished by mounting both the line scanner mechanism and the optical head on a rotating table ( $0^\circ$  up to max.  $333^\circ$ ). The angle step width is  $0.072^\circ$  to  $0.36^\circ$  with an angle readout accuracy of  $0.018^\circ$ . For the obstacle detection, a two parts detection algorithm is used: first the segmentation of the 3D image in regions and second the recognition of the obstacle (particularly road vehicles) among these regions. A region growing algorithm is used to perform the segmentation of the 3D image. A region, including shots located at nearly the same distance  $d$  with a tolerance  $\Delta d$ , is parameterized by a vector which includes the size of the target and the position of target's center in the laser scanner reference. These characteristics are then compared to a car model. If parameters of a region are close to those of the model, this region is declared as an obstacle. Finally, a measurement vector  $z^{(c)} =$

$\begin{pmatrix} x^{(c)} \\ y^{(c)} \end{pmatrix}$  and its associated covariance  $R^{(c)} = \begin{pmatrix} \sigma_{x^{(c)}}^2 & 0 \\ 0 & \sigma_{y^{(c)}}^2 \end{pmatrix}$  are constructed.

After detection of different obstacles, we are able to track them in consecutive frames using a constant velocity Kalman filter and a nearest neighbor standard data association method. Each target is characterized by a state vector  $x^{(c)}$  and its associated covariance  $P^{(c)}$ . It is noticed that we are able to detect and track several types of obstacles (cars and trucks) (Blanc et al., 2005). The data association system based on research of nearest neighbor seems sufficient for this system. It is thus not necessary to use methods of type JPDAF or with multiple assumptions. Moreover, the precision of the lidar measures allows data association to easily integrate observation which corresponds best to the considered track. We will be able, for example, to use the obstacle size as one of the criteria of associations if several measurements fall into the validation window. The advantage of this method is based on the measurements precision delivered by the lidar and on a high detection probability. Moreover, in a road context, the number of tracks to follow in front of our experimental vehicle is weak. That reduces considerably necessary calculations to the data association systems.



Fig. 2. 3D image and obstacle detection

*Radar sensor:* the key interests to use a Radar in this project are on the one hand the accuracy of the obstacle speed estimate and on the other hand the quality of its information up to 150 m in spite of difficult weather conditions.

Firstly, the radar data are treated to determine the distance and the relative speed of the objects (or obstacles) located in the enlightened space by the Radar beam. The reader can refer to (Blanc et al., 2004) for many details on the radar data processing. Every 8 ms the radar delivers a measurement of time, amplitude, range and an index speed for all echoes. The range gate is  $\delta R = 22.5$  m and an index speed corresponds to a speed of  $\delta v = 0.238$  m/s. In radar measurements, one target can generate several echoes in close range gate as well as neighbor speed samples. A pretreatment is thus necessary in order to gather the echoes emanating from the same target. In a second step, a measurement vector  $z^{(r)} = \begin{pmatrix} r^{(r)} \\ \dot{r}^{(r)} \end{pmatrix}$

and its covariance matrix  $R^{(r)} = \begin{pmatrix} \sigma_{r^{(r)}}^2 & 0 \\ 0 & \sigma_{\dot{r}^{(r)}}^2 \end{pmatrix}$  are associated to each resulting target.

The Radar tracking is based on Kalman filter and yields to a more accurate range estimate than the gate value (22.5 m) (see Fig. 3). Each target is characterized by a state vector  $x^{(r)}$  and its associated covariance  $P^{(r)}$ .

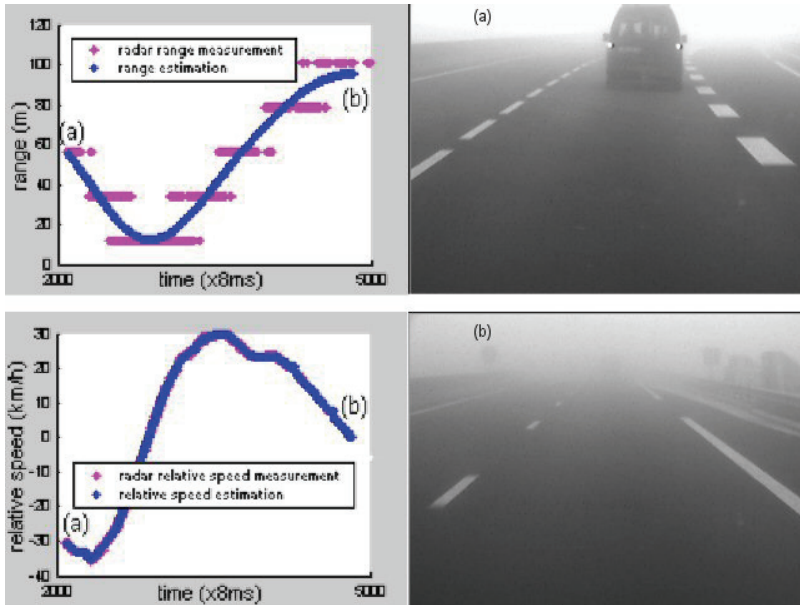


Fig. 3. Radar tracking results in foggy conditions

Obviously, all the sensors have different sample rates. Hypotheses made in this paper are:

- sensors are synchronous
- radar sample rate  $T_r$  is lower or equal to the other one  $T_c$  (lidar), e.g.  $T_c = n \times T_r$  with  $n \in \mathbb{N}^+$ ;

Moreover, to avoid an additional level of complexity due to false detections and multiple target scenarios, we assume:

- unity probability of detection and zero probability of false alarm;
- all sensor measurements are associated for each target: we don't address the problem of data association.

Thus, the aim of the fusion center is to estimate Cartesian target state  $X$  and  $P = cov(X)$ , its associated covariance, with respect to every  $T_r$ , using all available information  $Y$ . Estimation fusion can be classified into three categories, depending on which information is available at the fusion module (Li et al., 2003):

- $Y = \{\{z^{(c)}, R^{(c)}\}, \{z^{(r)}, R^{(r)}\}\}$ : centralized fusion (CF) if all measurements are available at the fusion center.
- $Y = \{\{x^{(c)}, P^{(c)}\}, \{x^{(r)}, P^{(r)}\}\}$ : decentralized fusion (DF) if local estimates are available at the fusion center
- hybrid fusion if available information at the fusion center includes both unprocessed data from one sensor and processed data from the other one.

In this chapter, we focus on centralized and decentralized fusion.

Moreover, between two available Cartesian measurements, the problem can be seen as a problem of target tracking with range-only measurements, i.e. range and range rate measurements. According to our bibliographical research, few publications are devoted to



this problem (Song, 1999) (Ristic et al., 2002). In (Song, 1999) the author discusses the conditions for target observability from range-only measurements. He concludes the same condition as the observability criterion for the related and extensively studied problem of bearing-only target motion analysis: if the target is moving at a constant velocity, the observer must be moving with a non-zero acceleration or if the target is moving at a constant acceleration, the observer must be moving with a non-zero jerk in order to observe the target. In (Ristic et al., 2002), the authors show that, for a typical scenario, tracking algorithms based on range and range-rate measurements can converge toward a steady state.

## 2.2 Mathematical formulation

Let us consider the fusion target state vector:

$$X = (x, \dot{x}, y, \dot{y})^t \quad (1)$$

where  $x, y$  are the tracked positions in the reference frame which is common to both sensors, and  $\dot{x}, \dot{y}$  the tracked relative speed. Evolution model can be represented in a matrix form by:

$$X_{k+1} = F_k X_k + G_k V_k, G_k V_k \sim N(0, Q_k) \quad (2)$$

where  $F_k$  is the transition matrix which models the evolution of  $X_k$ , and  $Q_k$  the covariance matrix of  $V_k$  which represents the acceleration.

$$F_k = \begin{pmatrix} 1 & t_k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & t_k \\ 0 & 0 & 0 & 1 \end{pmatrix}, Q_k = G_k \begin{pmatrix} \sigma_{ax}^2 & 0 \\ 0 & \sigma_{ay}^2 \end{pmatrix} G_k^t, G_k = \begin{pmatrix} \frac{t_k^2}{2} & 0 \\ t_k & 0 \\ 0 & \frac{t_k^2}{2} \\ 0 & t_k \end{pmatrix} \quad (3)$$

The available information at time  $t_k$  is defined by:

- Centralized fusion:  
if  $z^{(c)}$  is available, i.e.  $t_k = n \times T_c$

$$z_k = \begin{pmatrix} z^{(c)} \\ z^{(r)} \end{pmatrix} = \begin{pmatrix} x_k^{(c)} \\ y_k^{(c)} \\ r_k^{(r)} \\ \dot{r}_k^{(r)} \end{pmatrix} = h^{CF}(X^k) + w_k^{CF} = \begin{pmatrix} h_x(X_k) \\ h_y(X_k) \\ h_r(X_k) \\ h_{\dot{r}}(X_k) \end{pmatrix} + w_k^{CF}, w_k^{CF} \sim N(0, R_k^{CF}) \quad (4)$$

$$R_k^{CF} = \begin{pmatrix} R_k^{(c)} & 0 \\ 0 & R_k^{(r)} \end{pmatrix} \quad (5)$$

else

$$z_k = \begin{pmatrix} z^{(r)} \end{pmatrix} = \begin{pmatrix} r_k^{(r)} \\ \dot{r}_k^{(r)} \end{pmatrix} = h^{CF}(X^k) + w_k^{CF} = \begin{pmatrix} h_r(X_k) \\ h_{\dot{r}}(X_k) \end{pmatrix} + w_k^{CF}, w_k^{CF} \sim N(0, R_k^{CF}) \quad (6)$$

$$R_k^{CF} = (R_k^{(r)}) \quad (7)$$

- Decentralized fusion  
if  $x^{(c)}$  is available, i.e.  $t_k = n \times T_c$

$$z_k = \begin{pmatrix} x^{(c)} \\ x^{(r)} \end{pmatrix} = \begin{pmatrix} \tilde{x}_k \\ \tilde{x}_k \\ \tilde{y}_k \\ \tilde{y}_k \\ \tilde{r}_k \\ \tilde{r}_k \end{pmatrix} = h^{DF}(X^k) + w_k^{DF} = \begin{pmatrix} h_x(X_k) \\ h_{\dot{x}}(X_k) \\ h_y(X_k) \\ h_{\dot{y}}(X_k) \\ h_r(X_k) \\ h_{\dot{r}}(X_k) \end{pmatrix} + w_k^{DF}, w_k^{DF} \sim N(0, R_k^{DF}) \quad (8)$$

$$R_k^{DF} = \begin{pmatrix} P_k^{(c)} & P_k^{(cr)} \\ P_k^{(cr)} & P_k^{(r)} \end{pmatrix} \quad (9)$$

where  $P_k^{(cr)}$  is the cross covariance matrix  
else

$$z_k = (x^{(r)}) = \begin{pmatrix} \tilde{r}_k \\ \tilde{r}_k \end{pmatrix} = h^{DF}(X^k) + w_k^{DF} = \begin{pmatrix} h_r(X_k) \\ h_{\dot{r}}(X_k) \end{pmatrix} + w_k^{DF}, w_k^{DF} \sim N(0, R_k^{DF}) \quad (10)$$

$$R_k^{DF} = (P_k^{(r)}) \quad (11)$$

$$\left\{ \begin{array}{l} h_x(X_k) = x_k \\ h_{\dot{x}}(X_k) = \dot{x}_k \\ h_y(X_k) = y_k \\ h_{\dot{y}}(X_k) = \dot{y}_k \\ h_r(X_k) = \sqrt{x_k^2 + y_k^2} \\ h_{\dot{r}}(X_k) = \frac{x_k^2 \dot{x}_k^2 + y_k^2 \dot{y}_k^2}{\sqrt{x_k^2 + y_k^2}} \end{array} \right. \quad (12)$$

### 3. Posterior Cramer-Rao lower bounds

#### 3.1 Derivation of the bounds

The system defined by both the evolution and the measurement model, respectively defined in (2) and (4,5) is considered. If  $\hat{X}_{k/k}$  is an unbiased estimator of  $X_k$ , calculated from the measurement sequence  $Z_k = \{z_1, \dots, z_k\}$  and from the knowledge of  $p(X_0)$  (initial pdf), then the covariance matrix of  $\hat{X}_{k/k}$ , noted  $P_{k/k}$  admits a lower bound given by:

$$P_{k/k} \triangleq E\{(\hat{X}_{k/k} - X_k)(\hat{X}_{k/k} - X_k)^t\} \geq J_k^{-1} \quad (13)$$

where  $J_k$  is the Fisher information matrix which we want to determine. Tichavsky et al. (Tichavsky et al., 1998) proposed a method to calculate  $J_k$  recursively:

$$J_{k+1} = D_k^{22} - D_k^{21}(J_k + D_k^{11})^{-1}D_k^{12} \quad (14)$$

where since the evolution model is linear and noises are Gaussian:

$$D_k^{11} = F_k^t Q_k^{-1} F_k \quad (15)$$

$$D_k^{12} = -F_k^t Q_k^{-1} = [D_k^{21}]^t \quad (16)$$

$$D_k^{22} = \begin{cases} Q_k^{-1} + E \left\{ [\tilde{H}_{k+1}^{CF}]^t [R_{k+1}^{CF}]^{-1} \tilde{H}_{k+1}^{CF} \right\} & \text{if } CF \\ Q_k^{-1} + E \left\{ [\tilde{H}_{k+1}^{DF}]^t [R_{k+1}^{DF}]^{-1} \tilde{H}_{k+1}^{DF} \right\} & \text{if } DF \end{cases} \quad (17)$$

where  $\tilde{H}_{k+1}^{CF}$  and  $\tilde{H}_{k+1}^{DF}$  are respectively the Jacobian matrix of  $h_{k+1}^{CF}(X_k)$  and  $h_{k+1}^{DF}(X_k)$  evaluated at the true value of  $X_{k+1}$ . Finally, by using the inversion matrix lemma, the recursive information matrix calculation is:

$$J_{k+1} = (Q_k + F_k J_k^{-1} F_k^t)^{-1} + \begin{cases} E \left\{ [\tilde{H}_{k+1}^{CF}]^t [R_{k+1}^{CF}]^{-1} \tilde{H}_{k+1}^{CF} \right\} & \text{if } CF \\ E \left\{ [\tilde{H}_{k+1}^{DF}]^t [R_{k+1}^{DF}]^{-1} \tilde{H}_{k+1}^{DF} \right\} & \text{if } DF \end{cases} \quad (18)$$

In practice, the most difficult problem is the calculations of the expected value operator  $E$  in (18). The expectation is only taken with respect to the state vector  $X_k$  (the bound is independent of the actual measurement sequence). A Monte Carlo approximation can be applied to implement the theoretical PCRB formulae. One first needs to create a set of state vector realizations, the so-called target trajectories. Then the appropriate term in (12) is computed as the average over this set.

The recursions start with the initial information matrix  $J_0$  computed from the initial density  $p(X_0)$ . If  $p(X_0)$  is Gaussian then  $J_0 = P_0^{-1}$  else  $J_0 = E \left\{ \Delta_{X_0}^{X_0} \log p(X_0) \right\}$ .

### 3.2 Analysis of the bounds

The scenario used is constructed from a ground truth approximation. It is a typical case of adaptive cruise control (ACC) scenario. For this, VELAC and only one obstacle are equipped with DGPS. Their locations are acquired every second. Obstacle position is sent to VELAC by MF communications. Relative positions and velocities are shown in Fig. 4 after approximation. VELAC and obstacle move at a speed bounded by 20km/h and 90km/h.

As described in (Blanc et al., 2007) it is not necessary to stack  $r$  in the  $Z_k$  measurement vector except if  $\sigma_r \ll \sigma_x, \sigma_y$  (with our sensors  $\sigma_r > \sigma_x, \sigma_y$ ). Firstly, the Fig. 5 confirms, as expected, that the more data are available the more performance increases. As we can see, PCRLBs of  $x$  and  $\dot{x}$  increase between two Cartesian measurements as it the trajectory taken does not respect the observability criterion. For the calculations 100 MC runs are used and  $R^{(c)} = \begin{pmatrix} 2^2 & 0 \\ 0 & 2^2 \end{pmatrix}$ ,  $R^{(r)} = \begin{pmatrix} 7^2 & 0 \\ 0 & 0.2^2 \end{pmatrix}$ . Moreover, decentralized and centralized estimation fusion architectures have an equivalent good performance for this ACC scenario.

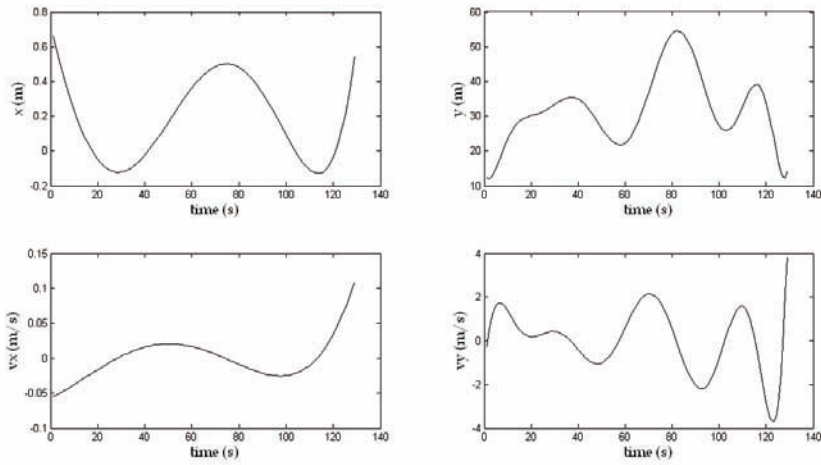


Fig. 4. A scenario used for the analysis of Posterior Cramer-Rao lower bounds

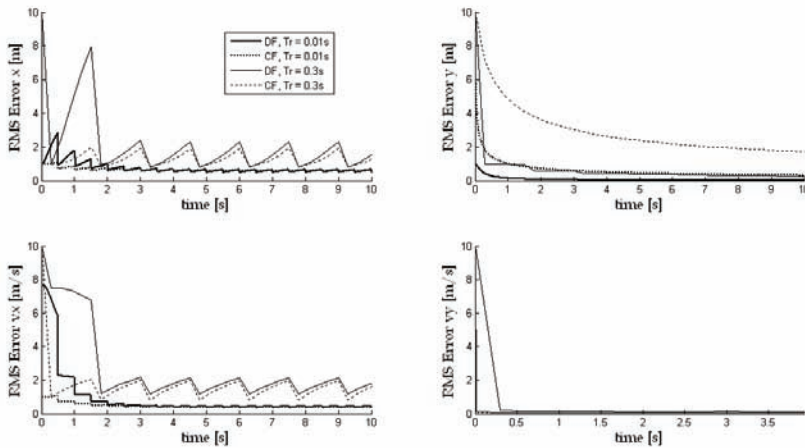


Fig. 5. Decentralized fusion (solid line) and centralized fusion (dashed line) PCRLB of  $x, \dot{x}, y, \dot{y}$  with a fixed  $T_c = 0.5 s$ , a varying  $T_r = 0.01 s, 0.3 s$

## 4. Tracking algorithms

### 4.1 Extended Kalman filter

Estimation fusion using range, range-rate and Cartesian measurements, is a non-linear dynamic state estimation problem because the measurement equation is non-linear (12). The Kalman filter is therefore inappropriate. The conventional approach is to approximate Eqs. (4, 6, 8, 10) by a series expansion and then to use an equivalent measurement matrix in the ordinary Kalman filter equations: Extended Kalman filter (EKF) for a first-order series

expansion (linearization) of the non-linear measurement equation. The recursive equations of the EKF are presented below to describe the evaluation of the relative estimated state  $X_{k+1/k+1}$  and its associated covariance matrix  $P_{k+1/k+1}$  by using the measurement  $z_k$  and the relative state vector  $X_{k/k}$  with its associated covariance matrix  $P_{k/k}$ . The state equation prediction is worked out by Eq. (2) while the covariance matrix is given by:

$$P_{k+1/k+1} = F_k P_{k/k} F_k^t + G_k Q_k G_k^t \quad (19)$$

The measurement prediction is given by Eqs. (4, 6, 8, 10). We don't stack  $r$  in the measurement vector for Eqs. (4, 8) as it is proposed in previous section. The Kalman gain matrix can be evaluated as

$$K_{k+1}^m = P_{k+1/k+1} (H_{k+1}^m)^t [H_{k+1}^m P_{k+1/k+1} (H_{k+1}^m)^t + R_{k+1}^m]^{-1} \quad (20)$$

where  $m$  represents CF or DF, and  $H_{k+1}^m$  is the linearised measurement matrix evaluated at the predicted state.

Finally, the updated fusion state and its associated covariance matrix are given by:

$$X_{k+1}^m = X_{k+1/k}^m + K_{k+1}^m (z_{k+1} - h^m(X_{k+1/k}^m)) \quad (21)$$

$$P_{k+1}^m = (I - K_{k+1}^m H_{k+1}^m) P_{k+1/k}^m \quad (22)$$

## 4.2 Particle filter

Originally developed in the tracking community (Gordon et al., 1993), the particle filtering is currently enjoying a strong development in many research fields (vision, localization, navigation, robotics, etc.), in particular in multi-target tracking. This filter is a sequential Monte-Carlo method in which particles traverse the state space in an independent way, and interact under the effect of a probability function which automatically concentrates the particles in the state space areas of interest. This method has the advantage of not requiring linear or Gaussian assumptions on the model. Moreover, it is very easy to implement, since it is enough to know how to simulate independent various trajectories of the model. We propose here, to carry out a particle filtering on the fusion module level. A fusion state is initialized. From this vector, a set of  $N_s$  particles is built. Noise particles are generated ( $B_0^{(i)}, i = 1, \dots, N_s$ ) and applied to the initial vector:

$$X_{0/0}^{m(i)} = X_{0/0}^m + B_0^{(i)}, \forall i \in [1 \dots N_s] \quad (23)$$

Then, the model defined in (2) is applied to  $N_s$  particles in a prediction step. Correction is carried out on the level of the calculation of the weights. We calculate  $N_s$  weights assigned to the  $N_s$  predicted particles. We have:

$$w_k^{m(i)} = p(z_k / X_{k/k-1}^{m(i)}) \quad (24)$$

Weights are then normalized, and finally the fused state considered is given by:

$$X_k^m = \sum_{i=1}^{N_s} w_k^{m(i)} X_{k/k-1}^{m(i)} \quad (25)$$

and its covariance by:

$$P_k^m = \sum_{i=1}^{N_s} w_k^{m(i)} \left( X_{k/k-1}^{m(i)} - X_{k/k-1}^m \right) \left( X_{k/k-1}^{m(i)} - X_{k/k-1}^m \right)^t \quad (26)$$

The particles are resampled and returned to the prediction step.

### 4.2 Algorithm performance and comparison

The performance of estimation fusion for two algorithms is analyzed by Monte-Carlo simulations. The analysis is made for the trajectory presented in Fig. 4. The measurement covariances are assumed to be  $R^{(r)} = \text{diag}[7^2, 0.2^2]$  and  $R^{(c)} = \text{diag}[2^2, 2^2]$ .

The sampling rates are  $T_r = 0.01 \text{ s}$  and  $T_c = 0.5 \text{ s}$ . The resulting error curves were computed to the theoretically derived PCRLB. The estimation error is defined as  $e_k = X_k^m - X_k$ . The performance is measured by the root mean square error (RMSE), which, for component  $j$  of the state vector, is defined as  $\sigma_k^j = \sqrt{E(e_k^j)^2}$  where the expectation operator was computed by averaging over 50 independent Monte-Carlo runs. Results are shown in Fig. 6.

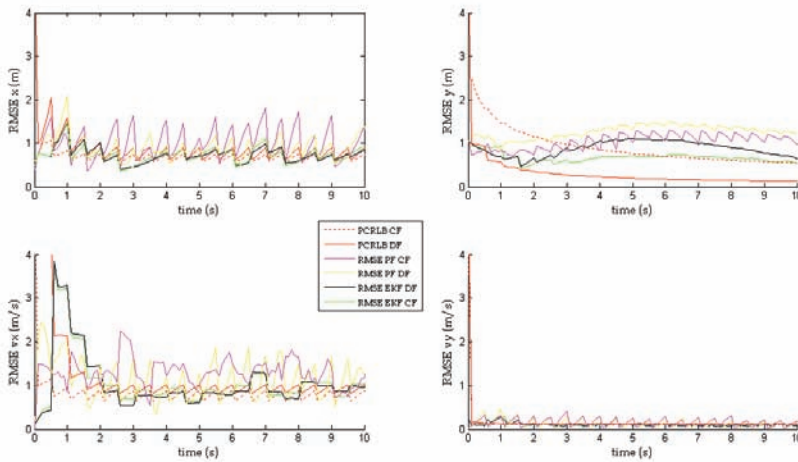


Fig. 6. Performance of EKF and PF against PCRLB

Firstly, we may notice that the EKF and particle filter errors are initially smaller than the square root of PCRLB which is unexpected. This result is due to the initialization, which does not exactly match to the inverse of the initially used information matrix in the PCRLB computation. For clarity, results for particle filter are present only with 2000 particles. Nevertheless, we observe that as the number of particles is increased, the performance of the particle filter improves, and approaches the PCRLB's. This improved accuracy of the particle filter, however, is at the expense of the computational load. Particle filter with 10000 particles is equivalent to EKF. Thus, in practical operation system, the EKF appears more suitable for implementation than the particle filter.

## 5. Experiments

Quantitative results for decentralized fusion are obtained with a ground truth. Velac and only one obstacle are equipped with DGPS. Their locations are acquired every second. Obstacle position is sent to Velac by MF communication. In the same time, the obstacle is tracked by both processes (Radar/Lidar). Decentralized fusion process and comparison are done offline. The measurement system includes a differential GPS Omnistar which delivers trames with format TSIP (Trimble Standard Interface Protocol). This GPS gives, in the best configuration, a position with a  $\pm 40$  cm accuracy. It delivers a coefficient, called *gdop*: the current accuracy is *gdop* times 40 cm. DGPS errors are shown in Fig. 7. Moreover, for data communication, a radio modem of Satel receives trames coming from the obstacle. Fig. 7 shows the results for the range estimation by extended Kalman filter and particle filter. Moreover, it shows the radar and lidar estimate. We see that radar estimate is less accurate than lidar estimate in this particular scenario. The DGPS reference allows computing root mean square error and its standard deviation (*std*) for both filters. As it is shown on results, performance of range estimation is correct for both filters. Errors and DGPS accuracy have almost the same order. Moreover, as expected, the EKF have very small computation time compared with the particle filters. Only EKF allows real time utilization because of the radar data rate which is 8 ms.

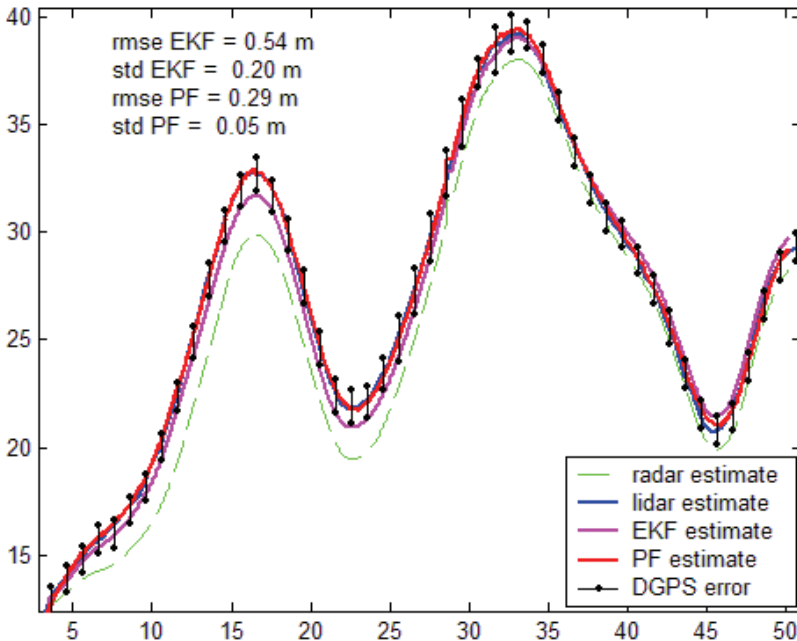


Fig. 7. Range estimation by EKF and particle filter

## 6. Conclusion

This paper has discussed the problem of centralized/decentralized fusion estimation for target tracking using range, range-rate and Cartesian measurements for road obstacle tracking. The Posterior Cramer-Rao Lower Bounds were derived and two tracking algorithms were developed. The PCRLBs allow us to predict the best achievable performance under various conditions such as the relative target trajectory, and measurement sample rate. Two algorithms have been considered: the extended Kalman filter and the particle filter. This study has shown that both algorithms are efficient for this kind of scenario even if the particle filter is unattractive for implementation in a practical operational system. Our future work will consist in developing a new method of laser/camera fusion for pedestrian detection. This work will take place in the context of the LOVE (Logiciel d'Observations des Vulnérables) which aims at improving road safety, mainly focusing on pedestrian security.

## 7. References

- Blanc, C. ; Aufrère, R. ; Malaterre, L. ; Gallice, J. & Alison, J. (2004). Obstacle detection and tracking by millimetre wave radar, *5<sup>th</sup> IFAC Symposium on Intelligent Autonomous Vehicles IAV*, Portugal, July 2004, Lisboa
- Blanc, C. ; Trassoudaine, L. & Gallice, J. (2005). EKF and particle track-to-track fusion: a quantitative comparison from radar/lidar obstacle tracks, *8<sup>th</sup> International Conference on Information Fusion*, USA, July 2005, Philadelphia
- Blanc, C. ; Checchin, P. ; Gidel, S. & Trassoudaine, L. (2007). Data fusion performance evaluation for range measurements combined with Cartesian ones for road obstacle tracking, *International Conference on Vehicular Electronics and Safety*, China, December 2007, Beijing
- Gordon, N.J. ; Salmond, D.J.; Smith, A.F.M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, Vol. 140, No.2, (April 1993) 107-113, ISSN 0956-375X.
- Hancock, J. (1999). Laser Intensity-Based Obstacle Detection and Tracking, *PhD Thesis*, Carnegie Mellon University, The Robotics Institute
- Li, X.R. & Jilkov, V.P. (2001). A Survey of Manoeuvring Target Tracking Part III, *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, USA, July-August 2001, San Diego
- Li, X.R.; Yunmin, Z. ; Jie, W. & Chongzao, H. (2003). Optimal linear estimation fusion .I. Unified fusion rules. *IEEE Transactions on Information Theory*, Vol.49, No.9, (September 2003) 2192-2208, ISSN 0018-9448
- Ristic, B.; Arulampalam, S. & McCarthy, J.(2002). Target motion analysis using range-only measurements: algorithms, performance and application to ISAR data. *Signal Processing*, Vol. 82, No.2, (February 2002) 273-296, ISSN 0165-1684
- Song, T.L. (1999). Observability of target tracking with range-only measurements. *IEEE Journal of Oceanic Engineering*, Vol. 24, No.3, (July 1999) 383-387, ISSN 0364-9059.



Tichavsky, P.; Muravchik, C.H. & Nehorai, A. (1998). Posterior Cramer-Rao bounds for discrete-time nonlinear filtering. *IEEE Transactions on Signal Processing*, Vol. 46, No.5, (May 1998) 1386-1396, ISSN 1053-587X.

# IR Barrier Data Integration for Obstacle Detection

J. Jesús García, Jesús Ureña, Manuel Mazo and Álvaro Hernández  
*University of Alcalá  
Spain*

## 1. Introduction

The great development of the railway as a mean of transport makes necessary more and more reliable the required safety systems. Among these systems, those called detectors of the fall of objects can be remarked, mainly demanded in the high-speed lines, to detect the existence of objects on the track supposing a risk (larger than 50x50x05cm) for the railway circulation in some specific areas such as tunnels and overpasses (GIF, 2001). This type of systems is based on different sensory elements; in (GIF, 2001) it is proposed one based on infrared (IR) barriers, where links are established between emitters and receivers. The detection of objects is carried out by the interruption of these links. But this system can be lacking in reliability, mainly due to outdoor problems related to IR sensors.

In this chapter, it is proposed a new system, based on an IR barrier as well, but including an algorithm to validate the existence of obstacles, increasing the reliability of the detection system. First of all, to be able to discriminate the different emissions in a receiver, it is necessary to encode them. The detection of the emissions is carried out by correlation (Tseng, Shu-Ming & Bell, 2000), where a detection threshold is defined to evaluate if the link is active or not. Because the detection of objects is based on the radiation lack at the receivers, this circumstance does not always imply the existence of a dangerous object for the railway, generating false alarms. The adverse climatology also gives the degradation of the optical channel, so the radiation lack in these circumstances can be confused with the existence of objects.

Furthermore, in some cases small objects can interrupt the links (falling leaves, small animals, etc.), or even a wrong operation of a sensor can be confused with the existence of an object, generating false alarms. To avoid these situations, in this work different procedures are proposed to conclude if there are dangerous objects in the scanned area.

Therefore, the starting point is the information provided by the IR receivers: the result of the correlation, and a threshold output indicating if the link is active. The information about the state of the links (*on* or *off*) is combined by means of the Dempster-Shafer evidential theory (Klein, 2004) taking into account the channel degradation and the spatial diversity of the sensory system structure, so a certainty value can be obtained about the existence objects larger than 50x50x50 cm.

Regarding this problem, the following aspects will be analyzed in this chapter: description of the structure of the sensory system; main factors of infrared channel degradation; a brief explanation about the proposed encoding scheme and the detection of the emissions; sensor

data fusion to obtain the existence certainty of dangerous obstacles based on Dempster Shafer's rule; real tests; and finally, the most relevant conclusions.

## 2. Description of the sensory system

### 2.1 Sensory system

For the application described in the previous section, either infrared or laser emitters could be used. Irrespective of the sensor type chosen, all the details that will be discussed below can be applied to both types. The choice of the system may depend on financial considerations. In this work, the results shown have been obtained using infrared emitters, mainly because this kind of sensor was required by Spanish Railway Regulations (GIF, 2001).

Infrared barriers usually consist of emitter-receiver pairs, each placed on opposing sides of the line, so it is only possible to detect the presence of an obstacle, but not its exact position. In order to detect obstacles on the railway, and distinguish at least vital areas (on the track) from the non-vital areas (to the side of the track), a special structure has been designed. In this case, every emission is detected by several receivers, providing different optical links among the emitters and the receivers as is shown in Figure 1.

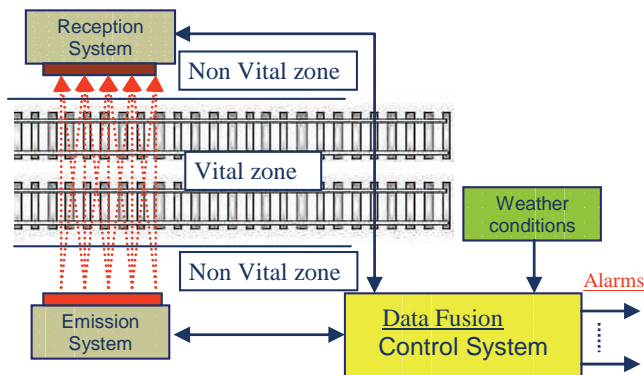


Fig. 1. Detection system placed on a section of track.

The distance between emitting sensors is 25 cm, in order to detect 50x50x50cm obstacles successfully (the size is determined by railway regulations) (GIF, 2001). The configured distance between emitters and receivers is 14 meters on a high-speed line. Basically, the method of obstacle detection, and its location on the railway, is based on the lack of reception by detectors. According to (GIF, 2001) the time scan of the system is 500 ms, and if an obstacle is inside the detection area more than 3 seconds, an alarm should be generated. For a more detailed study about the sensory system see (García et al., 2004). Figure 2 shows the scheme of the infrared barrier, and how when there a minimum dimension object, at least two links are interrupted.

### 2.2 Geometric distribution of the sensors

Taking into account the infrared emitter beam angle ( $\approx \pm 2^\circ$ ), if the range is 14 meters, every emission reaches five receivers, as Figure 3 shows; and reciprocally, every receiver has to detect five emissions.

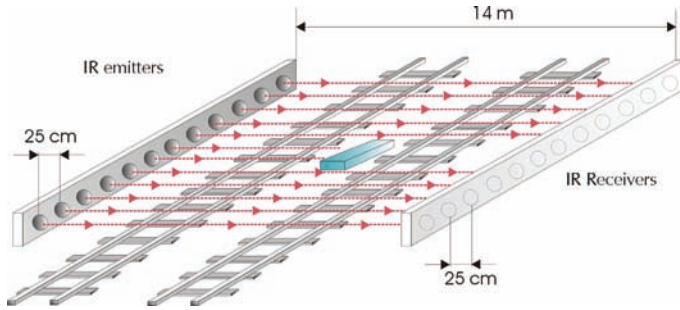


Fig. 2. Scheme of the infrared barrier.

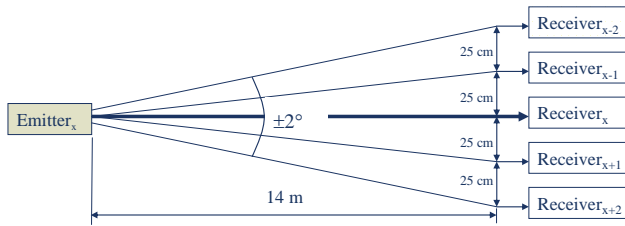


Fig. 3. Emitter-receivers links.

Figure 4 shows the distribution of sensors at a 2.25m segment of the barrier, displaying the five links that every emitter provides. Regarding the sensors distribution, and as it will be explained in Section 4, it is necessary to use five different codes to distinguish every emission in a receiver.

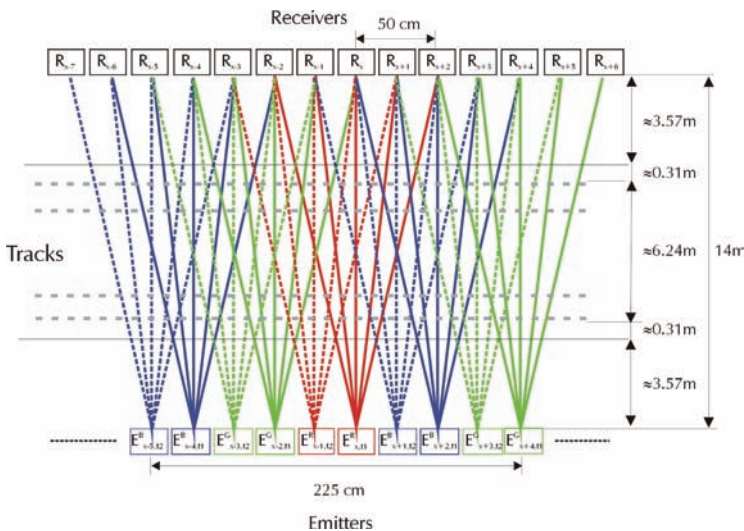


Fig. 4. Distribution of emitters and receivers.

There are some improvements with the structure shown in Figure 4. On the one hand, if there is a minimum dimension obstacle in the supervised area, at least ten links are

interrupted. In (GIF, 2001, 2004) only two interrupted links are required to detect the obstacle. On the other hand, due to the fact that detection is based on link interruption, if a sensor is not working, it can be mistaken for the detection of one obstacle. The number of interrupted links allows the system to distinguish the presence of an obstacle from an out of order sensor, being of particular advantage to those carrying out maintenance tasks. Finally, the obstacles can be located with this structure in two main zones: on the tracks or outside of the tracks.

### 3. Infrared channel degradation

The outdoor infrared emission suffers from diverse losses, which can produce a wrong detection in the receiver. If the receiver does not detect one emission during a predefined time, an alarm will be generated, informing that there is an obstacle. But if the obstacle does not exist, the alarm is actually false. As far as possible, it is necessary to avoid the false alarms generation. So, it is very important to consider those circumstances that can produce infrared channel degradation.

In these outdoor optical systems there are some phenomena that can provide false alarms, mainly the weather condition and the solar radiation. There are other reasons, as propagation losses or wrong alignment among emitters and receivers. Assuming that the last ones have been already considered in the link design, only the two first are described below:

*Atmospheric attenuation.* Snow, fog and rain are considered. Although there are numerous studies about the losses due to the meteorology, the expression (1) is used to quantify atmospheric attenuation (Yokota et al., 2002).

$$L_{attm} (dB) = \frac{13}{V} \cdot R \quad (1)$$

where  $V$  is the visibility in kilometres and  $R$  the link range in kilometres. The technical definition of visibility or visual range is the distance that light decreases to 2% of the original power, or qualitatively, visibility is the distance at which it is just possible to distinguish a dark object against the horizon (Kim et al., 2000). Table 1 shows the relation between weather condition and the visibility.

Visibility $V$	Weather condition
$V > 50\text{km}$	Very clear
$6\text{km} < V < 50\text{km}$	Clear
$1\text{km} < V < 6\text{km}$	Haze / snow / light rain
$0.5\text{km} < V < 1\text{km}$	Light fog / snow / heavy rain
$V < 0.5\text{km}$	Thick fog

Table 1. Relation between visibility and weather condition.

Taking into account (1), in a 14 meters link (the distance among emitters and receivers in the obstacle detection system), atmospheric attenuation  $\theta_k$  in the time instant  $k$  is

$$\theta_k = 10^{-\frac{18.5}{V_k}} \quad (2)$$

where  $V_k$  is the value of the visibility in the instant  $k$ . Some authors use a different expression (Kim et al., 1998) for atmospheric losses, taking into account the wavelength of the emission and the size distribution of the scattering particles, related to the visibility. Nevertheless, the obtained results with both expressions are similar.

*Solar interference.* As the photodiode wavelength (850nm) is inside the solar spectrum, natural background light can potentially interfere with signal reception. In some circumstances, direct sunlight may cause link outages for periods of several minutes when the sun is within the receiver's field of view (Bloom et al., 2003). However, the times, when the receiver is most susceptible to the effects of direct solar radiation (either at dawn or at dusk), can be easily estimated. There are some solutions to mitigate this problem, like proper orientation or use of a narrow-bandwidth light filters, but it is almost impossible to avoid them completely. It is important to remember that interference by reflected sunlight is possible as well. The solar effect in the IR barrier is the photodiode saturation.

#### 4. Encoding scheme and detection of the emissions

As has been previously described, a multi-mode operation is carried out in the barrier (simultaneous multi-emission and multi-reception). Therefore it is necessary to encode every emission in order to avoid interferences among the different emissions and to discriminate them at the receiving block.

Due to the fact that the obstacle detection is based on a lack of signal in the receiver, sometimes it can be produced by atmospheric attenuation. Furthermore, solar interference is a high source of noise that can make impossible to distinguish the emissions in the reception system, producing false alarms.

For the mentioned reasons, it is necessary to choose an encoding scheme that permits the multi-mode operation under low signal-to-noise ratios. Trying to answer these requirements, mutually orthogonal complementary sets of sequences have been used (Tseng, Shu-Ming & Bell, 2000)(Chow, 2003)(De Marziani, 2007) for encoding the emissions. More details about the codification scheme and the adaptation to the infrared sensor can be found in (Diaz et al., 2007).

In Figure 4, every emitter uses a different code (shown by different colour in the diagram). The detection of the different emissions is carried out by means of a correlation process, where the output of every receiver provides five measurements, corresponding to the correlation values obtained for every link, as Figure 5 shows. Due to the fact that the emitters transmit periodically, the correlation output obtained for every link is a periodic signal as shows (3).

$$y_k^{(j,x)} = G \cdot \theta_k \cdot \sigma_j \cdot \sum_{i=0}^{i=\infty} \delta[k - i \cdot T] + \phi_{\eta,k} \quad (3)$$

In (3),  $G$  is the process gain, according to the encoding scheme (Diaz et al., 2007);  $T$  is the emission period (around 100ms);  $\theta_k$  represents atmospheric attenuation described in (2);  $\sigma_j$  is the lateral attenuation (it depends on the sensor emission pattern (Diaz et al., 2007));  $\phi_{\eta,k}$  is the noise component after the correlation;  $x$  is the position of the receiver in the barrier and  $j \in \{1, 2, 3, 4, 5\}$  represents every link established in a receiver, as shown in Figure 5. The index  $k$  represents the time instant when data are captured.

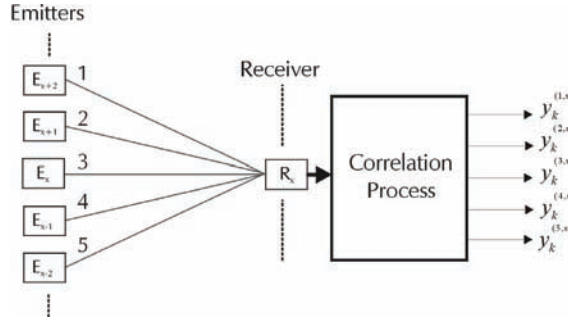


Fig. 5. Block diagram of the detection process.

According to Figure 5, the output of every receiver can be represented as a vector  $\mathbf{y}_k^{(x)}$  of five measurements, as (4) shows

$$\mathbf{y}_k^{(x)} = \begin{bmatrix} y_k^{(1,x)} & y_k^{(2,x)} & y_k^{(3,x)} & y_k^{(4,x)} & y_k^{(5,x)} \end{bmatrix}^T \quad (4)$$

where  $x$  is the position of the receiver in the barrier; and  $k$  represents the time instant. Every component of vector  $\mathbf{y}_k^{(x)}$  is represented in (3). Figure 6 shows the multi-detection carried out in an IR receiver in absence of obstacles. The amplitude differences depend on the lateral deviations between emitters and receiver (see Figures 3 and 4), so the maximum correlation output is provided by the emitter placed in the axial axis.

In (5), the matrix  $\mathbf{Y}_k$  contains the correlation output of all the links in the IR barrier. Every column of matrix  $\mathbf{Y}_k$  is the belonging vector shown in (4), being  $X$  the number of receivers in the barrier.

$$\mathbf{Y}_k = \begin{bmatrix} y_k^{(1,1)} & \dots & y_k^{(1,x)} & \dots & y_k^{(1,X)} \\ y_k^{(2,1)} & \dots & y_k^{(2,x)} & \dots & y_k^{(2,X)} \\ y_k^{(3,1)} & \dots & y_k^{(3,x)} & \dots & y_k^{(3,X)} \\ y_k^{(4,1)} & \dots & y_k^{(4,x)} & \dots & y_k^{(4,X)} \\ y_k^{(5,1)} & \dots & y_k^{(5,x)} & \dots & y_k^{(5,X)} \end{bmatrix}_{5 \times X} \quad (5)$$

To evaluate if an established link  $j$  in the receiver  $x$  is not interrupted, the correlation value  $y_k^{(j,x)}$  should be higher than a determined detection threshold ( $T_H$ ). But when the tracks are free of obstacles, the correlation value could be lower than the threshold due to the channel degradation (as was previously described), and consequently false alarms would be produced. To reduce false alarms, it was proposed in (Garcia et al., 2005) the use of a dynamic threshold,  $T_{H,k}^{(j,x)}$ . This threshold is based on an  $H_\infty$  filter (Simon, 2000), and it is dynamically adapted considering meteorology and solar interference. The  $H_\infty$  filter gives an estimate  $\hat{y}_k^{(j,x)}$  of the correlation value  $y_k^{(j,x)}$  for every link  $j$  in every receiver  $x$  at every time instant, establishing the threshold in time instant  $k$ . But to avoid false alarms, it is necessary

to set a minimum threshold  $T_{H-min}$  depending of the codification scheme and the expected noise. Equation (6) shows how the dynamic threshold is generated. Finally, for every link is generated an output  $z_k^{(j,x)}$  corresponding to its state: *on* or *off*, as shown in (7). Similarly as (5), one matrix  $Z_k$  is obtained representing the state of all the links (1, on; 0, off). Figure 7 depicts the diagram block of a receiver tuned to a generic code C considering the described process.

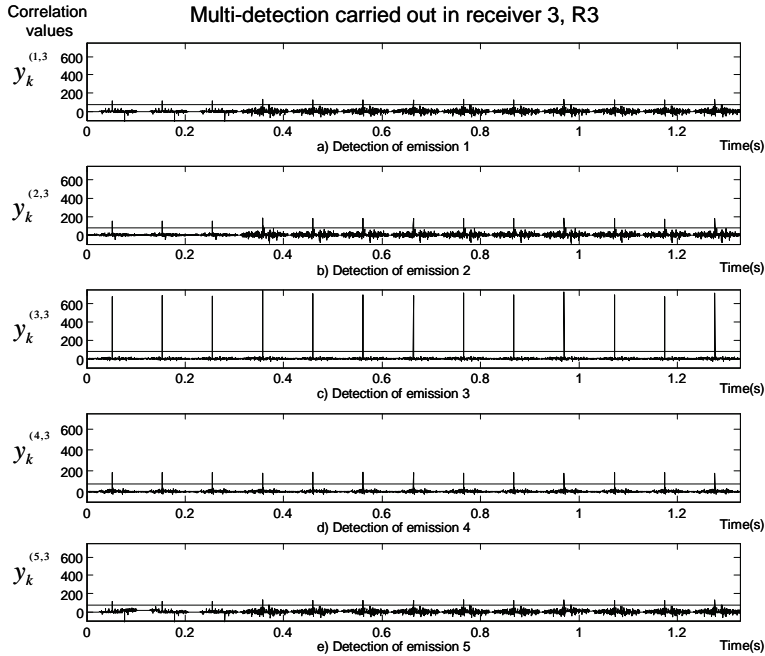


Fig. 6. Real correlation outputs obtained in an IR receiver when the links are not interrupted.

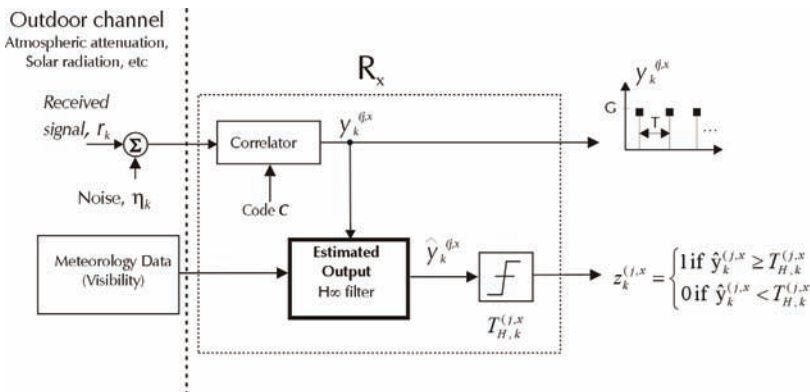


Fig. 7. Block diagram of a receiver Rx tuned to a generic code C.



$$T_{H,k}^{(j,x)} = \max \left( \frac{\hat{y}_{k-1}^{(j,x)}}{2}, T_{H-\min} \right) \quad (6)$$

$$z_k^{(j,x)} = \begin{cases} 1 & \text{if } \hat{y}_k^{(j,x)} \geq T_{H,k}^{(j,x)} \\ 0 & \text{if } \hat{y}_k^{(j,x)} < T_{H,k}^{(j,x)} \end{cases} \quad (7)$$

As it is concluded from (6), threshold in time instant  $k$  depends on the estimated correlation output in time instant  $k-1$ , and the threshold will be adapted to the slow changes of the channel produced by weather conditions or solar radiation. On the contrary, if there exists an obstacle it will generate a fast change of the channel, causing a lack of signal in the receiver, and consequently, a low correlation value. As it is shown in (Garcia et al., 2005), this strategy reduces false alarms due to the channel degradation.

## 5. Sensor data fusion

In a railway environment, typical situations that can generate a false alarm must be identified. Although, the occurrence of false alarms has been notably reduced by using dynamic threshold in the detection stage, it is still possible for some receivers not to detect the emissions because a small object (moving leaves, small animals, etc.) has temporarily interrupted the link, or because weather conditions are severe, or, simply, because the corresponding emitter is damaged. These situations should not cause alarm activations for the existence of objects.

In detection, two pieces of information must be present in every sensor output (Klein, 2004). Firstly, the detection itself; the barriers provides this information in matrices  $\mathbf{Y}_k$  and  $\mathbf{Z}_k$ . Secondly, how well, or with what confidence the sensor has been able to detect an object. For the latter, it is necessary to combine data from different sources, taking into account external variables such as weather conditions or sensor degradation. For the IR barrier, weather conditions can be modeled by considering *visibility*, as has been previously explained. There is a direct relation between visibility and atmospheric attenuation. It is also worth noting that the failure of just one individual link may be considered insignificant, since if a dangerous object exists, at least 10 links will be interrupted, as was described in Section 2.

Considering the above remarks, the data fusion has been carried out at two levels. Firstly, the detection area has been divided into 25cm-wide influence areas according to the receivers, so if a dangerous object exists, it is detected in two consecutive influence areas. The result of this level is a measurement of the certainty of existence of objects in every influence area. Secondly, values for the certainty of the existence of objects belonging to two consecutive influence areas have been fused by means of Dempster-Shafer evidential theory (Klein, 2004) to obtain a final value for the certainty of existence of objects larger than 50x50x50cm. Figure 8 shows a block diagram of the validation of obstacle detection considering the information that provides the barrier.

### 5.1 Certainty of existence of objects in the influence area, $A_{(x)}$

Figure 9 shows the division of the detection area into influence areas. The influence area of the receiver  $R_x$  is represented by  $A_{(x)}$ . After the analysis of the influence area, a value for the certainty of existence of objects is obtained represented by  $c_{A(x)} \in [0, 1]$ .

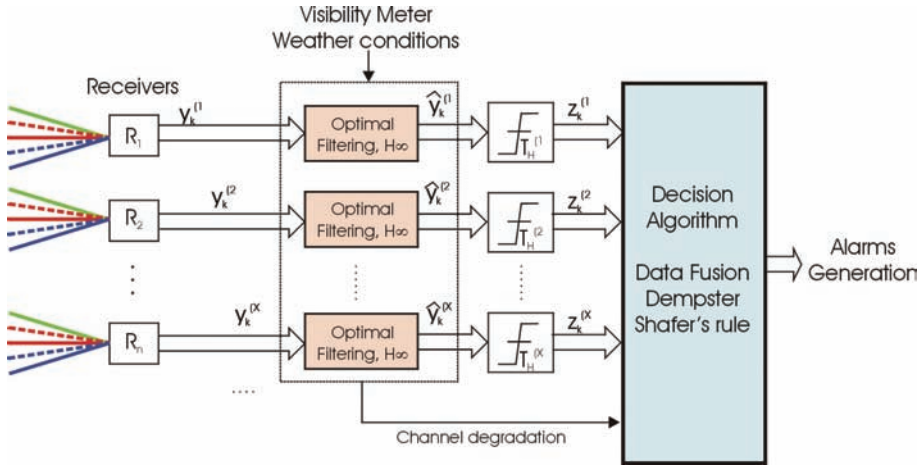


Fig. 8. Validation of obstacle detection.

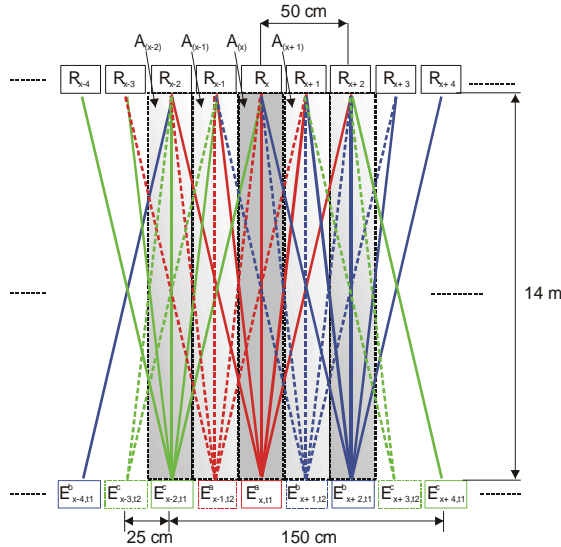


Fig. 9. Influence areas.

As Figure 9 shows, and more detailed in Figure 10, there exist eleven links for every influence area, established between five emitters and five receivers. These links cross several areas, except for the link between the emitter  $x$  and the receiver  $x$ , that only exists in the area  $A_{(x)}$ .

To determine if a link is interrupted, it is only necessary to evaluate the state (*on*, *off*) of the corresponding element in the matrix  $Z_k$ . Due to the fact that the channel degradation can generate a lack of signal in the detector, this situation can be mistaken for the existence of an object. For this reason, if at any  $k$  instant,  $z_k^{j,x}$  was zero –existence of an obstacle–, but a high-level channel degradation occurred at the  $k-1$  instant, it would be very unlikely that the lack

of signal was produced by an object. Therefore, to obtain the certainty of the link interrupted by an object, the link degradation has to be considered.

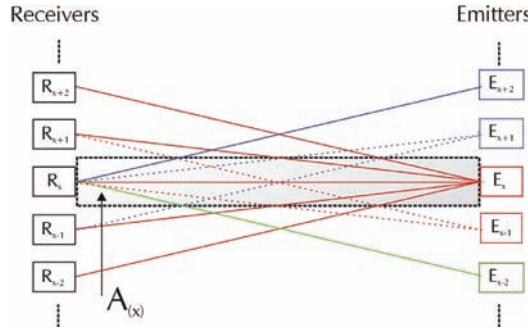


Fig. 10. Links for influence area  $A_{(x)}$ .

Furthermore, the probability of an object interrupting a link in the area  $A_{(x)}$  depends on the percentage of the range of the link placed in such area. For the link between emitter  $x$  and receiver  $x$  (link  $l_{x,x}$ ) the probability is one, but for the rest is 0.5 or 0.25. Table 2 shows the probability  $\rho_{e,r}$  for every link inside a detection area. Sub-indexes  $e$  and  $r$  denote the  $x$  position of the emitter and the receiver respectively.

Index $i$	-5	-4	-3	-2	-1	0	1	2	3	4	5
Link $l_{e,r}$	$l_{x,x-2}$	$l_{x,x-1}$	$l_{x+1,x-1}$	$l_{x-2,x}$	$l_{x-1,x}$	$l_{x,x}$	$l_{x+1,x}$	$l_{x+2,x}$	$l_{x-1,x+1}$	$l_{x,x+1}$	$l_{x,x+2}$
Probability $\rho_{e,r}$	0.25	0.5	0.5	0.25	0.5	1	0.5	0.25	0.5	0.5	0.25

Table 2. Probability of an object interrupting a link in the area  $A_{(x)}$ .

Regarding this, the certainty of interruption of a link by an object between the emitter  $e$  and the receiver  $r$  is the following:

$$\sigma_{e,r} = \alpha_{e,r} \cdot \rho_{e,r} \tag{8}$$

where  $\alpha_{e,r}$  is the channel degradation before the interruption of the link; and  $\rho_{e,r}$  is the probability of the object to be in the area  $A_{(x)}$ . The value of  $\alpha_{e,r}$  is empirically computed according to Figure 11.

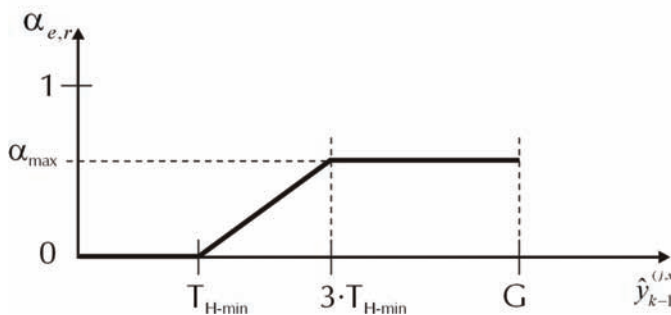


Fig. 11. Calculation of  $\alpha_{e,r}$ .

In Figure 11,  $\alpha_{max}=0.5$ ;  $T_{H-min}$  is the minimum threshold;  $G$  is the process gain -see (3)-; and  $\hat{y}_{k-1}^{(j,x)}$  is the estimate of the correlation carried out by the  $H_\infty$  filter at  $k-1$  instant (before the interruption of the link). This estimate can be considered as a channel degradation measurement, and it corresponds to the link between the emitter  $e$  and the receiver  $r$ . After obtaining  $\sigma_{e,r}$ , the value for the certainty of the existence of obstacles in the area  $A_{(x)}$  is computed considering the eleven links. If it is assumed that the probability of the link  $l_{e,r}$  being interrupted by an object is  $\sigma_{e,r}$ , then  $c_{A(x)}$  is obtained as the union probability of independent events, as (9) shows. In (9), for convenience, the probability of every link is represented by  $P(l_i)$  ( $P(l_i) = \sigma_{e,r}$ ), and  $i$  represents every link according to the assignment done in Table 2.

$$\begin{aligned}
 P(l) &= 0; \\
 \text{for } i &= -5; i \leq 5; i++ \\
 P(l_i \cup l) &= P(l_i) + P(l) - P(l_i) \cdot P(l) \\
 P(l) &= P(l_i \cup l) \\
 \text{endfor} \\
 c_{A(x)} &= P(l)
 \end{aligned} \tag{9}$$

It is important to remark that if the number of interrupted links in a barrier provides a value for the corresponding  $c_{A(x)} > 0.5$ , then it is very possible than an object exists in the area  $A_{(x)}$ , but it can not be concluded if it is larger than minimum dimensions.

## 5.2 Dempster-Shafer's theory application

Once the values for the certainty of existence of objects are available for every area, they can be combined between consecutive areas  $A_{(x)}$  and  $A_{(x+1)}$ , in order to obtain the certainty of existence of objects larger than 50x50x50cm. According to the Dempster-Shafer's theory (Klein, 2004), if  $c_{A(x)}$  is considered as the probability mass of the certainty of existence of objects in the area  $A_{(x)}$ , a value  $c_{o,x} \in [0,1]$  can be obtained as shows (10), representing the result of the fusion between the areas  $A_{(x)}$  and  $A_{(x+1)}$ .

$$c_{o,x} = \frac{c_{A(x)} \cdot c_{A(x+1)}}{1 - (1 - c_{A(x)}) \cdot c_{A(x+1)} - (1 - c_{A(x+1)}) \cdot c_{A(x)}} \tag{10}$$

The higher the value of  $c_{o,x}$  is, the higher the certainty of existence of object larger than 50x50x50cm is. But if  $c_{o,x}$  takes a value of 0.5, it means that there is a situation of uncertainty (it can not be concluded if there exists a dangerous object). Values of  $c_{o,x}$  lower than 0.5 mean there is a certainty of absence of object.

All the components  $c_{o,x}$  can be arranged in a vector  $\mathbf{C}_O$  as (11) shows, being  $N_Z$  the number of influence areas. According to the number of consecutive components of vector  $\mathbf{C}_O$  higher than 0.5, can be concluded how large the object is.

$$\mathbf{C}_O = [c_{o,1} \quad \cdots \quad c_{o,x} \quad \cdots \quad c_{o,(N_Z-1)}] \tag{11}$$

### 6. Real results

To evaluate the feasibility of the proposed data fusion algorithms, a prototype of the barrier has been used. Figure 12 shows an IR barrier prototype based on 4 emitters and 8 receivers. Figure 13 shows the established links among emitters and receivers and the detection areas. Notice that any area have eleven links as was explained, due to the fact that it is necessary 5 emitters, and in this situation only 4 have been used. This prototype carries out the encoding and detection schemes that have been previously described.



Fig. 12. IR barrier prototype: (a) Emitter barrier; (b) Receiver barrier.

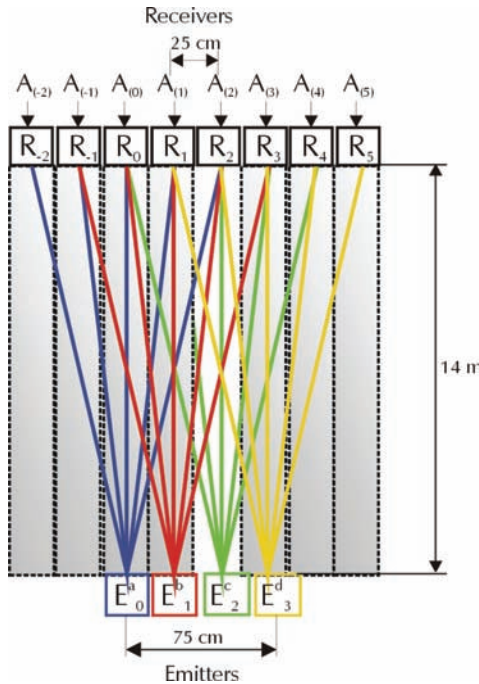


Fig. 13. Established links and detection areas using the prototype of the barrier.

Focusing on the data fusion algorithms, two situations have been tested: firstly, when an object larger than 50 cm of side is between the barriers; and secondly, when there exist random cuts of the links.

Figure 14 shows the scheme of a real example of detection, when one dangerous object (larger than 50x50x50cm) is inside the detection area. Table 3 shows all the relevant information that it is necessary for applying the algorithm belonging to  $A_{(1)}$  and  $A_{(2)}$  (the rest of areas have been omitted for simplicity): first column shows the established links in each area; second column, the estimate of the correlation values carried out by the  $H_\infty$  filter in  $k-1$  instant for each link,  $\hat{y}_{k-1}^{(j,x)}$ , when any link was interrupted; next columns show the channel degradation ( $\alpha_{e,r}$ ) before the interruption of the link and the probability  $\rho_{e,r}$  of the object to be in the area  $A_{(x)}$ ; and finally, last column shows the certainty of interruption of a link by an object,  $\sigma_{e,r}$ . In this test,  $\alpha_{e,r}$  has been obtained as was shown in Figure 11. The minimum threshold  $T_{H-min}$  is fixed to 50,  $\alpha_{max}=0.5$  and the process gain  $G$  is 1024.

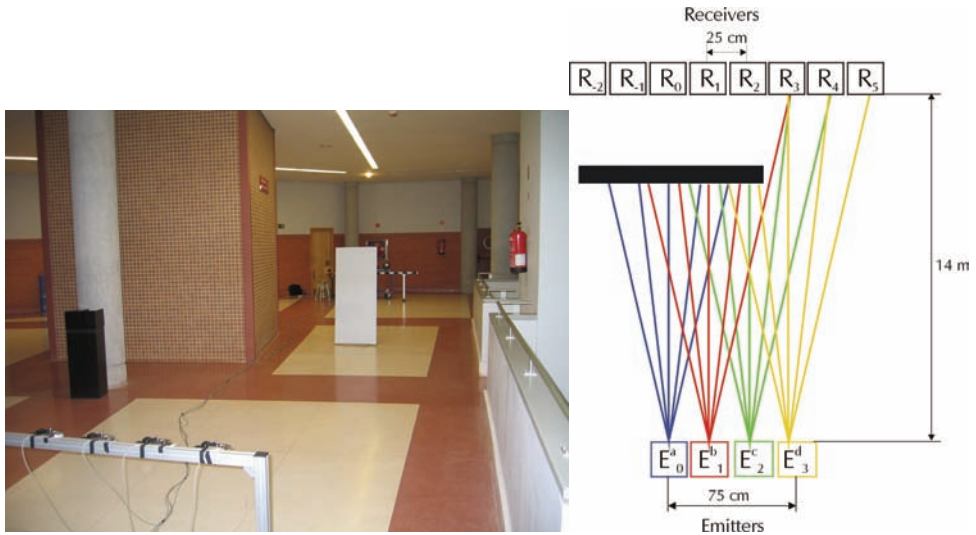


Fig. 14. Interrupted links when an object larger than 50 cm of side exists.

Then, the certainties of existence of objects in areas ( $c_{A(x)}$ ) have been computed using the algorithm described (8)-(9). Finally, consecutive areas have been combined by using (10) to obtain the vector  $C_o$ , containing the certainty of existence of objects larger than 50x50x50cm. Table 4 shows the interrupted links ( $e,r$ ), and the obtained results for  $c_{A(x)}$  and the vector  $C_o$ . As the results show there is a high certainty (higher than 0.7) of the existence of an object larger than 50x50x50cm, between areas  $A_{(-1)}$  y  $A_{(0)}$ ,  $A_{(0)}$  and  $A_{(1)}$ , and  $A_{(1)}$  y  $A_{(2)}$ . Due to the fact that these areas are consecutive ones, it could be concluded that the size of the object ( $x$  dimension) is between 75 and 100 cm.

Figure 15 displays a different situation. In this case, there are random cuts of the links, due to the existence of small objects. In a real situation it can be due to small animals, flying leaves, a sensor failure, etc. The applied method is the same that has been described in the

previous example. Table 5 shows the data fusion results, showing clearly a low certainty of the existence of dangerous objects (larger than 50x50x50 cm).

<b>Area <math>A_{(1)}</math></b>				
Link ( $e,r$ )	$\hat{y}_{k-1}$	$\alpha_{e,r}$	$\rho_{e,r}$	$\sigma_{e,r}$
1,-1	60	0.05	0.25	0.012
1,0	207.2	0.5	0.5	0.25
2,0	102.8	0.26	0.5	0.13
0,1	145.1	0.47	0.5	0.23
1,1	416.8	0.5	1	0.5
2,1	99.3	0.24	0.5	0.12
3,1	55	0.02	0.25	0.005
0,2	66.7	0.08	0.5	0.04
1,2	252.5	0.5	0.5	0.25
1,3	73	0.11	0.25	0.027
<b>Area <math>A_{(2)}</math></b>				
Link ( $e,r$ )	$\hat{y}_{k-1}$	$\alpha_{e,r}$	$\rho_{e,r}$	$\sigma_{e,r}$
2,0	102.8	0.26	0.25	0.065
2,1	99.3	0.24	0.5	0.12
3,1	55	0.02	0.5	0.01
0,2	66.7	0.08	0.25	0.02
1,2	252.5	0.5	0.5	0.25
2,2	321.3	0.5	1	0.5
3,2	130.3	0.4	0.5	0.2
1,3	73	0.11	0.5	0.055
2,3	345.1	0.5	0.5	0.25
2,4	203.2	0.5	0.25	0.125

Table 3. Real data belonging to detection areas  $A_{(1)}$  and  $A_{(2)}$  for the situation shown in Figure 14.

Interrupted links ( $e,r$ ): (0,-2);(0,-1);(1,-1);(0,0);(1,0);(2,0);(0,1);(1,1);(2,1);(3,1); (0,2); (1,2); (2,2); (3,2)							
$A_{(-2)}$	$A_{(-1)}$	$A_{(0)}$	$A_{(1)}$	$A_{(2)}$	$A_{(3)}$	$A_{(4)}$	$A_{(5)}$
$C_{A_{(-2)}}=0.125$	$C_{A_{(-1)}}=0.44$	$C_{A_{(0)}}=0.8$	$C_{A_{(1)}}=0.84$	$C_{A_{(2)}}=0.76$	$C_{A_{(3)}}=0.2$	$C_{A_{(4)}}=0$	$C_{A_{(5)}}=0$
$c_{0,-2}=0.1$							
	$c_{0,-1}=0.75$						
		$c_{0,0}=0.95$					
			$c_{0,1}=0.94$				
				$c_{0,2}=0.44$			
					$c_{0,3}=0$		
						$c_{0,4}=0$	
$C_o = [0.1 \quad 0.75 \quad 0.95 \quad 0.94 \quad 0.44 \quad 0 \quad 0]$							

Table 4. Data fusion results for the test shown in Figure 14.

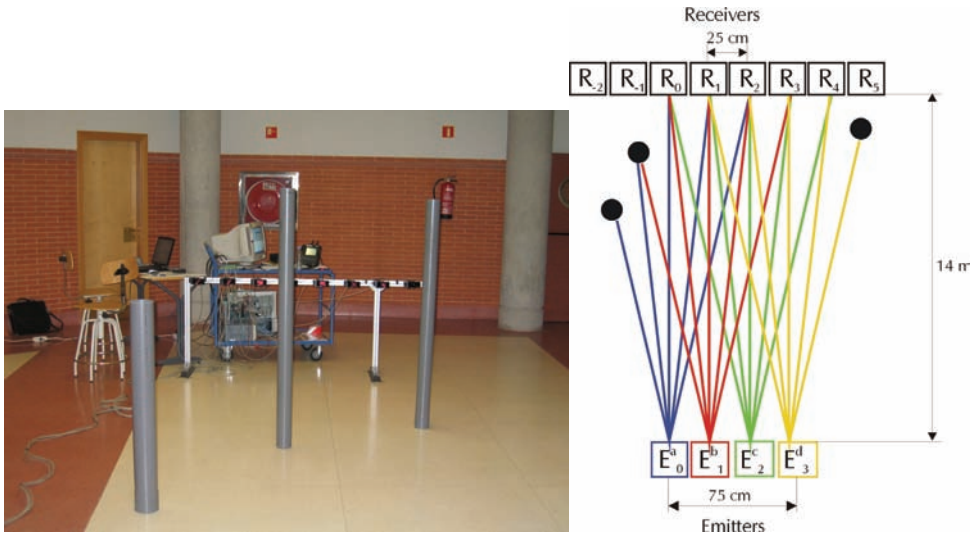


Fig. 15. Interrupted links when there are random cuts of the links.



Interrupted links $(e,r): (0,-2);(0,-1);(1,-1);(3,5)$							
$A_{(-2)}$	$A_{(-1)}$	$A_{(0)}$	$A_{(1)}$	$A_{(2)}$	$A_{(3)}$	$A_{(4)}$	$A_{(5)}$
$C_{A(-2)}=0.125$	$C_{A(-1)}=0.5$	$C_{A(0)}=0.25$	$C_{A(1)}=0$	$C_{A(2)}=0$	$C_{A(3)}=0.125$	$C_{A(4)}=0.25$	$C_{A(5)}=0.125$
$c_{0,-2}=0.125$							
	$c_{0,-1}=0.25$						
		$c_{0,0}=0$					
			$c_{0,1}=0$				
				$c_{0,2}=0$			
					$c_{0,3}=0.04$		
						$c_{0,4}=0.04$	
$C_o = [0.125 \quad 0.25 \quad 0 \quad 0 \quad 0 \quad 0.04 \quad 0.04]$							

Table 5. Data fusion results for the test shown in Figure 15.

## 7. Conclusion

In this chapter, a real prototype of an infrared barrier for obstacle detection on the tracks has been presented. Due to the fact that detection is based on the lack of radiation in the receivers, the channel degradation can be mistaken with the existence of obstacles. For this reason, validation algorithms are necessary, in order to increase the reliability of the detection.

Sensor data fusion based on the evidential theory has been applied in order to obtain the certainty of the existence of dangerous obstacles for the railway traffic. The proposed fusion algorithm takes into account the spatial diversity of the links that are established in the barrier and how the channel degradation affects them. Real tests have been carried out, in order to validate the described algorithms, showing them successful results.

To improve the safety level required in this application, it is necessary to incorporate new sensory systems, as can be cameras or ultrasounds to make up for the infrared deficiencies. All the information provided by these new sensory systems, have to be fused with the IR barrier, in order to obtain reliable information about the existence of objects on the tracks that can suppose a risk for railway traffic.

## 8. Acknowledgments

The work described in this paper has been possible by funding from Spanish Ministry of Education and Science (RESELAI project, ref. TIN2006-14896-C02-01); the Spanish Ministry of Public Works (VIATOR project, ref. 70025/T05), and the University of Alcalá-Community of Madrid (MEFASRET project, ref. CCG06-UAH/DPI-0748).

## 9. References

- Bloom, Scott; Korevaar, Eric; Schuster, John; Willebrand, Heinz. "Understanding the performance of free-space optics [Invited]". *Journal of Optical Networking*, June 2003, Vol 2 n° 6, pp 178-200
- Chow, A. "Performance of spreading Codes for direct sequence code division multiple access (DS-CDMA)", Technical Report, 5 December 2003, Stanford University.
- De Marziani, Carlos; Ureña, Jesús; Hernández, Álvaro; Mazo, Manuel; Álvarez, Fernando J. ; García, Juan Jesús; Donato, Patricio. "Modular Architecture for Efficient Generation and Correlation of Complementary Set of Sequences", *IEEE Transactions on Signal Processing*, Vol. 55, No. 5, May 2007, pp. 2323-2337
- Díaz, M. J. ; García, J. Jesús; Hernández, Álvaro; Losada, Cristina; García, Enrique. "Advanced Multisensorial Barrier for Obstacle Detection". *Proceedings of IEEE International Symposium on Intelligent Signal Processing. WISP 2007*, Alcalá de Henares, Madrid, Spain, pp. 549-554
- García, J. Jesús ; Losada, Cristina ; Espinosa, Felipe ; Ureña, Jesús ; Hernández, Álvaro Hernández ; Mazo, Manuel ; De Marziani, Carlos ; Jiménez, Ana ; Álvarez Fernando ; Jiménez, José A. "Optimal estimation techniques to reduce false alarms in railway obstacle detection". *Proceedings of IEEE International Conference on Industrial Technology. ICIT 2005*, Hong-Kong, China, pp 459-464
- García, J. J., Ureña, J., Hernández, Á., Mazo, M., García, J.C., Álvarez, F., Jiménez, J. A., Donato, P., Pérez, C. "IR sensor array configuration and signal processing for detecting obstacles in railways" *Proceedings Third IEEE Sensor Array and Multichannel Signal Processing Workshop SAM 2004*.
- GIF, 2001. Gestor de Infraestructuras Ferroviarias. "System for falling obstacle detection on railway. Technical and functional requirements". (Published in Spanish). 2001.
- GIF, 2004. Gestor de Infraestructuras Ferroviarias. "Functional description of the object fall detector based on infrared sensors. (Published in Spanish). 2004
- Kim, Isaac I. ; Stieger, Ron ; Koontz, Joseph A.; Moursund, Carter; Barclay, Micah ; Adhikari, Prasanna ; Schuster, John ; Korevaar, Eric; Ruigrok, Richard; DeCusatis, Casimer. "Wireless optical transmission of fast ethernet, FDDI, ATM, and ESCON protocol data using the TerraLink laser communication system", *Optical Engineering*, Vol. 37 No. 12, December 1998
- Kim, Isaac I.; McArthur, Bruce; Korevaar, Eric. "Comparison of laser beam propagation at 785 nm and 1550 nm in fog and haze for optical wireless communications", whitepapers, Optical Access, January 2000.
- Klein, Lawrence A., "Data and sensor fusion: a tool for information assessment and decision making", *Spie Press*, 2004.
- Simon, D. "From here to infinity ". *Embedded Systems Programming Magazine*. July, 2000. pp. 2-9
- Tseng, Shu-Ming & Bell, M.R.; "Asynchronous multicarrier DS-CDMA using mutually orthogonal complementary sets of sequences", *IEEE Transactions on Communications*, Volume 48, Issue 1, Jan. 2000, pp 53 - 59.

Yokota, Tomoharu; Asaka, Kazuhiko; Matsumoto, Hideki; Yamaguchi, Tsuyoshi; Kasai, Chiharu. "A Study on Improving the Link Quality of the Free Space Optical Transmission System", Online. [http://www.icsa.gr.jp/english/e\\_article\\_0103.htm](http://www.icsa.gr.jp/english/e_article_0103.htm), 2002.

# A Model of Federated Evidence Fusion for Real-Time Traffic State Estimation

Qing-Jie Kong and Yuncai Liu

*Department of Automation, Shanghai Jiao Tong University  
Shanghai  
China*

## 1. Introduction

Due to the rapid development of Intelligent Transportation Systems (ITS), more and more different types of sensors are employed to detect traffic state information, so as to serve traffic agencies and travelers. However, each type of traffic detectors has its inherent drawbacks. For instance, loop detectors, as a kind of economical and efficacious detectors, have been widely used in most advanced cities. However, lots of errors are often induced by their high failure ratio and inaccurate traffic state conversion arithmetic. Similarly, probe vehicles are another type of popular detectors, which also has some problems, such as poor statistical representation and errors in the map matching. Therefore, how to make full use of the data from these detectors to obtain more accurate and comprehensive traffic state information becomes a new urgent problem need to be solved.

Recent years, information fusion as a new technology has been introduced to solve this problem, expecting to get better results by integrating information from multiple types of detectors. In this field, some researchers advanced their fusion methods on how to combine the data from loop detectors and GPS probe vehicles, and achieved good effectiveness to some extent. For example, R.-L. Cheu et al. developed a neural network based model to perform the fusion (Cheu et al., 2001); K. Choi and Y. Chung presented a fusion algorithm based on fuzzy regression (Choi & Chung, 2002); T. Park and S. Lee researched this problem using Bayesian approach, who got good effect in simulation data (Park & Lee, 2004); H.-S. Zhang et al. proposed an architecture to manage, analyze and unify the traffic data (Zhang et al., 2005).

More recently, pointing to the incompleteness and inaccurateness of traffic detector data, El Faouzi and Lefevre originally put forward a classifiers fusion method based on Evidence Theory (El Faouzi & Lefevre, 2006), which provided a new idea toward solving this problem. Also, a prospect to build an adaptive and dynamic fusion scheme was given at the end of their article. According to this prospect, we introduce a new fusion model in this article to meet the requirement of real-time fusion. This model advances over D-S Evidence Theory (Dempster, 1967; Shafer, 1976) in temporal domain, and the idea comes from some thought of the Federated Kalman Filter initially proposed by N. A. Carlson (Carlson, 1988; 1990). Therefore, we call it the Federated Evidence Fusion Model (FEFM). It can be used to fuse not only the two kinds of detectors referred above, but also other information sources

(cameras, mobile phones etc.). In addition, sensor reliability is considered in this model by the form of evidence reliability to increase the accuracy of estimation. In the experiments, a simulation test is first assumed to explain the advantage of the proposed model, in comparison with conventional D-S Evidence Theory and the other two transformed models. After that, an application case is presented to embody the validity of the model in engineering practice, using the real-world data from the SCATS loop detectors and GPS-equipped taxis in Shanghai.

## 2. A brief review of Evidence Theory

Evidence Theory was initially introduced by Dempster (Dempster, 1967), and then Shafer (Shafer, 1976) showed the benefits of belief functions for modeling uncertain knowledge. In this section, some mathematical elements of Evidence Theory are recalled.

### 2.1 Basic concepts

Let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_N\}$  be a frame of discernment, in which all elements are assumed to be mutually exclusive and exhaustive. The power set of  $\Omega$  is denoted by  $2^\Omega = \{A \mid A \subseteq \Omega\}$ . Basic Probability Assignment (BPA) is a function that can be mathematically defined by  $2^\Omega$  in  $[0, 1]$ , such that  $m(\Phi) = 0$  where  $\Phi$  denotes an empty set, and  $\sum_{A \subseteq \Omega} m(A) = 1$ .

The belief function (bel) and the plausibility function (pl) are defined as follows:

$$\begin{cases} bel(A) = \sum_{\Phi \neq B \subseteq A} m(B) & \forall A \subseteq \Omega \\ pl(A) = \sum_{B \cap A \neq \Phi} m(B) & \forall A \subseteq \Omega \end{cases} \quad (1)$$

in which  $bel(A)$  represents the sum of masses in all subsets of  $A$ , whereas  $pl(A)$  corresponds to the sum of masses committed to those subsets which don't discredit  $A$ .

### 2.2 Combination of belief functions

Multiple evidences can be fused by using Dempster's combination rules, shown in equation (2), which also is known as the orthogonal sum. This sum is both commutative and associative.

$$m(C) = \begin{cases} 0, & A \cap B = \Phi \\ \frac{1}{1-K} \sum_{A \cap B = C, \forall A, B \subseteq \Omega} m_i(A) \cdot m_j(B), & A \cap B \neq \Phi \end{cases} \quad (2)$$

with

$$K = \sum_{A \cap B = \Phi, \forall A, B \subseteq \Omega} m_i(A) \cdot m_j(B) \quad (3)$$

where the term  $K$  is called the conflict factor between two evidences, which reflects the conflict degree between them.

### 2.3 Evidence reliability

When the information provided by sensors is not totally reliable to result in the belief functions, a coefficient  $a$  is used to discount the belief. This coefficient will transfer the belief into the set  $\Omega$ . Thus, the discounted belief function  $m_a$  can be obtained by the following formula:

$$\begin{cases} m_a(A) = am(A) \\ m_a(\Omega) = am(\Omega) + 1 - a \end{cases} \quad A \subset \Omega \quad (4)$$

where  $a \in [0, 1]$ .

### 2.4 Evidence distance

A. L. Jousselme et al. presented a principled metric distance between two BPAs (Jousselme et al., 2001). The authors treat BPA as a vector in a  $2^{|\Omega|}$  linear space, where  $|\Omega|$  denotes the cardinality of  $\Omega$ . Then, they define the distance between  $m_i$  and  $m_j$  as

$$d_{BPA}(m_i, m_j) = \sqrt{\frac{1}{2}(\bar{m}_i - \bar{m}_j)D(\bar{m}_i - \bar{m}_j)^T} \quad (5)$$

in which  $D$  is a matrix with size of  $2^{|\Omega|} \times 2^{|\Omega|}$ , whose elements can be calculated by formula (6).

$$D(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad \forall A, B \subseteq \Omega \quad (6)$$

Furthermore, the evidence distance satisfies the below restriction:

$$0 \leq d_{BPA}(m_i, m_j) \leq 1 \quad (7)$$

## 3. Federated evidence fusion model

In this section, we introduce the FEFM in three steps: first, the reliability matrix is discussed; then, we build the frame of the FEFM; finally, the fusion algorithm is presented. Besides, the other two models with different forms are also given like the Federated Kalman Filter.

### 3.1 The improved evidence reliability

From engineering practice, we find that different evidence sources have different reliabilities in estimating the same state; similarly, the same evidence sources also have different reliabilities in measuring different states. Therefore, we define a reliability weight  $w_{i,j}$  ( $0 \leq w_{i,j} \leq 1$ ), which is used to reflect the degree of the reliability that one evidence corresponds to each state. The reliability matrix  $W$  is shown in (8).

$$W = \begin{matrix} & S_1 & S_2 & \cdots & S_N \\ \begin{matrix} E_1 \\ E_2 \\ \vdots \\ E_M \end{matrix} & \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,N} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{M,1} & w_{M,2} & \cdots & w_{M,N} \end{bmatrix} \end{matrix} \quad (8)$$

where  $w_{i,j} \in [0,1], (i=1,2,\dots,M; j=1,2,\dots,N)$ ;  $S_j$  denotes the  $j$ th independent state to be recognized;  $E_i$  denotes the  $i$ th independent evidence.

In some sense, the reliability that the evidence corresponds to different states can be deemed as the probability that the evidence exactly estimates the state, because it accords with the definition of probability. Thus, we can draw a conclusion that, a reliability matrix  $\mathbf{W}^*$  between evidences and all of the subsets of  $\Omega$  is defined as follows:

$$\mathbf{W}^* = \begin{matrix} & \{S_1\} & \{S_2\} & \cdots & \{S_N\} & \{S_1, S_2\} & \{S_1, S_3\} & \cdots & \{S_2, S_3\} & \cdots & \{S_j | j \neq N\} & \Omega \\ \begin{matrix} E_1 \\ E_2 \\ \vdots \\ E_M \end{matrix} & \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,N} & \overset{P}{w}_{1,2} & \overset{P}{w}_{1,3} & \cdots & \overset{P}{w}_{1,2} & \cdots & \overset{P}{w}_{1,j} & \overset{P}{w}_{1,j} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,N} & \overset{P}{w}_{2,2} & \overset{P}{w}_{2,3} & \cdots & \overset{P}{w}_{2,2} & \cdots & \overset{P}{w}_{2,j} & \overset{P}{w}_{2,j} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ w_{M,1} & w_{M,2} & \cdots & w_{M,N} & \overset{P}{w}_{M,j} & \overset{P}{w}_{M,j} & \cdots & \overset{P}{w}_{M,j} & \cdots & \overset{P}{w}_{M,j} & \overset{P}{w}_{M,j} \end{bmatrix} \end{matrix} \quad (9)$$

When  $i=1$ , we have

$$\begin{cases} \overset{P}{w}_{1,2} = w_{1,1} \hat{+} w_{1,2} = w_{1,1} + w_{1,2} - w_{1,1} \cdot w_{1,2} \\ \overset{P}{w}_{1,2,3} = w_{1,1} \hat{+} w_{1,2} \hat{+} w_{1,3} = w_{1,1} + w_{1,2} - w_{1,1} \cdot w_{1,2} - w_{1,1} \cdot w_{1,3} - w_{1,2} \cdot w_{1,3} + w_{1,1} \cdot w_{1,2} \cdot w_{1,3} \\ \vdots \\ \overset{P}{w}_{1,2,\dots,N} = w_{1,1} \hat{+} w_{1,2} \hat{+} \cdots \hat{+} w_{1,N} \end{cases} \quad (10)$$

where  $\hat{+}$  denotes the addition operation in probability theory.

Likewise, when  $i=2,\dots,M$ , we can also obtain the above conclusions respectively. This issue comes from the addition formula in probability theory, as the  $N$  states are irrelevant with each other.

Representing every weight with  $v_{i,j'}$ , a reliability index matrix  $\mathbf{V}$  can be shown as:

$$\mathbf{V} = \begin{matrix} & \{S_1\} & \{S_2\} & \cdots & \{S_N\} & \{S_1, S_2\} & \{S_1, S_3\} & \cdots & \{S_2, S_3\} & \cdots & \{S_j | j \neq N\} & \Omega \\ \begin{matrix} E_1 \\ E_2 \\ \vdots \\ E_M \end{matrix} & \begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,N} & v_{1,N+1} & v_{1,N+2} & \cdots & v_{1,2N} & \cdots & v_{1,2^N-2} & v_{1,2^N-1} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,N} & v_{2,N+1} & v_{2,N+2} & \cdots & v_{2,2N} & \cdots & v_{2,2^N-2} & v_{2,2^N-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ v_{M,1} & v_{M,2} & \cdots & v_{M,N} & v_{M,N+1} & v_{M,N+2} & \cdots & v_{M,2N} & \cdots & v_{M,2^N-2} & v_{M,2^N-1} \end{bmatrix} \end{matrix} \quad (11)$$

where  $v_{i,j'} \in [0,1], (i=1,2,\dots,M; j'=1,2,\dots,2^N-1)$ .

In this case, the modified BPA can be shown in equation (12):

$$\begin{cases} m_i^l(\mathbf{A}) = v_{i,j'} \cdot m_i(\mathbf{A}), \\ m_i^l(\Omega) = 1 - \sum_{\mathbf{A} \subset \Omega} v_{i,j'} \cdot m_i(\mathbf{A}), \end{cases} \quad \mathbf{A} \subset \Omega \quad (12)$$

where  $m_i^l(\ )$  indicates that the BPA has been modified by the reliability index of evidences.

The proposed reliability index matrix  $V$  can also be generalized to denote other weight meanings, such as measure accuracy or evidence importance. Among them evidence importance was referred in (Fan & Zuo, 2006). For example,  $v_{1,1} > v_{2,1}$  indicates that evidence 1 is more reliable than evidence 2 to judge state 1. We can also think that evidence 1 is more important than evidence 2. All of the other generalizations can be explained as the same.

By considering evidence reliability, the uncertainty and inaccuracy of evidences are greatly decreased. Meanwhile, the conflict between two evidences may be weakened to some degree.

### 3.2 Federated evidence fusion frame

The combination rule proposed by Dempster provided a convenient tool for us to fuse multi-source information. In our case, we are going to fuse the data obtained from multiple types of traffic detectors in real-time. Therefore, we build the FEFM in a structure with feedback, which was inspired by the theory of Federated Kalman Filter first proposed by Carlson (Carlson, 1988; 1990). The proposed fusion frame with feedback is illustrated in figure 1. As the figure shows, the whole fusion system consists of four parts, which are the input level, the feature extracting level, the fusion level and the output level. Among them, the fusion level can be further divided into two components: main fusion system and sub-fusion system.

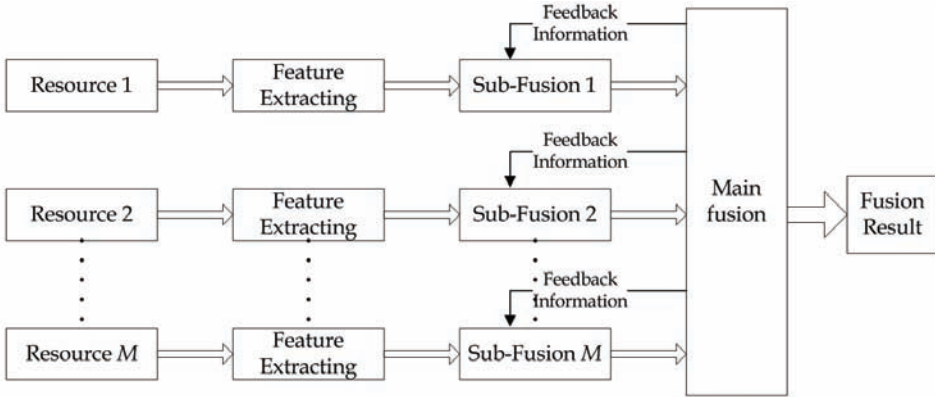


Fig. 1. Frame of the FEFM.

### 3.3 Federated fusion algorithm

In this algorithm, we use  $m_i(A_{i,t}), i = 1, 2, \dots, M$  to represent the BPA extracted from the data from the  $i$ th type of detectors at time  $t$ .

For the sub-fusion systems, the fusing rule makes use of the combining formula (2). It is shown as follows:

$$\begin{aligned}
 m_i(B_{i,t}) &= m(C_{t-1}) \oplus m_i(A_{i,t}) \\
 &= \frac{\sum_{C_{t-1} \cap A_{i,t} = B_{i,t}} g(m(C_{t-1})) \times m_i(A_{i,t})}{1 - \sum_{C_{t-1} \cap A_{i,t} = \Phi} g(m(C_{t-1})) \times m_i(A_{i,t})} \quad (13)
 \end{aligned}$$



where  $m_i(B_{i,t}), i=1,2,\dots,M$  denotes the fusion result of subsystem  $i$  at time  $t$ , and  $m(C_{t-1})$  represents the fusion result of the main system at time  $t-1$ . They both are in the form of BPA. The function  $g(\cdot)$  is the operation shown as follows:

$$\begin{cases} g(m(C_{t-1})) = \lambda m(C_{t-1}) \\ g(m(\Omega)) = 1 - \sum_{C_{t-1} \subset \Omega} \lambda m(C_{t-1}) \end{cases} \quad (14)$$

where  $\lambda$  is a variable, which represents the degree that  $m(C_{t-1})$  is weakened, and its value satisfies the restriction  $0 \leq \lambda \leq 1$ . The value of this parameter can be determined under the condition that the fusion result is identical with the real state at all time in the training set. By this means, we can weaken the feedback, i.e. avoid the feedback leading the fusion result at this time.

For the main fusion system, the fusion rule also use the combining algorithm of D-S Evidence Theory, which is

$$\begin{aligned} m(C_t) &= m_1(B_{1,t}) \oplus m_2(B_{2,t}) \oplus \dots \oplus m_M(B_{M,t}) \\ &= \frac{\sum_{\bigcap_{i=1}^M B_{i,t} = C_t} \left( \prod_{i=1}^M m_i(B_{i,t}) \right)}{1 - \sum_{\bigcap_{i=1}^M B_{i,t} = \Phi} \left( \prod_{i=1}^M m_i(B_{i,t}) \right)} \end{aligned} \quad (15)$$

where  $m(C_t)$  denotes the integrated result of the main fusion system at time  $t$ , which also is the final fusion result at time  $t$ .

Then, we can obtain the conclusion of state estimation at that time through a certain decision rule: the maximum belief or the maximum plausibility etc.

### 3.4 The other two structures

Similar to the Federated Kalman Filter, the FEFM also has the other two transformed structures. They are named the Distributed Feedback Fusion (DFF) and the no feedback fusion (NFF).

#### 3.4.1 Structure of the distributed feedback fusion

The distributed feedback fusion structure is shown in figure 2. In this structure, feedback information to every subsystem does not come from the main fusion system any more, but be produced by themselves. After every fusion cycle, the fusion results obtained by the sub-fusion systems are all sent back to their inputs to be integrated with the inputted state features at the next time. The detailed algorithm is given as follows:

For the sub-fusion systems,

$$\begin{aligned} m_i(B_{i,t}) &= m_i(B_{i,t-1}) \oplus m_i(A_{i,t}) \\ &= \frac{\sum_{B_{i,t-1} \cap A_{i,t} = B_{i,t}} g(m_i(B_{i,t-1})) \times m_i(A_{i,t})}{1 - \sum_{B_{i,t-1} \cap A_{i,t} = \Phi} g(m_i(B_{i,t-1})) \times m_i(A_{i,t})} \end{aligned} \quad (16)$$

in which,  $m_i(B_{i,t-1}), i=1,2,\dots,M$  denotes the feedback information from the output of subsystem  $i$ , which is the fusion result of this subsystem at time  $t-1$ .

As to the main fusion system, the formula is the same as the standard FEFM algorithm.

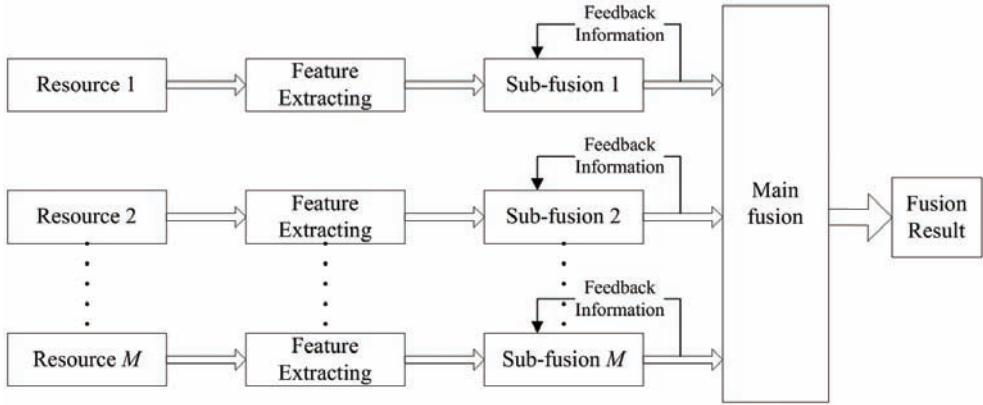


Fig. 2. Frame of the distributed feedback fusion

### 3.4.2 Structure of the no feedback fusion

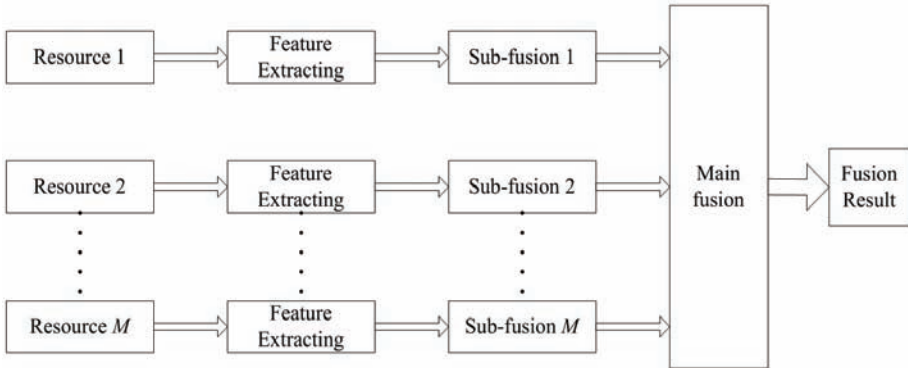


Fig. 3. Frame of the no feedback fusion

As is shown above, this type of structure has neither any feedback information from the main fusion system, nor from the sub-fusion system. Therefore, in this model, the fusion outputs of the sub-fusion systems are directly sent into the main fusion systems. This fusion outputs are attained by combining the inputs of the sub-fusion systems at this time and those at the last time. The difference of this algorithm lies in:

For the sub-fusion systems,

$$\begin{aligned}
 m_i(B_{i,t}) &= m_i(A_{i,t-1}) \oplus m_i(A_{i,t}) \\
 &= \frac{\sum_{A_{i,t-1} \cap A_{i,t} = B_{i,t}} g(m_i(A_{i,t-1})) \times m_i(A_{i,t})}{1 - \sum_{A_{i,t-1} \cap A_{i,t} = \emptyset} g(m_i(A_{i,t-1})) \times m_i(A_{i,t})} \quad (17)
 \end{aligned}$$

in which,  $m_i(A_{i,t-1}), i = 1, 2, \dots, M$  denotes the BPA of state feature put in sub-fusion system  $i$  at time  $t-1$ .

Likewise, in the main fusion system, the fusion algorithm also is the same as the standard FEFM algorithm.

#### 4. Implementation of fusion algorithm

In the last section, we have introduced the core algorithms of the FEFM; however, it still needs more procedures, if we want to veritably apply the whole model to the practice. They will be presented in this section.

##### 4.1 Determination of reliability matrix

In Reference (Guo et al., 2006), reliability was divided into static reliability and dynamic reliability. Herein, we only consider the static one. Thus, reliability weight  $w_{i,j}^s$  can be calculated by (18).

$$\begin{cases} w_{i,j}^s = f(d_{BPA}(m_{i,j}^s, m_{i,j}^o)) \\ f = b - a \cdot d_{BPA} \end{cases} \quad (18)$$

in which  $d_{BPA}$  is the evidence distance that can be computed by (5);  $m_{i,j}^s$  denotes the BPA output of the sensor  $i$  about the state class  $j$  in the training set;  $m_{i,j}^o$  represents the BPA of what we have known about the class membership of the same data;  $a=1$  and  $b=1$ , due to the boundary condition:  $w_{i,j} \in [0,1]$ .

##### 4.2 Creation of masses

Above all, we use the negative exponential proposed by Denoeux (Denoeux, 1995) to create the masses, shown in (19).

$$m_i(\{\omega_n\}) = \exp(-r_i d_i^\beta) \quad (19)$$

where  $d_i$  is a type of distance between the data detected by the  $i$ th kind of detectors and the prototype of each state class. The prototype can be designated artificially or be obtained by clustering the historical data. The parameters  $\beta$  and  $r_i$  are decided depending on the real-world data in the training set.

Afterward, we define the conversions below to create the BPAs.

$$\begin{cases} kth \max(m_i^*(\{\omega_k\})) = \frac{kth \max(m_i(\{\omega_k\}))}{\sum_{l=1}^{kth} \max(m_i(\{\omega_l\}))} \\ kth \max(m_i^l(\{\omega_k\})) = v_{i,k}^s \cdot kth \max(m_i^*(\{\omega_k\})) \\ m_i^l(\{\Omega\}) = 1 - \sum_{k=1}^N kth \max(m_i^l(\{\omega_k\})) \end{cases} \quad (20)$$

in which the  $kth \max$  denotes the  $kth$  maximum value in all the masses derived from the results computed by (19);  $k$  is a natural number in the range from 1 to  $N$ ;  $N$  is the total

number of elements in the frame of discernment;  $v_{i,k}^s$  represents the static reliability index weight, which has been computed beforehand.

### 4.3 Decision-making rules

Herein, we choose the maximum belief rule as the decision-making principle. Also, some additional conditions are provided according to the context of traffic engineering. The decision-making rules are shown as follows:

$$Dst = \begin{cases} st_{\max m}, & \text{if } st_{\max m} - st_{2^{th} \max} = 1, \\ \Omega, & \text{otherwise,} \end{cases} \quad (21)$$

where  $Dst$  indicates the decision output of state value;  $st_{\max m}$  denotes the state value corresponding to the mass with the maximum value; if the output is  $\Omega$ , it means no verdict. In this case, we may give an output of state 0.

## 5. Experiments

### 5.1 Synthetic data

First, we use synthetic traffic data to demonstrate the effectiveness of the proposed model. Assume the discernment framework of traffic state is  $\Omega = \{S_1, S_2, S_3, S_4\}$ , and the evidence set is  $E = \{E_1, E_2\}$ . The BPAs based on both evidences are listed in Table 1.

Time	Evidence	{S1}	{S2}	{S3}	{S4}	$\Omega$
T <sub>1</sub>	E <sub>1</sub> : $m_1(A_{1, t1})$	0.7	0.1	0	0	0.2
	E <sub>2</sub> : $m_2(A_{2, t1})$	0.1	0.8	0	0	0.1
T <sub>2</sub>	E <sub>1</sub> : $m_1(A_{1, t2})$	0.25	0.55	0	0	0.2
	E <sub>2</sub> : $m_2(A_{2, t2})$	0	0.6	0.25	0	0.15
T <sub>3</sub>	E <sub>1</sub> : $m_1(A_{1, t3})$	0	0.1	0.8	0	0.1
	E <sub>2</sub> : $m_2(A_{2, t3})$	0	0	0.6	0.25	0.15
T <sub>4</sub>	E <sub>1</sub> : $m_1(A_{1, t4})$	0	0	0.2	0.7	0.1
	E <sub>2</sub> : $m_2(A_{2, t4})$	0.8	0.1	0	0	0.1
T <sub>5</sub>	E <sub>1</sub> : $m_1(A_{1, t5})$	0	0	0.1	0.8	0.1
	E <sub>2</sub> : $m_2(A_{2, t5})$	0	0	0.2	0.75	0.05

Table 1. BPAs for the case

Herein, we provide 5 pairs of evidences at 5 continuous times respectively, among which the two evidences at time T<sub>1</sub> have a partial conflict, and the pair of evidences is completely conflict at time T<sub>4</sub>. Whereas, the evidences at other three times have little conflicts and only embody the state transferring.

The BPAs after being fused are listed in Table 2, which shows the comparison of the fusion results of the five models. They are the conventional Evidence Theory, Evidence Theory

considering reliability, standard FEFM, DFF and NFF, which are orderly denoted by  $m(C_{t1})$ ,  $m'(C_{t1})$ ,  $ml(C_{t1})$ ,  $mD(C_{t1})$  and  $mN(C_{t1})$ .

Time	Evidence	{S1}	{S2}	{S3}	{S4}	$\Omega$
T <sub>1</sub>	$m(C_{t1})$	0.37	0.58	0	0	0.05
	$m'(C_{t1})$	0.48	0.33	0	0	0.19
	$ml(C_{t1})$	0.48	0.33	0	0	0.19
	$mD(C_{t1})$	0.48	0.33	0	0	0.19
	$mN(C_{t1})$	0.48	0.33	0	0	0.19
T <sub>2</sub>	$m(C_{t2})$	0.06	0.82	0	0.08	0.04
	$m'(C_{t2})$	0.12	0.62	0.09	0	0.17
	$ml(C_{t2})$	0.23	0.63	0.05	0	0.09
	$mD(C_{t2})$	0.28	0.64	0.03	0	0.05
	$mN(C_{t2})$	0.18	0.64	0.06	0	0.12
T <sub>3</sub>	$m(C_{t3})$	0	0.02	0.92	0.03	0.03
	$m'(C_{t3})$	0	0.03	0.71	0.13	0.13
	$ml(C_{t3})$	0.05	0.19	0.57	0.09	0.1
	$mD(C_{t3})$	0.06	0.14	0.6	0.1	0.1
	$mN(C_{t3})$	0.02	0.11	0.66	0.1	0.1
T <sub>4</sub>	$m(C_{t4})$	0.42	0.05	0.11	0.37	0.05
	$m'(C_{t4})$	0.31	0.04	0.08	0.23	0.34
	$ml(C_{t4})$	0.19	0.1	0.34	0.18	0.19
	$mD(C_{t4})$	0.25	0.08	0.25	0.2	0.22
	$mN(C_{t4})$	0.2	0.04	0.3	0.23	0.23
T <sub>5</sub>	$m(C_{t5})$	0	0	0.06	0.93	0.01
	$m'(C_{t5})$	0	0	0.12	0.81	0.07
	$ml(C_{t5})$	0.02	0.01	0.17	0.76	0.04
	$mD(C_{t5})$	0.02	0.01	0.13	0.79	0.05
	$mN(C_{t5})$	0.02	0	0.11	0.82	0.05

Table 2. Results of fusion.  $m(C_{t1})$ ,  $m'(C_{t1})$ ,  $ml(C_{t1})$ ,  $mD(C_{t1})$  and  $mN(C_{t1})$  respectively denotes the conventional Evidence Theory, Evidence Theory considering reliability, standard FEFM, DFF and NFF.

In figure 4(a), the bar plot of the fusion consequences at time T<sub>4</sub> is shown to display the predominance of the FEFM. In the third row, the integrated feedback fusion represents the standard FEFM. From the figure, we can find that the former two methods can not give a

clear answer due to the huge conflict. However, the standard FEFM gives a definite choice. We can also see that the other two forms of the FEFM do not perform well at this situation, although a little of progress is made. This phenomenon can also be seen in figure 4(b), which shows us a course of estimation during the whole 5 times. Moreover, we can find a different determination between the first model and the other four models at time  $T_1$ , which is because there is a partial conflict between the two evidences, whereas the reliability mitigates its effect to some extent.

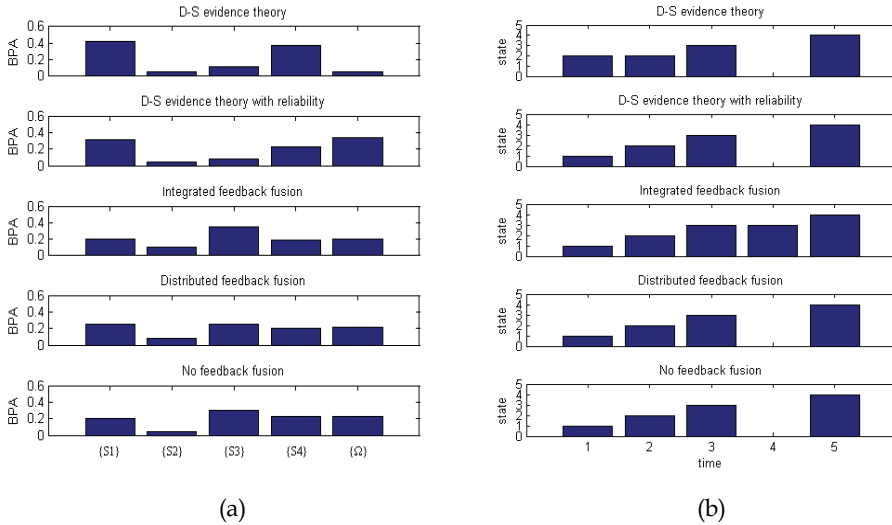


Fig. 4. Bar plot of the fusion results. (a) is the fusion results using the five models at time  $T_4$ ; (b) is the fusion results using these models during the 5 times.

**5.2 Data from real traffic**

In this section, we employ two types of traffic mean-speeds on an urban link in Shanghai to carry out the fusion estimation experiment. The two speeds were derived by estimating with the SCATS loop detector data and GPS-equipped taxi data. The detailed algorithms were provided in reference (Kong et al., 2007). The real-world data were collected at a section of Zhao Jia Bang Road of Shanghai through a whole day, which was from 0 o'clock to 24 o'clock on Sep. 26, 2006, and the average speeds were computed in every five minutes. Also, we screened a segment of surveillance video during 2:00 PM-4:00 PM at this link on the same day in order to validate the model.

The fusion consequence of the standard FEFM is shown in figure 5, from which we can clearly see that the traffic state is reasoned and tracked at the feature level. According to the verifying test by replaying the video, the estimation accuracy of our model is beyond 95%. Herein, we define five different traffic states, corresponding to 'very congested', 'congested', 'medium', 'smooth' or 'very smooth', respectively.

Moreover, the fusion algorithm was embedded into the Shanghai Urban Traffic Information System developed by our laboratory to implement the fusion estimation. The estimation

results were displayed on the traffic information display platform, as figure 6 shows. Figure 6(a) and 6(b) respectively illustrates the results estimated only by the SCATS data or the GPS data; and figure 6(c) shows the results by fusing the two types of data. In these figures, the color of the road sections, red, orange, yellow, green, or dark green, represents the corresponding traffic states defined above respectively.

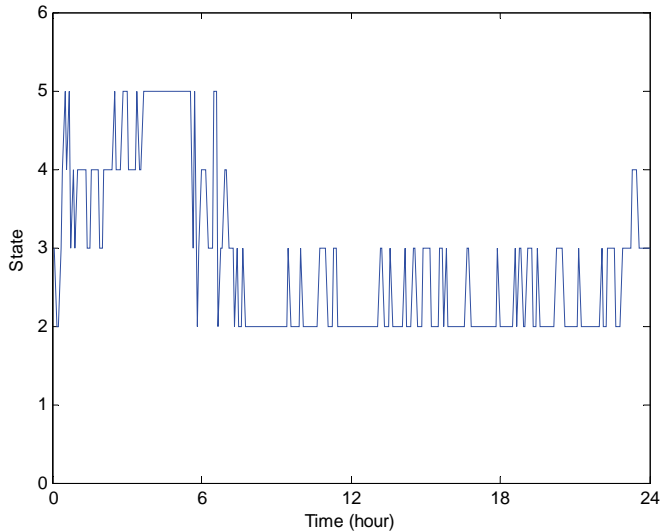
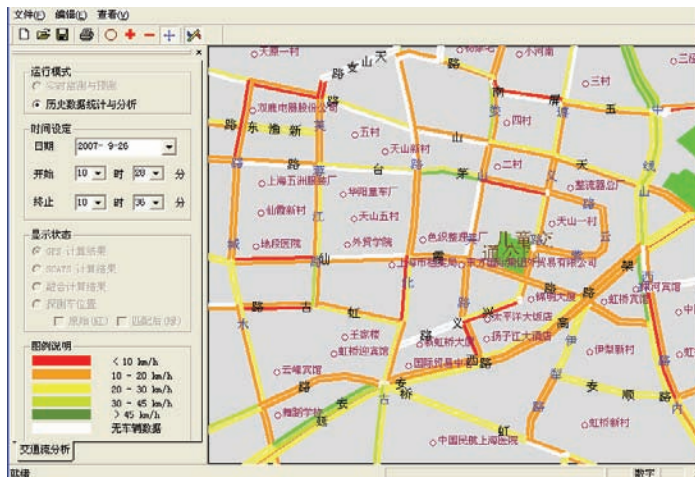


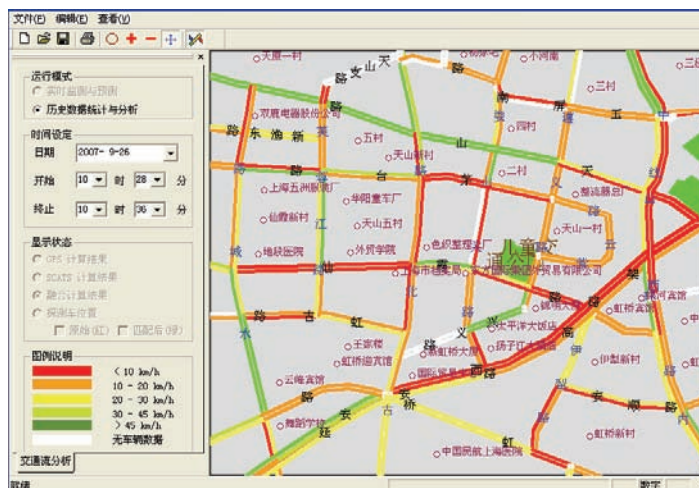
Fig. 5. Traffic state estimation results of the standard FEFM with the real-world data on the link in Shanghai through 24 hours.



(a)



(b)



(c)

Fig. 6. Estimation results displayed on the platform. (a) is the result from the SCATS loop detector data; (b) is the result from the GPS-equipped taxi data; (c) is the result by fusing the two types of data.

## 6. Conclusions

This paper has proposed a model for real-time traffic state estimation by developing D-S Evidence Theory in temporal domain. As it realizes online fusion of heterogeneous detector data at the feature level, the method has strong application potentials in fusing data from many other different types of sensors (cameras, cell phones, etc.). Furthermore, the evidence reliability to every state is considered in the FEFM. Finally, the model shows great



advantages over conventional D-S Evidence Theory in the simulation test and good accuracy by the tests with the real-world data.

## 7. References

- Carlson, N. A. (1988). Federated filter for fault-tolerant integrated navigation systems, *Proceedings of IEEE Position Location and Navigation Symposium*, pp. 110-119, Orlando, FL, USA, Nov. 1988, IEEE
- Carlson, N. A. (1990). Federated square root filter for decentralized parallel process. *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 26, (1990) pp. 517-525
- Cheu, R.-L.; Lee, D.-H. & Xie, C. (2001). An arterial speed estimation model fusing data from stationary and mobile sensors, *Proceedings of IEEE Intelligent Transportation Systems Conference*, pp. 573-578, Oakland, CA, USA, Aug. 2001, IEEE
- Choi, K. & Chung, Y. (2002). A data fusion algorithm for estimating link travel time. *Journal of Intelligent Transportation Systems*, Vol. 7, (2002) pp. 235-260
- Dempster, A. P. (1967). Upper and lower probabilities induced by multi-valued mapping. *The Annals of Mathematical Statistics*, Vol. 38, No. 4, (1967) pp. 325-339
- Denoeux, T. (1995). A k-nearest neighbor classification rule based on Dempster-Shafer Theory. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 25, No. 5, (1995) pp. 804-813
- El Faouzi, N.-E. & Lefevre, E. (2006). Classifiers and distance-based evidential fusion for road travel time estimation, *Proceedings of SPIE: Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications*, pp. 1-16, Orlando, FL, USA, Apr. 2006, SPIE
- Fan, X. & Zuo, M.-J. (2006). Fault diagnosis of machines based on D-S evidence theory. Part 1: D-S evidence theory and its improvement. *Pattern Recognition Letter*, Vol. 27, (2006) pp. 366-376
- Guo, H.; Shi, W. & Deng, Y. (2006). Evaluating sensor reliability in classification problems based on evidence theory. *IEEE Transactions on Systems, Man, and Cybernetics: Part B*, Vol. 36, No. 5, (2006) pp. 970-981
- Jousselme, A. L.; Grenier, D. & Bossé, É. (2001). A new distance between two bodies of evidence. *Information Fusion*, Vol. 2, No. 2, (2001) pp. 91-101
- Kong, Q.-J.; Chen, Y. & Gao, L. (2007). Technical handbook of key technology research and demonstrated application on traffic information fusion, Intelligent Transportation and Computer Vision Lab of SJTU, Shanghai, China
- Park, T. & Lee, S. (2004). A Bayesian approach for estimating link travel time on urban arterial road network, In: *Computational Science and Its Applications*, pp. 1017-1025, Springer, 978-3-540-22054-1, Heidelberg, Germany
- Shafer, G. (1976). *A Mathematical Theory of Evidence*, Princeton Univ. Press, Princeton, NJ, USA
- Zhang, H.-S.; Zhang, Y.; Yao, D.-Y. & Hu, D.-C. (2005). An architecture of traffic state analysis based on multi-sensor fusion, *Proceedings of IEEE 8th International Conference on Intelligent Transportation Systems*, pp. 855-860, Vienna, Austria, 2005, IEEE

# Multi Sensor Data Fusion Architectures for Air Traffic Control Applications

Baud Olivier, Gomord Pierre, Honoré Nicolas,  
Ostorero Loïc, Taupin Olivier and Tubery Philippe  
*THALES Air Systems*  
*France*

## 1. Introduction

Nowadays, the radar is no longer the sole technology which is able to ensure the surveillance of air traffic. The extensive deployment of satellite systems and air-to-ground data links leads to the emergence of complementary means and techniques on which a great deal of research and experiments have been carried out over the past ten years.

In such an environment, the sensor data processing, which is a key element in any Air Traffic Control (ATC) centre, has been continuously upgraded so as to follow the sensor technology evolution and in the meantime improves the quality in term of continuity, integrity and accuracy criteria.

This book chapter proposes a comprehensive description of the state of art and the roadmap for the future of the multi sensor data fusion architectures and techniques in use in ATC centres.

The first part of the chapter describes the background of ATC centres, while the second part of the chapter points out various data fusion techniques. Multi radar data processing architecture is analysed and a brief definition of internal core tracking algorithms is given as well as a comparative benchmark based on their respective advantages and drawbacks.

The third part of the chapter focuses on the most recent evolution that leads from a Multi Radar Tracking System to a Multi Sensor Tracking System.

The last part of the chapter deals with the sensor data processing that will be put in operation in the next ten years. The main challenge will be to provide the same level of services in both surface and air surveillance areas in order to offer:

- highly accurate air and surface situation awareness to air traffic controllers,
- situational awareness via Traffic Information System - Broadcast (TIS-B) services to pilots and vehicle drivers, and
- new air and surface safety, capacity and efficiency applications to airports and airlines.

## 2. Air traffic control

Air Traffic Control (ATC) is a service provided to regulate the airline traffic. Main functions of the ATC system are used by controllers to (i) avoid collisions between aircrafts, (ii) avoid collisions on manoeuvring areas between aircrafts and obstructions on the ground and (iii) expediting and maintaining the orderly flow of air traffic.

An ATC system shall adapt itself to the control context, determined by the airspace to be controlled:

- en-route area: control of aircrafts located at a high and medium altitude,
- terminal / approach areas: terminal area is restricted to major airports while approach area is dedicated to align aircrafts at arrival or in departure in order to pave their way for the Flight Information Region (FIR),
- runways / ground areas: management of aircrafts on airports and on ground between runways and taxiways.

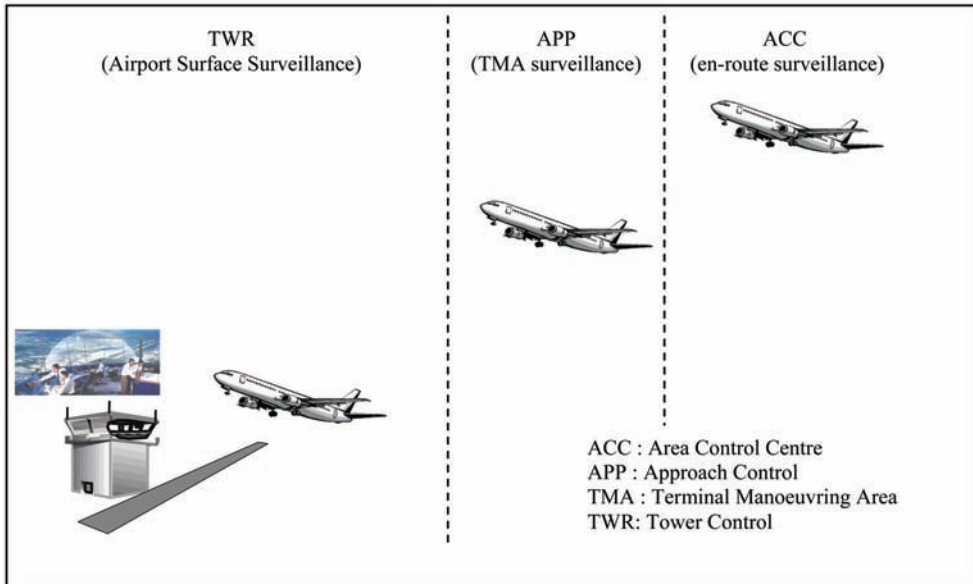


Fig. 1. Air Traffic Control principal areas of applications

In order to be efficient in all the above mentioned situations, the system shall be adaptable and shall be completed by several other sub-systems in order to face:

- differences in aircraft evolution: trajectory for en-route aircraft is more stable than trajectory of an aircraft in terminal area,
- great variety of separation norms: in en-route area for example, the required accuracy for the aircraft positioning is less important than in terminal area,
- failure importance: the loss of radar picture is more important in terminal area than in en-route area where aircrafts are less close to each another.

## 2.1 ATC system

Automatic air traffic management control systems implement main ATC functions which address related ATC services. ATC functions are adaptable to the following rules:

- operational control: real traffic control,
- test and evaluation: all sub-systems are tested and operationally validated (shadow procedures),
- training: training of air traffic controllers on simulated air traffic. All external actors are simulated (i.e. radars, foreign centres, etc.)

- archiving, replay and visualization for legal reasons (accidents, failures, etc.) but also to evaluate the tax amount imposed to the airlines.

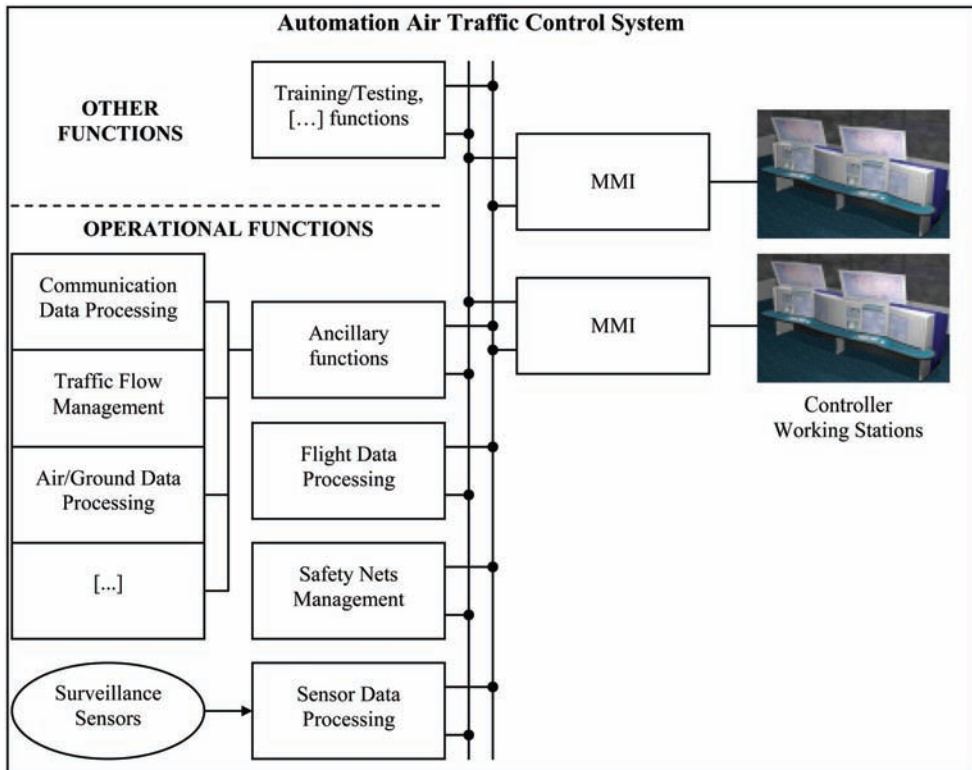


Fig. 2. Automation ATC System synoptic

## 2.2 ATC system services

### 2.2.1 Airspace surveillance service

Surveillance is a key function for airspace management and supports both strategic separation assurance of aircraft and strategic planning of traffic flows.

### 2.2.2 Ground surveillance service

Aircrafts movements on the ground are managed mainly by an A-SMGCS (Advanced Surface Movement Guidance and Control System), depending on the traffic density of an airport.

This system receives information coming from all available sensors (notably from primary surface radar, multilateration, ADS-B and mode S radar) and then processes them into its own fusion module before displaying these data on a specific visualization. This system is able to provide with proximity alerts function (control) and aircraft trajectories optimisation (movement guidance).

This system shall also detects vehicles which are located on runways and tarmacs areas: for this purpose they are equipped with specific beacons.

### 2.2.3 Sensor data processing systems

An ATC automation centre shall take into account data which are sent by numerous surveillance sensors. The rule of a tracking system is then to process and to unify all data, in order to provide a fused information to the visualisation and the safety nets systems.

### 2.2.4 Safety nets management and separation assurance

The surveillance function provides with current aircraft state information to the controller position and to the separation automation functions, i.e. the short term conflict alert (STCA) system for the detection of immediate path conflicts and the Minimum Safe Altitude Warning (MSAW) system for the detection of potential flight into terrain. These automation functions require enhanced surveillance in order to provide with accurate and reliable path predictions for medium term look ahead periods.

### 2.2.5 Visualization systems

A visualization system shows the air situation picture to the controller. The presentation of the information regarding an aircraft shall comply with restricted rules. The following elements are shown on the display:

- a symbol corresponding to the current position and the type of sensor detection,
- other symbols associated to the past-time positions,
- an optional speed vector oriented depending on the course and whose length is proportional to the ground speed,
- a customisable label showing aircraft information: Mode 3/A, Mode C, ground speed, Flight Plan information, etc.

### 2.2.6 Flight data processing services

The Flight Data Processing management service contains all the sub-systems in charge of the flight plan life management (including its modifications and its distribution to other sub-systems).

## 2.3 Surveillance sensors

Surveillance sensors are at the beginning of the chain: the aim of these systems is to detect the aircrafts and to send all the available information to the tracking systems.

Current surveillance systems use redundant primary and secondary radars. The progressive deployment of the GPS-based ADS systems shall gradually change the role of the ground based radars. The evolution to the next generation of surveillance system shall also take into account the interoperability and compatibility with current systems in use.

The figure 3 shows a mix of radar, ADS and Multilateration technologies which will be integrated and fused in ATC centres in order to provide with a high integrity and high accuracy surveillance based on multiple sensor inputs.

### 2.3.1 Primary Surveillance Radar (PSR)

Primary radars use the electromagnetic waves reflection principle. The system measures the time difference between the emission and the reception of the reflected wave on a target in order to determine its range. The target position is determined by measuring the antenna azimuth at the time of the detection.

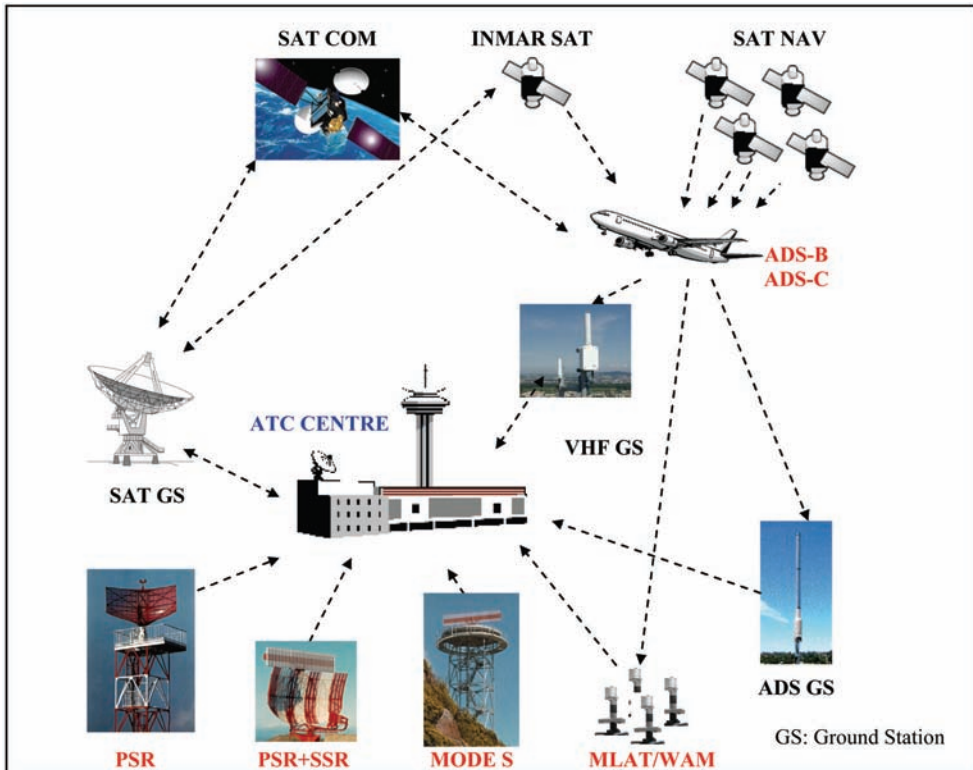


Fig. 3. Surveillance sensor environment

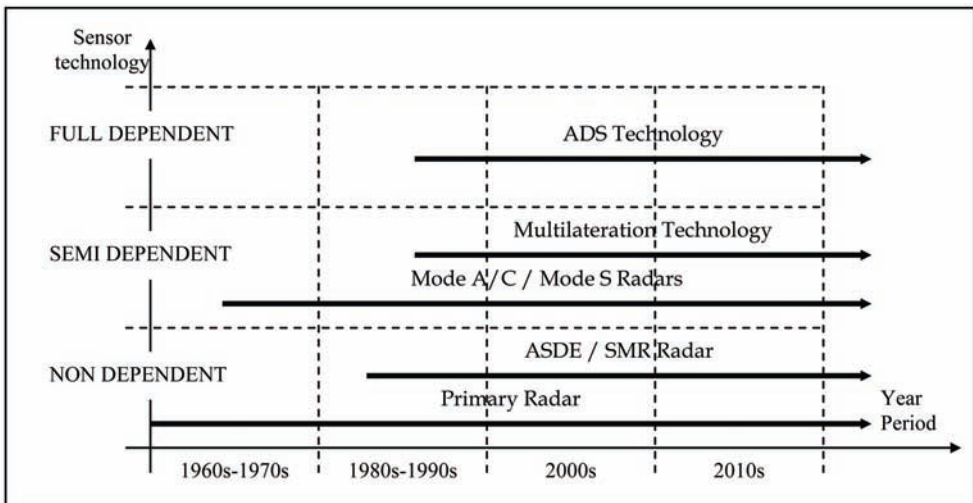


Fig. 4. Historical perspective of surveillance sensor technology

Reflections occur on the targets (i.e. aircrafts) but unfortunately also on fixed objects (buildings) or mobile objects (trucks). These kind of detections are considered as parasites and the "radar data processing" function is in charge of their suppression.

The primary surveillance technology applies also to Airport Surface Detection Equipment (ASDE) and Surface Movement Radar (SMR).

### 2.3.2 Secondary Surveillance Radar (SSR)

Secondary Surveillance Radar includes two elements: an interrogative ground station and a transponder on board of the aircraft. The transponder answers to the ground station interrogations giving its range and its azimuth.

The development of the SSR occurs with the use of Mode A/C and then Mode S for the civil aviation.

Mode A/C transponders give the identification (Mode A code) and the altitude (Mode C code). Consequently, the ground station knows the 3-dimension position and the identity of the targets.

Mode S is an improvement of the Mode A/C as it contains all its functions and allows a selective interrogation of the targets thanks to the use of a unique address coded on 24 bits as well as a bi-directional data link which allows the exchange of information between air and ground.

### 2.3.3 Multilateration sensors

A multilateration system is composed of several beacons which receive the signals which are emitted by the aircraft transponder. The purpose is still to be able to localize the aircraft. These signals are either unsolicited (squitters) or answers (SSR or Mode S) to the interrogations of a nearby radar. Localization is performed thanks to the Time Difference Of Arrival (TDOA) principle. For each beacons pair, hyperbolic surfaces whose difference in distance to these beacons is constant are determined. The aircraft position is at the intersection of these surfaces.

The accuracy of a multilateration system depends on the geometry of the system formed by the aircraft and the beacons as well as the precision of the measurement time of arrival.

Nowadays, multilateration is used mainly for ground movement's surveillance and for the airport approaches (MLAT). Its use for en-route surveillance is on the way of deployment (Wide Area Multilateration (WAM)).

### 2.3.4 Automatic Dependant Surveillance – Contract (ADS-C)

The aircraft uses its satellite-based or inertial systems to determined and send to the ATC centre its position and any other information as:

- aircraft position,
- expected road,
- ground/air speeds
- Meteo data (wind direction and speed, temperature, etc).

ADS-C information are transmitted through point-to-point communications via VHF or via satellite. Ground and on-board equipments managed the transmit conditions (periodical, on event, on emergency, ...)

ADS-C is used typically on desert or oceanic areas as radars cannot insure any surveillance on those areas.

### 2.3.5 Automatic Dependant Surveillance – Broadcast (ADS-B)

The aircraft uses its satellite-based or inertial systems to determine and send to the ATC centre its position and other sort of information. Aircraft position and speed are transmitted one time per second at least.

ADS-B messages (squitters) are sent, contrary to ADS-C messages which are transmitted via a point-to-point communication. By way of consequence, the ADS-B system is used both for ATC surveillance and on-board surveillance applications.

### 2.3.6 Respective advantages and drawbacks

Sensor type	Advantages	Drawbacks
Primary radar (Non-dependant surveillance sensor)	Non cooperative targets detection as no on-board equipment is required. Can be used for ground surveillance. High data integrity level.	Targets cannot be identified. Target altitude cannot be determined. High power emission is required which limits its range. High latency and low update rate.
Secondary radar (Semi-dependant surveillance sensor)	Identity and altitude determination as well as range and azimuth. Less sensitive to interferences than primary radar. Its range is more important than the primary radar as the interrogation and the answer have only one-way distance to cover. Mode S introduces the air-to-ground data link. Medium data integrity level.	Does not work for the ground surveillance due to the loss of accuracy introduced by the delay of the transponder processing. Mode A/C has a lot of issues related to the question/answer confusion. Mode S solves this problem by interrogating the targets in a selective manner. High latency and low update rate.

Table 1. Past and current technology sensor advantages and drawbacks

Sensor type	Advantages	Drawbacks
Multilateration (Semi-dependant surveillance sensor)	SSR technology can be used (does not need any evolution of on-board equipments). Suitable for ground surveillance: needs Mode S equipment as Mode A/C transponders are deactivated mainly on ground in order to limit radio pollution. Small latency. High update rate. Position accuracy.	Signals shall be received correctly by 4 beacons at least which raise the issue of beacons location, especially for en-route surveillance.



Sensor type	Advantages	Drawbacks
ADS-C (Full-dependant surveillance sensor)	Use of surveillance area with no radar coverage. Information “expected road” available. Air/ground data link available.	Depends only on the aircraft (equipped or not) and on the data correction that it sends. Time stamping errors. Very low update rate. GPS outages.
ADS-B (Full-dependant surveillance sensor)	Use for ATC and for on-board surveillance applications. High refresh rate (1s at least). Air/ground data link available. Small latency. High update rate. Position accuracy.	Depends on the aircraft only (equipped or not) and on the data correction which is sent. Not all the aircrafts are equipped at this time. Time stamping errors. GPS outages.

Table 2. New and emerging sensor technology advantages and drawbacks

**2.3.7 Sensor data processing**

As shown in figure 5 hereunder, a sensor data processing is composed generally of two redundant trackers. Radar (including Surface Movement Radar) and ADS-C data are received directly by the trackers while ADS-B and WAM sensor gateways help in reducing the data flow as well as checking integrity and consistency.

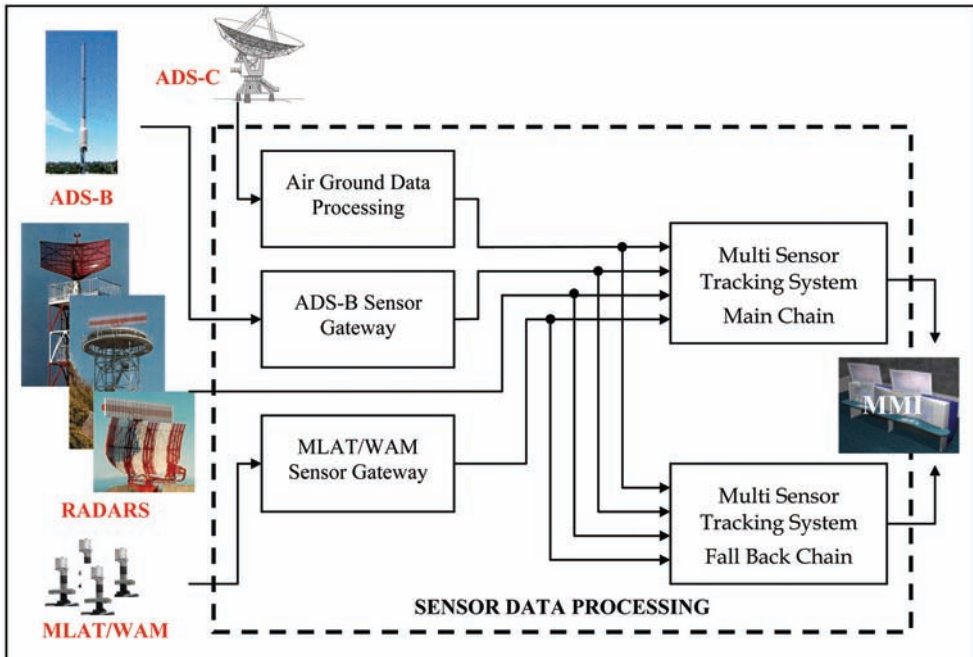


Fig. 5. Sensor Data Processing

As shown in figure 5 above, trackers are potentially redundant in order to prevent from sub-systems failure.

### 3. Sensor data processing architectures

#### 3.1 Data fusion techniques

This paragraph presents the data fusion techniques for radar data processing.

##### 3.1.1 Introduction

Several level of data fusion are available:

- plots (radar reports are directly used into the fusion module),
- tracks (radar reports are used to update local radar tracks which are used into the fusion module),
- mixture between tracks and plots,
- signals (research area).

Several techniques using various level of data fusion are analysed in this paragraph.

Multi radar tracking establishes one track per aircraft which is common to all radars. A great number of methods have been used in ATC centres. These methods can be divided into two main categories:

- selection techniques and
- average or weighted techniques.

An historical perspective of data fusion techniques which have been used in ATC centres is proposed on figure 6 below.

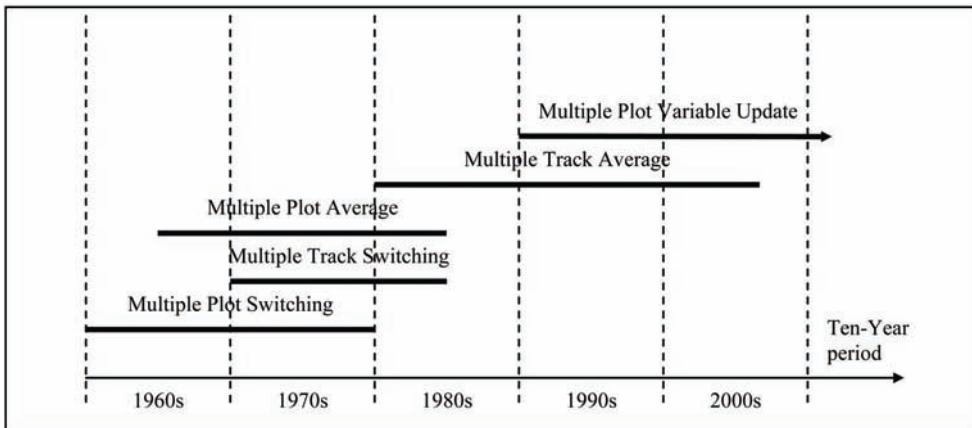


Fig. 6. Historical perspective of principle data fusion techniques

#### 3.1.2 Selection techniques

The selection techniques are also known as “mosaic systems”. Airspace is divided into cells with a pre-determined preferred sensor for each of them. The system receives data from all sensors and choose the appropriate information of each cell in which aircraft is detected.

##### 3.1.2.1 Multiple plot switching method

This method consists in a selection at the radar plots level. Then, at each processing cycle, only one plot is selected among several plots coming from various radars which detect the

aircraft. This plot is used to create or update a common track. Selection is realized according to geographic priority rules (mosaic system).

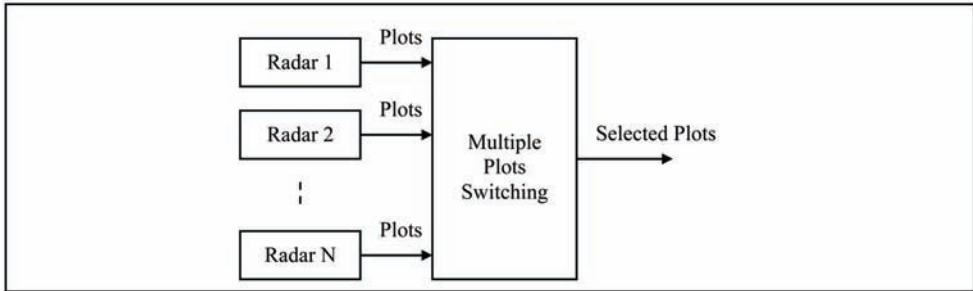


Fig. 7. Multiple plots switching method

### 3.1.2.2 Multiple track switching method

This method consists in a selection at the track level. Then, mono sensor independent tracks are updating for each radar, given several local tracks. Then, one of these local tracks is selected, depending on the relative position of the radars (mosaic) or their respective accuracy.

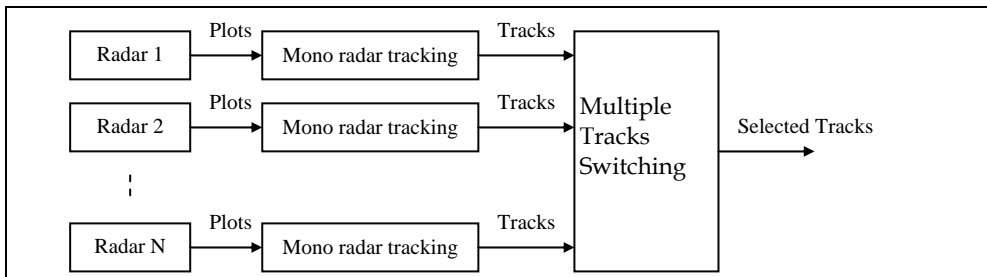


Fig. 8. Multiple tracks switching method

### 3.1.3 Average techniques

#### 3.1.3.1 Multiple track average method

A mono radar track is independently elaborated for each aircraft per radar. When several local tracks are available, the common track is established by weighting (barycentre) of these local tracks (Chong et al., 2000) (Bar Shalom et al., 1988).

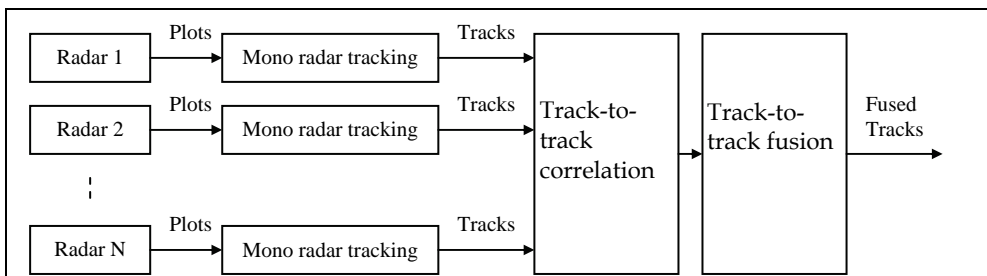


Fig. 9. Multiple tracks average method

### 3.1.3.2 Multiple plot average method

At each processing cycle, the common track established by weighting (barycentre) of the plots coming from various radars.

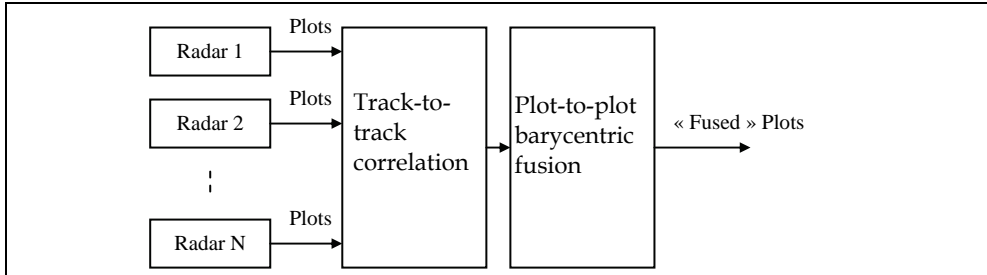


Fig. 10. Multiple plots average method

### 3.1.4 Variable update technique

The major disadvantage of the above-mentioned methods is that they do not use all the available information for one aircraft at one time. This leads to a sub-optimal tracking (Bar Shalom, 1989).

Moreover, the multi radar processing is depreciated in all the above-mentioned methods as the main tracking sub functions are performed in mono radar and a combination of local information is realized afterwards.

The Variable Update method consists in using all the plots coming from any radar to update a unique synthetic common track.

The track update is performed in the fly as soon as sensor reports are received. The reduction of the meantime update in multi-radar configuration improves the accuracy of the track parameters estimation.

In addition, sensors with different characteristics and qualities can be introduced in the same processing. Obviously, Variable Update based tracking constitutes a centralized processing.

Systems using this method are the most efficient. However, they implement a more complex algorithmic as they shall take into account the characteristics of the various sensors and an asynchronous processing of radar plots.

### 3.1.5 Comparison of the various data fusion techniques

Figure 11 below gives an idea of the tracking performance evolution depending on the data fusion techniques versus the CPU load. Indeed, the hardware performance has durably limited the deployment of newest data fusion techniques in ATC system. This is no longer the case with the introduction of the PC technology in early systems.

## 3.2 Radar data processing architecture

The radar data processing proposed in this paragraph is based on a Variable Update data fusion technique. The main chain tracker generally uses this technique while fallback tracker is based mainly on an older technology such as Multiple Track Average technique.

As entry data, the multi radar tracking function has several kind of plots at its disposal which can be primary, secondary or combined. Then, measurements from different radars are allocated so as to update radar tracks (Bar Shalom, 1992).

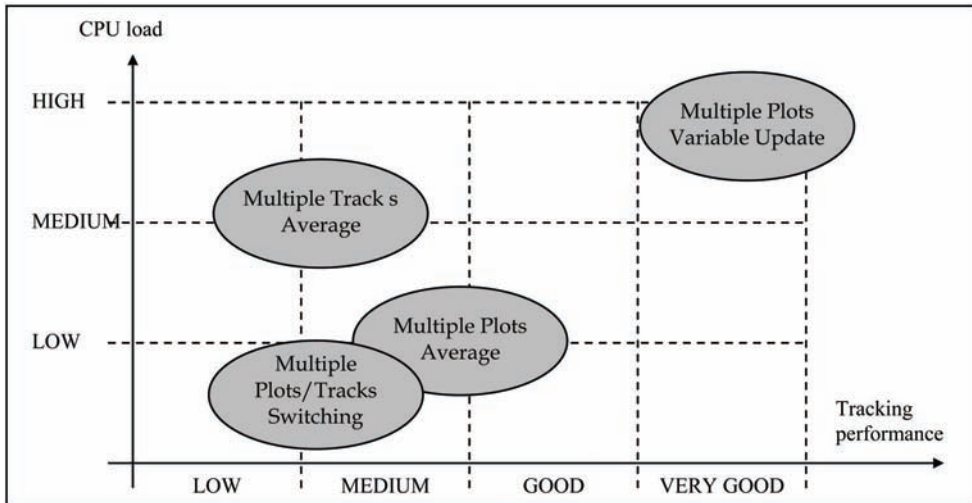


Fig. 11. Data fusion techniques comparison (CPU load versus tracking performance)

The table 3 below gives a quick comparison between Average and Variable Update techniques.

Average technique	Variable Update technique
No variable scanning rate adaptation	Variable scanning rate adaptation
Low CPU load	Medium up to high CPU load
Low track accuracy	Good track accuracy
Low track discrimination	Good tracks discrimination
Manoeuvre detection in long time	Manoeuvre detection in short time
Long initiation time delay (mono sensor)	Short initiation time delay (multi sensor)
Several sensor types integration vulnerability	Several sensor types integration robustness

Table 3. Comparison between Average and Variable Update techniques

A complete description of the Radar Data processing functions is available in (Baud et al., 2006).

The association function is based on NNPD (Nearest Neighbour Probabilistic Data Association) method. Tracks are automatically initialised. The tracking filter which is used is an Interacting Multiple Model (IMM) in the System Cartesian frame. Manoeuvre components are managed through a Multiple Hypothesis Tracking (MHT) method. Bias registration is performed by using a dedicated Kalman filter. All tracks from the air situation picture are distributed periodically (“broadcast mode”) at a specified update rate (set to 5s in most cases).

#### 4. Sensor data processing architectures

Sensor data processing architectures proposed in this paragraph trace the evolutions from the Multi Radar Tracking System (described in paragraph 3 above) to the Multi Sensor Tracking System.

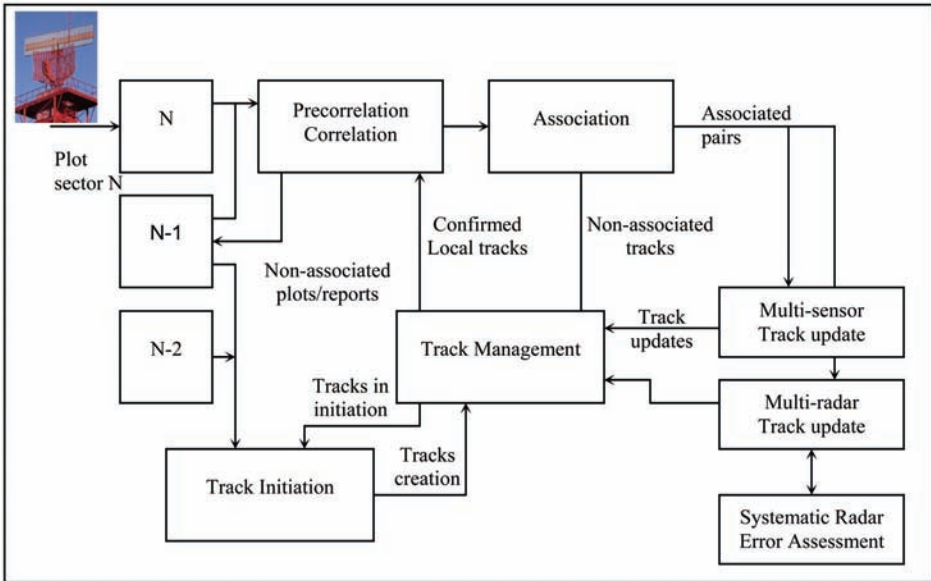


Fig. 12. Multi Radar Tracking System (MRTS) architecture

**4.1 Mode S enhanced-tracking architecture**

This architecture is fully described in paper (Baud et al., 2007). Figure 13 below shows how the Mode S data are introduced into a sensor data processing.

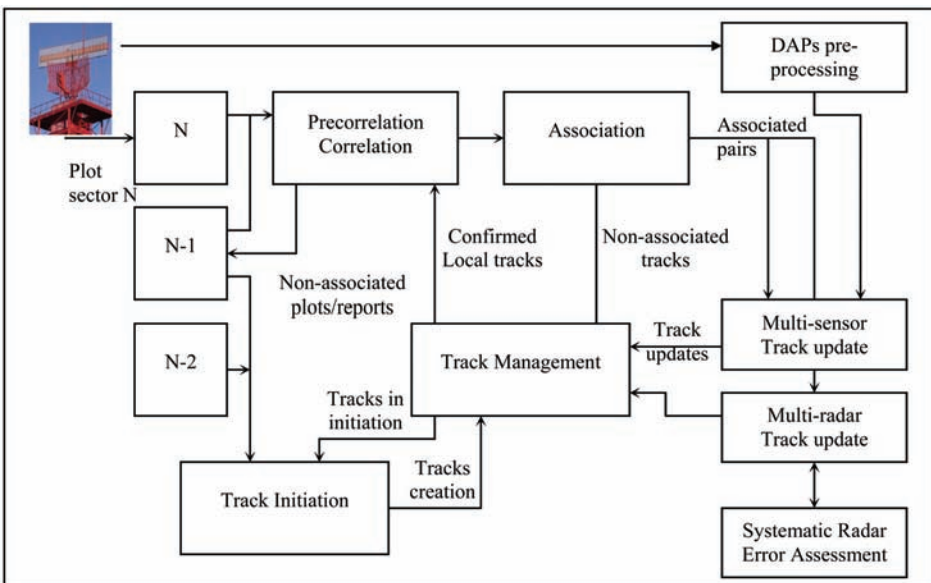


Fig. 13. Mode S enhanced-tracking architecture

Correlation and association processing take into account the 24-bit ICAO Address unique identifier to facilitate the association of a plot to any existing tracks. The Downlinked Aircraft Parameters (DAPs) are used after their own consistency checking to speed up the track initiation and to update the track state vector. The introduction of on-board parameters when updating the track improves the global tracking accuracy especially during manoeuvres (Bar Shalom, 1992).

The track distribution is enhanced so as to provide additional information to the air traffic controllers.

#### 4.2 Radar / ADS data fusion architecture

This architecture is partially described in paper (Baud et al., 2006).

The radar / ADS-B data processing architecture takes advantage of the multiple report variable update technique. A semi-centralized and semi-distributed hybrid architecture is proposed: an internal ADS-B only air situation picture is elaborated as well as a complete Radar / ADS-B fused air situation.

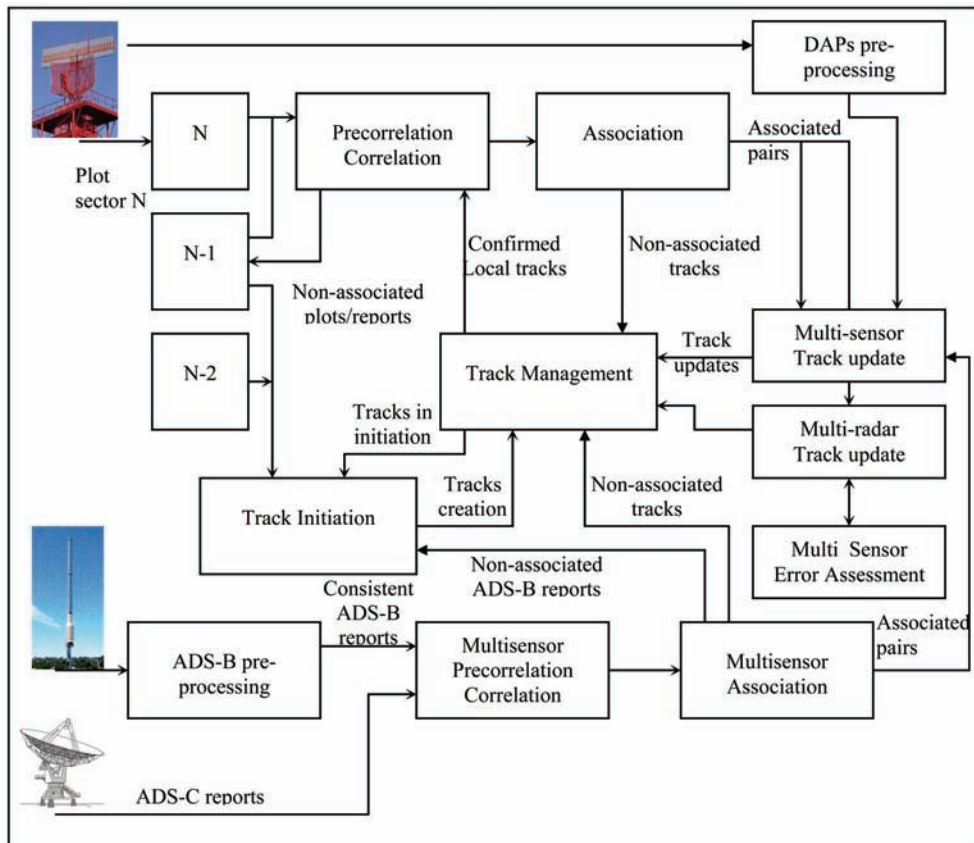


Fig. 14. Radar / ADS-B / ADS-C data fusion architecture

The ADS-B pre-processing sub-function (Besada et al., 2000) consists in the following processing:

- ADS-B report validation against ADS-B only air situation picture,
- ADS-B report validation against other surveillance sources.

This pre-processing is required in order to cope with GPS outages, time stamping issues and GPS/INS on-board switching.

Target models parameterization remains the same as for radar because the addressed targets are identical. Consistent ADS-B reports are used also to improve the radar bias registration. CPU loads issues can be encountered due to the very low data refresh rate (1Hz or more). This is the reason why an ADS-B sensor gateway (Figure 5 above) is used in high traffic density areas.

On the contrary, ADS-C data are directly used by multi sensor pre-correlation, correlation and association processing prior to the update of the multi sensor track state vector. Specific track management processing is performed so as to cope with the low refresh rate of such data. ADS-C data serves mainly in oceanic areas as a gap filler.

### 4.3 Radar / WAM data fusion architecture

Figure 15 below shows how the WAM data are introduced into a sensor data processing.

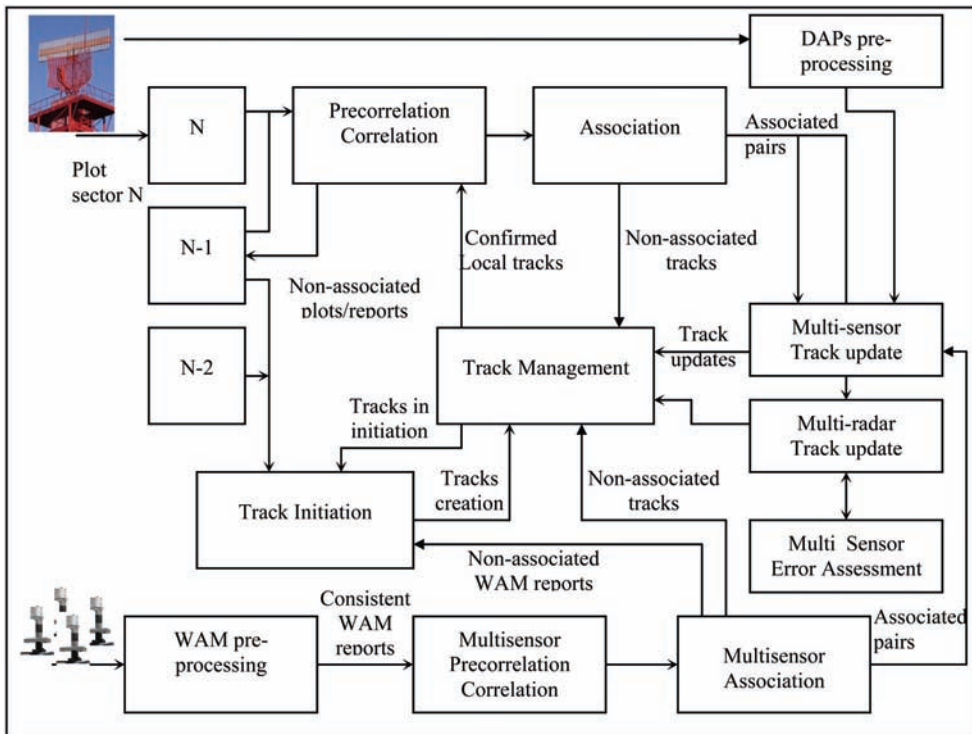


Fig. 15. Radar / WAM data fusion architecture

The radar / WAM data processing architecture takes advantage of the multiple report variable update technique. A semi-centralized and semi-distributed hybrid architecture is



proposed: an internal WAM only air situation picture is elaborated as well as complete Radar / WAM fused air situation picture (Daskalakis et al., 2005).

The WAM pre-processing sub-function consists in the following processing:

- WAM report validation against WAM only air situation picture,
- WAM report validation against other surveillance sources.

This pre-processing is required in order to cope with WAM receivers clock synchronization issues and to determine, given WAM report accuracy, the observability of the target, and then avoid sub-optimal or erratic track behaviour.

Target models parameterisation remains the same as for radar because the addressed targets are identical.

CPU loads issues can be encountered due to the very low data refresh rate (1Hz or more), reason for which a WAM sensor gateway (Figure 5 above) is used in high traffic density areas.

#### **4.4 Architecture enhancements for A-SMGCS**

The first step of A-SMGCS data application into a Multi Sensor Tracking System is briefly explained in this paragraph. The second step is part of the future gate-to-gate surveillance concepts addressed in paragraph 5.2 below.

A-SMGCS sensor is expected to send local tracks that are used to speed up the multi sensor track initiation, especially on Parallel Runway Monitoring (PRM) volumes.

A-SMGCS sensor data are processed as for the WAM data but without dedicated pre-processing.

### **5. Architectures for the future**

The main enhancement in the surveillance environment, which influences the transition from the current conventional environment towards the future CNS/ATM system, is the introduction of new types of sensors (i.e. SSR Mode-S, ADS-B, ADS-C, ASDE, and Multilateration Systems) and the resulting capability to acquire on-board data through the various air-ground data links. The advanced features of the future CNS/ATM environment create the need for modifications both in the internal functionality and the interfaces with the functional entities of the environment (that is to say data sources and users).

First enhancements that concern the use of SSR Mode-S, ADS-B, ADS-C and WAM have been explained in the paragraph 4 above.

#### **5.1 Traffic Information service – Broadcast (TIS-B)**

TIS-B is a service that provides current aircraft surveillance information to airborne systems (and usually the pilot). This is a broadcast service from ground stations sending surveillance information from ground to air. As such, there is no TIS-B data transfer from aircraft to ground and there is no acknowledgement of the reception of TIS-B messages.

An Air Traffic Service Provider (ATSP) collects and correlates surveillance data from radar, multilateration systems and from ADS-B ground stations in order to provide a TIS-B service.

Then, individual surveillance systems data are fused in the sensor data processing, which is in charge of the transfer of fused tracks to the TIS-B system. This determines which TIS-

B targets are required to be broadcasted (by considering the available radar and ADS-B data). Finally the TIS-B ground station broadcasts these targets at regular intervals in order to be received by properly equipped aircraft where the information is presented then to the pilot.

The sensor data processing is in charge of the multi sensor data fusion as well as the validation of the incoming ADS-B reports against radar/multilateration surveillance data. The sensor data processing shall be enhanced so as to cope with the little latency which is allocated to the data fusion system (less than 1s) and the fact that it shall send tracks each time the track state vector is updated ("update mode" versus the above-mentioned "broadcast mode").

## 5.2 Gate-to-gate concept

Typical airport surveillance systems consists in an Advanced – Surface Movement Control and Guidance System (A-SMCGS). These systems mainly use Surface Movement Radar (SMR) or Airport Surface Detection Equipment (ASDE) data, ADS-B and Airport Multilateration (MLAT) data inputs to build the airport tracking situation picture which is displayed to the controllers.

The purpose of the gate-to-gate concept from a surveillance prospective is to provide the controller with a synthetic information that covers aircraft movements for ground, approach and en-route areas. Then, this requires the inclusion of the airport surface surveillance sensors as an input of the sensor data processing system. Enhancements of tracker functionalities are the following:

- dedicated processing which deals with sensor reports anomalies (reflections, side-lobes, outliers and blunders),
- correlation/association improvements to cope with high density multi targets environment,
- target models which deals with new object dynamics,
- target classification and airport map interfacing which helps in the appropriate selection of the models.

Two solutions are proposed:

- distributed architecture with the use of the outputs of an A-SMGCS system (described in paragraph 4.4 above),
- centralized architecture with the fusion of data from all various airport data sources in the sensor data processing module (refer to figures 16 and 17 below).

Figure 17 below does not show the ADS, WAM processing even if they are still persist in the complete integrated architecture.

MLAT pre-processing complies with what is done in WAM pre-processing module.

In normal conditions, mobiles (both aircraft and ground vehicles) are constrained to move only within some restricted areas of the airport (runways, taxiways and roads), each one of which imposing particular motion patterns and kinematics bounds. The restrictions can be either of a physical nature (shape of ways, obstacles, etc.) or rule-based procedures (permitted manoeuvres, circulation direction, etc.).

Ground information is included in the two most important aspects of the estimation process: sensor data pre-processing (characterization of error covariance) and target dynamic modeling to improve the accuracy of final state vector estimations.

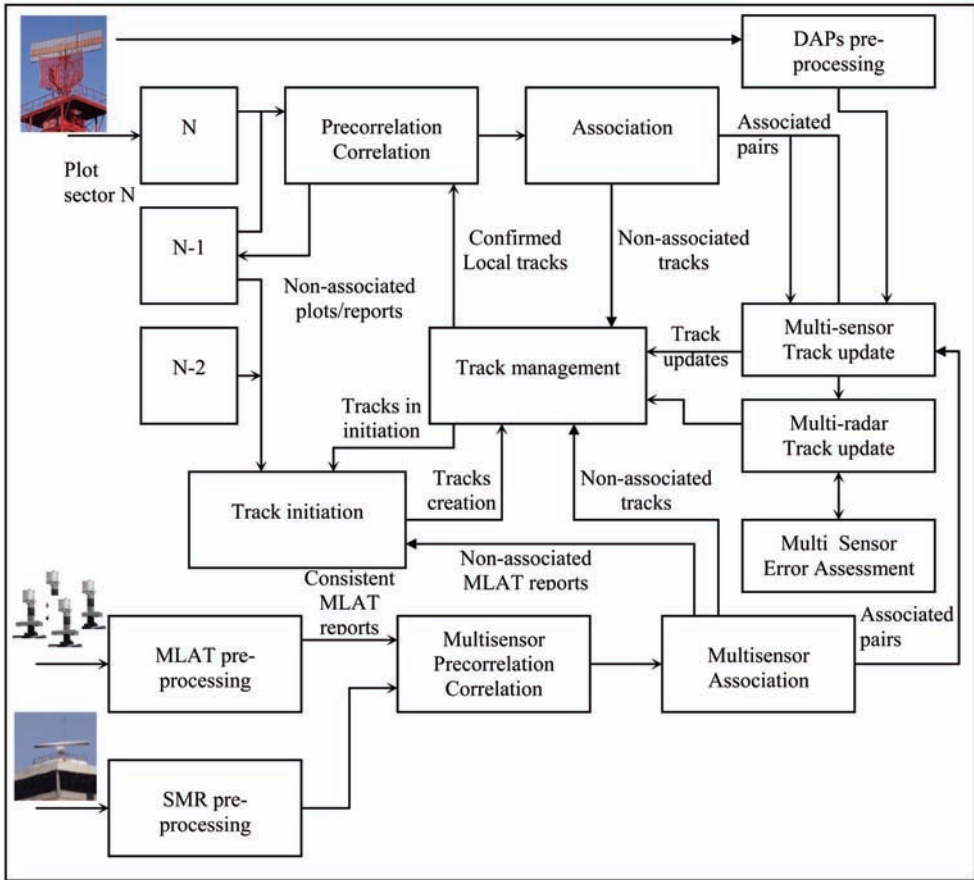


Fig. 16. Centralized air / ground data fusion architecture

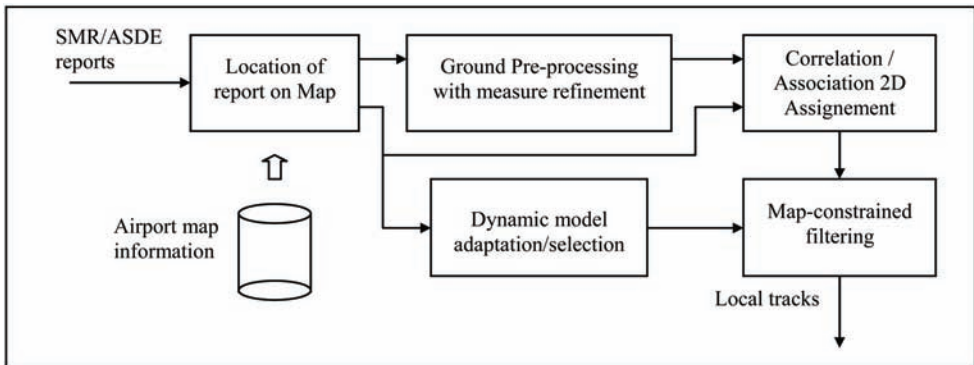


Fig. 17. SMR/ASDE pre-processing synoptic

Either distributed or centralized architectures can be chosen according to the kind of sensor to be integrated and to the tracking accuracy, continuity and integrity metrics to be verified.

## 7. Conclusion

Nowadays, the development of advanced ATM systems is realised by the implementation of advanced means of communication, navigation and surveillance for air traffic control (CNS/ATM).

The definition of a new set of surveillance standards has allowed the emergence of a post-radar infrastructure based on data-link technology. The integration of this new technology into gate-to-gate architectures has notably the following purposes:

- fluxing air traffic which is growing continuously,
- increasing safety related to aircraft operations,
- reducing global costs (fuel cost is increasing quickly and this seems to be a long-term tendency), and
- reducing radio-radiation and improving the ecological situation.

In this context, sensor data processing will continue to play its key role and its software as well as its hardware architecture is expected to evolve in the meantime. The performance requirements of such a sub-system (accuracy, latency, continuity and integrity) is expected to become more and more strict in the incoming years. This issue will be discussed in a future chapter.

## 8. References

- Bar Shalom, Y. (1989). *Multitarget-Multisensor Tracking: Advanced Applications*, Artech House, ISBN 978-0964831223, Norwood
- Bar Shalom, Y. (1992). *Multitarget-Multisensor Tracking: Applications and Advances*, Artech House, ISBN 978-0890065174, Norwood
- Bar Shalom, Y. & Fortmann, T. E. (1988). *Tracking and Data Association*, Academic Press, ISBN 978-0120797608, San Diego
- Baud, O.; Honoré, N. & Taupin, O. (2006). *Radar / ADS-B data fusion architecture for experimentation purpose*, ISIF'06, 9<sup>th</sup> International Conference on Information Fusion, pp. 1-6, July 2006.
- Baud, O.; Honoré, N. ; Rozé, Y. & Taupin, O. (2007). *Use of downlinked aircraft parameters in enhanced tracking architecture*, IEEE Aerospace Conference 2007, pp. 1-9, March 2007.
- Besada, J. A.; Garcia, J. ; De Miguel, G. ; Jimenez, J. F. ; Gavin, G. & Casar, J. R. (2000). *Data fusion algorithms based on radar and ADS measurements for ATC application*, IEEE International Radar Conference 2000, pp. 98-103, May 2000.
- Chong, C. Y.; Mori, S. ; Barker, W. H. & Chang K. C. (2000). *Architectures and algorithms for track association and fusion*, Aerospace and Electronic Systems Magazine, IEEE, Volume 15, Issue 1, pp. 5-13, January 2000.
- Daskalakis, A. & Martone, P. (2005). *Assessing Wide Area Multilateration and ADS-B as alternative surveillance technology*, AIAA 5<sup>th</sup> ATIO and 16<sup>th</sup> Lighter-Than-Air

Sys. Tech. and Balloon Systems Conference, Sep. 26-28, 2005, Arlington, Virginia.

# Sensor Data Fusion in Automotive Applications

Panagiotis Lytrivis, George Thomaidis and Angelos Amditis  
*Institute of Communication and Computer Systems*  
Greece

## 1. Introduction

Sensor data fusion plays an important role in current and future vehicular active safety systems. The development of new advanced sensors is not sufficient enough without the utilisation of enhanced signal processing techniques such as the data fusion methods. A stand alone sensor cannot overcome certain physical limitations as for example the limited range and the field of view. Therefore combining information coming from different sensors broadens the area around the vehicle covered by sensors and increases the reliability of the whole system in case of sensor failure.

In general, data fusion is not something innovative in research; a lot has been done for military applications, but it is rather a new approach in the automotive field. The state-of-the-art in the automotive field is the fusion of many heterogeneous onboard sensors, e.g. radars, laserscanners, cameras, GPS devices and inertial sensors, and the use of map data coming from digital map databases.

A functional model very similar to the Joint Directors of Laboratories (JDL), which is the most prevalent in data fusion, is used in automotive fusion. According to this model the data processing is divided to the following levels: signal, object, situation and application. All these levels communicate and exchange data through a storage and system manager.

The JDL model is only a functional model which allows different architectures for fusion implementation. These architectures are divided in centralized, distributed and hybrid; each one has advantages and disadvantages.

In the data fusion process the main focus is on object and situation refinement levels, which refer to the state estimation of objects and the relations among them, correspondingly. The discrimination between these levels is also made by using the terms low and high level fusion instead of object and situation refinement.

There are several vehicular applications that fusion of data coming from many different sensors is necessary. These can be divided into three main categories: longitudinal support, lateral support and intersection safety applications.

There is a current tendency to exploit also wireless communications in vehicles. Talking cars forming ad hoc networks may be useful in future applications to cover more safety cases that can not be covered so far, due to physical limitations of onboard sensors. In this way the electronic horizon and the awareness of the driver can be extended even to some kilometres away. A lot of ongoing research is focused on the design of efficient protocols and architectures for vehicular ad hoc networks and on the standardization of this kind of vehicular communication.

## 2. The revised JDL model

Sensor data fusion systems can be met in several applications, from military to civilian. Despite the wide variety of all those application domains the data fusion functional model is common and it was developed in 1985 by the U.S. Joint Directors of Laboratories (JDL) Data Fusion Group. The goal of this group was to develop a model that would help theoreticians, engineers, managers and users of data fusion techniques to have a common understanding of the fusion process and its multiple levels. Since then the model was constantly revised and updated and the one described in Fig. 1 is from the 1998 revision (Hall & Llinas, 2001).

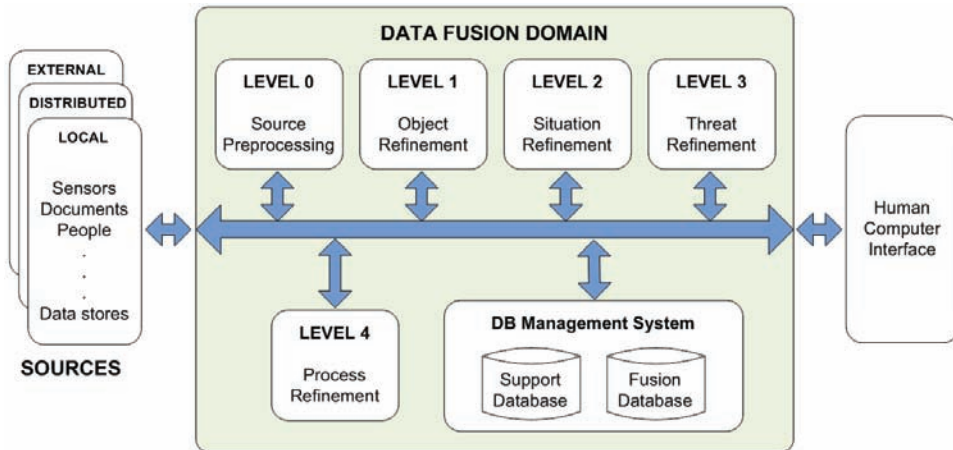


Fig. 1. Joint Directors of Laboratories (JDL) model

- *Level 0*: Preprocessing of sensor measurements (pixel/signal-level processing).
- *Level 1*: Estimation and prediction of entity states on the basis of inferences from observations.
- *Level 2*: Estimation and prediction of entity states on the basis of inferred relations among entities.
- *Level 3*: Estimation and prediction of effects on situations of planned or estimated/predicted actions by the participants.
- *Level 4*: Adaptive data acquisition and processing related to resource management and process refinement.

The question raised is how this model can be applied in multi-sensor automotive safety systems (Polychronopoulos et al., 2006). The corresponding revised JDL model is depicted in Fig. 2.

According to the automotive fusion community, level 4 does not belong to the core fusion process and hence it has been left out of the model in Fig. 2. A key topic in the automotive industry is Level 5, which corresponds to the Human Machine Interface, but it is not considered as part of the data fusion domain (see Fig. 1). While the scope of the first data fusion systems was to replace the human inference and let the system decide on its own, recently the human became more and more important in the fusion process and there are thoughts on extending the JDL model in order to include the human in the loop.

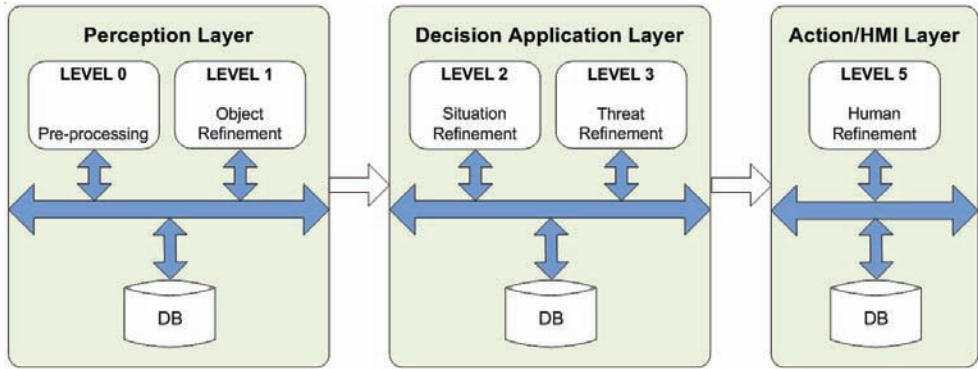


Fig. 2. Revised JDL model for automotive applications

### 3. Fusion architectures

The revised JDL model does not imply explicitly how the fusion process is implemented and how information among different levels is exchanged. Due to this fact a variety of architectures can be extracted from this functional model. Based on the way that information is fused, three different architectures may be implemented: centralized, distributed and hybrid. Each one has its own advantages which are mentioned in the following paragraphs and inside Table 1 (Blackman & Popoli, 1999).

#### 3.1 Centralized architecture

This architecture is theoretically the simplest and ideally has the best performance when all the sensors are accurately aligned, that is when the sensors measure identical physical quantities. In this architecture the raw measurements from all sensors are collected in a central processing level (Fig. 3). On the one hand, this is the main advantage of the centralized architecture, that all raw data is available at the data fusion algorithm. On the other hand, the data fusion algorithm is much more complex compared to the one used in

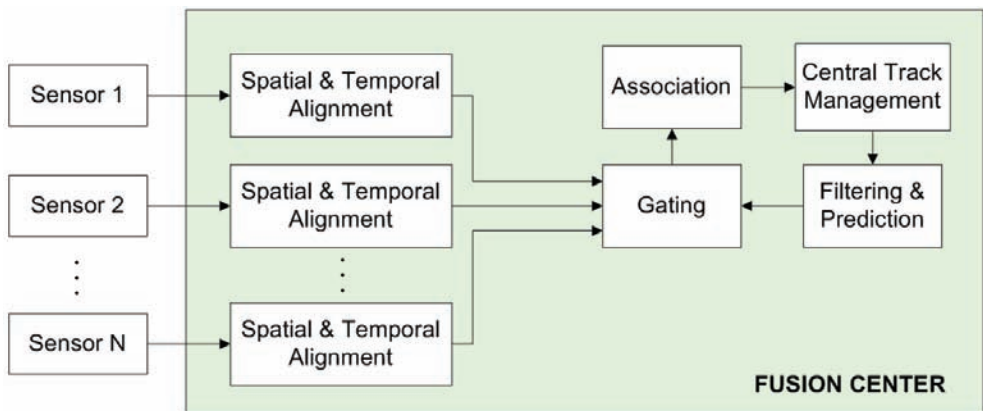


Fig. 3. Centralized Fusion Architecture



the case of distributed architecture, since it has to analyze and process raw data at a higher rate. The Multiple Hypothesis Tracking (MHT) algorithm is easily implemented since all data is available inside the central processor.

Inefficiencies in this method can occur due to the large amount of data that have to be transferred on time in the central processor.

Centralized Architecture	Distributed Architecture
<ul style="list-style-type: none"> <li>• Accurate data association and tracking</li> <li>• Optimization of the estimated position and track of an object</li> <li>• Reduced weight, volume, power and productive cost with regard to distributed architecture (less processors used)</li> <li>• Increased HW reliability (less processors needed in the data fusion chain)</li> <li>• Logic and implementation are direct</li> <li>• Use of Multiple Hypothesis Tracking (MHT) algorithm is direct</li> </ul>	<ul style="list-style-type: none"> <li>• Pre-processing of data reduces the load in the central processor (moderate data transfer requirements)</li> <li>• More efficient utilization of the individual sensor characteristics</li> <li>• Optimization of signal processing in each sensor</li> <li>• Least vulnerable to sensor failure</li> <li>• Flexibility in the number and type of sensors used which allows addition, removal or change of sensors without significant changes in the structure of the fusion algorithm</li> <li>• Cost effective since it allows additional fusion in an existing multi-sensor configuration</li> </ul>

Table 1. Advantages of centralized and distributed architecture

### 3.2 Distributed architecture

The distributed fusion architecture is depicted in Fig. 4. The main advantage of a decentralized architecture is the lack of sensitivity regarding the correct alignment of the sensors. Additionally, this architecture has scalable structure, avoiding centralized computational bottlenecks, is robust against sensor failure and modular.

In the case of distributed fusion pre-processed data is the input in the central processor. For each sensor the signal level processing can be carried out in the frequency domain or in the time domain or in pixel based (image processing) and the final input to the central processor will be the entity with its attributes, with a certain level of confidence for further fusion in central level. The hidden assumption made here is that the sensors are acting independently, which is not true for all the cases. Suffering from redundant information is the main drawback of this architecture.

### 3.3 Hybrid architecture

In hybrid architecture the centralized architecture is complemented from different signal processing algorithms for each sensor, which can provide also input to a backup sensor level data fusion algorithm (Fig. 5). The hybrid architecture keeps all the advantages of the centralized architecture and additionally allows the fusion of tracks coming from individual sensors in a sensor level fusion process. The main disadvantages of this hybrid approach are the increased complexity of the process, the potential high requirements in data transfer and the probable cross correlation between local and central trackers.

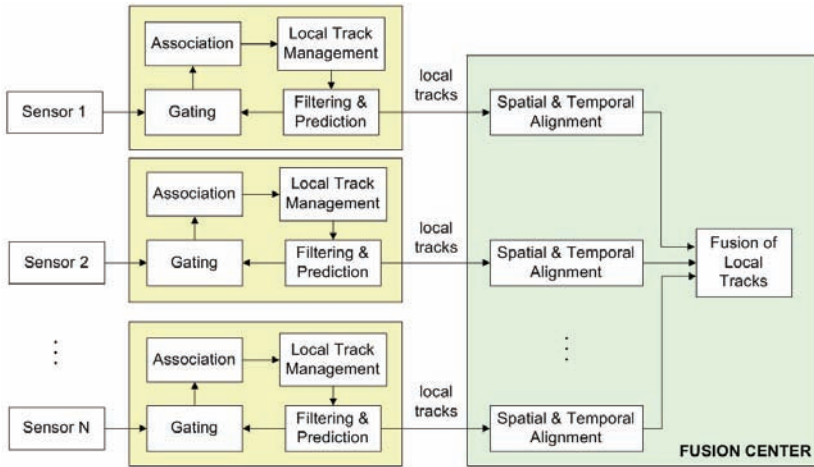


Fig. 4. Distributed Fusion Architecture

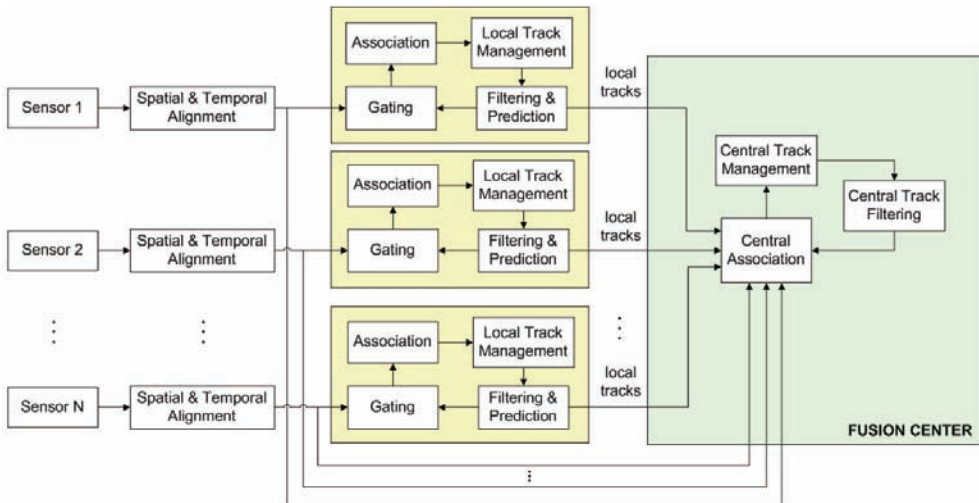


Fig. 5. Hybrid Fusion Architecture

#### 4. Object refinement

Object refinement lies on the first level of the JDL fusion model and it concerns the estimation of the states of discrete physical objects (vehicles in our case). The analysis in this paragraph is based on the distributed architecture that was described previously. The reason for selecting the distributed approach is mainly due to its modularity and theoretically easier adaptation to different vehicles (independently of the sensors used with slight further processing). Hence, it can be considered as the most promising approach for future vehicular applications, if a level of processing is carried out inside each sensor or sensor system and no raw data is used. The main parts of object refinement are the following:

- Measurements pre-processing
- Sensor level tracking
- Spatial & temporal alignment
- Track-to-track association
- Track level fusion algorithm
- Road geometry estimation

#### 4.1 Measurements pre-processing

Sometimes, in practical problems, when the sensors provide raw data a first step of pre-processing is required. For example, the long range radar sensors, used for automotive applications, provide object data as output, which do not need further pre-processing, while the laserscanner sensors provide polygons that need to be classified to vehicles and road borders by implementing appropriate pre-processing.

#### 4.2 Sensor level tracking

This function corresponds to the first boxes in Fig.4, which take as input the sensor measurements. In these boxes gating, association, filtering and local track management take place. First of all, for the tracking algorithm a motion model should be selected for updating the Kalman filter (the transition matrix of the Kalman filter). The motion models that are widely used in the automotive field are the constant acceleration (CA) and the constant acceleration and turn rate model (CTRA) that are described in detail in (Bar-Shalom & Li, 1993; Blackman & Popoli, 1999).

After the selection of the motion model follows the measurement-to-track association problem that is the problem of finding the best association between tracks and measurements. Several association methods exist and the most common are the Global Nearest Neighbor (GNN) and the Joint Probabilistic Data Association (JPDA). The former is one-to-one measurement to track assignment, while the latter uses more than one measurement to update one track and more than one track can be updated by the same measurement. The selection of one of the two methods depends on the quality and nature of the sensor measurements. For instance, for a tracking algorithm carried out for a long range radar sensor, the GNN approach is adequate (Blackman & Popoli, 1999; Floudas et al., 2007). Then, according to the results of the association problem, the track management module should decide for initialization of new tracks and confirmation or deletion of existing tracks. The decision process is based on simple rules of consecutive "hits" and "misses", where a hit is defined when there is a successful association between at least one measurement and a track and a miss when a track remains without an assigned measurement for this cycle of the process (Floudas et al., 2008).

The final step in sensor level tracking is the filtering and prediction function, where the new tentative tracks (unassigned measurements) and the previously existing confirmed tracks are filtered and outputted to the track-to-track association procedure. Also according to the selected motion model the future position of the updated tracks (new and existing) is predicted and the gate for further track processing is calculated. The scope of this gate is to reduce the computational load of checking all measurements with all tracks and just investigate the association of the tracks with measurements that fall inside their gates.

### 4.3 Spatial & temporal alignment

The next step, right after the sensor level tracking, is the spatial and temporal alignment of all the tracks that are coming from the different sensors. For further association and fusion of these tracks a common coordinate system and time reference are needed. In most cases the coordinate system that is used has its origin in the geometrical center of the vehicle and the longitudinal axis is the x-axis. As a time reference the time provided by the Controller Area Network (CAN) bus is used. CAN is actually a network protocol, designed specifically for automotive applications, that allows communication among the electronic control unit(s) of each vehicle with other devices and sensors connected to it. All tracks that are coming from different sensors are fed into the CAN bus and in this way time synchronization is accomplished.

### 4.4 Track-to-track association

After the tracks that are coming from the sensor level tracking have been aligned in space and time, the track-to-track association is executed. The aim of the association is to decide which tracks that are coming from different sensors correspond to the same object. This is useful in cases that we have multiple sensors with common or complementary areas of surveillance. The multidimensional assignment approach is used in case that three or more sensors are observing the same object. For this kind of problems the Lagrangian relaxation method is directly applicable (Deb et al., 1997).

### 4.5 Track level fusion algorithm

There are several methods to update two or more tracks (using state vectors and covariance matrices) with track-to-track fusion; some of them are summarized in the following lines.

Regarding the selection of fusion method for two tracks update several methods are applicable; starting from Simple Fusion (Singer & Kanyuck, 1971) that implies that the tracks are uncorrelated thus it is a suboptimal method. The Weighted Covariance Fusion (Bar-Shalom, 1981; Blackman & Popoli, 1999) accounts for correlation between trackers (common process noise) producing the cross covariance matrix from the existing covariance matrices.

The fusion finally selected when reliable tracks are available is the Covariance Intersection method (Uhlmann, 1995). Covariance Intersection method deals with the problem of invalid incorporation of redundant information.

The Covariance Union method (Uhlmann, 2003) solves the problem of information corruption from spurious estimates. Covariance Union method guarantees consistency as long as both the system and the measurement estimates are consistent, but it is computationally demanding. Covariance intersection method is a conservative solution but superior to weighted covariance method.

However in many practical cases the covariance of obviously not reliable tracks can lead to inaccurate estimates, and therefore a constant predefined weight can be used for these cases. Finally, according to the road environment, the computational load, the process noise, the correlation of sensor measurements and the independency assumption, the proper method for fusion can be selected.

### 4.6 Road geometry estimation

The role of object refinement is not only to estimate the state of each vehicle, but also to estimate the status of other objects in the road environment such as the road borders. Parallel to sensor level tracking a road geometry estimation algorithm is running. The

mathematical model for the road geometry representation could be either the clothoid (Lamm et al., 1999) or the B-Splines (Piegl & Tiller, 1996) model. The basic sensor used for extracting the road geometry is a camera. This camera, after image processing, provides information about the lanes, the lane markings, the curvature of the road etc. and utilizing the clothoid or the B-Splines model a first estimation of the road geometry is calculated.

Moreover, the road geometry is estimated based on information coming from a digital map database. The current position of the vehicle in the map is extracted based on a GPS or a differential GPS sensor and advanced map matching techniques. A way of extracting the road geometry using digital maps is described in detail by (Tsogas et al., 2008a).

The fusion of these estimations to obtain the final road geometry estimation is carried out using a fuzzy system (Jang et al., 1997) or a Dempster-Shafer (Dempster, 1968; Shafer, 1976) reasoning system. Additionally, other active sensors, e.g. radars or laserscanners, can be used as input to the fusion process to increase the robustness of the system (Polychronopoulos et al., 2007; Tsogas et al., 2008a). The fusion process is based on the assumption that camera and laserscanner data is more reliable close to the vehicle, while map data is more accurate far ahead from the vehicle.

## 5. Situation refinement

Situation refinement belongs to the second level of the JDL model and it refers to the relations among the various objects in the road environment. Quite often the term high level fusion is used instead. Within situation refinement the meaning of the current situation around the vehicle is tried to be comprehended. Some questions that are dealt with here are: 'Is this group of slow moving vehicles involved in a traffic jam?', 'Are the trajectories of two vehicles approaching each other intersecting? Is there a danger of collision?' and so on.

The three most known theories that are used in high level fusion, proportional to the problem, are: Fuzzy systems (Dubois & Prade, 1980; Jang et al., 1997), Bayesian probability theory (Bernardo & Smith, 2000; Bolstad, 2007) and Dempster-Shafer theory (Dempster, 1968; Shafer, 1976). In this chapter, an overview of the main parts of situation refinement will be outlined, but the selection of the most appropriate theory is not explicitly indicated.

The outcome of situation refinement enriches the environment model including additional attributes of the ego-vehicle and other objects (e.g. predicted paths, detected maneuvers).

Summarizing, it can be said that situation refinement is the basis to assess the risk of present and predicted future situations, given that all involved participants act in a predictable way.

Finally, situation refinement can be uncertain due to incompleteness of knowledge and uncertain information sources (Tsogas et al., 2007; Tsogas et al., 2008b).

The main parts of situation refinement discussed here, are the following:

- Path prediction
- Maneuver detection
- Driver intention
- Assignment of a lane to an object
- High level events

### 5.1 Path prediction

Path prediction is a key component of situation refinement and it can be divided into three parts. The first part is to calculate the future path of the vehicle based on its current

dynamical state and the adoption of a specific motion model. This model could be the Constant Velocity (CV), Constant Acceleration (CA), Constant Turn Rate (CTR) Constant Turn Rate and Acceleration (CTRA) or Bicycle Model (BM) (Liu & Peng, 1996; Pacejka, 2006), a combination of two or three of these models with the use of an Interacting Multiple Model (IMM) filter or a dynamically adaptive rule-based model. A Kalman Filter is also useful for smoothing the vehicle's dynamics (e.g. speed, yaw rate) and reducing the measurement noise. The second part consists of the extraction of the future path based on the estimation of the road borders and assuming that the driver will follow the road geometry without performing any maneuver. Moreover in this part a dedicated motion model is required. Almost always a CV model suffices. The third and more sophisticated part is the combination of the first two parts. The fusion of these paths can be performed in several different ways. The simplest way is to use a weighted average estimation. For the calculation of the short term future path the dynamic state of the vehicle is more important, while for the long term path the estimation of the road geometry has major influence. Significant work on this issue has been carried out by (Polychronopoulos et al., 2007).

## 5.2 Maneuver detection

The purpose of this algorithm is to identify the maneuver performed by the driver. This calculation can be realized with a Dempster-Shafer reasoning system. At the beginning the set of the maneuvers that the system can detect should be formed. An example set is the following:

$$\Omega = \{\text{free motion, lane change, overtaking, following another vehicle}\}$$

According to the above set and the information sources, the Dempster-Shafer reasoning system can estimate the actual maneuver that is performed by the vehicle. The information sources could be: the estimated time that the vehicle will cross the lane, the minimum distance of the vehicle to the lane marking, the time in which this minimum distance will be achieved, the curvature of the road, the curvature of the vehicle's path and the distance from the vehicle in front. For each one of these sources a basic probability assignment function will be assigned for calculating the evidence masses. Then the fused evidence masses will be calculated and the belief and plausibility values will be extracted in order to evaluate the final confidence.

Here is an example, how the algorithm calculates the performed maneuver: First of all let's assume that the ego vehicle is overtaking another vehicle. The time to cross the lane should have very small values and the ego vehicle should be following another vehicle in relatively small distance. If this is the input information to the system, then the algorithm should detect an overtaking maneuver with high confidence.

## 5.3 Driver intention

Another important function in the situation refinement domain is checking whether the maneuver performed by the driver was intended or not. This can be of great importance especially for the Human Machine Interface (HMI) application. For example, if the output of the driver intention module is that the current performed maneuver is not intended, then there is a high possibility of an upcoming unpleasant situation, so the HMI system should intervene and inform the driver before it is too late. A Dempster-Shafer or Rule based or Fuzzy inference system can be used for identifying the driver's intention. The input sources

to this system comprise the output of maneuver detection algorithm, the type of the road (rural, highway, construction area etc.), the curvature of the road, and other vehicle data such as the status of the indicator (ON/OFF), the velocity etc.

The formulation of the rules in a rule based system or the membership functions in a Fuzzy inference system or the basic probability assignment functions in a Dempster-Shafer system are based on simple guidelines. For instance, the possibility for intended lane change in sharp-curved road segments is lower than in cases of straight road segments. When the curvature exceeds a threshold then it is very unlikely that the driver will change lane.

#### **5.4 Assignment of a lane to an object**

This part of situation refinement is responsible for assigning a lane index to every fused object relative to the future path of the ego vehicle. It indicates the relationship among the detected objects in the road, the lanes of the road and the ego vehicle. A Dempster-Shafer reasoning system is applicable also in this case.

The sources that can be used to estimate the assigned lane index to the object are the following:

- offset of the position of each vehicle from the position of the ego vehicle exploiting the future path calculated previously using different motion models (CA, CTR & CTRA)
- distance of the detected object from the ego vehicle

The offset is calculated using the future trajectory of the ego vehicle and the coordinates of the detected object.

The basic probability assignment functions are formulated based on the following rules:

- The closer the detected object is located to the lane borders, the lower evidence mass is assigned to the corresponding proposition.
- The further the detected object is, the lower the evidence mass assigned to the corresponding information source is.

#### **5.5 High level events**

Since situation refinement is also called high level fusion, high level events such as estimation of weather conditions and traffic, should be taken into account within this fusion level. Both the estimation of the traffic density and of the weather conditions could be based on a Bayesian network approach (Jensen & Nielsen, 2007; Korb & Nicholson, 2004).

As far as the traffic is concerned, it could be classified in light, medium or dense traffic. For this calculation the fused objects from object refinement as well as the road attributes such as lane markings, road offset, lane offset, road width, lane width and heading, curvature and curvature rate of the corresponding segment are needed.

The estimation of the weather conditions (fog, rain, icy road) is much more complex, because for this kind of calculations, input from specific sensors is needed.

### **6. Application and use cases**

In the automotive field there are several applications that fusion of data of various sensors is necessary. For all around coverage and for supporting at the same time a lot of different applications, data fusion becomes a complicated procedure. The sensors used are very heterogeneous and vary in quality. Some sensors are of poor quality, others, like the long range radars, are of high quality but due to their limited field of view support from other

sensors with wider coverage area is necessary. The synchronization of all these sensors, the processing power needed (many embedded PCs), the space they need for installation in the car and the cost comprise constraints for the fast incorporation of such systems in the market. Despite all the above facts, the key challenge in all these applications which would lead the future active safety systems in success is a robust and reliable data fusion.

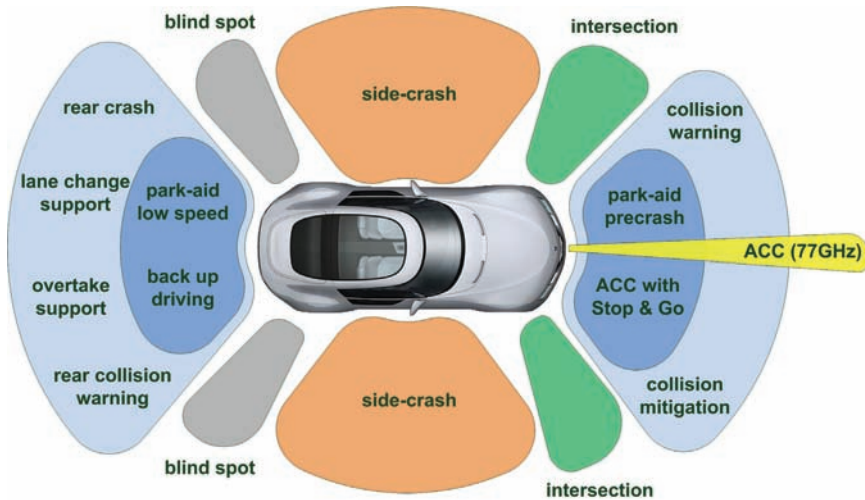


Fig. 6. Coverage areas for various automotive safety applications

The figure above shows many different automotive safety applications and their coverage areas. It is obvious that there is a significant variety of applications in the automotive field, such as Adaptive Cruise Control (ACC), front/rear collision mitigation, parking aid, front/rear collision avoidance, blind spot support, lane change and lane keeping support, vulnerable road users (e.g. pedestrians, cyclists) protection and so on.

The aim of this chapter is not to refer to all these applications but to highlight the most important ones and these that will contribute to the reduction of road accidents and respectively to the fatalities.

### 6.1 Intersection safety

Intersections comprise a major accident hotspot according to statistics, as proved by the data taken out of CARE2005 and provided by Renault. Above 40% of all injury accidents in Europe take place at intersections, while approximately 25% and 35% of the fatalities and the serious injuries come out from intersections respectively. The aim of intersection safety applications is to assist and protect not only the drivers, but also the vulnerable road users (e.g. pedestrians, cyclists). Accident scenarios at intersections are amongst the most complicated, since intersections are frequented by many and different road users approaching from different directions. Some examples of accident scenarios are the following:

- Collisions with oncoming/crossing traffic while turning into or crossing over an intersection
- Violation of the traffic light (red light runner)



For this type of applications advanced on-board sensor systems are necessary, but even such sensors maybe don't suffice. The exploitation of wireless cooperation among the road users and especially infrastructure support at intersections is more than essential. In this paragraph the analysis will be restricted on the in-vehicle systems.

An example of an equipped vehicle with advanced on-board sensors for intersection scenarios is depicted in Fig. 7.

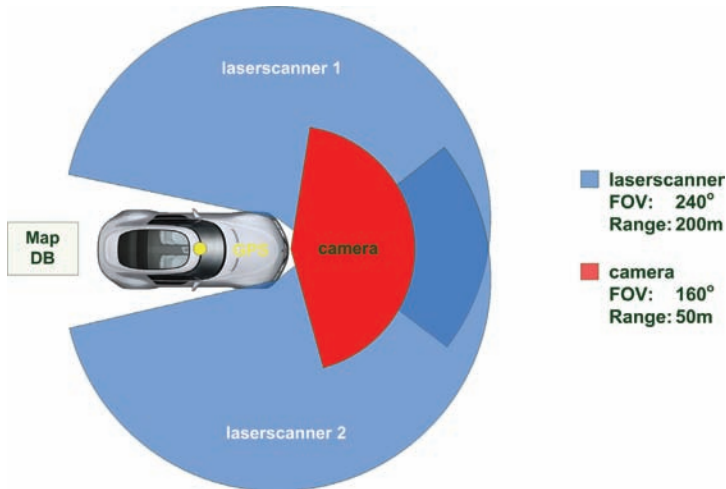


Fig. 7. Equipped vehicle for intersection safety applications

The key factors at intersection are the use of sensors with wide field of view, like the laserscanners, and highly accurate vehicle localisation. The laserscanner can detect other vehicles, pedestrians, cyclists and natural landmarks. The camera, after image processing, can extract information about the lane markings. Highly accurate vehicle localisation can be performed by fusing information from camera, laserscanner and map data extracted from a detailed map of the intersection with the use of a GPS/DGPS sensor.

## 6.2 Safe speed and safe distance

This application belongs to the more general category of longitudinal support systems, which comprise Adaptive Cruise Control (ACC), front/rear collision avoidance, stop and go etc. At this point it should be mentioned that ACC systems were the first systems introduced to a vehicle, which made use of a long range radar sensor. The aim of ACC was to automatically adjust the vehicle's speed and distance from the vehicle ahead. Safe speed and safe distance application is an extension of the traditional ACC system.

In Europe numerous car accidents happen due to inappropriate vehicle's speed or headway. According to European Transport Safety Council, more than 40% of fatal accidents are caused by excessive or inappropriate speed. The higher impact speed the greater likelihood of serious and fatal injury. In addition, rear-end and chain accidents represent a significant part of road accidents in Europe as well.

The aim of a safe speed and safe distance application is to aid the driver in avoiding accidents related to excessive speed or too short headway. Specifically, the sensorial suite of

such a system consists of a long range radar and two medium range radars for obstacle detection, a vision-based system for lane detection and a combination of differential GPS and digital map data for global positioning.

Figure 8 highlights an equipped vehicle for safe speed and safe distance applications.

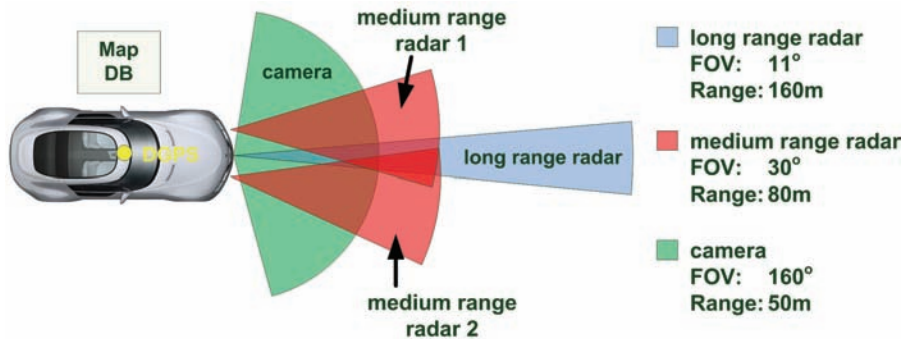


Fig. 8. Equipped vehicle for safe speed and safe distance applications

Data fusion takes place at multiple levels in order to provide an enhanced view of the environment. Differential GPS and inertial sensors are fused together with map data for acquiring a more accurate positioning. Data from the radars is fused and a more complete representation of the environment is achieved. Also for this application the predicted paths of the host vehicle and other vehicles play a key role.

### 6.3 Lane keeping support

Lane keeping support may be considered as a member of the lateral safety applications family. This category comprises also other important applications such as lane change assistance, lateral collision avoidance and lane departure warning. The aim of lane keeping support systems is to assist the driver to keep the vehicle safely in its own lane. For this reason vision-based sensing systems are utilized, which observe the curvature of the road and the position of the vehicle in the lane. In contrast to other warning safety systems, the lane keeping support system utilizes an actuator, which applies a vibration to the steering wheel in order to keep the host vehicle in the lane.

The objective of lane keeping support systems is not to control the vehicle completely automatically, but mainly to give the driver an intuitive support by turning the steering wheel in the right direction. However, the driver has to react and take control of the vehicle so as to avoid lane departure.

If the system detects a lane departure, it issues a warning to the driver and at the same time it activates the steering actuator. This approach exploits data fusion between camera, digital maps and other active sensors like radars or laserscanners. The fact that the system comprises an actuator means that the system is a hard real-time system. This in turn poses tight requirements on the performance of the data fusion algorithms. The figure below shows an equipped vehicle that supports lane keeping applications.

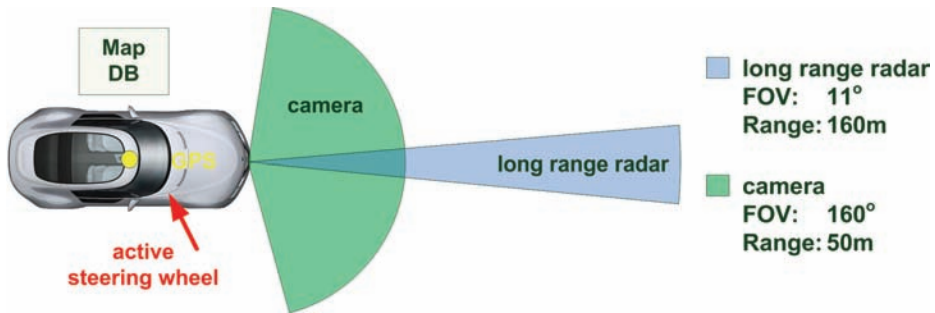


Fig. 9. Equipped vehicle for lane keeping support applications

## 7. Future trends

In the past decade the advances in autonomous sensor technologies and the major objective of the European Union to reduce to a half road accidents and fatalities by 2010, led to the development of advanced driver assistance systems. The fusion of data coming from different advanced in-vehicle sensors was initially in the centre of this attempt. However, this approach suffers from serious limitations. Specifically:

- the perception environment of the vehicle cannot go beyond the sensing range
- the sensor systems cannot perform well in all environments (the urban roads comprise a major challenge)
- in several cases the system is not able to perceive the situation in time in order to warn the driver and suggest a corrective action
- the cost of the sensor systems is too high and so their installation is feasible only at luxurious vehicles.

However, recently there is a lot ongoing research on cooperative vehicles, which focuses on overcoming all the above limitations. There are two different types of communication: roadside-to-vehicle and vehicle-to-vehicle, as pointed out by CAR 2 CAR communication Consortium (Fig. 10). In addition, the exploitation of wireless communications in vehicular environments will enhance and expand currently available safety and comfort applications (e.g. tunnel support, upgrade of intersection safety, internet in the vehicle, ecological driving). A cooperative collision warning application is presented in detail by (Lytrivis et al., 2008).

The limited bandwidth, security issues, privacy, reliability and propagation are some of the emerging disadvantages of the wireless connectivity in vehicles. For all the above reasons new organizations, initiatives and working groups, such as DSRC, WAVE, C2C-CC, were created.

Dedicated Short Range Communications (DSRC) is a short to medium range (1000 meters) communications service that supports both public safety and private operations in roadside-to-vehicle and vehicle-to-vehicle communication environments by providing very high data transfer rates. It operates at 5.9 GHz and provides a spectrum of 75 MHz.

The design of an effective communication protocol that deals with privacy, security, multi-channel propagation and management of resources is a challenging task that is currently under intensive scientific research. A dedicated working group has been assigned this specific task by IEEE and the ongoing protocol suite is the IEEE 1609, mostly known as WAVE (Wireless Access in Vehicular Environments).

The CAR 2 CAR Communication Consortium (C2C-CC) is a non-profit organization initiated by European vehicle manufacturers, which is open for suppliers, research organizations and other partners. The goal of the C2C-CC is to standardize interfaces and protocols of wireless communications between vehicles and their environment in order to make the vehicles of different manufacturers interoperable and also enable them to communicate with road-side units.

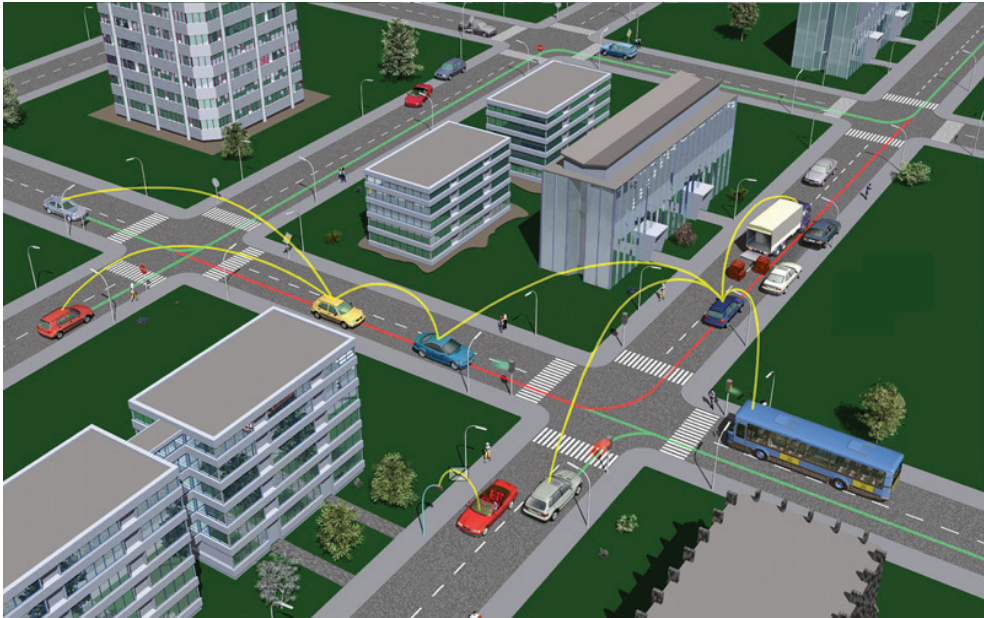


Fig. 10. Vehicles cooperating with other vehicles and roadside units (<http://www.car-to-car.org/index.php?id=131>)

Additionally, new challenges are posed to the data fusion process. The association and synchronization of data from on-board sensors together with the wireless network data is the main challenge. Moreover, the manipulation of delayed information and the reliability of the information transferred via the network are other important issues.

## 8. Conclusion

This chapter has summarized the state-of-the-art in sensor data fusion for automotive applications, showing that this is a relatively new discipline in the automotive research area, compared to signal processing, image processing or radar processing. Thus, there is a

tendency of using already available knowledge from other research areas, such as the military or robotic areas. The initial and the revised JDL functional fusion model, applicable for automotive industry, have been highlighted. Several discrete architectures were described. On the one hand, it can be stated that central fusion architecture, which uses more sensor data at the processing level, is able to deliver the higher performance. On the other hand, the processing demands and the integration effort are much more significant compared to the distributed fusion architecture. Moreover the two main levels of fusion, object and situation refinement, and their corresponding functions were outlined. Additionally, some automotive applications which make use of data fusion were described. Finally, a brief report about the current research activity and the new challenges derived from the exploitation of wireless communications were indicated.

## 9. References

- Bar-Shalom, Y. (1981). On the track-to-track correlation problem. *IEEE Transactions on Automatic Control*, Vol. 26, No. 2, April 1981, pp. 571-572, ISSN 0018-9286
- Bar-Shalom, Y. & Li, X. R. (1993). *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Norwood
- Bernardo J. & Smith A. (2000). *Bayesian Theory (Wiley Series in Probability and Statistics)*, John Wiley & Sons Ltd, ISBN 978-0471494645
- Blackman, S. & Popoli, R. (1999). *Design and Analysis of Modern Tracking Systems*, Artech House, ISBN 1-58053-006-0, United States of America
- Bolstad, W. (2007). *Introduction to Bayesian Statistics (second edition)*, Wiley-Interscience, ISBN 978-0470141151, United States of America
- Deb, S.; Yeddanapudi, M.; Pattipati, K. & Bar-Shalom, Y. (2007). A Generalized S-D Assignment Algorithm for Multisensor-Multitarget State Estimation. *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, No. 2, April 1997, pp. 523-538, ISSN 0018-9251
- Dempster, A. (1968). A generalization of Bayesian inference. *Journal of the Royal Statistical Society, Series B*, Vol. 30, 1968, pp. 205-247
- Dubois, D. & Prade, H. (1980). *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, New York
- Floudas, N.; Lytrivis, P.; Polychronopoulos, A. & Amditis, A. (2007). On the track-to-track association problem in road environments, *Proceedings of 10<sup>th</sup> International Conference on Information Fusion*, ISBN 978-0-662-45804-3, Québec, Canada, July 2007
- Floudas, N.; Lytrivis, P.; Avgoustidis, H.; Ahrholdt, M.; Thomaidis, G. & Amditis, A. (2008). Track Based Multi Sensor Data Fusion for Collision Mitigation, *Proceedings of 11<sup>th</sup> International Conference on Information Fusion*, Cologne, Germany, July 2008
- Hall, D. & Llinas, J. (2001). *Handbook of Multisensor Data Fusion*, CRC press LLC, ISBN 0-8493-2379-7, United States of America

- Jang J.-S. R.; Sun, C.-T. & Mizutani E. (1997). *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice Hall, ISBN 978-0132610667, United States of America
- Jensen, F. & Nielsen, T. (2007). *Bayesian Networks and Decision Graphs*, Springer-Verlag, ISBN 978-0-387-68281-5, New York
- Korb, K. & Nicholson A. (2004). *Bayesian Artificial Intelligence*, Chapman & Hall/CRC, ISBN 1-58488-387-1, United states of America
- Lamm, R.; Psarianos, B. & Mailaender, T. (1999). *Highway design and traffic safety engineering handbook*, McGraw-Hill, ISBN 978-0070382954, New York
- Liu, C.-S. & Peng, H. (1996). Road Friction Coefficient Estimation For Vehicle Path Prediction. *Vehicle System Dynamics*, Vol.25 Suppl., pp.413-425
- Lytrivis, P.; Thomaidis, G. & Amditis, A. (2008). A vehicle-to-vehicle cooperative collision warning application, *Proceedings of 10<sup>th</sup> International Conference on Application of Advanced Technologies in Transportation*, Athens, Greece, May 2008
- Pacejka, H. (2006). *Tyre and Vehicle Dynamics (second edition)*, Butterworth-Heinemann Ltd, ISBN 980-0-7506-6918-4, United Kingdom
- Piegl, L. & Tiller, W. (1996). *The NURBS Book (second edition)*, Springer, ISBN 978-3540615453, Germany
- Polychronopoulos, A.; Amditis, A.; Scheunert, U. & Tatschke, T. (2006). Revisiting JDL model for automotive safety applications: the PF2 functional model, *Proceedings of 9<sup>th</sup> International Conference on Information Fusion*, ISBN 0-9721844-6-5, Florence, Italy, July 2006
- Polychronopoulos, A.; Tsogas, M.; Amditis, A. & Andreone, L. (2007). Sensor Fusion for Predicting Vehicles' Path for Collision Avoidance Systems. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 8, No. 3, September 2007, pp. 549-562, ISSN 1524-9050
- Shafer, G. (1976). *A Mathematical Theory of Evidence*, Princeton University Press, ISBN 0-608-02508-9, Princeton
- Singer, R. A., & Kanyuck, A. T. (1971). Computer control of multiple site track correlation. *Automatica*, Vol. 7, No. 3, pp. 455–463, 1971
- Tsogas, M.; Polychronopoulos, A. ; Floudas, N. & Amditis, A. (2007). Situation refinement for vehicle maneuver identification and driver's intention prediction, *Proceedings of 10<sup>th</sup> International Conference on Information Fusion*, ISBN 978-0-662-45804-3, Québec, Canada, July 2007
- Tsogas, M.; Lytrivis, P. & Amditis, A. (2008). Enhanced curve speed warning application using multiple sources of information for extracting road geometry, *Proceedings of 10<sup>th</sup> International Conference on Application of Advanced Technologies in Transportation*, Athens, Greece, May 2008
- Tsogas, M.; Dai, X.; Thomaidis, G.; Lytrivis, P. & Amditis, A. (2008). Detection of maneuvers using evidence theory, *Proceedings of IEEE Intelligent Vehicles Symposium 2008*, pp. 126-131, ISBN 978-1-4244-2568-6, Eindhoven, The Netherlands, June 2008
- Uhlmann, J. (1995). Dynamic map building and localization: new theoretical foundations. Ph.D. Thesis, University of Oxford

Uhlmann, J. (2003). Covariance Consistency Methods for Fault-Tolerant Distributed Data Fusion. *Elsevier Information Fusion*, Vol. 4, No. 3, September 2003, pp. 201-215

# Multisensor Data Fusion Strategies for Advanced Driver Assistance Systems

Mahdi Rezaei Ghahroudi<sup>1</sup> and Reza Sabzevari<sup>2</sup>

<sup>1</sup>*Department of Computer Engineering, Islamic Azad University of Qazvin*

<sup>2</sup>*Member of Young Researchers' Club (YRC)*

*I.R. Iran*

## 1. Introduction

Multisensor data fusion and integration is a rapidly evolving research area that requires interdisciplinary knowledge in control theory, signal processing, artificial intelligence, probability and statistics, etc. Multisensor data fusion refers to the synergistic combination of sensory data from multiple sensors and related information to provide more reliable and accurate information than could be achieved using a single, independent sensor (Luo et al., 2007). Actually Multisensor data fusion is a multilevel, multifaceted process dealing with automatic detection, association, correlation, estimation, and combination of data from single and multiple information sources. The results of data fusion process help users make decisions in complicated scenarios. Integration of multiple sensor data was originally needed for military applications in ocean surveillance, air-to air and surface-to-air defence, or battlefield intelligence. More recently, multisensor data fusion has also included the non-military fields of remote environmental sensing, medical diagnosis, automated monitoring of equipment, robotics, and automotive systems (Macci et al., 2008).

The potential advantages of multisensor fusion and integration are redundancy, complementarity, timeliness, and cost of the information. The integration or fusion of redundant information can reduce overall uncertainty and thus serve to increase the accuracy with which the features are perceived by the system. Multiple sensors providing redundant information can also serve to increase reliability in the case of sensor error or failure. Complementary information from multiple sensors allows features in the environment to be perceived that are impossible to perceive using just the information from each individual sensor operating separately. (Luo et al., 2007)

Besides, driving as one of our daily activities is a complex task involving a great amount of interaction between driver and vehicle. Drivers regularly share their attention among operating the vehicle, monitoring traffic and nearby obstacles, and performing secondary tasks such as conversing, adjusting comfort settings (e.g. temperature, radio.) The complexity of the task and uncertainty of the driving environment make driving a very dangerous task, as according to a study in the European member states, there are more than 1,200,000 traffic accidents a year with over 40,000 fatalities. This fact points up the growing demand for automotive safety systems, which aim for a significant contribution to the overall road safety (Tatschke et al., 2006). Therefore, recently, there are an increased number of research activities focusing on the Driver Assistance System (DAS) development in order



to reduce the driver's workload and prevent driving accidents and several types of safety systems have therefore been proposed to help lessen the danger and assist the driver (Hsieh et al., 2007). Current technology field of the automotive industry focuses on the development of active safety applications and advanced driver assistant systems (ADAS) instead of passive safety systems. Passive safety systems such as seatbelts and airbags provide protection in the case of collision; more recently however, active safety systems have been introduced to help the driver avoid collisions in the first place. Nowadays, systems such as lane departure warning and rear-end collision avoidance have been introduced (Amditis et al., 2006); (Mammar et al., 2006). These active safety systems are required to interact much more with the driver than passive safety systems, creating a closed loop between driver, vehicle, and the environment. Examples of such systems could be found in the Laboratory for Intelligent and Safe Automobiles (LISA) (Trivedi et al., 2007). An ADAS shall support the driver in his/her task to drive the vehicle providing additional information or warning when encountering any dangerous situation. The system should be equipped with various types of sensors to observe the environment around the vehicle. For example, radar and laser scanners are sensors used to measure distance and velocity of objects, and video cameras are used to detect the road surface and lane markings or to provide additional visual information (Leohold, 2004). This aspire a reduction or at least an alleviation of traffic accidents by the means of collision mitigation procedures, lane departure warning, lateral control, safe speed and safe following measures (see figure 1).

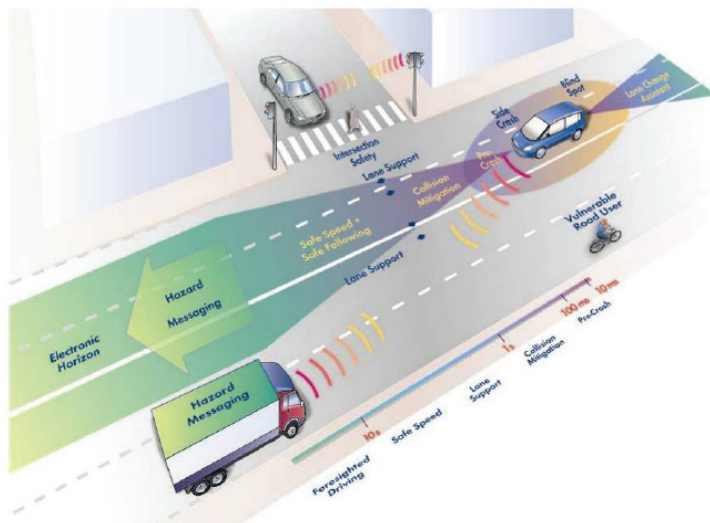


Fig. 1. Safety zone around the vehicle by implementation of multi sensory (PreVENT, 2006)

This chapter describes data fusion concepts, an applicable model, paradigm of multisensor fusion algorithms, current sensor technologies and some applications such as object tracking, identification and classification and a providence view on next-generation car safety and driver assistance systems. The later applications are particularly suitable to provide an overview of multisensor data fusion starting from the plain detection of multiple objects around a given host vehicle to inferring: the position and speed of possible obstacles, the type of objects in the road environment, the relative movement, and distribution of

obstacles over a given area, the early detection of a possible collision, possible suggestions for prompt and effective countermeasures (e.g., sudden braking, steering wheel adjustment, etc.). In this chapter, a new framework of active ADAS is proposed. The framework implements a data-fusion architecture of multiple sensors, including several key functionalities in ADAS, and related applications for different driving conditions and vehicle's surround as well as drivers state monitoring to detect the amount of driver's fatigue and his/her inattention. Our proposed ADAS consists of functionally isolated sensors to observe the environment around the vehicle and the vehicle inside situation. Commonly used sensors in an ADAS are included in the framework and the multi-sensor selection is formulated as an optimal programming problem and implemented in a two relevant simulator. The Ultimate goal of proposed ADAS is to create safety zones around vehicles by the means of embedded multisensor data fusion systems that sense environment, so detects the type and significance of impending dangers. Depending on the nature of the threat, an active and preventive safety system will inform, warn, and actively assist the driver to avoid an accident or mitigate its possible consequences.

## 2. Requirements of advance driver assistance systems

Many existing robotics technologies apply to intelligent assistance driving; however, much research works neglect the preview of the driver and driver response delay. Moreover, the behavior of high-speed vehicles differs greatly from other robots. To obtain safe driving, a driver should be in the center of the "safety analysis" not on the center of driving process; that is one of the key differences of our method than other researches. Driver's response delay, together with other factors, restricts the driving path of a vehicle. In proposed model not only we use multiple and optimal sensor fusion technique but also apply the driver's behavior and degree of his/her alertness in overall control of the host vehicle in a real time manner through a supervisor unit (Rezaei Ghahroudi & Fasih, 2007). The supervisor unit is a decision unit that may sometimes directly interpose the actuators to control the vehicle in dangerous situations.

The other key difference in this method, unlike other researches that focused in one particular application, is that we believe in order to control the vehicle similar to an expert and sober driver, the ADAS systems should not work independently. Instead, after sensor data fusion and decision making, it should run several driving assistance system simultaneously for reliable and robust control of the vehicle. It means in a real word, several actuators should take an appropriate action simultaneously, while most of current DAS systems on the market just carry out one single system at a time; for example Automatic Cruise Control (ACC) or Park assistance each of them act individually. For better understanding, imagine the following example. Encountering an unexpected obstacle in the road will cause heavy braking by all of the vehicles in all of the lanes; this may bring about deviation of other vehicles in an unpredictable way. In such situations, in addition to decreasing speed, the system should correct its direction (steer angle) to mitigate from accident or lessen the scathes and in ideal state, preventing incidence (Figure 2).

In figure 2 there are four successive steps come one by one. At First because of e.g. falling a tree or collapse of hillside (unpredictable obstacle) on the road, the entire lines blockaded and consequently all the vehicles should stop immediately. Suddenly ACC system of the host vehicle (Green car) detects that the foreside vehicle (blue car) has stopped (step1). Then emergency braking system starts to decrease the speed of the vehicle (step2). During step

two, side ultrasonic sensors detect a car on the left side that is coming close and close (step 3), so the supervisor unit decides to intervene and correct the steer angle to escape from the left car (silver car) and correct its path (step 4).

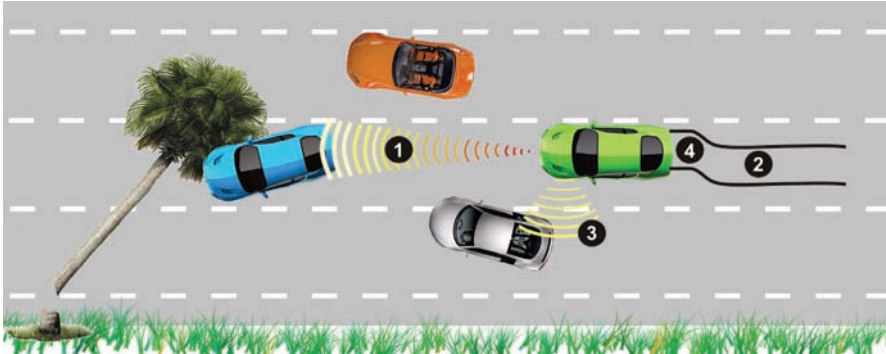


Fig. 2. Schematic of a typical accident in real world

It means, in such cases it needs to act simultaneously several driver assistance systems such as emergency braking system, steer angle correction, lateral collision warning system, front monitoring and keep in lane monitoring. Now let's discuss the design and development of a multi sensor selection in order to cover most of essential vital driver assistance systems in a dynamic scenario. In order to cover different driving conditions, a set of sensor should be established for identifying the driving environment.

### 3. Optimal sensor selection and multi sensor assembly

In a typical scenario, a combination of vehicles in different position of the road are facing various driving conditions such as driving straight, turning, overtaking a vehicle, meeting pedestrians, etc. For these different conditions, the ADAS equipped vehicle need different sets of sensors to detect environment-related information and determine a correct driving condition. The decision on selecting a proper set of object-detecting sensors should be made based on the capability of available sensors and real-time driving condition. Now, In order to formulate and simulate the selection of object-detecting sensors with respect to various driving situations, the sensors should be capable of evaluating *Driver commands* (steer angle setting, backing, changing lane, turning a corner and overtaking a vehicle), *Relative Vehicle's Velocity*, *Traffic Flow* (Low or Dense), and *Driver's behavior* (Observant, sleepy, drowsy, aggressive, using cell phone, etc.), (Hsieh et al., 2007).

The combination of these parameters will be used to reflect a proper diving situation encountered by the driver. So we need an optimal selection of some appropriate sensors to monitor all these four factors (Kaempchen and Dietmayer, 2003). But, which sensor is better and optimal? Image sensors have some drawbacks, such as low ability of sensing depth and advantage of higher ability of discrimination than LIDAR and RADAR. Radar shows limited lateral spatial information because it is not available at all, the field of view is narrow, or the resolution is reduced at large distances. Although LIDAR has a wide view field that solves part of the previous problems, there are other problems such as low ability of discrimination, clustering error, and recognition latency. These restrictions of the different sensor types explain the attention given to sensor fusion in research on object detection and

tracking (Cheng et al., 2007). According to advantages and drawbacks of mentioned sensors and various real driving situations, we implement an optimal selection of state of the art and most typical sensors that can be classified into four different types: Radars ( $R_L$ ,  $R_S$ ), Laser scanner (L), Camera Sensors (C) and Ultrasonic Sensor (U). In this project, 16 object-detecting sensors among four types of sensors are considered to be assembled on the test vehicle (Figure 3)

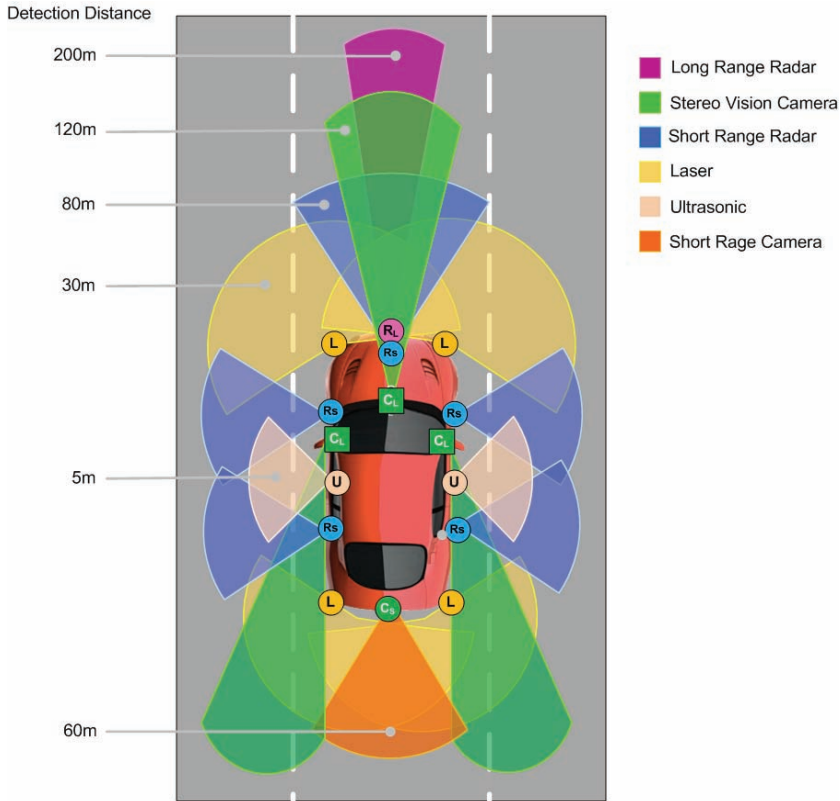


Fig. 3. Sensor placement with full coverage, redundancy and sufficient overlapping area

The sensor network assembly consists of one Long Range Radar-LRR (2nd generation long range radar by Bosch) mounted for front monitoring ( $R_L$ ), five Short Range Radar sensors (from M/A-COM / Tyco Electronics) four of them in both sides and one in the front ( $R_S$ ). Four laser scanners (from IBEO GmbH) with broad range and wide viewing angle, two of them in front and the others for rear side (L), three CMOS cameras INKANSC640PG by Aglaia GmbH (Amditis et al., 2006) two of them in side mirrors for rear and blind spot coverage and one in the middle of the front windscreen to face forward ( $C_L$ ); A short range monocular camera for backside ( $C_S$ ), and finally two ultrasonic on both sides (U). The placement of these object-detecting sensors and the main characteristics, such as detecting distance and the field of view, for these object-detecting sensors can be seen in figure 3. All the sensors transmit data through a CAN-interface (Wahl & Doree, 2000) to the perception layer.

The placement of these 16 object-detection sensors are based on the following six main functionalities required in a state of the art ADAS. The six key functionalities are adaptive cruise control (ACC), lane departure warning (LDW), lane change assistant (LCA), rear view (RV), lane keeping assistance (LKA) and Emergency Braking System (EBS).

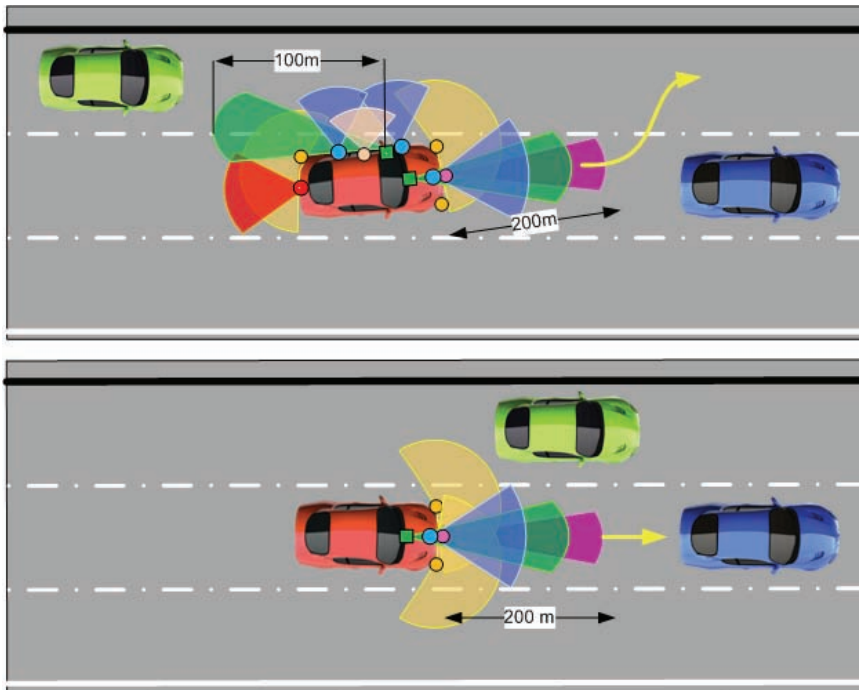


Fig. 4. Active sensors for lane change and overtaking (Top) and vehicle following with safe distance and safe speed according to ACC set point (Bottom).

These functionalities are required to match seven scenarios of straight, back, follow a vehicle, overtake a vehicle, lane change, turn corner, and emergency stop.

- Driving straight is the forward driving on a lane without leaving the lane.
- Driving back is the backward driving without leaving the lane.
- Following a vehicle is the driving accompanied a vehicle which driving in the front and keeping a proper relative distance and velocity.
- Overtaking a vehicle is cutting in a road by overtaking a vehicle which is driving in the front and there are left overtaking and right overtaking when driving on a road.
- Lane changing is the straight driving but changing the lane to another lane. Turning corner is to make a turn when driving on a road including intersection and forked road and there are left turning corner and right turning corner.
- Emergency stop is reducing the speed with a logical deceleration in order to eliminate an unexpected obstacle.

Figure 4 shows two sample scenario and active sensors for *Overtaking* (Top) and *ACC and vehicle following* (Bottom) schemes.

### 4. Driver assistance framework based on multi sensor data fusion

After implementing sensor assembly, a practical framework needs to be designed. We assume a driver assistance system as a vehicle–driver–environment interactive closed-loop system; moreover, we focus on not only the current situation of vehicle and driver, but also the future situation by predicting the potential collision probability distribution. In this framework, onboard sensors provide real-time information about drivers, traffic environment, and vehicles. How to configure these sensors is closely related to the application domain. For example, for multisensor ACC systems, radar and camera often suffice, but for pedestrian protection systems, an infrared sensor is essential to robust detection under various weather conditions.

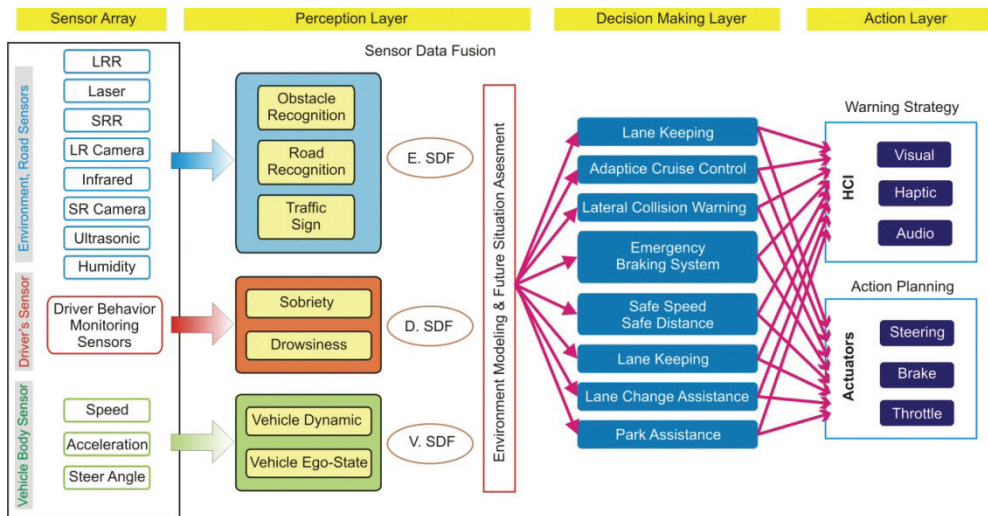


Fig. 5. Overall Architecture of Driver Assistance System based of sensor data fusion

As shown in figure 5 on the basis of various factors, an integrated interactive road safety analysis framework is introduced, where the system consists of the following modules:

**1-Sensor Layer:** including onboard environment monitoring sensor, inside driver monitoring sensor and vehicle body sensor e.g. vehicle ego-state and vehicle dynamic module. In General, the external sensors capture object appearance, range, and sound outside a vehicle, a vision based system monitor and record driver’s behavior, and the interior sensors collect vehicle state, such as speed, acceleration, and steering angle. Figure 6 and 7 describe characteristics of two main typical sensors required for this system, LASER and RADAR respectively (Schneider, 2005), (Langheim et al., 2001)

The main technical characteristics of a typical Long Range Radar sensor are:

- Frequency: 76-77 GHz
- Range: 1-150 m
- Search Area: 12°
- Speed measurement precision: < 0.2 km/h
- Angular Precision: < 0.3°



Fig. 6. Bosch 2<sup>nd</sup> Generation Long Range Radar Sensor Unit

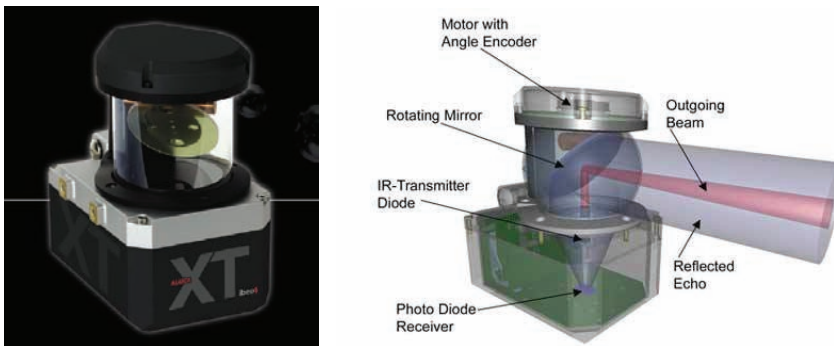


Fig. 7. ALASCA XT, the last Laserscanner from IBEO Automobile Sensor GmbH

The main characteristics of the current laser sensor are:

- Scan Frequency: 12.5 Hz or 25 Hz
- Horizontal Viewing angle: up to 240° field of view
- Effective Distance range (at reflectivity 5 %): up to 40 m
- Standard deviation of measured distance: +/- 5 cm
- Eye-safe laser (laser class 1)

**2- Perception Layer:** this layer aims to give a realistic representation of the environment to the applications; it can be seen as an intermediate layer between the sensorial sub-system and the system applications. It includes sensor fusion of previous layer in order to environment modeling, overall sensor fusion, and future situation assessment.

The role of the perception layer is to:

- Carry out a perception enhancement of the external scenario, even independently on any given application
- Describe the environment and the traffic scenario (obstacles and host vehicle dynamics, road geometry, etc.) in a formal way.
- Support the specific needs of the ADAS system in the reconstruction of the environment
- Act as an intermediate layer between sensors and functions, defined by the I/O protocols and interfaces.

The main functions of environment modeling and sensor fusion are to sense and recognize obstacles, lanes, pedestrians and traffic signs, etc.; where sensor information of different kinds is fused to model the environment. For the future situation assessment module, future road safety situations are assessed by combining traffic rules, vehicle dynamics, and environment prediction. Since the safety distance varies with the speed of a host vehicle, we adopt preview time rather than safety distance as the measurement of safety response. The safety response time is given as (Cheng et al., 2007):

$$T_0 = \frac{d_r + d_v + d_s}{v} \quad (2)$$

Where  $d_r$  is the distance required to respond to the nearest object due to driver response delay,  $d_v$  is the distance to slow down,  $d_s$  is the safety distance between the host vehicle and obstacles, and  $v$  is the speed of the host vehicle.

**3- Decision making agents layer:** as its names implies, perform decision carrying out one or several various driver assistant depending the status of previous layers. For example to make decision about "safe distance" in following a vehicle, it refers to the decision about determining a distance in meters that needs to be maintained to a vehicle ahead in order to avoid a collision in case the vehicle suddenly brakes. Safe-distance decision depends on a number of factors such as vehicle type, reaction time, braking distance, road surface and traffic conditions, driver awareness, etc. Over the years, different criteria have been suggested to determine a safe following distance for driving. Often cited is the two seconds rule illustrated in Figure 8 (Irion et al, 2006).

Decision-making agents have two functions: first to generate warning strategies for warning systems, such as route guide systems, haptic alert and warning display devices, and the secondly to plan the actions of the actuators to control the path of the host vehicle. As we know, the parameters of a vehicle model are affected by many factors and vary with the outside environment. Therefore, real-time learning algorithms using fuzzy logic have been applied to solving lateral and longitudinal control problems for a long time and have shown good performance (Cheng et al., 2007).

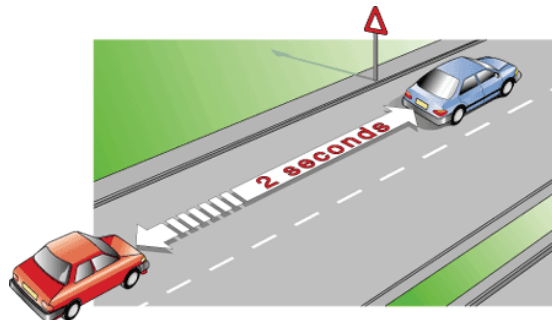


Fig. 8. Safe Distance decision by applying two seconds rule

**4-Action Layer:** Including two primary sections HMI and Actuators. This layer is responsible for handling the output from all the different applications in order to carry out appropriate warning and intervention strategies and then, to pass the information, suggestions or alarms to the driver through a given HMI. An in-vehicle HCI module



presents visual, sound and haptic warning information from the decision module and the path planning module to the driver. On the other hand action planning affects on actuators to prevent collisions and mitigate the dangerous. These are our final objective in order to approach to safe driving.

## 5. Sensor fusion processing

In this section we focus on data fusion and some common terminologies in literatures (Strobel & Coue, 2004) that take place in the perception layer. In general, sensor data fusion means processing steps which take the sensor data of several single sensors and combine the information from these sensors to a common and a certainly better result comparing with the outcome of each single sensor's processing could provide. This effect of getting better or more adequate results out of sensor data fusion compared to single sensor processing is called 'synergy' or 'synergetic effect'. Figure 9 shows the process of 'multi sensor processing', starting with the sensor data acquisition. Next, the sensors processing, divided into several tasks, as 'Calibration', 'Feature Extraction', 'Object Detection', etc., begins to analyze the sensors data and, in the end, serves the application with a more or less detailed model of the environment.

Before the sensors acquisition task is taking place, the sensors should be calibrated one by one in both time and space. That means, opposed to single sensor, a multi sensor system has to be synchronized, or the data acquired has to be time-aligned. Next, the sensors have to be 'space-aligned', meaning that displacements (in a mathematical sense) between the different (spatial) sensor coordinate systems have to be determined. Figure 9 gives a non-exhaustive exemplary overview on the tasks necessary for sensor data fusion processing.

In multi sensory, there are sequences for following the 'level of abstraction'. The calibration, time-alignment and reconstruction tasks, for example, take place at low level, while the 'objects detection' and the 'obstacles classification' are quite high level tasks within the processing. In some points of view, these classifications called early and late sensor data fusion, as well.

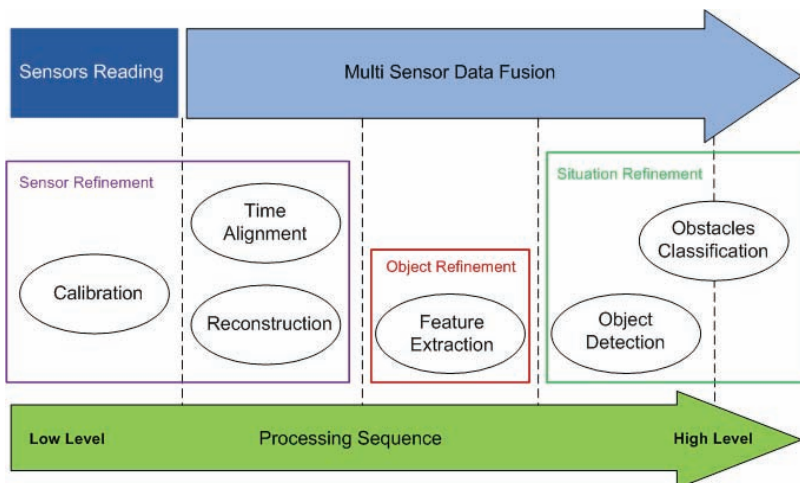


Fig. 9. Sensor Fusion Processing

From now, interest is focused on to sensor data fusion taking place exclusively within the perception layer. Generally, current trends in sensor data fusion inside the perception layer are classified in three sections of "Sensor refinement", "Object Refinement" and "Situation Refinement" (Strobel & Coue, 2004).

### 5.1 Sensor refinement

Sensor Refinement itself covers three tasks of "Calibration", "Time Alignment" and "Reconstruction of Sensor Models". All of them try to approach best observations from sensors. On the whole, the sensor refinement consists of the lowest, respectively the earliest processing steps. Hence, it builds the interface between sensor and sensor processing system, too.

Generally speaking, because of its natural complexity, the low level processing tasks have not been inspected as detailed as processing steps which are taking place within the object and situation refinement. Actually compared to the object and the situation refinement level, these low level tasks have to cope with a huge amount of raw sensor data. In addition, due to the quite different sensor characteristics, imaging, ranging, etc. the raw sensor data may appear completely different as well.

In principle, the problem is obvious; in recent approaches most of the tasks taking place within the sensor refinement has been examined and to some extent even solved for single sensor processing in a rather long term period. But then, the same results have been applied for multi sensor systems. So, even in a multi sensor system each sensor is treated as a single sensor. On the other hand, these tasks should be easier to cope with and could be solved more accurate while more information has been considered during processing. So, on the low level, we have got all the information needed to tackle them in this sense. In the following, we discuss sub-sections of sensor refinement block with some well chosen examples of interest that are employed to present this fact.

#### 5.1.1 Calibration and time alignment

Calibration refers to the process of determining the relation between the output of measuring instrument (here multi sensors) and the value of the input quantity or attribute according to a standar measurement. The problem with time-alignment is something different. In general, multi sensor systems are not synchronized, i.e. each sensor acquires its measuring at a different and specific time. Actually it would be difficult or useless to establish a multi sensor processing and the perception of the environment based on measurements from different sensors taken at different times; because they probably show contradictory states of the environment. Hence, a multi sensor system has to be time aligned. Within the 'SAVE-U' project ([www.cordis.europa.eu](http://www.cordis.europa.eu)) (Marchal et al., 2003), for example, the calibration of a multi sensor system consisting of a coloured camera (resolution: 640 x 480 pixels) and a far infrared sensor - at a resolution of 320 x 240 pixels - has been examined and implemented. Based on a manual selection of 20 corresponding points in two images, one from the coloured camera and another from the far infrared sensor, showing the same scenario the coordinate systems of both, the camera and the far infrared sensor are aligned; That means, a transformation which maps the coordinates of the camera to the coordinate system of the far infrared sensor, respectively vice versa, is derived from these corresponding points. On the calibration problem for single sensors and single sensor systems, e.g. for monocular stereo cameras and panoramic cameras, some efforts

have been taken already (Li et al, 2003) (Huang et al., 2006). Especially for stereo cameras even automatic calibration methods have already been proposed. But these existing methods still provide no reasonable solutions for the automotive sector.

### 5.1.2 Sensor modeling

Another group of problems to cope with on the sensor refinement level concerns the sensor models and a common representation and description of raw data. In automotive applications, the term 'sensor model' is understood as a model of sensor's data and how the sensor information matches the environment, including both model of the sensor's data and model of the vehicle's environment. This is mostly done individually for some types of sensors, e.g. for RADAR sensor probabilistic models are widely accepted, but not for a multi sensor system 'as a whole'. Until now, no common model was found which is able to represent all the different types of sensor data emerging in the field of preventive safety applications on these low levels e.g. radar, laser, images, etc.

### 5.2 Object refinement

The objective of the object refinement level is to interpret the multiple sensor 'observations', subsets of the sensors' raw and slightly preprocessed data, as individual objects or as parts of the environment observed (e. g. pedestrians, other vehicles, buildings, guard rails, trees, etc.). The input of this level consists of the output of the sensor refinement, called observations, which should be interpreted correctly. As Figure 10 illustrates 'observations' coming from several sensors (illustrated as ellipses in different colours) - represented in a common model as claimed above - are interpreted as objects (for example other vehicles or pedestrians) during the object refinement.

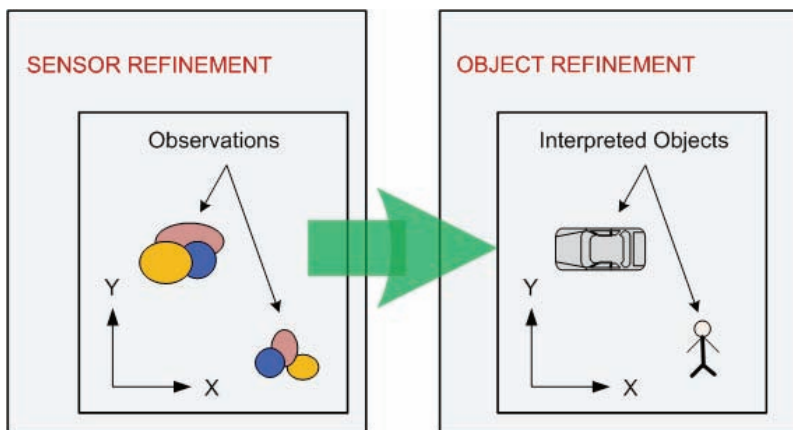


Fig. 10. Correlation between input-output of sensor refinement and object refinement

Hence, the output of the object refinement level generally consists of a list of objects with specific parameters or attributes (e. g. their position, their velocity, their height ...). The key functions of 'object refinement' are, among others, considered to be feature extraction, (single or multi) object tracking, 'observation-to-track' association and track maintenance (in the case of multi object tracking), as well as object classification (Strobel & Coue, 2004).

**5.2.1 Feature extraction**

Generally speaking, feature extraction could be regarded as the process of deducing ‘features’ or eye-catching properties, like segments of sensor data corresponding to the same object in the environment for example, from the sensor data. Concepts or techniques which are basically used for feature extraction could be segmentation, contour extraction, etc.

The following example shows feature extraction via segmentation in LIDAR sensor data:

In order to segmentation of the environment, first we have to perform *clustering*. The readings, e.g. from a LIDAR are subdivided into sets of neighbor points (Figure 11, left) by taking the proximity between each two consecutive points of the scan into account. A cluster is hence, a set of measures (points of the scan) close enough to each other, which due to their proximity; probably belong to the same object (Mendes et al., 2004). The segmentation criterion is based on two consecutive measurements of laser emitter,  $r_k, r_{k+1}$ ; we can conclude that they belong to the same segment if the distance between them fulfils the following expression:

$$R_{k,k+1} \leq L_0 + R_{\min} \cdot \frac{\tan \alpha \cdot \sqrt{2(1 - \cos \theta)}}{\cos\left(\frac{\theta}{2}\right) - \sin\left(\frac{\theta}{2}\right)} \tag{3}$$

Where  $R_{\min} = \min \{R_k, R_{k+1}\}$ ,  $R_{k+k+1} = |R_k - R_{k+1}|$  and  $\theta$  is angular resolution of Laser sensor.  $\alpha$  was introduced to reduce the dependency of segmentation with respect to the distance between the Laser Emitter and the object, and  $L_0$  to handle the longitudinal error of the sensor. If  $L_0 = 0$ , then  $\alpha$  represents the maximum absolute inclination that an object's face can have to be detected as a unique segment (Figure 11). Continuing the line fitting (Mendes et al., 2004) and considering the whole near lines as a single object we would be able to determine the object’s type, comparing with a model-based database. In that step using a technique called “Identity estimation”, relies on special classification algorithms that are used to recognize an object on the basis of some significant extracted features-the shape or

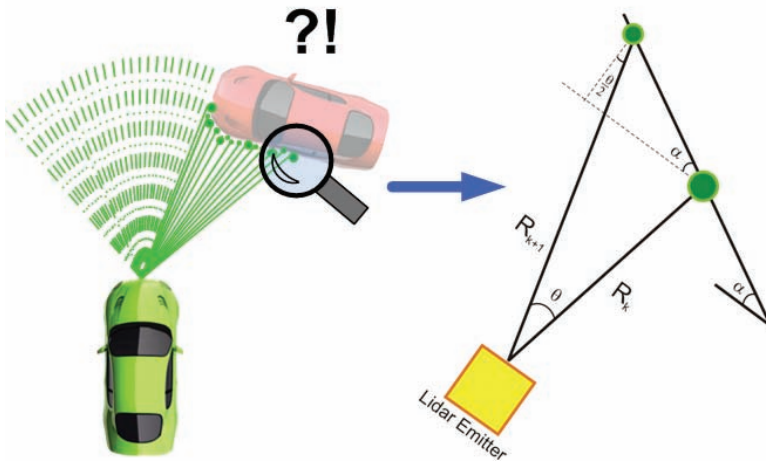


Fig. 11. Laser Scanning (left) and clustering method (right)

patterns- of various vehicles detected on a road, for example. In car safety applications, these kinds of algorithms can also be employed to construct a list of the objects (e.g., a tree, a motorcycle, a road sign, etc.) surrounding the host vehicle.

Figure 12 (left side) shows an example of observations, namely raw data (a list of points) coming from a scanning LIDAR at an angular resolution of  $0.5^\circ$  from a parking.

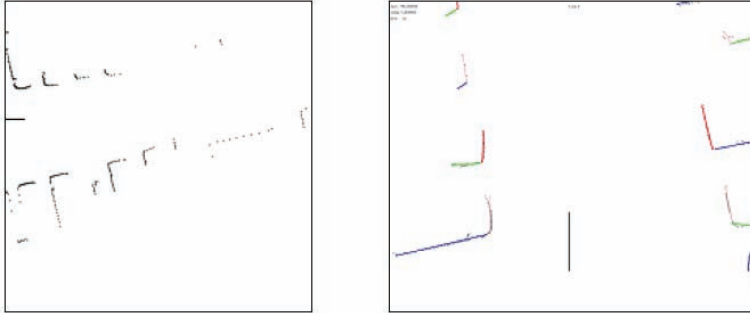


Fig. 12. LIDAR raw data (left side) and extracted features on the right side (provided by INRIA Rhone-Alpes)

In Figure 12 (on the right) we see the result from the first step of this feature extraction: the segmentation of raw LIDAR data in a set of segments (all marked in different colours). This is done using ‘Split and Merge’ (SaM) techniques as proposed in (Lee, 2001). During the ‘splitting’ step, the set of observations is recursively subdivided and a new segment is associated to each subdivision. Then, during the ‘merging’ step, segments lying closed to each other are merged.

Now, we have to decide which segments belong to the same object, a vehicle or a building close to the parking, and to estimate the position as well as the orientation of the object. This could be done using the Hough transform (Duda & Hart, 1972), (Tardos et al., 2002) under the hypothesis that the object could be represented as a rectangle. Then, a probability histogram on the object position and its orientation is constructed using the Hough transform, each mode of this probability distribution corresponding to one object (a vehicle or a building). But, as the vague term ‘feature’ already indicates, for feature extraction numerous procedures exist, depending only on the concrete definition of the term ‘feature’: features could be segments of corresponding sensor data, contours, prominent points, etc, for example. And at the end, as much as the meaning or the definition of the term ‘feature’ varies throughout the applications new or slightly adapted methods have been proposed to perform the feature extraction and to tackle the newly stated problem in these cases. In addition to object detection, we have to determine the absolute velocity of the moving object around the vehicle.

Normally the vehicle speed is measured by an encoder coupled with axle. But in sensor based system we can perform a vehicle speed estimation using two observations from a static object during a period of  $T$ . If we assume  $(r_1, \theta_1)$  and  $(r_2, \theta_2)$  as two observations like figure 13 left, then  $v_h$ , speed of host vehicle is:

$$v_h = \frac{r_1^2 + r_2^2 - 2r_1r_2\cos(\theta_1 - \theta_2)}{mT} \quad (4)$$

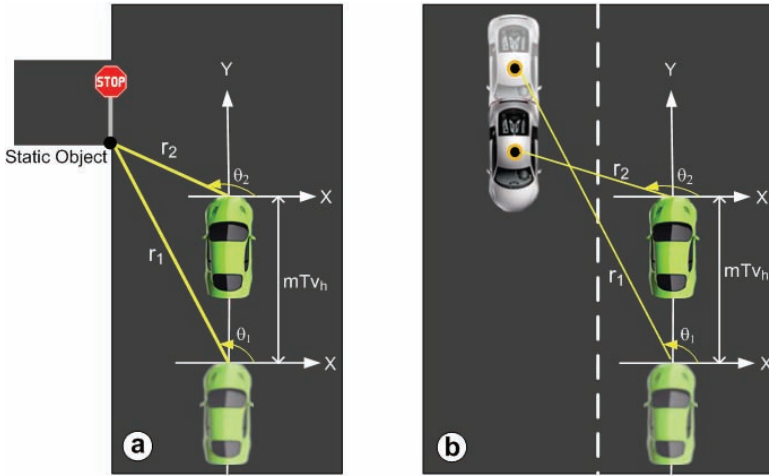


Fig. 13. Speed estimation approach

Where  $m$  is the number of the consecutive frames and is generally  $\geq 2$  for improving the accuracy of velocity. Here, we assumed that over a small interval of time  $mT$ , the driving direction of the host vehicle is consistent with the  $Y$ -axis in the Cartesian coordinates system  $XOY$ . Now after determining the speed of vehicle host, we can define the coordinate system of two segments at position  $P_0=(x_0, y_0)$  and  $P_1=(x_1, y_1)$ , while:

$$x_0 = -r_1 \cos \theta_1 \quad , \quad y_0 = r_1 \sin \theta_1 \tag{5}$$

$$x_1 = -r_2 \cos \theta_2 \quad , \quad y_1 = r_2 mTv_h + r_2 \sin \theta_2 \tag{6}$$

### 5.2.2 Sequential estimation and filtering problems

Estimation algorithms generally return the values of some quantitative entity parameters or attributes that are particularly significant for the application considered (R. Goodman et al., 1997). For instance, in car safety and driver assistance systems, estimations could be made for kinematic parameters (e.g., the position and the relative velocity) of the objects observed outside the host vehicle. To improve the estimation of the objects' attributes, or the objects' state (from objects detected around the vehicle), it is helpful to use several time-discrete estimations of the same objects determined at different times. The best would be if we could model the evolution of the objects' attributes in time. This is basically a filtering problem - called the 'sequential estimation problem'. One of the most famous filter used in the automotive sector to cope with these problems is called 'Kalman filter' (see (Kalman, 1960) and (Welch & Bishop, 2001) for details) and its extended version, the 'extended Kalman filter'. This probabilistic filter is based on a model of the evolution of the object's attributes in time, called the dynamic model, and on a sensor model (Strobel & Coue, 2004). Here, the concept is to determine the estimation of the objects' state at a given time  $k+1$  in two stages: a prediction stage uses the dynamic model and the estimation of the state at time  $k$  to determine a prediction of the state of the object at time  $k+1$ . Then an update stage confronts

this prediction with the new sensor observations at time  $k+1$  and updates the objects' attributes, accordingly.

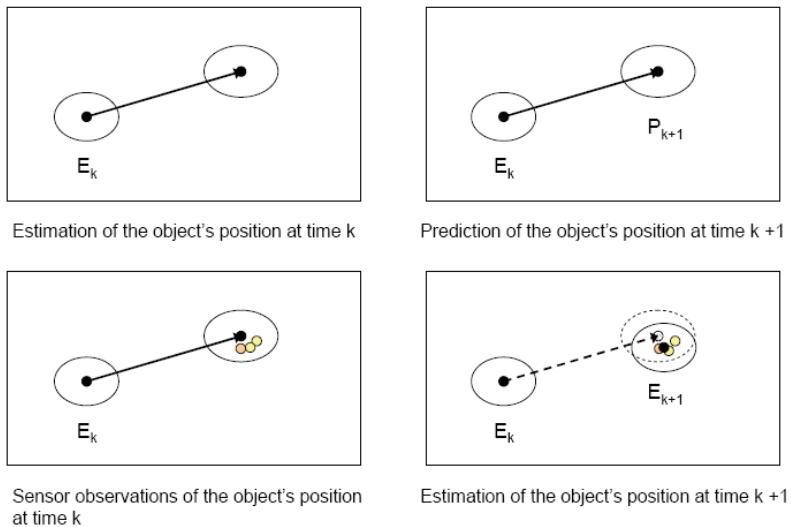


Fig. 14. An Example of Prediction scheme used in the 'Kalman filter'

Figure 14 shows the general prediction scheme of the 'Kalman filter' in the case of a two dimensional problem. Indeed both, the 'Kalman' and the 'extended Kalman filter' are simply efficient implementations of the general 'Bayesian filter' (compare (A.H. Jazwinski, 1970), for example)) - with strong constraints to linearity of the dynamic model and 'Gaussian noise' in the case of 'Kalman filter'. But more recent implementations of the 'Bayesian filter', namely the so called 'particle filters' (compare (A. Doucet et al., 2000), (S. Arulampalam et al., 2002)) become more and more popular in these fields, as they allow an efficient approximation of at least all 'Bayesian' filtering problems. On the other hand, 'Bayesian filters' are not the only technique used to come up with the 'sequential estimation problem'. For example, solutions based on the fuzzy logic are proposed as well (see Ggruyer, 1999)), but they use the same prediction scheme (see Figure 14).

## 6. Fuzzy fusion methodology

Here we propose a fuzzy logic approach as sensor fusion, just for ACC driver assistance system which is applicable on other driver assistance system in a similar manner. Then a new filtering method is performed -different from typical Kalman and Bayesian approaches- to reach more desirable results. The fuzzy logic system is carried out by *FuzzyTECH* simulator and the filtering has performed by MATLAB R2007b.

As you see in figure 4 (Bottom), to follow a vehicle by an ACC system we must keep safe distance by measuring the front vehicle distance to the host vehicle. On the other hand, four types and five sensors among total 16 sensors should be considered, each of which with different coverage area and may be infected by some environments noise, consequently with deferent measurements regarding the position of front vehicle. In such situation, we fuse all sensor data by fuzzifying them so determine a near real distance (Figure 15).

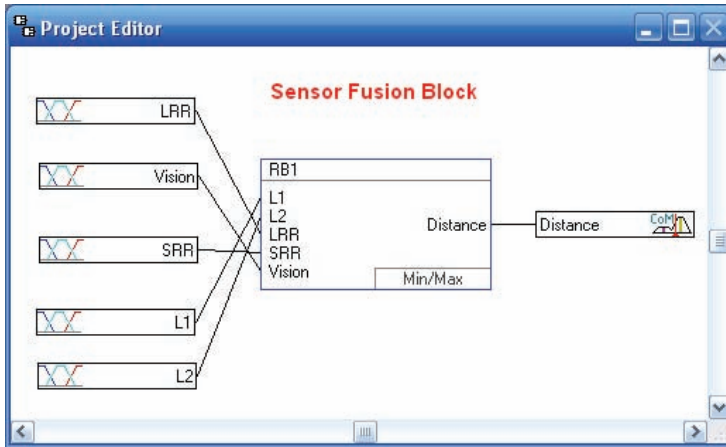


Fig. 15. Sensor fusion scheme using fuzzy logic method

**Structure of fuzzy logic system:**

The initial system structure is defined by input and output variables with their linguistic terms. Linguistic variables are components of fuzzy logic systems that “transform” real, crisp values, here from sensors, into linguistic values. The output also defined by some linguistic variables but finally should be defuzzified in to real output value (Figure 16).

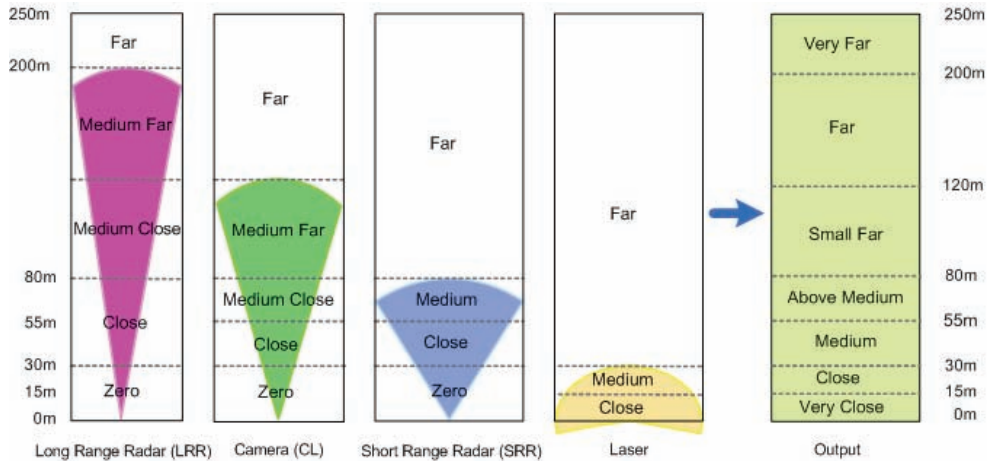


Fig. 16. Linguistic variable definition for sensors (inputs) and fused data (output)

In addition to the linguistic variables, a preliminary rule set must also be created for the system prototype. In this system, each of sensor measurements are matched to some membership functions and then a distance rule block is defined as output (Figure 17).

Here are two sample rules according to figure 16:

- IF (LRR=Medium Far AND Vision, SRR, L1, L2 = Far) THEN Distance = Far
- IF (LRR=Close AND Vision=Medium Close AND SRR=Medium AND L1, L2=Far) THEN Distance = Above Medium



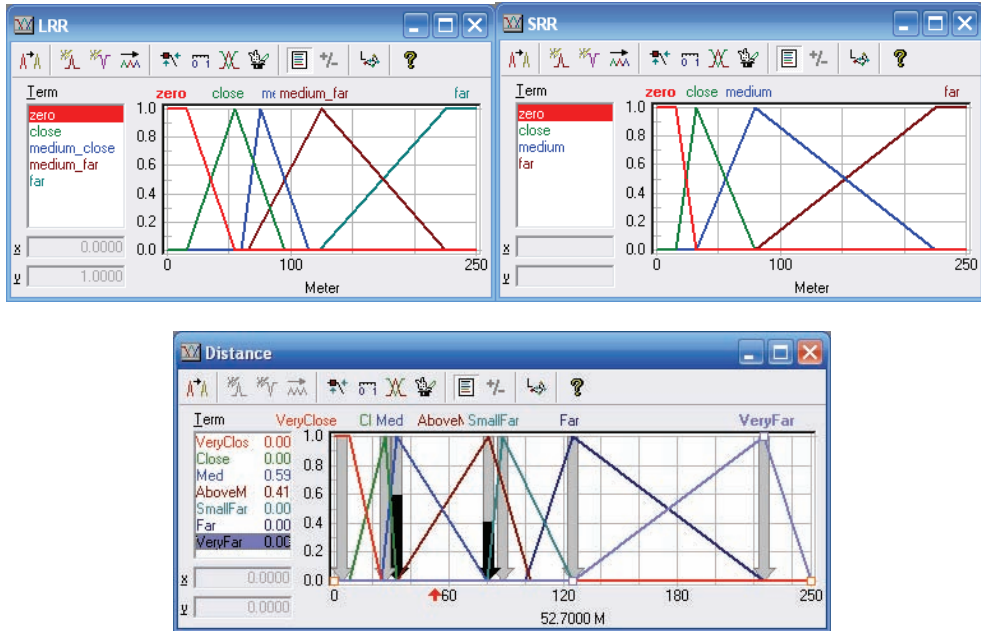


Fig. 17. Some of Membership functions for sensors (LRR, SRR) and output of sensor fusion (Distance)

Then a sample input data entered as the distance of the front vehicle with various speed and acceleration to examine the robustness of the system. The results was very interesting, after several modification and improvement of membership functions with Min-Max Aggregation operator of *FuzzyTECH* Simulator, finally a satisfactory following by the host vehicle is extracted. Let’s have a brief description on figure 18. In this graph, Blue curve is the real position of front vehicle with different acceleration rate at different times. The Red Curve is estimated distance according to multisensor data fusion using fuzzy approach. As can be seen from figure 18 in the area with multi detecting sensors, (e.g. in distances < 50m) we saw more fluctuations, But in far distance (distance > 100m) we saw better following, even with just a single LRR sensor! The reason is very clear; because the nature of different multi sensors (In lower distances), they feed the system a little bit different measurements and data, which will cause some fluctuations in overall distance detection. But in general, this is more reliable than a single sensor in far distance, despite a little fluctuation. The worst deviation found in this stage was about  $\pm 5.26$  meters.

### 7. Moving average filter

In Continuation, it is tried to improve red (fusion) curve by keeping both reliability of multi sensory in lower distance and reducing the fluctuations in multisensor areas. In this stage a filtering method is applied as it follows. A slight improvement in computational efficiency can be achieved if we perform the calculation of the mean in a *recursive* fashion. To illustrate this, consider the following development: Suppose that at any instant  $k$ , the average of the latest  $n$  samples of a data sequence,  $x_i$ , is given by:

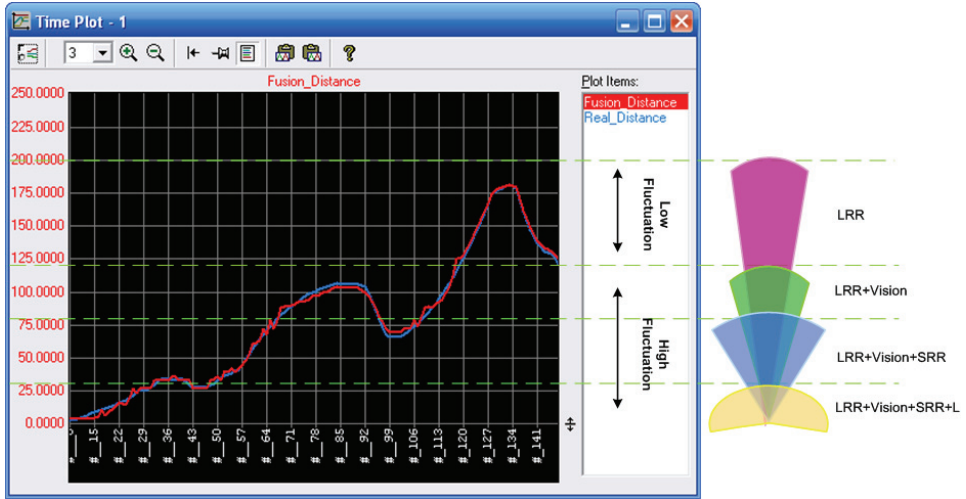


Fig. 18. Host vehicle following based on sensor data fusion

$$\bar{x}_k = \frac{1}{n} \sum_{i=k-n+1}^k x_i \tag{7}$$

Similarly, at the previous time instant,  $k-1$ , the average of the latest  $n$  samples is:

$$\bar{x}_{k-1} = \frac{1}{n} \sum_{i=k-n}^{k-1} x_i \tag{8}$$

$$\text{Therefore, } \bar{x}_k - \bar{x}_{k-1} = \frac{1}{n} \left[ \sum_{i=k-n+1}^k x_i - \sum_{i=k-n}^{k-1} x_i \right] = \frac{1}{n} [x_k - x_{k-n}] \tag{9}$$

which on rearrangement gives:

$$\bar{x}_k = \bar{x}_{k-1} + \frac{1}{n} [x_k - x_{k-n}] \tag{10}$$

This is known as a **moving average** filter because the average at  $k$ th instant is based on the most recent set of  $n$  values. In other words, at any instant, a **moving window** of  $n$  values is used to calculate the average of the data sequence (see Figure 19).

When used as a filter, the value of  $\bar{x}_k$  is taken as the filtered value of  $x_k$ . The expression is a recursive one, because the value of  $\bar{x}_k$  is calculated using its previous value,  $\bar{x}_{k-1}$ , as reference. This is always the case, regardless of the number of data points ( $n$ ) we consider, calculating the current filtered value requires the use of  $\bar{x}_{k-n}$ , i.e. the measurement  $n$  time-steps in the past. This means that:

1. the filtering cannot be initiated reliably until  $n$  measurements have been made, and

2. We need to store the value of  $\bar{x}_{k-n}$  which, depending on the way the algorithm is coded, may require up to  $n$  storage locations.

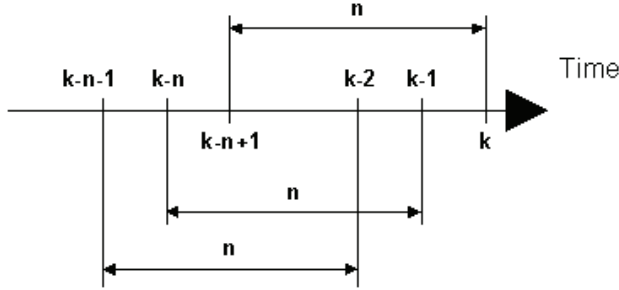


Fig. 19. Moving Window of  $n$  Data Points

Additionally, the technique places equal emphasis on all data points. Thus a value in the past will have the same influence as a more current measurement when calculating the filtered signal. This may be a desirable feature when the mean value of the measurement is almost constant, but not when the vehicle moves at various acceleration rates. These problems can however, be reduced by generating the filtered value in a slightly different manner. Actually, in dynamic systems, such as forward vehicle monitoring, the most current values tend to reflect better the state of the process. A filter that places more emphasis on the most recent data would therefore be more useful. Such a filter can be designed by following the procedure used in developing the moving average filter. As before, the starting point is the mean expressed as:

$$\bar{x}_k = \frac{1}{n} \sum_{i=k-n+1}^k x_i \quad (11)$$

But in this case, consider also the mean with one additional point

$$\bar{x}_{k+1} = \frac{1}{n+1} \sum_{i=k-n+1}^{k+1} x_i = \frac{1}{n+1} \left[ x_{k+1} + \sum_{i=k-n+1}^k x_i \right] \quad (12)$$

Since  $\sum_{i=k-n+1}^k x_i = n\bar{x}_k$ ,

therefore,

$$\bar{x}_{k+1} = \frac{1}{n+1} [x_{k+1} + n\bar{x}_k] = \left( \frac{1}{n+1} \right) x_{k+1} + \left( \frac{n}{n+1} \right) \bar{x}_k \quad (13)$$

By shifting the time index back one time-step, we obtain the corresponding expression for  $\bar{x}_k$  as:

$$\bar{x}_k = \left(\frac{1}{n+1}\right)x_k + \left(\frac{n}{n+1}\right)\bar{x}_{k-1} \quad (14)$$

To simplify the notation, let  $\alpha = \left(\frac{n}{n+1}\right)$ , which implies that  $(1-\alpha) = \left(\frac{1}{n+1}\right)$ . We can write the filter as:

$$\bar{x}_k = \alpha \bar{x}_{k-1} + (1-\alpha)x_k \quad (15)$$

This expression is Exponentially Weighted Moving Average Filter. When used as a filter, the value of  $\bar{x}_k$  is again taken as the filtered value of  $x_k$ . Notice that now, calculation of  $\bar{x}_k$  does not require storage of past values of  $x$ , and that only 1 addition, 1 subtraction, and 2 multiplication operations are required. The value of the **filter constant**,  $\alpha$ , dictates the **degree of filtering**, i.e. how strong the filtering action will be. Since  $n \geq 0$ , this means that  $0 \leq \alpha < 1$ . When a large number of points are being considered,  $\alpha \rightarrow 1$ , and  $\bar{x}_k \rightarrow \bar{x}_{k-1}$ . This means that the degree of filtering is so great that the measurement does not play a part in the calculation of the average! On the other extreme, if  $n \rightarrow 0$ , then  $\bar{x}_k \rightarrow x_k$  which means that virtually no filtering is being performed. The Exponentially Weighted Moving Average filter places more importance to more recent data by discounting older data in an exponential manner (hence the name). This characteristic can be illustrated simply by describing the current average value in terms of past data.

For example, since  $\bar{x}_k = \alpha \bar{x}_{k-1} + (1-\alpha)x_k$ , Then

$$\bar{x}_{k-1} = \alpha \bar{x}_{k-2} + (1-\alpha)x_{k-1} \quad (16)$$

Therefore,

$$\bar{x}_k = \alpha \bar{x}_{k-1} + (1-\alpha)x_k = \alpha[\alpha \bar{x}_{k-2} + (1-\alpha)x_{k-1}] + (1-\alpha)x_k \quad (17)$$

i.e.

$$\bar{x}_k = \alpha^2 \bar{x}_{k-2} + \alpha(1-\alpha)x_{k-1} + (1-\alpha)x_k \quad (18)$$

But

$$\bar{x}_{k-2} = \alpha \bar{x}_{k-3} + (1-\alpha)x_{k-2} \quad (19)$$

Therefore,

$$\begin{aligned} \bar{x}_k &= \alpha^2 [\alpha x_{k-3} + (1-\alpha)x_{k-2}] + \alpha(1-\alpha)x_{k-1} + (1-\alpha)x_k \\ &= \alpha^3 \bar{x}_{k-3} + \alpha^2(1-\alpha)x_{k-2} + \alpha(1-\alpha)x_{k-1} + (1-\alpha)x_k \end{aligned} \quad (20)$$

If we keep on expanding  $x$  terms on the right hand side, we will see that the contribution of older values of  $x_i$  are weighted by increasing powers of  $\alpha$ . Since  $\alpha$  is less than 1, the contribution of older values of  $x_i$  becomes progressively smaller. The weighting on  $x_i$  may be represented graphically by the following plot (Figure 20):

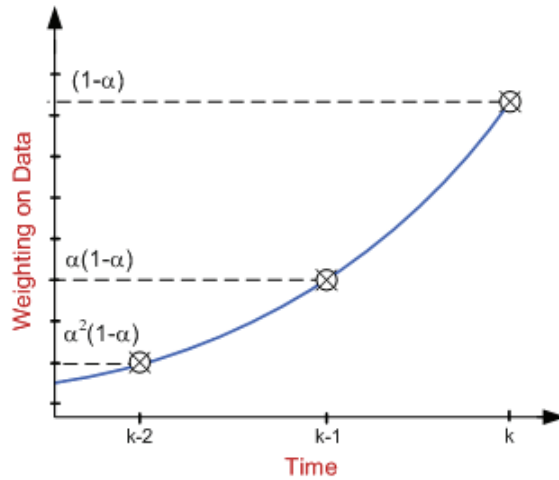


Fig. 20. Exponential weighting effect

What this means is that in calculating the filtered value, more emphasis is given to more recent measurements. By applying this approach to red curve of figure 18 and using MATLAB, we obtain more satisfactory results than before (very close to real distance) in overall fusion system (Figure 21)

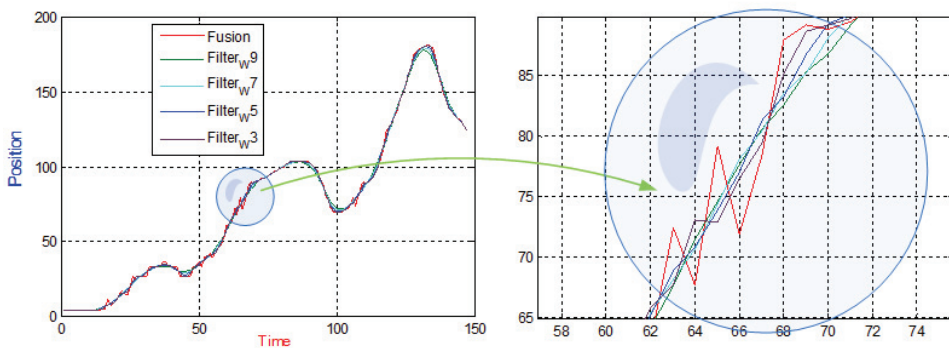


Fig. 21. Applying moving average filter with different windows sizes of 3,5,7,9

According to several experiments and as can be seen from figure 21, moving average filter with windows size of 5 met better following and more smoothing on our fusion graph. Figure 22 shows Fusion graph before (red) and after applying the filtering approach (green).

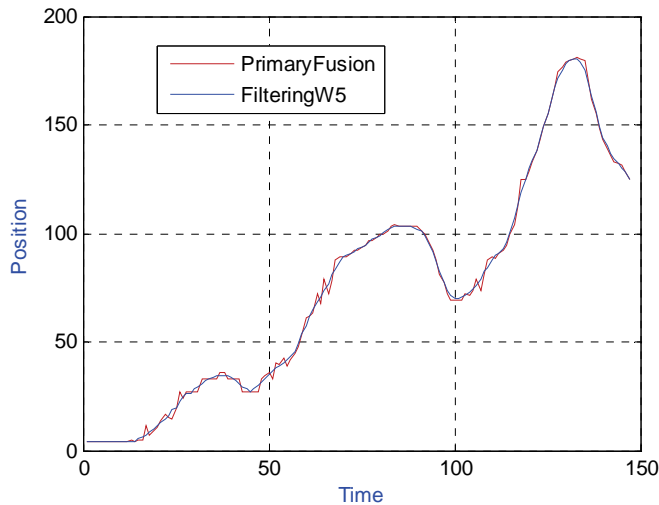


Fig. 22. Sensor Data Fusion before and after applying moving average filter

## 8. Future works

For future works, it could be continued with the last two sections of fusion processing (Figure 9) called situation refinement. During situation refinement it is tried to understand the *meaning* of the current situation around the vehicle: questions like “is this group of objects some meters in front of the vehicle a group of vulnerable pedestrians willing to cross the street?” or they are some bicyclists?; “Will the traffic light ahead turn to red in a few seconds?”. These could be done by considering a priori knowledge and additional environment information coming from digital maps, vehicle-to-vehicle as well as vehicle-to-infrastructure communication, etc.

And finally, *Trajectory Prediction*, could be done by taking the history of an object’s state parameters and attributes; for example its position and its respective speed in the past, and try to predict the object’s state in the near term future.

## 9. Conclusion

As the cars play an important and wide spreading role in transportation systems of all countries, the chance of having accident is also greatly increased. A number of cars, automobiles, bicycles and also pedestrians may involve in such accidents which sometimes cause miserable disasters. This fact attracts increasing numbers of researchers to work on driver assistance systems to be installed on modern automobiles. Such systems involve lots of considerations ranging from technical to cultural issues. One of interesting technical challenges on this field is the way that sensors communicate with each other and

the control unit and also the methods to collect, interpret and blend data come from these sensors, which is considered in the fields of sensor networks and sensor fusion techniques.

In this literature, the technologies of driver assistant systems and emerge of these systems are discussed. Also key features for an ADAS are introduced in addition with common problems and challenges that will appear in the implementation procedure. Furthermore a generalized framework for selecting proper sensors and mounting them on vehicle body in addition with a comprehensive methodology for employing sensors, initializing them and fusion of data from such sensors are introduced to be suitable for lots of applications and working environments.

Finally, a modern approach based on fuzzy logic and a filtering improvement is discussed to work as the multi-sensor data fusion core, which is implemented in FuzzyTech and MATLAB and the related experimental results are illustrated within the text.

## 10. References

- Amditis, A.; Floudas, N.; Polychronopoulos, A.; Bank, D.; Broek, B & Oechsle, F. (2006), Sensor Data Fusion for Lateral Safe Applications, *13th World Congress & Exhibition on Intelligent Transport Systems and Services*, 12 October 2006, London, UK.
- Cheng, H.; Zheng, N.; Zhang, X.; Qin, J. & Wetering, H. (2007), Interactive Road Situation Analysis for Driver Assistance and Safety Warning Systems: Framework and Algorithms, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 8, No. 1, March 2007
- Duda, R. O. & Hart, P. E. (1972), Use of the Hough transformation to detect lines and curves in pictures, *Communications of the ACM*, Vol. 15 , Issue 1, pp. 11-15, January 1972
- Hsieh, Y.; Lian, F. & Hsu, C. (2007), Optimal Multi-Sensor Selection for Driver Assistance Systems under Dynamical Driving Environment, *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference*, pp. 696-701, Sept. 30 - Oct. 3, 2007, Seattle, WA, USA
- Huang, F; Wei, S.K & Klette, R. (2006), Calibration of Line-based Panoramic Cameras, *Computational Imaging and Vision*, pages 55-84, Springer, Dordrecht, 2006
- Irion, J.; Flament, M.; Mäkinen, T.; Polychronopoulos, A. et al. Specifications and architecture for PREVENT safety functions, *Sixth framework program of European Commission*, Contract number FP6-507075, Version 1.0, March 2006
- Kaempchen, N. & Dietmayer, K. (2003), Data synchronization strategies for multi-sensor fusion, *10th World Congress on Intelligent Transportation Systems*, Issued No. 2250, November 2003, Madrid, Spain
- Kalman, R. E. (1960), a New Approach to Linear Filtering and Prediction Problems, *Transaction of ASME Journal of Basic Engineering*, 1960
- Langheim, J.; Buchanan, A.; Lages, U. & Wahl, M. (2001), CARSENSE - New environment sensing for advanced driver assistance systems, *2001 IEEE Intelligent Vehicle Symposium (IV 2001)*, pp. 89-94, May 13-17, 2001, Tokyo, Japan
- Langheim, J.; Henrio, J. & Liabeuf, B. (1999), ACC Radar System « Autocruise » with 77 GHz MMIC Radar, *ATA*, Florence, 1999.

- Lee, K. J. (2001), Reactive navigation for an outdoor autonomous Vehicle, *Technical Report, University of Sidney*, December 2001, Australia
- Leohold, J. & Schmidt, C. (2004), Communication Requirements of Future Driver Assistance Systems in Automobiles, *Proceedings of 2004 IEEE International Workshop on Factory Communication Systems*, pp. 167-174, Sept. 2004, Germany
- Li, F.; Zang, Q.; Klette, R. (2003), Accuracy Improvement in Camera Calibration, *International Conference of Image and Vision Computing*, pages 338--342, Massey University, 2003, New Zealand
- Luo, Ren C.; Chou, Y. & Chen O. (2007), Multisensor Fusion and Integration: Algorithms, Applications, and Future Research Directions, *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, pp. 1986-1991, August 5 - 8, 2007, Harbin, China.
- Macci, D.; Boni, A.; Cecco, M. & Petri, D. (2008), Multisensor Data Fusion, *IEEE Instrumentation & Measurement Magazine*, Part 14, pp. 24-33, June 2008
- Marchal, P.; Gavrilă, D.; Latellier, L.; Meinecke, M.; Morris, R. & Töns, M. (2003), SAVE-U: An innovative sensor platform for Vulnerable Road User protection, *Intelligent Transport Systems and Services*, 16-20 November 2003, Madrid, Spain
- Mammar, S.; Glaser, S. & Netto, M. (2006), Time to Line Crossing for Lane Departure Avoidance: A Theoretical Study and an Experimental Setting, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 7, No. 2, pp. 226-241, June 2006
- Mendes, A.; Bento, L. & Nunes, U. (2004), Multi-target Detection and Tracking with a Laserscanner, *2004 IEEE Intelligent Vehicles Symposium*, pp. 796-801, June 14-17, 2004, Parma, Italy
- Rezaei Ghahroudi, M. & Fasih, A. (2007), A Hybrid Method in Driver and Multisensor Data Fusion, Using a Fuzzy Logic Supervisor for Vehicle Intelligence, *2007 International Conference on Sensor Technologies and Applications (SENSORCOMM2007)*, pp. 393-398, October 14-20, 2007, Valencia, Spain
- Schneider, M. (2005), Automotive Radar - Status and Trends, *Proceedings of the German Microwave Conference (GeMiC 2005)*, Hildesheim, pp. 144-147, April 5-7, 2005, Ulm, Germany
- Strobel, T.; & Coue, C. (2004), Compendium on Sensor Data Fusions, *Profusion, Sixth Framework Programme of PreVENT Project*, Germany, 2004
- Tardos, J. D.; Neira, J.; Newman, P. M. & Leonard, J. J. (2002), Robust mapping and localization in indoor environments using sonar data, *The International Journal of Robotics Research*, Vol. 21, No. 4, 311-330, April 2002
- Tatschke, T.; Park, S.; Amditis, A.; Polychronopoulos, A.; Scheunert, O. & Aycard, O. (2006), ProFusion2 - Towards a modular, robust and reliable fusion architecture for automotive environment perception, *Jürgen Valldorf, Wolfgang Gessner (Ed.): Advanced Microsystems for Automotive Applications 2006 (AMAA)*, pp. 451-469. Springer, Berlin, Germany
- Trivedi, M. M.; Gandhi, T. & McCall, J. (2007), Looking-In and Looking-Out of a Vehicle: Computer-Vision-Based Enhanced Vehicle Safety, *IEEE Transactions on Intelligent Transportation Systems*, VOL. 8, NO. 1, pp. 108-120, March 2007



- Wahl, M. & Doree, G. (2000), A CAN application layer for an experimental real time obstacle detection study, *Proceedings of 2000 IEEE intelligent Transportation Systems*, pp. 276-281, October 1-3, 2000, Dearborn (MI), USA
- Welch, G. & Bishop, G. (2001), an Introduction to Kalman Filter, *Course 8 of SIGGRAPH 2001*, By ACM, Inc., August 21, 2001, Los Angeles, CA, USA

# Trajectory Generation and Object Tracking of Mobile Robot Using Multiple Image Fusion

TaeSeok Jin<sup>1</sup> and Hideki Hashimoto<sup>2</sup>

<sup>1</sup>*Dept. of Mechatronics Eng., DongSeo University,*

<sup>2</sup>*Institute of Industrial Science, the University of Tokyo*

<sup>1</sup>*Korea*

<sup>2</sup>*Japan*

## 1. Introduction

Detection of moving objects has been utilized in industrial robotic systems, for example, in the recognition and monitoring of unmanned systems that also require compression of moving images [1], [2], [3], [4]. Trajectory prediction of moving objects is required for a mobile manipulator that aims at the control and observation of motion information such as object position, velocity, and acceleration. Prediction and estimation algorithms have generally been required for industrial robots. For a simple example, in a pick-and-place operation with a manipulator, the precise motion estimation of the object on the conveyor belt is a critical factor in stable grasping. A well-structured environment, such as the moving-jig that carries the object on the conveyor belt and stops when the manipulator grasps the object, might obviate the motion estimation requirement. However, a well-structured environment limits the flexibility of the production system, requires skillful designers for the jig, and incurs a high maintenance expense; eventually it will disappear from automated production lines. To overcome these problems, to tracking a moving object stably without stopping the motion, the trajectory prediction of the moving object on the conveyor belt is necessary [5]. The manipulator control system needs to estimate the most accurate position, velocity, and acceleration at any instance to capture the moving object safely without collision and to pick up the object stably without slippage. When the motion trajectory is not highly random and continuous, it can be modeled analytically to predict the near-future values based on previously measured data [6]. However, this kind of approach requires significant computational time for high-degrees-of-freedom motion, and its computational complexity increases rapidly when there are many modeling errors. In addition, performance is highly sensitive to the change of the environment. Those state-of-the-art techniques perform efficiently to trace the movement of one or two moving objects but the operational efficiency decreases dramatically when tracking the movement of many moving objects because systems implementing multiple hypotheses and multiple targets suffer from a combinatorial explosion, rendering those approaches computationally very expensive for real-time object tracking [7].

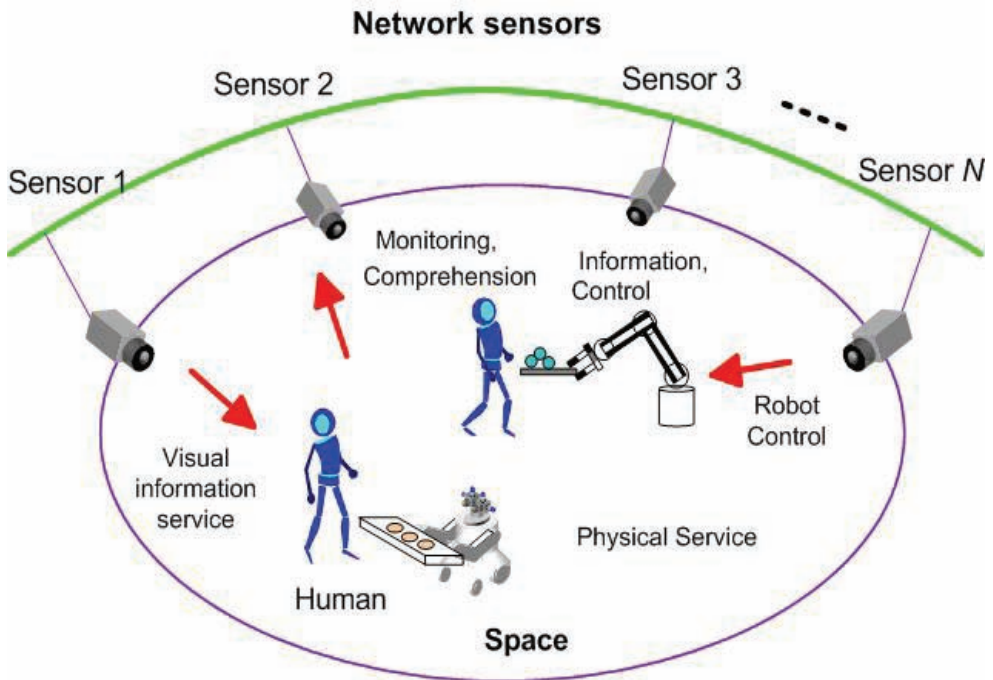


Fig. 1. Intelligent environment by distributed cameras

## 2. Vision systems in robotic space

### 2.1 Structure of robotic space

Fig. 2 shows the system configuration of distributed cameras in Intelligent Space. Since many autonomous cameras are distributed, this system is autonomous distributed system and has robustness and flexibility. Tracking and position estimation of objects is characterized as the basic function of each camera. Each camera must perform the basic function independently at least because over cooperation in basic level between cameras loses the robustness of autonomous distributed system. On the other hand, cooperation between many cameras is needed for accurate position estimation, control of the human following robot [8], guiding robots beyond the monitoring area of one camera [9], and so on. These are advanced functions of this system. This distributed camera system of Intelligent Space is separated into two parts as shown in Fig. 2. This paper will focus on the tracking of multiple objects in the basic function. Each camera has to perform the basic function independently of condition of other cameras, because of keeping the robustness and the flexibility of the system. On the other hand, cooperation between many cameras is needed for accurate position estimation, control of mobile robots to supporting human [10],[11], guiding robots beyond the monitoring area of one camera[12], and so on. These are advanced functions of this system.

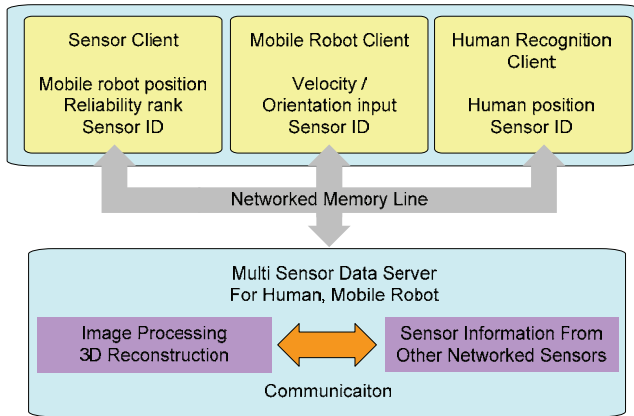


Fig. 2. Configuration of distributed camera system

## 2.2 Previous research for tracking

Neural networks can be classified into two categories: supervised learning- and unsupervised learning methods. In most of the previous research, the supervised learning method was adopted to overcome the nonlinear properties [10],[12]. Since the supervised learning method requires the relation between input and output [9] at all times, it is not suitable for real-time trajectory estimation for which the input-output relation cannot be obtained instantaneously in the unstructured environment. Therefore, in this study, SOM (Self Organizing Map), that is, a type of unsupervised learning method, was selected to estimate the highly nonlinear trajectory that cannot be properly predicted by the Particle filter. Also, SOM is a data-sorting algorithm, which is necessary for real-time image processing since there is so much data to be processed. Among the most popular data-sorting algorithms, VQ (Vector Quantization), SOM, and LVQ (Learning Vector Quantization), SOM is selected to sort the data in this approach since it is capable of unsupervised learning. Since VQ is limited to the very special case of a zero neighborhood and LVQ requires preliminary information for classes, neither of them is suitable for the unsupervised learning of the moving trajectory. Fig. 3 shows the estimation and tracking system for this research. The input for the dynamic model comes from either the Particle filter or SOM according to the following decision equation:

$$P_{predicted\ value} = k \cdot Particle\ Filter_{out} + (1 - k) \cdot SOM_{out}$$

where  $k=1$  for  $error \leq threshold$  and  $k=0$  for  $error > threshold$ .

The threshold value is empirically determined based on the size of the estimated position error.

## 3. Processing flow

### 3.1 Extraction of object

Classifying the moving-object pattern in the dynamically changing unstructured environment has not yet been tackled successfully [13]. Therefore, in this research, the

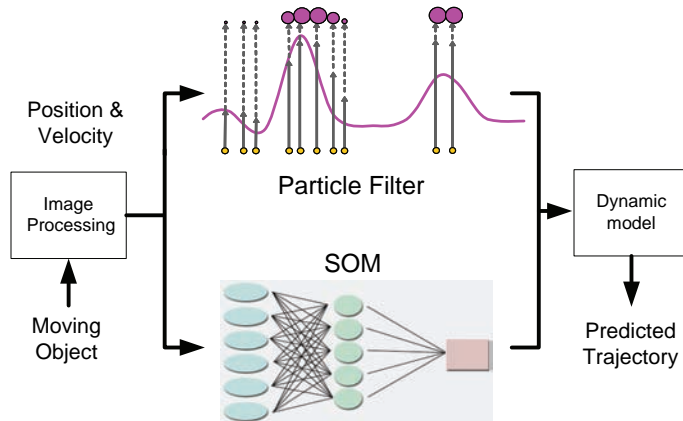


Fig. 3. Estimation and tracking system of robotic space

camera was fixed on a stable platform in order to capture static environment images. To estimate the states of the motion characteristics, the trajectory of the moving object was pre-recorded and analyzed. Fig. 4(a) and Fig. (b) represent the object images at (t-1) instance and (t) instance, respectively.

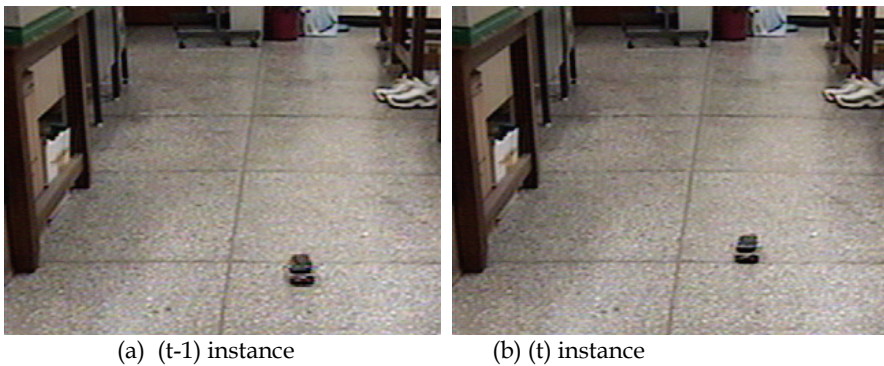


Fig. 4. Detected image of moving object at each instance

As recognized in the images, most parts of the CCD image correspond to the background. After eliminating the background, the difference between the two consecutive image frames can be obtained to estimate the moving-object motion. To compute the difference, either the absolute values of the two image frames or the assigned values can be used. The difference method is popular in image pre-processing for extracting desired information from the whole image frame, which can be expressed as

$$\text{Output}(x, y) = \text{Image1}(x, y) - \text{Image2}(x, y)$$

The difference image between Fig. 4(a) and Fig. 4(b) is represented in Fig. 5. When the difference image for the whole time interval can be obtained, the trajectory of the moving object can be calculated precisely.



Fig. 5. Difference image between (t) and (t-1) instance images

### 3.2 Target regions encoded in a state vector using particle filter

Particle filtering provides a robust tracking framework, as it models uncertainty. Particle filters are very flexible in that they do not require any assumptions about the probability distributions of data. In order to track moving objects (e.g. pedestrians) in video sequences, a classical particle filter continuously looks throughout the  $2D$ -image space to determine which image regions belong to which moving objects (target regions). For that a moving region can be encoded in a state vector. In the tracking problem the object identity must be maintained throughout the video sequences. The image features used therefore can involve low-level or high-level approaches (such as the colored-based image features, a subspace image decomposition or appearance models) to build a state vector. A target region over the  $2D$ -image space can be represented for instance as follows:

$$\mathbf{r} = \{l, s, m, \gamma\}$$

where  $l$  is the location of the region,  $s$  is the region size,  $m$  is its motion and  $\gamma$  is its direction. In the standard formulation of the particle filter algorithm, the location  $l$ , of the hypothesis, is fixed in the prediction stage using only the previous approximation of the state density. Moreover, the importance of using an adaptive-target model to tackle the problems such as the occlusions and large-scale changes has been largely recognized. For example, the update of the target model can be implemented by the equation

$$\bar{\mathbf{r}}_t = (1 - \lambda)\bar{\mathbf{r}}_{t-1} + \lambda E[\mathbf{r}_t]$$

where  $\lambda$  weights the contribution of the mean state to the target region. So, we update the target model during slowly changing image observations.

## 4. Tracking moving objects

### 4.1 State-space over the top-view plan

In a practical particle filter [5],[6] implementation, the prediction density is obtained by applying a dynamic model to the output of the previous time-step. This is appropriate when the hypothesis set approximation of the state density is accurate. But the random nature of the motion model induces some non-zero probability everywhere in state-space that the object is present at that point. The tracking error can be reduced by increasing the number of hypotheses (particles) with considerable influence on the computational complexity of the algorithm. However in the case of tracking pedestrians we propose to use the top-view information to refine the predictions and reduce the state-space, which permits an efficient discrete representation. In this top-view plan the displacements become Euclidean distances. The prediction can be defined according to the physical limitations of the pedestrians and their kinematics. In this paper we use a simpler dynamic model, where the actions of the pedestrians are modeled by incorporating internal (or personal) factors only. The displacements  $M_{topview}^t$  follows the expression

$$M_{topview}^t = A(\gamma_{topview})M_{topview}^{t-1} + N$$

where  $A(\cdot)$  is the rotation matrix,  $\gamma_{topview}$  is the rotation angle defined over top-view plan and follows a Gaussian function  $g(\gamma_{topview}; \sigma_\gamma)$ , and  $N$  is a stochastic component. This model proposes an anisotropic propagation of  $M$ : the highest probability is obtained by preserving the same direction. The evolution of a sample set is calculated by propagating each sample according to the dynamic model. So, that procedure generates the hypotheses.

### 4.2 Estimation of region size

The size of the search region represents a critical point. In our case, we use the *a-priori* information about the target object (the pedestrian) to solve this tedious problem. We assume an averaged height of people equal to 160 cm, ignoring the error introduced by this approximation. That means, we can estimate the region size  $s$  of the hypothetical bounding box containing the region of interest  $r = \{l, s, m, \gamma\}$  by projecting the hypothetical positions from top-view plan in Fig. 6.

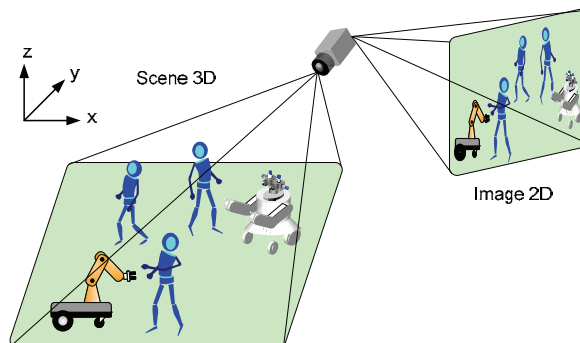


Fig. 6. Approximation of Top-view plan by image plan with a monocular camera

A camera calibration step is necessary to verify the hypotheses by projecting the bounding boxes. So this automatic scale selection is a useful tool to distinguish regions. In this way for each visual tracker we can perform a realistic partitioning (bounding boxes) with consequent reduction in the computational cost. The distortion model of the camera's lenses has not been incorporated in this article. Under this approach, the processing time is dependent on the region size.

### 4.3 Object model update

In multi-motion tracking, the hypotheses are verified at each time step by incorporating the new observations (images). A well known measure of association (strength) of the relationship between two images is the normalized correlation.

$$dc_{i,j} = \text{corr}_{\text{nor}}(\text{target}_i ; \text{hypothesis}_{i,j})$$

where  $i$  : target region, and  $j$  : an hypothesis of the target region  $i$ . The observation of each hypothesis is weighted by a Gaussian function with variance  $\sigma$ .

$$h^{(i;j)} = \frac{1}{\sqrt{2\pi}\sigma_{dc}} e^{-\frac{(1-dc_{i,j})^2}{2\sigma_{dc}^2}}$$

where  $h^{(i;j)}$  is the observation probability of the hypothesis  $j$  tracking the target  $i$ . The obvious drawback of this technique is the choice of the region size (defined in previous section) that will have a great impact on the results. Larger region sizes are less plagued by noise effects.

## 5. Experiments

To compare the tracking performance of a mobile robot using the algorithms of the Particle filter and SOM, experiments of capturing a micro mouse with random motion by the mobile robot were performed. As can be recognized from Fig. 7, SOM based Particle Filter provided better performance to the mobile robot in capturing the random motion object than the other algorithms. As shown in Fig. 7, the mobile robot with SOM based Particle Filter has a smooth curve to capture the moving object. As the result, the capturing time for the moving object is the shortest with SOM based Particle Filter. Finally, as an application experiments were performed to show the tracking and capturing a mobile object in robotic space.

Fig. 8 shows the experimental results for tracking a moving object that is an 8x6[cm] red-colored mouse and has two wheels with random velocities in the range of 25-35[cm/sec]. First, mobile robot detects the moving object using an active camera. When a moving object is detected within view, robot tracks it following the proposed method.



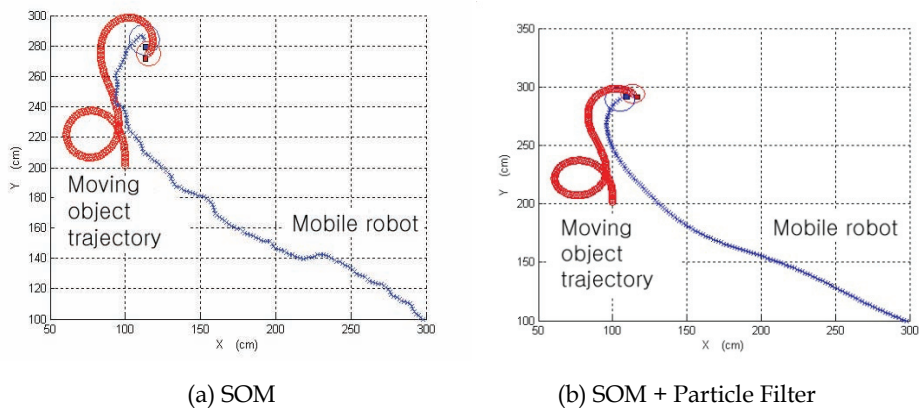


Fig. 7. Tracking trajectory by SOM and SOM based particle filter

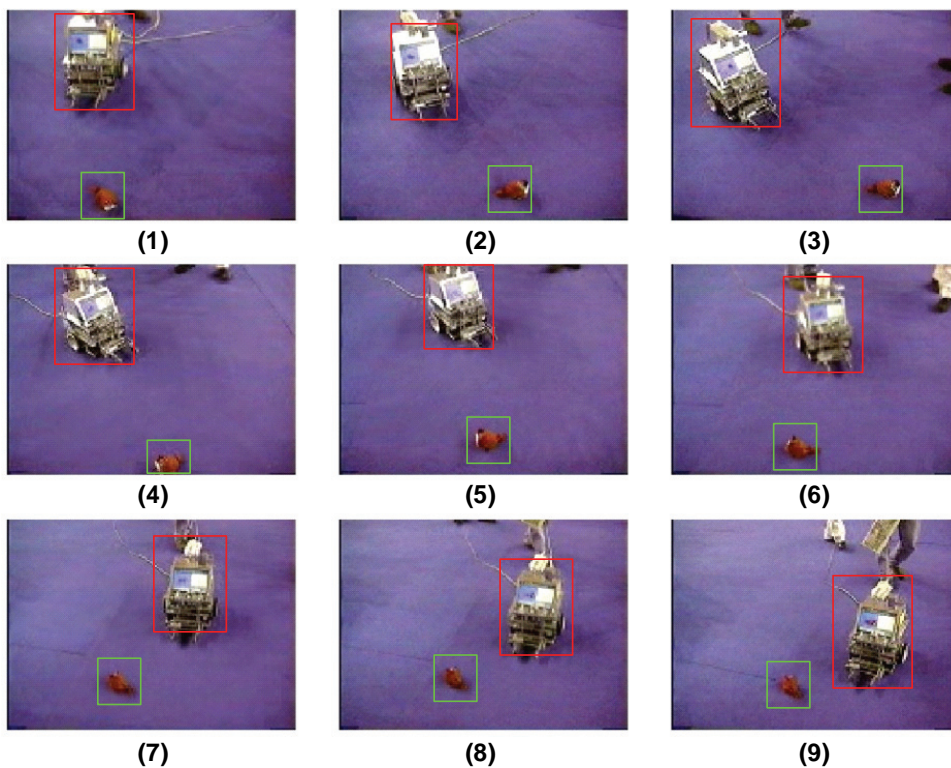


Fig. 8. Experimental results for tracking a moving object

## 6. Conclusions

In this paper, the proposed tracking method adds an adaptive appearance model based on color distributions to particle filtering. The color-based tracker can efficiently and successfully handle non-rigid and fast moving objects under different appearance changes. Moreover, as multiple hypotheses are processed, objects can be tracked well in cases of occlusion or clutter. This research proposes estimation and tracking scheme for a moving object using images captured by multi cameras. In the scheme, the state estimator has two algorithms: the particle filter that estimates the states for the linear approximated region, and SOM for the nonlinear region. The decision for the switchover is made based on the size of the position estimation error that becomes low enough for the linear region and becomes large enough for the nonlinear region. The effectiveness and superiority of the proposed algorithm was verified through experimental data and comparison. The adaptability of the algorithm was also observed during the experiments. For the sake of simplicity, this research was limited to the environment of a fixed-camera view. However, this can be expanded to the moving camera environment, where the input data might suffer from higher noises and uncertainties. As future research, selection of a precise learning pattern for SOM in order to improve the estimation accuracy and the recognition ratio, and development of an illumination robust image processing algorithm, remain.

## 7. Acknowledgments

This work is resulted from the project of Human Resources Development for Industrial Demand financially supported by the Ministry of Education and Human Resources Development (MOE).

## 8. References

- [1] Senior, A.: Tracking with Probabilistic Appearance Models. In Proc. ECCV workshop on Performance Evaluation of Tracking and Surveillance Systems (2002) 48-55.
- [2] Bierlaire, M., Antonini, G., Weber, M.: Behavioural Dynamics for Pedestrians," in K. Axhausen (Ed.), Moving through nets: the physical and social dimensions of travel, Elsevier (2003) 1-18.
- [3] Nummiaro, K., Koller-Meier, E., Van Gool, L.J.: Object Tracking with an Adaptive Color-Based Particle Filter. DAGM-Symposium Pattern Recognition (2002) 353-360.
- [4] P. K. Allen, A. Tmcenko, B. Yoshimi, and P. Michelman, "Trajectory filtering and prediction for automated tracking and grasping of a moving object," *IEEE International Conference on Robotics and Automation* (1992) 1850-1856.
- [5] Yi Ma, J. Kosecka, and S. S. Sastry, "Vision guided navigation for a nonholonomic mobile robot," *IEEE Transaction on Robotics and Automation*, vol. 15, no. 3, pp. 521-536, 1999.
- [6] Choo, K., Fleet, D.J.: People tracking using hybrid Monte Carlo filtering. In Proc. Int. Conf. Computer Vision, vol. II (2001) 321-328.
- [7] Anderson, B., Moore, J.: Optimal Filtering. Prentice-Hall, Englewood Cliffs (1979).
- [8] Kitagawa, G.: Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. *Journal of Computational and Graphical Statistics*, Vol. 5 (1996) 1-25.
- [9] Yi-Yuan Chen and Kuu-young Young, "An intelligent radar predictor for high-speed moving-target tracking," *TENCON '02. Proceedings. IEEE Region 10 Conference on*

- Computers, Communications, Control and Power Engineering*, vol. 3, pp. 1638 -1641, 2002.
- [10] J. M. Roberts, D. J. Mills, D. Charnley, and C. J. Harris, "Improved Kalman filter initialization using neuro-fuzzy estimation," *Int'l. Conf. on Artificial Neural Networks*, pp. 329-334, 1995.
- [11] Norlund, P., Eklundh, J.O.: Towards a Seeing Agent. Proc. of First Int. Workshop on Cooperative Distributed Vision (1997) 93-120.
- [12] Atsushi, N., Hirokazu, K., Shinsaku, H., Seiji, I.: Tracking Multiple People using Distributed Vision Systems. Proc. of the 2002 IEEE Int. Conf. on Robotics & Automation (2002) 2974-2981.
- [13] Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: finder: Real-Time Tracking of the Human Body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19 (1997) 780-785.
- [14] Gardner, W.F., Lawton, D.T.: Interactive model based vehicle tracking. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol.18 (1996) 1115-1121.
- [15] Swain, M.J., Ballard, D.H.: Color indexing. *Int. J. of Computer Vision*, Vol.7, No.1, (1991)11-32.

# Multisensory Data Fusion for Ubiquitous Robotics Services

Ren C. Luo<sup>1</sup> and Ogst Chen<sup>2</sup>

<sup>1</sup>*Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan*

<sup>2</sup>*Department of Electrical Engineering, National Chung Cheng University, Chia-Yi, Taiwan*

## 1. Introduction

Real time human tracking in indoor environment is highly demand and important to many applications for ubiquitous robotics services. There are several technologies for human tracking in the indoor environment, such as vision, pressure, infrared, and ultrasonic have been proposed. However, some technologies are costly, or have many restrictions when they are applied. For example, implant several cameras around indoor environment through real time image processing for tracking people is frequently discussed. However, the computer vision method does not function in dark situation and the privacy issue also generated.

Recent advances in radio frequency, microprocessor and sensor technologies enable the wireless sensor network (WSN) system. Wireless sensor network system used for target tracking can be found (Liu, J. et al., 2007); (Hua Li et al., 2008); (Djuric, P. M. et al., 2008). In general, the smart home aims to provide appropriate intelligent services to assist the resident's living. Autonomous and multi-functional robots plays important role in the smart home. However, if robots depend on their own sensors, the applications will be limited. Sensor network system uses multi sensor combined with microprocessor and radio transmission into a device and deploy in the monitoring environment. With the ambient information provided by sensor network based ubiquitous robotics services system, robots can serve people more quickly and accurately.

There are a lot of researches discuss sensor network theories or applications. However, these researches (Yoon Gu Kim et al., 2006); (Taehong Kim et al., 2008) deal with implement sensor network in indoor environment just implant additional sensor nodes on the desks or ceiling. We propose a solution on implementing the sensor network system for ubiquitous robotics services in the indoor environment.

Since most building have implant traditional smoke detector on ceiling for fire detection. We try to implement sensor network system in indoor environment and modify these smoke sensors. We developed a prototype of wireless multi-functional detector; the sensor consists of pyroelectric infrared (PIR) sensor for detecting human radiation, temperature sensor for measuring the ambient temperature and smoke sensor to replace the original smoke detector function on ceiling. With this multi-functional detector, robot can locate the remote people in the environment and provide the services.

## 2. Related work

There are some technologies used for tracking targets: Vision based algorithm using sequences of images from cameras that moving people can be tracked (Q. Cai & J. K. Aggarwal, 1998) or the number of people can be found (D. B. Yang et al., 2003). The infrared also can be used to facilitate the location of people. However, they can only record the count of people enter or exit in a certain area such as the door of a room (ACOREL Corp.). And this technology requires careful and dense deployment, and does not work in a more complicated environment. Some localization technologies adopt the acceleration and air pressure sensors to detect the location of people (R. J. Orr & G. D. Abowd, 2000). The obvious drawbacks of this technology are costly and need careful deployment. Some researches use ultrasonic sensor technology and adopt time of flight (TOF) method to obtain the location information. The "cricket" localization system uses a combination of ultrasonic and radio frequency (RF) to provide a location support service (Yunbo Wang et al., 2007).

## 3. Ubiquitous service space for smart home

Realize smart home, identification card with radio frequency transceiver on it which called "iCard" are needed as in Fig.1. iCard is a simple IEEE 802.15.4 (ZigBee) protocol transceiver and can be made as identify card.

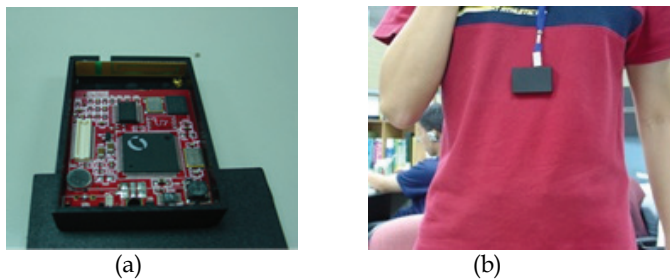


Fig. 1. (a) iCard (b) Wear on human

ZigBee protocol provides the unique ID information in the same ZigBee wireless environment, thus it can easily identify people with their own ID code.

Since the pyroelectric sensors are suffered from multi target tracking, therefore the assistance from radio frequency signal contain with the people ID information is needed for sensor network multi human tracking system.

iSensor is a multi-functional detector composed with ZigBee protocol transceiver, microprocessor, pyroelectric sensor, temperature sensor and smoke sensor. ZigBee protocol is designed as low-cost and low-power for home automation or hospital care. Pyroelectric sensor can detect the radiation from human body; with multi pyroelectric sensors the accuracy of people localization can be improved by data fusion technique. Temperature sensor data can be provided to HVAC system and adjust the temperature of environment. Smoke sensor is the original function and designed to detect fire disaster. The Architecture of iSensor is illustrated in Fig. 2.

Recent researches discuss on human tracking use pyroelectric sensor or radio frequency signal alone. However, the pyroelectric signal might suffer from tracking multi-targets, and accuracy of radio frequency tracking method might not be reliable. The iSensor will

integrate two types of radio and pyroelectric signal that can trace multi targets and the fidelity also can be guaranteed.

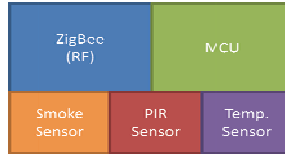


Fig. 2. Architecture of iSensor

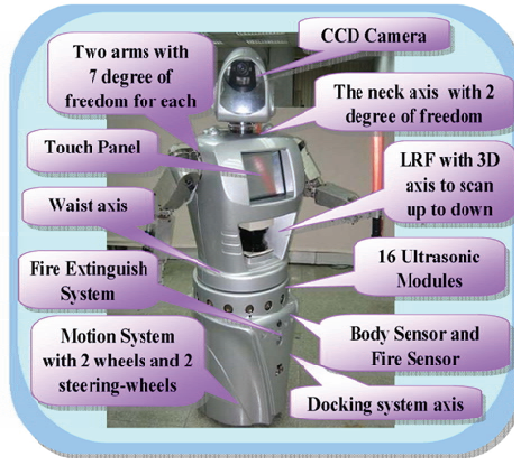


Fig. 3. System architecture of NCCU security warrior robot (Luo, R. C. et al., 2007).

We have developed a multisensor based intelligent robotics system as shown in Fig. 3. The multi-functional service robot can provide various services locally, however if remote area need services, how to cooperate with robot is an important issue. For example, as shown in Fig.4 people in the room 3 require services, if there are no sensor network system how the robot to determine and locate the service target?

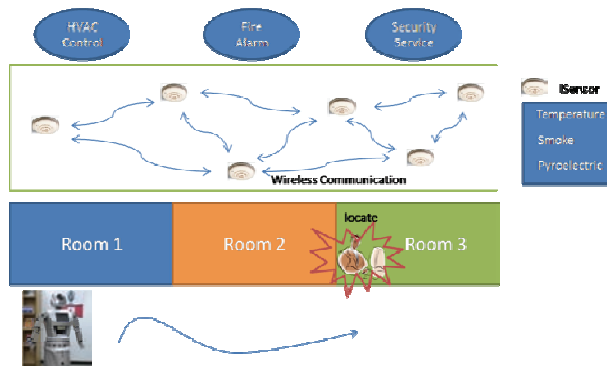


Fig. 4. Architecture of ubiquitous robotics services environment

The robot is equipped with ZigBee compatible communication module for communicating with sensor network. The iSensor network system installed in the ceiling, they collect ambient information and report to the multi-functional robot. Robots can perform tasks such as fire fighting, resident following or intruder detection.

#### 4. Pyroelectric sensor

The indoor tracking system should be implemented with many sensors installed in rooms; therefore low cost is the main consideration of smart home service. Since each room is different in shape and size, the location of obstacles which prevent the normal operation of sensors are also varied. A good localization system must be robust from noise and reduce the influence of surroundings.

The PIR sensor can detect the infrared wavelength emitted from humans. They are robust to their surroundings. iSensor are easily installed on the ceiling, where they are not easily affected by the structure of rooms or obstacles.

##### 4.1 Working principle

For a linear sensor, the response signal of  $n$  sensors  $S(t) \in \mathbb{R}^n$  is given by

$$S(t) = h(t) * \int_{\Omega} v(r) \Psi(r, t) dr \quad (1)$$

where  $*$  represents convolution operation,  $h(t)$  is the impulse function of a sensor,  $\Omega$  is the object,  $v(r)$  is the positive visibility function between  $n$  sensor and the object space,  $\Psi(r, t)$  is the radiation from the target.

The visibility  $v(r_1, r_2)$  denotes the contribution by the field at point  $r_2$  to the field at  $r_1$ .

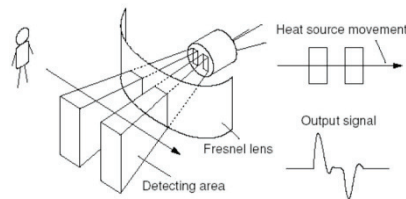


Fig. 5. Pyroelectric sensor behavior

Pyroelectric sensor signals are proportional to the change in temperature of the crystal rather than the temperature of environment. Fig. 5 shows a human walking through a pyroelectric sensor and the corresponding output signal. The response time of the transducer amplifier of the detector limits the maximum frequency. The resultant transfer function turns to be a bandpass one.

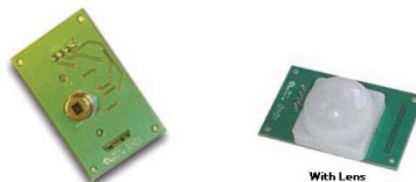


Fig. 6. pyroelectric detector

### 4.2 Pyroelectric sensor deployment and overlapping issue

In order to determine the location of residents within the monitoring area, an array of PIR sensors is used as shown in Fig. 7.

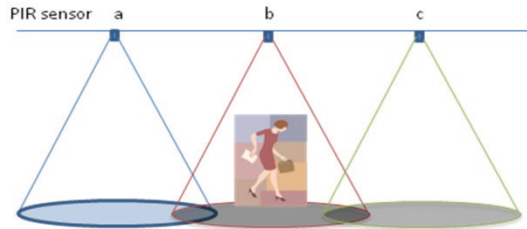


Fig. 7. The localization for PIR sensors

We find some interested questions when developing pyroelectric sensor system. A field of view of our sensor is 100° on directivity-horizontal and 60° on directivity-vertical, as shown in Fig 8. In other words, a field of view is an ellipse, and we must discuss this issue when deploying sensor nodes.

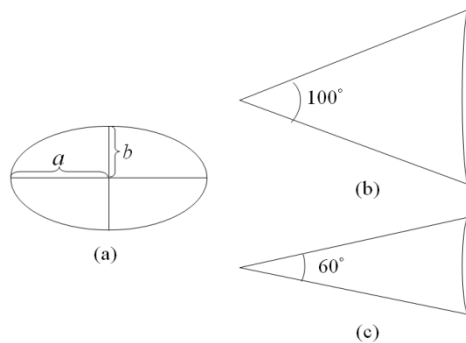


Fig. 8. The general architecture of sensor network system

Given an interested sensing field A, our approach is to get the signal of human radiation in an indoor environment. According to a specification of PIR sensor, a field of view of PIR sensor is an ellipse. And function of an ellipse is expressed as

$$(x - n)^2 / a^2 + (y - m)^2 / b^2 = 1 \tag{2}$$

and 
$$\begin{cases} x = a * \sin \theta + n \\ y = b * \cos \theta + m \end{cases}$$

Where n, m are coordinate of node; a, b are major and minor axis of an ellipse, respectively. When deploying node, calculate the relationship of distance between each node should be afresh. Fig. 9 illustrates the deployment of node without overlap; we find that an overlapping issue of ellipse is not like that of a circle. A graph that is account by three center of a circle is an isosceles triangle, not an equilateral triangle.

Assume that coordinate of n1, n2, and n3 is (0, 0), (0, 2a), and (a, a tanθ), respectively and  $\angle n_3 n_1 n_2 = \theta$ . And centre of gravity of the triangle can be obtained as  $(a, \frac{1}{3} * a * \tan \theta)$  in Fig. 9.



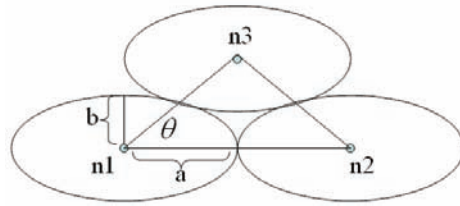


Fig. 9. Normal deployment without overlapping

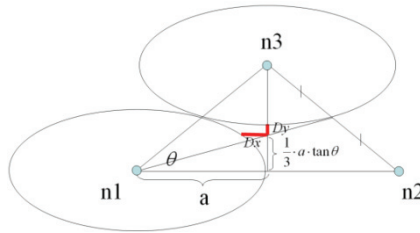


Fig. 10. The concept of  $D_x$  and  $D_y$

Consider the distance between nodes should be corrected by decreasing red and boldface lines ( $D_x, D_y$ ) to reach a minimal overlap.

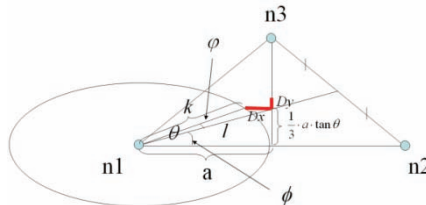


Fig. 11. Geometry of each angle and length

Use approximation theory to get a length of  $D_x$ , a degree of angle  $\phi$  is narrow enough that a length of  $(k+D_x)$  is similar to a length of  $l$  (i.e.  $D_x \cong l - k$ ) in Fig 9. A length of  $l$  is expressed as

$$l = \frac{1}{3} a * \tan\theta * \csc\phi \tag{3}$$

where  $\phi = \tan^{-1}(\frac{1}{3} * \tan\theta)$

And function of length  $k$  is as (4); because a degree of  $\phi$  is smaller than that of  $\theta$  (i.e.  $(\theta + \phi) \cong \theta$ ), approximate function can be obtained as (5).

$$k = \sqrt{[a \cdot \sin(\phi + \theta)]^2 + [b \cdot \cos(\phi + \theta)]^2} \tag{4}$$

$$k = \sqrt{(a \cdot \sin \phi)^2 + (b \cdot \cos \phi)^2}$$

Combining function of (3) and (5), get a function of length of  $D_x$  that we want to decrease and correct a distance between nodes in (6).

$$\begin{aligned}
 Dx &= (l - k) \\
 &= \frac{1}{3} \cdot a \cdot \tan \theta \cdot \csc \phi - \sqrt{(a \cdot \sin \theta)^2 + (b \cdot \cos \theta)^2}
 \end{aligned} \tag{6}$$

And function of length of  $D_y$  is expressed as

$$D_y = \frac{2}{3} a * \tan \theta - b \tag{7}$$

Combining (6) and (7), the distance between each node can be corrected. In this issue, we discuss questions of deployment and overlapping, and a discussion is not identical among ellipse and circle.

## 5. Localization through radio frequency signal

### 5.1 ZigBee radio frequency transceiver

Our indoor RF localization system utilizes the CC2420 chip. The CC2420 is a single-chip IEEE 802.15.4 compliant and ZigBee™ ready RF transceiver. It provides a highly integrated, flexible low-cost solution for applications using the world wide unlicensed 2.4 GHz frequency band. The mobile user also equips with CC2420 transceiver and floods the beacons every 50 ms.

For the direct line of sight propagation path, according to the free space model, the power received by the receiver is given by the Friis space equation (T. S. Rappaport, 2002) as

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2} \tag{8}$$

Where  $P_t$  is the transmitted power in watts,  $P_r$  is the received power,  $G_t$  is the transmitter antenna gain,  $G_r$  is the receiver antenna gain,  $\lambda$  is the wave length in meters, and  $d$  is the distance from transmitter to receiver.

### 5.2 Received signal strength

In order to locate residents, the estimated distance between sensor nodes and users is needed. Taking consideration of the estimation distance into localization algorithms; the coordination of ZigBee transceiver can be obtained. Radio signal propagation is easily influenced by diffraction, reflections, and scattering of radio induced obstacles in a building, thus the RSS signal measurement is contaminated by the measuring error and NLOS error. The measurement error results from the measuring processes in a noisy channel and can be improved with better signal-to-noise ratio (SNR). NLOS errors depend on the multi path dominated environments and change from time to time.

We record the received signal strength index (RSSI) and distances between the sensor nodes and users. By using maximum likelihood method the propagation model can be found for fading channel. This model provides the mean  $RSS(d)$  that received from mobile user, and equation (8) state that. The  $RSS(d_0)$  is the received signal strength in dB at a reference distance, and  $n$  is denoted the path loss exponent. The measured RSS is calculated by ML and find the parameter  $RSS(d_0)$  and  $n$ . The measured RSS fit into channel model is obtained by using the equation (8).  $\hat{d}$  is the estimation of the distance from equation (9), and  $X_o$  is the random variable that denoted the estimation error with variance  $\sigma^2$  from equation (10). We find that random variable  $X_o$  increases with distance between the sensor node and mobile user.

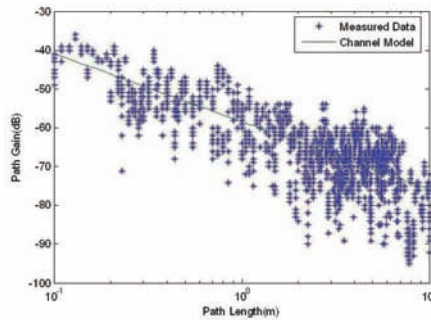


Fig. 12. The measured RSSI of radio signal

$$\text{RSS}(d) = \text{RSS}(d_0) - 10\log(d/d_0) \quad (9)$$

$$\hat{d} = d_0 10^{(\text{RSS}(d_0) - \text{RSS}(d))/10n} \quad (10)$$

$$\hat{d} = d + X_\sigma \quad (11)$$

## 6. Covariance intersection method

In the distributed environment such as sensor network, we cannot keep track on “which node received estimations from which level nodes”. Thus we do not know the degree of redundant information exists in an estimation a node received. It means that the error between predicted and actual position covariance will be underestimated. The covariance information must keep consistency to avoid the disastrous consequences of redundant data on Kalman filter type estimators. However, it is not possible to maintain cross covariance consistent with distributed. This makes the estimated state based on the assumed state model with little correction from the new measurements. Thus, drifts the state estimate away from the actual state. The Covariance Intersection (CI) (S. Julier & J. Uhlmann, 2001) can be treated as a generalized Kalman filter. The primary advantage of CI is that it permits filter and data fusion to be performed on probabilistically defined estimates without knowing the degree of correlation among those estimates. Thus CI does not need assumptions of the dependency of the two data of information, when it fuses them. If the cross-variance of the data is unknown, it is not possible to compute the exact covariance matrix of the estimate, but still desirable to have a pair estimate-covariance that is consistent, as defined below.

Set  $Z$  a random variable with mean  $\bar{z}$  and estimation  $\hat{z}$ . The estimation error can be given by  $\tilde{z} = \hat{z} - \bar{z}$  and the covariance associated with this error is  $P_{zz} = E\{\tilde{z}\tilde{z}^T\}$ . Let  $\overline{P_{zz}}$  be an estimation of the covariance of  $\hat{z}$ , then the pair  $\{\hat{z}, P_{zz}\}$  is said to be consistent if

$$P_{zz} \geq \overline{P_{zz}} \quad (12)$$

The proof can be found in (S. Julier & J. Uhlmann, 2001). The pair of estimate-covariance is consistent if the estimated covariance matrix is in the upper bound of the actual covariance of the estimate.

Set  $x$  and  $y$  be two random variables which have means and covariance matrices are  $E\{x\}=X$ ,  $E\{y\}=Y$  separately

$$\text{Cov}\{x\}=\text{P}_{xx}, \text{Cov}\{y\}=\text{P}_{yy}, \text{Cov}\{xy\}=\text{P}_{xy} \tag{13}$$

Define the estimate  $Z$  as a linear combination of  $x$  and  $y$ : where  $x$  and  $y$  might represent either a prior estimate of  $Z$  with certain covariance matrix or a measurement which has its own uncertainty.

The covariance intersection method is a data fusion algorithm which uses a convex combination of the means and covariances in the information field. This approach is referenced on a geometric interpretation of the Kalman filter process. The general form of the Kalman filter is

$$\hat{z} = W_x X + W_y Y \tag{14}$$

$$P_{zz} = W_x P_{xx} W_x^T + W_x P_{xy} W_x^T + W_y P_{yx} W_y^T + W_y P_{yy} W_y^T \tag{15}$$

The weights  $W_x$  and  $W_y$  are chosen to minimize the trace of  $P_{zz}$ . If the estimates are independent ( $P_{xy}=0$ ), the form of the conventional Kalman filter can be reduce.

The Covariance Intersection method provides estimation and a covariance matrix which their covariance ellipsoid encloses the intersection region. The estimate is consistent independent of the unknown value of  $P$ . Given the upper bound  $P_{xx} \geq \bar{P}_{xx}$  and  $P_{yy} \geq \bar{P}_{yy}$ , the covariance intersection estimator are defined as follows:

$$Z = P_{zz} \left\{ w_x P_{xx}^{-1} X + w_y P_{yy}^{-1} Y \right\} \tag{16}$$

$$P_{zz}^{-1} = w_x P_{xx}^{-1} + w_y P_{yy}^{-1} \tag{17}$$

$$w_x + w_y = 1, 0 \leq w_x, w_y \leq 1. \tag{18}$$

The parameter  $w_i$  gives the relative weights assigned to  $x$  and  $y$ . Different choices of  $w_i$  can be used to optimize the covariance estimate with different performance criteria such as minimizing the trace or the determinant of  $P_{zz}$ .

$$\text{Let } \alpha \equiv \sqrt{\text{tr}\{W_x P_{xx} W_x^T\}} \tag{19}$$

$$\beta \equiv \sqrt{\text{tr}\{W_y P_{yy} W_y^T\}} \tag{20}$$

$$\text{Thus } P_{zz} = \left( \frac{\alpha}{\alpha+\beta} P_{xx}^{-1} + \frac{\alpha}{\alpha+\beta} P_{yy}^{-1} \right)^{-1} \tag{21}$$

and the gains are

$$W_x = \frac{\alpha}{\alpha + \beta} P_{zz} P_{xx}^{-1} \quad W_y = \frac{\alpha}{\alpha + \beta} P_{zz} P_{yy}^{-1} \tag{22}$$

This theorem presents the advantage of the optimality of the best  $w_i$  in CI algorithm.

The main benefit of parallel estimation for each measurement type is the independence of the data fusion method for the location estimation calculation that has different measurement type each. The estimation calculations can be substituted with another location estimate calculation with an associated error covariance estimate and the data fusion technique can still be used.

## 7. Simulation and experimental results

The experimental test with investigation area is 5x5m<sup>2</sup>. The prototype of iSensor is installed on the ceiling. The red circles indicate the multiple sensor units in Fig. 13.



Fig. 13. Experiment geometry

The tracking strategy in our experiment for wireless pyroelectric sensor includes event detection, event digitization, covert sampled signals into event index, motion inference, fuse measurements from pyroelectric sensor and radio frequency signals, trajectory smoothing and estimate the trajectory of users.

An event is defined as an occurrence of thermal radiation detected by a pyroelectric sensor above a threshold value, and the thermal signal will be associated with motion across one detection region, we define such signal as an event.

When a human passes through the Fresnel modulated lens, the response signals are generated. Fig.14 illustrates the event time windows which generated from pyroelectric sensor.

Fig. 15 shows the signal when human user walked along the diagonal of the room, the response of 4 elements of a pyroelectric sensor module. After process the signals, we can convert those signals into angular and distance displacement.

Fig.16 shows the relation between RSS and the distance from mobile user to reference sensor node. The red line named "regression function" is a linear regression function calculated from mean value of measured data. The distance from mobile user to reference sensor node can be predicted accordingly by entering a measured RSS value to the regression function(22).

$$RSS = c_{m-1}d_{RSS}^{m-1} + c_m d_{RSS}^m + \dots + c_2 d_{RSS}^2 + c_1 d_{RSS} + c_0 \quad (23)$$

where

$$c = [-0.009 \ 0.944 \ -3.2332 \ -18.6146]^T$$

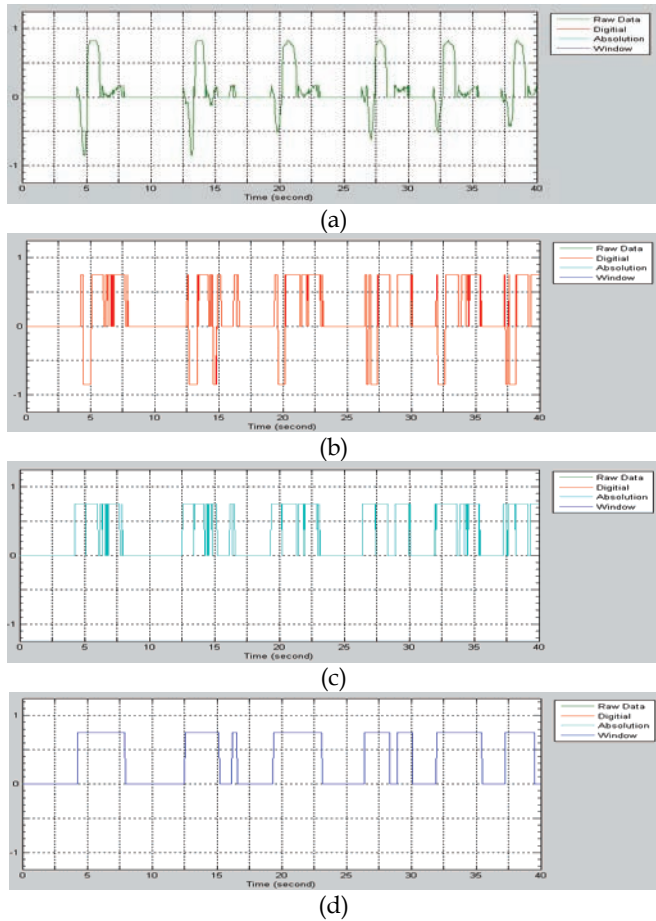


Fig. 14. Event detection from matched filter (a) Raw data (b) Digitized signals (c) Logic signals (d) Event windows.

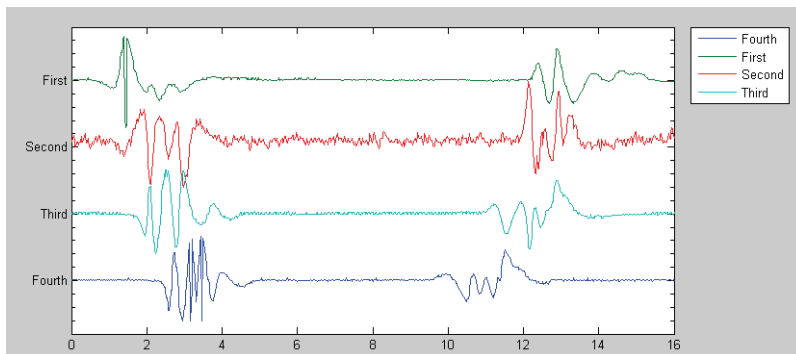


Fig. 15. Four elements signal when human user walk pass the pyroelectric sensor system

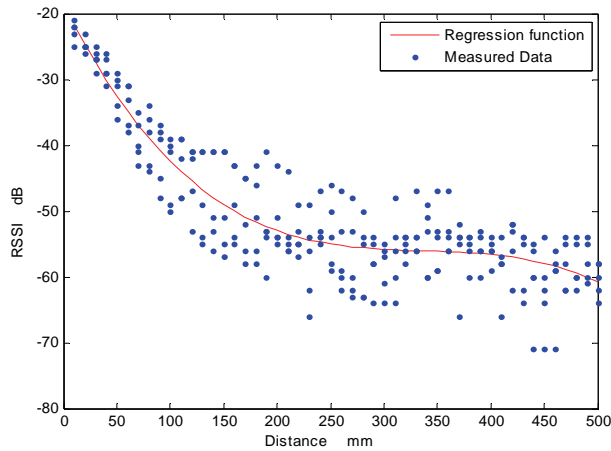


Fig. 16. RSS measured data

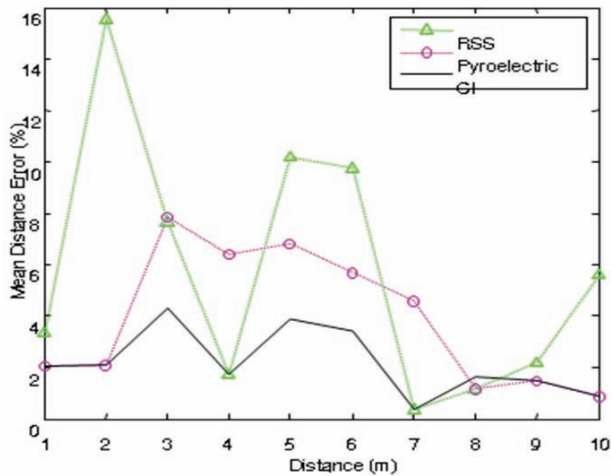


Fig. 17. Covariance Intersection with RSS and Pyroelectric measurement

Fig. 17 shows that covariance intersection fuses pyroelectric sensor and RSS localization estimation. The maximum mean distance error of RSS is near to sixteen percentages and the maximum mean distance error from pyroelectric measurement is near to eight percentages. Using covariance intersection algorithm, the mean distance error is below four percentages. The accuracy of locating residents under our system can be increased by using covariance intersection.

In the Fig. 18, the circle mark is the actual test trace, square is the measurement through pyroelectric sensor, triangular is the result from RSS and diamond is the fusion result from CI, The tracking error can be reduced through data fusion technique.

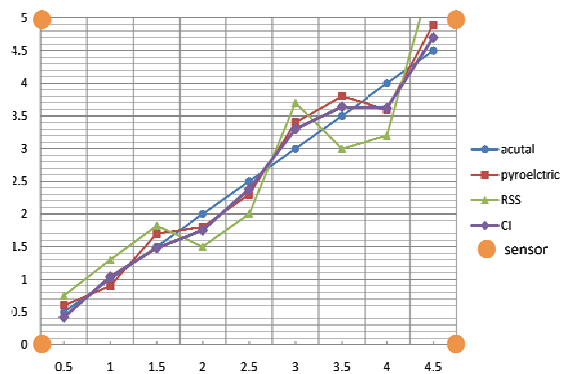


Fig. 18. Moving target tracking

## 8. Conclusion

This article presents a wireless multi-functional sensor call “iSensor” that is composed with radio transmission, microprocessor, pyroelectric, temperature and smoke sensor. With such device, the sensor network system can be implemented in indoor environment just replace original smoke sensor. This system can estimate the resident’s location and cooperate with robots to provide ubiquitous service space. In this paper, we build a prototype of wireless human tracking system by using pyroelectric and radio frequency signal. The measurement error from pyroelectric and radio frequency signal can be reduced through covariance intersection data fusion method. This work should extend to tracking multi targets in the near future. The data obtain from pyroelectric, temperature and radio signals, are also considered provide to air condition and adjust the temperature of the room thus energy saving for building can be realized.

## 9. References

- ACOREL Corporation, <http://www.acorel.com>.
- D. B. Yang, H. H. Gonzalez-Banos, and L. J. Guibas, "Counting people in crowds with a real-time network of simple image sensors," in *Proceedings of the Ninth IEEE International Conference on Computer Vision*, Page(s):122 - 129 vol.1, 13-16 Oct. 2003
- Djuric, P.M.; Vemula, M.; Bugallo, M.F.; "Target Tracking by Particle Filtering in Binary Sensor Networks" *IEEE Transactions on Signal Processing*, [see also *IEEE Transactions on Acoustics, Speech, and Signal Processing*], Volume 56, Issue 6, June 2008 Page(s):2229 - 2238
- Hao, Q.; Brady, D. J.; Guenther, B. D.; Burchett, J.; Shankar, M.; Feller, S.; "Human Tracking With Wireless Distributed Pyroelectric Sensors" *IEEE Sensors Journal*, Volume 6, Issue 6, Dec. 2006 Page(s):1683 - 1696
- Hua Li; Barbosa, P.R.; Chong, E.K.P.; Hannig, J.; Kulkarni, S.R.; "Zero-error target tracking with limited communication", *IEEE Journal on Selected Areas in Communications*, Volume 26, Issue 4, May 2008 Page(s):686 - 694



- Liu, J.; Chu, M.; Reich, J.E.; "Multitarget Tracking in Distributed Sensor Networks" *IEEE Signal Processing Magazine*, Volume 24, Issue 3, May 2007 Page(s):36 - 46
- Luo, R.C.; Chou, Y.T.; Liao, C.T.; Lai, C.C.; Tsai, A.C.; "NCCU Security Warrior: An Intelligent Security Robot System", *33rd Annual Conference of the IEEE Industrial Electronics Society, 2007. IECON 2007.*, Page(s):2960 - 2965, 5-8 Nov. 2007
- Q. Cai and J. K. Aggarwal, "Automatic tracking of human motion in indoor scenes across multiple synchronized video streams," in *Proceedings of the Sixth International Conference on Computer Vision*, Page(s):356 - 362, 4-7 Jan. 1998
- R. J. Orr and G. D. Abowd, "The Smart Floor: A Mechanism for Natural User Identification and Tracking," *Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems: Extended Abstracts*, ACM Press, New York (2000), pp. 275276.
- S. Julier and J. Uhlmann, "General decentralized data fusion with Covariance Intersection (CI)," in *Handbook of Multisensor Data Fusion*, D. Hall and J. Llians, Eds. Boca Raton, FL: CRC Press, ch. 12, pp. 12-1-12-25.. 2001,
- Suk Lee; Kyoung Nam Ha; Kyung Chang Lee; "A pyroelectric infrared sensor-based indoor location-aware system for the smart home", *IEEE Transactions on Consumer Electronics*, Volume 52, Issue 4, Nov. 2006, Page(s):1311 - 1317
- Taehong Kim; Young-Guk Ha; Jihoon Kang; Daeyoung Kim; Chong Poh Kit; Joo-Chan Sohn; "Experiments on Building Ubiquitous Robotic Space for Mobile Robot Using Wireless Sensor Networks", *22nd International Conference on Advanced Information Networking and Applications - Workshops, 2008. AINAW 2008.* Page(s):662 - 667, 25-28 March 2008
- T. S. Rappaport, "*Wireless communications: principles and practice*," in 2nd ed., Upper Saddle River, NJ: Prentice Hall PTR, 2002.
- Yoon-Gu Kim; Han-Kil Kim; Suk-Gyu Lee; Ki-Dong Lee; "Ubiquitous Home Security Robot Based on Sensor Network" *International Conference on Intelligent Agent Technology, 2006. IAT '06. IEEE/WIC/ACM*, Page(s):700 - 704, Dec. 2006
- Yunbo Wang; Goddard, S.; Perez, L.C. A study on the cricket location-support system communication protocols, *IEEE International Conference on Electro/Information Technology, 2007*, Page(s):257 - 262, 17-20 May 2007

# Design of an Intelligent Housing System Using Sensor Data Fusion Approaches

Arezou\_Moussavi Khalkhali, Behzad\_ Moshiri, Hamid Reza\_ Momeni  
*Telecommunication Infrastructure Company of Iran,  
 Ministry of Information and Communication Technology,  
 Control and Intelligent Processing Centre of Excellence, School of ECE,  
 University of Tehran  
 Department of Electrical Engineering, Tarbiat Modarres University  
 Tehran, Iran*

## 1. Introduction

Today's security systems are expected to function more accurately and efficiently. Intelligent homes as a high demanded future technology need to become less expensive and more certain. These goals could be achieved by using fusion methods. In the following, two theories are applied to a home security system which are Dempster-Shafer evidential theory and Proportional Conflict Redistribution Rule no.5 (PCR5). The significant aim of this research is to show the application of fusion methods in order to establish the security system used in Intelligent Housing Systems (IHS) and detecting the precise location of the intruder at home, which are not viable through the traditional systems. The exactitude of PCR5 method to Dempster-Shafer Theory (DST) has been considered after applying them to the system. In the following, home security system is simulated by MATLAB. In sections 2 and 3, this research work will review the DST and PCR5 and their combinational rules. Section 4 deals with the security system and applying the theories to a scenario, and finally part 5 presents the conclusions obtained by simulating the attack scenarios.

## 2. Dempster-Shafer evidential theory basis

In Dempster-Shafer Theory (DST), there is a frame of discernment  $\theta$ , which its elements are all possible states of a system. Therefore, the Dempster-Shafer (DS) fusion process is based on  $2^\theta$  elements called propositions.

To every subset in this frame a probability mass is assigned which is called basic probability assignment or basic belief assignment (bpa or m).

m; must satisfy the following conditions :

$$m: 2^\theta \rightarrow [0, 1], m(\emptyset) = 0, \sum_{A \in 2^\theta} m(A) = 1 \quad (1)$$

The probability that the true answer is A denoted by a *confidence interval*: [Belief (A), Plausibility (A)] in which,

$$\text{Bel}(A) = \sum_{B \subseteq A} m(B) \quad (2)$$

$$\text{Pl}(A) = 1 - \sum_{B \cap A = \emptyset} m(B) \quad (3)$$

The width of the interval therefore represents the amount of uncertainty in A, given the evidence.

The belief function  $\text{Bel}(A)$  in a subset, entails belief in subsets containing that subset. The plausibility function measures the total belief mass that can move into A. For combining two belief functions over the same frame of discernment with different bps ( $m_1$  and  $m_2$ ) and different sources, DS combination rule is used:

$$m(C) = [m_1 \oplus m_2](C) = \frac{\sum_{A \cap B = C} m_1(A)m_2(B)}{1 - \sum_{A \cap B = \emptyset} m_1(A)m_2(B)} \quad (4)$$

In which  $k = \sum_{A \cap B = \emptyset} m_1(A)m_2(B)$  is interpreted as a measure of conflict among the various sources (Blaylock & Allen, 2004; Sentz & Ferson, 2002; Wu et al., 2002).

As an example consider a frame of discernment with three possible states  $H = \{A, B, C\}$ , then all subsets of H are 2<sup>3</sup> elements which are:

$\{A\}, \{B\}, \{C\}, \{A, B\}, \{A, C\}, \{B, C\}, \{A, B, C\}, \{\emptyset\}$ .

$\text{Bel}(B, C) = m(\{B\}) + m(\{C\}) + m(\{B, C\})$

$\text{Pl}(B, C) = m(\{B\}) + m(\{C\}) + m(\{B, C\}) + m(\{B, A\}) + m(\{A, B, C\}) + m(\{C, A\})$

Suppose that

$m_1(G) = 0.6, m_1(V) = 0.3, \theta_1 = (GUV) = 0.1,$

$m_2(G) = 0.5, m_2(V) = 0.35, \theta_2 = (GUV) = 0.15$

Then,

$m(G) = [(0.6*0.5) + (0.6*0.15) + (0.1*0.5)] / [1 - (0.6*0.35) - (0.3*0.5)] = 0.6875$

It could be seen that the combinational probability is more than the single probabilities of each source.

### 3. Proportional conflict redistribution rule no.5 (PCR5) basis

The basic idea of PCR rules is reallocating the partial conflicts to the non-empty sets that contribute in the conflict mass.

The process of applying PCR to calculating the total mass of several sources is as follows:

- Applying the conjunctive rule to calculate the belief masses of sources;
- Calculating the total or partial conflicting masses;
- Redistributing the total or partial conflicting masses to the non-empty sets contributed in the conflicts correspondingly to their original masses.

Several versions of PCR rules derived from the way in which the conflicting mass is reallocating to the non-empty sets.

According to the PCR5 rule the partial conflicting mass redistributes to the non-empty sets that involved in the partial conflict.

The PCR 5 formula for combining two sources is as follows (Dezert & Smarandache, 2006, (a)):

$mPCR5(\emptyset) = 0$  and  $\forall X \in G \setminus \{\theta\}$

$$mPCR5(X) = m12(X) + \sum_{\substack{Y \in G \setminus \{X\} \\ X \cap Y = \emptyset}} \left[ \frac{m_1(X)^2 m_2(Y)}{m_1(X) + m_2(Y)} + \frac{m_2(X)^2 m_1(Y)}{m_2(X) + m_1(Y)} \right] \quad (5)$$

Where  $m12(X) = \sum_{\substack{X, Y \in G \\ X \cap Y = X}} m_1(X) m_2(Y)$

Proof and complete explanation about PCR5 rule and other PCR rules could be found at (Dezert & Smarandache, 2006(a), Smarandache & Dezert, 2006(b)).

In this paper we use the above formula to combine the result of two sources with the third one and so forth.

PCR5 rule is more exact than Dempster's rule. This is because of how it redistributes the conflicting mass to the sets, which are involved in the conflict rather than redistributing to all non-empty sets like what happens in Dempster's rule.

#### 4. Simulation results

In order to simulate the security system, imagine a home with sensors located in different areas according to Fig. 1. The security system discussed here is a system, capable of detecting intruders. If there is need to protect the home from fire, as a result the smoke detectors and heat detectors should be used. Now four kinds of sensors are used to implement the system:

- Wall vibration intended to detect mechanical vibrations caused by chopping, sawing, drilling, ramming or any type of physical intrusion.
- PIR/Microwave in which microwave and PIR (passive Infrared) sensors are electronically connecting together with AND logic. Microwave sensors are active devices, which cover a zone or an area with electrical field and detect movement and PIRs are passive, which detect a heat-emitting source (human bodies).
- Sound detectors that "listen" to the noises produced by the intruder.
- Glass-Break detectors, which are sensitive to 5 kHz, shock and frequencies produced if a glass is broken.

It is tried to use almost maximum number of sensors, but it can be changed by the designer's opinion. In designing the IHS, it is tried to indicate the zone that the intruder attacks there. The home is divided into 6 areas as shown in figure 1.

Considering the table1, the probability of detection of sensors is estimated as following (Rowshan & Simonetta, 2003):

PIR/Microwave: VL = 0-0.2, L = 0.2-0.4, M/H = 0.4-0.6, H = 0.6-0.8, VH = 0.8-1

Sound detector: VL = 0-0.3, L = 0.3-0.5, M/H = 0.5-0.7, H = 0.7-0.9, VH = 0.9-1

The worst condition for the system is when an intruder is crawling as given in table 1. The threshold probability for detection of sound detectors set to 0.3 and for PIR/Microwaves, set to 0.45. These values are supposed to be 0.25 for wall sensor and 0.4 for glass-break detectors. The ignorance of all sensors is set to 0.1.

The system checks the 22 sensors' sample for every 0.5 seconds. As soon as one sensor rises up the threshold, the system looks for another and combines the output of them to check if there is a real attacking. If an intrusion happened, depending on which zone's sensors participate in combination, the corresponding zone alarm would be triggered.

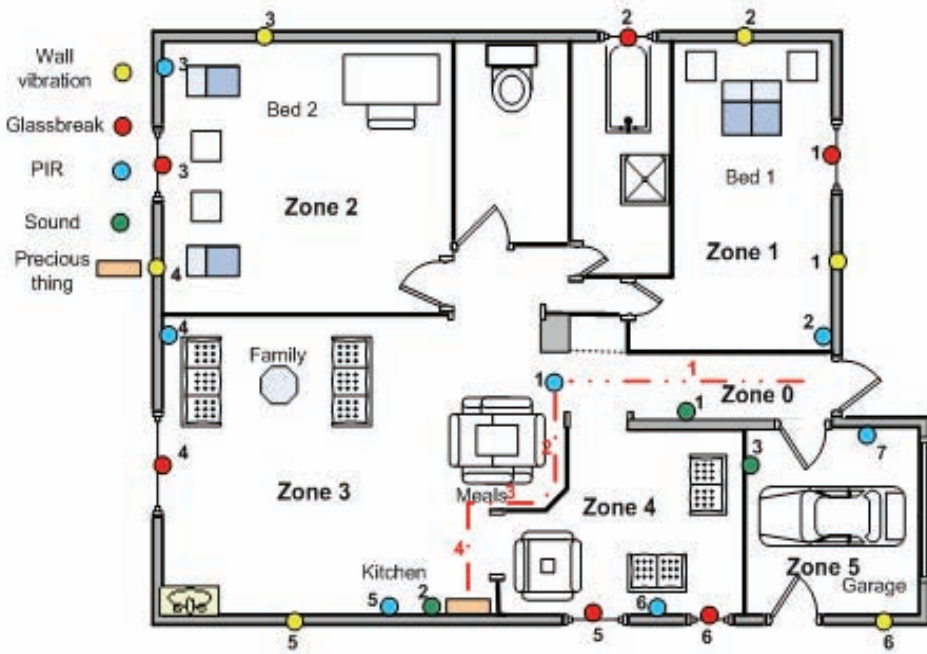


Fig. 1. House plan with sensors located in it

The sensors are sensitive to the delay between two detections and the system resets if the intruder delayed between two actions, so the system is programmed in a way that each time one sensor's output increases the threshold value, the system holds it for 10 minutes. As soon as the new value becomes greater than the last one, the newer one will be held and subsequently this recent new value would be considered in the corresponding calculation process.

First, consider a room with the five sensors mentioned above: two glass-break detectors, two PIRs, and one wall sensor. The mentioned system is simulated by the Monte-Carlo method in which, one mathematical experiment with random numbers is repeated for thousands of times (C.Henderson et al., 2005).

The probabilities for detection produced by the sensors are random numbers. Applying the output value for the sensors is repeated for 1000 times. About 100 times out of 1000, the results of two methods contradicted each other. One of those contradictory conditions is considered below. The mass of intruder and secure for five sensors are as follows:

$$\begin{array}{lllll}
 m1(i) = 0.17 & m2(i) = 0.76 & m3(i) = 0.16 & m4(i) = 0.15 & m5(i) = 0.89 \\
 m1(s) = 0.73 & m2(s) = 0.14 & m3(s) = 0.74 & m4(s) = 0.75 & m5(s) = 0.01 \\
 \theta = 0.1
 \end{array}$$

With these probabilities, DST method considered the situation safe; however, PCR5 detects the condition as a dangerous one. The calculation process for DST and PCR5 is shown in table.2 and table.3.

Sensor Systems	Slow Walk	Walk	Run	Crawl	Roll	Jump
Sensor Lists- Estimate Probability of Detection- very low VL, low L, medium M, high H, very high VH, N/A not applicable	-	-	-	-	-	-
Binary Sensors	N/A	N/A	N/A	N/A	N/A	N/A
Fix Barrier/ Wall Sensor	N/A	N/A	N/A	N/A	N/A	N/A
Infrared Sensors						
Infrared Beambreak Detector	VH	VH	VH	M/H	H	H
Passive infra-Red Sensor (PIR) Detector(Heat sensor)	VH	VH	VH	M/H	H	H
Microwave Sensors						
Microwave Bistatic	H	VH	H	M/H	M/H	M/H
Microwave Monostatic	H	VH	H	M/H	M/H	M/H
Other Sensors						
Dual Technology Passive IR/Microwave	VH	VH	VH	M/H	H	H
Sound Sensors	L	M	M/H	VL	L	M

Table 1. The estimate probability of detection for sensors

The step of calculation is according to the direction of arrows in the upper left cell of the tables. I.e. by fusing m1 and m2, m12 is achieved, then by fusing m3 and m12, m123 is deduced, and so forth.

Take note that although sensors number 2 and 5 display a high range of perilous situation, the DST method did not pay attention and announce the safe mode, however using PCR5 method by which conflict mass redistribute proportionally to the partial masses, danger mode is detected.

	$m1(i)=0.17$ $m1(s)=0.73$ $\theta1=0.1$	$m3(i)=0.16$ $m3(s)=0.74$ $\theta3=0.1$	$m4(i)=0.15$ $m4(s)=0.75$ $\theta4=0.1$	$m5(i)=0.89$ $m5(s)=0.01$ $\theta5=0.1$
	$m2(i)=0.76$ $m2(s)=0.14$ $\theta2=0.1$	$K=0.4214$ $m12(i)=0.5273$ $m12(s)=0.4490$ $\theta12=0.0237$	$K=0.5380$ $m123(i)=0.2619$ $m123(s)=0.7337$ $\theta123=0.0044$	$K=0.6935$ $m1234(i)=0.0954$ $m1234(s)=0.9040$ $\theta1234=6.36e-004$

Table 2. Calculation of the mass of intruder and secure with DST method

	$m1(i)=0.17$ $m1(s)=0.73$ $\theta1=0.1$	$m3(i)=0.16$ $m3(s)=0.74$ $\theta3=0.1$	$m4(i)=0.15$ $m4(s)=0.75$ $\theta4=0.1$	$m5(i)=0.89$ $m5(s)=0.01$ $\theta5=0.1$
	$m2(i)=0.76$ $m2(s)=0.14$ $\theta2=0.1$	$m12(i)=0.5182$ $m12(s)=0.4718$ $\theta12=0.01$	$m123(i)=0.3134$ $m123(s)=0.6856$ $\theta123=0.001$	$m1234(i)=0.1662$ $m1234(s)=0.8337$ $\theta1234=1e-004$

Table 3. Calculation of the mass of intruder and secure with PCR5 method

The above situation can be mapped into a dining room with five sensors included two glass-break detectors, two PIRs, and one wall sensor. One of the glass-break detectors and one of the PIRs detect the intruder with the mass 0.89 and 0.76 respectively. The other sensors cannot detect anything according to their masses. Applying two methods implied that by using PCR5 rule, which is more exact than DST, the system could detect the dangerous situation. Now by implementing the main scenario to IHS with 22 sensors, there can be a better understanding between two theories by comparing the differences.

Fig. 2 (1 to 5) indicated the function of detecting by the sensors, i.e. it shows the sensors' detection status corresponding to the movement of an intruder.

It is assumed that the total time for traversing the path to reach the object shown in Fig. 1 is 130 seconds. Another assumption is that each sensor takes a sample in every 0.5 seconds.

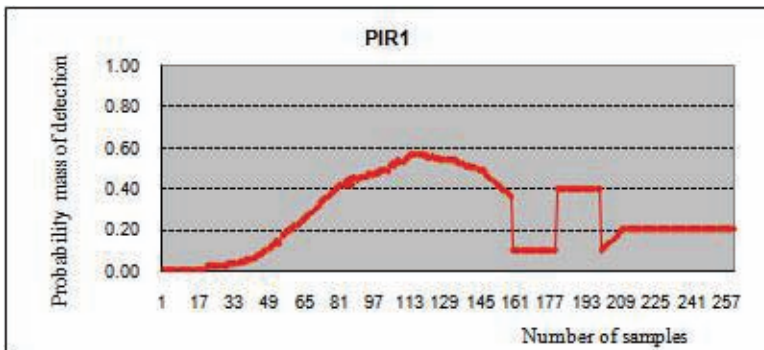


Fig. 2.1. Output pattern of PIR1 located in zone 0

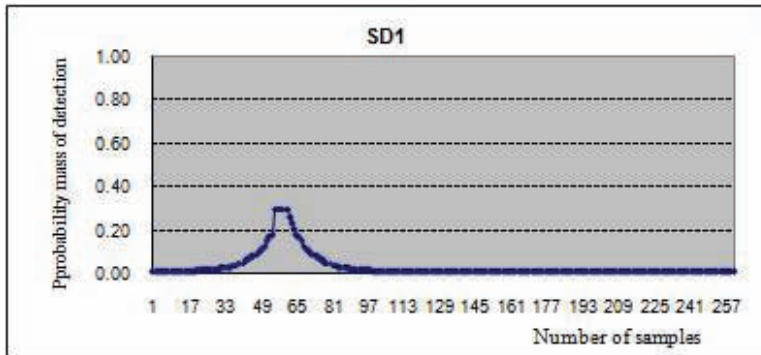


Fig. 2.2. Output pattern of SD1 located in zone 0

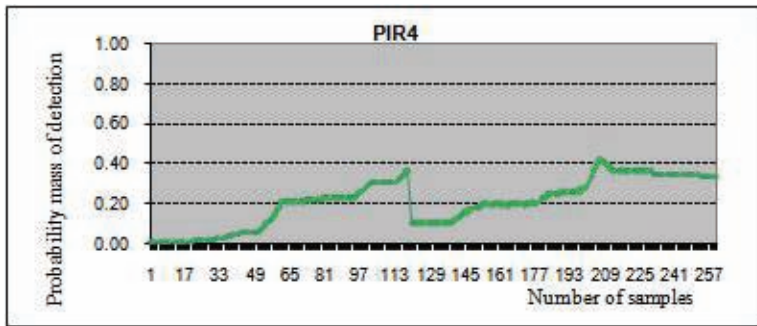


Fig. 2.3. Output pattern of PIR4 located in zone 3

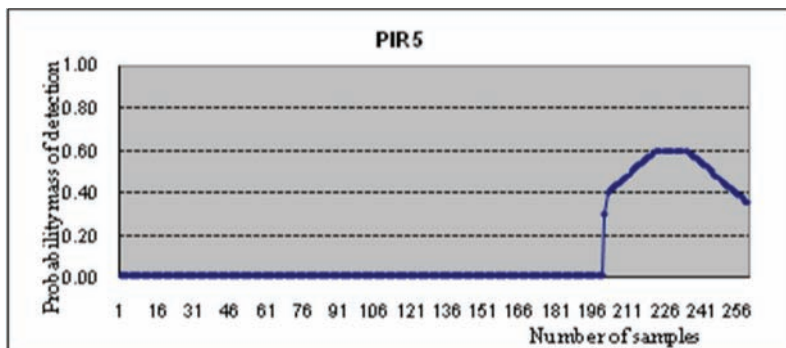


Fig. 2.4. Output pattern of PIR5 located in zone 3



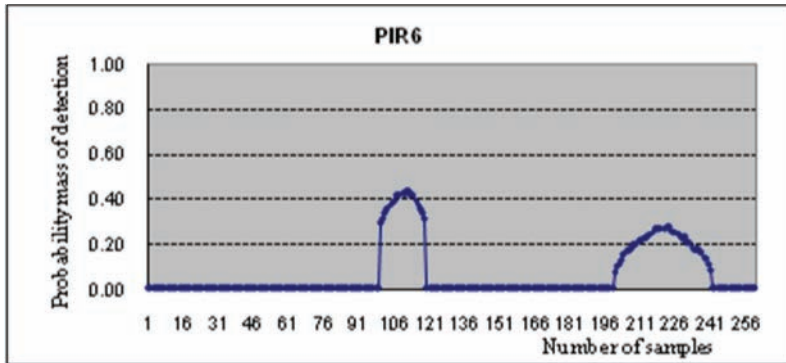


Fig. 2.5. Output pattern of PIR6 located in zone 3

Therefore, there are 260 samples in total. The horizontal axis in Fig.2 shows the samples and the vertical one indicates the probability of the detection.

Considering the SD1 graph as an example, it is noticed that the sensor began to detect the thief until around the sample 57, where the peak of the probability of the detection of the sensor appears. This is due to the minimum distance between the intruder and the sensor. Afterward when the intruder receded from the sound detector, the probability of the detection is also decreased.

It is assumed that it takes 60 seconds (120 samples) to pass the corridor (path 1), 30 seconds to pass the meal table (path 2), 10 seconds to crawl the path 3, and 30 seconds to reach the object (path 4).

By simulating the security system with MATLAB, around sample 56 the PCR5 method announces the danger mode, but DST is safe yet until the 208 sample. At sample 208 DST alarms that the intruder attacks in zone 3. In sample 56 the mass of intruder and secure of PIR4, SD1, and PIR1 in the entrance of the home are as follows:

$$\begin{array}{lll}
 m_{PIR4}(i) = 0.47 & m_{PIR4}(s) = 0.43 & \theta_{PIR4} = 0.1 \\
 m_{SD1}(i) = 0.3 & m_{SD1}(s) = 0.6 & \theta_{SD1} = 0.1 \\
 m_{PIR1}(s) = 0.58 & m_{PIR1}(i) = 0.32 & \theta_{PIR1} = 0.1
 \end{array}$$

With using formulas number 4 and 5 it can be seen that by DST calculation the system is in the safe mode and by PCR5 is in the attack mode. Therefore, the system realized the intruder at the first minutes of his entrance and alarmed or called the local police station, or even locked the doors or it can done any prevention actions by which it is programmed. The process for calculating the PCR5 mass of intruder and secure is shown below.

$$m_{12PCR5}(i) = 0.218 + [(0.13254/1.07) + (0.0387/0.73)] = 0.3949$$

$$m_{12PCR5}(s) = 0.361 + [(0.05547/0.73) + (0.1692/1.07)] = 0.5951$$

$$\theta_{12PCR5} = 0.01$$

$$m_{123PCR5}(i) = 0.5154$$

$$m_{123PCR5}(s) = 0.4845$$

$$\theta_{123PCR5} = 0.001$$

## 5. Conclusion

As shown in Fig.2 the probabilities of the sensors for activating are very low. At least PIR's are more sensitive than what were supposed here and the sensors had to detect the intrusion with higher probability. Meanwhile the worst conditions of the sensors for detection have been considered. However, the proposed system based on data fusion concept could easily detect the intruder.

One of the advantages of using this system is detecting the zone where the intruder attacked, so based on the location of the house, the different mechanisms could be used in order to trap the intruder.

The higher reliability of the simulated security system was achieved due to the redundancy and complementary characteristics of the sensor fusion itself, and the nature of the parallel processing of sensor data fusion approach provides less costly information processing. In this scenario the "m (intruder  $\cap$  secure) =  $\emptyset$ ", as a result, the conflicting masses in PCR5 are transferred to those that contributed in conflicting mass. For future work, "m (intruder  $\cap$  secure)  $\neq \emptyset$ " could be considered; meanwhile, the other fusion approaches using fuzzy integral operator or neuro-fuzzy method could be applied.

## 6. References

- Blaylock, N. ; and Allen, J. (2004). Statistical Goal Parameter Recognition, *Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling (ICAPS'04)*, PP. 297-304, Whistler, British Columbia, June 3-7, 2004, AAAI Press, Whistler
- C.Henderson, T.; Grant, E. & Luthy, K. (2005). *Precision Localization in Monte Carlo Sensor Networks*, Proceedings of the ISCA 18th International Conference on Computer Applications in Industry and Engineering PP.26-31, ISBN 1-880843-57-9, Sheraton Moana Surfrider, Honolulu, Hawaii, U.S.A., November 9-11, 2005, ISCA
- (a) Dezert J.; Smarandache F. (2006). Introduction to the Fusion of Quantitative and Qualitative Beliefs, *Information & Security, an International Journal*, Vol., 20, (June 2006), pp. 9-49
- Rowshan, Sh.; Simonetta, R. (2003). *Intrusion Detection for Public Transportation Facilities Handbook*, Transportation Research Board of the National Academies, ISBN 030906760X, Washington D.C., U.S.A.
- Sentz, K. ; Ferson, S. (2002). Combination of Evidence in Dempster-Shafer Theory, *Technical Report SAND 2002-0835*, 96 pages, Sandia National Laboratories, Albuquerque, NM
- (b) Smarandache F.; Dezert J. (2006). *Advances and applications of DSMT for Information Fusion*, (Collected works), Vol.2, American Research Press, ISBN 1-59973-000-6, Rehoboth, U.S.A.

- Wu, H.; Siegel, M.; Stiefelhagen, R. & Yang, J. (2002). Sensor Fusion Using Dempster-Shafer Theory, *Proceedings of 19th IEEE Instrumentation and Measurement Technology Conference*, IMTC '03, Vol. 2, PP. 907-912, , May 20-22, 2002, IEEE, Anchorage

# Model-based Data Fusion in Industrial Process Instrumentation

Gerald Steiner

*Institute of Electrical Measurement and Measurement Signal Processing  
Graz University of Technology  
Austria*

## 1. Introduction

Process sensors form an essential component of modern industrial production processes. They usually have to be operated under stringent environmental, safety, and cost constraints. Some of the key requirements on process instrumentation are: operation under harsh and varying environmental conditions, high reliability, fault tolerance, and low cost. The failure of process sensors may cause high losses due to plant breakdowns or out-of-specification products. Therefore it is important to know as much as possible about the momentary states of a production process. In addition, the estimates need to have low uncertainty and high reliability.

Sensor and data fusion can be key techniques to economically reach these goals and achieve higher performance than with isolated single point sensors alone. These methods enable the quantification of otherwise inaccessible quantities that cannot be deduced from a single sensor or management principle. Examples are the concentration measurement of ternary solutions or the tomographic estimation of spatially distributed material parameters from arrays of single point sensors. Industrial processes are usually operated within a defined environment, although there may be very harsh conditions like temperature variations, aggressive fluids, and high humidity. There are certain limits of operation. The nominal parameters, like desired product specifications as well as normal or acceptable fluctuations are known in advance while unknown encounters like in classical sensor fusion for target tracking, autonomous guidance, and battlefield surveillance are not within the scope of operation.

This fact potentially establishes a precise and specific model of the industrial process at hand. The knowledge then contained in the process model can be fruitfully exploited in model-based data fusion. Generally, model-based approaches reach beyond straightforward methods like physical redundancy with majority votings or heuristic filtering operations. The achievable measurement precision as well as decision making reliability is usually higher in model-based approaches due to the additional regularization of the state or hypotheses space that is achieved with an appropriate model. However, at the same time special care needs to be taken to choose a model that allows for sufficient and representative variation. This is the only way the inherent variability of a process can be adequately represented.

The well-known JDL (Joint Directors of Laboratories) data fusion process model is a popular and useful conceptual framework for the classification and comparison of data fusion approaches (Hall & Llinas, 1997), (Varshney, 1997), (Macii et al., 2008). It was originally developed with military applications like surveillance and target tracking in mind. It has been pointed out that the JDL model does not fully address data fusion problems from non-military areas, like image fusion. However, a classification of such fusion algorithms may be useful for a common understanding. After a preprocessing stage the JDL model distinguishes four levels of processing:

- Object refinement (level 1)
- Situation refinement (level 2)
- Threat refinement (level 3)
- Process refinement (level 4)

Due to the frequent availability of a well-defined process description, the model-based data fusion approaches in industrial process instrumentation are classified as level 1 processes in most cases. In this object refinement stage parametric, locational, and identity information are combined. Major functions are the transformation and alignment of data to a suitable reference frame, and the estimation and prediction of states. In terms of process instrumentation, the desired output of the object refinement process are quantitative and unambiguous figures. Due to the precise knowledge of the desired target quantities these figures can be straightforwardly used for an objective assessment of the industrial process. Therefore the higher levels of the JDL model do not play an as important role in industrial as in military applications.

According to another popular characterization of data fusion approaches the fusion can take place at different stages of the signal processing chain. A common categorization uses three levels (Varshney, 1997), (Hall et al., 1999):

- Data-level fusion
- Feature-level fusion
- Decision-level fusion

In data-level fusion the raw data from each of the sensors are combined. In this context raw data are single measurements like temperature or pressure readings. All further processing is based on the totality of data. This approach is able to yield the most accurate results, but it requires the sensors to be commensurate, i.e. that the different data can be processed in a common framework. If the data are in different regimes they have to be registered first, e.g. through coordinate transformations. Data-level fusion requires centralized data processing, since the totality of raw data has to be simultaneously available. A high communication bandwidth is necessary since all raw data have to be transferred. In feature-level fusion the raw data of each sensor are processed locally. A feature vector is generated from the corresponding observations. Features can, e.g., be volume fractions of materials, flow rates, or flow profiles, which are derived from multiple single point measurements. The different vectors are then fused to give a single feature vector. The necessary communication bandwidth is reduced compared to data-level fusion. However, the generation of the single feature vectors results in some data loss in general. Finally, in decision-level fusion, each sensor derives higher-level decisions from its own data and features. A decision could be whether there is a process malfunction or not. The individual decisions are finally fused by some sort of voting to give the final inference. The high required bandwidth in data-level fusion may be a severe disadvantage in large scale distributed target tracking applications,

but is not regarded as a major issue in process instrumentation. Sensors that share a common or similar state space or reference frame are usually installed in close vicinity. In distributed production environments the derived feature and decision information is passed on to the control room.

This chapter is intended to give an overview of applications of model-based data fusion in the context of industrial instrumentation and process monitoring. The diverse examples addressed are grouped according to a classification in: uniquely determined, overdetermined, underdetermined, or sequential data fusion (Tanner, 2003). Data fusion may already take place in instruments that are received as single sensor installations from an outside perspective (Ruhm, 2007). Examples are sensors that rely on additional measured quantities to compensate for unwanted cross-sensitivities. Internal temperature compensation of the primary measurand, e.g., is indispensable in many instruments. A next step is data fusion of multiple independent and non-redundant sensors to compensate for cross-sensitivities among the primary measurands that can be explicitly modelled. A typical application is the concentration measurement in ternary or multinary solutions. A characteristic feature of this class of sensor fusion problems is that the number of unknown parameters can be uniquely determined from the number of input quantities of the fusion process.

A concept more easily perceived as data fusion is the combination of identical parallel sensors to provide redundant information. This leads to a higher degree of security and reduces measurement uncertainty. The use of such redundant sensor arrays has a long tradition in industrial process instrumentation as sensor breakdowns and false decisions can have dramatic implications. Even single sensor failures may be able to cause malfunctions of large-scale facilities like power plants if security issues are not properly addressed in the system architecture. A slightly different approach than redundancy maximization is the use of identical sensors in a specific spatial configuration. By using appropriate models it is then possible to deduce quantities that cannot be measured with a single point sensor of the same measurement principle. Further benefits are increased accuracy of the estimates and powerful error compensation without additional sensors. This approach is convincing if very simple and cost-efficient elements can be used and will be illustrated by means of capacitive and magnetic sensors for angular position measurement. In these cases the number of measurement values exceeds the number of unknown parameters to be determined.

The third class of data fusion applications covers the converse case where the number of independent measurements from a homogeneous sensor array is actually smaller than the number of unknown parameters. This occurs when spatially varying material parameters are to be determined using industrial process tomography. The number of measurements obtained at the boundary of a problem domain is limited compared to the appropriate discretization of the domain. This leads to ill-posed inverse problems. The incorporation of additional prior knowledge through the process model is essential in order to obtain a meaningful solution of the problem. Here the concept of model-based measurement peaks in its relevance as a framework for multi-sensor data fusion. Solution strategies for tomographic problems are introduced in the corresponding section and the importance of the process model is discussed. Besides single-modality tomographic data fusion, multi-modality fusion is also addressed. For this case a sequential fusion process is proposed.

In subsequent sections, grouped according to uniqueness of the data fusion solution, several model formulations will be introduced. These model formulations range from static to dynamic, ideal to those including nuisance effects and noise, and explicit to implicit. Also different fusion algorithms including response surfaces, stochastic filters, and sequential fusion will be addressed within the respective applications.

## 2. Non-redundant measurement processes

### 2.1 Error compensation

Measurement processes and the superordinate fusion algorithms can be modelled in various ways. Depending on the actual situation and the aim of the measurement they may be formulated as static or dynamic systems. For certain applications a static model may be sufficient although virtually every sensor shows some kind of dynamic behaviour. For the following considerations we assume a vector  $z^T = (z_1 \ z_2 \ \dots)$  of two or more measurements as input to a nonlinear static fusion equation  $f$ .

$$x = f(z) \quad (1)$$

The output of the fusion process is the scalar variable  $x$ . This kind of data fusion can already be found in most of modern process instruments. In the simplest case a single input variable is directly related to the output of the fusion relation and defines the primary measurement equation. An illustrative example that can be found throughout the process industries is density measurement. A classical instrument for that purpose is the vibrating tube densimeter (Ihmels et al., 2000) (Laznickova & Huemer, 1998). The operating principle based on a spring-mass system is sketched in figure 1 using a U-shaped tube. The process fluid under test flows through a metal or glass tube that is decoupled from the surrounding through a base mass. In order to measure the fluid density the tube is excited by a force acting on the bend. The tube then vibrates at its resonant frequency, which is sensed by the pick-up mechanism.

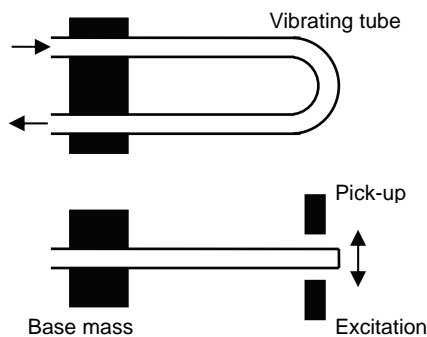


Fig. 1. Principle of the vibrating tube densimeter.

The resonant frequency depends on the mass  $m_T$  of the tube and the mass of the fluid  $m_F$  inside the tube. With the spring constant  $c$  the relation for the resonant frequency  $f_R$  and the period of oscillation  $\tau_R$  can be obtained.

$$f_R = \frac{1}{\tau_R} = \frac{1}{2\pi} \sqrt{\frac{c}{m_T + m_F}} \quad (2)$$

As the volume of the tube is constant and known the measurement is proportional to the fluid density. Equation (2) can be reformulated to give the density  $\rho$  as a function of the period of oscillation, where  $A$  and  $B$  are constants.

$$\rho = A\tau^2 + B \quad (3)$$

In practice equation (3) is not sufficient to obtain accurate results over a wide operating range as there are several other factors influencing the period of oscillation. Temperature changes lead to changes in the tube volume and the spring constant. The same holds for changes of the fluid pressure  $p$ . To compensate for these effects a polynomial expansion can be applied to the constants in the measurement equation.

$$A = \sum_j a_j T^j + \sum_j b_j p^j \quad (4)$$

$$B = \sum_j c_j T^j + \sum_j d_j p^j \quad (5)$$

For increased accuracy also mixed terms can be included. Combining equations (3) - (5) we obtain the fusion equation to compensate for the nuisance effects of temperature and pressure on the density reading.

$$\rho = f(\tau, T, p) \quad (6)$$

As the accuracy requirements are increased even further influences need to be compensated for. Fluid viscosity, as an example, is known to have a small effect on the period of oscillation (Krasser & Senn, 2007).

## 2.2 Multidimensional parameter estimation

In the error compensation case several auxiliary measurands act on individual scalar output quantities in a unidirectional way. A more complex fusion process can be introduced if a vector of output quantities  $x^T = (x_1 \ x_2 \ \dots)$  is introduced in equation (1). Then the input parameters can interact with all of the outputs simultaneously. This allows for the use of more flexible and powerful fusion methods. Figure 2 shows flow charts of the scalar and multidimensional data fusion approaches. The data fusion process  $f_{12}$  can range from a matrix in the linear case to arbitrary response surface methods, e.g. based on polynomial expansions.

The concentration measurement of ternary and multinary solutions is a typical problem in this class of fusion procedures. The density measurement introduced in section 2.1 is often used for concentration measurement of binary solutions. However, if there are more than two components only the sum of contributions of the individual components can be measured. The measurement problem may be resolved by fusing suitable methods for binary concentration measurement. As a representative application the measurement of extract and alcohol concentration in beer production is presented (Vasarhelyi, 1977). Beer is



basically a ternary mixture of water, extract and alcohol. So density measurement alone is not sufficient. The same is true for sound velocity and refractive index measurements, which are also classical methods for density and concentration determination (Hauptmann et al., 2002). If the response curves of two measured quantities to extract and alcohol variations are linearly independent in a certain range the problem can be solved with data fusion. Then both quantities can be uniquely determined from the primary measurements. The procedure is sketched in figure 3.

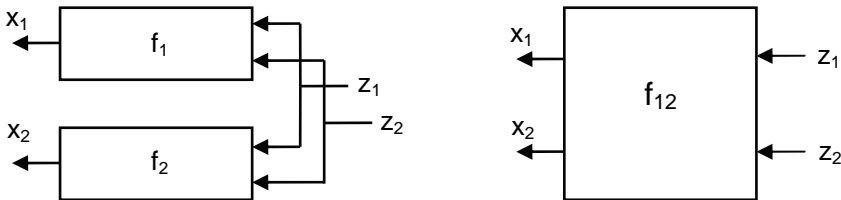


Fig. 2. Data fusion by means of unidirectional error compensation (left) and multi-dimensional parameter transformation (right).

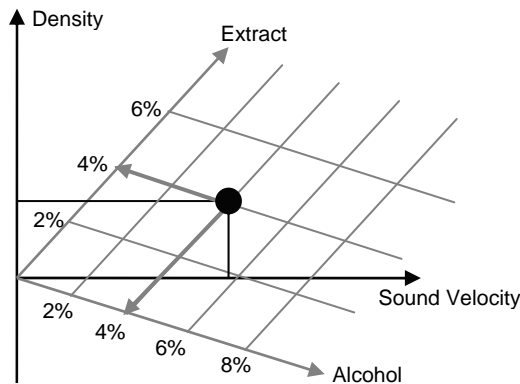


Fig. 3. Determination of extract and alcohol concentration in beer from density and sound velocity measurements.

The example qualitatively shows the extract and alcohol determination from density and sound velocity measurements. It can be seen that an extract change increases both sound velocity and density. On the contrary, a change of the alcohol concentration increases sound velocity and decreases density. This orthogonality allows for the unique determination of the target quantities. The inversion procedure corresponds to the transformation of a point in the Euclidean density/sound velocity space to a different, in general curvilinear, coordinate system.

### 3. Redundant measurement processes

#### 3.1 Sensor fault detection and isolation

One of the most prominent applications of data fusion in industrial process instrumentation is sensor fault detection and isolation (FDI). It is of fundamental importance in all safety-

critical processes where the reliability of sensor data needs to be assured. Typical examples are nuclear power plants and other assets processing dangerous materials or handling very high power. In such cases false sensor readings may cause expensive and harmful damage. The central step in FDI is the generation of residuals. Residuals are functions of sensor outputs and must rise in case of a sensor fault (Betta & Pietrosanto, 2000) (Dorr et al., 1997) (Simani, 2000). Faults can then be detected by comparing the residuals with thresholds that are suitably defined with respect to the normal sensor operation. Finally the faulty sensors are isolated by analyzing the different residuals.

Physical and analytical redundancies are the two possible types of redundancy necessary for the calculation of residuals. Physical redundancy means the use of several sensors measuring the same physical quantity in parallel. Analytical redundancy is concerned with the application of analytical models in order to produce estimates of sensor signals from other sources of data. It requires the precise knowledge of the underlying process. Typically applied in this category are stochastic filters and observers like the Kalman filter. The application study in section 3.2 treats a sensor system using an extended Kalman filter with integrated FDI functionality.

Physical redundancy is more general as it does not rely on such information. However, it is more expensive to implement as multiple sensors for every quantity to be measured need to be employed. Residuals for a redundant set of sensors are computed by comparing each sensor signal with an estimate of the true value of the measured physical quantity  $\zeta$ . A straightforward linear measurement and fault model can be used to relate the true value to the measurement  $z_j$  of sensor  $j$ .

$$z_j = \zeta + v_j + \delta_j \quad (7)$$

Each of the  $n$  sensor readings is corrupted with a random noise component  $v$ . The component  $\delta_j$  is considered the contribution of an abnormal degradation or fault of sensor  $j$ . The individual sensor fault contributions are assumed independent. The estimate of the true value  $\zeta$  is obtained from a weighted sum of the sensor readings. Constant weights are used if the random contributions  $v$  are uncorrelated white Gaussian noise with the same variance. There are many different possibilities to derive and analyze the residual vector  $r$  for the redundant instruments. The proper choice depends on the assumed fault mechanisms and effects. A good choice for the introduced linear fault model is the difference of the individual sensor values and the calculated mean of the respective remaining measurements.

$$r_j = z_j - \frac{1}{n-1} \sum_{k=1, k \neq j}^n z_k \quad (8)$$

In the fault-free case the expectation of the residuals is zero and the standard deviation depends on the individual sensor standard deviations and the number of sensors.

$$\sigma(r_j) = \sqrt{\frac{n}{n-1}} \sigma(z_j) \quad (9)$$

In the case of a single instrument error the residual of this sensor has a mean equal to the fault amplitude, while the other residuals show a mean that is lower by a factor  $1/(n-1)$ . With this information suitable thresholds for the detection of occurring faults can be easily established.

Other possibilities for the detection of faults include the calculation of statistical, spectral, and temporal characteristics of the residuals and artificial intelligence methods. The use of fuzzy logic allows for a flexible integration of different aspects of failure modes with empirical knowledge, for which analytical models are difficult to define (Park & Lee, 1993). Input quantities of the system like differences of sensor values and other characteristics are fuzzified using linguistic variables. An example is shown in figure 4.

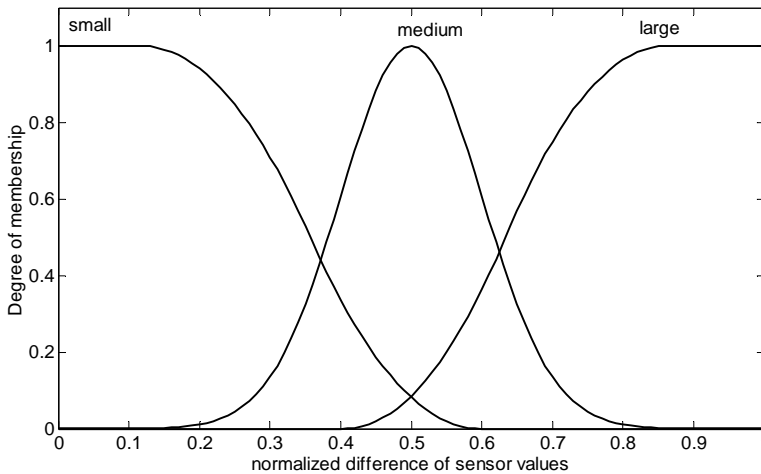


Fig. 4. Membership functions used for fuzzy-based fault detection and isolation.

The differences of sensor readings are normalized according to the standard uncertainties of the sensors. In the shown example three membership functions 'small', 'medium', and 'large', based on Gaussian and spline functions, are employed for the fuzzification (Steiner & Schweighofer, 2006). Further input variables can be elaborated from other observations related to error and fault occurrence. The actual fault model is contained in the rule base that is used for fuzzy inference. Rules are formulated as if-then relations.

$$\text{IF residual 1=small AND residual 2=small THEN} \\ \text{operation=normal} \quad (10)$$

The de-fuzzification stage of the Mamdani-type fuzzy system finally yields quantitative measures of the output fault parameters. Fuzzy systems have the advantage that they can be easily extended with more detailed application-specific process knowledge.

### 3.2 Integrated sensor arrays

With suitably designed sensor arrays and data fusion algorithms the tasks of error compensation and fault detection and isolation can be readily combined within a single instrument. Another potential feature is the highly accurate estimation of quantities of

interest through the combination of multiple low accuracy sensing elements. The achievable performance is illustrated in the following by means of an integrated smart capacitive sensor array for the determination of angular position and speed (Watzzenig et al., 2003) (Watzzenig & Steiner, 2004). The approach is based on the use of the extended Kalman filter (EKF), which is frequently used for data fusion in applications like target tracking. It offers a powerful framework also for industrial applications of data fusion. In contrast to the methods introduced in this chapter it is based on dynamical state space models and allows to monitor and exploit dynamical effects of the involved sensors and processes.

The capacitive angular position sensor consists of a rotor mounted coaxially between two stator plates. One stator plate corresponds to the transmitter. It is divided into 16 segments, the other stator contains the receiving electrode. The 16 segments of the transmitting electrode are electrically isolated from each other. The two stator plates are both bounded by an inner and an outer guarding ring connected to ground potential. The electrically conductive rotor is also grounded. It affects the coupling capacitances between the transmitter segments and the receiving electrode, dependent on its angular position. The used sinusoidal rotor shape yields a sinusoidal capacitance distribution. Segment driver and receiver electronics ensure that the received voltage signal amplitude is linearly dependent on the coupling capacitances. Figure 5 illustrates the axial view of a capacitive angular position sensor with approximately sinusoidal capacitance variation. In particular, a four-blade rotor in front of the transmitting electrode with 16 segments is shown. The receiving electrode is similar to the transmitting electrode, but without segmentation. Thus, the sensor array consists of 16 simple low resolution measurement channels which acquire data at different spatial orientations. The sensible combination of the channel data allows for the accurate estimation of angular position. The same principle can be applied to the measurement of other quantities like inclination angle, torque, liquid level, and flow. It can also be used with other physical sensing effects.

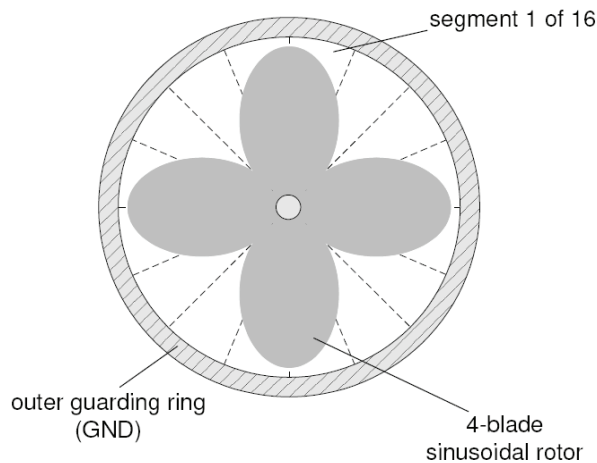


Fig. 5. Axial view of a capacitive angular position sensor. A four-blade rotor with a sinusoidal rotor shape is placed in front of the transmitting electrode, which is divided into 16 segments. The receiving electrode above the rotor is not shown.

The data fusion algorithm is based on the EKF. Therefore a discrete-time state space model of the sensor array is introduced. It is derived from a continuous second order system using a sampling interval  $T$ . The state vector  $x_k$  at time step  $k$  is composed of the angular position  $\varphi_k$  and the angular speed  $\omega_k$ .

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} T \\ 1 \end{bmatrix} w_{k-1} \quad (11)$$

The measurement equation (12) relates the state vector to the vector  $z_k$  containing the 16 segment readings through the nonlinear measurement function  $h$ . Due to the special shape of the rotor the segment voltages are basically phase-shifted sinusoids as a function of the angular position. Both state and measurement equations are corrupted by process and measurement noise sequences  $w$  and  $v$ , respectively. They are assumed to be uncorrelated white Gaussian sequences with covariance matrices  $Q$  and  $R$ .

$$z_k = h(x_k) + v_k \quad (12)$$

The EKF recursively estimates the process states based on the current values of the states and the state error covariance matrix  $P_k$ . The algorithm can be grouped in a prediction step and a correction step. First the states and covariance are projected to the next time step based on the process model. The projected values are indicated by the superscript ' $-$ '. The correction uses the current measurement to update the projected states. The central equation is the calculation of the Kalman gain matrix  $K_k$ . It is based on a linearization of the measurement equation using the Jacobian  $H_k$  of  $h$ .

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R)^{-1} \quad (13)$$

The gain matrix is then used to update the state estimate using the current measurement.

$$\hat{x}_k = \hat{x}_k^- + K_k [z_k - h(\hat{x}_k^-)] \quad (14)$$

The term in brackets, which is the difference between actual measurement and estimated measurement, is called innovation sequence. It is a sensitive indicator of differences between a fault-free model and the current system. If the measurements and the model are in agreement, it has a mean of zero. Thereby Kalman filtering can also conveniently be used for fault detection and isolation using analytical redundancy. For the present application the innovation sequence  $s_k$  can be used to compensate for occurring segment offsets. The measurement equation (12) can be extended by offset voltages  $\xi$  for the individual segments.

$$z_k = h(x_k) + v_k + \xi_k \quad (15)$$

The offset values can be estimated without increasing the size of the Kalman filter equation systems by directly integrating the innovation sequence.

$$\xi_{k+1} = \xi_k + \tau \cdot s_k \quad (16)$$

The choice of the integration time constant  $\tau$  determines the trade off between smoothness of the estimates and bandwidth of the offset compensation. The relation of the discrete-time EKF to the capacitive sensor system is illustrated in figure 6. The 16 measured segment voltages are used as inputs to the EKF. Based on the fusion of all signals the EKF is able to calculate accurate estimates of angular position and speed.

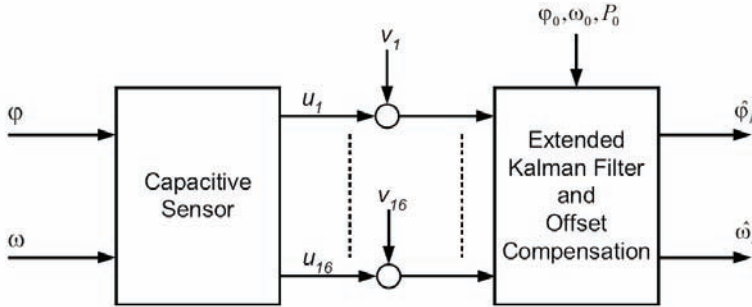


Fig. 6. Block diagram of the capacitive sensor with the extended Kalman Filtering with integrated offset estimation and compensation.

Figure 7 illustrates the performance of the EKF and the offset compensation approach. The first subfigure shows three of the measured sinusoidal voltages of adjacent segments. The second figure depicts the estimated segment offsets. There is a slight global offset at time step 0, which is compensated after about 10 samples. At time step 170 the voltage of segment 1, plotted bold, is stressed by an offset of 30% of the signal amplitude. The monitored innovation sequence is illustrated in subfigure 3. The sudden change in the amplitude of the sequence corresponding to segment 1 at time instant 170 (dashed vertical line) is obvious. The last chart shows the progression of the absolute angular position error. A further approach to perform error compensation for the capacitive sensor array that is capable of handling additional errors like line faults, short circuits, driver failures, and electromagnetic disturbances, is to use several parallel Kalman filters and integrate the estimates in an additional data fusion step. Decentralized Kalman filtering also drastically reduces the computational requirements for the signal processing of the sensor array. In the present case the 16 measurement signals can be pairwise used as inputs to eight parallel EKFs. The inputs are wired in such a way that a single EKF processes signals that have a phase shift of 90°. The block diagram of the whole sensor system including the final fusion stage is sketched in figure 8.

All filters operate on the same process model, but use different measurement equations. Assuming equivalent error covariances for all filters, the decentralized fusion can be done by averaging the eight Kalman Filter state estimates. This assumption holds as long as no segment fault occurs.

$$\hat{\phi}_k = \frac{1}{8} \sum_{j=1}^8 \hat{\phi}_k^{(j)} \tag{17}$$

By introducing fault detection with subsequent deactivation of certain filters with deficient state estimates the decentralized fusion can be applied for the general case. The fusion

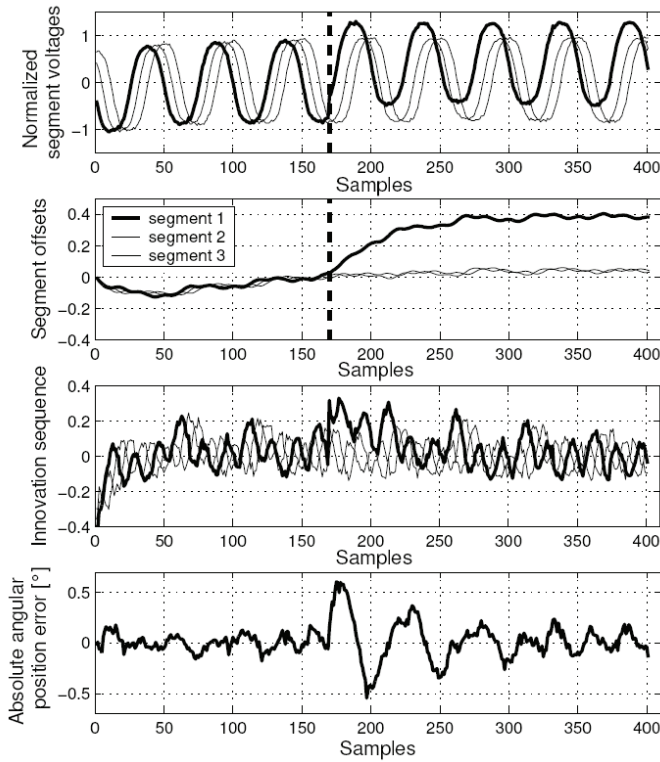


Fig. 7. Readings of the first three segments, segment offsets, monitored innovation sequence and curve of remaining angular position error. The signal of segment 1 is stressed by a DC offset at time step 170.

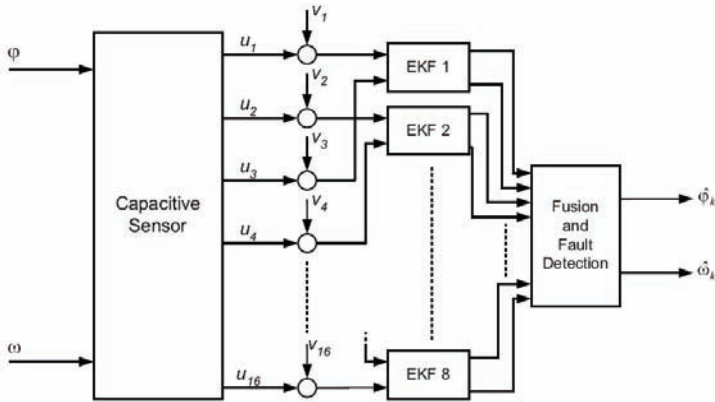


Fig. 8. Block diagram of the capacitive sensor with decentralized Kalman Filtering and subsequent data fusion with fault detection.

equation (11) does not consider the correlation of the single Kalman filter estimates due to the common process noise states. However, a comparison with the optimal fusion filter (Hashemipour et al., 1998) applied to the same problem shows only minor differences. The computational cost of the proposed averaging is very low compared to the optimal filter and numerical problems inherent in optimal fusion are avoided.

The availability of eight state estimates in parallel can be exploited for fault detection. The correlation between the different outputs and the final averaged estimate provides a confidence figure (an estimate for the variance) for each filter output.

$$V_k^{(j)} = \frac{1}{N} \sum_{j=1}^N \left( \hat{\phi}_k^{(j)} - \hat{\phi}_k \right)^2 \quad (18)$$

The last  $N$  measurements are used for the calculation of the mean square difference. In order to exclude a faulty filter some threshold level must be defined. This threshold level should be adaptive in a sense that only a segment that is significantly worse than the others is eliminated. A possible choice is the mean of the confidence figures with a tuning parameter  $\alpha$ .

$$V_k^{\max} = \frac{\alpha}{8} \sum_{j=1}^8 V_k^{(j)} \quad (19)$$

The performance of the decentralized filtering approach to data fusion with the capacitive sensor array is demonstrated with two failure modes; segment drift, and segment disturbance. In capacitive sensing a line break does not necessarily lead to a full breakdown of the segment signal, because the capacitive coupling over a break is still considerable. Since this coupling may depend on external influences such as temperature or vibrations, a noisy segment signal as shown in figure 9 may occur. The noise contamination of one signal instantaneously affects the position estimate of the corresponding filter. The confidence figure of this filter quickly rises above the threshold, so that the corrupted EKF is excluded from the calculation of the final estimate. It can be included again when the variance estimate is below the threshold again.

Another failure mode is demonstrated in figure 10. The offset for one segment continuously increases. This occurs in practical implementations when, e.g., water drops or dirt accumulate on a segment, because the high conductivity or permittivity of such a contamination lead to an amplification of the signal. Again, this error is quickly detected, long before even a very restrictive range checking algorithm would have the chance to detect this problem. Consequently the angular position estimate remains unaffected from the disturbance.

## 4. Ill-posed measurement processes

### 4.1 Industrial process tomography

In some measurement problems the available data is not sufficient to fully characterize the process at hand. This is often the case in distributed scenarios where spatially varying quantities that cannot be measured directly need to be resolved. For example, tomographic measurement techniques are able to provide two-dimensional or three-dimensional information about internal states of industrial processes. The knowledge of the internal



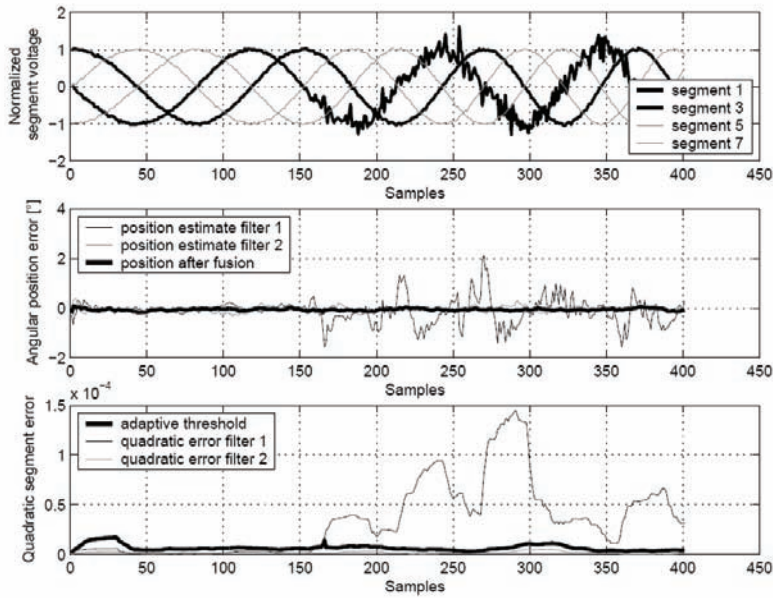


Fig. 9. Segment readings, output of selected filters, fused result, and adaptive threshold level in the case of a segment disturbance caused by a crack in the line.

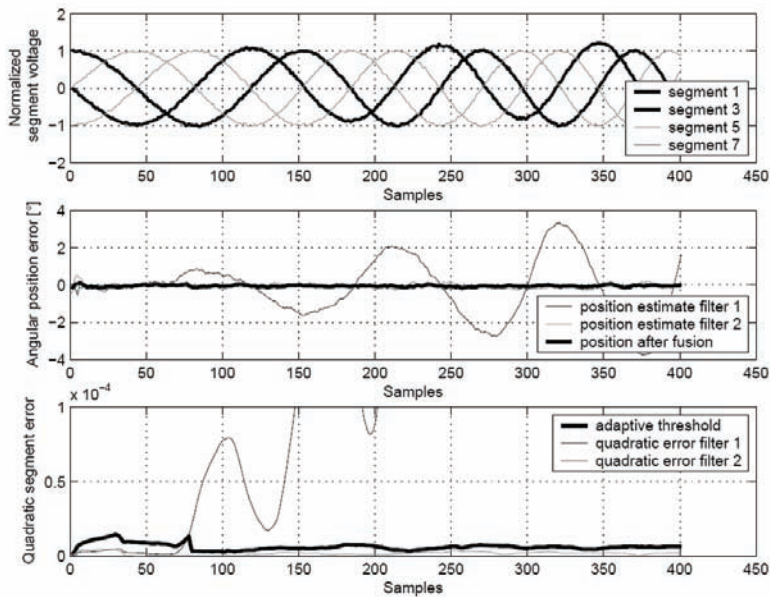


Fig. 10. Segment readings, output of selected filters, fused result and adaptive threshold level in presence of a segment offset drift.

behavior of such processes can be used for process design, prediction and process control in order to increase product quality and process efficiency. Depending on the application, different tomographic sensing modalities have been developed for industrial purposes, e.g. electrical capacitance tomography (ECT), electrical resistance tomography (ERT), ultrasonic reflection tomography (URT), positron emission tomography (PET) and X-Ray tomography (Scott & McCann, 2005) (Plaskowski et al., 1995). These techniques have in common that the spatially distributed parameters are reconstructed from a limited number of measurements. The sensors are usually distributed around the boundary of the problem domain. The fusion of this boundary data in order to obtain estimates of the spatial distributions of the quantities of interest is an ill-posed inverse problems. This implies that there is no unique solution to the problem. Ill-posed problems are very sensitive to noise and special measures need to be taken to obtain a stable meaningful solution (Kak & Slaney, 2001). This includes the choice of a suitable model of the measurement process and the incorporation of available prior knowledge through regularization. A multitude of reconstruction algorithms has been proposed, ranging from simple linear backprojection methods to model-based approaches based on nonlinear optimization methods and stochastic filtering methods like extended Kalman filters and particle filters.

Electrical capacitance tomography presents as a representative example. The objective in industrial ECT is to estimate the dielectric properties of heterogeneous mixtures or distinct transitions between occurring phases based on capacitance measurements between certain electrodes at the boundary of a closed container like a pipeline. Figure 11 illustrates the schematic of an ECT sensor based on the measurement of displacement currents (Wegleiter et al., 2005). The cross-section of a non-conducting pipe is used as measurement plane. 16 electrodes are evenly spaced around the circumference of the pipe. The setup is protected from electromagnetic interference by a grounded outer shield.

Every single electrode can be alternately used as a transmitter and a receiver. The front-end electronics of an electrode consists of a transmitting amplifier and an input stage comprising a current-to-voltage converter, a bandpass filter, and a high frequency peak rectifier. A single measurement frame consists of 16 projections, according to the 16 available transmitting electrodes. For one projection a specific electrode acts as transmitter while all the others sense the displacement current. A measurement frame consequently consists of 16 by 15 = 240 entries.

The reconstruction of the permittivity distribution within the pipe from the boundary measurements requires a mathematical model of the measurement process. This forward model establishes the functional mapping between the cross-sectional material distribution  $\varepsilon_r(x, y)$  and the measured displacement currents  $q$ . Under the assumption of non-conducting materials, negligible magnetic fields, and wave propagation effects it can be modelled as an electrostatic field problem in the interior of the screen. This leads to a generalized Laplace equation for the electric potential  $v$  (Watzenig et al., 2007b).

$$\nabla \cdot (\varepsilon_r \nabla v) = 0 \quad (20)$$

Dirichlet boundary conditions  $v = v_0$  at the transmitting electrode and  $v = 0$  at the sensing electrodes are prescribed. The charges at the electrodes are proportional to the displacement currents occurring with time-harmonic excitation. The electrode charges are determined by integration of the electric displacement along the electrode boundaries. The forward model

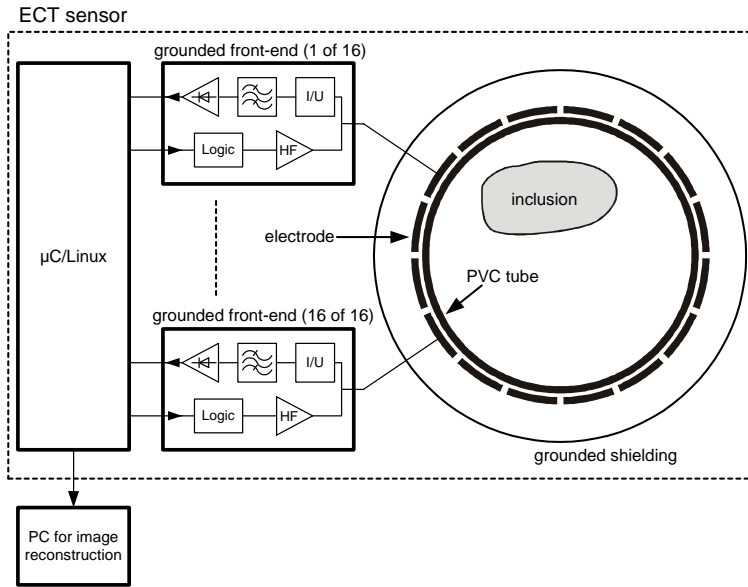


Fig. 11. Measurement configuration and schematic of a typical ECT sensor. The measurement electrodes are placed around the pipe containing the imaging domain. Every electrode features dedicated transmitting and receiving hardware. The acquired data is transferred to a PC where the signal processing is performed.

is numerically solved using the finite element method (FEM). The problem domain is thereby discretized into  $n$  triangular finite elements, where the permittivity is assumed constant within a single element. The discretization of equation (20) leads to a linear equation system for the vector  $v$  of the potentials of the finite element nodes.

$$K(\varepsilon_r)v = r \quad (21)$$

The stiffness matrix  $K$  reflects the geometry of the problem and the permittivities of the finite elements. Due to the sparsity of the stiffness matrix the equation system can be solved efficiently using specialized algorithms. However, in the context of industrial process tomography a compromise between the spatial resolution and accuracy of the finite element mesh and the computation time still has to be met. Figure 12 shows one quadrant of a typical finite element discretization. The domain is bounded by the outer screen. The interior contains the electrodes, the pipe and the imaging plane. The whole interior of the pipe is segmented into 316 elements, which is the number of unknown parameters to be reconstructed.

The unknown permittivity distribution can be estimated from the measured displacement currents by inverting the known measurement relation. A key issue associated with this inverse problem is its ill-posedness. This basically means that there is no unique solution and that it does not depend continuously on the data. The available reconstruction methods can be generally classified in non-iterative and iterative algorithms. Non-iterative methods assume a linear relationship between permittivities and displacement currents through the sensitivity matrix  $S$ .

$$q = S\varepsilon_r \tag{22}$$

This is an approximation since the actual relation is nonlinear. A linear reconstruction generates permittivity estimates from a pseudoinverse of the sensitivity matrix.

$$\hat{\varepsilon}_r = Dq \tag{23}$$

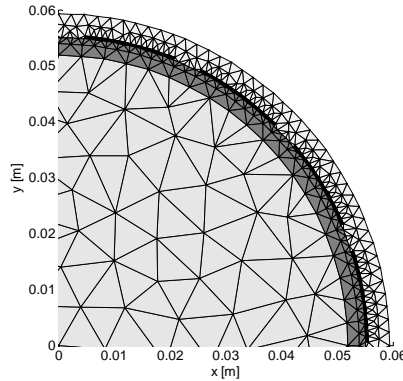


Fig. 12. Finite element discretization of one quadrant of the ECT problem domain. It consists of the electrodes (black), pipe (gray) and the imaging plane (light gray). The domain is bounded by the outer screen.

The pseudoinverse can be calculated in various ways. The simplest method is to use the transpose of the sensitivity matrix. More elaborate approaches like the offline iteration online reconstruction (OIOR) use an iterative precomputation.

More accurate results can be achieved with iterative methods that are based on the full nonlinear forward model.

$$q = f(\varepsilon_r) \tag{24}$$

In this case the inverse problem can be solved by minimizing a least squares cost functional. The minimization can be performed using the Gauss-Newton algorithm. A flow chart of the procedure starting from an initial guess for the permittivity distribution is sketched in figure 13. The minimization is terminated when the residual is below a predefined threshold.

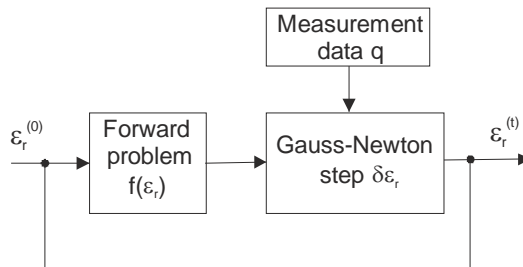


Fig. 13. Flow chart of the nonlinear least squares reconstruction for ECT.

$$C(\varepsilon_r) = \|q_m - f(\varepsilon_r)\|_2^2 + \alpha R(\varepsilon_r) \quad (25)$$

The first term on the right hand side of equation (25) is the sum of squared errors between measured displacement currents  $q_m$  and the simulated values. The minimization of this term alone would not yield sufficient results due to the ill-posedness of the inverse problem. Therefore a second term, the regularization term, has to be added in order to stabilize the solution. The relative weight of the two terms is controlled by the regularization parameter  $\alpha$ . The assumptions that are placed in this term introduce prior knowledge about the assumed material distributions and can take various forms. A popular choice in the absence of specialized knowledge is generalized Tikhonov regularization.

$$R(\varepsilon_r) = \|L\varepsilon_r\|_2^2 \quad (26)$$

The regularization matrix  $L$  is a discrete approximation of the Laplace operator, leading to high values of  $R$  for jumps between neighboring finite elements. Consequently, this choice leads to a smoothing of the reconstructed permittivity distribution.

Figure 14 illustrates an ECT sensor with a predefined two-phase material composition of gravel and air (left). The reconstructed cross-sectional material distribution based on the described least squares reconstruction method with Tikhonov regularization is shown on the right. The relative permittivity values are coded in gray scale.

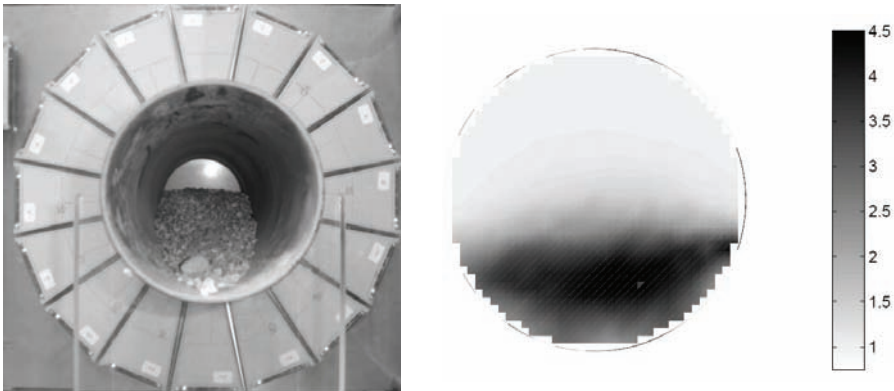


Fig. 14. ECT sensor partly filled with gravel (left) and the corresponding least squares reconstruction (right).

Depending on the industrial process at hand, models of the material distribution other than those described may replicate the situation more accurately. This allows to incorporate more detailed prior knowledge and helps to obtain accurate reconstruction results. Usually the materials involved in a process are known in advance. This sets constraints on the admissible parameter range (Steiner & Watzenig, 2008). When the distribution is piece-wise constant, like in discrete multi-phase flow, closed contour models would be appropriate. Boundaries between different materials are explicitly modelled, which introduces process-specific additional prior knowledge. Contour models can e.g. be based on polynomial splines, Fourier series expansions, and level set functions (Kortschak et al., 2007) (Watzenig

et al., 2007a). Closed Fourier contours in two dimensions can be obtained through parameterizations of the  $x$  and  $y$  coordinates with a period  $T=1$ .

$$x(t) = \frac{a_{x,0}}{2} + \frac{1}{2\pi} \sum_{j=1}^{N-1} [a_{x,j} \cos(2\pi j s) + a_{b,j} \sin(2\pi j s)] \quad (27)$$

The parameterization is similar for the other coordinate. The complexity of shapes that can be modelled can be increased by using more terms. The reconstruction can be performed by fitting the model parameters to the measurements, similar to the bulk model based on the finite element discretization.

Another sensing modality for industrial process tomography is ultrasound reflection tomography (URT). A common approach records reflections of transmitted ultrasonic waves at material boundaries. The measured travel times of reflected waves from many different directions can be used to reconstruct the locations and contours of material inhomogeneities. Also for URT there are different reconstruction approaches, from simple backprojection to model-based approaches utilizing specific contour models (Steiner et al., 2006). Results obtained with simple backprojection and a B-spline-based contour model and least squares minimization are illustrated in figure 15. The backprojection algorithm can be more generally applied to a wide range of problems. However, the results suffer from noise and blurring. If the process can be clearly characterized a proper model can lead to much better results, as demonstrated in the right subfigure.

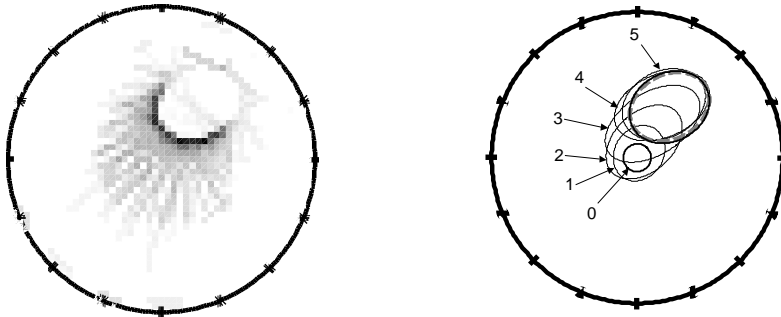


Fig. 15. URT reconstruction of a gas bubble in liquid. The left picture shows the results of a backprojection reconstruction. The right image shows the results obtained through using a contour model based on B-splines. The dashed ellipse shows the original bubble. The black curves show the evolution of the contour over the indicated iterations of the algorithm.

#### 4.2 Tomographic sensor fusion

Multimodality tomography systems combine two or more different sensing modalities. The rationale is to increase the reconstruction accuracy by data fusion of complementary data. If sensibly combined, the multimodal data may contain more information about the state of the imaging domain than could be achieved with a single sensing modality alone. Multimodal sensors therefore offer the possibility to monitor complex processes that cannot be dealt with by single modalities. They can be fruitfully applied to three-phase flow, like

oil-gas-water flow occurring in oil production, where a single modality only gives good contrast for two of the involved materials. ECT and URT, e.g., are well suited for data fusion. The main motivation is that the electrical modalities are sensitive to the bulk properties of materials while ultrasound is sensitive to phase boundaries, yielding the desired complementarity. So URT can give accurate boundary information not available with electrical tomography and electrical tomography can give information about connected volumes not achievable with URT alone (Steiner, 2006).

The most widely used principle of dual modality data fusion in terms of the data flow is sequential coupling of the modalities, which is illustrated in figure 16. After individual acquisition of the raw data with the two sensor arrays the inversion of the first modality is independently performed. The result is then used as additional input for the reconstruction of the second modality. This can be seen as providing a priori knowledge about the process state for the second stage. Another option for the combination of two modalities is parallel processing of the totality of raw data. However, this raises serious issues of data association (Steiner, 2007).

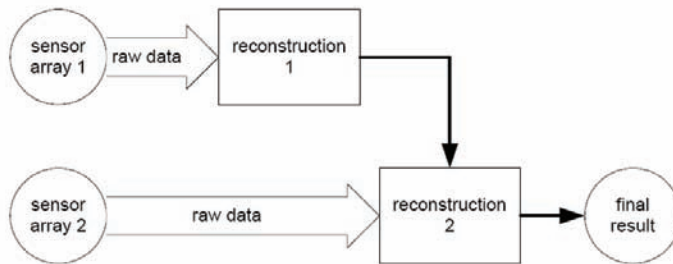


Fig. 16. Sensor fusion of two tomographic sensing modalities through sequential coupling of the raw data.

For the particular combination of sequential URT-ECT fusion the URT reconstruction can be used to deduce an outer approximation of the inclusion region containing the disperse phase of the material distribution, i.e. to uniquely assign parts of the imaging domain to the background region. If used in the subsequent ECT reconstruction, this information reduces the degrees of freedom of the inverse problem and thus the ill-posedness of the problem. Another approach is to use the incomplete information about object edges in the URT image to relax the smoothness assumption of ECT incorporated by the regularization term. This provides physically sound regularization as locations with a high probability of occurring material interfaces are allowed to show steeper permittivity gradients. A combination of these two sequential fusion approaches is compared to a single ECT reconstruction of two objects in figure 17. The left picture shows the ECT result, where the two objects cannot be resolved by the least squares reconstruction algorithm with smoothing Tikhonov regularization. ECT is least sensitive in the center of the imaging region, leading to a low spatial resolution. In contrast, URT offers highest sensitivity in the central region. The fusion reconstruction, due to the additional prior information from URT, distinguishes clearly between the two material inclusions.

An URT image as prior information is able to supply information about material boundaries. This can straightforwardly be included in contour-based reconstruction

algorithms for ECT. An example result is shown in figure 18. The capacitance data was reconstructed using a level set approach, where the contour is generated from equipotential curves of a two-dimensional function. A common regularization approach in this case is to penalize the arc length of the contour. The URT reconstruction can be added to the regularization term, forcing the level set contour towards the URT reconstruction while still allowing for deviations. The ECT reconstruction of two bubbles in figure 18 shows some blurring compared to the true object contours. The URT reconstruction, containing a sparse collection of points located just at the object boundaries, gives the extra information to allow for a close match of the true contours.

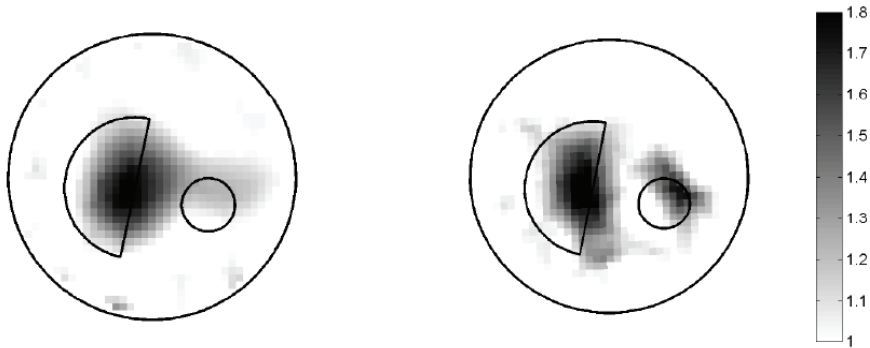


Fig. 17. Comparison of single ECT (left) and sequential URT-ECT fusion (right) reconstructions with a FEM-based spatial discretization of the imaging domain. The black curves show the true phase boundaries of the two material inclusions.

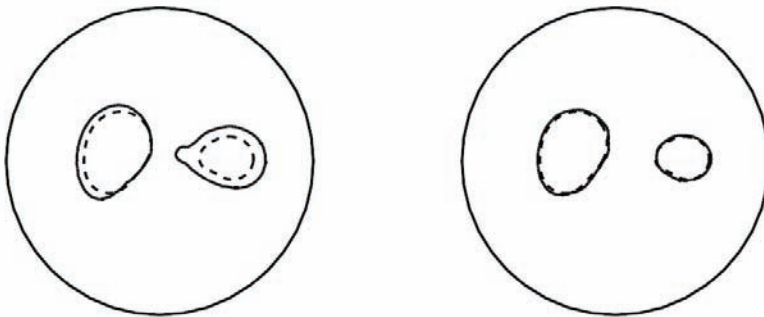


Fig. 18. Comparison of single ECT (left) and sequential URT-ECT fusion (right) reconstructions with a level set-based contour model. The dashed curves show the true phase boundaries of two material inclusions.

## 5. Conclusion

With increasing demands on the quality and efficiency of industrial processes as well as environmental and safety regulations, industrial process instrumentation is required to



acquire more accurate and comprehensive information. As the complexity of industrial processes increases, the same holds for the instrumentation. Data fusion techniques maximize the amount of useful information that can be extracted from raw sensor data.

This chapter gives an overview of model-based data fusion methods used in industrial process instrumentation with several typical application examples. They are intended to demonstrate the wide range of data fusion applications and are grouped in ascending complexity; from error compensation to multidimensional parameter estimation, sensor fault detection and isolation, integrated sensor arrays, industrial process tomography, and tomographic data fusion. It is expected that the consistent use of process models and data fusion methods will allow for an even more comprehensive and accurate characterization of industrial processes in the future.

## 6. Acknowledgement

This work was partially funded by the Austrian Science Fund (FWF) through the Translational Research Project L261-N04.

## 7. References

- Betta, G. & Pietrosanto A. (2000). Instrument fault detection and isolation: State of the art and new research trends. *IEEE Transactions on Instrumentation and Measurement*, Vol. 49, No. 1, 100-107
- Dorr R.; Kratz F.; Ragot J.; Loisy F. & Germain J.-L. (1997). Detection, isolation, and identification of sensor faults in nuclear power plants. *IEEE Transactions on Control Systems Technology*, Vol. 5, No. 1, 42-60
- Hall D. L. & Llinas J. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE*, Vol. 85, No. 1, 6-23
- Hall I. D.; McNab A. & Hayward G. (1999). Improved ultrasonic image generation through tomographic image fusion. *Ultrasonics*, Vol. 37, 433-443
- Hashemipour H. R.; Roy S. & Laub A. J. (1998). Decentralized structures for parallel Kalman filtering. *IEEE Transactions on Automatic Control*, Vol. 33, No. 1, 88-93
- Hauptmann P.; Hoppe N. & Puettmmer A. (2002). Application of ultrasonic sensors in the process industry. *Measurement Science & Technology*, Vol. 13, R73-R83
- Ihmels E. C.; Aufderhaar C.; Rarey J. & Gmehling J. (2000) Computer-controlled vibrating tube densimeter for liquid density measurement in a wide temperature and pressure range. *Chemical Engineering Technology*, Vol. 23, No. 5, 409-412
- Kak A. C. & Slaney M. (2001). *Principles of Computerized Tomographic Imaging*. Society of Industrial and Applied Mathematics, ISBN 139780898714944
- Kortschak B.; Wegleiter H. & Brandstaetter B. (2007). Formulation of cost functionals for different measurement principles in nonlinear capacitance tomography. *Measurement Science and Technology*, Vol. 18, 71-78
- Krasser E. & Senn H. (2007). Simultaneous measurements at U-tube density sensors in fundamental and harmonic oscillation. *Proceedings of the EUROCON 2007, Warsaw*, September 9-12, 551-555

- Laznickova R. & Huemer H. (1998). Investigations on the limits of uncertainty of gas density measurements with vibrating tube densimeters. *Measurement Science & Technology*, Vol. 9, 719-733
- Macii D.; Boni A.; De Cecco M. & Petri D. (2008). Multisensor data fusion. *IEEE Instrumentation & Measurement Magazine*, Vol. 11, No. 3, 24-33
- Park S. & Lee C. S. G. (1993). Fusion-based sensor fault detection. *Proceedings of the International Symposium on Intelligent Control*, Chicago, August, 156-161
- Plaskowski A.; Beck M. S.; Thorn R. & Dyakowski T. (1995). *Imaging Industrial Flows: Applications of Electrical Process Tomography*. Institute of Physics Publishing, ISBN 0750302968, Bristol, UK
- Ruhm K. H. (2007) Sensor fusion and data fusion - Mapping and reconstruction. *Measurement*, Vol. 40, No. 2, 145-157
- Scott D. M. & McCann H. (Ed.) (2005). *Process Imaging for Automatic Control*. Taylor & Francis, ISBN 0824759206
- Simani S.; Fantuzzi C. & Beghelli S. (2000). Diagnosis techniques for sensor faults of industrial processes. *IEEE Transactions on Control Systems Technology*, Vol. 8, No.5, 848-855
- Steiner G. (2006). Sequential fusion of ultrasound and electrical capacitance tomography. *International Journal of Information and System Sciences*, Vol. 2, No. 4, 484-497
- Steiner G. & Schweighofer B. (2006). Parameter identification for a complex lead-acid battery model by combining fuzzy control and stochastic optimization. *Inverse Problems in Science and Engineering*, Vol. 14, No. 6, 665-685
- Steiner G.; Podd F. J.W.; Brandner M. & Watzenig D. (2006) Iterative model-based image reconstruction for ultrasound process tomography. *Proceedings of the XVIII IMEKO World Congress*, Rio de Janeiro, September 17-22
- Steiner G. (2007). Application and data fusion of different sensor modalities in tomographic imaging. *Elektrotechnik & Informationstechnik*, Vol. 124, No. 7/8, 232-239
- Steiner G. & Watzenig D. (2008) Electrical capacitance tomography with physical bound constraints. *Proceedings of the SICE Annual Conference*, Tokyo, August 20-22, 1100-1105
- Tanner R. & Loh N. K. (2003). A taxonomy of multi-sensor fusion. *Journal of Manufacturing Systems*, Vol. 11, No. 5, 314-325
- Varshney P. K. (1997). Multisensor data fusion. *Electronics and Communication Engineering Journal*, Vol. 9, No. 6, 245-253
- Vasarhelyi G. (1977). New possibilities of calculating alcohol, extract and original gravity of beer using refraction and density measurements. *Brauwissenschaft*, Vol. 30, No. 8, 239-244
- Watzenig D.; Steiner G. & Zangl H. (2003). Capacitive sensor signal processing based on decentralized Kalman filtering. *Proceedings of the IEEE International Conference on Industrial Technology (ICIT)*, Maribor, December 10-12, 333-338
- Watzenig D. & Steiner G. (2004). Offset compensation for capacitive angular position sensors by evaluating the Kalman filter innovation sequence. *Proceedings of the IEEE International Conference on Industrial Technology (ICIT)*, Hammamet, December 8-10

- Watzenig D.; Brandner M. & Steiner G. (2007a). A particle filter approach for tomographic imaging based on different state-space representations. *Measurement Science and Technology*, Vol. 18, 30-40
- Watzenig D.; Steiner G.; Fuchs A. & Brasseur G. (2007b). On-line estimation of process parameters in multi-phase flows from measured electrical capacitance data. Proceedings of the 13th International Sensor Conference, Nuernberg, May 22-24, 223-228.
- Wegleiter H.; Fuchs A.; Holler G. & Kortschak B. (2005) Analysis of hardware concepts for electrical capacitance tomography applications. *Proceedings of the IEEE Conference on Sensors*, October 31 - November 3, Irvine, 688-691

# Multi-Sensor Data Fusion in Presence of Uncertainty and Inconsistency in Data

Manish Kumar and Devendra P. Garg  
*University of Cincinnati and Duke University*  
USA

## 1. Introduction

Modern control systems are characterized by increased complexity, flexibility, intelligence and enhanced ability to handle uncertainty. In order to incorporate the above features, the control systems need to possess significant capabilities such as perception, knowledge acquisition, learning, adaptability, and reasoning. The ability to perceive its environment forms a very important characteristic of such control systems. The revolutionary advancement in the field of sensor technology that has led to the development of superior sensing capabilities, and progress in computing and information processing, has made it possible to develop systems with enhanced perceptive abilities. Modern control systems generally employ multiple sensors to provide diverse, complementary as well as redundant information. These multiple sensor systems necessitate the development of sensor fusion algorithms that can combine information in a coherent and synergistic manner to yield a robust, accurate, and consistent description of quantities of interest in the environment.

There are several issues that arise when fusing information (Brooks & Iyengar, 1998, Hall & Llinas 2001) from multiple sources, some of which include data association, sensor uncertainty, and data management. The most fundamental of these issues arise from the inherent uncertainty in sensor measurement. The uncertainties in sensor measurement are caused not only by the device impreciseness and noise, but also manifest themselves from the ambiguities and inconsistencies present within the environment, and from an inability to distinguish between them. The strategies used to fuse data from multiple sensors should be capable of handling these uncertainties, and combining different types of information to obtain a consistent description of the environment. Some of the popular techniques for sensor fusion that are explored extensively in literature include Dempster-Shafer theory for evidential reasoning (Dempster, 1968, Shafer, 1976), fuzzy logic (Yager & Zadeh, 1991, Mahajan et. al., 2001), neural network (Garg & Kumar, 2007, Chin, 1994), Bayesian approach (Press, 1989, Berger, 1985), and statistical techniques (McKendall & Mintz, 1992) such as Kalman filter (Maybeck, 1979, Kalman, 1960, Sasiadek, 2002). All of these methods differ in the manner they attempt to model the uncertainties inherent in the sensor measurements.

Another possible uncertainty that arises in the sensor measurement process occurs when the measurements become corrupted and appear spurious in nature. Such corrupted measurements are difficult to model because they are not directly attributable to the inherent noise or other sources of uncertainty mentioned above. The cause of the corruption

may be due to events such as permanent sensor failures, short duration spike faults, or nascent (slowly developing) failures. Previous attempts at developing experimental models usually preclude the use of spurious measurements, and represent uncertainties attributable only to sensor noise and inherent limitations. Fusion techniques based on these incomplete models provide inaccurate estimation that can eventually result in potentially damaging action by the control system. Hence, a sensor validation scheme is necessary to identify spurious measurements so that they can be eliminated before the fusion process. There are several techniques reported in the literature for sensor validation and identification of inconsistent data. Many of them are limiting because they are based on specific failure models; these techniques can work well for events that occur due to known failure modes, however, they do not capture all possible failure events and often perform poorly when unmodeled failures occur. As a means to detect inconsistency, there should be either redundancy in the data, or some availability of *a priori* information. For example, in the case where *a priori* information is available, researchers have used the Nadaraya-Watson Estimator (Wellington et al., 2002) and *a priori* observations to validate sensor measurements. A few researchers have used a model based Kalman filter approach (Del Gobbo et al., 2001), while others have used covariance (Nicholson, 2004, Benaskeur, 2002), probability (Soika, 1997, Ibarguengoytia et al., 2001), fuzzy logic (Frolick et al., 2001), and neural network (Rizzo & Xibilia, 2002) based approaches. Some of these methods are explicit model-based, whereas others require tuning and training. In the general case, where *a priori* information is often not available, these approaches are typically deficient and can often lead to undesirable results.

This chapter presents a unified sensor fusion strategy based on a modified Bayesian approach that can take uncertainty of sensor data into account and automatically identify the inconsistency in sensor measurements so that the spurious measurements can be eliminated from the data fusion process. First, a novel strategy to accurately and adaptively represent uncertainty in sensor data in the form of probabilistic sensor model is developed. The strategy establishes the dependence of sensor's uncertainties on some of the environmental parameters or parameters of any feature extraction algorithm used in estimation based on sensor's outputs. In order to establish this dependence, the approach makes use of a neural network that is trained via an innovative technique that obtains training signal from a maximum likelihood estimator. The proposed method, then, adds a term to the commonly used Bayesian formulation. This term is an estimate of the probability that the data is not spurious, based upon the measured data and the unknown value of the true state. In fusing two measurements, it has the effect of increasing the variance of the posterior distribution when measurement from one of the sensors is inconsistent with respect to the other. The increase or decrease in variance can be estimated using the information theoretic measure "entropy". The proposed strategy was verified with the help of extensive computations performed on simulated data from three sensors. A comparison was made between two different fusion schemes: centralized fusion in which data obtained from all sensors were fused simultaneously, and a decentralized or sequential Bayesian scheme that proved useful for identifying and eliminating spurious data from the fusion process. The simulations verified that the proposed strategy was able to identify spurious sensor measurements and eliminate them from the fusion process, thus leading to a better overall estimate of the true state. The proposed strategy was also validated with the help of experiments performed using stereo vision cameras, one infra-red proximity sensor, and one

laser proximity sensor. The information from these three sensing sources was fused to obtain an occupancy profile of the robotic workspace.

This chapter is organized as follows: First, it introduces Bayesian technique for sensor fusion in Section 2. Next, in Section 3, it presents the neural network based sensor modeling technique. The proposed strategy for inconsistency detection and data fusion in Bayesian framework is presented in Section 4. Simulation studies to verify the proposed method for inconsistency detection is presented in Section 5. Section 6 presents the experimental validation carried out in a robotic workcell using three independent sensory sources. Finally, conclusions are presented in Section 7.

## 2. Bayesian technique for sensor fusion

Bayesian inference (Press, 1989, Berger, 1985) is a data fusion algorithm based on Bayes' theorem (Bayes, 1763) that calculates posterior probability distribution of  $n$ -dimensional state vector ' $X$ ', after the observation or measurement denoted by ' $Z$ ' has been made. The probabilistic information contained in  $Z$  about  $X$  is described by a probability density function (p.d.f.)  $p(Z | X)$ , known as likelihood function, or the sensor model, which is a sensor dependent objective function based on observation. The likelihood function relates the extent to which the *a posteriori* probability is subject to change, and is evaluated either via offline experiments or by utilizing the available information about the system. If the information about the state  $X$  is made available independently before any observation is made, then the likelihood function can be improved to provide more accurate results. Such *a priori* information about  $X$  can be encapsulated as the prior probability and is regarded as subjective because it is not based on observed data. Bayes' theorem provides the posterior conditional distribution of  $X = x$ , given  $Z = z$ , as

$$p(X = x | Z = z) = \frac{p(Z = z | X = x)P(X = x)}{\int p(Z = z | X = x)P(X = x)dx} = \frac{p(Z = z | X = x)P(X = x)}{P(Z = z)} \quad (1)$$

Since the denominator depends only on the measurement (the integration is carried out over all possible values of state), an intuitive estimation can be made by maximizing this posterior distribution, i.e., by maximizing the numerator of Equation (1). This is called *Maximum a posteriori* (or MAP) estimate, and is given by:

$$\hat{x}_{MAP} = \arg \max_x p(X = x | Z = z) = \arg \max_x p(Z = z | X = x)P(X = x) \quad (2)$$

The data from multiple sensors can be fused simultaneously (centralized fusion scheme), or sequentially (decentralized fusion). In this chapter, we will focus on decentralized fusion scheme in which, at any given instant, only two measurements or beliefs are fused. The recent interest in sensor networks, where distributed nodes possess capability to process information, has necessitated the development of algorithms to fuse information in a decentralized manner. The decentralized approach can be easily implemented in a distributed Bayesian framework where the posterior distribution obtained from old measurements becomes the prior distribution. Hence, the addition of new sensor measurement  $z_n$  to the belief obtained from  $n-1$  sensors ( $\bar{Z}_{1:n-1} = z_1, z_2, \dots, z_{n-1}$ ) can be achieved in an incremental manner via Equation (3):

$$p(X = x | \bar{Z}_{1..n} = z_1, z_2, \dots, z_n) = \frac{p(Z = z_n | X = x)p(X = x | \bar{Z}_{1..n-1} = z_1, z_2, \dots, z_{n-1})}{p(z_n)} \quad (3)$$

It may be noted that Equation (3) is valid only when measurements from different sensors are conditionally *independent*.

The Bayesian approach offers several advantages, including: appropriate representation of uncertainties using probability distributions; a well-defined mechanism to combine prior information with current sensor information; the existence of several machine learning algorithms to carry out the calculation of estimates and predictions; and thorough statistical characterization of the quantities of interest. Since the estimation takes into account available data from all previous as well as current experiments, the approach leads to a theoretically optimal solution. However, for most practical applications, a lack of priors or use of non-informative priors presents difficulties for Bayesian-based sensor fusion approaches. Assumptions regarding informative priors creates the possibility of unreasonable fusion between priors and likelihood functions. Moreover, most of the fusion strategies based on Bayesian approaches reported in the literature handle inconsistency in data rather poorly. In practical real-world scenarios, where data generated by sensors might be incomplete, incoherent or inconsistent, this approach might lead to erroneous results. Consequently, the inconsistency in data needs to be dealt with accordingly when Bayesian approaches are used.

### 3. Sensor modeling

Sensor modeling (Manyika & Durrant-Whyte, 1994, Kumar et al., 2005b, Kumar et al., 2006a) deals with developing an understanding of the nature of measurements provided by the sensor, the limitations of the sensor, and probabilistic understanding of the sensor performance in terms of the uncertainties. The information supplied by a sensor is usually modeled as a mean about a true value, with uncertainty due to noise represented by a variance that depends on both the measured quantities themselves and the operational parameters of the sensor. A probabilistic sensor model is particularly useful because it facilitates the determination of the statistical characteristics of the data obtained. This probabilistic model is usually expressed in the form of probability density function (p.d.f.)  $p(z|x)$  that captures the probability distribution of measurement by the sensor ( $z$ ) when the state of the measured quantity ( $x$ ) is known. This distribution is extremely sensor specific and can be experimentally determined (Durrant-Whyte, 1988).

#### 3.1 Estimation of sensor model parameters

*Maximum Likelihood (ML) method* is a procedure for finding the value of one or more parameters for a given statistical data which maximizes the known likelihood distribution. If Gaussian distribution is considered, the distribution representing the sensor model is given by:

$$p_{D_i}(z_i | \sigma, x_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left\{-\frac{(z_i - x_i)^2}{2\sigma^2}\right\}} \quad i=1,2,\dots,n \quad (4)$$

where the event  $D_i$  represents the data ( $z_i, x_i$ ) ( $x_i$  is the true value of state, and  $z_i$  is the corresponding sensor measurement), and  $\sigma$ , the standard deviation of the distribution, is the

parameter to be estimated. The likelihood function is the joint probability of the data given by:

$$L(\sigma) = \prod_{i=1}^n \frac{1}{\sigma\sqrt{2\pi}} e^{\left\{ \frac{-(z_i - x_i)^2}{2\sigma^2} \right\}} = \frac{1}{\sigma^n (2\pi)^{n/2}} e^{\left\{ \frac{-\sum_{i=1}^n (z_i - x_i)^2}{2\sigma^2} \right\}} \quad (5)$$

and the parameter  $\sigma$  can be estimated via *ML* method by maximizing  $L(\sigma)$  given by Equation (5). This results in a constant value of  $\sigma$  representing a rigid sensor model.

Most of the published research on sensor fusion is based on the development of rigid sensor models. In practice, the performance of sensors or any source of information depends upon several factors. These include, for example, the environmental conditions under which the measurements were made, and the performance of estimation/calibration algorithm. Establishing dependence of a sensor's performance on various parameters of environment and other signal/feature extraction algorithms is not a trivial task. Statistical techniques such as correlation analysis can be used to determine the manner in which these factors affect the sensor's output. Selecting the factors that can possibly affect the sensor output is difficult, and is mostly based on heuristics. Many feature extraction algorithms include goodness-of-fit function that can be investigated to observe the correlation with uncertainties in sensor output.

After the factor which affects the sensor's performance has been selected, the next challenge is to establish a functional correspondence between the factor and the uncertainty in the sensor's output. Statistical system identification, regression analysis, or any mapping algorithm can be investigated to establish the correspondence. It might be difficult, if not impossible, to obtain the mathematical relation, and in the absence of such mathematical relation, model-based statistical approach would be difficult to use. In this chapter, the universal approximation capabilities of neural networks have been used to establish this correspondence.

### 3.2 Proposed neural network based sensor modeling

A neural network (NN) (Rumelhart & McClelland, 1988, Haykin, 1998) is an information-processing paradigm inspired by the way in which the heavily interconnected, parallel structure of the human brain processes information. They are often effective for solving complex problems that do not have an analytical solution or for which an analytical solution is too difficult to be found. Currently, they are being applied in many real-world problems (Garg & Kumar, 2007). Three-layered NNs (i.e., one input layer, one output layer and one hidden layer), with hidden layer having sufficient nodes and a sigmoid transfer function, and linear transfer function in the input and output layers (Hornik, 1989) are considered to be universal approximators. In this chapter, a three-layered NN has been used to obtain a correspondence between the parameters of the distribution representing the sensor model and the factors which affect sensor's performance. The input to the neural network is the vector  $Q$  which represents vector of environmental or algorithmic factors that affect the sensor's performance. Output of the network is the vector of parameters of the distribution representing the sensor model. Hence, if the sensor model is represented by a Gaussian distribution, the parameter  $\sigma$  is given by:



$$\sigma = \text{NNET}(Q, W, B) \quad (6)$$

$W$  is the weight matrix, and  $B$  is the bias matrix. Back-propagation (BP), based on gradient descent technique, is a fairly popular method for training neural networks that establishes a particular set of weights obtained by adjusting them based on the errors between the actual and target output signals. For the neural network considered for the system in this research, however, the target data for  $\sigma$  is unknown, and cannot be obtained directly from experiments. Here, the neural network is trained in a novel manner from the signals obtained from *Maximum Likelihood* parameter estimation approach. Likelihood function that needs to be maximized is given by Equation (5), in which parameter  $\sigma$  is represented by a neural network function given by Equation (6). Hence, the likelihood function that needs to be maximized by choosing appropriate weights and biases of the neural network is given by:

$$L(W, B) = \frac{1}{[\text{NNET}(Q, W, B)]^n (2\pi)^{n/2}} e^{\left\{ \frac{-\sum_{i=1}^n (z_i - x_i)^2}{2[\text{NNET}(Q, W, B)]^2} \right\}} \quad (7)$$

The weights and biases can be calculated using the gradient descent method or via evolutionary strategies (Goldberg, 1989). The technique described above has been used to obtain models of infra-red proximity sensor and vision sensors in stereo configuration.

#### 4. Fusion of inconsistent multi-sensor data

Sensors often provide spurious data (Kumar et al., 2006b, 2007) which can be due to sensor failure or due to some inherent limitation of the sensor and/or some ambiguity in the environment. The Bayesian approach described in Section 2 is inadequate in handling this type of spurious data. The approach does not have a mechanism to identify when data from sensors is incorrect. The following paragraphs describe the use of a Bayesian-based approach for fusion of data from multiple sensors that takes into account measurement inconsistency.

While building a stochastic sensor model, generally spurious data are identified and eliminated. Hence these experimentally developed sensor models represent uncertainties arising only from sensor noise. If the event  $s = 0$  represents that the data obtained from a sensor is not spurious, then the sensor model developed in this manner actually represents the distribution  $p(Z = z | X = x, s = 0)$ . From Bayes' theorem, the probability that the data  $z_i$  measured by sensor 'i' is not spurious conditioned upon the actual state  $x$ , is given by:

$$[p(s = 0 | X = x, Z = z_i)]_i = \frac{[P(s = 0)]_i [p(Z = z_i | X = x, s = 0)]}{\sum_s [P(s)]_i [p(Z = z_i | X = x, s)]_i} \quad (8)$$

$[P(s = 0)]_i$  is the sensor specific prior probability that the data provided by Sensor  $i$  is not spurious. The denominator of the right hand side of the above equation is a summation carried over all possible values of  $s$  which are 0 and 1. The above equation can be re-written as:

$$[p(Z = z_i | X = x)]_i = \frac{[P(s=0)]_i [p(Z = z_i | X = x, s = 0)]_i}{[p(s=0 | X = x, Z = z_i)]_i} \quad (9)$$

To combine the sensor measurement from sensor  $n$  sequentially with the current belief obtained from sensors '1, 2... $n-1$ ', Equation (3) can be re-written as:

$$p(X = x | \bar{Z}_{1...n} = z_1, z_2, \dots, z_n) = \frac{[P(s=0)]_n p(Z = z_n | X = x, s = 0) p(X = x | \bar{Z}_{1...n-1} = z_1, z_2, \dots, z_{n-1})}{p(z_n) [p(s=0 | X = x, Z = z_n)]_n} \quad (10)$$

Hence, the introduction of term  $[p(s=0 | X = x, Z = z_n)]_n$  in the denominator has the effect of increasing the spread (variance) of the posterior if the new measurement has a greater probability of being spurious, and decreasing the spread of the posterior if the new measurement has a lower probability of being spurious. The increase or decrease in the spread of the posterior distribution can be easily ascertained by determining the information content given by the entropy of distribution obtained from the following equation:

$$H(X) = \int -p(X = x | Z = z_1, z_2, \dots, z_n) \log(p(X = x | Z = z_1, z_2, \dots, z_n)) dx \quad (11)$$

Entropy of a variable represents the uncertainty in that variable. A larger value of entropy implies more uncertainty and hence less information content. The fusion of a new measurement should always lead to a decrease in entropy, and fusion should always be done in order to reduce entropy. Based on increasing or decreasing the entropy of the posterior, this method can identify and eliminate spurious data from a sensor. It is noted that the prior probability  $[P(s=0)]_n$  has a constant value and simply acts as a constant weighting factor in Equation (10). This value does not influence the posterior distribution nor the *MAP* estimate of the state.

#### 4.1 Bayesian fusion without consideration of spuriousness in data (method 1)

If the spurious nature of the sensor data is not considered, and the models of the ' $n$ ' sensors are given by the following Gaussian likelihood function:

$$p(Z = z_k | X = x) = \frac{1}{\sigma_k \sqrt{2\pi}} e^{\left\{ \frac{-(x-z_k)^2}{2\sigma_k^2} \right\}} \quad k=1,2,\dots,n \quad (12)$$

then, from Bayes' Theorem the fused *MAP* estimate is given by:

$$\hat{x}_{MAP} = \arg \max_x [p(Z = z_1 | X = x) p(Z = z_2 | X = x) \dots p(Z = z_n | X = x)] \quad (13)$$

#### 4.2 Bayesian fusion with consideration of spuriousness in data (method 2)

If the spurious nature of the sensor data is considered, then the Gaussian sensor model represented by distribution  $p(Z = z | X = x, s = 0)$  is given by:

$$[p(Z = z | X = x, s = 0)]_k = \frac{1}{\sigma_k \sqrt{2\pi}} e^{\left\{ \frac{-(x-z_k)^2}{2\sigma_k^2} \right\}}_{k=1,2,\dots,n} \quad (14)$$

The probability that the measurement from Sensor 'k' is not spurious given the true state 'x' and measurement 'z<sub>k</sub>', is assumed to be represented by the following equation:

$$[p(s = 0 | X = x, Z = z_k)]_k = e^{\left\{ \frac{-(x-z_k)^2}{a_k^2} \right\}} \quad (15)$$

An advantage of choosing the above formulation for representing the probability is that the probability is 1 when measurement 'z<sub>k</sub>' is equal to the true state 'x', and decreases when the measured value moves away from the true state. The rate at which the probability decreases when the measured value moves away from the true estimate depends upon the parameter 'a<sub>k</sub>'. The value of the parameter is dependent on the variances of the sensor models and the distance between the output of sensor k with respect to other sensors.

In the decentralized or sequential fusion scheme, measurements from only two sources are fused at once. The belief resulting from the fusion of two sensors is then fused with the next sensor, and the process continues henceforth. Fusion of two sensors 'k' and 'k+1' using Equation (10) yields:

$$p(X = x | Z = z_k, z_{k+1}) = \frac{[P(s = 0)]_k [p(Z = z_k | X = x, s = 0)]_k}{[p(s = 0 | X = x, Z = z_k)]_k} \quad (16)$$

$$\times \frac{[P(s = 0)]_{k+1} [p(Z = z_{k+1} | X = x, s = 0)]_{k+1}}{[p(s = 0 | X = x, Z = z_{k+1})]_{k+1}} \times \frac{P(X = x)}{P(Z = z_k, z_{k+1})}$$

The value of parameter 'a<sub>k</sub>' in Equation (15) is assumed to be given by:

$$a_k^2 = \frac{b_k^2}{(z_k - z_{k+1})^2} \quad (17)$$

which leads to

$$p(X = x | Z = z_k, z_{k+1}) = \frac{P(X = x)}{P(Z = z_k, z_{k+1})} \times [P(s = 0)]_k \frac{1}{\sigma_k \sqrt{2\pi}} e^{-\frac{(x-z_k)^2}{2\sigma_k^2} \left\{ \frac{1}{2\sigma_k^2} \frac{(z_k - z_{k+1})^2}{b_k^2} \right\}} \quad (18)$$

$$\times [P(s = 0)]_{k+1} \frac{1}{\sigma_{k+1} \sqrt{2\pi}} e^{-\frac{(x-z_{k+1})^2}{2\sigma_{k+1}^2} \left\{ \frac{1}{2\sigma_{k+1}^2} \frac{(z_k - z_{k+1})^2}{b_{k+1}^2} \right\}}$$

The value of parameter 'b<sub>k</sub>' is chosen to satisfy the following inequality:

$$b_k^2 \geq 2\sigma_k^2 (z_k - z_{k+1})^2 \quad (19)$$

Satisfaction of this inequality ensures that the posterior distribution in Equation (18) remains Gaussian and hence has a single peak. The parameter value should be chosen based on

maximum expected difference (represented by 'm') between the sensor readings so that inequality (19) is always satisfied. Hence,

$$b_k^2 = 2\sigma_k^2 m^2 \quad (20)$$

Substituting Equation (20) in Equation (18) gives:

$$p(X = x | Z = z_k, z_{k+1}) = \frac{P(X = x)}{P(Z = z_k, z_{k+1})} \times [P(s = 0)]_k \frac{1}{\sigma_k \sqrt{2\pi}} e^{-\frac{(x-z_k)^2}{2\sigma_k^2 \left\{ \frac{m^2}{m^2 - (z_k - z_{k+1})^2} \right\}}} \times [P(s = 0)]_{k+1} \frac{1}{\sigma_{k+1} \sqrt{2\pi}} e^{-\frac{(x-z_{k+1})^2}{2\sigma_{k+1}^2 \left\{ \frac{m^2}{m^2 - (z_k - z_{k+1})^2} \right\}}} \quad (21)$$

It is apparent that the entire process has the effect of increasing the value of the variance of individual distribution by a factor of  $\left\{ \frac{m^2}{m^2 - (z_k - z_{k+1})^2} \right\}$ . Larger differences in sensor measurement imply that the variance increases by a bigger factor. Depending on the squared difference in measurements from the two sensors, the variance of the posterior distribution may increase or decrease as compared to the variance of individual Gaussian distributions representing the sensor models. Therefore, the strategy is capable of determining if fusion of the two measurements would lead to an increase or decrease of the variance of the posterior distribution. In information theoretic terms, the strategy is capable of determining if the fusion leads to an increase in information content (or entropy given by Equation (11)) or not. Based on increasing or decreasing of entropy in the posterior, a decision can be made whether to fuse those two sensors or not. This approach provides an opportunity to eliminate sensor measurements that are spurious and fuse measurements from only those sensors that are consistent, ensuring an increase in information content after fusion.

## 5. Simulation results

A simulation study was carried out to validate the effectiveness of the proposed strategy in identifying inconsistent data while fusing data from three sensors. A comparative analysis was performed to study the efficiency with which the two methods (described in Section 4) were able to handle inconsistency in data. The following parameters were assumed in the simulation:

Sensor 1:  $[P(s = 0)]_1 = 0.90$  and  $\sigma_1 = 3$

Sensor 2:  $[P(s = 0)]_2 = 0.98$  and  $\sigma_2 = 2$

Sensor 3:  $[P(s = 0)]_3 = 0.94$  and  $\sigma_3 = 2.5$

True value of state:  $x = 20$

Simulation data was generated so that Sensor 1 provided 90% of the time normally distributed random data with a mean value of 20 and variance 9. It provided incorrect data

10% of the time which was uniformly distributed outside the Gaussian distribution. Sensor 2 provided 98% of the time normally distributed random data with a mean value of 20 and variance 4, and 2% of the time it provided incorrect data. Similarly, Sensor 3 provided 94% of the time normally distributed random data with a mean value of 20 and variance 6.25, and 6% of the time it provided incorrect data. It may be noted here that the values for  $[P(s = 0)]_k$  have been assumed simply for the purpose of generating simulated data. These are not used in the fusion algorithm. Since these values are constants, they do not have any effect on the posterior distribution or the *MAP* estimate.

Figure 1.(a) illustrates a case when all of the three sensors are in agreement, and measurement from none of the sensors is inconsistent with the rest. It can be seen that posterior distributions obtained from both methods coincide resulting in the same value of *MAP* estimate. In Figure 1.(b), measurement from Sensor 1 is in disagreement from the other two sensors. Method 1, which is a simple Bayesian fusion and does not take into account inconsistency of data, results in the weighted average of the three measurements. Method 2 identifies the sensor which provides spurious measurements and eliminates that from the fusion process. Hence, it simply considers measurements from Sensors 2 and 3, and fuses

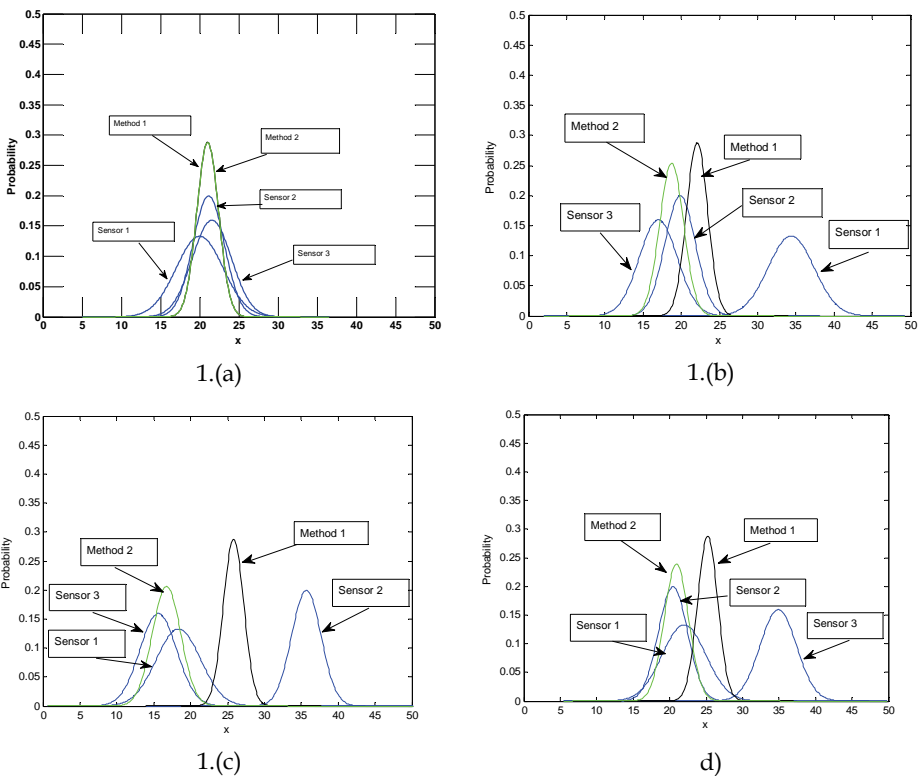


Fig. 1. Fusion of Three Sensors 1.a: All Sensors in Agreement, 1.b: Sensor 1 in Disagreement, 1.c: Sensor 2 in Disagreement, 1.d: Sensor 3 in Disagreement

them appropriately using Equation (21). In a similar manner, Figure 1.(c) and Figure 1.(d) respectively show that measurements from Sensor 2 and Sensor 3 are spurious. The figures show the efficiency with which Method 2 identifies and eliminates spurious measurements, and results in better estimates (closer to the true value) of the variable.

Ten thousand (10,000) data points were generated in the manner described above and fusion was carried out using both methods. The mean value of the sum of squared error (MSE) between the fused value and true value for all ten thousand data points was computed. The values of MSE were found as 6.94 for Method 1 and 5.50 for Method 2. Hence, Method 2 was able to reduce the mean square error by approximately 21% when compared to Method 1.

## 6. Experimental results

The theories developed in Sections 3 and 4 were validated with the help of experiments performed in the Robotics and Manufacturing Automation (RAMA) Laboratory at Duke University. The objective of the experiment was to obtain a three-dimensional occupancy profile of the robotic workspace using three independent sensory sources: stereo vision, an infra-red proximity sensor, and a laser proximity sensor. This section provides in detail first the sensor modeling process for the three sensory sources, and then the experiments for fusing data from them.

### 6.1 Sensor modeling

**Stereo Vision:** One of the most important components of stereo vision algorithm is stereo matching (Garg & Kumar, 2003) which involves finding out the location of the point in right image plane corresponding to a point in the left image plane. The relative displacement of these two points, called disparity, is used to estimate the three-dimensional position of the point. The accuracy with which stereo vision sensors are able to specify three-dimensional positional information about a point depends on how precisely the stereo vision algorithm is able to find the match of the point. The correlation score (Zhang et al, 1995) of the matched points, which measures the correlation between two template windows from left and right images, is a measure of “goodness-of-match” of the two points. The score ranges from -1 to +1, -1 representing not similar at all, and +1 representing most similar. The sensor modeling technique formulated in Section 3 has been used to develop a model for the stereo vision sensors that could take into account the effect of performance of the stereo matching algorithm on the uncertainty in sensor’s output.

An experiment was carried out in the RAMA Laboratory, wherein a set of fifty data points was obtained. The data set consisted of 3-D location of point in world coordinate system obtained via stereo vision sensors (via transformation as discussed in reference (Kumar & Garg, 2004)), correlation score for that point, and the actual 3-D location of the point in world coordinate frame.

The strategy presented in Section 3.2 was used to develop a Gaussian model of the sensor. In this model the standard deviation of the distribution, which represents the uncertainty of the data, is dependent on the correlation score for the specific point. This dependence was modeled with the help of a neural network with five nodes in the hidden layer. This neural network takes correlation score as input, and outputs the value of standard deviation ( $\sigma$ ) for that particular correlation score. The neural network was trained via the process based on the Maximum Likelihood technique as presented in Section 3.2. Genetic Algorithm (GA) was used to maximize the likelihood function given by Equation (7). Though

computationally intensive as compared to back-propagation based methods, GAs provide globally optimum results. The likelihood function given by Equation (7) was calculated by obtaining the data set representing the actual 3D location of points and the corresponding measurements from the stereo vision. Figure 2 shows the graph of standard deviation  $\sigma$  of the probability distribution function representing the model of stereo vision sensor as obtained by the neural network plotted against the correlation score of stereo matched points. As illustrated in Figure 2, the sensor model obtained from this approach separately for X, Y, and Z directions showed the intuitive trend that as the correlation score increases, i.e., as the stereo match gets better, the standard deviation decreases. Smaller value of standard deviations implies that the positional information obtained from stereo vision is less uncertain, and hence the degree of belief in the sensor output is more.

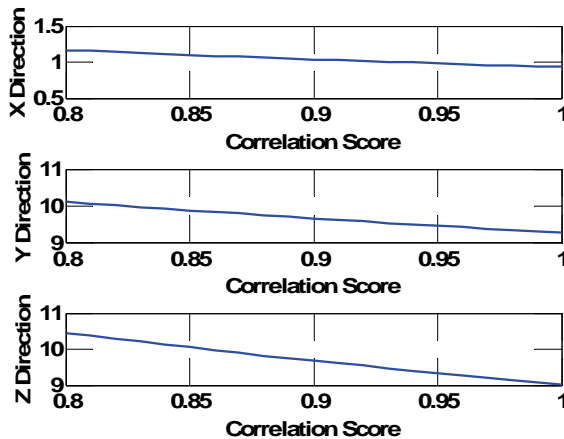


Fig. 2. Stereo Vision Sensor Model: Variation of Standard Deviation of Sensor Model in X, Y, Z Directions with respect to the Correlation Score of Stereo Matched Points

**Infra-Red (IR) Proximity Sensor:** The output of the IR proximity sensor is an analog voltage which is indicative of the distance of the object detected by the sensor. From the test data obtained from experiments, it was seen that the uncertainty in data increases when the distance to the object increases. Since the output of the sensor is indicative of the distance, sensor modeling process tries to capture the relationship between sensor's uncertainties and sensor output.

In the laboratory experiments, the Infra-Red sensor was mounted on the wrist of the robot so that it looked vertically down (negative Z direction in world coordinate frame). The IR sensor provided the information about the distance to the nearest object detected directly in front of the object. Information about the position of end effector was obtained from the encoders of the robot. Hence, IR sensor can be effectively used in conjunction with robot encoders to provide 3-D information about any object. Similar to vision sensor, in this research, the model of the IR sensor was obtained separately for all three X, Y, and Z directions based on the method described in Section 3.2. The variation of standard deviation of the Gaussian sensor model obtained from this approach, as illustrated by Figure 3, showed a decrease when the sensor output increased which means that when the distance to

the object decreases (i.e. sensor's output is larger) the standard deviation becomes smaller, and the sensor's measurement becomes less uncertain.

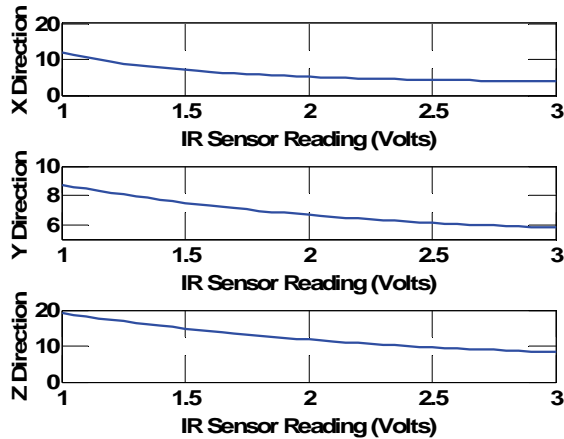


Fig. 3. Infra-Red Proximity Sensor Model: Variation of Standard Deviation of Sensor Model in X, Y, and Z Directions with respect to the Sensor Output

**Laser Proximity Sensor:** Similar to IR proximity sensor, the output of laser sensor is indicative distance to the detected object. Sensor modeling for laser proximity sensor was done in a similar manner as the IR proximity sensor. The variation of standard deviation of the Gaussian sensor model obtained from this approach, as illustrated by Figure 4, showed a flat curve which means that the uncertainty in sensor measurement remained indifferent to distance to the detected object. In practice, the laser proximity sensor was very accurate, and the uncertainty in sensor measurement was not dependent on the distance.

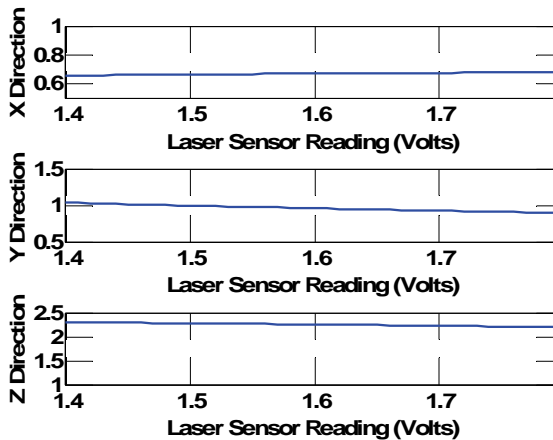


Fig. 4. Laser Proximity Sensor Model: Variation of Standard Deviation of Sensor Model in X, Y, and Z Directions with respect to the Laser Sensor Output



## 6.2 Sensor fusion for 3D modeling of workspace

The model of workspace was obtained in an occupancy grid framework. The occupancy grid (Elfes, 1992, Kumar et al, 2005a) is a multi-dimensional field (usually of dimension two or three) where each cell (or unit of the grid) stores or represents the probabilistic estimate of the state of spatial occupancy. Occupancy grids are one of the most common low-level models of an environment, which provide an excellent framework for robust fusion of uncertain and noisy data. If the state variable (occupancy, in this case) associated with a cell,  $C_i$ , is denoted by  $s(C_i)$ , then the occupancy probability  $P[s(C_i)]$  represents the probabilistic estimate of occupancy of that particular cell. If  $P[s(C_i) = occ] \approx 0$ , then the cell is assumed to be empty, while, if  $P[s(C_i) = occ] \approx 1$ , then the cell is assumed to be occupied. If a single sensor is used to obtain the occupancy grid, Bayes' Theorem can be used in the following manner to determine the state of the cell:

$$P[s(C_i) = occ | z] = \frac{p[z | s(C_i) = occ]P[s(C_i) = occ]}{\sum_{s(C_i)} p[z | s(C_i)]P[s(C_i)]} \quad (22)$$

where  $z$  is the sensor measurement. The probability density function (p.d.f.)  $p[z | s(C_i) = occ]$  is dependent on the sensor characteristics and is called the sensor model. The probability  $P[s(C_i) = occ]$  is called prior probability mass function and specifies the information made available prior to any observation.

Occupancy grids were obtained individually for stereo vision, infra-red, and laser proximity sensors, and then the individual grids were fused using two techniques: i) Simple Bayesian Fusion, and ii) Sequential Bayesian Fusion with Proposed Inconsistency Detection and Elimination Strategy. The details of the process for obtaining occupancy grids and sensor fusion are explained in reference (Kumar et al, 2005a).

In the experiment a cylindrical object was placed on the robot's work-table. Figure 5 shows the images of the work-table obtained from the stereo cameras. Figure 6.a shows the actual occupancy grid of the workspace. This was obtained based on the geometric dimensions of the object and its location in the workspace. For the occupancy grid developed in this research, each grid is of size 5mm X 5mm X 5mm. Figures 6.b, 6.c, and 6.d show the occupancy grids independently obtained from stereo vision, IR proximity sensor, and laser proximity sensor respectively. Figure 6.e shows the occupancy grid obtained from simple Bayesian approach, and Figure 6.f shows the occupancy grid obtained from the Bayesian approach that utilizes the inconsistency detection and elimination technique proposed earlier.

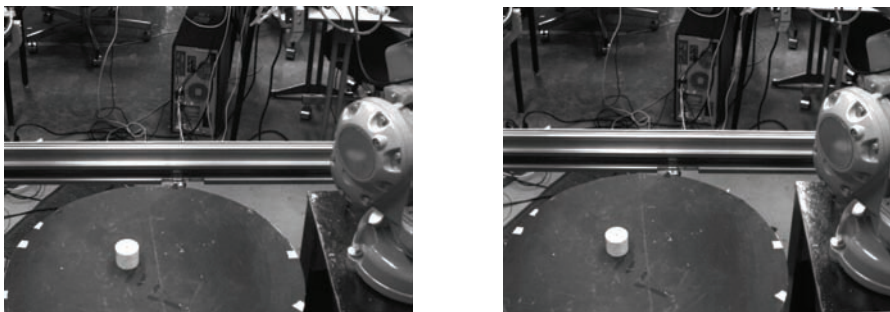


Fig. 5. Images of the Worktable Obtained from the Left and the Right Cameras

Stereo Vision	IR Proximity	Laser Proximity	Simple Bayesian	Bayesian with Proposed Inconsistency Detection
1279	1062	399	459	384

Table 1. Error Associated with Occupancy Grids Obtained from Fusion Process

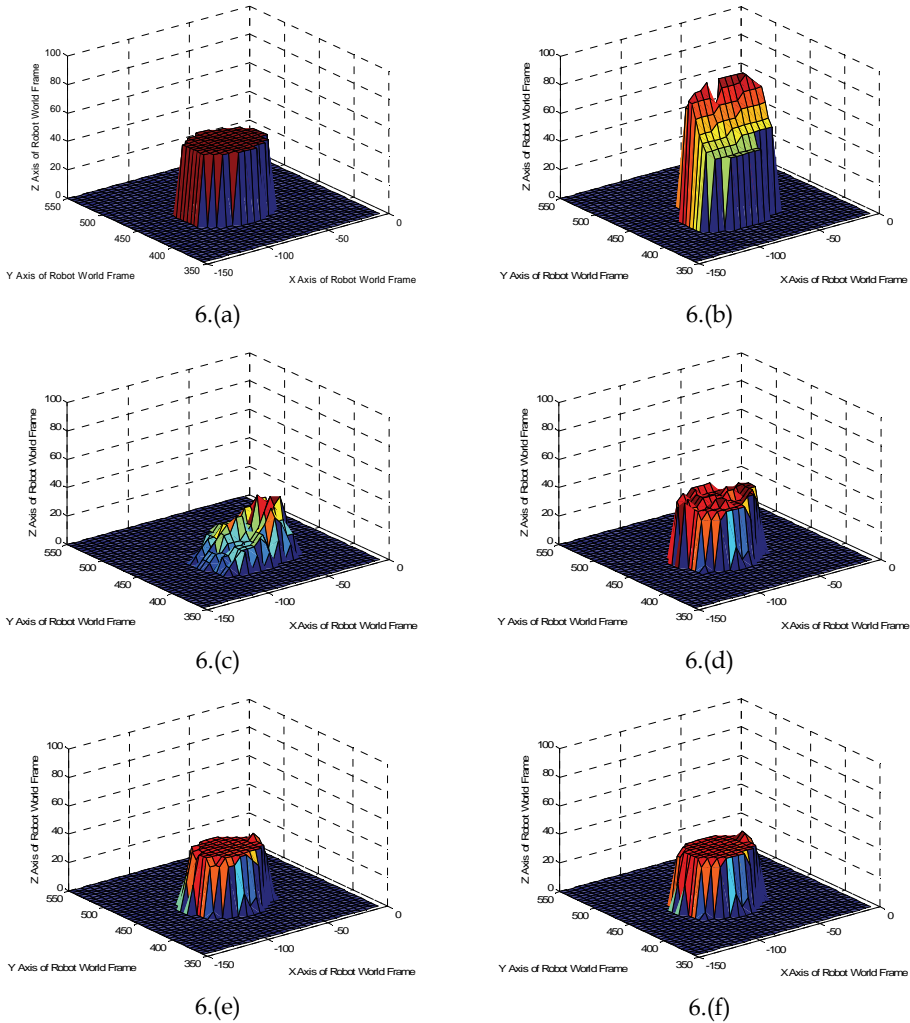


Fig. 6. Occupancy Grids a) Actual Grid, b) Grid Obtained from Stereo Vision, c) Grid Obtained from IR Proximity Sensor, d) Grid Obtained from Laser Proximity Sensor, e) Fused Grid (Simple Bayesian Approach, Method 1), and f) Fused Grid (Proposed Bayesian Fusion with Inconsistency Detection and Elimination, Method 2)

To facilitate a comparison of the performance of the fusion process via different algorithms, a measure of error was formulated which is given by the following equation:

$$Error = \sum_{C_i} [ |s(C_i)_{actual} - |s(C_i)_{sensor} | ]^2 \quad (23)$$

where  $|s(C_i)_{actual}$  is the actual state of the cell, and  $|s(C_i)_{sensor}$  is the state of the cell obtained from the sensor and/or fusion process. The state of the cell is either 1 (for occupied) or 0 (for empty). Table I provides the error value associated with the occupancy grid obtained from the fusion process described above. The table compares the error value obtained via the two approaches. The first approach is based on the simple Bayesian fusion scheme, and the second approach is based on the proposed Bayesian fusion scheme embedded with the mechanism for inconsistency detection and elimination.

From the figures as well as from the table of results, it is evident that the proposed fusion scheme based on Bayesian approach with inbuilt mechanism to identify and eliminate spurious/inconsistent measurement presented in this chapter has been able to reduce the uncertainty inherent in individual sensors. The proposed method has been able to reduce the error by approximately 70% as compared to stereo vision, 64% as compared to IR proximity sensor, and 4% as compared to laser proximity sensor. On the other hand, simple Bayesian technique was able to reduce the error by approximately 64% as compared to stereo vision and by 56% as compared to IR proximity sensor. The technique based on simple Bayesian approach led to an increase in error by approximately 15% as compared to laser proximity sensor. The increase in error demonstrates the fact that it is not necessary that incorporation of additional sensor data will lead to improved accuracy of estimation. This is particularly more evident in cases when the accuracy of measurements from sensors differs by a large amount. In this case, the measurements from laser proximity are far more accurate (as seen from sensor models) than measurements from the stereo vision or IR proximity sensor, and fusion of measurements from the laser with stereo vision and IR proximity leads to an increase in error. However, the proposed technique has an inbuilt mechanism to determine if the fusion process leads to an increase in the information content, and, in this way was able to eliminate inconsistent data and improve the overall accuracy of the fusion process. Of the 24000 points (or cells) where the fusion of data from three sensors occurred (fusion occurred at 30x40x20 cells of the occupancy grid), the proposed technique detected 393 points where data from IR sensor were inconsistent and 1028 points where data from stereo vision were inconsistent. None of the data from laser sensor were detected to be inconsistent. This observation is consistent with the fact that laser sensor was far more accurate than the other two sensors.

One of the limitations of the proposed technique is that when there is a large number of sensors supporting an inconsistent measurement, then, based on the beliefs of the individual measurements, the technique may consider inconsistent measurement to be the correct one, and might disregard the correct measurements obtained by fewer numbers of sensors. In psychology, this kind of problem is termed as group conformity. For example, when an individual's opinion differs significantly from that of others in a group, the individual is likely to feel extensive pressure to align his or her opinion with others. In the case of sensor

systems, this kind of condition is more likely to occur in adversarial situations, such as the battlefield, where events are prone to be camouflaged to escape detection. Hence, a formal criterion to establish the difference between spuriousness and opinion difference must be developed for the sensor fusion process to be accurately carried out in such adversarial situations. For example, in these situations, the technique proposed in this chapter could be applied if sensor models could be developed that represent the possibility/likelihood of events being camouflaged. Real time implementation and scalability aspects of the proposed sequential scheme have to be considered. To improve real time applicability of decentralized sensor fusion approaches, concepts from parallelization of processing can be incorporated. The recent interest in distributed sensing can incorporate such parallel/distributed framework of processing and sensor fusion.

## 7. Conclusions

Sensors measurements are inherently uncertain and often inconsistent. Appropriate consideration of uncertainty and identification/elimination of inconsistent measurements are essential for carrying out accurate estimation. The research reported in this chapter proposes a unified and formalized approach to fuse data from multiple sources which can take uncertainty of sensor data into account and automatically identify inconsistency in sensor data. Appropriate modeling of uncertainties in sensor measurement is necessary. This chapter presents an innovative neural network based method to model sensor's uncertainties. Further, the chapter presents a strategy that adds a term to the popular Bayesian approach corresponding to a belief that the sensor data is not spurious conditioned upon the data and true state. An information theoretic measure is utilized to observe the information content of the posterior distribution to identify and eliminate inconsistent data. An extensive simulation study was performed where data from three sensors was fused. It was observed that the presented method was very effective in identifying spurious data, and, elimination of spurious data ensured more accurate results. Finally, the effectiveness of the proposed technique to identify and eliminate inconsistent sensor data in sequential Bayesian fusion was demonstrated with the help of an experiment performed in a robotic workcell where measurements from stereo vision, infra-red proximity, and laser proximity sensor were fused to obtain three-dimensional occupancy profile of robotic workspace.

## 8. References

- Bayes T. (1763). An essay towards solving a problem in doctrine of chances, *Philosophical Transactions*, Vol. 53, 1763, pp. 370 - 418.
- Benaskeur, A. R. (2002). Consistent fusion of correlated data sources, *Proceedings of the IEEE Annual Conference of the Industrial Electronics Society*, Vol. 4, 5-8 Nov. 2002, pp. 2652 - 2656.
- Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*, Springer-Verlag, Berlin.
- Brooks, R. R. and Iyengar, S. S. (1998). *Multi-Sensor Fusion: Fundamentals and Applications with Software*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

- Chin, L. (1994). Application of neural networks in target tracking data fusion", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 30, No. 1, January 1994, pp. 281 - 287.
- Del Gobbo, D., Napolitano, M., Famouri, P., and Innocenti, M. (2001). Experimental application of extended Kalman filtering for sensor validation, *IEEE Transactions on Control Systems Technology*, Vol. 9, No. 2, March 2001, pp. 376 - 380.
- Dempster, A. P. (1968). A generalization of Bayesian inference, *Journal of Royal Statistical Society, B*, Vol. 30, No. 2, 1968, pp. 205 - 247.
- Durrant-Whyte, H.F. (1988). *Integration, Coordination and Control of Multi-Sensor Robot Systems*, Kluwer Academic Publishers, Norwell, MA.
- Elfes, A. (1992). Multi-source spatial data fusion using Bayesian reasoning, *Data Fusion in Robotics and Machine Intelligence*, Abidi, M. A. and Gonzalez, R. A. (eds.), Academic Press.
- Frolik, J., Abdelrahman, M., and Kandasamy, P. (2001). A confidence-based approach to the self-validation, fusion and reconstruction of quasi-redundant sensor data", *IEEE Transactions on Instrumentation and Measurement*, Vol. 50, No. 6, Dec. 2001, pp. 1761 - 1769.
- Garg, D. and Kumar, M. (2003). Object classification via stereo vision in a flexible manufacturing work cell, *Proceedings of the 10<sup>th</sup> International Conference on Mechatronics and Machine Vision in Practice*, Perth, Western Australia, December 9-11, 2003.
- Garg, D. and Kumar, M. (2007). Neural controllers, In *Wiley Encyclopedia of Computer Science and Engineering* (Benjamin Wah, ed.), Hoboken: John Wiley & Sons, Inc.
- Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning*. Addison Wesley Longman, Inc..
- Hall, D. L. and Llinas, J. (2001). *Handbook of Multisensor Data Fusion*, CRC Press, Boca Raton, FL.
- Haykin, S. S. (1998). *Neural Networks: A Comprehensive Foundation*, Prentice Hall Press.
- Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward network are universal approximators, *Neural Networks*, Vol. 2, 1989, pp. 359 - 366.
- Ibarguengoytia, P. H., Sucar, L. E., and Vadera, V. (2001). Real time intelligent sensor validation, *IEEE Transactions on Power Systems*, Vol. 16, No. 4, Nov. 2001, pp. 770 - 775.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems, *Transactions of the ASME-Journal of Basic Engineering*, Vol. 82, Ser. D, 1960, pp. 35 - 45.
- Kumar, M. and Garg, D. (2004). Intelligent multi sensor fusion techniques in flexible manufacturing workcells, *Proceedings of American Control Conference*, 2004, Boston, MA, pp. 5375 - 5380.
- Kumar, M., Garg, D., and Zachery, R. (2005a). Multi-sensor fusion strategy to obtain 3-D occupancy profile", *Proceedings of the 31<sup>st</sup> Annual Conference of the IEEE Industrial Electronics Society (IECON)*, Raleigh, NC, November 2005, pp. 2083 - 2088.

- Kumar, M., Garg, D., and Zachery, R. (2005b). Intelligent sensor modeling and data fusion via neural network and maximum likelihood estimation", Paper No. IMECE2005-80972, *Proceedings of the ASME International Mechanical Engineering Congress and Exposition*, November 2005, Orlando, FL.
- Kumar, M., Garg, D., and Zachery, R. (2006a). Stochastic adaptive sensor modeling and data fusion, *Proceedings of SPIE Conference on Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, San Diego, CA, Feb-March 2006, pp. 61740C-1 to 61740C-11.
- Kumar, M., Garg, D., and Zachery, R. (2006b). A generalized approach for inconsistency detection in data fusion from multiple sensors, *Proceedings of the American Control Conference*, Minneapolis, MN, 2006, pp. 2078 – 2083.
- Kumar, M., Garg, D., and Zachery, R. (2007). A method for judicious fusion of inconsistent multiple sensor data, *IEEE Sensors Journal*, Vol. 7, No. 5, May 2007, pp. 723-733.
- Mahajan, A., Wang, K., and Ray, P. K. (2001). Multisensor integration and fusion model that uses a fuzzy inference system, *IEEE/ASME Transactions on Mechatronics*, Vol. 6, Issue 2, June 2001, pp. 188 – 196.
- Manyika, J. and Durrant-Whyte, H. (1994). *Data Fusion and Sensor Management: a Decentralized Information-Theoretic Approach*, Ellis Howard Limited.
- McKendall, R. and Mintz, M. (1992). Data fusion techniques using robust statistics, In *Data Fusion in Robotics and Machine Intelligence*, Abidi, M. A. and Gonzalez, R. A. (eds.), Academic Press.
- Maybeck, P. S. (1979). *Stochastic Models, Estimation and Control*, Vol. 1, Academic Press, Inc..
- Nicholson, D. (2004). An automatic method for eliminating spurious data from sensor networks, *Proceedings of the IEE Conference on Target Tracking: Algorithms and Applications*, 23-24 March 2004, pp. 57 – 61.
- Press, S. J. (1989). *Bayesian Statistics: Principles, Models and Applications*, John Wiley and Sons.
- Rizzo, A. and Xibilia, M.G. (2002). An innovative intelligent system for sensor validation in Tokamak machines, *IEEE Transactions on Control Systems Technology*, Vol. 10, No. 3, May 2002, pp. 421 – 431.
- Rumelhart, D. E. and McClelland, J. L. (1988). *Explorations in Parallel Distributed Processing: a Handbook of Models, Programs, and Exercises*, MIT Press.
- Sasiadek, J. Z. (2002). Sensor fusion, *Annual Reviews in Control*, Vol. 26, 2002, pp. 203-228.
- Shafer, G. (1976). *A Mathematical Theory of Evidence*, Princeton University Press, Princeton, NJ.
- Soika, M. (1997). A sensor failure detection framework for autonomous mobile robots, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 3, 7-11 Sept. 1997, pp. 1735 – 1740.
- Wellington, S. J., Atkinson, J. K., and Sion, R. P. (2002). Sensor validation and fusion using the Nadaraya-Watson statistical estimator, *Proceedings of the IEEE International Conference on Information Fusion*, Vol. 1, July 2002, pp. 321 – 326.
- Yager, R. R. and Zadeh, L. A., (Eds.) (1991). *An Introduction to Fuzzy Logic Applications in Intelligent Systems*, Kluwer Academic Publishers.

Zhang, Z., Deriche, R., Faugeras, O., and Luong, Q.-T. (1995). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry, *Artificial Intelligence Journal*, Vol.78, October 1995, pp. 87 - 119.

# Updating Scarce High Resolution Images with Time Series of Coarser Images: a Bayesian Data Fusion Solution

Dominique Fasbender<sup>1</sup>, Valérie Obsomer<sup>1,2</sup>,  
Patrick Bogaert<sup>1</sup> and Pierre Defourny<sup>1</sup>

<sup>1</sup>*Dpt of Environmental Sciences and Land Use Planning, Université catholique de Louvain*

<sup>2</sup>*Institute of Tropical Medicine, Antwerpen  
Belgium*

## 1. Introduction

As a consequence of the great variability between sensors, the characteristics of remotely sensed data widely differ with respect to spectral and spatial resolutions. Additionally to their respective technical characteristics and peculiarities, sensors also have different temporal frequencies of acquisition. Coarser sensors (e.g. SPOT VEGETATION or TERRA MODIS) have generally close to daily acquisition rates while high spatial resolution sensors (e.g. SPOT HRVIR or IKONOS) have lower acquisition rates. Cloud-free high resolution imagery may therefore not be available at the required period unlike coarser resolution images. On top of this, high resolution images are sometimes so highly priced that updating past high resolution images with recent coarse images can be cost effective. For these reasons, there is a real need for a sound theoretical framework that aims at merging information coming from two or more different sensors while taking explicitly into account the spatial resolution discrepancies between images. Typically, for cost effective applications, this could involve predicting a high resolution image by updating a past one with more recent but coarser images.

It is a common fact that remote sensors have different spatial resolution. This change of resolution is thus a typical issue in remote sensing applications. Depending on users' needs and the heterogeneity of the study areas, different algorithms of fusion were proposed for the spatial enhancement of remotely sensed images. These include Brovey method (Pohl & van Genderen, 1998), Intensity-Hue-Saturation (IHS; Harrison & Jupp, 1990), Principal Component Analysis (PCA; Pohl & van Genderen, 1998), wavelet-based Multi-Resolution Analyses (MRA; Zhou et al., 1998; Garzelli & Nencini, 2005; Ranchin et al., 2003), High-Pass Filter (HPF; Chavez et al., 1991), generalized Laplacian methods (Aiazzi et al., 2002) and downscaling cokriging (Pardo-Iguzquiza et al., 2006), just to quote a few of them. Detailed reviews of the numerous available algorithms can be found in (Pohl & van Genderen, 1998; Chavez et al., 1991; Wang et al., 2005; Ballester et al., 2006 or Laporterie et al., 2005). Unfortunately, most of these methods are devoted to the case of spatial enhancement of remotely sensed images only in the case of simultaneous images. In other words, the images to be fused are assumed to be taken at the same time but with different spectral bands and



different spatial resolutions. It is therefore relevant to focus our attention on data fusion methods that typically enable us to account for several information sources in order to produce a single but improved image. Recently, a new Bayesian Data Fusion (BDF) framework was proposed in a general space-time prediction context by (Bogaert & Fasbender, 2007), with the aim of merging various kind of information sources that are all different but relevant for a same target variable. Though initially developed with stochastic space-time random fields applications in mind, the method proved to be efficient for remotely sensed applications as well (e.g. Fasbender et al., 2008b).

This chapter shows how the BDF approach can be used for the update of high resolution images with coarser images. In order to illustrate the general principle of the method, a synthetic case study was created from SPOT VEGETATION composite images (1km resolution) available at different dates. Spatially degraded 10km and 100km images are generated and used as coarse images. From the whole time-series of coarser images and only few of the initial images over time, it is then shown how BDF allows predicting the high resolution image at, say, date 2 by combining at the same time (i) the high-resolution image at a previous date 1, (ii) the coarser image at date 2, and (iii) the evolution of the coarser images between dates 1 and 2. Based on a quality assessment conducted by comparing the BDF-predicted images with the corresponding original 1 km images, it is shown that the method is able to provide both consistent results and improved images. Built on sound theoretical grounds, easy to implement and computationally fast, this method opens new avenues in the field of cost effective and efficient data fusion techniques for remotely sensed data.

## 2. Bayesian data fusion

Combining different sources of information into a single final result (i.e. data fusion) is a problem of general concern for a large panel of applications, that goes far beyond satellite imagery and encompasses a wide array of potential methods. Among them, Bayesian approaches have led to interesting applications with respect to various problems such as image surveillance (Jones et al. 2003), object recognition (Chung & Shen, 2000), object localization (Pinheiro & Lima, 2004), robotic (Moshiri et al., 2002), image processing (Rajan & Chaudhuri, 2002), classification of remote sensing images (Bruzzone et al., 2002) and environmental modelling (Wikle et al., 2001), just to quote few of them. The main advantage of a Bayesian approach is to set the problem of data fusion into a clear probabilistic framework. The present chapter relies on a general Bayesian Data Fusion approach in the context of spatial data (Bogaert & Fasbender, 2007). Its specific implementation will focus here on the problem of updating high resolution images with time series of coarser images.

### 2.1 General formulation

The basic concept of BDF as presented in (Bogaert & Fasbender, 2007) relies on the idea that variables of interest, denoted as vector  $Z = (Z_1, \dots, Z_n)'$ , cannot be directly observed. Instead, they are linked to the observable variables  $Y_{i,j}$  through an error-like model, with

$$Y_{i,j} = g_{i,j}(Z_i) + E_{i,j} \quad (1)$$

where  $g_{i,j}(\cdot)$ 's are functionals and  $E$  is a vector of random errors that are stochastically independent from  $Z$ . Assuming that the  $E_{i,j}$ 's of the random vector  $E$  are stochastically

independent, it is easy to obtain the conditional probability density function (pdf) of the vector of interest given the observed variables, with

$$f(z | y) \propto f_Z(z) \prod_{i=1}^n \prod_{j=1}^{p_i} f_{E_{i,j}}(y_{i,j} - g_{i,j}(z_i)) \quad (2)$$

where  $i$  corresponds to the channel number and  $p_i$  is the number of secondary information corresponding to the same  $Z_i$  (see Bogaert & Fasbender, 2007 for more details).

Using Bayes theorem again for  $f_{E_{i,j}}(y_{i,j} - g_{i,j}(z_i)) = f(y_{i,j} | z_i)$ , Eq. (2) becomes

$$f(z | y) \propto f_Z(z) \prod_{i=1}^n \prod_{j=1}^{p_i} \frac{f(z_i | y_{i,j})}{f(z_i)} \quad (3)$$

where  $f(z_i)$  is the *a priori* distribution of  $Z_i$ . According to the information sources at hand, intermediate stages between Eq. (2) and Eq. (3) are possible as well (e.g. an expression mixing both  $f_{E_{i,j}}(\cdot)$  and  $f(\cdot | y_{i,j})$  distributions).

It is worth noting that this BDF framework has interesting similarities with other multi-sensor data fusion methods (see Mitchell, 2007 for more details). The advantage of the BDF framework over other existing multi-sensor data fusion methods is that it proposes a general formulation when accounting for several secondary information sources whatever the nature of these secondary information. Thanks to this, the range of applications that can be tackled by the BDF approach is wider and is far beyond the scope of the traditional multi-sensor fusion issue.

## 2.2 Specific assumptions

The previous section presented the general BDF framework but it is important here to choose some specific assumptions in order to tackle the issue of updating high resolution images. In this context, there are three sources of information available at each location, namely the high resolution image at date 1, the coarser image at date 1 and the coarser image at date 2. In our implementation, the following notations will be used :

- $Z$  is the unknown multispectral reflectance values for the finer resolution pixel at date 2
- $Y_H$  is the multispectral reflectance values for the finer resolution pixel at date 1
- $Y_{L1}$  is the multispectral reflectance values for the coarser resolution pixel at date 1
- $Y_{L2}$  is the multispectral reflectance values for the coarser resolution pixel at date 2

Let us first assume that the finer and the coarser images share the same spectral bands. In these conditions, several predictions methods for the unknown pixel  $Z$ , all based on these information sources, can be used. In this application, only two methods will be evaluated. The first one simply consists in using the coarser multispectral  $Y_{L2}$  has a raw estimate of the unknown pixel  $Z$ . Of course, although the image composed by the pixels  $Y_{L2}$  is smoother than the target high resolution image, the global fluctuations are accounted for in the smooth image. The unexplained part of the variability is then the local fluctuations with a mean equal to zero for each of the spectral bands and a variance  $\Sigma_1$  that can be estimated from the difference  $Y_H - Y_{L1}$  between high and coarser resolution images at date 1.

The second prediction considered here is based on the following High-Pass Filtering approach :

1. the high resolution image at date 1 is first decomposed into a lower frequencies image and a higher frequencies image using a Gaussian filter.
2. the higher frequencies image is then combined with the coarser image at date 2.

The resulting image has thus the advantages that details are provided by the high resolution image at date 1 whereas the global fluctuations are provided by the coarser image at date 2. The mean of this image is thus expected to be the same as the objective image and its variance  $\Sigma_2$  can be estimated from the variance of  $Y_{L1} - Y_{L2}$ .

Now, assuming multivariate Gaussian distribution for both secondary information sources as well as a non-informative *prior* distribution (i.e. constant distribution over the domain), the fused distribution is also a multivariate Gaussian distribution with a mean vector  $\mu$  and a covariance matrix  $\Sigma$  given by

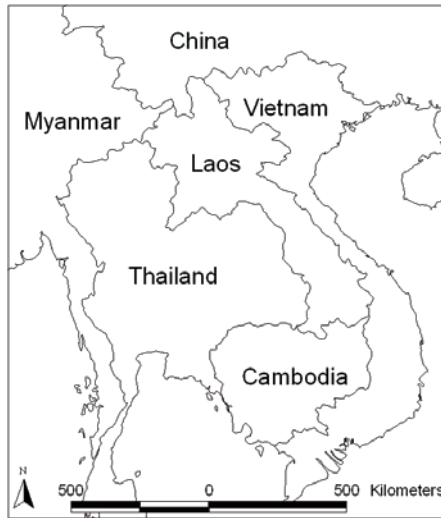


Fig. 1. Study site

$$\Sigma^{-1} = \Sigma_1^{-1} + \Sigma_2^{-1} \quad (4)$$

$$\mu = \Sigma(\Sigma_1^{-1} \mu_1 + \Sigma_2^{-1} \mu_2) \quad (5)$$

where  $\mu_1$  and  $\mu_2$  are respectively the two predictions described above. The value  $\mu$  is thus a relevant candidate for the prediction of the unknown finer resolution pixel  $Z$ .

Up to this point, it was assumed that finer and coarser images share the same spectral bands. However, this situation rarely occurs in real case applications. Fortunately, the generalization of the previous approaches is straightforward for non fully overlapping spectral bands. Indeed, one could easily generalize  $Y_{L1} = Z + E$  into, e.g.,

$Y_{L1} = AZ + b + E$  with  $b \in \mathbb{R}$  and  $A \in \mathbb{R}^{n \times n}$  (i.e. the spectral bands of the coarser image are linked to the spectral bands of the target image through a translation vector  $b$  and a

scaling matrix  $A$ ). Thanks to this, it is now theoretically possible to update high resolution images from a first sensor like e.g. SPOT HRVIR or IKONOS with time series of coarser images from a second sensor like e.g. TERRA MODIS or SPOT VEGETATION.

### 3. Demonstration case study

#### 3.1 Simulated data

In this case study, coarser images were simulated from biweekly composite images based on real SPOT VEGETATION images with a spatial resolution of 1km. The original images were taken in the South-East Asia region (Fig. 1). Composite images were computed on a biweekly basis using the mean compositing method (Vancutsem et al., 2007). The covered period is 2004-2005 so that 50 images were available for this study. Fig. 2 shows this evolution for the year 2005. After a clouds screening, each original image was degraded at 10 km and 100km resolution by averaging all 1km pixels corresponding to the area. Resulting images were then resampled at 1km in order to match the original 1km spatial resolution (Fig. 3).

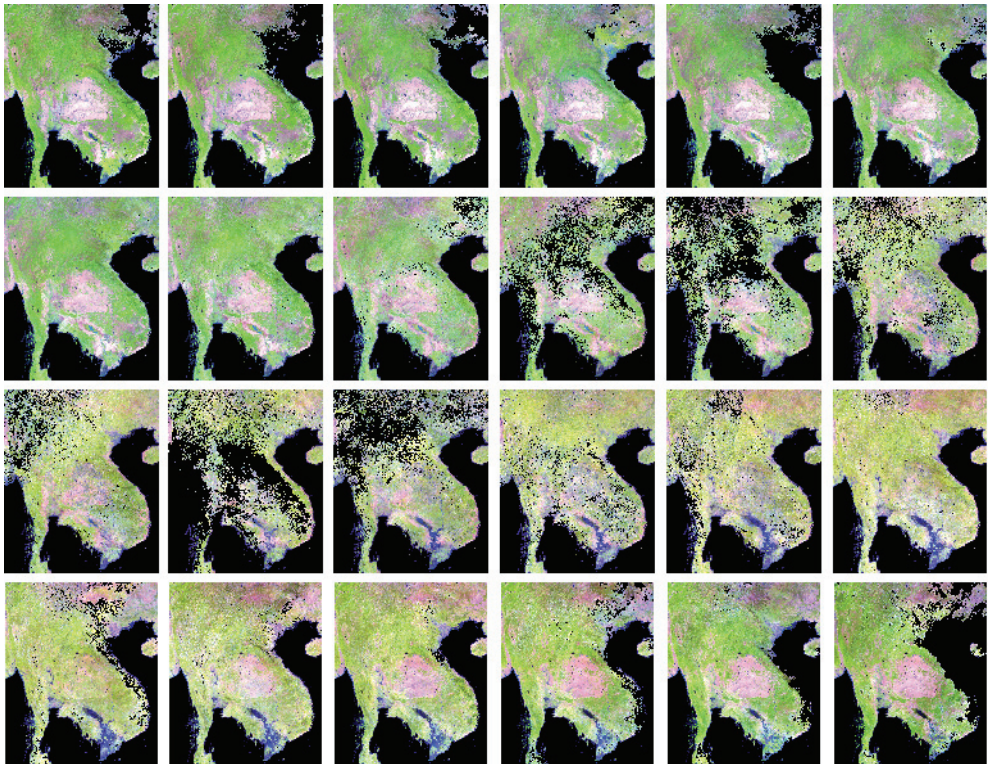


Fig. 2. Evolution of the original biweekly images at 1km resolution.

In the context of this chapter, original 1km images are assumed to be the finer images and blurry simulated ones are the coarser images. Furthermore, it is worth noting that spectral

bands of both the finer and the coarser images are the same so that the specific assumptions described in the previous section can be directly applied here without modifications.

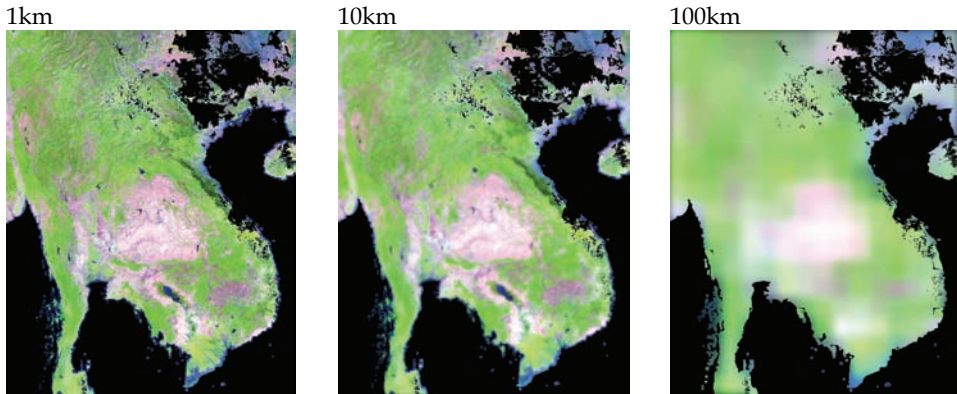


Fig. 3. Examples of original finer (1km) and corresponding simulated coarser images (10km and 100km).

### 3.2 Results

Two different situations are considered in this application. In the first situation, one assumes that the finer resolution images are exhaustively known for the previous year. Consequently, one can choose the most relevant previous finer image. In the second situation, one assumes that there is only one previous finer resolution image, so that there may be a seasonal shift between the previous and the objective finer resolution images (see Fig. 2 for the differences between seasons). Both situations are described and compared hereafter in the next subsections.

#### 3.2.1 Situation 1: finer resolution images exhaustively known for the previous year

The methodology of Section 2.2 was applied here in the case where finer resolution images are exhaustively known for the previous year and when using coarser resolution input images either at 10km or at 100km (Fig. 4). Using the finer resolution image one year before the target date (i.e. at the same biweekly composite number but at the previous year) is a relevant choice for the finer resolution input in the fusion method : in that case, the previous finer resolution image and the new objective image correspond to the same period. As a consequence, there will be less changes due to the seasonal shift between both finer images.

Fig. 5 shows the evolution of the fused image for the year 2005 when using 100km coarser images. By comparison with the true evolution of the 1km images in Fig. 2, there is no significant color difference between the predicted images from the BDF method and the objective images. Furthermore, it is clear that the details from the finer resolution images are correctly accounted for in the fused images while the colors are updated by the coarser images (Fig. 6 illustrates this for the 21<sup>st</sup> biweekly composite image and with 100km coarser image). Results (not shown here) were also convincing with the 10km coarser input images.

It is also worth noting that, although only one finer image was chosen in this application for sake of brevity, there is no theoretical limitation on the number of finer resolution images to be accounted for within the BDF framework. The methodology presented in Section 2.2 and

more specifically Eqs. 4 and 5 can easily be generalized to the case of multiple finer resolution input images with no additional theoretical development. It is thus possible to account for the entire time series of the finer resolution images (before and after the objective image) . This is left for further researches at this point.

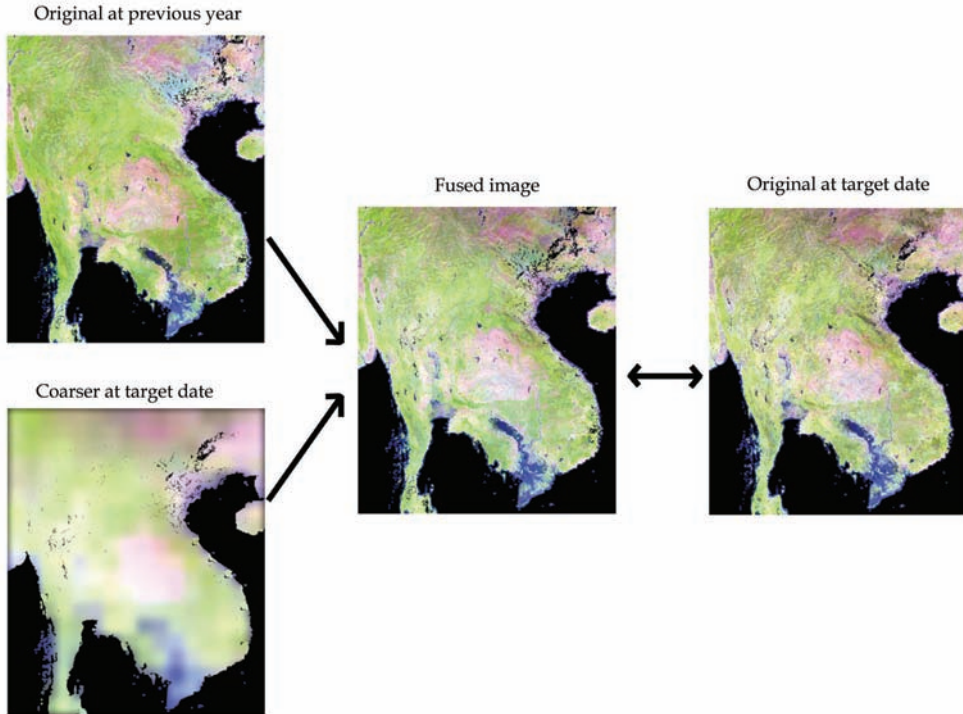


Fig. 4. Situation 1. Inputs for the fusion are the original 1km image at previous year and the coarser image (here 100km) at the target date. Fused images are then compared with the true original 1km images at the target date.

### 3.2.2 Situation 2: finer resolution image known for only one previous date

In the second situation, the finer resolution input image is assumed to be available at only one biweekly composite image. By doing this, the previous situation is thus generalized to the case where there is less available information at the finer resolution. The past finer image can be located anywhere in the current year (or even be related to a previous year) so that the seasonal effects are not necessarily seen in the finer image. Furthermore, the information relevance of the finer image with respect to the fused one is expected to drop along with the change of seasons (e.g. rainy season *versus* dry season). As a consequence, the seasonal trend will only be included by the coarser image and details from the finer image will be included as long as the past and target biweekly images will correspond to the same season. In order to illustrate this situation, the only finer resolution image was assumed to be the first 1km biweekly image of the year 2005. Fig. 7 illustrates this situation in the case of 10km coarser images, although the methodology is of course the same for 100km.

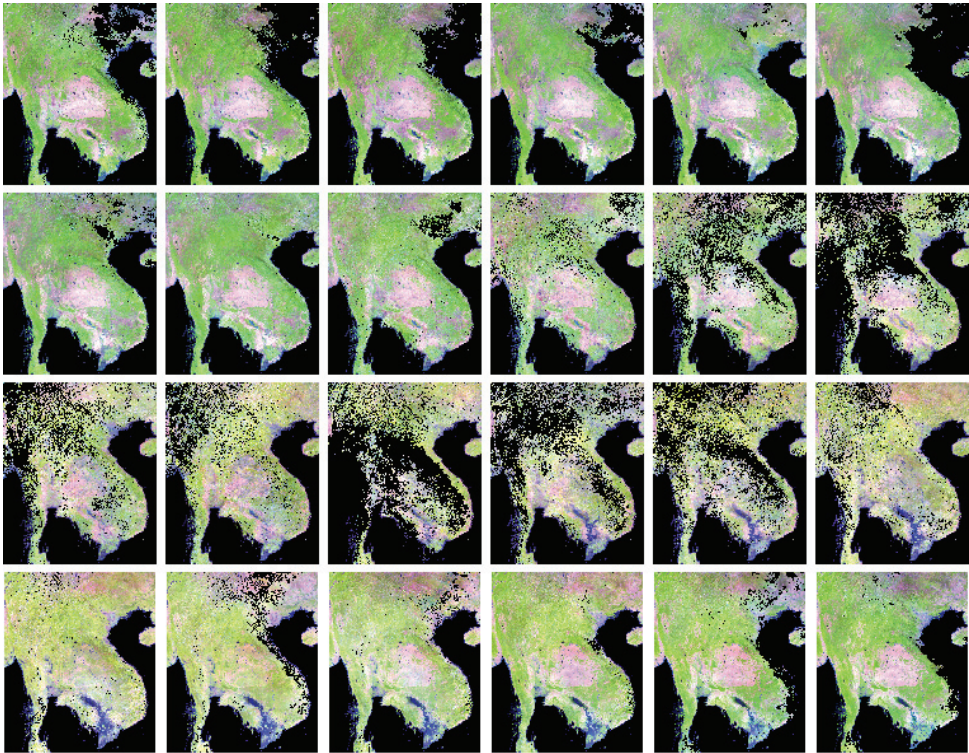


Fig. 5. Evolution of the decadal fused images for year 2005 when using the biweekly composite high spatial resolution image of the previous year. The resolution of the coarser images is equal to 100km here.

Similarly to the previous situation, the same methodology was applied for both spatial resolution for the coarser image (10km and 100km). Fig. 8 shows the temporal evolution of the fused images when using 10km coarser images. Again, a simple comparison between Figs. 2 and 8 shows that this temporal evolution is in good accordance with the temporal evolution of the target images at 1km resolution (it is worth noting that pixels that were covered by clouds in the finer input image were of course not updated in the fused images). Furthermore, Fig. 9 illustrates the effect of the inputs on the fused results. Indeed, one can clearly see that the change of colors between the first and the second dates is correctly accounted for thanks to the 10km coarser image while the details from the previous finer resolution image are well preserved. Results (not shown here) were also in good accordance when using the 100km coarser images.

### 3.2.3 Quality assessment and comparison between both situations

Several indices were chosen for the quality assessment of the fused images. As time series are complete for each of the three spatial resolution, it is straightforward to compare updated images with the true images at 1km at each of the 25 biweekly period of year 2005. More specifically, one can compute the Mean Error (ME), the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) with

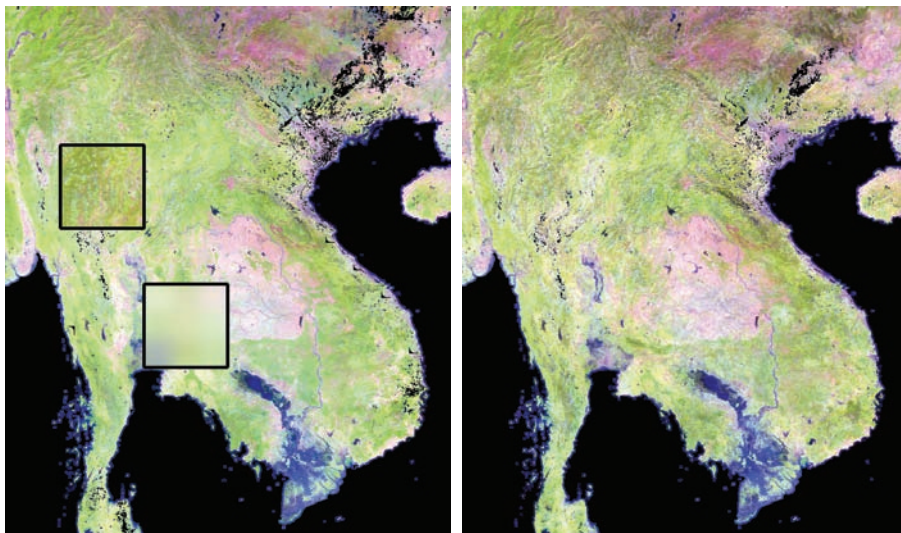


Fig. 6. On the left : the 21<sup>st</sup> biweekly fused image when using (i) original 1km image at previous year (upper rectangle) and (ii) coarser 100km image at target date (lower rectangle). On the right : original 21<sup>st</sup> biweekly 1km image.

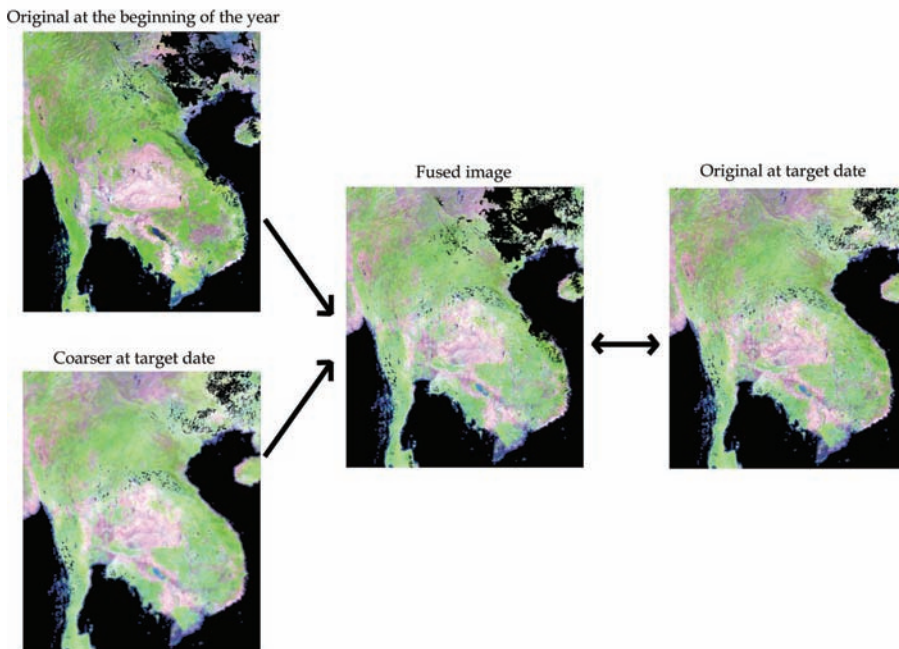


Fig. 7. Situation 2. Inputs for the fusion are the first biweekly original 1km image of the year and the coarser image (here 10km) at the target date. Fused images are then compared with the true original 1km images at the target date.



$$ME = \frac{1}{N} \sum_{k=1}^N E_k \quad (6)$$

$$MAE = \frac{1}{N} \sum_{k=1}^N |E_k| \quad (7)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^N E_k^2} \quad (8)$$

where  $E_k$  is the difference between pixels values of the true and the fused images and  $N$  is the number of pixels in the images. It is worth noting that these indices were computed without cloud pixels (i.e. pixels that are detected as clouds at least once in the two images).

Fig. 10 shows the evolution of the ME for Situations 1 and 2, both for the two coarser resolution (each curve corresponds to a different spectral band). It is clear from these results that using the finer image of the previous year and the 10km coarser image provides the best results regarding the ME, while the ME values have larger amplitudes in the other cases (especially in the case of only one finer image and 100km coarser image). However, these amplitudes are significantly small (reflectance values belong to  $[0,1]$  interval) in order to conclude that the method is unbiased.

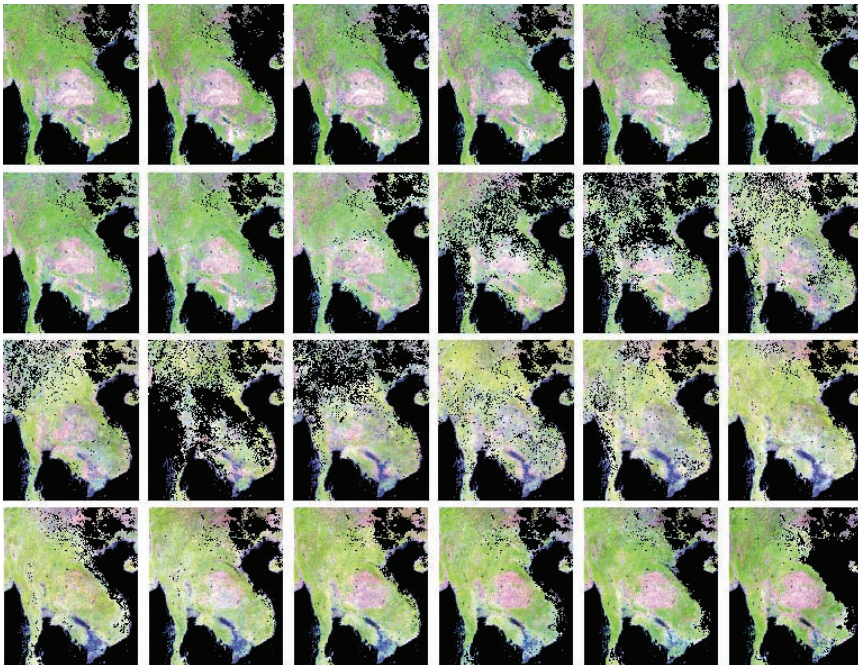


Fig. 8. Evolution of the biweekly fused images for the year 2005 when using the first biweekly finer image of the current year. The resolution of the coarser images was equal to 10km here.

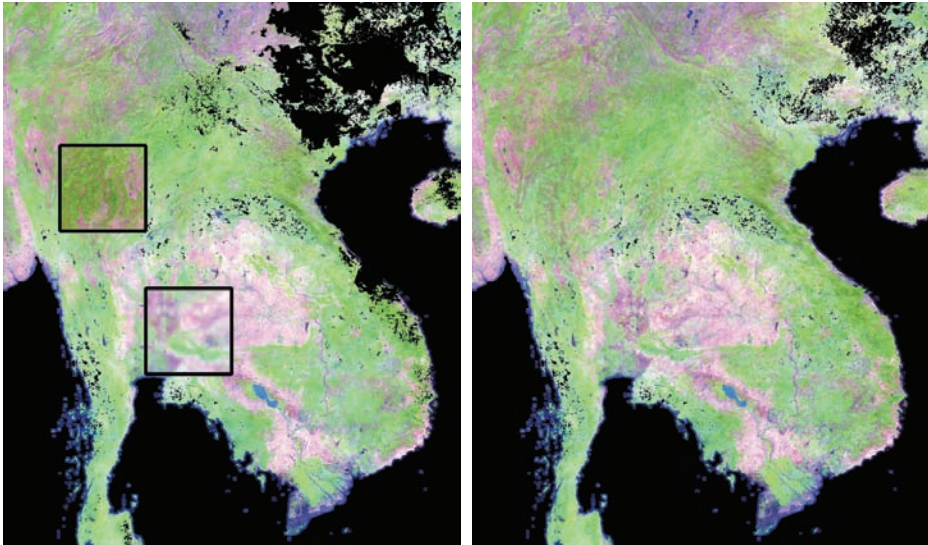


Fig. 9. On the left : the 9<sup>th</sup> biweekly fused image when using (i) original 1km image at beginning of the year (upper rectangle) and (ii) coarser 10km image at target date (lower rectangle). On the right : original 9<sup>th</sup> biweekly 1km image.

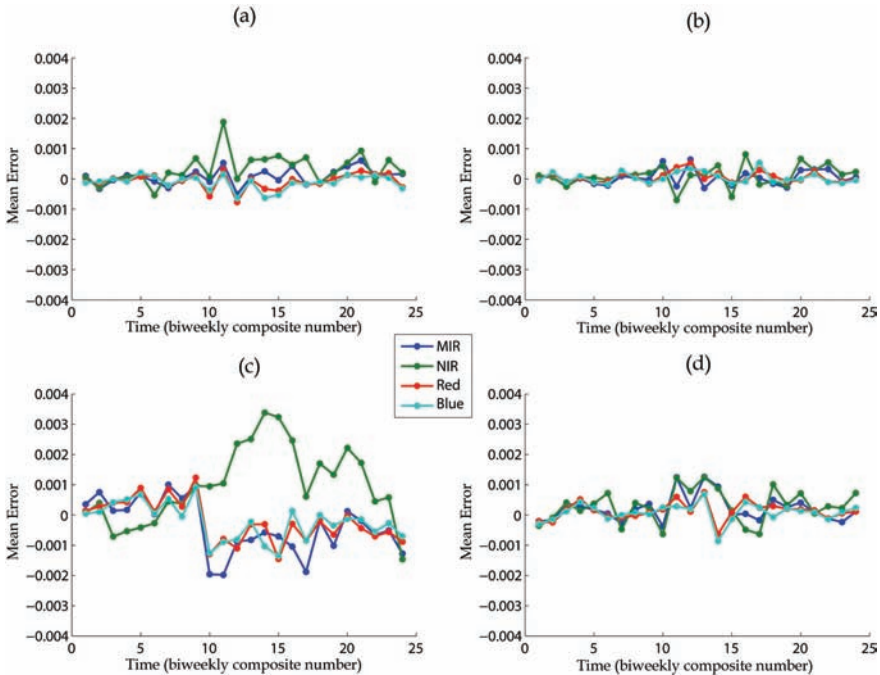


Fig. 10. Evolution of the Mean Error (reflectance) as a function of the time for Situations 1 (b and d) and 2 (a and c), both for the 10km (a and b) and the 100km (c and d) coarser resolutions.

Similarly, Figs. 11 and 12 show respectively the evolutions of the MAE and RMSE for the various cases. Again, it is clear from these results that it is preferable to use the finer resolution image of the previous year than to rely on a unique finer image at another season. However, the MAE and RMSE significantly increase between the 10<sup>th</sup> and the 20<sup>th</sup> biweekly composite numbers (i.e. the end of the dry season and the beginning of the rain season). Therefore, fused images are less precise for this period of the year. This is probably a consequence of both the intra and inter annual variation of the vegetation in the studied area.

As there were two secondary information sources (i.e. the past finer image and the coarser one at target date), it is also interesting to focus on the influence of these sources on the fused images. For this, correlation coefficients were computed between corresponding spectral bands of the different images. Again, these coefficients were computed at each biweekly composite number, allowing us to see how they evolve over time.

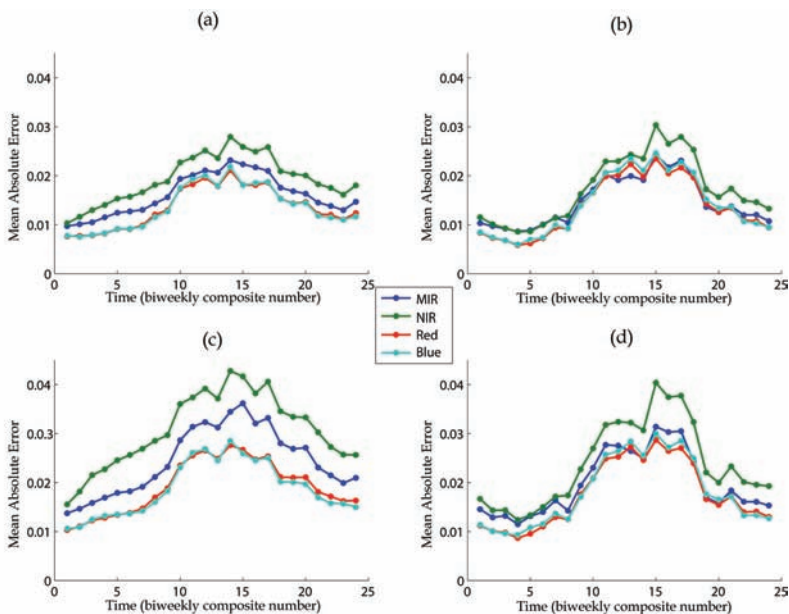


Fig. 11. Evolution of the Mean Absolute Error (reflectance) as a function of the time for Situations 1 (b and d) and 2 (a and c), both for the 10km (a and b) and the 100km (c and d) coarser resolutions.

Correlation coefficients between the fused and the original 1km input images are represented in Fig. 13. It is worth noting that the influence of the finer resolution image decreases rapidly at the end of the first quarter of the year (i.e. end of the dry season) when using the finer resolution image at the first biweekly composite, whereas this influence remains rather constant when using the finer image of the previous year. This is of course

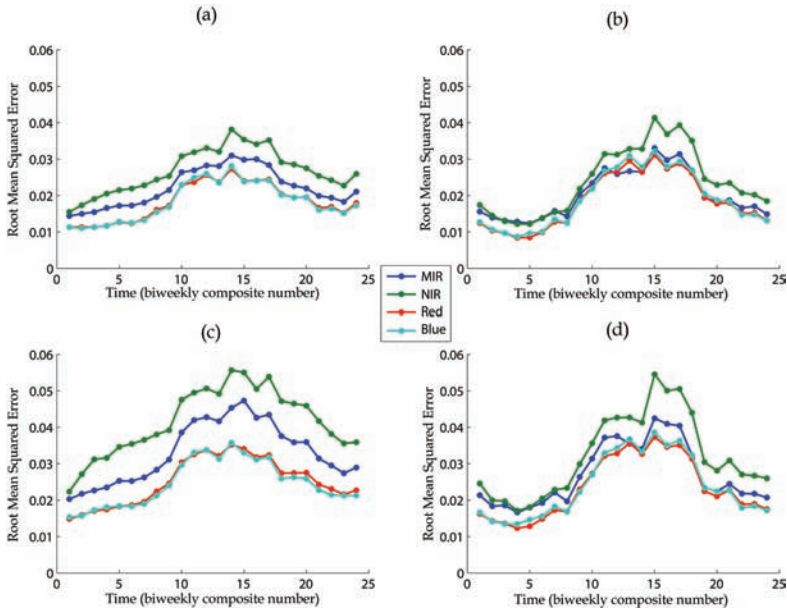


Fig. 12. Evolution of the Root Mean Squared Error (reflectance) as a function of the time for Situations 1 (b and d) and 2 (a and c), both for the 10km (a and b) and the 100km (c and d) coarser resolutions.

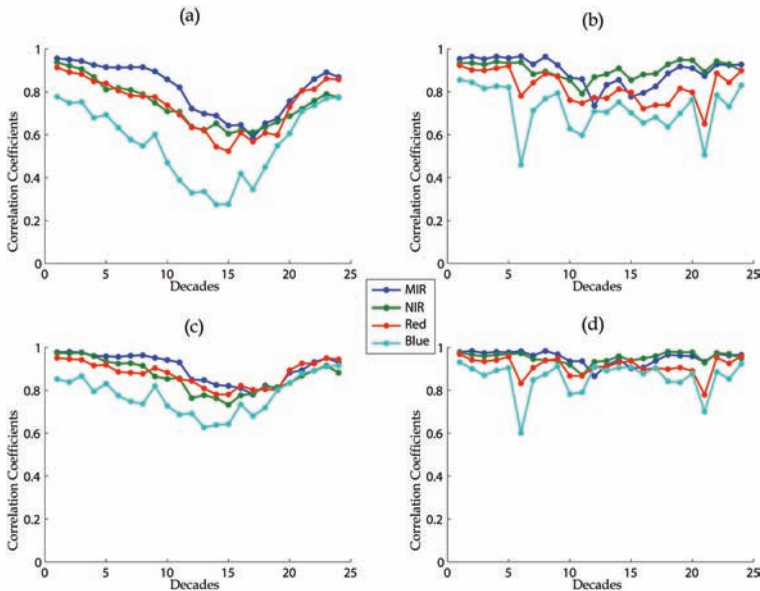


Fig. 13. Evolution of the correlation coefficients between fused and original 1km input images as a function of the time for Situations 1 (b and d) and 2 (a and c), both for the 10km (a and b) and the 100km (c and d) coarser resolutions.

mainly due to the fact that there is a significant seasonal shift of vegetation between the first biweekly composite of the year (i.e. beginning of the dry season) and the middle of the year (i.e. rainy season). Furthermore, the correlation coefficients increase significantly during the rainy season, i.e. when the vegetation starts to grow again. It is also worth noting that the influence of the finer input image on the fusion results is bigger when using 100km coarser image. This is simply because 10km images are more relevant for the prediction at 1km than 100km ones. Thus, as 100km images are less informative, their influence on the fused images is smaller.

Similarly, it is interesting to focus on the evolution of the coarser images's influences on the fused results (Fig. 14). It is worth noting that this influence is rather constant when using the first biweekly finer image of the current year, while it drops in the middle of the year when using the finer image of the previous year. These behaviors are thus precisely opposite with those observed for the finer resolution images in Fig. 13. Again, it is mainly due to the fact that the finer images of the previous year correspond to the same season as the target unknown image, thus globally exhibiting the same vegetation conditions. As a consequence, fused images rely a little bit less on the coarser image for the update of the previous finer images (i.e. correlation coefficients are smaller).

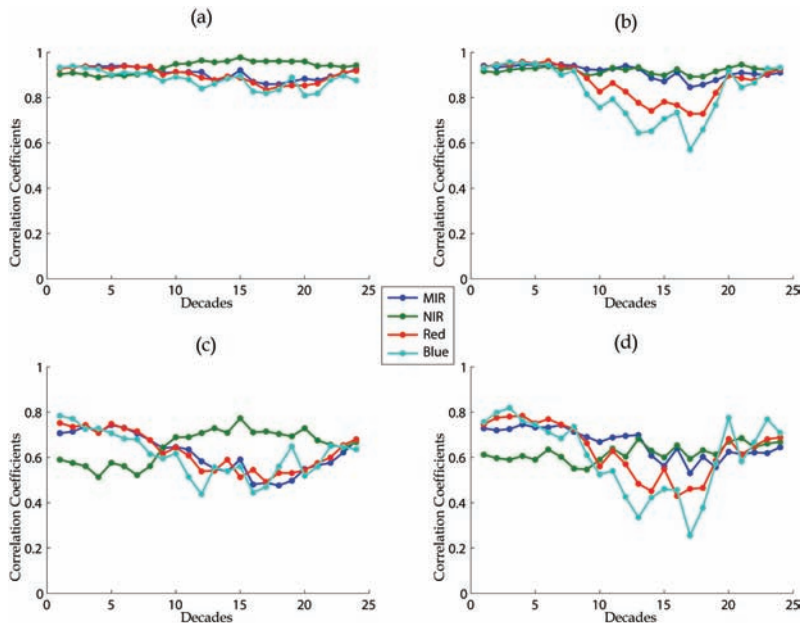


Fig. 14. Evolution of the correlation coefficients between fused and coarser input images as a function of the time for Situations 1 (b and d) and 2 (a and c), both for the 10km (a and b) and the 100km (c and d) coarser resolutions.

## 4. Conclusion

In this chapter, a Bayesian data fusion (BDF; Bogaert & Fasbender, 2007) framework was applied for the update of scarce high resolution images with time series of coarser images. This BDF framework aims at reconciling various secondary information sources into a unique prediction. Although initially proposed in a spatial prediction context, a generalization of this BDF approach was presented here for space-time predictions. It is worth noting that, as information are known exhaustively over space, the use of remotely sensed images is a singular case of secondary information sources so that interpolation (or spatial predictions) is not even needed here. Other applications using this BDF framework can be found in (Fasbender et al., 2008a; Fasbender et al., 2008b; Fasbender et al., 2008c).

After a brief general description of the BDF framework, several specific hypotheses were assumed in order to account for the three available information sources (the coarser images at date 1 and 2 and the finer image at date 1). Based on these three images, two methods were considered for the prediction of the target image (i.e. the finer image at date 2). The first method was to consider the coarser image at date 2 as a raw estimation of the target image. The second method was based on a High-Pass Filter (HPF) approach for which the lower frequencies of the finer image at date 1 are substituted by the coarser image at date 2. Consequently, details are provided by the high resolution image at date 1 whereas the global fluctuations are provided by the coarser image at date 2. The final prediction is eventually based on the combination of both methods within the BDF framework.

In this chapter, the proposed methodology was applied to a synthetic case study. Coarser images were simulated from biweekly composite images based on real 1km SPOT VEGETATION images in the South-East Asia region. Two coarser resolutions were tested here : 10km and 100km (even if the ratio between finer and coarser resolution images is expected to be smaller for real case applications). Moreover, two situations differing with respect to the amount of information sources at finer resolution were considered in this illustration. In Situation 1, finer resolution images were assumed to be available for the whole previous year, whereas only one finer image was available in Situation 2 (here, the first biweekly image of year 2005). Results showed that the method correctly accounted for the important seasonal trend due to the dry and rain seasons whatever the resolution of the coarser images (10km or 100km) or the amount of available finer images (only one or the whole previous year). Although visual interpretations were clearly in favor of BDF predictions, a validation was also performed using the true 1km images. Mean Error (ME), Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) were computed for each spectral band, both situations and both coarser resolutions, as a function of the time. Although the ME values fluctuate around zero (showing the accuracy of the method), the MAE and the RMSE values increase during the dry season (showing thus that there is a drop of precision for the method in this period of the year). This effect is most probably due to local changes of vegetation that are not observable at the coarser resolutions. However, results showed that using the finer image of previous year and 10km coarser images led to the most efficient updates (i.e. smaller ME, MAE and RMSE amplitudes).

As there were two methods of prediction to be merged within the BDF framework, it is also interesting to focus on their respective influences on the fused images. Results showed that the influence of the finer input images is rather constant in Situation 1, while this influence drops during the dry season in Situation 2. This is of course mainly due to the fact that, in Situation 1, finer input images and target ones are assumed to correspond to the same season, exhibiting thus the same stage of vegetation. Conversely, the influence of the coarser images on the fused images drops during the dry season in Situation 1, while being rather constant in Situation 2. Again, this inversion is probably due to the fact that the finer input and the target images are more similar in Situation 1 than in Situation 2, thus relying less on the coarser input image in Situation 1.

Although we only applied it here to a synthetic case study, generalizations of this BDF method are possible in order to tackle real case applications. As examples, using finer and coarser images with different spectral bands and using more than one past finer images for the prediction are just two possibilities for future researches. It thus opens new avenues in the context of updating high resolution images with coarser images.

## 5. Acknowledgment

The authors would like to thank the Belgian Scientific Policy for funding the DYNMAP project through the STEREO II earth observation program.

## 6. References

- Aiazzi, B.; Alparone, L.; Baronti, S. & Garzelli, A. (2002). Context-driven fusion of high spatial and spectral resolution images based on oversampled multiresolution analysis, *IEEE Trans. Geosci. Remote Sens.*, Vol. 40, No. 10, October 2002, 2300-2312.
- Ballester, C.; Caselles, V.; Igual, L.; Verdera, J. & Rougé, B. (2006). A variational model for P+XS image fusion, *Int. J. Comput. Vision*, Vol. 69, No. 1, August 2006, 43-58.
- Bogaert, P. & Fasbender, D. (2007). Bayesian data fusion in a spatial prediction context : a general formulation, *Stoch. Env. Res. Risk A.*, Vol. 21, No. 6, November 2007, 695-709, ISSN 1436-3240.
- Bruzzone, L.; Cossu, R. & Vernazza, G. (2002). Combining parametric and non-parametric algorithms for a partially unsupervised classification of multitemporal remote-sensing images, *Inf. Fusion*, Vol. 3, No. 4, December 2002, 289-297.
- Chavez, P. S.; Sides, S. C. & Anderson, A. (1991). Comparison of three different methods to merge multiresolution and multispectral data: Landsat TM and SPOT panchromatic. *Photogram. Eng. Remote Sens.*, Vol. 57, No. 3, June 1991, 295-303.
- Chung, A. C. S. & Shen, H. C. (2000). Entropy-based Markov chains for multisensor fusion, *J. Intell. Robot. Syst.*, Vol. 29, No. 2, October 2000, 161-189.
- Fasbender, D.; Peeters, L.; Bogaert, P. & Dassargues, A. (2008a). Bayesian data fusion applied to water table spatial mapping, *Water Resour. Res.*, Vol. 44, W12422, doi:10.1029/2008WR006921

- Fasbender, D.; Radoux, J. & Bogaert, P. (2008b). Bayesian data fusion for adaptable image pansharpening, *IEEE Trans. Geosci. Remote Sens.*, Vol. 46, No. 6, June 2008, 1847-1857, ISSN 0196-2892.
- Fasbender, D.; Tuia, D.; Bogaert, P. & Kanevski, M. (2008c). Support-based implementation of Bayesian data fusion for spatial enhancement: Application to ASTER thermal images, *IEEE Geosci. Remote Sens. Letters*, Vol. 5, No. 4, October 2008, 598-602, ISSN 1545-598X.
- Garzelli, A. & Nencini, F. (2005). Interband structure modeling for Pan-sharpening of very high-resolution multispectral images, *Inf. Fusion*, Vol. 6, No. 3, September 2005, 213-224.
- Harrison, B. A. & Jupp, D. L. B. (1990). *Introduction to image processing*, CSIRO Division of Water Resources, ISBN 978-0643051089, Canberra.
- Jones, G. D.; Allsop, R. E. & Gilby, J. H. (2003). Bayesian analysis for fusion of data from disparate imaging systems for surveillance, *Image and Vision Computing*, Vol. 21, No. 10, September 2003, 843-849.
- Laporterie-Déjean, F.; de Boissezon, H.; Flouzat, G. & Lefèvre-Fonollosa, M.-J. (2005). Thematic and statistical evaluations of five panchromatic/multispectral fusion methods on simulated PLEIADES-HR images, *Inf. Fusion*, Vol. 6, No. 3, September 2005, 193-212.
- Mitchell, H.B. (2007). *Multi-Sensor Data Fusion*, Springer-Verlag, ISBN 978-3-540-71463-7, Berlin Heidelberg.
- Moshiri, B.; Asharif, M. R. & Hosein Nezhad, R. (2002). Pseudo information measure: a new concept for extension of Bayesian fusion in robotic map building, *Inf. Fusion*, Vol. 3, No. 1, March 2002, 51-68.
- Pardo-Iguzquiza, C.; Chica-Olmo, M. & Atkinson, P. M. (2006). Downscaling cokriging for image pansharpening, *Remote Sens. Environ.*, Vol. 102, No. 1-2, May 2006, 86-98.
- Pinheiro, P. & Lima, P. (2004). Bayesian Sensor Fusion for Cooperative Object Localization and World Modeling, *Proceedings of the 8th Conference on Intelligent Autonomous Systems, IAS-8*, Amsterdam, The Netherlands.
- Pohl, C. & van Genderen, J. L. (1998). Multisensor image fusion in remote sensing : concepts, methods and applications, *Int. J. Remote Sens.*, Vol. 19, No. 5, March 1998, 823-854.
- Rajan, D. & Chaudhuri, S. (2002), Data fusion techniques for super-resolution imaging, *Inf. Fusion*, Vol. 3, No. 1, March 2002, 25-38.
- Ranchin, T.; Aiazzi, B.; Alparone, L.; Baronti, S. & Wald, L. (2003). Image fusion - the ARSIS concept and some successful implementation schemes, *ISPRS J. Photogram. Remote Sens.*, Vol. 58, No. 1-2, June 2003, 4-18.
- Vancutsem, C.; Pekel, J.F.; Bogaert, P. & Defourny, P. (2007). Mean compositing, an alternative strategy for producing temporal syntheses. Concepts and performance assessment for SPOT VEGETATION time series. *Int. J. Remote Sens.*, Vol. 28, No. 22, November 2007, 5123-5141.
- Wang, Z.; Ziou, D.; Armenakis, C.; Li, D. & Li, Q. (2005). A comparative analysis of image fusion methods, *IEEE Trans. Geosci. Remote Sens.*, Vol. 43, No. 6, June 2005, 1391-1402.



- Wikle, C. K.; Milliff, R. F.; Nychka, D. & Berliner, L. M. (2001). Spatial-temporal hierarchical Bayesian modeling: Tropical ocean surface winds, *J. Am. Stat. Assoc.*, Vol. 96, No. 454, June 2001, 382-397, ISSN 0162-1459.
- Zhou, J.; Civco, D. L. & Silander, J. A. (1998). A wavelet transform method to merge Landsat TM and SPOT panchromatic data, *Int. J. Remote Sens.*, Vol. 19, No. 4, March 1998, 743-757.

# Multi-Sensor & Temporal Data Fusion for Cloud-Free Vegetation Index Composites

Bijay Shrestha, Charles O'Hara and Preeti Mali  
*GeoResources Institute, Mississippi State University*  
USA

## 1. Introduction

Remotely sensed data from satellite sensors such as Moderate Resolution Imaging Spectroradiometer (MODIS) and Advanced Very High Resolution Radiometer (AVHRR) provide almost daily global coverage. Satellite sensor data are used to create scientific data products that include surface reflectance, land surface temperature, sea-surface temperature and many others, as well as ancillary metadata like satellite viewing angle and data quality information. Vegetation indices, like Normalized Difference Vegetation Index (NDVI) (Jensen, 2000), derived from reflectance products of satellite sensors, are generally used as indicators of relative abundance and activity of green vegetation, often including leaf-area index, percentage green cover, chlorophyll content, green biomass, and absorbed photo-synthetically active radiation.

Frequently reflectance data products needed to create vegetation indices include undesired cloud, water vapour, aerosols, or other poor quality pixels. Continuous monitoring of occurrences such as droughts, frosts, floods, major fires, forest stress, or natural disasters are just a few of the circumstances when daily cloud-free vegetation index composites data are of high utility. The traditional approach to creating a single synthetic cloud-free image that includes ideal values selected from a temporal set of possibly cloudy satellite images collected over a continuous time period of interest is called multi-temporal compositing (MTC). MTC compositing is generally used to create vegetation indices images from data products with high temporal and low spatial resolution such as those produced by the National Oceanic and Atmospheric Administration's (NOAA) AVHRR sensor or NASA's MODIS (Justice, 1998). Various methods of MTC have been utilized to produce scientific data products including Maximum Value Compositing (MVC), Constrained View Maximum Value Compositing (CV-MVC) (Cihlar et. al., 1994, Heute et. al., 1999), and CV-MVC which incorporates sensor data quality information.

The motivation for investigating multi-sensor and temporal fusion for creating high-temporal frequency composites is to overcome the limitations of single-sensor MTC methods and deliver continuous monitoring capabilities that exceed the temporal frequency of currently available 8-day, 10-day, 14-day, and 16-day composite vegetation index data products. Currently available composite products do not provide sufficient frequency and temporal detail to capture and quantify important events, do not deliver data for continuous environmental monitoring, and provide temporally sparse inputs precluding effective agricultural productivity modelling.

In this chapter, new approaches are presented for creating high-temporal resolution vegetation index data products by multi-sensor and temporal fusion (MSTF) (Shrestha et al., 2006). MSTF methods are detailed that employ rule-based image fusion methods to combine data streams from similar sensors to deliver almost daily cloud-free high-temporal resolution vegetation index composites. Results of daily MSTF cloud-free vegetation index compositing are presented to illustrate their significance and potential operational utility for 1) providing regional vegetation and ecosystem condition monitoring as applied to a disaster event such as Hurricane Katrina, and 2) providing bioproductivity monitoring and crop modelling insight for an agricultural study area of interest with large crop production fields that include a plurality of pixels. In the earlier case, daily data sets are shown as well as cross-platform fused products providing visible indication of decreasing greenness, whereas in the latter case, the MSTF approach is implemented, products are demonstrated, phenological growth stages of crops are examined using daily data products, and results are compared to traditional standard scientific data product produced by MTC methods.

### 1.1 NDVI

Vegetation indices are conventionally used as a representative of vegetation that characterizes the vegetation vigor (Rouse et al., 1974). Vegetation indices are defined as dimensionless, radiometric measures that function as indicators of relative abundance and activity of green vegetation, often including leaf-area index, percentage green cover, chlorophyll content, green biomass, and absorbed photo-synthetically active radiation (Jensen, 2000). In this study, we have used Normalized Difference Vegetation Index (NDVI); the most widely used form of VI, which was introduced by Deering (Deering, 1978) and Tucker (Tucker, 1979). The principle underlying NDVI is the strong reflectance of healthy, chlorophyll-based vegetation at near-infrared wavelengths and its relatively weak reflectance in the visible red. NDVI is simply defined as the ratio of the difference between these reflectance normalized by their sum as shown by the equation 1. This yields a dimensionless quantity ranging in theory from -1 to 1, but in practice the lowest value seldom falls below -0.25. The value increases from -1 to +1 with the increase in vegetation. The clouds are in the lower end of the NDVI value range.

$$\text{NDVI} = \frac{\text{NearIR} - R}{\text{NearIR} + R} \quad (1)$$

The NDVI is calculated using equation 1, where NearIR represents near infra-red reflectance and R represents red reflectance. Unfortunately, these images frequently include undesired cloud and water cover. Areas of cloud or water cover preclude analysis and interpretation of terrestrial land cover, vegetation vigor, and/or analysis of change. Figure 1 shows NDVI images of Mississippi and Arkansas, US, created using reflectance data from AVHRR (a) and Terra MODIS (b) on May 7, 2004.

This chapter presents multi-sensor and temporal fusion (MSTF) of NDVI datasets from the MODIS sensor, a key NASA sensor system aboard two satellites, the Terra (EOS AM) and Aqua (EOS PM). Terra and Aqua MODIS collect image data for the entire Earth's surface every 1 to 2 days, acquiring data in 36 spectral bands, or groups of wavelengths. The spatial resolution of MODIS images varies from 250m x 250 m to 1000m x 1000 m.

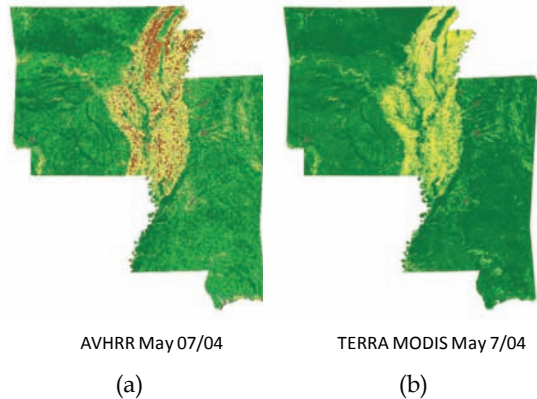


Fig. 1. Color coded NDVI for states of Mississippi and Arkansas, US from AVHRR (a) and Terra MODIS (b) sensors for 7 May 2004 (Mali et al., 2005).

## 2. Compositing

To more effectively monitor the dynamics of changing conditions on the surface of our planet, removing the frequent and extensive cloud cover that obscures daily observations over many parts of Earth's is the primary motivating reason for developing methods for producing spatially and temporally continuous and consistent NDVI images. The traditional approach of creating a single synthetic cloud-free image that includes ideal values selected from a temporal set of possibly cloudy satellite images collected over a continuous time period of interest is called multi-temporal compositing (MTC). There are several MTC methods that are generally based on the temporal NDVI values, satellite view angle, and also the quality flags for the pixel.

The MTC process of selecting pixels with highest NDVI value for each individual pixel across a set of temporal collection for compositing purposes is known as Maximum Value Compositing (MVC). MVC is the procedure used to generate composited AVHRR-NDVI product (Holben, 1986). Presence of a high concentration of water vapour lowers the near infrared channel response and thereby lowers the derived NDVI value. Therefore, the highest NDVI value, also referred to as the greenest pixel, is presumed to select the observation under conditions when the atmosphere contained the least amount of water vapour. The shortest path length for reflected radiation occurs at nadir, and it is least affected by water vapour concentration (USGS). Furthermore, pixel size increases with view angle; therefore nadir-view pixels are considered optimal because they possess minimal spatial distortions (Gao et al., 2003). In addition, high-angle views enable the relative altitude of vegetation above the ground to present more leaf area and less soil cover to the observing system sensor. For all of these reasons, MVC processing approaches have been refined to include constraints that remove such bias by employing a scan angle or zenith angle constraint. In this refined approach to MVC, the selection criteria allows the use of pixels observed at angles near nadir during the compositing process and discards pixels observed at angles far from nadir. MODIS-VI compositing algorithm uses either reflectance based BRDF or constraint view maximum value compositing (CV-MVC) algorithm over a period of 16 days (Huete et al., 1999).

The presence of clouds in the daily data products precludes the use of NDVI for continuous vegetation monitoring using satellite observation. Additionally, the traditional MVC is a representative NDVI (highest value) for a given temporal range is selected for 8/10/16 days and up to 30 days period. For example, the NASA standard scientific data product, MOD13 (LPDAAC, 2008) is a comprehensive standard vegetation index product that includes NDVI composited over 16 day period. These standard cloud-free composites of NDVI have high utility/usefulness for many applications where regular continuous vegetation monitoring may not be required. However, in certain cases where regular vegetation monitoring is required, an 8/10/16 day or longer temporal range may not be sufficient. For example, in the case of crop monitoring, changes in NDVI values can provide significant insight to crop vigor and provide information for identifying changes from one stage of vegetation growth to other. Therefore, there is a need for cloud-free NDVI products that provide high temporal resolution. Multi-sensor and temporal fusion (MSTF) provides methods needed to generate daily cloud-free NDVI products with broad areas of useful application.

### 3. Multi-Sensor and Temporal Fusion (MSTF)

The MODIS sensor system is onboard two satellites Aqua and Terra and has high temporal resolution providing almost daily revisits around the globe. Therefore, using the MODIS sensors onboard the two satellites, NDVI images created from MODIS land surface reflectance datasets (MOD09) provide an ideal opportunity for multi-sensor fusion. In addition to observation reflectance data, the LPDAAC also provides additional metadata associated with the datasets for Global Geolocation Angles (MODMGGAD) and Quality Assurance (MODGST). The fusion algorithm uses the associated MODIS geolocation angles and MODIS quality assurance (QA) metadata (extracted from MODGST) to differentiate between land, water, cloud, and snow observation.

The principles behind the presented rule-based algorithm for multi-sensor and temporal fusion are a) that observations selected for the day of interest are preferred, but values may be considered that persist from observations made over previous days; b) that pixels with lower view angles contain less noise; and c) that associated MODIS QA metadata may be used to select land observations. After the preprocessing steps to extract NDVI, zenith angle, and quality data, a stepwise temporal selection process is followed to identify observation pixels, from both Aqua and Terra, which fulfill the necessary criteria for fusion. The criteria that are enforced are for i) a zenith angle for the pixel less than  $48^\circ$ , and ii) a quality code for the underlying pixel indicating classification as a clear land observation. Thereby, pixels classified as water, cloud, snow, and no data values are exempted from consideration for fusion. The zenith angle threshold (MAX\_ANGLE) of  $48^\circ$  is used, such that only pixels with associated zenith angles less than  $48^\circ$  are selected. The zenith angle threshold was chosen to reduce striping effects that were observed in the composites created using higher zenith angles (Mali et al., 2005). Given the view angles and quality metadata for each pixel of the images from different satellites, the following pseudo code illustrates the rule-based fusion (O'Hara et al., 2008, Shrestha et al., 2005).

```

For day from 0 to MAX_DAY
  if (angle is less than MAX_ANGLE) AND (mask is MASK_LAND)
    select pixel, zenith angle, mask
    compute confidence
  else

```

```
        get next pixel
    If no pixel selected
        select pixel with highest NDVI, zenith angle, mask
        assign least confidence
```

In both the cases that are presented in this chapter, a six day temporal window (MAX\_DAY), 48° zenith angle (MAX\_ANGLE), and the quality mask observation classified clear observation of land (MASK\_LAND) are used. When both of the observations for a day meet the required constraints, either the observation with lower zenith angle or the observation with higher NDVI can be selected for fusion. In this case, we have chosen observation with higher NDVI.

## 4. Results and analysis

Using the technique for multi-sensor and temporal fusion described in earlier section, experiments were conducted to illustrate use cases for vegetation and ecosystem condition monitoring for a disaster event and for agricultural crop bioproductivity monitoring and modelling. The locations for the use cases considered two sites - 1) the Mississippi Delta, USA and 2) the Pampas of Argentina. Use case results are presented in the following sections.

### 4.1 Mississippi delta, USA

The texts and results in this section are extracted from (O'Hara et al., 2008). This section illustrates the use case for MSTF fused NDVI created of Mississippi Delta region in Unites States during the time frame of Hurricane Katrina. For this use case, the objective was to utilize MSTF to demonstrate temporal compositing methods with significant implications for monitoring continuous vegetation status. To accomplish this objective, it was decided that a major meteorological event would provide unique insight as to whether the application would enable status tracking, identification, and potentially quantification of pre-event conditions, event-related stress, and post-event recovery of greenness. To accomplish this, an area within Mississippi was selected above the coast, but within the areas impacted by Hurricane Katrina. The ability to remove clouds and characterize the daily vegetation conditions before and after a major hurricane was deemed to be a good application for testing the ability to remove heavy or complete cloud cover for a desired geographic area and a temporal interval of interest (O'Hara et al., 2008).

For a thirty-two (32) day period from August 20 to September 20, during a time frame covering the temporal period prior to as well as after Hurricane Katrina (August 29<sup>th</sup>, 2005), MODIS - Aqua and Terra datasets were downloaded for the Area of Interest (AOI). The AOI is focused in the Mississippi Delta region, and corresponds to MODIS grid H10V5. These daily data were downloaded, pre-processed to NDVI and associated metadata of zenith angle and quality, and were used to compute fused daily NDVI. The daily MODIS reflectance, geolocation angle, and quality assurance datasets were downloaded from the NASA LPDAAC.

Figures 2 and 3 represent the color coded daily NDVI images computed from Aqua and Terra MODIS land surface reflectance datasets for August 20, 2005 to September 20, 2005. Figure 4 represents the fused dataset created using multi-sensor and temporal fusion for the same time period.

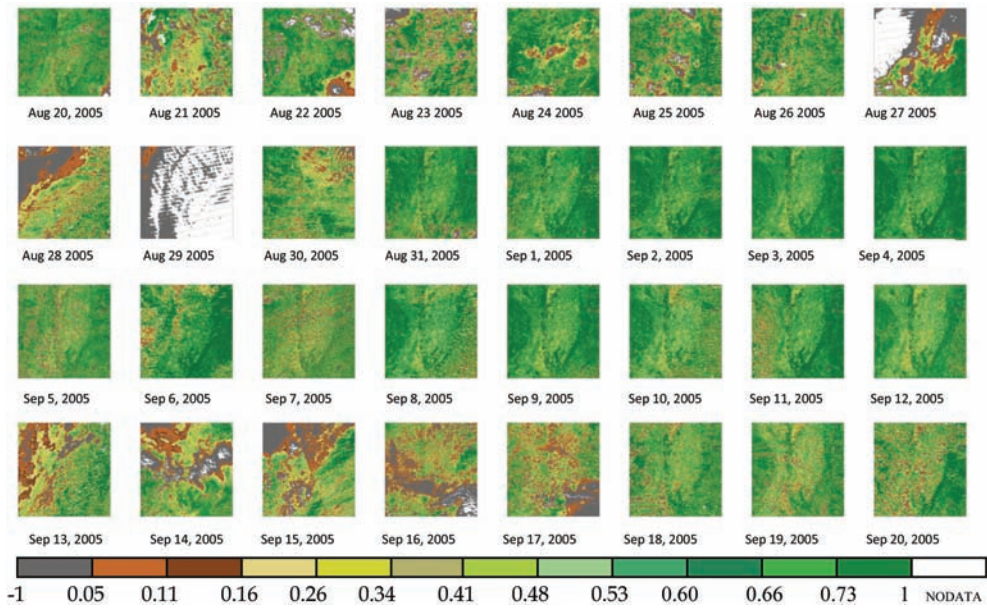


Fig. 2. Color coded Aqua NDVI Data for 32 days (Aug 20 – Sep 20, 2005). Here Grey color shows cloud cover, the low NDVI values are in orange and yellow colors, the increase in NDVI is shown by increase in darkness of green color, and the white color represent the ‘No Data’ values; and are related to the NDVI values represented in the color bar (O’Hara et al., 2008).

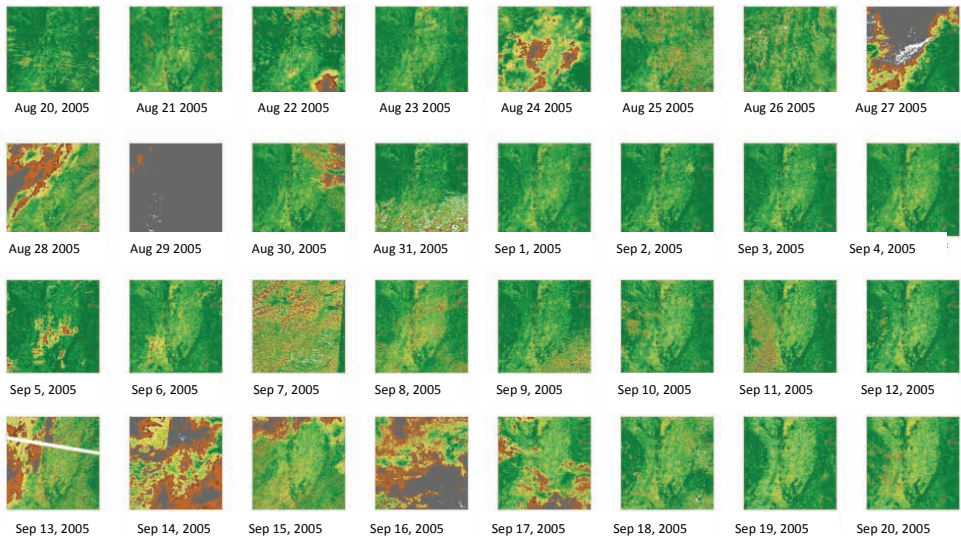


Fig. 3. Color coded Terra NDVI data for 32 days (Aug 20- Sep 20, 2005). The color mapping is the same as employed in Figure 2 (O’Hara et al., 2008).

Through visual inspection of daily Aqua, Terra, and fused NDVI images, from Figures 2, 3, and 4, initial indications of cleanness, completeness, and continuity of data coverage as well as general trends in greenness are observed in the fused images compared to the cloud obscured and noisy appearance of NDVI products created from daily reflectance data. Here, visual representations of the regional NDVI datasets are provided. Detailed field level zonal analyses of fused datasets with respect to daily and composited standard single sensor MODIS datasets will be published in upcoming publication (O'Hara et al., 2008). Visual inspection of MSTF results show a clear trend, when viewing along a diagonal from top-left to lower right (earliest date to latest date), of general decrease in greenness and increase in yellow content. This clearly illustrates the usefulness of the products to provide improved understanding of the significant decreases in vegetation vigor due to loss of leaves, downed trees, and general vegetation stress as a result of Hurricane Katrina event. In the next section, the use of MSTF fused NDVI for crop productivity and continuous vegetation monitoring of large agricultural fields in Argentina is presented.

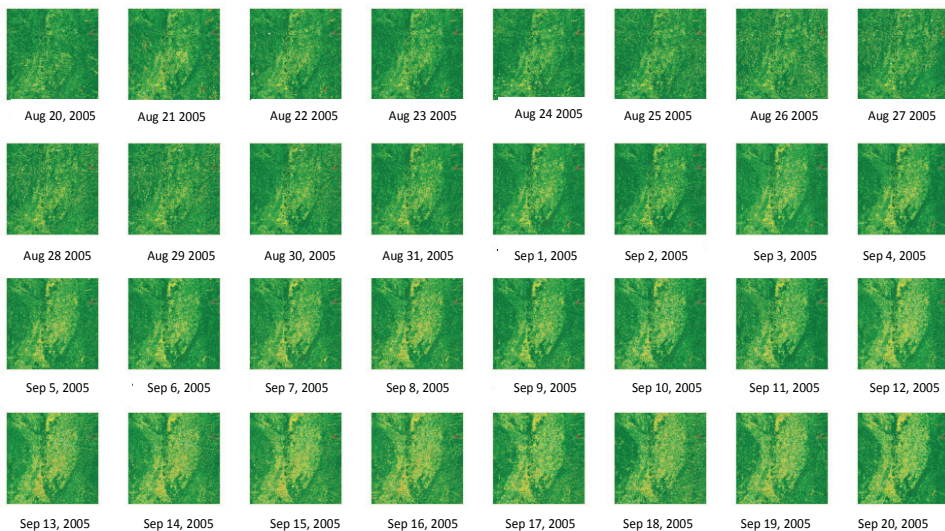


Fig. 4. Color coded fused output NDVI data for 32 days (Aug 20- Sep 20, 2005). The color mapping is the same as employed in Figure 2 (O'Hara et al., 2008).

#### 4.2 The pampas of Argentina

The second use case demonstrates the use of fused daily NDVI data to create data products that deliver greatly enhanced capabilities to analyze current conditions and long-term crop signatures. These signatures as shown as greenness curves that depict representative vegetation conditions characterized by NDVI and enable improved understanding of crop phenological growth states. As studies have shown in the past, multi-temporal observation data may be analyzed to detect phenological changes useful for crop classification (Haralick, et al., 1980) as well as to provide signature curves of crop growth which utilize NDVI to model crop growth, phenological growth stages, and crop maturity. This approach is



presented in terms that illustrate utility for delivering enhanced understanding of specific crop types as well as development and vigor over time during a crop season as well as over multiple seasons to provide insight about crop rotation, management practices, double cropping, and timing of planting and maturity of crops.

This use case demonstrates the use of fused NDVI Data products derived from satellite observations of Aqua and Terra MODIS to provide a major observational source of field level vegetation data.

South America is considered as the major soybean growing region of the world. For that reason, Argentina was chosen as the main area of interest for this experiment. The study area lies in the *Pampas*, which includes the Cordoba region and other nearby areas, is a major soybean farming region within Argentina. Figure 5 shows the eight field sites in Argentina. The field sites were chosen with the help of a local expert such that the MODIS tile with grid ID H12V12 contained all the field sites. For each field site, large soybean fields were chosen, and the field boundaries were decided and validated. By using large crop production fields it was insured that the fields included a plurality of pixels in coarse-resolution (250m x 250m) MODIS reflectance datasets. Most of the fields selected are more than 50 hectares in area to comply with coarse resolution of MODIS datasets. For this chapter, we show the results from one of the sites in Marco Juarez. Figure 6 shows the three field boundaries at Marco Juarez in a) high resolution AWiFS image and b) low resolution fused MODIS image.

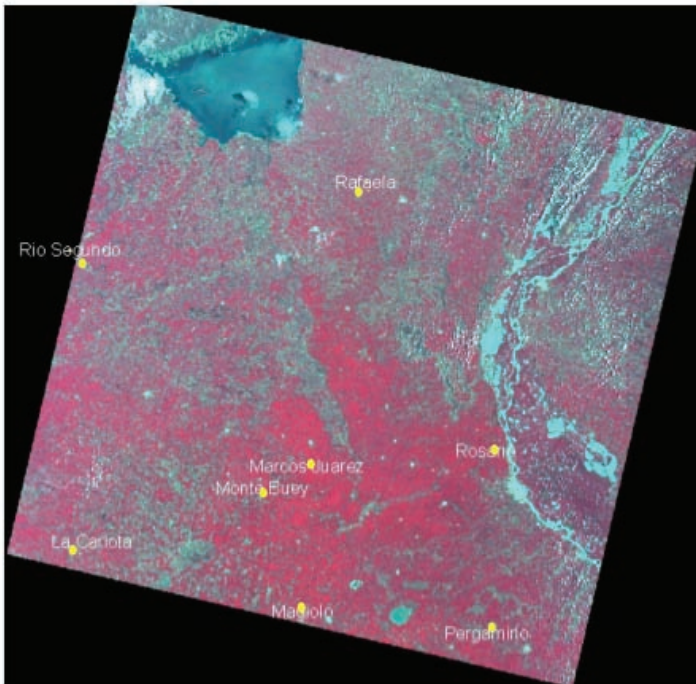


Fig. 5. The eight field sites in Argentina shown overlaying AWiFS imagery (56m resolution).

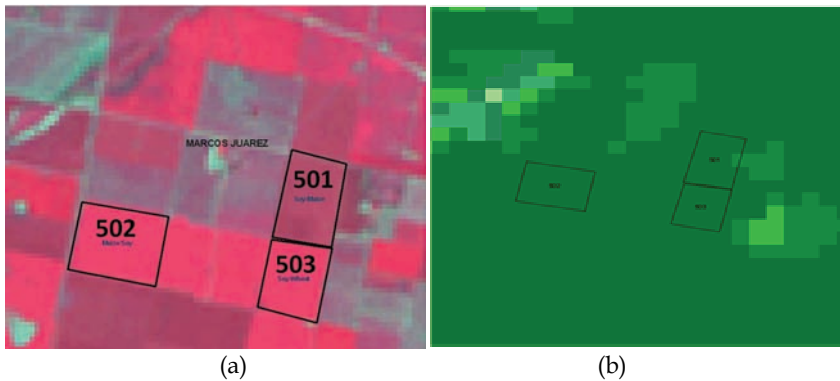


Fig. 6. (a) Three digitized field boundaries shown for Marcos Juarez overlaying high resolution AWIFS false color composite. The fields are numbered 501, 502, and 503 for identification purposes (b) Fields in Marco Juarez shown in lower resolution (250m x 250m) color coded fused MODIS imagery for February 1, 2007.

Daily MODIS reflectance (MOD09), Geolocation Angle (MODMGGAD), and Quality Assurance (MODGST) data for the H12V12 tile were acquired for Aqua and Terra MODIS for three years (August 2004-August 2007). These daily Aqua and Terra MODIS datasets were pre-processed to NDVI and associated metadata for quality assurance were computed. The pre-processed daily datasets were then used to compute fused daily NDVI using the multi-sensor and temporal fusion (MSTF). Figure 6 b) shows the fused NDVI product for February 1, 2007 for Marco Juarez with the overlay of three fields in the region.

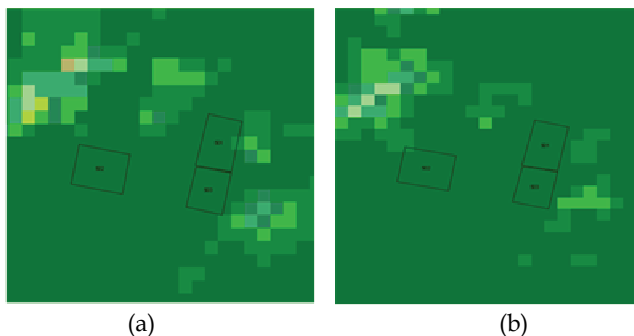


Fig. 7. (a) Color coded Aqua 16-day composite MYD13 NDVI for January 25- February 10, 2007 (b) Color coded Terra 16 day-composite MOD13 NDVI for January 17- February 2, 2007 showing the fields in Marco Juarez. The color mapping is the same as employed in Figure 2.

To quantify the vegetation vigor of a field that is represented by multiple NDVI pixels, an average of all the NDVI pixel values within the area of the field boundary is calculated. After the fusion of the datasets using MSFT, daily average NDVI value for each field was calculated from the fused daily datasets. Using the daily average NDVI values for the field, temporal NDVI signatures were created for each field. The temporal NDVI signatures for three fields in Marco Juarez (fig. 6) are shown in figures 8, 9, and 10. The temporal NDVI signatures represent the phenology curves for the changing vegetation on the field. To make

comparisons with standard datasets, average NDVI value were also calculated from 16-day MOD13 standard composites from Aqua and Terra. Figure 7 a) and b) show the MOD13 composites for Aqua and Terra that contain the February 1, 2007 respectively.

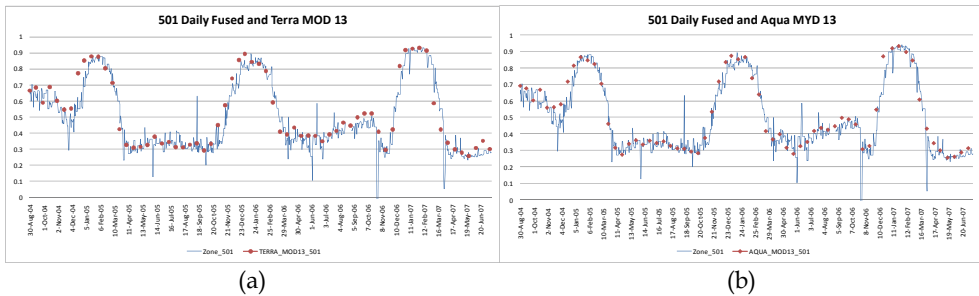


Fig. 8. Temporal signatures created using fused NDVI for field 501 at Marco Juarez. The red circles on (a) represent mean NDVI for the field from 16 day MOD13 composites from Terra and red squares on (b) represent mean NDVI from 16 day MOD13 composites from Aqua.

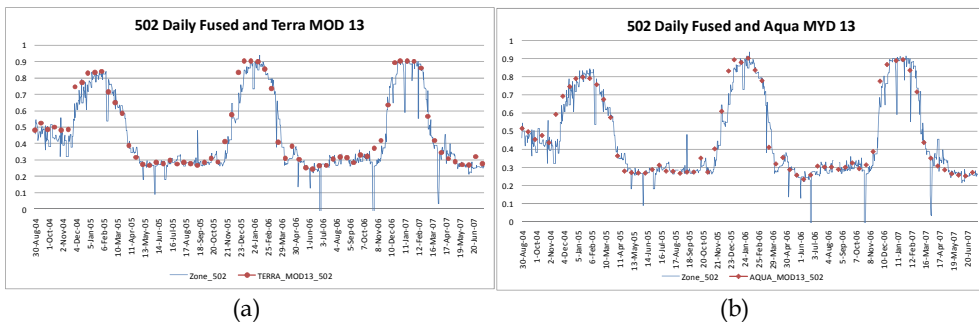


Fig. 9. Temporal signatures created using fused NDVI for a field 502 at Marco Juarez. The red circles on (a) represent mean NDVI for the field from 16 day MOD13 composites from Terra and red squares on (b) represent mean NDVI from 16 day MOD13 composites from Aqua.

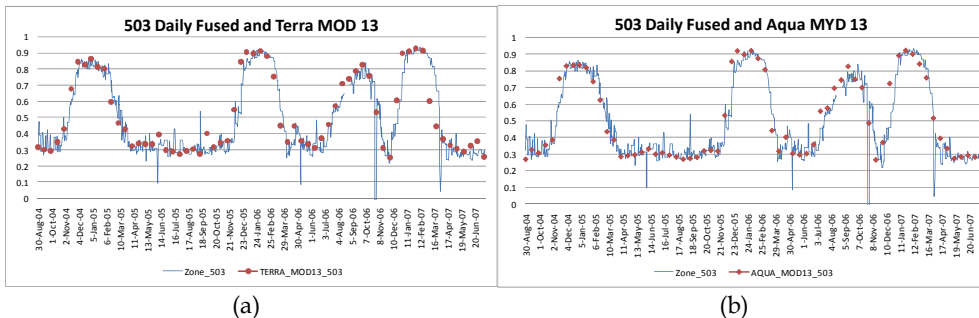


Fig. 10. Temporal signatures created using fused NDVI for a field 503 at Marco Juarez. The red circles on (a) represent mean NDVI for the field from 16 day MOD13 composites from Terra and red squares on (b) represent mean NDVI from 16 day MOD13 composites from Aqua.

In figures 8, 9 and 10, the NDVI temporal signatures using mean NDVI for three different fields in Marco Juarez are presented for August 2004 - July 2007. Also presented are the Aqua and Terra 16 day MOD13 average NDVI values for the fields using the red circles and red squares respectively. In periods outside the crop season, low NDVI values represent fallow conditions when there is little or no greenness or vegetation within the field boundary. At the beginning of the crop season, seeds are planted, crops emerge, leaf area accumulates, and vegetation growth may be monitored by tracking changes and increases in NDVI values. As the crop matures, NDVI values increase, and with higher crop vigor, NDVI values continue to increase up to the maturation stage for the plants at which point leaf growth terminates and observed greenness starts to decrease. Crop harvesting and fallow periods (for out of seasons) are part of the typical cycle for agricultural regions and this phenomena is well represented in 8,9, and 10. The NDVI values for all the fields in the figures exhibit fallow periods between the crop seasons during which vegetation growth can be readily observed for the three (3) year period presented. It can also be seen from the figures that the fused NDVI follows the trend demonstrated by the 16 day MOD13 NDVI composite values. The fused daily NDVI, however, provide a much greater detail of change in NDVI than the 16-day composites.

For field 503 shown in figure 10, for the planting season for 2006-2007, two discreet crop signatures are present as evidenced by the significant and separate increases in NDVI during that crop season. The two readily observed "humps" represent double cropping, which means that two crops were planted during the same season. Typical of management practices in the region, double cropping is usually conducted wherein an early planting of wheat is followed by a late planting of soybean. For the rest of the fields (501 and 502), single cropping practices may be observed across the three years presented.

This use case presents results indicating that MSTF provides unique and previously unavailable capabilities to continuously monitor and analyze crop conditions, development, management practices, maturity, and overall productivity for fields, farms, and regions of agricultural interest. The method of MSTF presented is intended to demonstrate robust results that can be obtained by a simple rule-based implementation of cross-platform fusion. The method presented in no way alters, modifies, or changes observation values. Therefore, for any observation system, there will be noise present that accompanies the desired "signal" portion of the results. The degree of agreement between the MSTF results and standard NASA MOD13 products is remarkable and clearly illustrates the powerful capabilities that MSTF delivers for continuous temporal monitoring of vegetation conditions. Clearly, there are significant opportunities to improve on these results, and this can be readily accomplished by implementing approaches to noise removal. It can be seen from the figures that even on the relatively cleaner fused datasets, there is some noise present in the NDVI temporal signatures. Further research and implementation testing may be conducted to evaluate different filtering methodologies to denoise the temporal NDVI signatures. Approaches that may be readily implemented include simple median filtering, moving-average, or wavelet based denoising (Bruce et al., 2006) or using Savitzky-Golay Filtering (Chen et al., 2004). Regardless of the noise component, the MSTF results presented in this use case clearly illustrate the useful nature of daily products and the utility of this method to deliver needed products for agricultural applications.

## 5. Conclusion

Vegetation indices, like Normalized Difference Vegetation Index (NDVI), derived from reflectance products of satellite sensors with high temporal resolution (e.g. MODIS) are very useful resource for vegetation monitoring. However, these images frequently include undesired cloud and water cover. Such areas of cloud or water cover preclude analysis and interpretation of terrestrial land cover, vegetation vigor, and/or analysis of change. Multi-sensor and temporal fusion (MSTF) is a useful tool to perform fusion to remove clouds from NDVI datasets originating from multiple sensors that collect observations at slightly different times of each day across a geographic area of interest (and may be selected for a temporal window of interest). MSTF is performed by using both the reflectance observation data as well as the available ancillary metadata for the available reflectance datasets. The cloud-free daily NDVI composite data products obtained using the MSTF datasets provide an advantage for a wide variety of applications, such as disaster monitoring, vegetation stress monitoring, crop health monitoring, and regional or global bio-productivity modelling that would benefit greatly by the availability of higher temporal resolution NDVI data.

In this chapter, two use case examples were presented clearly showing the utility of MSTF for the fusion of NDVI datasets from Aqua and Terra MODIS. In the first use case, fusion based NDVI products were created and illustrated the usefulness of the process to provide cloud-free image datasets during the time frame of Hurricane Katrina. This use case clearly shows that MSTF provides capabilities of value to vegetation and ecosystem monitoring and change analysis. In the second use case, fusion based NDVI products were used for remote vegetation observation for appropriately sized agricultural fields in Argentina to model and quantify crop growth and development. MODIS NDVI products and field boundaries were employed to extract data that were presented in the form of crop growth phenology curves which are evidence of the ability to show in high temporal detail the stage of growth and maturity of crops. This approach has obvious application to detecting the onset of greenness that may be used to refine estimations of planting date, crop growth status, and maturity. In both of the presented use cases, a 6 day maximum temporal window was chosen. The presented temporal compositing period is provides daily data and can be observed to be highly complementary to standard NASA data sources such as 16 day MOD13 composites. It is also observed that the fused products exhibit the trend of NDVI change the MOD13 composites and provide a greater detail of change in NDVI than the 16-day composites. Here, we have used datasets from Aqua and Terra MODIS which have the same spatial resolution and have same spectral characteristics.

Further studies may be conducted to consider refinements to the MSTF process. The easiest pathway to improved products lies in considering and evaluating methods to remove noise and enhance the signal component of MSTF products. More challenging improvements may be considered aimed at increasing the richness of MSTF products by evaluating how best to include in MSTF processing data streams from additional sensors with different spectral, spatial, and temporal characteristics. In all MSTF provides a new set of capabilities for improving capabilities to observe the dynamic and changing conditions of the earth's surface. This data fusion approach offer major opportunities for extensible and highly transferable application as well as operational implementations to meet a broad variety of earth science and societal needs.

## 6. Acknowledgements

This research was funded by Mississippi Research Consortium NASA ISS Agriculture Efficiency project NAS13-03032; NNS06AA65D and NASA RPC Agriculture Efficiency project: NAS13-03032; NNS06AA69D.

## 7. References

- Bruce, L.M., Mathur, A., Byrd, J.D. (2006), Denoising and Wavelet-Based Feature Extraction of MODIS Multi-Temporal Vegetation Signatures, *GIScience & Remote Sensing*, vol. 43, pp. 170-180, 2006.
- Chen, J., Jonsson, P., Tamura, M., Gu, Z.; Matsushita, B., Eklundh, L. (2004), A simple method for reconstructing a high-quality NDVI time-series dataset based on the Savitzky-Golay filter, *Remote Sensing of Environment*, Vol. 91, No. 3-4. (30 June 2004), pp. 332-344.
- Cihlar, J., Manak, D., and D'Iorio, M. (1994), Evaluation of Compositing Algorithms for AVHRR Data Over Land, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 32, no. 2, March 1994, pp. 427 - 437.
- Deering, D.W. (1978), Rangeland reflectance characteristics measured by aircraft and spacecraft sensors. Ph.D. Diss. Texas A&M Univ., College Station, 338p.
- Haralick, R.M., Hlavka, C. A., Yokoyama, R., and Carlyle, S. M. (1980), Spectral-Temporal Classification Using Vegetation Phenology, *IEEE Transactions on Geoscience and Remote Sensing*, Volume GE-18, Issue 2, April, 1980.
- Holben, B. N. (1986), Characteristics of Maximum-Value Composite Images from Temporal AVHRR Data, *International Journal of Remote Sensing*, vol. 7, no. 11, pp. 1417-1434.
- Huete, A., Justice, C., Leeuwen, W. V. (1999), (MODIS Vegetation Index (MOD 13) Algorithm Theoretical Basis Document (ATBD)
- Jensen, J. R. (2000), *Remote Sensing of the Environment: An Earth Resource Perspective* (Prentice Hall).
- Justice, C. O., Vermote, E., Townshend, J. R. G., Defries, R., Roy, D. P., Hall, D. K., Salomonson, V. V., Privette, J. L., Riggs, G., Strahler, A., Lucht, W., Myneni, Knyazikhin, Y., Running, S. W., Nemani, R. R., Wan, Z., Huete, A. R., Leeuwen, W. van, Wolfe, R. E., Giglio, L., Muller, J. P., Lewis, P., and Barnsley, M. J. (1998), The Moderate Resolution Imaging Spectroradiometer (MODIS): Land remote sensing for global change research, *IEEE Trans. Geosci. Remote Sensing*, vol. 36, pp. 1228-1249.
- LPDAAC (Land Processes Distributed Active Archive Center), 2008, MODIS 1km Daily Surface Reflectance State QA Science Data Sets, Available online on [http://lpdaac.usgs.gov/modis/moyd09gst\\_qa\\_v4.asp](http://lpdaac.usgs.gov/modis/moyd09gst_qa_v4.asp) (accessed 10 July 2008)
- Mali, P., O'Hara, C. G, Shrestha, B., and Vijayraj, V. (2005), Use and Analysis of Temporal Map Algebra for Vegetation Index Compositing, *Analysis of Multi-Temporal Remote Sensing Images, 2005 International Workshop on the*, May, pp. 90-94.
- NOAA (2000), NOAA KLM User's Guide, U.S. Department of Commerce, National Oceanic and Atmospheric Administration, National Environmental Satellite, Data, and Information Service.

- O'Hara, C. G., Shrestha, B., and Mali, P. (2008), Fusion of Cross-Platform Satellite Data and Metadata for High-Temporal Resolution Cloud-Free Compositing, *International Journal of Remote Sensing* (submitted)
- Rouse, Jr., J.W., Haas, R. H., Deering, D. W., Schell, J. A., & Harlan, J. C. (1974). Monitoring the vernal advancement and retrogradation (greenwave effect) of natural vegetation. NASA/GSFC Type III Final Rpt. Greenbelt, MD, USA: Goddard Space Flight Center.
- Shrestha, B., O'Hara, C. G., and Mali, P. (2006), Data Fusion, De-noising, and Filtering to Produce Cloud-Free High Quality Temporal Composites Employing Parallel Temporal Map Algebra, *AIPR 2006: Washington DC, October 11-13*.
- Tucker, C.J. (1979), Red and Photographic Infrared Linear Combinations for Monitoring Vegetation, *Remote Sensing of Environment*, 8(2), 127-150.

# Three Strategies for Fusion of Land Cover Classification Results of Polarimetric SAR Data

Nada Milisavljević<sup>1</sup>, Isabelle Bloch<sup>2</sup>, Vito Alberga<sup>1</sup> and Giuseppe Satalino<sup>3</sup>

<sup>1</sup>Signal and Image Centre, Royal Military Academy Brussels,

<sup>2</sup>TELECOM ParisTech (ENST) CNRS UMR 5141 LTCI Paris,

<sup>3</sup>Institute of Intelligent Systems for Automations (ISSIA)

National Research Council (CNR) Bari,

<sup>1</sup>Belgium

<sup>2</sup>France

<sup>3</sup>Italy

## 1. Introduction

A fully polarimetric synthetic aperture radar (SAR) is a device that is able to transmit and receive both orthogonal (horizontal and vertical) components of an electromagnetic wave (Touzi et al., 2004). Its signals include the magnitude and phase information, rendered as complex variables. Many classification algorithms have been proposed for SAR images (Cloude & Pottier, 1997; Ferro-Famil et al., 2001; Hoekman & Vissers, 2003; Kersten et al., 2005; Lee et al., 1994; Lee et al., 1999, Lee et al., 2004), that can be grouped into three main types (Chen et al., 2003): 1) algorithms based on image processing techniques, 2) algorithms based on a statistical model, 3) algorithms based on the scattering mechanism of the electromagnetic waves. Our interest is in the first type, since such algorithms permit a general approach to the images, potentially after investigating the physical properties of natural media.

In a previous work (Alberga et al., 2006), several ways of representing polarimetric SAR data (Cloude & Pottier, 1996; Cameron et al., 1996; Cloude & Pottier, 1997; Freeman & Durden, 1998) have been analyzed and their usefulness for land cover classification compared. The classifiers used were the minimum distance classifier, the maximum likelihood classifier and a neural network - the Multi-Layer Perceptron (MLP), trained by the Back-Propagation (BP) learning rule. The MLP outperformed the other two classifiers. In addition, the MLP does not need any *a priori* knowledge on the statistics of the input data, thus it can be applied to any possible polarimetric observable, permitting an unbiased use of the classification results (Benediktsson, et al., 1990). For these reasons, the MLP is used also in this work.

The accuracy of the classification of a given polarimetric representation (*i.e.*, set of polarimetric parameters) was taken in (Alberga et al., 2006) as a measure of the usefulness of that set and compared with the accuracies obtained using other representations of the data. The substantial equivalence of the parameters in these terms was shown. However, no attempt was made to take advantage of the complementary information provided by the



different parameters nor was their fusion performed in order to improve the classification performance. This is the scope of the present research.

Papers on polarimetric data fusion can be rarely found in the literature. In (Mascle et al., 1997; Le Hégarat-Mascle et al., 1998), an unsupervised classification is performed on each image separately and the results are fused using a strategy based on belief functions (Shafer, 1976; Smets, 1990).

In this paper, classification results coming from several sets of polarimetric parameters are fused following different strategies (SMART, 2004), two based on the belief function framework (Shafer, 1976; Smets, 1990) and one based on fuzzy theory (Dubois et al., 1999). After decision combination on pixel level, as a final step of the fusion module, a spatial regularization is performed.

In Section 2, the sets of polarimetric parameters under consideration are described, and the classifier used is presented in Section 3. The main aspects of the belief function theory are underlined in Section 4. The applied fusion approaches are detailed in Section 5. Section 6 presents the experimental data and the characteristics of the analyzed scene, while Section 7 reports on the results of data fusion and on their comparison with the classification results of each set of polarimetric parameters. Finally, Section 8 is devoted to conclusions, followed by acknowledgements and references.

## 2. Investigated polarimetric parameters

Fully polarimetric radars can transmit and receive both orthogonal components of an electromagnetic wave (Touzi et al., 2004). Thus, its vector nature is taken into account ensuring that complete scattering information carried by radar echo signals may be used for target detection and identification. Within this framework, different representations exist of the scattering interactions, such as, *e.g.*, the  $2 \times 2$  scattering matrix [S] or the higher order ones, the coherency and covariance matrices.

In the field of terrain classification, the choice of a given representation has been related to considerations on both the statistics of the data and the physics of the scattering mechanisms. In particular, the use of incoherent parameters (*i.e.*, those derived from the second order matrices) has become predominant with respect to that of the coherent ones related to the [S] matrix. For multi-look data represented as covariance or coherency matrices, Lee et al. (Lee et al. 1994) defined a distance measure for the membership of a pixel to a class based on the complex Wishart distribution and this measure could be incorporated in several classification algorithms (Ferro-Famil et al., 2001; Lee et al. 1999). Accordingly, only second order representations were considered when operating these classifiers (Ferro-Famil et al., 2001; Lee et al. 2004); not much attention was paid, until now, to coherent parameters.

Different viewpoints are adopted when choosing a coherent or incoherent representation: in the first case, the hypothesis is made that the scattering interaction within a resolution cell involves only one or few point scatterers. Their phase can then be measured and analyzed or taken into account when deriving new parameters. Moreover, in the case of independent scattering mechanisms, these may be singularly recognized (coherent target decomposition theorems perform just this separation). Given their definition, the reliability of coherent methods is higher when dealing with man-made artifacts, which correspond better to such a scenario. Thus, in our specific experimental case, we expect good classification performance of coherent representations especially for urban areas or other targets with a “stable”

behavior. On the contrary, incoherent methods perform immediately an averaging of the returned signals that yields the loss of the direct reference to the phase of the elementary targets. However, they provide a statistically sounder description of the behavior of "dynamic" natural environments. For these environments, target decomposition theorems may be applied and permit to recognize different scatterers. Namely, they separate the responses of different mechanisms considered as "average" ones (*e.g.*, within forested areas they can distinguish between distributed volumes and surface scatterers).

Since coherent representations are better suited to targets approximating ideal point scatterers (as it happens in urban areas) and incoherent ones to randomly distributed targets (forests or fields), it can be expected that their use would provide complementary information that, by classification applications, could be better exploited by means of a fusion approach.

The representations reported in the following subsections have been taken into account in order to benefit from the different types of information they provide as well as because they are the most often used and known.

### 2.1 First and second order matrices

When a horizontally or vertically polarized wave is incident upon a target, the backscattered wave can have contributions in both horizontal and vertical polarizations. Thus, the backscattering of the target can be completely described by a scattering matrix:

$$[\mathbf{S}] = \begin{bmatrix} S_{hh} & S_{hv} \\ S_{vh} & S_{vv} \end{bmatrix}. \quad (1)$$

In other words, the four complex elements of the scattering matrix describe the transformation of the polarization of the incident wave to the polarization of the backscattered wave. For monostatic configurations (the ones in which the receive antenna is co-located with the transmit antenna), the scattering matrix becomes symmetric, *i.e.*,  $S_{hv} = S_{vh}$ . The real and imaginary parts of the three complex terms of the scattering matrix provide six variables in total that can be reduced to five independent parameters by normalization with respect to a given phase term. For our tests, we normalize with respect to the phase of the  $hh$  term and the results of the classification, performed using this representation, are indicated as *cl1* in the remaining part of the chapter.

Real systems involve scatterers situated in dynamic environments and subject to space and/or time variations (*i.e.*, non-deterministic scatterers). This causes the electromagnetic waves to be partially polarized and thus prevents the scattering process from being described by a single matrix  $[\mathbf{S}]$ . Hence, averaging processes are needed to cope with the statistical variation of the polarization. The covariance and coherency matrices, the definition of which includes such averaging, take into account these variations and permit their description.

In the monostatic case, the  $3 \times 3$  covariance matrix has the form:

$$[\mathbf{C}_{(3)}] = \begin{bmatrix} \langle |S_{hh}|^2 \rangle & \sqrt{2} \langle S_{hh} S_{hv}^* \rangle & \langle S_{hh} S_{vv}^* \rangle \\ \sqrt{2} \langle S_{hv} S_{hh}^* \rangle & 2 \langle |S_{hv}|^2 \rangle & \sqrt{2} \langle S_{hv} S_{vv}^* \rangle \\ \langle S_{vv} S_{hh}^* \rangle & \sqrt{2} \langle S_{vv} S_{hv}^* \rangle & \langle |S_{vv}|^2 \rangle \end{bmatrix}, \quad (2)$$

with its elements derived from the ones of the scattering matrix. By definition, the covariance matrix is Hermitian positive semidefinite, hence, its symmetric elements are complex conjugates and only nine independent parameters are necessary in order to completely characterize it. For classification, these nine parameters (the three real main diagonal elements and the real and imaginary parts of the three non-redundant off-diagonal elements) have been given as input to the classifier. In the following, the MLP output results based on the covariance matrix elements are denoted as *cl2*.

Alternatively, an incoherent representation is provided by the coherency matrix, which is defined as:

$$[\mathbf{T}_{(3)}] = \frac{1}{2} \begin{bmatrix} \langle |S_{hh} + S_{vv}|^2 \rangle & \langle (S_{hh} + S_{vv})(S_{hh} - S_{vv})^* \rangle & 2\langle S_{hv}^*(S_{hh} + S_{vv}) \rangle \\ \langle (S_{hh} + S_{vv})^*(S_{hh} - S_{vv}) \rangle & \langle |S_{hh} - S_{vv}|^2 \rangle & 2\langle S_{hv}^*(S_{hh} - S_{vv}) \rangle \\ 2\langle S_{hv}(S_{hh} + S_{vv})^* \rangle & 2\langle S_{hv}(S_{hh} - S_{vv})^* \rangle & 4\langle |S_{hv}|^2 \rangle \end{bmatrix}. \quad (3)$$

A direct interpretation of the measured values is possible, *e.g.*, in terms of the scattering model to be adopted (and this depending, in turn, on the surface roughness). The predominance of one  $[\mathbf{S}]$  matrix term over the other is connected to the type of scattering on the illuminated surface (Ulaby et al. 1982; Born & Wolf, 1985; Curlander & McDonough, 1991). More precisely,  $|S_{hh}| > |S_{vv}|$  when the incident beam is scattered according to the Fresnel model, valid for almost flat surfaces, whereas the case of  $|S_{hh}| < |S_{vv}|$  is verified for scattering from rough surfaces described by the Bragg model.

## 2.2 Target decomposition theorems

Target decomposition (TD) theorems permit to identify different scattering mechanisms corresponding to sets of theoretical models (Cloude & Pottier, 1996; Corr & Rodrigues, 2002; Moriyama et al., 2004). This further means that these methods intrinsically perform a classification, since they recognize and weight the contributions of different model targets in a scene.

We have applied here the principal decomposition theorems: the Pauli and the Cameron decomposition, as examples of the coherent methods (that operate on the scattering matrix), and the Freeman decomposition, as an example of the incoherent ones (based on the covariance and coherency matrices).

The classification results corresponding to the Pauli decomposition are referred to as *cl3*. By means of the set of the Pauli matrices, it is possible to write a generic matrix

$$[\mathbf{S}] = \begin{bmatrix} S_{hh} & S_{hv} \\ S_{vh} & S_{vv} \end{bmatrix} = \begin{bmatrix} a+b & c-jd \\ c+jd & a-b \end{bmatrix} \\ = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} + c \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + d \begin{bmatrix} 0 & -j \\ j & 0 \end{bmatrix}' \quad (4)$$

where  $a$ ,  $b$ ,  $c$  and  $d$  are complex numbers. The first decomposition term represents single scattering from a plane surface or a sphere, the second and third term correspond to double-bounce scattering from diplane reflectors with a relative orientation of  $45^\circ$ , and the fourth term - to a scatterer that rotates every incident polarization by  $90^\circ$ . As it causes  $[\mathbf{S}]$  to be non-symmetric, the fourth term disappears in reciprocal backscattering cases.

The Cameron decomposition (Cameron et al., 1996) (*cl4*) is a more generalized example of the model fitting seen with the previous decomposition. A generic matrix  $[S]$  (not only in the monostatic case) can be characterized by its tendency of being more or less symmetric according to the reciprocity rule and it can be split into two terms representing reciprocal and non-reciprocal scattering mechanisms. The reciprocal term represents a target which is more or less symmetric with respect to an axis in the plane orthogonal to the radar line-of-sight and, again, a distinction can be made between the most and the least dominant symmetric target components.<sup>1</sup> Thus, the decomposition follows the scheme (Cameron et al., 1996):

$$\begin{array}{l}
 \rightarrow [S]_{rec} \rightarrow [S]_{sym}^{max} \\
 \rightarrow [S]_{sym}^{min} \\
 [S] \\
 \rightarrow [S]_{non-rec}
 \end{array}$$

The degrees of reciprocity and symmetry are evaluated in terms of projection angles of the scattering vectors onto the corresponding subspace and subsets via proper projection operators. As SAR data are calibrated in order to fulfill reciprocity constraints, the basic distinction among scatterers made by this method is based on their geometrical symmetry. For this reason, only the decomposition of the  $[S]$  matrix into its most dominant and least dominant symmetric terms,  $[S]_{sym}^{max}$  and  $[S]_{sym}^{min}$ , has to be performed.

The principle behind incoherent decomposition theorems consists in modeling the scattering interaction so that the received power may be expressed as sum of contributions due to different basic mechanisms. As seen above, similar modeling based on the  $[S]$  matrix yields normally combinations of terms where typical scatterers, like spheres and diplanes, may be recognized. In (Freeman & Durden, 1998), another method is presented, less bound to pure mathematical models and more to real scatterers. The Freeman decomposition (*cl5*) describes the scattering as due to three physical mechanisms: first-order (surface) scattering,  $s$ , a double-bounce scattering mechanism (corner reflector),  $d$ , and canopy (or volume) scattering from randomly oriented dipoles,  $v$ . According to this model, the measured power  $P$  may be finally expressed as (Freeman & Durden, 1998):

$$P = \langle |S_{hh}|^2 \rangle + \langle |S_{vv}|^2 \rangle + 2\langle |S_{hv}|^2 \rangle = P_s + P_d + P_v, \quad (5)$$

with the three decomposition terms being related via a system of linear equations to the covariance matrix elements.

---

<sup>1</sup> Note the difference in the use of the word "symmetry" when referred to scattering matrices and to targets. According to the Cameron decomposition, scattering matrices which are symmetric due to the reciprocity constraint may describe targets which are geometrically more or less symmetric in the plane orthogonal to the radar line-of-sight (in the case of a helix, a symmetric scattering matrix represents a target which is not geometrically symmetric).

### 2.3 Entropy/ $\alpha$ analysis

A generic coherency matrix may be diagonalized and decomposed by determining its eigenvalues and eigenvectors (Cloude, 1986; Cloude & Pottier, 1996; Cloude & Pottier, 1997). Using this decomposition, the different relevance of each scattering mechanism (within a given resolution cell) is expressed by means of its eigenvalues. Indeed, while the eigenvectors discriminate the presence of different scattering mechanisms, the eigenvalues underline their intensity. A quantity that measures the randomness of these scattering processes is the polarimetric scattering entropy,  $H$ :

$$H = \sum_{i=1}^3 -P_i \log_3 P_i, \quad (6)$$

where:

$$P_i = \frac{\lambda_i}{\sum_{j=1}^3 \lambda_j}. \quad (7)$$

The three  $\lambda_i$  are the calculated eigenvalues, conventionally ordered such that  $0 \leq \lambda_3 \leq \lambda_2 \leq \lambda_1$ , and  $P_i$  represents the appearance probability of each contribute.  $H$  ranges from 0 to 1:  $H = 0$  stands for a deterministic scattering process (the coherency matrix has only one non-zero eigenvalue), while  $H = 1$  indicates a degenerated eigenvalues spectrum, typical of random noise processes (the coherency matrix has three identical eigenvalues).

To estimate the relative importance of the different scattering mechanisms, the polarimetric anisotropy ( $A$ ) has been introduced:

$$A = \frac{\lambda_2 - \lambda_3}{\lambda_2 + \lambda_3}. \quad (8)$$

A medium entropy means that more than one single scattering mechanism contributes to the backscattered signal, but it is not clear how many additional mechanisms are present (one or two). In this case, a high  $A$  states that only the second scattering mechanism is important, whereas a low  $A$  indicates a remarkable contribute also of the third one.

Finally, a further parameter, the  $\alpha$  angle, may be derived from the coherency matrix eigenvectors, which is associated to the type of scattering mechanism and can vary in the range  $[0, \pi/2]$ .  $\alpha = 0$  stands for isotropic surfaces,  $\alpha = \pi/2$  for isotropic diplanes or helices. Low values of  $\alpha$  represent all-anisotropic scattering mechanisms with  $S_{hh}$  different from  $S_{vv}$ . The boundary between anisotropic surfaces and diplanes is represented by the case  $\alpha = \pi/4$ , which describes a horizontal dipole. An average  $\alpha$  angle is normally used in polarimetric SAR data analysis. Henceforth, the classification results of the  $H/\bar{\alpha}/A$  parameters are referred to as *cl6*.

### 3. Classification algorithm

The neural network classifier chosen in this work is the Multi-Layer Perceptron architecture, with one hidden layer, trained by the Back-Propagation learning rule. The MLP is a fully connected feed-forward neural network, composed of nodes arranged in layers. It can be used to perform every non-linear input-output mapping, such as classification functions

(Hertz et al., 1991), or more complex tasks such as the approximation of continuous functions (Funahashi, 1989). For this purpose, it is necessary to submit the MLP to a training phase that searches the optimum set of weights minimizing a cost measure, usually given by the mean square error between estimated and expected outputs. This training phase, performed by the well-known BP learning rule (Hertz et al., 1991), requires a set of input-output examples. In this context, the input examples are obtained by a supervised procedure that identifies on the mono- or multi-band image, and for each class of interest, a corresponding region of points. Consequently, the output examples are given by the class labels  $j, j=1, 2, \dots, n$  ( $n$  being the number of classes), represented by  $n$ -dimensional Boolean vectors. The MLP dimension to be used is related to the current classification problem. In particular, the number of MLP input nodes corresponds to the number of polarimetric features under investigation, whereas the number of output nodes is set equal to the number  $n$  of classes identified on the images. The performance of each trained MLP is estimated on an independent test data set.

The same MLP classification procedure is applied to each of the six sets of polarimetric parameters described in Section 2, and the classification results are referred to as *cl1* - *cl6* (Table 1). The results are good for some sets and exhibit lower performances for other sets, in terms of global accuracy (see Section 7). However, the overall accuracy does not completely describe the behavior of the classifier. Looking more precisely at the performances for each class, it appears that even a globally good set of parameters can lead to classification results for a particular class which are worse than the ones obtained using another set of parameters. The main reason is that classes may be well separated in some polarimetric representations and not in other ones, and there is no single set of parameters that separates correctly all classes. These observations advocate for a fusion of all classification results obtained on the different sets of polarimetric parameters. An important contribution of this paper is to show how this fusion can be performed using the belief function theory.

Classification	Corresponding set of parameters
<i>cl1</i>	scattering matrix
<i>cl2</i>	covariance matrix
<i>cl3</i>	Pauli decomposition
<i>cl4</i>	Cameron decomposition
<i>cl5</i>	Freeman decomposition
<i>cl6</i>	$H/\bar{\alpha}/A$ parameters

Table 1. Set of parameters used to obtain each classification

#### 4. On belief functions

Belief function theory or Dempster-Shafer (DS) evidence theory has been already widely used in satellite image processing (Masclé et al., 1997; Le Hégarat-Masclé et al., 1998; Cleynebreugel et al., 1991; Tupin et al. 1999; Milisavljević & Bloch, 2003). DS theory allows representing both imprecision and uncertainty, using plausibility and belief functions derived from a mass function. The mass of a proposition  $A$  is a part of the initial unitary amount of belief that supports that the solution is exactly in  $A$ . It is defined as a function  $m$  from  $2^\Theta$  into  $[0, 1]$ .  $\Theta$  is the decision space (frame of discernment) and it is a set of  $N$  possible

solutions, *e.g.*, classes  $C_i$ ,  $i = 1, 2, \dots, N$  ( $\Theta = \{C_1, C_2, \dots, C_N\}$ ), while the power set, denoted  $2^\Theta$ , consists of  $2^N$  subsets of  $\Theta$ :

$$2^\Theta = \{\emptyset, C_1, C_2, \dots, C_N, \{C_1 \cup C_2\}, \dots, \Theta\}, \quad (9)$$

meaning that it contains not only single hypotheses (singletons) of  $\Theta$ , but also all possible unions of the singletons, called compound hypotheses or disjunctions. Thus, in this formalism, any combination of possible decisions from the decision space can be quantified rather than considering only the singletons of  $\Theta$  (Shafer, 1976; Smets, 1990). This is one of the main advantages of the DS approach, as it leads to a very flexible and rich modeling, able to fit a large class of situations, occurring in image fusion in particular. Another advantage of this method over the probabilistic ones is in allowing an easy way of representing the state of a total ignorance by means of the so-called vacuous basic belief assignment:  $m(\Theta) = 1$ ,  $m(A) = 0$ , for all  $A \neq \Theta$ . The basic difficulty that some other theory, such as probabilistic, faces in these cases is the inability of distinguishing between lack of belief and disbelief (Shafer, 1976).

In image processing, mass functions may be derived at three different levels: the most abstract level, an intermediate level, and the pixel level. At the pixel level, which is the most interesting here, mass assignment is inspired from statistical pattern recognition. The most widely used approach is as follows: masses on simple hypotheses are computed from probabilities or from the distance to a class center (Appriou, 1993; Dencœux, 1995; Appriou, 1998). Then a global ignorance  $m(\Theta)$  is introduced as a discounting factor, often as a constant on all pixels (Lee & Leahy, 1990). In most cases, no other compound hypothesis is considered, and this drastically under-exploits the power of DS. The mass assignment in (Bloch, 1996) is based on a reasoning approach where knowledge about the information provided by each image is used to choose the focal elements (*i.e.*, subsets with non-zero mass values). A similar reasoning is used in (Milisavljević & Bloch, 2001). However, in the case of large numbers of classes, this process would become too tedious, and unsupervised methods are needed, such as (Masclé et al., 1997) for SAR imaging or (Ménard et al., 1996) for fusion of several classifiers.

If we have evidence issued from  $M$  sources, modeled in terms of previously defined mass functions, these masses are combined applying Dempster's rule of combination (Shafer, 1976; Smets, 1993). This rule has two main forms, normalized (Shafer, 1976) (by imposing  $m(\emptyset) = 0$ ), and unnormalized (Smets, 1993), corresponding to the closed-world and the open-world assumptions, respectively. For  $m_j$  being the mass function associated with source  $j$  ( $j = 1, 2, \dots, M$ ), the unnormalized Dempster's rule of combination is:

$$\forall A \in 2^\Theta, \quad (m_1 \oplus \dots \oplus m_M)(A) = \frac{\sum_{B_1 \cap \dots \cap B_M = A} m_1(B_1) \cdot m_2(B_2) \cdot \dots \cdot m_M(B_M)}{1 - \sum_{B_1 \cap \dots \cap B_M = \emptyset} m_1(B_1) \cdot m_2(B_2) \cdot \dots \cdot m_M(B_M)}, \quad (10)$$

hence preserving the mass assignment to the empty or zero set, that indicates the strength of the possibility that "something else" happens (either the full set is not an exhaustive set of hypotheses or there is some contradiction between the sources, *e.g.*, some of them are not reliable).

Dempster's rule of combination is commutative and associative. Also, it behaves in a conjunctive way, meaning that when more sources are combined, mass is more focalized

(thus imprecision decreases) while conflict behaves in a disjunctive way (increases) (Bloch, 1996).

Sources under combination have to fulfill the condition of being independent in the cognitive sense (Shafer, 1976; Smets, 1993). This is related to the notion of distinctness, meaning “no double counting” of pieces of evidence (Dempster, 1967). In case of non distinct pieces of evidences, other combination rules should be employed, such as the cautious rule proposed in (Denœux, 2008). In our experiments, we assume independence and distinctness.

From a mass function, one can derive a belief function, being the degree of minimum or necessary specific (Smets, 1993) support for A:

$$Bel(A) = \sum_{B \subseteq A, B \neq \emptyset} m(B) \quad (11)$$

and a plausibility function, as the degree of maximum specific support for A:

$$Pl(A) = \sum_{A \cap B \neq \emptyset} m(B) . \quad (12)$$

After combination, the final decision is usually taken in favor of a simple hypothesis using one of several rules (Denœux, 1995; Denœux, 1997): *e.g.*, the maximum of plausibility (generally over simple hypotheses), the maximum of belief, the pignistic decision rule (Smets & Kennes, 1994), *etc.* For some applications, such as humanitarian demining, it may also be necessary to give more importance to some classes (*e.g.*, mines, since they must not be missed) at the decision level. Then maximum of plausibility can be used for the classes that should not be missed, and maximum of belief for the others (Milisavljević et al., 2003; Milisavljević & Bloch, 2005).

Not many papers can be found dealing with fusion of polarimetric data. In (Masclé et al., 1997), the main idea is to perform unsupervised classification on each image of a polarimetric data set separately. The intersections of the obtained classes define the set of discernment and the initial classes can be expressed as disjunctions of some of these intersections. This allows determining automatically both the singletons and the interesting disjunctions. This paper also shows the interest of using a fusion approach and of combining data from several polarimetric images. In (Le Hégarat-Masclé et al., 1998), the idea is to introduce neighborhood information as a mass function, so as to take a spatial regularity constraint into account and to consider it as a source of information.

## 5. Fusion

### 5.1 Fusion strategies

#### Fusion strategy no. 1 - including a global discounting factor (F1):

Some of the classification results in this work differentiate well two classes and some others do not. In addition, the overall reliability of each of the classification results is different. This fact should be taken into account in the fusion process so that the more reliable classification results influence the fusion result more than the less reliable ones. These are cases in which DS theory may be successful so it is our first choice of fusion approaches.

A first, simple method would consist in considering each of the classification results *cl1 - cl6* as one information source. The focal elements would be simply the classes, using the outputs directly as mass functions. As no confidence values are provided but only decision



images, the mass would assume only values 0 or 1. This approach would inevitably result in a high conflict after the combination. Moreover, only the classes detected by all classifiers would be obtained as resulting focal elements, so no good result could be expected. This shows the interest of really using belief function theory or any other theory that takes into account the specificities of the classifiers, disjunctions of classes and ignorance (mass assigned to the full set,  $\Theta$ ).

In our first fusion strategy, we still consider each classifier output as one information source, but the focal elements are the singletons and  $\Theta$ . The definition of  $m(\Theta)$  takes into account both the fact that some classes are not detected (thus it should be equal to 1 at points where 0 is obtained for all detected classes) as well as global errors. We propose to use a discounting factor  $\gamma$  (Xu et al., 1992) equal to the overall accuracy of a classification result, *i.e.*, the sum of the diagonal elements of the confusion matrix, divided by the cardinality of the training areas. This discounting is applied to all masses defined as in the previous, straightforward approach. Then, if the output value of classification result  $cl_k$  ( $k = 1, 2, \dots, 6$ ) is  $C_i$  at a given pixel, the masses for that pixel and that classification output are assigned as follows:

$$m_k(C_i) = \gamma_k, \quad (13)$$

$$m_k(\Theta) = 1 - \gamma_k. \quad (14)$$

Note that this strategy explicitly uses the confidence matrix, which should be computed on the training areas for each classification output. Hence, at each step of the fusion, the focal elements are always singletons and  $\Theta$ . After assigning masses by all classification outputs in the above way, the DS fusion is performed. Decision rule can be maximum of belief, of mass or of pignistic probability (all being equivalent in this case). This approach is very easy to implement and models in a simple way the fact that classification outputs may not give any information on some classes and may be imperfect. Results are explained by the conjunctive behavior of the Dempster's rule of combination.

### **Fusion strategy no. 2 - including class-dependent discounting factors (F2):**

As a second alternative, we propose to use the confusion matrices for defining more specific discounting for each class. This approach is close to the one proposed in (Mercier et al., 2005; Mercier et al., 2008). Each output of the classifier is still one information source, and the focal elements are the singletons and  $\Theta$ . From the confusion matrix computed from classifier output  $cl_k$  ( $k = 1, 2, \dots, 6$ ) and from the training areas, we use the diagonal coefficients  $conf_k(i, i)$  for discounting. Thus, if the output value of  $cl_k$  is  $C_i$  at a given pixel, the masses for that pixel and that classifier output are assigned as follows:

$$m_k(C_i) = conf_k(i, i), \quad (15)$$

$$m_k(\Theta) = 1 - conf_k(i, i). \quad (16)$$

In comparison with the previous method, the richness of the information provided by a classifier output is better exploited as the class-dependent classification accuracy is used instead of a global accuracy of the classifier.

### **Fusion strategy no. 3 - fuzzy fusion (F3):**

In order to compare the previous methods with a fuzzy approach, we have tested a simple method, where for each class we choose the best classification results, and combine them

with a maximum operator (possibly with some weights). Then a decision is made according to a maximum rule. The choice is made based on the confusion matrix for each classification result, by comparing the diagonal elements in all matrices for each class. This approach is interesting because it is very fast. It uses only a part of the information, which could also be a drawback if this part is not chosen appropriately. Some weights have to be tuned, which may need some user interaction in some cases. Although it may sound somewhat *ad hoc*, it is interesting to show what can be obtained using the best parts of all classifier outputs.

In a next step, in order to take more benefit from the information and to avoid the *ad hoc* tuning of the weights, we use for each class all the classifier outputs. If the value of classification result  $cl_k$  at a given pixel is  $C_i$ , it participates in the combination for that class discounted by the corresponding diagonal element of the confusion matrix,  $conf_k(i,i)$ . Such discounted classifier outputs are combined using a maximum operator. The decision is made again applying a maximum rule. Thus, in this approach, all classification results that have the same class as output participate in the combination performed for that class, discounted by their accuracy for that class.

Finally, we develop a third fuzzy fusion strategy, where for each class we use again all the classifier outputs. As in the previous strategy, if  $cl_k$  output at a given pixel is  $C_i$ , this classifier participates in the combination for that class discounted by the corresponding diagonal element of the confusion matrix,  $conf_k(i,i)$ . In addition, even if the output of classification result  $cl_k$  at a given pixel is  $C_j$ ,  $j \neq i$ , it participates in the combination for class  $C_i$  if the confusion between the two classes, expressed by  $conf_k(i,j)$  in the confusion matrix, is high enough (above a threshold). In that case, this element of the confusion matrix is used for discounting the classifier output prior to combination per class using a maximum operator. The decision is again made using a maximum rule. Since the results obtained in this way make the largest use of the information and are the best of the three strategies proposed in this subsection, this fuzzy strategy is used here. Note that this approach takes into account the fact that if the decision is  $C_j$ , the true class is possibly  $C_i$ , to some degree.

The actual list of classification results used for each class in our application is detailed in Subsection 7.3.

## 5.2 Spatial regularization

Spatial regularization is the final step in our fusion approach, applied to the output of each of the proposed strategies. Namely, it is very unlikely that isolated pixels of one class can appear in another class. Hence, several local filters have been tested, such as a majority filter, a median filter, or morphological filters, applied on the decision image. A Markovian regularization approach on local neighborhoods was tested too. All these filters give similar and good results, and improve results of the previous fusion steps (see Section 6). A recently proposed approach (Bloch, 2008) could be used as well, by integrating spatial information directly in the definition of mass functions.

## 6. Data and experimental approach

For the tests presented here, we have used single-look complex data of the area of Oberpfaffenhofen, Germany, acquired by the E-SAR airborne sensor of the German Aerospace Centre (DLR) during a measurement campaign in October 1999 (see Table 2). The data consist of L-band scattering matrices measured in the *hv*-basis. The size of the data set is 1000 x 4050 pixels (in range and azimuth, respectively).

Noise removal and speckle reduction are obtained by means of simple boxcar filtering with fixed window dimensions. The averaging window adopted for all sets of parameters is  $5 \times 11$  pixels (in range and azimuth, respectively). In this way, data processing could be performed following a consistent approach, *i.e.*, choosing the same averaging window size to define the covariance and coherency matrices or to filter data derived from coherent methods.

The imaged area is situated approximately 25 km South-West of the city of Munich and includes several interesting features: the DLR centre, the former Fairchild Dornier airplane factory and the airfield shared by the two firms (see Figure 1). Not far from them, a small lake and the village of Gilching are located. Other important man-made structures are the motorway and the railway line stretching across the image. The vegetation patches consist of coniferous and mixed forests, meadows and crops.

A common classification procedure has been adopted for all the polarimetric parameters (Alberga et al., 2006). At first, a set of seven ground cover classes has been defined: “water”, “houses”, “roads”, “trees”, “grass”, “field 1” and “field 2”. For each of them, separated areas of training and test samples have been identified having a comparable number of pixels (at this scope, aerial photographs and a cartographic map have been used as complementary sources of information). Then, the training pixels from each class have been fed into the classifiers to perform the training stage. As a following step, all the data have been classified using the MLP. Finally, the test samples have been used to measure the classification performance. The fusion of the results has been performed at this stage, according to the three strategies described in Section 5 and providing a re-assignment of the pixels.



Fig. 1. Backscattered intensity image and regions of interest: [1] “water”, [2] “houses”, [3] “roads”, [4] “trees”, [5] “grass”, [6] “field 1”, [7] “field 2”.

The class “roads” has been defined using only the runway of the airport, so it refers to a relatively wide asphalt surface. Regarding the class indicated as “houses”, it is related to areas with family houses surrounded by gardens (often including trees). In other words, it represents an impure class, characterized by the presence of different scattering components: flat surfaces, dihedrals, volumes (surrounding vegetation), and rough surfaces (ground).

<i>Flight date</i>	6/10/1999
<i>RF-band</i>	L
<i>Centre frequency</i>	1.3 GHz
<i>Wavelength</i>	23 cm
<i>Bandwidth</i>	100 MHz
<i>Range resolution</i>	1.5 m
<i>Azimuth resolution</i>	0.89 m

Table 2. Main data and E-SAR system parameters

## 7. Results

In the following, different measures are used to assess the classification accuracy, based on the confusion matrices obtained on the test areas (note that the columns and the rows in the confusion matrices correspond to the ground-truth map and to the classification output map, respectively):

- $\kappa$  coefficient, as a measure of the quality of the classified map compared to a randomly generated map:

$$\kappa = \frac{N \sum_{i=1}^r \text{conf}(i, i) - \sum_{i=1}^r \text{conf}(i, +) \text{conf}(+, i)}{N^2 - \sum_{i=1}^r \text{conf}(i, +) \text{conf}(+, i)}, \quad (17)$$

where  $N$  is the total number of pixels in the confusion matrix,  $r$  is the dimension of the confusion matrix,  $\text{conf}(i, i)$  is the number of pixels on the main diagonal, *i.e.*, in row and column  $i$  of the confusion matrix,  $\text{conf}(i, +)$  is the total number of pixels in row  $i$ , and  $\text{conf}(+, i)$  is the total number of pixels in column  $i$  of the confusion matrix;

- overall accuracy  $\gamma$ , *i.e.*, the percentage of correctly classified pixels:

$$\gamma = \frac{\sum_{i=1}^r \text{conf}(i, i)}{N}, \quad (18)$$

- user's accuracy ( $UA$ ), being the probability that a given pixel will appear on the ground as it is classified, so for class  $j$ , it can be defined as:

$$UA_j = \frac{\text{conf}(j, j)}{\text{conf}(j, +)}, \quad (19)$$

- producer's accuracy ( $PA$ ), *i.e.*, the percentage of a given class that is identified correctly on the map, being for class  $j$  calculated as:

$$PA_j = \frac{\text{conf}(j, j)}{\text{conf}(+, j)}. \quad (20)$$

Figures 2-4 contain images of classification results *cl1* - *cl6*, which are inputs for the fusion module. The accuracy estimates obtained for each of the classification results are given in

Table 5. The use of the covariance matrix leads to the best overall accuracy. This confirms the capability of incoherent observables to describe complex scenarios better than coherent ones. Indeed, the Freeman decomposition has the second best overall performance. The limits of the  $H/\bar{\alpha}/A$  parameters are bound to their definition: neither of the three parameters expresses an intensity or a power measurement but they directly provide a semantic interpretation of the type and statistical behavior of the imaged targets. At the class level, again, the Freeman decomposition provides some of the best  $PA$  and  $UA$  values but then also some coherent representations yield specific class maxima, indicating the capability of the given model to describe the typical target of that class.

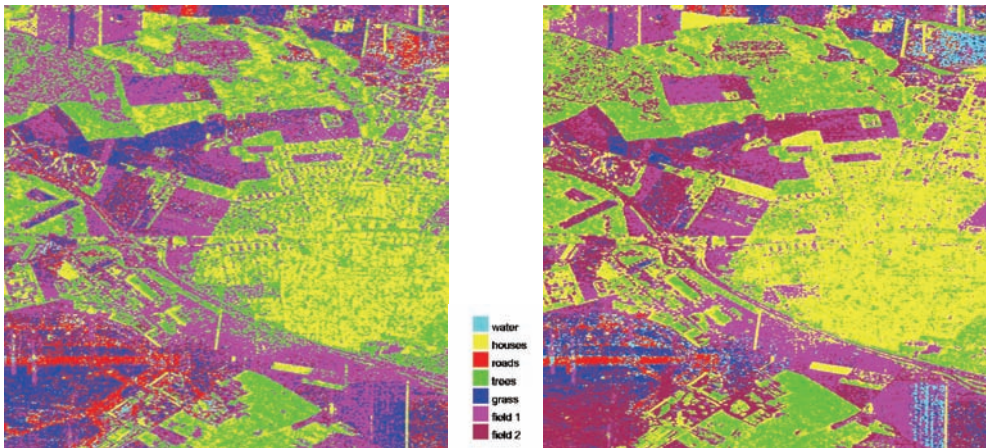


Fig. 2. MLP classification results: left -  $[S]$  matrix ( $cl1$ ), right - covariance matrix ( $cl2$ ).

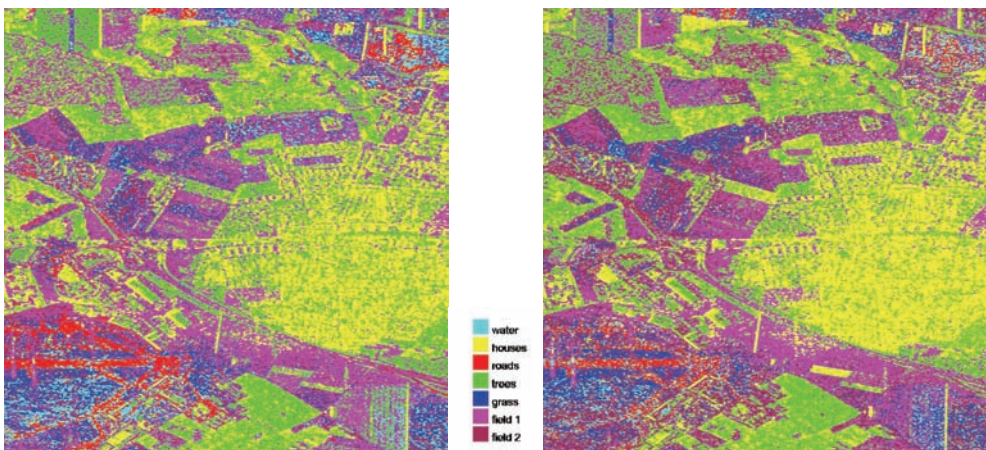


Fig. 3. MLP classification results: left - Pauli decomposition ( $cl3$ ), right - Cameron decomposition ( $cl4$ )

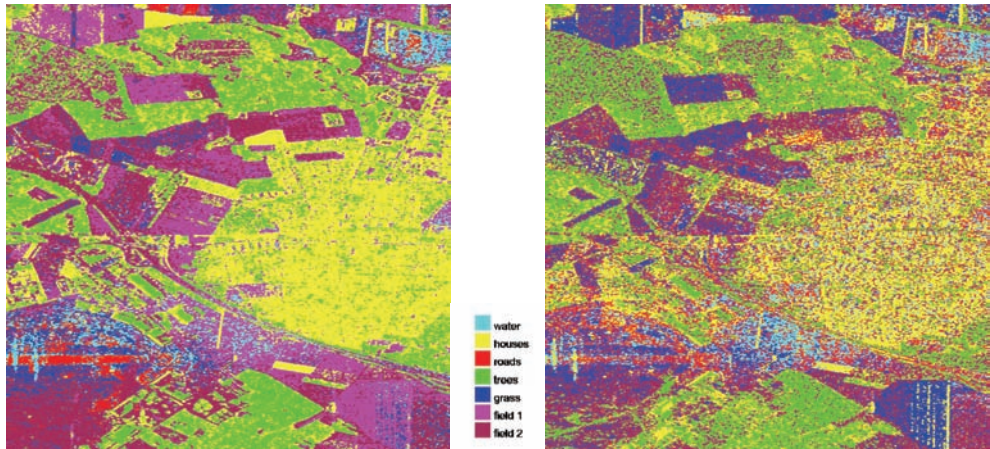


Fig. 4. MLP classification results: left - Freeman decomposition (*cl5*), right -  $H/\bar{\alpha}/A$  (*cl6*).

Besides the fusion results, the fusion module provides confidence and stability images too. At each pixel, the confidence image contains the confidence degree of the decided class, while the stability image represents the difference between the confidence in the decided class and the confidence in the second most possible class. Examples of these two images are shown in Figure 8.

**7.1 Results with F1**

Here, we provide the results obtained using the method F1 (Subsection 5.1). The values of the discounting factors  $\gamma$ , corresponding to the six classification results *cl1* - *cl6* are given in Table 3. For each classification result, this factor is calculated as the normalized sum of the diagonal elements of the confusion matrix obtained on the training areas (*i.e.*, the overall accuracy).

Classification result	<i>cl1</i>	<i>cl2</i>	<i>cl3</i>	<i>cl4</i>	<i>cl5</i>	<i>cl6</i>
$\gamma$	0.66	0.84	0.66	0.71	0.77	0.55

Table 3. Discounting factors for F1

Figure 5 shows the result of fusion using the F1 approach and spatial regularization. Table 5 indicates that F1 outperforms the best classification results in most of the cases. The only exceptions are *UA* for “roads” and *PA* for “grass”. The overall accuracy improvement with respect to the single representations ranges from 13% to 37% but the main advantage of the fusion approach lies in the fact that, at class level, all features are recognized with good *PA* and *UA*. For example, the low *PA* of “water” using the [S] matrix elements or of “field 1” using the  $H/\bar{\alpha}/A$  parameters have been largely compensated by means of the information provided by the other representations.

**7.2 Results with F2**

The discounting factors for the method F2, described in Subsection 5.1, are given in Table 4. They correspond to the diagonal elements of the confusion matrices obtained on the training areas, for each of the six classification results.

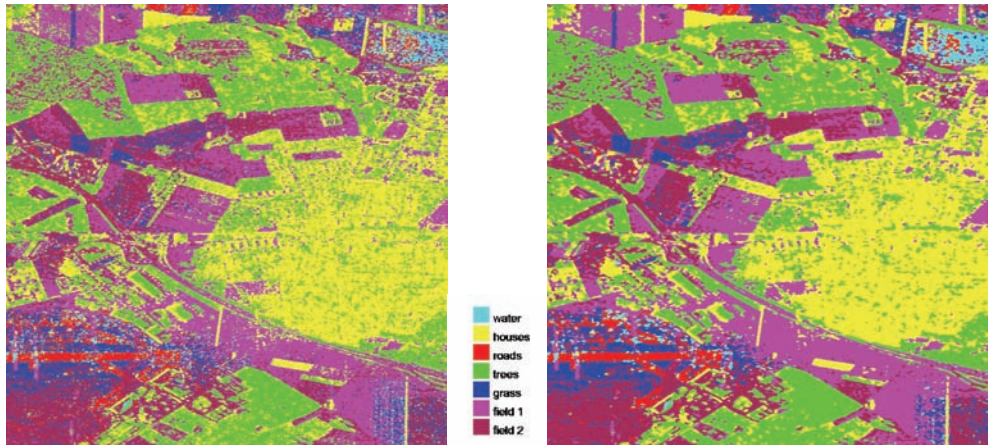


Fig. 5. Results of F1 before (left) and after (right) spatial regularization

Classification result → Class ↓	<i>cl1</i>	<i>cl2</i>	<i>cl3</i>	<i>cl4</i>	<i>cl5</i>	<i>cl6</i>
water	0.23	0.70	0.50	0.48	0.59	0.55
houses	0.70	0.89	0.68	0.83	0.86	0.57
roads	0.84	0.80	0.84	0.74	0.74	0.42
trees	0.89	0.98	0.89	0.88	0.92	0.87
grass	0.70	0.80	0.59	0.71	0.66	0.63
field 1	0.84	0.89	0.78	0.82	0.80	0.17
field 2	0.41	0.81	0.31	0.54	0.82	0.67

Table 4. Discounting factors for F2

According to Table 5, F2 provides better results than the best individual classification in most of the cases. However, there are only half of the classes for which this fusion method outperforms the previous one. Hence, regardless the specificity of the class discounting (varying depending on the class), no clear advantage is obtained by this type of fusion. An overall discounting factor for a given classification result (as for F1) seems to be sufficient. Figure 6 shows the result of the F2 fusion.

### 7.3 Results with F3

As explained in Subsection 5.1, with this approach we take into account the fact that for some classifier outputs, there exist pairs of classes whose confusion is quite strong. Thus, even if  $cl_k$  does not give class  $C_i$  as its output but class  $C_j$ , we include that classification result in the combination for class  $C_i$  if the corresponding coefficient of the confusion matrix obtained on the training data,  $conf_k(i,j)$  has a value higher than some threshold.

The following classifier outputs have been used, all discounted by the corresponding coefficients in the confusion matrices:

- for "water": all six classification results in case of "water", plus "grass" and "roads" for  $cl3$ , "field 1" and "roads" for  $cl6$  and "roads" for  $cl2$ ,  $cl4$  and  $cl5$ ;
- for "houses": all six classification results in case of "houses", plus "trees" for  $cl3$  and  $cl4$ ;

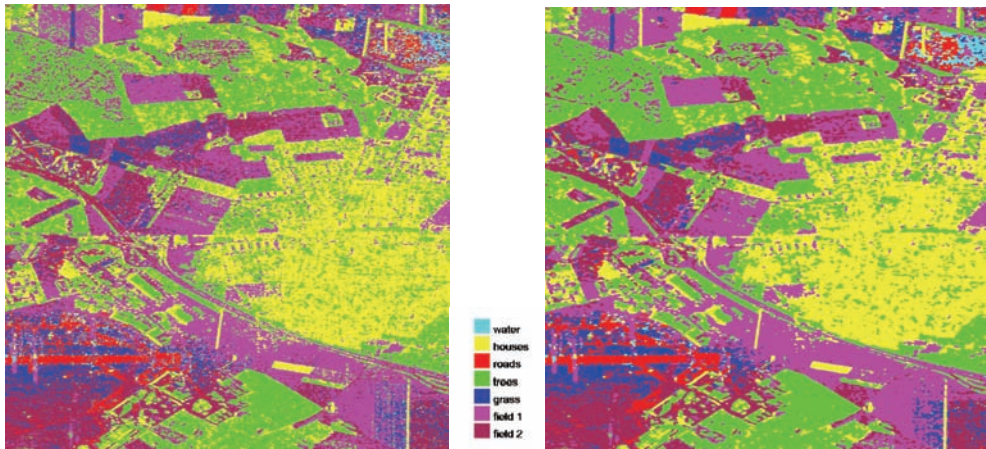


Fig. 6. Results of F2 before (left) and after (right) spatial regularization

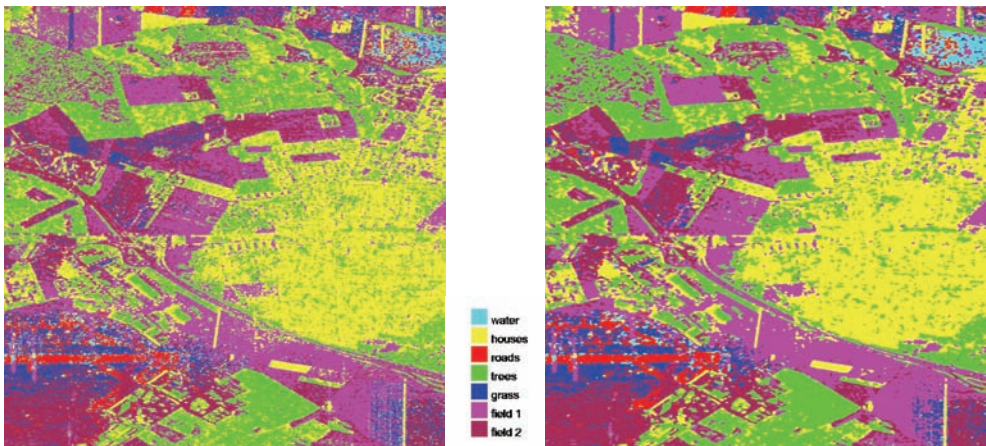


Fig. 7. Results of F3 before (left) and after (right) spatial regularization

- for “roads”: all six classification results in case of “roads”, plus “field 1” and “water” for  $cl1$  and  $cl6$ , and “water” for  $cl3$ ,  $cl4$  and  $cl5$ ;
- for “trees”: all six classification results in case of “trees” and “houses”;
- for “grass”: all six classification results in case of “grass”, plus “water” for  $cl1 - cl5$ , “field 1” for  $cl6$ , and “field 2” for  $cl1$ ,  $cl2$  and  $cl4$ ;
- for “field 1”: all six classification results in case of “field 1”, plus “roads” and “grass” for  $cl6$ , “grass” for  $cl3$  and “field 2” for  $cl1$ ;
- for “field 2”: all six classification outputs in case of “field 2” and “grass”, plus “roads” and “field1” for  $cl6$ .

Table 5 shows that F3 provides results similar to F1. Note that the typical sources of confusion (i.e., the classes characterized by reciprocal  $PA$  and  $UA$ ) are also those whose intensity is roughly comparable (e.g., “water” and “grass” or “houses” and “trees”). Thus, it seems that the radiometric information finally plays the most relevant role in the definition,



and hence classification, of a feature. Fig. 7 presents the result of F3 fusion. Confidence and stability images for this type of fusion are shown in Fig. 8.

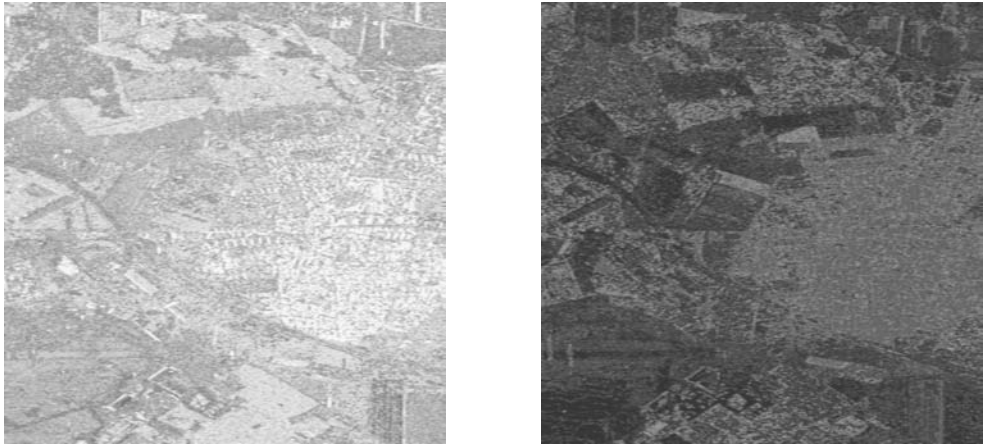


Fig. 8. Confidence (left) and stability (right) image for F3

		<i>cl1</i>	<i>cl2</i>	<i>cl3</i>	<i>cl4</i>	<i>cl5</i>	<i>cl6</i>	F1	F2	F3
$\gamma$		0.68	0.73	0.64	0.65	0.71	0.49	0.86	0.84	0.86
$\kappa$		0.62	0.68	0.58	0.59	0.66	0.41	0.84	0.81	0.84
$P$ $A$	water	0.25	0.60	0.43	0.40	0.61	0.51	0.75	0.53	0.74
	houses	0.72	0.90	0.71	0.82	0.92	0.66	0.94	0.93	0.95
	roads	0.72	0.70	0.72	0.59	0.71	0.32	0.85	0.90	0.85
	trees	0.88	0.92	0.83	0.85	0.92	0.87	0.99	0.99	0.99
	grass	0.80	0.60	0.57	0.68	0.52	0.54	0.68	0.69	0.66
	field 1	0.84	0.62	0.79	0.69	0.54	0.05	0.89	0.92	0.91
	field 2	0.42	0.70	0.25	0.43	0.70	0.60	0.83	0.80	0.83
$U$ $A$	water	0.73	0.84	0.62	0.59	0.79	0.62	0.96	0.99	0.95
	houses	0.86	0.81	0.80	0.83	0.81	0.70	0.96	0.97	0.96
	roads	0.52	0.92	0.67	0.56	0.85	0.35	0.89	0.75	0.89
	trees	0.70	0.85	0.69	0.75	0.89	0.69	0.93	0.90	0.94
	grass	0.54	0.49	0.39	0.49	0.54	0.50	0.69	0.66	0.68
	field 1	0.84	0.78	0.76	0.84	0.77	0.17	0.90	0.86	0.89
	field 2	0.45	0.44	0.37	0.34	0.37	0.25	0.62	0.66	0.63

Table 5. Results on the test data

## 8. Conclusion

In this chapter, three strategies for fusion of land cover classification results of polarimetric SAR data are proposed, two of them based on belief function theory and one based on fuzzy sets theory. As a final step of each of the fusion strategies, spatial regularization is performed. The proposed fusion strategies are applied to the outputs of a neural network classifier, corresponding to six different sets of polarimetric parameters as input. The values

of the different polarimetric parameters are extracted from single-look complex data of the area of Oberpfaffenhofen, Germany, acquired by the E-SAR airborne sensor of the German Aerospace Centre.

The proposed fusion strategies do not need statistical independence of their input information. Each of the strategies uses the confusion matrices obtained on the training data set as means for estimating the accuracy of each of the classification results and for weighing them prior to their fusion.

The test data set, different from the training data set, is used to compare the classification accuracy of the fusion results with the single representation classification results. The results obtained by each of the three fusion strategies show a significant improvement of the classification accuracy of the separate classification results.

The general improvement of the classification accuracy indicates the complementary nature of the information provided by the analyzed polarimetric representations. Although their classification performance is comparable, different characteristics of the scene are enhanced by each set of observables, so their fusion effectively takes advantage of their whole information content.

## 9. Acknowledgements

E-SAR data have been provided by the German Aerospace Centre (DLR). DLR takes no responsibility for the published results.

Part of the SAR data processing has been performed using the free RAT software developed at the Department of Computer Vision and Remote Sensing of the Technical University of Berlin, Germany.

## 10. References

- Touzi, R.; Boerner, W-M.; Lee, J-S. & Lueneburg, E. (2004). A Review of Polarimetry in the Context of Synthetic Aperture Radar: Concepts and Information Extraction. *Canadian Journal of Remote Sensing*, Vol. 30, No. 3, pp. 380-407.
- Cloude, S. R. & Pottier, E. (1997). An Entropy Based Classification Scheme for Land Application of Polarimetric SAR. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 35, No. 1, pp. 68-78.
- Ferro-Famil, L.; Pottier, E. & Lee, J-S. (2001). Unsupervised Classification of Multifrequency and Fully Polarimetric SAR Images Based on H/A/ $\alpha$ -Wishart Classifier. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 39, No. 11, pp. 2332-2342.
- Hoekman, D. H. & Vissers, M. A. M. (2003). A New Polarimetric Classification Approach Evaluated for Agricultural Crops. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 41, No. 12, pp. 2881-2889.
- Kersten, P. R.; Lee, J-S. & Ainsworth T. L. (2005). Unsupervised Classification of Polarimetric Synthetic Aperture Radar Images Using Fuzzy Clustering and EM Clustering. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 43, No. 3, pp. 519-527.
- Lee, J-S.; Grunes, M. R. & Kwok, R. (1994). Classification of Multi-Look Polarimetric SAR imagery Based on Complex Wishart Distribution. *International Journal of Remote Sensing*, Vol. 15, No. 11, pp. 2299-2311.
- Lee, J-S.; Grunes, M. R.; Ainsworth, T. L.; Du, L. J.; Schuler, D. L. & Cloude, S. R. (1999). Unsupervised Classification Using Polarimetric Decomposition and the Complex

- Wishart Classifier. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 37, No. 5, pp. 2249-2258.
- Lee, J-S.; Grunes, M. R.; Pottier, E. & Ferro-Famil, L. (2004). Unsupervised Terrain Classification Preserving Polarimetric Scattering Characteristics. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 42, No. 4, pp. 722-731.
- Chen, C-T.; Chen, K-S. & Lee, J-S. (2003). The Use of Fully Polarimetric Information for the Fuzzy Neural Classification of SAR Images. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 41, No. 9, pp. 2089-2099.
- Alberga, V.; Satalino, G. & Staykova, D. K. (2006). Polarimetric SAR Observables for Land Cover Classification: Analyses and Comparisons, *Proceedings of SPIE - SAR Image Analysis, Modeling, and Techniques VIII*, vol. 6363, Stockholm, Sweden, DOI: 636305.
- Cameron, W. L.; Youssef, N. N. & Leung, L. K. (1996). Simulated Polarimetric Signatures of Primitive Geometrical Shapes. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 34, No. 3, pp. 793-803.
- Cloude, S. R. & Pottier, E. (1996). A Review of Target Decomposition Theorems in Radar Polarimetry. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 34, No. 2, pp. 498-518.
- Freeman, A. & Durden, S. L. (1998). A Three-Component Scattering Model for Polarimetric SAR Data. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 36, No. 3, pp. 963-973.
- Benediktsson, J. A.; Swain, P. H. & Ersoy, O. K. (1990). Neural Network Approaches versus Statistical Methods in Classification of Multisource Remote Sensing Data. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 28, No. 4, pp. 540-551.
- Masclé, S.; Bloch, I. & Vidal-Madjar, D. (1997). Application of Dempster-Shafer Evidence Theory to Unsupervised Classification in Multisource Remote Sensing. *IEEE Trans. Geoscience and Remote Sensing*, Vol. 35, No. 4, pp. 1018-1031.
- Le Hégarat-Masclé, S.; Bloch, I. & Vidal-Madjar, D. (1998). Introduction of Neighborhood Information in Evidence Theory and Application to Data Fusion of Radar and Optical Images with Partial Cloud Cover. *Pattern Recognition*, Vol. 31, No. 11, pp. 1811-1823.
- Shafer, G. (1976). *A Mathematical Theory of Evidence*, Princeton, NJ: Princeton Univ. Press.
- Smets, P. (1990b). The Combination of Evidence in the Transferable Belief Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 5, pp. 447-458.
- SMART consortium (2004). SMART - final report.
- Dubois, D.; Prade, H. & Yager, R. (1999). Merging Fuzzy Information, In: *Handbook of Fuzzy Sets Series, Approximate Reasoning and Information Systems*, Bezdek, J.C.; Dubois, D. & Prade, H. (Eds.), ch. 6, Kluwer, Boston, USA.
- Ulaby, F. T.; Moore, R. K. & Fung, A. K. (1982). *Microwave remote sensing - Volume II: Radar remote sensing and surface scattering and emission theory*, Addison-Wesley Publishing Company, Reading, USA.
- Born, M. & Wolf, E. (1985). *Principles of optics*, 6th edition, Pergamon Press, New York, USA.
- Curlander, J. C. & McDonough, R. N. (1991). *Synthetic Aperture Radar - Systems and signal processing*, John Wiley & Sons, New York, USA.
- Corr, D. G. & Rodrigues, A. (2002). Alternative Basis Matrices for Polarimetric Decomposition, *Proceedings of EUSAR*, pp. 597-600, Cologne, Germany.
- Moriyama, T.; Uratsuka, S.; Umehara, T.; Maeno, H.; Satake, M.; Nadai, A. & Nakamura, K. (2004). Feature Extraction of Urban Area Based on Polarimetric Decomposition, *Proceedings of EUSAR*, pp. 435-438, Ulm, Germany.

- Cloude, S. R. (1986). Group Theory and Polarization Algebra. *Optik*, Vol. 75, No. 1, pp. 26-36.
- Hertz, A.; Krogh, A. & Palmer, R. G. (1991). *Introduction to the Theory of Neural Computation*, Addison Wesley, Redwood City, USA.
- Funahashi, K. (1989). On the Approximate Realizations of Continuous Mappings by Neural Networks. *Neural networks*, Vol. 2, No. 3, pp. 183-192.
- van Cleynenbreugel, I.; Osinga, S.A.; Fierens, F.; Suetens, P. & Oosterlinck, A. (1991). Road Extraction from Multi-temporal Satellite Images by an Evidential Reasoning Approach. *Pattern Recognition Letters*, Vol. 12, pp. 371-380.
- Tupin, F.; Bloch, I. & Maître, H. (1999). A First Step Towards Automatic Interpretation of SAR Images using Evidential Fusion of Several Structure Detectors. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 37, No. 3, pp. 1327-1343.
- Milisavljević, N. & Bloch, I. (2003). Sensor fusion in anti-personnel mine detection using a two-level belief function model. *IEEE Trans. Systems, Man and Cybernetics, Part C*, Vol. 33, No. 2, pp. 269-283.
- Appriou, A. (1993). Formulation et traitement de l'incertain en analyse multi-senseurs, *Proceedings of GRETSI*, pp. 951-954, Juan les Pins, France.
- Lee, R. H. & Leahy, R. (1990). Multi-spectral Classification of MR Images Using Sensor Fusion Approaches, *Proc. of SPIE Medical Imaging IV: Image Processing*, Vol. 1233, pp. 149-157.
- Dencœur, T. (1995). A k-nearest Neighbor Classification Rule based on Dempster-Shafer Theory. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 25, No. 5, pp. 804-813.
- Bloch, I. (1996). Some Aspects of Dempster-Shafer Evidence Theory for Classification of Multi-Modality Medical Images Taking Partial Volume Effect into Account, *Pattern Recognition Letters*, Vol. 17, No. 8, pp. 905-919.
- Milisavljević, N. & Bloch, I. (2001). A Two-Level Approach for Modeling and Fusion of Humanitarian Mine Detection Sensors within the Belief Function Framework, *Proc. of Applied Stochastic Models and Data Analysis*, Vol. 2, pp. 743-748, Compiègne, France.
- Ménard, M.; Zahzah, E.H. & Shahin, A. (1996). Mass Function Assessment: Case of Multiple Hypotheses for the Evidential Approach, *Proceedings of Europto Conf. on Image and Signal Processing for Remote Sensing*, Vol. 2955, pp. 214-218, Taormina, Italy.
- Smets, P. (1993). Belief Functions: The Disjunctive Rule of Combination and the Generalized Bayesian Theorem. *International Journal of Approximate Reasoning*, Vol. 9, pp. 1-35.
- Smets, P. (1995). The Transferable Belief Model for Uncertainty Representation, *Technical Report TR/IRIDIA/95-23*, IRIDIA, Université Libre de Bruxelles, Brussels, Belgium.
- Milisavljević, N.; Bloch, I.; van den Broek, S.P. & Acheroy, M. (2003). Improving mine recognition through processing and Dempster-Shafer fusion of ground-penetrating radar data. *Pattern Recognition*, Vol. 36, No. 5, pp. 1233-1250.
- Milisavljević, N. & Bloch, I. (2005). Improving Mine Recognition through Processing and Dempster-Shafer Fusion of Multisensor Data. In: *Computer-Aided Intelligent Recognition, Techniques and Applications*, Sarfraz, M. (Ed.), ch. 17, pp. 319-343, J. Wiley, New York, ISBN: 0-470-09414-1.
- Xu, L.; Krzyzak, A. & Suen, C. Y. (1992). Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 22, No. 3, pp. 418-435.

- Mercier, D.; Cron, G.; Dencœux, T. & Masson, M.-H. (2005). Fusion of multi-level decision systems using the transferable belief model, *Proceedings of FUSION'2005*, Philadelphia, PA, USA.
- Mercier, D.; Quost, B. & Dencœux, T. (2008). Refined modeling of sensor reliability in the belief function framework using contextual discounting. *Information Fusion*, Vol. 9, No. 2, pp. 246--258.
- Dencœux, T. (1997). Analysis of evidence-theoretic decision rules for pattern classification. *Pattern Recognition*, Vol. 30, No. 7, pp. 1095--1107.
- Dempster, A. P. (1967). Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics*, Vol. 38, pp. 325--339.
- Dencœux, T. (2008). Conjunctive and disjunctive combination of belief functions induced by non distinct bodies of evidence. *Artificial Intelligence*, Vol. 172, No. 2-3, pp. 234-264.
- Appriou, A. (1998). Uncertain data aggregation in classification and tracking processes. *Aggregation and Fusion of imperfect information*, Bouchon-Meunier, B. (Ed.), pp. 231-260, Physica-Verlag, Heidelberg.
- Smets, P. & Kennes, R. (1994). The transferable belief model. *Artificial Intelligence*, Vol. 66, pp. 191-234.
- Bloch, I. (2008). Defining Belief Functions using Mathematical Morphology - Application to Image Fusion under Imprecision. *International Journal of Approximate Reasoning*, Vol. 48, pp. 437-465.

# Multilevel Information Fusion: A Mixed Fuzzy Logic/Geometrical Approach with Applications in Brain Image Processing

Julien Montagner<sup>1</sup> and Vincent Barra<sup>2</sup>

<sup>1</sup> *LaTIM, INSERM U650; Institut TELECOM, TELECOM Bretagne; UEB,*

<sup>2</sup> *LIMOS, UMR CNRS 6158 Université Blaise Pascal  
France*

## 1. Introduction

### 1.1 A definition of information and image fusion

The principle of information fusion is widely used in image-based processes where the data are acquired from several sources, e.g. in the fields of remote sensing (Lennon et al., 2000), satellite (Wald, 2002) or biomedical imaging (Barra & Boire, 2001a). L. Wald defined the fusion operation as “a formal framework in which are expressed the means and tools for the alliance of data originating from different sources”. According to the same author, the aim of a fusion process is to improve the quality of the available information, where the notion of quality depends on the application context. I. Bloch et al. give a more specific definition of fusion: “data fusion consists in combining several pieces of information issued from different sources about the same phenomenon, in order to take a better decision on this phenomenon” (Bloch & Maître, 1997). In accordance with this statement, the fusion of image data is viewed as the joined use of heterogeneous images for decision aid.

The first and most obvious difference between these definitions (and the numerous others proposed in the literature) lies in the aim of the fusion process. That is, the goal to reach is here either to improve the information provided by the different sources, each of them considered as being imperfect (Dubois & Prade, 1994), or to take a decision about the observed scene. The first objective may be viewed as a qualitative improvement (reducing information imperfections), while in the second case the aim is to reduce the doubt about the validity of the decision by increasing the amount of available information. This last case may thus also be viewed as a quantitative improvement of the information.

Whatever the definition for the fusion process, data stemming from one source are generally used to compensate for a lack of information or as a medium for complementary features about the physical object or phenomenon studied. In this chapter, image fusion will refer to a computer-based process aiming at extracting knowledge from an image set, which was obviously not visible in the original images. This new information may consist in either image data (visual result) (Aguilar & New, 2002; Montagner et al., 2005a), numerical indexes (Wang et al., 1998; Montagner et al., 2005b), or even subsets of image regions (image segmentation in e.g. binary sets) (Bloch et al., 2003).

## 1.2 Relevance of information fusion in computer-based diagnosis-aid processes

In the general field of medical imaging, with a special interest here for the study of the human brain, the collection of various data coming from anatomical and functional imagery is becoming very common for the study of a given pathology. The treatment of these data is performed by a physicist, who analyses and aggregates them according to his knowledge. The aim is to provide a better medical decision, to propose a prognosis, or to assist physicians in a surgical intervention. This approach may clearly be modeled and automated by means of an information fusion process, with the interest of providing accurate numerical information to the physicist.

The sensitive nature of the brain makes imaging to be a major investigation tool (rather than a surgical approach), and one can find many medical examples and interests for such an automatic process. Morphological aspects of the human brain are mainly studied using magnetic resonance imaging (MRI). An MR image is a 3D data volume that gives information on structural composition of the organ (distribution of tissues, fine spatial resolution). It can be obtained using a wide range of parameters, so that the resulting images provide sufficient contrasts between the different structures to be located (tissue interfaces, tumors, etc.). The brain may also be studied from a functional point of view, using two major functional modalities: PET and SPECT (positron/single photon emission tomography). The images are obtained by injecting or inhaling a radioactive tracer that preferentially characterized a physiologic (e.g. glucose metabolism) or a biologic (e.g. blood flow) process, and then measuring the resulting particle emission. Both PET and SPECT provide 3D datasets reconstructed from these planar projections (Hudson & Larkin, 1994). Such images have a poor spatial resolution and are not informative on what they are not supposed to represent. Figure 1 shows examples of anatomical and functional images of a same brain.

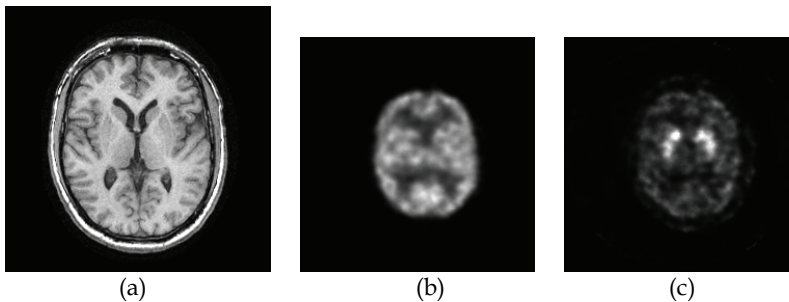


Fig. 1. Three views of a same brain: anatomical (a. MR image) and functional (b. and c. SPECT images respectively showing the blood perfusion and dopamine receptors density)

The main interest of a fusion between information stemming from both MR and such functional images is to supply anatomical information for the accurate detection of pathologic areas characterized in functional imaging by physiological abnormalities (Barra & Boire, 2000b, Barra & Boire, 2001a). Clinical implications are various and numerous, from the detection of functional abnormalities in the study of dementia (Julin & al., 1997) to the precise location of activity sites in neurotransmission SPECT imaging, or the accurate quantification of monoamine transmitters density. Several examples of the fusion scheme we propose will be detailed at the end of this chapter. From a more clinical standpoint, these applications of multisource medical brain imaging can be explained as follows:

- brain tissue segmentation can be carried out as an accurate mean of quantifying the volume of brain matters in diseases such as Alzheimer's dementia, epilepsy, or hydrocephalus, for purpose of diagnosis, treatment, and general understanding;
- accurate segmentation of subcortical brain structures is a fundamental issue in several applications like the assessment of structural brain abnormalities, the study of abnormal entities (e.g. carcinoma), the mapping of functional activation onto human anatomy, the study of brain disorders (e.g. schizophrenia) or computer-assisted neurosurgery;
- effects of Parkinson's disease and Parkinsonian syndromes on striatal structures are commonly characterized by mean of functional imaging (Catafau, 2001). However, an early detection of these effects remains inaccessible to the single visual examination of SPECT (or even PET) images. A quantification process can provide objective numerical indexes in relation with the pathology severity, but it highly depends on the difficult location of regions of interest (ROIs). A fusion with a morphological image of the same brain may represent a suitable solution to this critical segmentation problem;
- new diagnosis elements may also be obtained by the synthesis of an image holding both functional and anatomical information. An advanced fusion strategy (not limited to a color channel combination) allows the physicist to select image features to be displayed, to avoid spatial covering or frequency mixing that would hinder a good perception of the diagnostic information (e.g. hypoperfused gray matter for patients suffering from probable Alzheimer's disease (Colin & Boire, 1999)).

### 1.3 Managing characteristics and structure of heterogeneous image information

When considering the fusion process, input images intensities are often not directly compatible, owing to their numerical nature (numerical type, value range, etc.) and, above all, their physical meaning. Aggregating the information held by two corresponding pixels thus implies to model this information in a common formalism. Handling heterogeneous data also occurs when considering information stemming from different sensors, but of different types (e.g. signal and image sensor, expert knowledge and image sensor). Here again, a common theoretical formalism is needed to embed these data, and the choice of this framework is guided by the nature of available information.

As stated above, medical information acquired from automated sensors, and especially medical images, holds an imperfect information in a sense commonly admitted in the field of information fusion: data are often subject to many uncertainties (e.g. which tissue class(es) associated with a given image intensity range?), possibly due to both inaccuracies (e.g. quantization of image intensities, single value standing for large spatial regions) and/or some inherent ambiguities (e.g. several possible reasons/classes).

SPECT images are a relevant example of these uncertainties. In addition to the mandatory corrections of scattering and other quality-loss phenomena during the tomographic reconstruction (i.e. attenuation, depth-dependent resolution, etc.) (Soret et al., 2003), computer-based diagnosis processes have to face the problem of low spatial resolution intrinsically linked with SPECT images (PET images having a higher resolution, but being more expensive). Consequently, the 3D reconstruction process builds quite large-size voxels, and generates partial volume effects (PVE). Even after PVE correction (Boussion et al., 2006), the image information is therefore the object of some imprecision that makes it impossible to accurately define the boundaries of anatomical structures of interest. Furthermore, the functional information held by tomoscintigraphic images is intrinsically not well fitted to



such a task, because of its possible double meaning: if a low image intensity theoretically follows a low fixation rate of the tracer, it nevertheless remains difficult to determine from the image alone whether this level is associated with an out-of-structure site, or to an abnormal functional activity within structure boundaries (Fig. 2).

The three most common approaches used to represent and manage heterogeneous information in that context are the well known probability theory, the belief functions theory (Shafer, 1976), and the possibilistic logic based on the fuzzy sets formalism (Zadeh, 1978 ; Dubois & Prade, 2004).

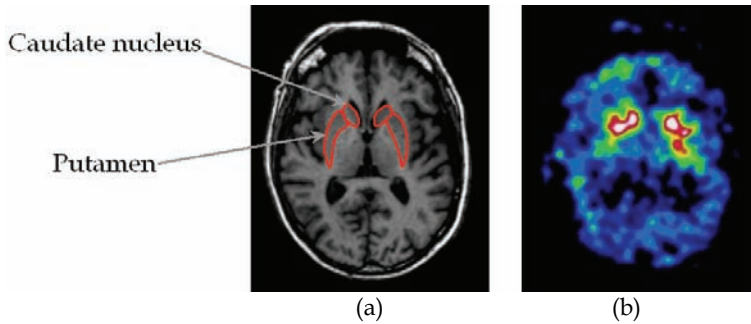


Fig. 2. Co-registered MR (a. showing striatal structures) and neurotransmission SPECT images (b); Outlines of caudate nuclei are roughly defined on (b), whereas boundaries of putamens disappear with the loss of dopamine receptors

Digital images stemming from acquisition processes cited in 1.2 are composed of discrete sets of numerical values (2D/3D arrays of image intensities), standing for given features of the physical phenomenon measured. A pixel (or voxel in 3D) from one of these images is viewed here as the region of the real space in which the associated value has been quantified. Both the numerical aspects and the spatial distribution of image intensities have therefore to be managed in the fusion process, each of these features being of equal relevance in the definition of image information. If the questions of information modeling and aggregation of numerical information using fuzzy sets or other uncertainty models have been widely studied, the problem of spatial matching of data sets to fuse is often considered of secondary importance.

In the case where all images have the same size and spatial resolution, a fusion process can sometimes be directly performed by aggregating information stemming from image numerical values (Bloch, 1996) associated with a unique pixel (with the same index in image arrays). This is only possible if acquisitions have been made from in the same geometrical referential, and if the object of interest is represented with the same size in each image. Since this configuration rarely occurs (because of practical acquisition constraints such as sensor size or spatial and temporal resolutions), most numerical values in a given image don't have direct spatial correspondence with intensities from other data sets (Fig. 3).

In order to manage both geometrical relations (only affine relations are addressed here because of the nature of organs we are interested in) and the difference in spatial resolution between images, most methods process the image information in a multiscale context. Such processes are either based on frequency analysis of data, e.g. managing all image information on a common wavelet base (Pajares & De La Cruz, 2004), or on a "resolution hierarchy" obtained by iterative degradation of original images (Matsopoulos et al., 1994).

The problem is often reduced to the trivial fusion case presented above by a simple registration of image data in a common geometrical base, including an interpolation of numerical data. But the choice of an interpolation method relies on strong assumptions (e.g. linear variation of the measured phenomenon in the real space) that on the one hand simplify the fusion, but also introduce some unwanted imprecision in the data that may hinder an accurate quantification of brain activity in MRI/SPECT fusion. In this case, authors only commonly align the MR image on the SPECT image in order to preserve data to quantify (Soret et al., 2003 ; Rousset et al., 1998). But this method dramatically decreases the anatomical precision.

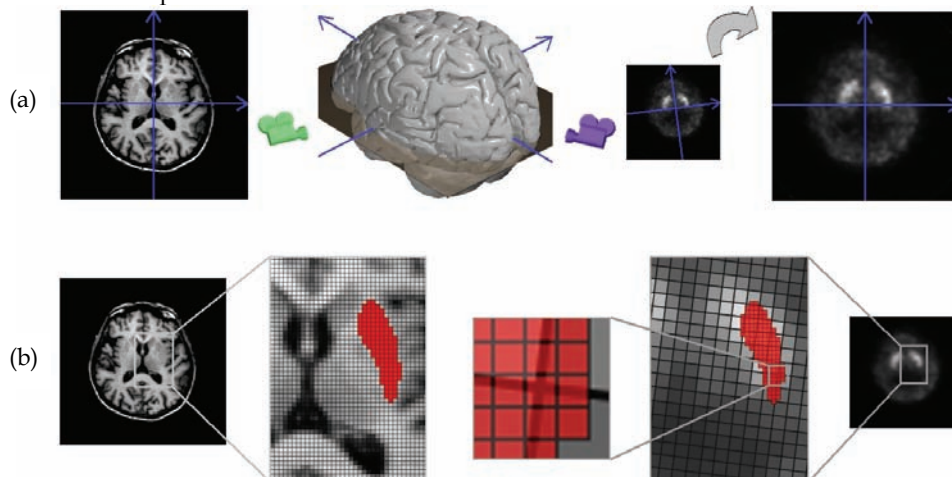


Fig. 3. a. A slice from MR (left) and SPECT (right - original and registered) images of the same brain; b. Consequences of co-registration on voxels from original images through the example of an anatomical structure (putamens)

#### 1.4 A fusion framework for medical images

This chapter describes an information fusion scheme devoted to 2D/3D medical images. Both the different stages of this process and the global architecture were designed to answer practical problems and give assistance for diagnosis using medical image processing and information fusion. This work is typically related to the fusion of different kind of medical images (in which numerical intensities express different physical phenomena), with different spatial orientation and resolution. In the following, the method is presented through examples of brain images processing and illustrated in the case of two input images, although it can be applied to both a more general fusion problem and to other organs. Details of the method are given under the assumption that input images are 3D datasets (collections of voxels), but the extension to the 2D images case is trivial. Most of the illustrations are given in 2D for visual convenience.

The fusion process we propose here is guided by the intrinsic nature of images, viewed as digital information embedded with a given geometrical structure. This multilevel fusion scheme is designed to manage the possibly different spatial distributions of image data apart from other heterogeneity sources, in order to preserve the accuracy of initial data through the whole process. The fuzzy formalism is used as a theoretical framework to

represent image data, and possibly other kind of information (e.g. expert knowledge). The management of image geometrical features is based on principles and algorithms stemming from the field of discrete geometry to preserve the accuracy of original data to fuse (avoiding early interpolation).

The first part of this chapter is centered on the presentation of methodological aspects of this work. Section 2 introduces the general structure of the generic fusion scheme. Sections 3 and 4 respectively focus on the theoretical background for the fuzzy modeling/fusion of image numerical data, and geometric principles and algorithms used for the management of the spatial structure of input images.

Depending on both the nature of the information to fuse and the objective of the fusion, the tools defined by this fusion scheme may be used in several scenarios. For example, fusing two images with different spatial resolution will highly take benefit from the management of image geometrical features, while the fusion of an image with symbolic information (i.e. with no particular spatial structure) will only require a single-level fusion based on fuzzy modeling. Moreover, this fusion scheme is versatile enough to allow multistage fusion too: during the modeling stage, the information representation may also consists in a first complete fusion process.

The second part of this chapter presents three practical examples of brain image fusion relying on the proposed scheme in the case of three different scenarios. Section 5.1 illustrates an example of a 1 level/1 stage fusion scenario aiming at the refinement of brain tissue segmentation from multispectral MR images. The second fusion level (management of image spatial features) is first illustrated in section 5.2 (1 stage only), with the synthesis of an unique image from multimodal information sources. Finally, section 5.3 illustrates the 2 levels/2 stages fusion process, by means of a functional quantification of the brain activity from SPECT images, using an MR image to locate anatomical structures of interest. The segmentation of such anatomical structures is driven by expert information in a first MRI/symbolic information fusion.

## **2. A theoretical framework for the fusion process**

### **2.1 General fusion scheme**

This fusion architecture is an improvement of an existing fusion process, that was designed making use of fuzzy logic (Barra & Boire, 2001a). Input images were supposed to be initially aligned, and this first registration stage was followed by three fusion steps:

1. Information modeling step: datasets were first represented in a common theoretical formalism, in order to compensate for the heterogeneous nature of the information provided by the images. The choice of using fuzzy logic is explained in section 3.1, and details on the modeling itself are given in sections 3.2.
2. Aggregation step: information models were injected into a fitted fusion operator, designed to either produce a new information expressed under the same formalism, or to emphasize information buried into original images (section 3.3).
3. Decision step: the information produced by the previous step was taken back to the relevant numerical domain or to the decision context in which the fusion process is involved, possibly resulting in a final formalism conversion. Moreover, this step came possibly out to a crisp information model ( $\alpha$ -cut, thresholding, etc.), once the amount of information aggregated during the fusion process was sufficient to take a decision.

As stated above, the major problem for e.g. activity quantification purpose using morphological and functional images is the initial registration stage that introduces some additional uncertainties about the information to measure. The generic fusion scheme proposed in this chapter is designed to be possibly used both for classical fusion tasks, i.e. to answer fusion problems that were previously coped thanks to the fusion process presented above, and to preserve as long as possible the accuracy of initial data for a use in more complex fusion problems based for example on multiresolution data.

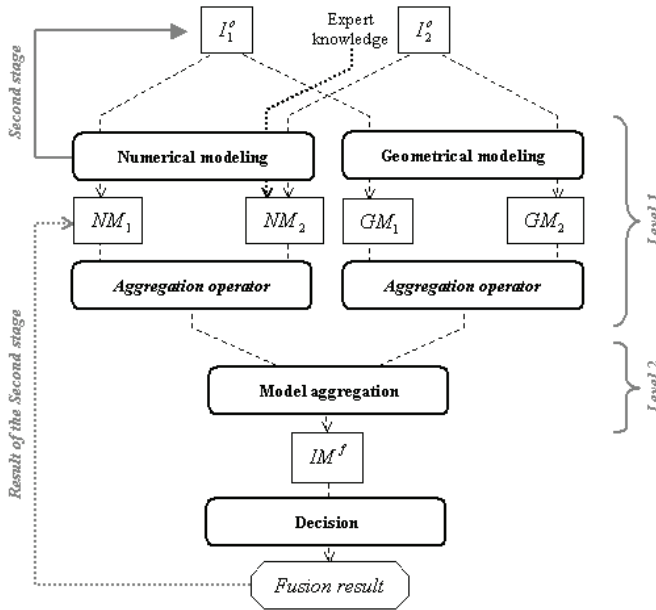


Fig. 4. A synoptic diagram of the fusion process designed in this chapter. This process is illustrated in the case of two input images  $I_i^o, i \in \{1,2\}$ , each being represented by both a numerical model  $NM_i$  and a geometrical model  $GM_i$ ; The final aggregation stage provides a fused information model  $IM^f$  that leads to a decision in the sense given above (point 3)

For this last purpose, all information sets cannot always be modeled using the same formalism, or can even not get modified at the modeling stage (e.g. binary masks of ROIs vs. original SPECT data). This methodological aspect is the first particularity of the fusion process we now propose: the modeling step is extended to the use of compatible theoretical frameworks, which have together a physical meaning, instead of strictly equivalent formalisms. The problem to be solved here is more particularly to give both information stemming from the modeling stage a spatial coherence.

To do so while preserving the accuracy of initial information, we choose to delay the management of this spatial correspondence as far as possible in the fusion process. The solution we propose consists in managing the image spatial structure apart from image numerical intensities. For each input image, these two sets of characteristics are modeled separately, leading to running (partially) two independent fusion processes in parallel.

Fusion operators are designed to aggregate information models in each domain. These operators are finally combined to provide the final aggregation function (Fig. 4), transforming information represented by intensities and structural information models together into the final fusion result.

## 2.2 Definitions and relations with classical information fusion processes

The modeling, aggregation and decision stages presented above are commonly accepted as a structure for fusion-based processes (Wald, 2002). The estimation of information models, considered as an additional task by some authors, is for us fully included in the modeling stage (as shown in 3.2). Each branch of the improved version of the process (Fig. 4) follows the same classical scheme, until aggregation operators are designed to achieve a global fusion between corresponding voxels from input images, represented in the corresponding formalism (i.e. voxel to voxel/geometrical matching of image structures at the voxel level, as described in section 4).

Dasarathy proposed three levels of fusion (Dasarathy, 1997), corresponding to three abstraction levels for representing information: data fusion (values directly provided by the sensors), feature fusion (information derived from previous data) and decision fusion (information expressing hypothesis to be confirmed). The process depicted on Figure 4 is composed of two fusion levels, the first one being performed separately on the two image features sets. This first level clearly occurs at the feature level, since input images are first modeled in each domain, and no decision is moreover taken until both aggregation models are combined. This last step can be associated to an additional fusion level considered by some authors: model fusion, aiming at combining pieces of information that represent a method or process.

A distinction is usually made between two kinds of information as an input of the fusion process: numerical information vs. symbolic information. The architecture of the fusion process described in this chapter allows the description of two other categories of information extracted from input images: structural information of digital images, managed through the right path of the schematic description of the process (Fig. 4), and numerical models of image semantic information (left path). This last category is obviously not independent from a classical description of input information, since it may refer to numerical representation (fuzzy logic) of both image numerical intensities and symbolic information transcribed in the same geometrical referential as image data. Because of the theoretical frames chosen to represent structural information and the previous group of information type, we may also refer to these domains as respectively the geometrical (or spatial) model and the numerical model.

## 2.3 The different scenarios of fusion

Depending on both the spatial referential in which are expressed the information to fuse and the aim of the fusion, applications based on the scheme described here may use only one or both processing paths illustrated on Figure 4. Considering the left branch alone (no geometrical information to manage), the model aggregation level is then avoided, and the combination operator at level 1 may thus achieve a single-level feature-based aggregation, directly providing  $IM^f$  (Fig. 7.a and 11.b).

The multistage aspect of the fusion architecture refers to the ability to achieve building the image numerical model as a complete iteration of the fusion process (Fig. 11). Image

intensities are injected as inputs for the sub-fusion process. Consequently, this fusion stage is carried out based on numerical models only, but may also involve external sources such as expert knowledge (symbolic information) (Fig. 11.b). The result of the sub-fusion stage is then used as a model of image numerical intensities in the first stage process.

### 3. Numerical information modeling and related aggregation operators

#### 3.1 Possibilistic logic as a theoretical framework

Data we will have to manage are quite imprecise and uncertain, due for example to partial volume effects or noise. We, thus chose to model these data with a theory managing uncertainty and imprecision, and we particularly focused on possibilistic logic. Possibilistic logic was introduced by Zadeh in (Zadeh, 1978) in order to simultaneously represent and manage imprecise and uncertain knowledge. In fuzzy set theory, a fuzzy measure is a representation of the uncertainty, giving for each subset  $Y$  of the universe of discourse  $X$  a coefficient in  $[0,1]$  assessing the degree of certitude for the realization of the event  $Y$ . In possibilistic logic, this fuzzy measure is modeled as a measure of possibility  $\Pi$  satisfying  $\Pi(\emptyset)=0, \Pi(X)=1$  and  $\Pi \cup_i Y_i = \sup_i \Pi(Y_i)$ . An event  $Y$  is fully possible if  $\Pi(Y)=1$ , and impossible if  $\Pi(Y)=0$ . Zadeh showed that  $\Pi$  could completely be defined from the assessment of the certitude on each singleton of  $X$ . Such a definition relies on the definition of a distribution of possibility  $\pi$ , satisfying  $\sum_{x \in X} \pi(x)=1$ . Fuzzy and crisp sets can then be represented by distributions of possibility, from the definition of their characteristic function.

We choose the possibilistic logic as the common theoretical frame for the representation of the available data. More precisely, we model all the information using distributions of possibility, and equivalently we represent this information using fuzzy sets.

#### 3.2 Information modeling: from image data to semantic knowledge

We consider in the following two types of information to be represented by numerical models: the information extracted from images, we call the numerical information, which mainly consists in tissue characterization (morphological images) or activity distribution (functional images), and the symbolic or semantic information modeling the linguistic data that may be provided by an expert.

- *Numerical information*: numerical information is directly extracted from images, and is modeled as distributions of possibility either representing brain tissues, (cerebrospinal fluid (CSF), white matter (WM) and gray matter (GM) distributions in morphological images), or distribution of the functional activity. These distributions are computed using a possibilistic clustering algorithm (Krishnapuram & Keller, 1993) on particular feature vectors representing voxels (Barra & Boire, 2000b). If each voxel  $1 \leq j \leq N$  is described by a  $p$ -dimensional feature vector  $x_j$ , the possibilistic clustering algorithm is an iterative algorithm that searches for  $C$  compact clusters gathering in  $X = \{x_j\}$  the  $x_j$ 's by computing both a fuzzy partition matrix  $U = (u_{i,j})$ ,  $1 \leq i \leq C$ ;  $1 \leq j \leq N$ ,  $u_{ij} = \pi_i(j)$  being the membership degree of  $x_j$  to class  $i$ , and unknown cluster centers  $B = (b_i)$ ,  $1 \leq i \leq C$ . The algorithm uses iterative optimizations to find the minimum of a constrained objective function

$$J(B, U, X) = \sum \sum u_{ij}^m d^2(x_j, b_i) + \sum \eta_i \sum (1 - u_{ij})^m \quad (1)$$

subject to  $u_{ij} \in [0, 1]$ ,  $\sum u_{ij} < N$ ,  $\max_i u_{ij} > 0$  and  $\eta_i$  being the intra-class mean fuzzy distance as proposed in (Krishnapuram & Keller, 1996). Parameter  $m$  controls the degree of fuzziness of  $U$ , and is chosen equal to 2 in the following.  $d$  is the Euclidean distance in  $R^p$ .

- *Semantic information*: symbolic information was given by clinicians and experts and consisted in topological and morphological information. These fuzzy propositions might typically be modeled in the possibilistic logic frame. We already propose a theoretical framework to model approximate distance and direction information as fuzzy sets and we refer the reader to (Barra & Boire, 2001b) for a detailed description of the modeling process.

### 3.3 Aggregating the information: definition of fusion operators

Fuzzy sets and the possibilistic logic both offer a wide range of combination operators and a flexible way to choose them. (Bloch, 1996) proposed a classification of these operators with respect to their behavior (in terms of conjunctive, disjunctive or compromise), the possible control of this behavior, their properties and their decisiveness, which proved to be useful for several applications in image processing.

Generally speaking, a fusion operator aggregates  $N_i$  distributions of possibility  $\pi^i$  into a fused distribution  $\pi$ , using a fusion operator  $F : \pi = F(\pi^1, \dots, \pi^N)$ . The definition of  $F$ , its mathematical properties and its behaviour with respect to the agreement and the conflict between the  $\pi^i$  are driven by the application. Some examples of such fusion operators are given in the application section.

## 4. Spatial information modeling and fusion using discrete geometry

The core of the proposed method is a geometrical model representing the spatial structure of input images at the voxel level, and allowing the fusion of the corresponding features considering geometrical relation between images. This model has been chosen considering the fusion process backward, from model aggregation to the modeling stage. The central point was to give the spatial information a final form that was compatible with the final information stemming from the fusion of numerical models, making it possible to merge fusion operators which produced them.

### 4.1 The redistribution principle: getting coefficients from spatial relations

In the case where a two-levels fusion is performed, the final aggregation stage we propose to carry out is called "redistribution" (i.e. redistribution of the numerical information held by image voxels). The information provided by the geometrical aggregation operator aims at representing spatial relations between images by simple sets of numerical values (spatial coefficients). These values, thus consistent with image numerical models, are thereafter used during the final aggregation stage, as weighting factors in the assignment of data from one image or information model to another one (usually from a low-resolution one to a high-

resolution one in a multiresolution problem), in order to compensate for the difference of spatial referential (orientation, resolution, etc.).

If we again consider the two images case, let  $\rho$  be a spatial coefficient locally modeling spatial relations between both images. Since the elementary part of a 3D digital image is the voxel,  $\rho = \rho(v_1, v_2)$  models the basic spatial relation between a voxel  $v_1$  from image  $I_1$  and a voxel  $v_2$  from image  $I_2$ . Note that in the following,  $v$  and  $V$  will stand for  $v_1$  and  $v_2$  for typographic convenience. This notation is moreover justified since images  $I_1$  and  $I_2$ , in the case where they have to be geometrically registered, often have different spatial resolution that implies voxels with different sizes. Voxel  $v$  thus denotes the spatial element with the highest resolution/the smallest voxel size, and  $V$  the element with the largest size.

Spatial coefficients have to express spatial relations between both images, in order to combine information really stemming from the same spatial location. Image numerical intensities are obtained by integrating the measured phenomenon in regions of the real space corresponding to image voxels. To study the spatial distribution of image intensities and spatial relations linking an image to another one, the model thus simply consists in representing the influence region of each numerical value during the measurement, i.e. modeling image voxels themselves. Modeling the whole image thus boils down to define a tiling of the image space by these voxel models.

For the sake of simplicity, only the case of cubic voxels is addressed here (an hexagonal model is for instance a better representation of a spherical influence area, but leads to much more time-consuming algorithms). Discrete geometrical tools used in the following have been chosen under this assumption. The digital nature of image information has also guided the choice of the processing operator used to obtain spatial coefficients from the geometrical models. The question of studying spatial relations between two voxel lattices is indeed very close to the classical discrete coordinates changing problem (Reveilles, 2001). Discrete and computational geometry provide efficient tools to answer such a problem.

The extension of the tiling created by image voxels to the whole space may be considered as a discrete coordinate system. Therefore, the problem of spatially matching a given image with another one is intrinsically linked with the question of accurately expressing, in a target basis  $B_2$ , discontinuous information from a basis  $B_1$ ,  $B_1 \neq B_2$ . A solution was proposed in (Reveilles, 2001) with a suitable formalism for 3D image fusion. When images have close or equal spatial resolutions, the ratio between edge length of  $v$  and  $V$  is not sufficient to disregard the committed error when rounding to integers the results of classical basis change formulas (Fig. 5). In this case, one shall determine the volume of the geometrical intersection between unit elements of the grids and use this value in an interpolation step. Hence, this last processing stage finally appears to be compulsory, but guided by all the available image spatial information.

#### 4.2 Geometrical modeling and fusion operators based on computational geometry

Both the difference in spatial resolution and the spatial misalignment of corresponding structures in images have to be managed during the modeling. When considering two images  $I_1$  and  $I_2$ , these spatial features make information from  $I_1$  and  $I_2$  to be expressed in two different geometrical spaces, which have to be linked through the geometrical model.

The geometrical transform resulting from the usual registration step is not applied to the data (and in particular data are not interpolated), but we use the corresponding mathematical function to generate the geometrical model. Both images are modeled as



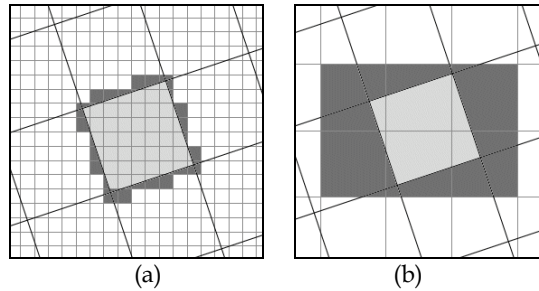


Fig. 5. Illustration of the discrete coordinates changing problem in 2D (a. different grid sizes, b. grids with quite similar pixels). Basis change formulas associate part of pixels in light gray to the right pixel in the other grid, while dark gray surfaces show possible errors

tilings of the geometrical space by cubic voxels, each grid being positioned with respect to each other using this transform. One of the input images  $I_1$  being used as a geometrical reference, let us assume that its voxels are represented by elementary unit volumes of the canonical basis. The underlying idea is to build a set of vectors from the transform, which will generate the cubes that represents voxels in general position (voxels from  $I_2$ ).

In the case of brain imaging, a rigid transform is often sufficient to match images of cerebral structures from both data sets, since the brain is considered as a non-deformable solid. The rigid transform is initially composed of rotations and translations (6 parameters in 3D). A difference in spatial resolution implies a third part in the transform, based on the application of a scaling factor. In this case, let  $I_1$  be the image with the highest spatial resolution, and suppose that the registration operation aims at aligning  $I_2$  on  $I_1$ .

Let  $T$  be the registration operator. The generating vectors of the base cube in general position are the images of canonical unit vectors by  $T$ , and this cube has its origin on a point  $p \neq 0$  because of translation components in  $T$  coming from the image registration ( $p = T \cdot (0\ 0\ 0\ 1)^t$  in homogeneous coordinates). Other cubes of the tiling are obtained by translation of this origin voxel. More details about the model building process may be found in (Montagner et al, 2005c).

Geometrical relations between the digital grids are then identified by processing spatial coefficients  $\rho$ , i.e. volumes of the polyhedra resulting from the intersection between cubic voxels in general position and unit cubes (Fig. 6.d).

The intersection volume between two given voxels  $V$  and  $v$  is computed using an efficient cube intersection algorithm (Reveilles, 2001). The processing cost is lower than the one obtained using a general convex polyhedra intersection algorithm, thanks to the use of cube symmetries and resulting analytical formulas. The algorithm runs through the 6 faces of both cubes, processing a polygonal boundary of the volume at each iteration. Let  $I$  be the polygon obtained as the intersection between the support plane  $P$  of the current cube square face  $F$  and the other cube (Fig. 6.a). Analytical formulas provide 3D coordinates of this intersection in a canonical space (Fig. 6.b), and the polygon is then brought back to its real position using the octahedral group of cube symmetries (Fig. 6.e). At this stage,  $I$  is possibly larger than the real polygonal face. It is therefore restricted to its common part with  $F$  (Fig. 6.c) using an adapted version of the O'Rourke's general polygon intersection algorithm (O'Rourke, 1998).

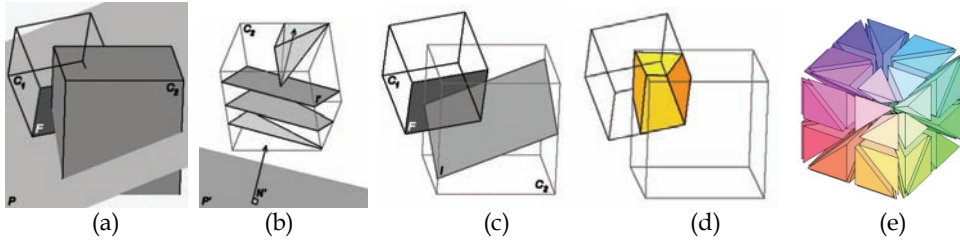


Fig. 6. a-c. Steps of the algorithm used to process the intersection polyhedron (d) between two cubic models of voxels  $C_1$  and  $C_2$ , using cube symmetries (e)

### 4.3 Final model aggregation stage: combination of numerical and geometrical information

If  $\rho$  is the result of the elementary fusion between basic elements  $v$  and  $V$  of the geometrical model, the final redistribution process can be expressed for general purpose as

$$\delta_{v,V} = \Delta\{\rho(v,V),g(V)\} \quad (2)$$

where  $\Delta\{\cdot,\cdot\}$  is the redistribution function, assigning to voxel  $v$  a part  $\delta_{v,V}$  of the information held by  $V$ , either in its direct form (initial image intensity in the case of the quantification process) or resulting from the image numerical modeling ( $g(\cdot)$  is a general information function). The final aggregation step then brings together the fusion operators from both domains to compute the result of the global fusion process. The elementary part of this result  $\varphi_{v,V}$ , for example in the spatial base of image  $I_1$  (e.g. at the highest spatial resolution) is processed as

$$\varphi_{v,V} = \Phi\{f(v),\delta_{v,V}\} = \Phi\{f(v),\rho(v,V),g(V)\} \quad (3)$$

where  $\Phi\{\cdot,\cdot\}$  is the aggregation operator, and  $f(\cdot)$  is another information function. Functions  $f(\cdot)$  and  $g(\cdot)$  represent information provided by image numerical models, or even stand for information derived from a first fusion stage in which spatial relations have no influence (case of the multistage use of the general fusion scheme).

Using the intersection volumes between voxels  $v$  and  $V$  as spatial coefficients  $\rho(v,V)$ , equation (2) thus simply becomes  $\delta_{v,V} = \rho(v,V)g(V)$ . Hence, the value assigned to voxel  $v$  is processed from the redistributed intensities of voxels  $V_i$  in a neighborhood of  $v$  as

$$\delta_v = \sum_{V_i \cap v} \rho(v,V_i)g(V_i) \quad (4)$$

where  $V_i \cap v$  refers to the set of voxels  $V_i$  having a non-null intersection with  $v$ . Formula (4) processes the value  $\delta_v$  associated with  $v$  as a combination of the information  $g(V_i)$  associated with surrounding voxels  $V_i$ , where the contribution of  $V_i$  is proportional to its common volume with  $v$ . As a matter of fact, this computation mode is very close to a linear interpolation of  $g(V_i)$ , in the case where  $v$  straddles several voxels  $V_i$ . But this process uses the maximum of information available in the image structure, thus minimizing the hypothesis required on the distribution of the physical phenomenon measured. When building image models, reference voxels  $v$  are moreover represented as unit volumes.

Spatial coefficients  $\rho(v, V)$  are thus numerical values ranging from 0 (no intersection between  $v$  and  $V$ ) to 1 (full intersection, i.e.  $v \subseteq V$ ), and  $\sum_{V_i \cap v} \rho(v, V_i) = 1$  for a given voxel  $v$ . The redistribution process is thus normalized so that each piece of information  $\delta_v$  naturally belongs to the same value range as the original numerical support of  $g(V)$ . In the case where  $v$  is fully included in  $V$  ( $\exists! V_i, V_i = V / V_i \cap v \neq \emptyset$ ), the information  $\delta_v = g(V)$  is kept unchanged since the whole unit volume stems from the intersection with a unique voxel  $V$ .

#### 4.4 Relations with data interpolation

As stated above, information redistribution expressed through equation (4) can be considered, from a computational point of view, as an interpolation of the numerical intensities from image  $I_2$ . But the interpolation is no longer uniform: the redistributed information is processed like in a nearest-neighbor interpolation when  $v \subseteq V$ , or is linearly interpolated with respect to intersection volumes when  $V$  intersects  $v$  boundaries. This adaptive way of processing minimizes the estimation error of the average activity in the set  $\{v_j / v_j \cap V \neq \emptyset\}$ , in comparison with a linear variation of  $\delta_{v_j}$  with respect to the distance to voxels  $V_i$ . Moreover, representing the spatial relation between input images through coefficients  $\rho$  establishes an accurate link between pieces of numerical information they hold without modifying them. This straightforward matching makes it possible to adapt numerical intensities stemming from image  $I_2$  to the content of image  $I_1$ , e.g. for measurement purpose. This second property will thus naturally find a direct application in the quantification of neuronal activity presented in section 5.3.

Finally, the redistribution principle is also characterized by two major methodological points: the nature of weighting factors in interpolation formulas, built from objective image spatial structure and relations, and the ability to introduce spatial coefficients into advanced aggregation models at the final aggregation step. Indeed, the general operation expressed by equation (2) can obviously be delayed, and split up into new parts of equation (3) so that coefficient  $\rho$ ,  $g(V)$  and  $f(v)$  are combined differently by the aggregation operator  $\phi$ .

## 5. Application to medical image analysis

### 5.1 Multispectral MR images fusion

The first application we propose is a 1 level/1 stage fusion scenario (Fig. 7.a), and concerns the fusion between several MR brain images stemming from different acquisition techniques. Images reflect the same type of knowledge (anatomical distribution of brain tissues), and provide complementary information with the use of different acquisition parameters. Numerous applications can benefit from the MR images fusion process, from abnormal tissue segmentation (tumors (Dou et al., 2006, Philipps et al., 1994), encephalitis, etc.) to the quantification of white matter, gray matter and cerebrospinal fluid volumes in normal (e.g. Wagner et al., 2006) or pathological people (e.g. Swayze et al., 2003).

The aim here is thus to extract  $C$  tissue classes from a set of  $n$  MR images. Following the theoretical framework proposed in this chapter, each MR image  $I_i$  first provides  $C$  distributions of possibility  $\pi_T^i$ , modeled as  $C$  fuzzy tissue maps. Given a tissue  $T$ , the corresponding  $C$  fuzzy maps have then to be fused using a fusion operator. Since the original MR images provide distinct but complementary information about the distribution

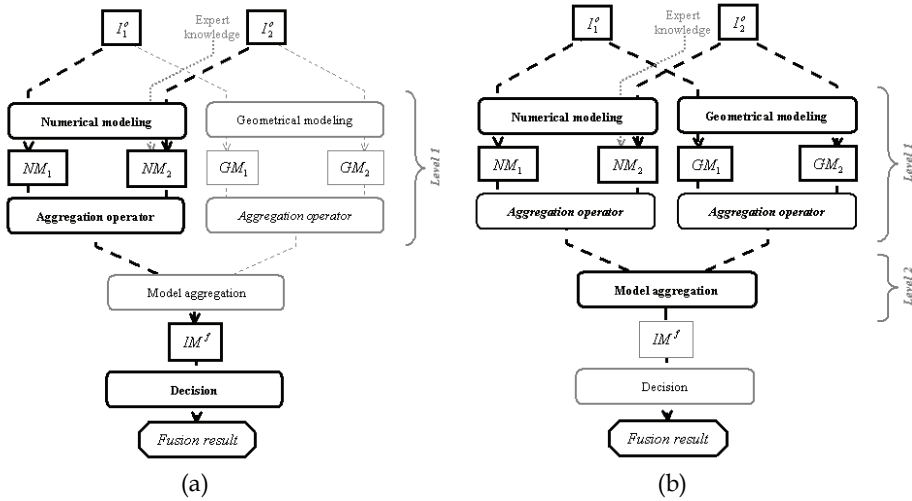


Fig. 7. Instantiation of the fusion scheme in the MR/MR fusion (a) and image synthesis examples (b). The decision step (defined in 2.1) concerns the production of segmented images in the first case, and is not applied in the second case (direct display of fusion result)

of  $T$  in the brain, the only areas of ambiguity may be due the transitions between  $T$  and neighboring tissues, or may be related to the pathological signature of  $T$  (e.g. a tumor imaged with a contrast agent may have a very significant hyper or hypo signal, not related to a pure anatomical acquisition). The fusion operator has thus not only to underline the redundancies between the  $\pi_T^i$ 's, but also to shed light on possible areas of conflict between the distributions of possibility. We illustrate the construction of the fusion operator in the case  $n=2$ , and the extension to any  $n$  is trivial (Dubois & Prade, 1992). If both distributions of possibility  $\pi_T^1$  and  $\pi_T^2$  agree and are reliable, a renormalized T-norm is used to aggregate the information:

$$\pi_T = \min(\pi_T^1, \pi_T^2) / h \tag{5}$$

where  $h = 1 - \frac{\sum |\pi_T^1(i) - \pi_T^2(i)|}{N}$  measures the agreement between both distributions of possibility. On the contrary, if only one of the distributions is reliable, the operator has to be cautious and gradually reports the confidence on the union of the distributions, guided by  $(1-h)$ , an estimation of the conflict:

$$\pi_T = \min[\max(\pi_T^1, \pi_T^2), 1 - h] \tag{6}$$

In order to manage both situation, the final operator acts as

$$\pi_T = \max\left(\frac{\min(\pi_T^1, \pi_T^2)}{h}, \min[\max(\pi_T^1, \pi_T^2), 1 - h]\right) \tag{7}$$

This gives for each tissue  $T$  a fused distribution of possibility, and each voxel  $i$  is thus defined by  $C$  values  $(\pi_1(i) \dots \pi_C(i))$ . Since the aim of the fusion is to provide a volumetric quantification of brain tissues, the decision stage labels a voxel  $i$  as belonging to tissue  $\hat{T}(i)$  such as

$$\hat{T}(i) = \text{ArgMax}_T \{\pi_T(i)\} \quad (8)$$

As test data, we used simulated MR images generated with the online MRI Simulator at the McConnell Brain Imaging Centre (BrainWeb, <http://www.bic.mni.mcgill.ca/brainweb/>) in Montreal. The data sets are based on an anatomical model of a normal brain that results from registering and preprocessing of 27 scans from the same individual with subsequent semi-automated segmentation. In this data set the different tissue types are well-defined, both fuzzy and crisp tissue membership are allocated to each voxel. From this tissue labeled brain volume the MR simulation algorithm, using discrete-event simulation of the pulse sequences based on the Bloch equations, predicts signal intensities and image contrast in a way that is equivalent to data acquired with a real MR-scanner ( $T_1$ -weighted,  $T_2$ -weighted and Proton Density). Both sequence parameters and the effect of partial volume averaging, noise, and intensity non-uniformity (RF) are incorporated in the simulation results (Kwan et al., 1999). In order to obtain the true, i.e. reference volumes, the voxels labeled as gray, white matter and CSF in the discrete brain phantom (noise=0%, RF=0%) were counted. 20 additional simulated BrainWeb data sets that were used are each based on an anatomical model of an individual normal brain (for details see (Aubert-Broche et al., 2006)).

Figure 8 presents the results obtained for the fusion of several couples of images of the model. The  $T_1/T_2$  fusion provides a very accurate WM map, a CSF map very close to the one provided by the  $T_2$  image only, and a GM map suffering from several drawbacks (mainly a poor definition of some gray matter structures (basal ganglia)). The  $T_1/PD$  and the  $T_2/PD$  fusions exhibit a poorly informative CSF map, since both acquisitions suffer from a lack on information on the distribution of this tissue, especially in sub-arachnoid spaces. The other fused distributions are very close to those provided by BrainWeb.

Figure 9 shows the final segmentation map, obtained with the three types of fusion. Each segmented map was assessed with respect to the reference map provided by BrainWeb, both using an expert evaluation (a neurosurgeon visually assessed the accuracy of the segmented maps), and quantitative indexes:

- the confusion matrix  $M$ , giving for each tissues  $T$  and  $T'$  the number  $M_{TT'}$  of voxels being classified as  $T$  in the BrainWeb segmentation, and as  $T'$  in the computed segmented image;
- the Tanimoto indexes (TI) (Duda et al., 2001) computed from  $M$ , and allowing an accurate comparison between two segmentation results;
- for each tissue  $T$ , the relative errors (RE) in volumetric quantification between the computed map and the reference one.

Table 1 presents some of these results for the three types of fusion. Globally, best results were obtained with the  $T_1/T_2$  fusion process, those images being indeed very discriminant for brain tissue segmentation (Kiviniity, 1984), but need a preliminary registration step.

Finally, Table 2 presents a comparison between the fusion method and three classical segmentation methods, either using a clustering algorithm on one image only, or a bidimensional histogram analysis in the  $(T_1, T_2)$  space. The best quantitative indexes were obtained using the fusion process, for all tissues.

	Réf	T <sub>1</sub> /T <sub>2</sub>		T <sub>1</sub> /PD		T <sub>2</sub> /PD				
	Vol (%)	TI	Vol (%)	RE <sub>T</sub> (%)	TI	Vol(%)	RE <sub>T</sub> (%)	TI	Vol(%)	RE <sub>T</sub> (%)
CSF	5,96	0.71	6.94	14.1	0.63	6.75	13.25	0.66	7,43	24.6
WM	23,5	0.82	22.72	3.32	0.85	23.13	1.57	0.75	24.14	2.72
GM	20,07	0.76	20.67	2.98	0.80	20.45	1.89	0.71	18.76	6.52

Table 1. Quantitative evaluation of the MR/MR fusion process






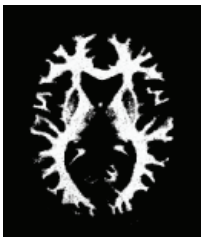
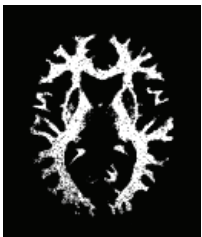


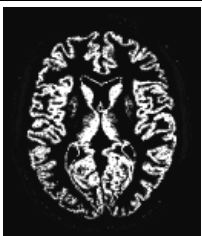

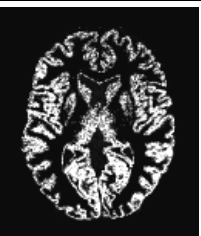
CSF				
WM				
GM				
	BrainWeb	T <sub>1</sub> /T <sub>2</sub>	T <sub>1</sub> /PD	T <sub>2</sub> /PD

Fig. 8. Fused tissue maps

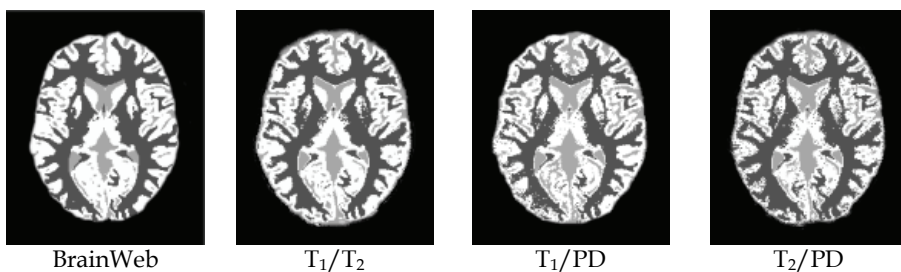


Fig. 9. Segmented images obtained from several fusion processes

	T <sub>1</sub> only		T <sub>2</sub> only		2D histo. analysis		T <sub>1</sub> /T <sub>2</sub> Fusion	
	TI	RE <sub>T</sub> (%)	TI	RE <sub>T</sub> (%)	TI	RE <sub>T</sub> (%)	TI	RE <sub>T</sub> (%)
CSF	0.63	14.2	0.79	13.6	0.67	15.98	0.77	14.1
GM	0.82	3.87	0.58	7.81	0.78	5.12	0.82	3.32
WM	0.76	3.14	0.65	5.45	0.76	3.05	0.76	2.98

Table 2. Comparison of segmentation results between the fusion method and classical algorithms

## 5.2 Image synthesis from multimodal information

This application of the general fusion scheme aims at accurately locating the functional information extracted from a SPECT image of the brain with respect to underlying anatomical structures. The difficulty of representing both complete pieces of information on the same image was pointed out in early studies on this subject (Hill, 1993), stating that the efficiency of displaying superimposed color layers decreases when the number of image features to show increases. Conversely, an advanced image fusion strategy, such as the one proposed in this chapter, can avoid spatial covering or frequency mixing by selecting features from each image that are relevant for diagnosis.

The image fusion is performed between an MR image, and either a brain perfusion or a neurotransmission SPECT image. The first functional imaging modality provides diagnostic information for brain pathologies where a visible reduction of blood supply is representative for the dead of neuronal cells in the related region of the brain, such as Alzheimer type dementia. The second one gives information about a specific neuronal activity, possibly responsible for pathologies such as Parkinson's disease and Parkinsonian syndromes (low level in target anatomical structures, mostly putamens and heads of caudate nuclei). In both cases, the difference in spatial resolution between MR and SPECT images requires a geometrical modeling of image spatial features, i.e. using the 2-level fusion capability of the fusion scheme (Fig. 7.b).

Referring to formula (2), the information function  $g(\cdot)$  is a direct reference to initial values associated with voxels  $V$  from image  $I_2 = I_f$  holding the functional information. Using the redistribution principle, the final model aggregation assigns to voxel  $v$  the redistributed intensity  $\delta_v$ , processed from formula (4) with  $g(V) = I_f(V)$ , the original numerical value associated with  $V$  in the SPECT image. Formula (4) thus finally becomes

$$\delta_v = \sum_{V_i \cap v} \rho(v, V_i) I_f(V_i) \quad (9)$$

Let  $\pi_C(v)$  be the fuzzy membership degree of voxel  $v$  to the tissue class  $C$  (white matter, gray matter, and cerebrospinal fluid) from the numerical model of the MR image (computed as explained in section 3.2). An activity  $\mu_C$  is associated with each anatomical class, either from an arbitrary gray level adapted to the human visual perception to emphasize important information, or from a mean functional activity for class  $C$ . In each case,  $\mu_C$  may be considered as the second part of the numerical model representing image  $I_2$ . Anyway, the anatomical information is preserved in the global shape of brain tissue classes, and local variations of the functional activity within each class are injected, in addition to  $\mu_C$  in the proposed model, through the adapted form of equation (3):

$$\varphi_v = \frac{\sum_C \pi_C(v) \mu_C + \delta_v}{\sum_C \pi_C(v) + \sum_{V \cap v} \rho(v, V)} \quad (10)$$

where  $\delta_v$  is the redistributed functional activity mixing the geometrical information with original intensities from  $I_2 = I_f$  (first part of the numerical model for this image). The other part of the numerator stems from the aggregation of numerical models only,  $\mu_C$  (second part of numerical model for  $I_2$ ) being independent of any spatial context. Formula (10) is thus a good example of the introduction of spatial coefficients into advanced aggregation models detailed in section 4.4.

The image synthesis process has been applied to both brain perfusion SPECT ( $[^{99m}\text{Tc}]\text{-ECD}$ ) and neurotransmission SPECT ( $[^{123}\text{I}]\text{-FP-CIT}$ ) images, in the case of a patient suffering from a multiple system atrophy (MSA, Parkinsonism plus syndrome – Fig. 1). A single  $T_1$ -weighted MR data set is associated with perfusion and neurotransmission SPECT images, which were simultaneously acquired (El Fakhri et al., 2001). Both SPECT images and the pre-processed MR image are isotropic, with voxel sizes of 2.33 mm and 1.5 mm respectively. The quality of synthetic images (Fig. 10) was assessed by an expert, focusing on the following diagnosis elements: activity peaks and lacks, shape and position of anatomical structures. He answered the following questions for both the brain perfusion and the neurotransmission images:

1. Is the ability to locate functional activity in relation to anatomical structures really improved?
2. Are diagnosis elements clearly visible on the synthetic image?

For purpose of assisting the diagnosis task, the perfusion information (Fig. 10.a to d) has to be emphasized in the region of cerebral cortex (gray matter). The concentration level of the tracer within CSF structures is null. Variations are visible in the WM and mostly in the GM (activity ratio estimated to 1/4), corresponding to the above requirement. High activity levels in the cortical region are clearly visible, this effect being reinforced by the already high mean activity of this structure. Nevertheless, hypoperfusion zones are mostly visible in regions with low base activity, i.e. thanks to remaining diffused activity in the surrounding white matter.

In the case of the neurotransmission image (Fig. 10.e to h), the high activity bound in the striatum implies a decrease of visual contrast in the surrounding region. However, deducing the shape and position of anatomical structures remains possible thanks to the outlines of close tissue classes. Indeed the functional activity presents only low variations outside the striatum (CSF in the ventricular system and GM in the cortex). Likewise, edges of the putamens and caudate nuclei can also be deduced from areas with a low activity levels within the striatum itself (see Fig. 10.h). The synthesis process thus emphasizes the information brought by lacks of activity.

### 5.3 Quantification of functional activity using a multistage information fusion

The solution we propose for measuring brain activities in a SPECT image uses both the multilevel (management of multiresolution discrete data) and the multistage aspects of the fusion architecture (Fig. 11.a). This last point refers to modeling image activities as binary maps of subcortical brain structures used as measurement ROIs. This segmentation task is achieved as a second iteration of the fusion process (1 level/1 stage) with MR images and symbolic information as inputs (Fig. 11.b). We refer the reader to (Barra & Boire, 2001b) for a detailed description of this modeling stage.



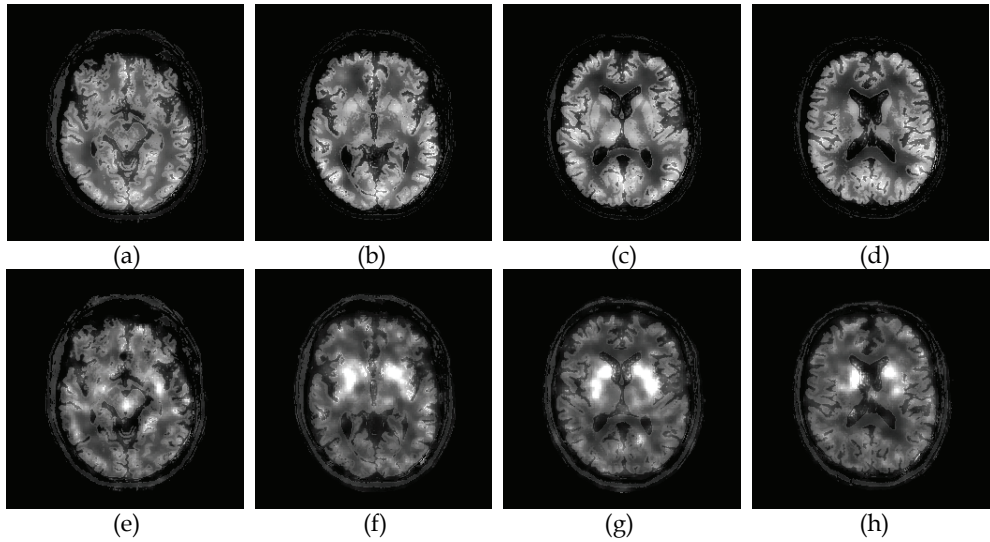


Fig. 10. Image synthesis from MR/perfusion fusion (a-d) and MR/neurotransmission fusion (e-h)

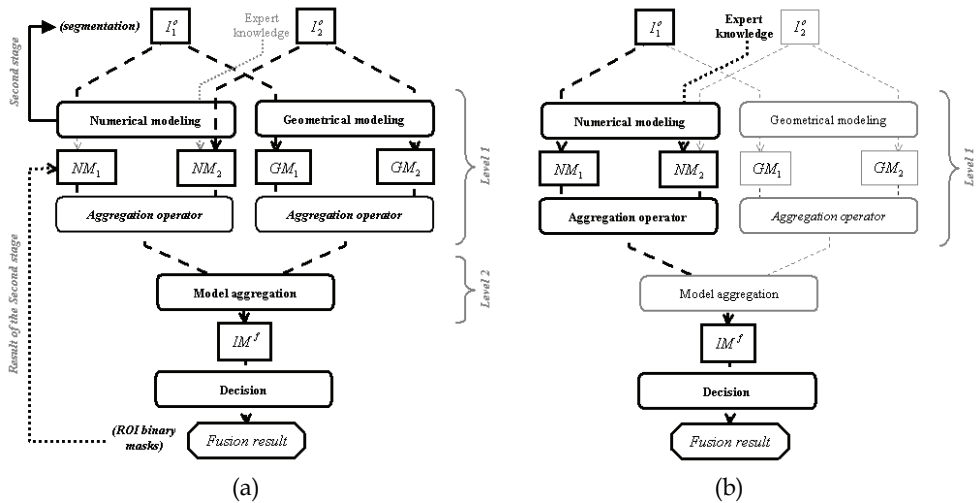


Fig. 11. Fusion scheme for the example of a brain activity quantification process (a); Original intensities from  $I_2^0$  (SPECT) are combined, through spatial coefficients, with ROIs Boolean models stemming from the second fusion stage (b); The decision in (a) may consist in classifying the whole SPECT exam according to the quantification index  $IM^f$

As in section 5.2 (and referring to equation (2)), the numerical model of a voxel  $V$  from the SPECT image is  $g(V) = I_f(V)$ , the original numerical value associated with  $V$ . Let  $S_a$  be this binary mask representing the brain structure of interest ('a' standing here for "anatomical"). The contribution of voxel  $v$  to the fusion result may thus be expressed as

$$\varphi_v = \sum_{V_i \cap v} \varphi_{v, V_i} = \sum_{V_i \cap v} S_a(v) \rho(v, V_i) I_f(V_i) = S_a(v) \sum_{V_i \cap v} \rho(v, V_i) I_f(V_i) = S_a(v) \delta_v \tag{11}$$

since formula (3) is also multiplicative with  $f(v) = S_a(v)$  (crisp membership) when using intersection volumes as spatial coefficients, and referring to equation (9). Consequently, the functional activity in out-of-structure anatomical voxels, in which the numerical expression of this activity is mostly due to the multiresolution partial volume effect (PVE), is naturally cancelled. The final result of the fusion process can be expressed as

$$\varphi = \frac{(\varphi_{S_t} - \varphi_{S_{ns}})}{\varphi_{S_{ns}}} \tag{12}$$

and is seen as an estimation of the radiotracer binding potential (Soret et al., 2003).  $\varphi_{S_t}$  is the global activity of the target anatomical structure (we use mean values), processed from the binary mask  $S_t$  through equations (11). In the same way, the quantity  $\varphi_{S_{ns}}$  is processed for a non-specific region of the brain (i.e. not influenced by the studied pathology).

The following tests have been carried out for a methodological assessment of the proposed quantification process. Since the method has been designed to overcome partial volume effects due to multiresolution, the assessment protocol aims at evaluating the accuracy of activity measurement from highly PVE-prone ROIs, in the context of several ratios of spatial resolutions between input images (this kind of PVE is mostly due to voxels from the functional image located at the external bound of the measurement region). The quantification has been performed on series of ROIs with a reference shape but different volumes (slice selection). Performances have been compared with the method considered as reference: registration of the MR image on the SPECT image.

The test dataset is made of a numerical phantom dedicated to the study of deep brain structures and dopamine neurotransmission phenomena involved in Parkinson’s disease and Parkinsonian syndromes. It results from a Monte-Carlo simulation of the striatum-based neurotransmission (El Fakhri et al., 2001) using theoretical binding values in associated structures from a brain morphological phantom (Zubal et al., 1996). This phantom is also used as an anatomical reference for the data fusion process, and specially to build the image geometrical model. The functional image resulting from the simulation is initially aligned with the input data. A linear transform is applied to anatomical data to preserve the original functional information (including image rescaling by various factors, so that the ratio between voxel sizes is successively 1/2, 1/4 and 1/8). The initial measurement ROI is limited to right and left putamens. In simulated data, the tracer fixation ratio between putamens and the non specific (NS) reference region (here the occipital cortex) corresponds to a healthy case.

Table 3 shows the results obtained when quantifying the phantom activity within regions described above. Results are assessed in terms of relative error  $|\varphi - \varphi_{ref}| / \varphi_{ref}$ , where  $\varphi$  is the estimated binding potential (BP) processed through equation (12) with  $\varphi_{S_t}$  relative to the truncated putamens and  $\varphi_{S_{ns}}$  standing for the non-specific occipital cortex region.  $\varphi_{ref}$  refers to the same quantity, processed from the *a priori* model activities. Error values are averaged from quantities measured after applying linear transforms with 5 sets of parameters (rotation angles and translation vectors).

The quantification error is always lower for the redistribution method. The maximum difference is obtained for a resolution ratio of 1/4, due to the value quantified within the non specific region, that has a high influence on the BP estimated (see equation (12)). Conversely, the irregular and thin shape of this structure allow results to be quite constant (between ratios

1/2 and 1/4) using the MR registration method. The higher the resolution ratio, the more constant the size of the interpolated ROI, because of the effects of thresholding mask coefficients. The minimum error always corresponds to the largest measurement ROI (close to initial volume), and the maximum value to the thinnest one, except in the case of redistributions with a ratio of 1/8, due to the influence of the NS region again.

		Resol. ratio 1/2		Resol. ratio 1/4		Resol. ratio 1/8	
		Error (%)	Volume	Error (%)	Volume	Error (%)	Volume
MR image regist.	Min.	20.4	9909	24.8	9625	44.6	8859
	Max.	34.0	1341	24.9	9710	44.6	8859
Redistribution	Min.	11.8	9923	2.9	9918	24.16	1961
	Max.	24.7	1980	8.6	1691	31.2	9920

Table 3. Minimum and maximum relative quantification errors (depending on the actual volume of the measurement ROI, also indicated in cm<sup>3</sup> for comparison with the volume of the original putamen structure = 10<sup>4</sup> cm<sup>3</sup>, that provided the quantitative reference value)

## 6. Conclusion

The collection of anatomical and functional images, as well as the expert knowledge and habits play nowadays in clinical routine an important role for the study of a given pathology. The clinician merges and aggregates all this complementary, redundant and sometimes conflicting information to provide a better diagnosis. We proposed in this chapter a theoretical framework mimicking this aggregation process, based on the use of fuzzy logic modeling, fusion operators, and we enrich this classical fusion process with the introduction of spatial information modeling. This allows the information to be preserved until the final fusion step, and gives the opportunity to introduce the original image information into complex fusion operators. We provide the clinician with several outputs, from segmented images to quantitative indexes or synthesis images, and we think our process to be generic enough to allow the introduction of other information sources.

The multilevel information fusion is applied to three clinical applications, involving anatomical and functional images, and also geometrical and structural information. Results prove the efficiency of the approach, and shed light on several potential new applications not only in brain imaging, but also in the multimodal study of other organ, or even to other branches where several images, with several geometrical properties, are used, registered and merged.

## 7. References

- Aguilar, M. & New, J.R (2002). Fusion of multi-modality volumetric medical imagery, *Proceedings of the Fifth International Conference on Information Fusion*, 2:1206-1212
- Aubert-Broche, B.; Collins, D. & Evans, A. (2006). A new improved version of the realistic digital brain phantom. *Neuroimage*, 32:138-145.
- Barra, V., & Boire, J. Y. (2000a), Tissue Characterization on MR Images by a possibilistic Clustering on a 3D Wavelet Representation, *Journal of Magnetic Resonance Imaging*, 11:267-278.
- Barra, V., & Boire, J. Y. (2000b). Aggregation of anatomical and functional Information by a MR/SPECT fusion process: Application to neurodegenerative Pathologies—*Proceedings of the Sixth Annual Meeting of the Organization for Human Brain Mapping*, San Antonio. *NeuroImage* 11.

- Barra, V. & Boire, J.Y (2001a): A general framework for the fusion of anatomical and functional medical images. *NeuroImage* 13:410-424.
- Barra, V. & Boire, J.Y (2001b): Automatic Segmentation of subcortical brain Structures in MR Images using Information Fusion, *IEEE Transactions on Medical Imaging*, 20:549-58.
- Bloch, I. (1996). Information combination operators for data fusion: a comparative review with classification. *IEEE Transactions on Systems, Man, and Cybernetics* 1:52-67.
- Bloch I. & Maître H. (1997). Fusion of Image Information under Imprecision. In B. Bouchon-Meunier, editor, *Aggregation and Fusion Of Imperfect Information*, Series Studies in Fuzziness. Physica Verlag, Springer.
- Bloch, I., Géraud, T. & Maître, H. (2003). Representation and fusion of heterogeneous fuzzy information in the 3D space for model-based structural recognition - Application to 3D brain imaging. *Artificial Intelligence* 148:141-175.
- Boussion N., Hatt M., Lamare F., Bizais Y., Turzo A., Cheze-Le Rest C. & Visvikis D. (2006). multiresolution image based approach for correction of partial volume effects in emission tomography, *Physics in Medicine and Biology*, 51:1857-1876.
- Catafau A. M. (2001). Brain SPECT of dopaminergic neurotransmission: a new tool with proved clinical impact, *Nuclear Medicine Communications*, 22:1059-1060.
- Colin, A. & Boire, J.Y (1999). MRI-SPECT fusion for the synthesis of high resolution 3D functional brain images: a preliminary study. *Computer Methods and Programs in Biomedicine* 60: 107-116.
- Dasarathy, B. V. (1997). Sensor fusion potential exploitation-innovative architectures and illustrative applications. *Proceeding of the IEEE*, 85(1):24-38.
- Dubois, D. & Prade, H. (1992). Combination of Information in the Framework of Possibility Theory, In: *Data Fusion in Robotics and Machine Intelligence*, M. Al Abidi et al. Eds, New York Academic.
- Dubois D. & Prade H. (1994). Possibility theory and data fusion in poorly informed environments. *Control Engineering Practice*, 2:811-823.
- Dubois D. & Prade H. (2004). Possibilistic logic: a retrospective and prospective view, *Fuzzy Sets and Systems* 144:2-23.
- Duda, R., Hart, P. & Stok, D. (2001). *Pattern classification*, New York: John Wiley & Sons.
- Dou, W.; Ruan, S.; Chen, Y.; Bloyet, D. & Constant, JM. (2006). A framework of fuzzy information fusion for the segmentation of brain tumor tissues on MR images, *Image and Vision Computing*, 25(2):164-171.
- El Fakhri G., Moore S.C., Maksud P., Aurengo A. & Foley Kijewski M. (2001). Absolute Activity Quantitation in Simultaneous 123I/99mTc Brain SPECT, *Journal of Nuclear Medicine*, 42:300-308.
- Hill D. (1993). Combination of 3D medical images from multiple modalities, *Thesis, University of London, London, England*.
- Hudson H. & Larkin R. (1994). Accelerate image reconstruction using ordered subsets of projection data, *IEEE Transactions on Medical Imaging* 13: 601-609.
- Julin, P., Lindqvist, J., Svensson, L., Slomka, P., & Wahlund, L. (1997). MRI-guided SPECT measurements of medial temporal lobe blood flow in Alzheimer's disease. *Journal of Nuclear Medicine*. 38:914-919.
- Kiviniitty, K. (1984). NMR relaxation Times in NMR Imaging, *Annals of Clinical Research*, 40 4-6.
- Krishnapuram R. & Keller J.A. (1993), A Possibilistic Approach to Clustering, *IEEE Transactions on Fuzzy Systems*, 1:98-110.
- Krishnapuram R. & Keller J.A. (1996), The Possibilistic C-Means Algorithm: Insights and Recommendations, *IEEE transactions on Fuzzy Systems*, 4:385-393.

- Kwan, R.; Evans, A. & Pike, G. (1999). MRI simulation-based evaluation of image processing and classification methods, *IEEE Transactions on Medical Imaging*, 18:1085-1097.
- Lennon, M.; Mouchot, M.C.; Mercier, G.; Solaiman, B. & Hubert-Moy, L. (2000) Segmentation of airborne hyperspectral images by integrating multi-level data fusion, *Proceedings of the Third International Conference on Information Fusion*, vol.2, 10-13 July 2000.
- Matsopoulos, G.K., Marshall, S. & Brunt, J.N.H. (1994). Multiresolution morphological fusion of MR and CT images of the human brain. *Proceedings of the IEEE Vision, Image and Signal Processing Conference* 141:137-142.
- Montagner J., Barra V. & Boire J.-Y. (2005a), Synthesis of a functional information with anatomical landmarks by multiresolution fusion of brain images, in *Proceedings of the 27th IEEE Engineering in Medicine and Biology Society Conference*, Shanghai, China.
- Montagner, J., Barra, V., Reveillès, J.P. & Boire, J.Y. (2005b). Multiresolution images fusion for the quantification of neuronal activity: a discrete approach. *Proceedings of the 3rd IASTED International Conference on Biomedical Engineering*, Innsbruck-Austria 1-6.
- Montagner, J., Barra, V. & Boire, J.Y. (2005c). Modeling of a multimodal image aggregation process using discrete geometry. *Proceedings of the 1st Open International Conference on Modeling and Simulation*, Clermont Ferrand-France 431-439.
- O'Rourke J. (1998). Computational geometry in C, 2nd ed. New-York, USA: Cambridge University Press.
- Pajares, G. & De La Cruz, J.M. (2004). A wavelet-based image fusion tutorial, *Pattern Recognition*, 37:1855-1872.
- Philipps, W.; Velthuizen, R.; Phuphanich, S.; Hall, L.; Clarke L. & Silbiger M. (1995). Application of Fuzzy C-Means Algorithm Segmentation Technique for Tissue Differentiation in MR Images of a Hemorrhagic Glioblastoma Multiforme, *Magnetic Resonance Imaging*, 13(4):277-290.
- Reveillès J.-P. (2001). The geometry of the intersection of voxels spaces, *Electronic Notes in Theoretical Computer Science*, 46:285-308.
- Rousset O. G., Ma Y., & Evans A. C. (1998). Correction for partial volume effects in PET: principle and validation. *Journal of Nuclear Medicine*, 39:904-911.
- Shafer, G. (1976). A mathematical theory of evidence. Princeton University Press, Princeton.
- Soret M., Malick-Koulibaly P., J.Darcourt, Hapdey S. & Buvat I. (2003). Quantitative accuracy of dopaminergic neurotransmission imaging with 123I SPECT, *Journal of Nuclear Medicine*, 44:1184-1193.
- Swayze, V.; Andersen, A.; Andreasen, N.; Arndt, S.; Sato, Y. & Ziebell, S. (2003). Brain tissue volume segmentation in patients with anorexia nervosa before and after weight normalization, *International Journal of Eating Disorders*, 33(1):33-44.
- Wagner, A.; Greer, P.; Bailera, U.; Frank, G.; Henry, S.; Putnam, K.; Meltzer, C.; Ziolkowski, S.; Hoge, J.; McConaha, C. & Kay, W. (2006). Normal Brain Tissue Volumes after Long-Term Recovery in Anorexia and Bulimia Nervosa, *Biological Psychiatry*, 59:291-293.
- Wald L. (2002) Data fusion : definitions and architectures - Fusion of images of different spatial resolutions, *Les Presses de l'Ecole des Mines (pub.)*, Paris, France.
- Wang Y, Freedman M.T., Xuan J., Zheng Q., & Mun S.K. (1998). Multimodality medical image fusion: Probabilistic quantification, segmentation, and registration, *Proceedings of SPIE: Image display*, San Diego CA, 22-24,3335:239-249.
- Zadeh, L.A. (1978). Fuzzy sets as a basis for a theory of possibility. *Information and Control*, 8:338-353.
- Zubal, I. G., Harrell, C. R., Smith, E. O., Smith, A. L. & Krischlunas, P. (1996). High resolution, MRI-based, segmented, computerized head phantom. *Proceedings of the 6th International Radiopharmaceutical Dosimetry Symposium*, Gatlinburg, 319-324

# Anomaly Detection & Behavior Prediction: Higher-Level Fusion Based on Computational Neuroscientific Principles

Bradley J. Rhodes, Neil A. Bomberger, Majid Zandipour, Lauren H. Stolzar,  
Denis Garagic, James R. Dankert and Michael Seibert  
*BAE Systems, Advanced Information Technologies, Burlington, MA,  
U.S.A.*

## 1. Introduction

Higher-level fusion aims to enhance situational awareness and assessment (Endsley, 1995). Enhancing the understanding analysts/operators derive from fused information is a key objective. Modern systems are capable of fusing information from multiple sensors, often using inhomogeneous modalities, into a single, coherent kinematic track picture. Although this provides a self-consistent representation of considerable data, having hundreds, or possibly thousands, of moving elements depicted on a display does not make for ease of comprehension (even with the best possible human-computer interface design). Automated assistance for operators that supports ready identification of those elements most worthy of their attention is one approach for effectively leveraging lower-level fusion products.

A straightforward, commonly employed method is to use rule-based motion analysis techniques. Pre-defined activity patterns can be detected and identified to operators. Detectable patterns range from simple trip-wire crossing or zone penetration to more sophisticated multi-element interactions, such as rendezvous. Despite having a degree of utility, rule-based methods do not provide a complete solution.

The complexity of real-world situations arises from the myriad combinations of conditions and contexts that make development of thorough, all-encompassing sets of rules impossible. Furthermore, it is also often the case that the events of interest and/or the conditions and contexts in which they are noteworthy can change at rates for which it is impractical to extend or modify large rule corpora. Also, pre-defined rules cannot assist operators interested in being able to determine whether any unusual activity is occurring in the track picture they are monitoring. Timely identification and assessment of anomalous activity within an area of interest is an increasingly important capability—one that falls under the enhanced situational awareness objective of higher-level fusion.

A precursor of being able to automatically notify operators about the presence of anomalous activity is the capability to detect deviations from normal behavior. To do this, a model of normal behavior is required. It is impractical to consider a rule-based approach for achieving such a task, so an adaptive method is required: that is, a capability to learn what is normal in a scene is required. This normalcy representation can then be used to assess new data in order to determine their degree of normalcy and provide notification when any

activity deviates beyond some level of tolerance from the learned representation of normal behavior. Additionally, learned normalcy models can be used to predict future vessel behavior over timescales beyond the capabilities of standard track fusion/stitching engines—that is, on the order of hours to days (depending on the application domain). Learning of kinematic normalcy models for use in anomalous behavior detection and future behavior prediction are the core elements of the problem addressed in this chapter.

Numerous real-world constraints must be addressed when developing such capabilities if the results are to have any practical, fieldable utility. Key drivers for these constraints are that new track data are continually collected, that activity patterns can change over time, and that operators play no more than a limited role in guiding the evolution of the activity pattern learning system. It is not realistic to train models in batch mode where all data contributing to the learned representation have to be available prior to training onset. Periodic re-training with datasets of ever increasing size is also untenable. A static representation (one that is trained from available data then frozen for use against new data) is a suspect approach for situations where activity patterns are not static. Given the potentially huge amount of data to be processed, it is not reasonable to expect that labels indicating which tracks (or portions thereof) are normal and which are not will be applied to the data. On the other hand, sometimes relevant data are relatively scarce. So, in addition to being able to handle very large amounts of data, it is also important to be able to learn useful representations from limited amounts of data. These factors further define the problem addressed here.

Additional considerations also inform our approach. For instance, it is self-evident from any real-world situation that behavior is often contingent on ambient context. For example, travel patterns of individuals would (it is to be hoped) differ between weekdays and weekends or between daytime and nighttime. These are relatively simple contexts, but even so, they provide an important role in helping produce accurate representations of normalcy. Another example would differentiate between peak hour and non-peak hour periods when considering traffic activity patterns on a highway. During non-peak hours, stopped vehicles (or even those moving slowly) would be unusual, and thus worthy of attention (or even suspicion), whereas during peak-hours slowly moving traffic may be the norm. Some contexts are far more subtle or difficult to determine *a priori*. Consider the case of a relatively permanent change in daily travel of an individual who changes jobs. Importantly, that individual's initial visit(s) to the new job location – during the interview process, for instance – would have registered as deviations from normal workday travel patterns. If the job location were available as context data to the system, then a new model for workday travel could be learned once the individual's status had been updated. In the absence of such context information, the original model would slowly be adapted to the new pattern due to the incremental learning that takes place. Prior to the new pattern becoming mature in the model, this pattern would still be considered deviant.

To partially address this type of shortcoming, our learning approach can take advantage of externally-generated feedback about its performance to refine the learned representations. Although they do not ever need supervision to learn normalcy models, our algorithms can certainly exploit human subject matter expertise. Via reinforcement learning, operators can influence the learned models in a number of ways. For instance, regarding the last example above, if an operator determines that the new pattern of workday travel is indeed normal, then that pattern can be selected from a display, labelled as normal, and fed to the learning algorithm, which would then label clusters associated with this new pattern as normal. In effect, this speeds up the learning process on the basis of superior human insight. By the

same token, an operator could select a trajectory that the learned model considers normal and indicate that it is to be considered anomalous, whereupon the system would thenceforth consider any similar trajectory anomalous and produce corresponding notification. Another route to reinforcement learning is available via responses to anomaly detections. An operator can respond to notifications with agreement or disagreement. Disagreement indicates that the responsible behaviour is not anomalous and should be considered normal.

Neurobiological systems such as the human central nervous system are eminently suited to the challenges of such problems, so we draw inspiration for the development of automated high-level fusion support systems from computational neuroscience. Complementary, neurobiologically-inspired learning algorithms reduce massive amounts of data to a rich set of information embodied in models of behavioral patterns represented at a variety of conceptual, spatial, and temporal levels. Our approach, based on neurobiological principles, learns incrementally as new data are available, adapts learned models as underlying activity patterns change, and does not rely on labeled data for learning. Before presenting our approach in more detail, a brief survey of related work follows.

## 2. Related work

Beyond that from our group, the literature on trajectory-based motion learning and pattern discovery for the type of surveillance outlined in the introduction to this chapter is relatively sparse, largely due to the nature of the application. However, the more limited field of video-based surveillance (surveyed in Hu *et al.*, 2004a and Liao, 2005) has reported advances using a variety of approaches, including Learning Vector Quantization (LVQ) (Johnson & Hogg, 1996), Self-Organising Maps (SOMs) (Owens & Hunter, 2000), hidden Markov Models (HMMs) (Alon *et al.*, 2003), fuzzy neural networks (Hu *et al.*, 2004b), and batch expectation-maximization (EM) (Makris & Ellis, 2005). Most of these techniques attempt to learn high-level motion behavior patterns from sample trajectories using discrete point-based flow vectors as input to a machine learning algorithm. For realistic motion sequences, convergence of these techniques is slow and the learning phase is usually carried out offline due to the high dimensionality of the input data space. In addition, many of these algorithms use supervised and/or batch learning and require statistically sufficient amounts of data for constructing normalcy models of motion pattern behavior upon which to base anomaly detection and prediction. A noteworthy example that uses on-line clustering has been reported by Piciarelli and Foresti (2006). Alas, the dependence of their approach upon data acquisition at fixed time intervals for encoding of temporal information in their representation is a limitation that cannot generally be satisfied in real-world applications.

Our work addresses a wider range of issues relevant to real-world applicability and utility than the approaches noted above. We use incremental, unsupervised learning of non-statistical and statistical representations to deal with variable amounts of data. This produces usable normalcy models early in the learning process while data are still limited, yet refines the specificity of the models as additional data become available.

## 3. Event-level normalcy learning and anomaly detection

Our approach for detecting anomalous behavior is to assume that normal activity occurs frequently, while activity that is sufficiently different from normal activity is rare and



anomalous. In addition, it must be possible to incorporate explicit knowledge about normal and anomalous activity when it is available. One example is a set of vessel traffic data that has been analyzed by an operator who has verified that it contains no anomalous activity. Such input may also occur after the training data has already been presented; for example, an operator is able to select a vessel track from live data and indicate that its activity is normal. Thus, it is required that normalcy models be (1) learned continuously in response to incoming vessel track data, (2) adaptable to operator input, and (3) capable of recovering from operator mistakes.

To learn context-sensitive models of vessel behavior, we have developed a neural network classifier which incrementally constructs a multidimensional Gaussian (hyper-ellipsoid) model of each category that is relatively insensitive to outliers and learns the normal pattern of behavior independent of the feature dimensions comprising the learning hyper-space. When a new data point falls into a particular category, the network updates its parameters adaptively to the incoming data and provides an accurate measure of normal/anomalous behavior. When a new data point is sufficiently beyond all learned categories, then a new category is formed. During classification, the network reports the distance from the data point to its closest category. If this distance is not within the predefined settable threshold, then that point is reported as a deviation from normalcy. The maximum size of each category hyper-ellipsoid is also a predefined location- and dimension-dependent variable, which controls the representational fineness by constraining the size of each category. The network is capable of learning (by updating the categories and their associated hyper-ellipsoids) and classifying (by comparison to the latest hyper-ellipsoid models) data on-the-fly without any operator intervention. As each model matures, the gradient of certain model parameters reaches an asymptote that can be automatically checked for and utilized to activate models for classification purposes. The speed and performance of this learning algorithm makes it suitable for real-time situations wherein an operator/analyst can interactively facilitate the learning process and/or control over the sensitivity level of system alerting to control false alarms. These reasons also make this technology suitable for event-level learning in maritime domain awareness (MDA) or other tracking applications.

### 3.1 Example results

Figure 1 illustrates a two-dimensional projection of the learned representation from vessel track data recorded in the Miami Harbor vicinity during August 2004. Each category is represented by an ellipse, which accounts for 99% (3 standard deviations around the mean) of the data within that category. One aspect of this learned representation is worthy of note here. Panning from west to east (left to right) across the figure the potential locations of vessels become less constrained. In fact, in the east-most section of the region, the learned representation spans the location space. It should also be noted that the great majority of the learned category ellipses in the east-most area are uniformly pale, an indication that the pattern of travel within this area does not follow particular navigation routes. The darker ellipses indicate higher traffic areas. Figure 2 shows the learned 4-dimensional model of same model illustrated in Figure 1. Note that as vessels get closer to the port, they reduce their speed and travel in east-west direction (red-blue ellipsoids) through a narrow channel. The left panel in Figure 3 shows the percentage of track reports as a function of Mahalanobis distance to the center of closest category for a two-dimensional model (based on position: longitude and latitude) of each individual vessel. The thick black curve shows the mean

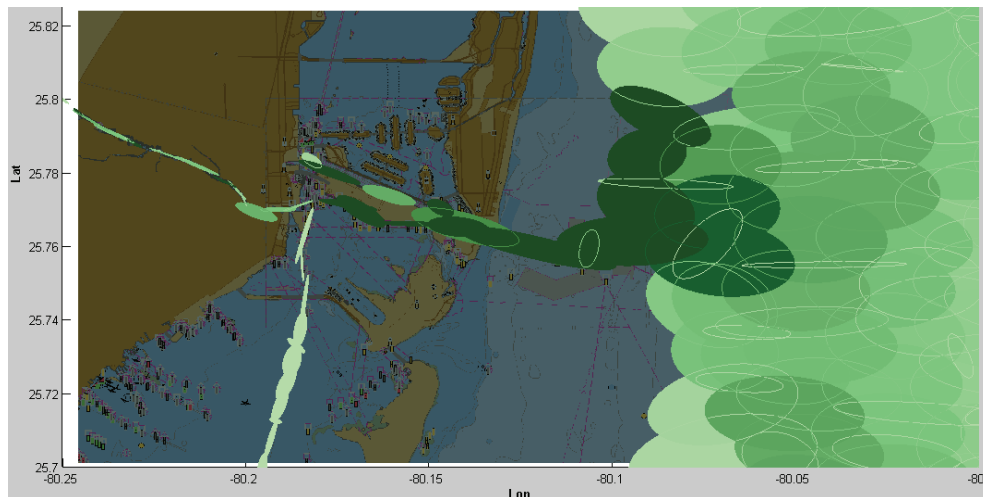


Fig. 1. Two-dimensional depiction of normalcy model learned from six months of real AIS vessel surveillance data from the Miami Harbor area (based on 4 dimensions – latitude, longitude, speed, and course). A map of the relevant region is overlaid with the learned representation of normal event activities as a set of shaded ellipses. Darker shading is proportional to the number of observations in an ellipse.

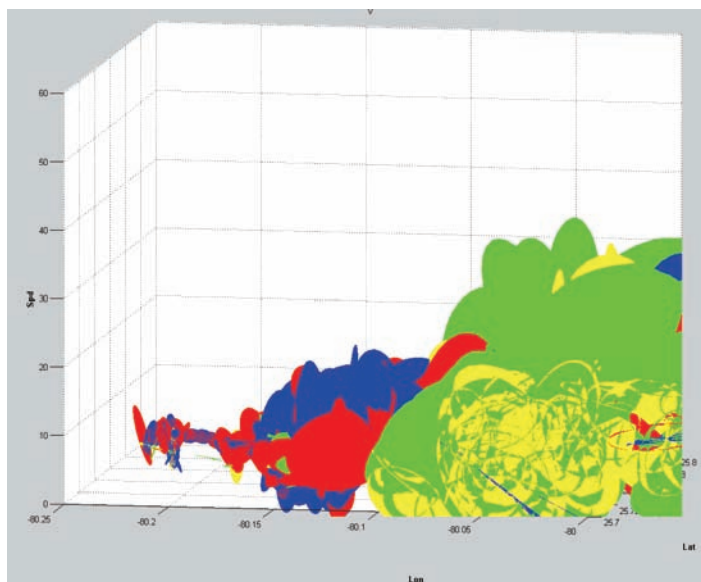


Fig. 2. Four-dimensional depiction of learned model illustrated in Figure 1. Ellipse coloring indicates principal vessel course: red = eastward, blue = westward, green = northward, yellow = southward. Towards the harbor region velocity decreases (as indicated by the lower ellipses).

across all vessels with more than 2000 track reports. The right panel in Figure 3 shows the mean of percent track reports for 2D,  $3D_{\text{speed}}$  (position, and speed),  $3D_{\text{course}}$  (position and course) and 4D (position, speed, and course) models of normalcy patterns. Note that all four curves approximately follow a Gaussian distribution pattern. As each category accumulates more data, the distribution of data within each category becomes closer to a Gaussian distribution.

In order to adaptively learn not only the model categories, but also the scale at which they are learned over time, we have developed an enhancement to our learning approach that applies the concept of *scale space* to our learning algorithm. This is a familiar concept in the field of computer vision, in which (Gaussian or Laplacian) image pyramids are used to efficiently represent and analyze images at multiple scales of image resolution (Burt & Adelson, 1983). In our multi-scale learning enhancement, multiple models are learned simultaneously as different model layers, with each successive layer having a scale parameter that results in a coarser scale model being learned than the model in the previous layer. This is an efficient learning representation because, while multiple model layers are learned, the coarse-resolution model layers use larger and fewer categories than the fine-resolution model layers (see Figure 4). Although multiple model layers are learned simultaneously, only one of the model layers is “active” at any given time for the purpose of detecting deviations and alerting. As learning proceeds the average category evidence in each layer is monitored, and this value is used as the criteria for switching between model layers.

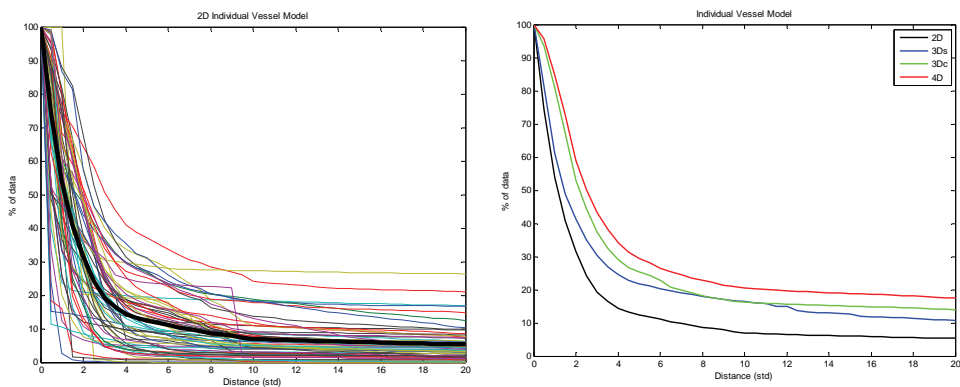


Fig. 3. Proportion of track reports beyond Mahalanobis distance from ellipse centroids as a function of ellipse standard deviations. *Left*: Individual vessel model functions; thick black line is the average function over all vessels. *Right*: Average functions for differing model dimensionality.

#### 4. Inter-event normalcy learning and anomaly detection – behavior prediction

Learning for behavior prediction aims to predict the future position of a vessel given its current behavior (location and velocity). Essentially, this involves learning links between behavioral events. It is important that the prediction learning system operates

autonomously so as to not make demands on already busy operators. Also essential is that learning occurs incrementally in order to allow the system to take advantage of increasing amounts of data without having to take the system offline in order to batch process massive amounts of data. An additional benefit of this incremental approach is that the system will be able to adapt to changing behavior patterns automatically. For these reasons, our learning approach for this task is based on the associative learning algorithm introduced in Rhodes (2007) and extended in Bomberger *et al.* (2006), Rhodes *et al.* (2007a), and Zandipour *et al.* (2008). Weights between grid locations change via presynaptically gated Hebbian learning. The set of weights in which learning takes place is determined by the velocity state of the vessel at the start of each temporal prediction window. Learning is based on the associative learning algorithm, as described in Rhodes *et al.* (2007a):

$$\Delta w_{ijk} = \frac{1}{N_{jk}} \cdot x_{jk} \cdot (x_{ik} - w_{ijk}) \quad (1)$$

where  $N_{jk}$  is the number of times that node  $j$  has been activated in the  $k^{\text{th}}$  set of weights (which corresponds to the vessel velocity state at the beginning of the prediction interval, indexed by  $k$ ),  $w_{ijk}$  is the connection weight from node  $j$  to node  $i$ , and  $x_{jk}$  and  $x_{ik}$  are the activations of grid locations  $j$  (location at the start of the period—the source location) and  $i$  (location at the end of the period—the target location) respectively. Note that the learning rate is node-dependent, such that it decreases with the amount of activity that has been encountered by a node. For a node  $j$  in the  $k^{\text{th}}$  set of weights, the learning rate first starts at a maximum of 1 and then decreases inversely with  $N_{jk}$ . Each node thus begins in a fast-learning mode, and then the weights are slowly tuned as more data is presented. Learning is presynaptically-gated by activation at the source location. If this location is not active, then no connections from this location to other locations will change their weights. If the source location is active, then links with active target locations will increase their weights and links with inactive target locations will decrease their weights. Given the binary activations used in the network, weights are bounded between 0 and 1 and the size of weight changes is governed by the learning rate and the size of the current weight. This data-dependent learning rate causes the learned weights to accurately track the conditional probabilities encountered in the training data. In contrast to neural network approaches that use batch learning to minimize a global error function with a limited set of hidden weights, this associative learning approach is both incremental and local, and each weight can be physically interpreted as part of a probability density function. The incremental and local nature of the learning process causes the model to adapt as new data is received and is less prone to convergence to local extrema since there is no global error function to be optimized. This form of learning has a number of attractive properties for the current application. First, more frequent combinations of source and target locations are rapidly learned, as indicated by larger weights. Second, random/infrequent combinations will cause learning when they occur but will also be unlearned through weight decay when they do not occur. This property also provides noise tolerance. Third, the system is able to automatically track changes in behavior over time. Fourth, the system is also able to maintain multiple sets of models for alternating operating conditions, for example, to capture seasonal differences or other factors. Fifth, the learning is entirely unsupervised, and requires no operator intervention.

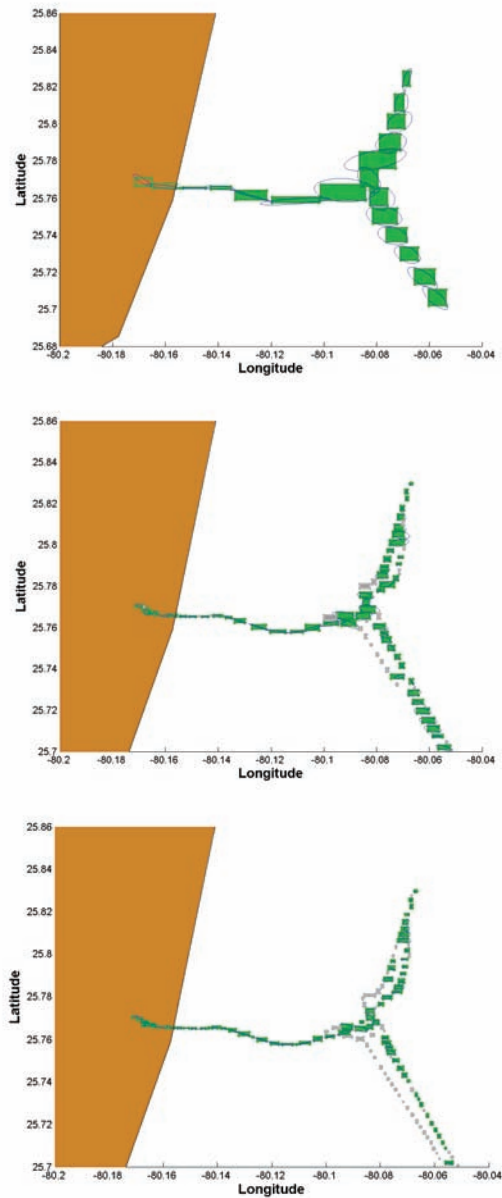


Fig. 4. As more data are received, the normalcy models fill in at the various spatial scales. Those with coarser resolution 'mature' earlier (top), but gradually those with finer resolution develop sufficiently to be used (middle, then bottom). Data-driven utilization of finer resolution models serves to maintain detection sensitivity (while enabling rapid initial use of less precise models).

#### 4.1 Example results

From the same recorded AIS dataset used in Section 3, we utilized vessel location (latitude and longitude) and velocity (course and speed) as the basis for demonstrating our mechanism for predicting future vessel behavior. We placed a square grid over the area of interest surrounding the port of Miami so as to discretize vessel location (see Figure 5). We also defined a discretization of vessel velocity that enables learning to be contextually specific to the behavior of the vessel. Thus for each vessel report, we were able to place the vessel in a grid location having a velocity state. For purposes of exposition, the chosen temporal prediction horizon is 15 minutes.

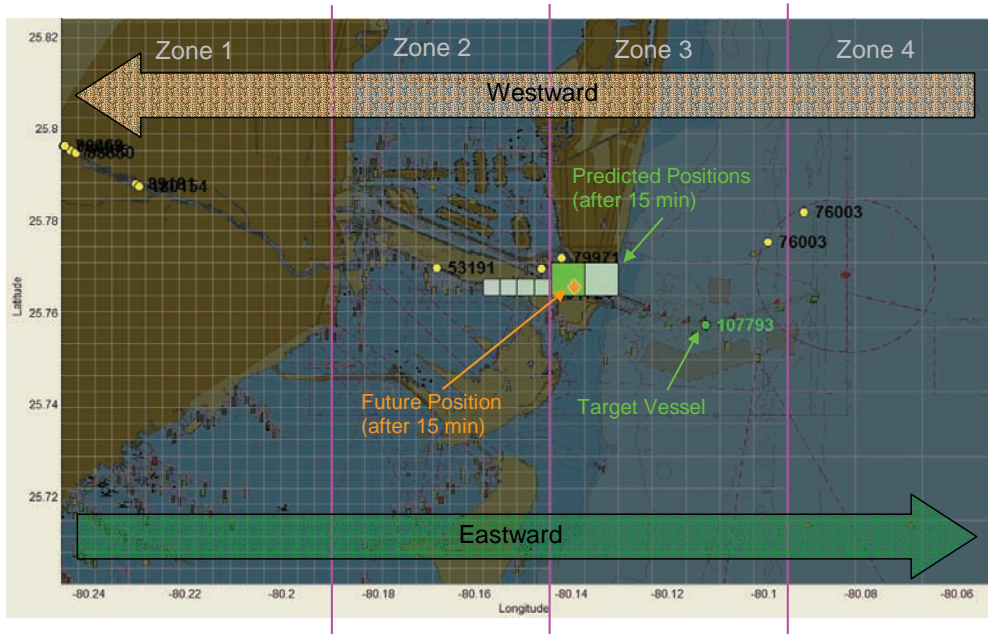


Fig. 5. Snapshot from Miami Harbor surrounds depicting system operation. The location multi-scale grid is superimposed over an ENC map of the area. Current vessel location is indicated on the map by circular markers and identification numbers. One vessel (ID 107793) has been selected for prediction display (as indicated by the larger, brighter marker). The actual future position of this vessel at the end of a 15 minute prediction horizon is indicated by the diamond. Model predictions of future location are indicated by highlighted grid locations. The strength of the weight underlying each prediction is indicated by the highlight intensity (pale=small weights; dark=large weights). Since the actual future location falls within a predicted grid location, this example represents a hit. The map is overlaid with zones that we have imposed for analysis of prediction results. Grids in zone 4 are four times larger than grids in zone 3, and 16 times larger than grids in zones 1 and 2.

To determine performance, we compared the set of grid locations (and corresponding weights) predicted by the model based on the current location and velocity state of each vessel to each vessel's corresponding actual location 15 minutes into the future.

Each location prediction consists of a set of grid locations (the target states) and the corresponding model weights from the grid location determined by the known location and velocity of the vessel (the source state). The set of weights from the source state to the target states forms a probability density function, where the weight to each target state represents the conditional probability that it will occur in 15 minutes given that the source state has occurred.

*Recall*, *precision*, *accuracy*, and *coverage* statistics were calculated periodically. *Coverage* provides a measure of how well the learning has progressed in terms of being able to make predictions for all events presented to the model. *Recall* and *precision* are standard information retrieval metrics for assessing model performance. *Recall* is equivalent to  $P_D$  (probability of correct detection) and is an absolute measure of prediction accuracy. *Precision* is related to  $P_{FA}$  (probability of false alarms), which decreases as *recall* increases. *Accuracy*—as defined here—is a relative measure of prediction accuracy in that it measures the probability of correct prediction made. In contrast, *recall* factors in all events irrespective of whether a prediction was made or not. Rhodes et al. (2007a) showed that *recall* is the most relevant metric for evaluation of prediction performance. In order to generate a prediction at a requested recall level, a subset of the predicted grid locations is selected by adding predicted locations (in order from highest to lowest weight) until the sum of the weights exceeds the requested recall level.

Due to fast learning (to a weight of 1) at a node when it is first activated, *coverage* less than 1 indicates that some of the vessel states for which predictions are to be made have never been previously encountered. *Accuracy* differs from *recall* only to the extent that vessel states for which predictions are to be made have not been encountered before. The important measure is whether the predicted grid locations contain the actual future vessel location with the same probability as the recall level that is requested. That is, does the *actual recall* match the *requested recall threshold* ( $T_R$ )? Ideally, *actual recall* should always match *requested recall*. Therefore, the plot of *actual recall* vs. *requested recall threshold* should ideally produce a straight line with slope of 1 (and 0 intercept).

The *recall* vs. *requested recall threshold* ( $T_R$ ) is plotted in Figure 6 for all zones and speed states, along with the *coverage*, *accuracy*, and *precision*. The solid black line (slope = 1) illustrates the *recall* performance of an ideal predictor for reference, for which the *actual recall* matches the *requested recall* level. As described earlier, *coverage* is less than 1 when vessel states for which predictions are to be made have not been encountered before, and thus is constant as  $T_R$  increases. *Accuracy* differs from *recall* only when *coverage* is less than 1. *Precision* decreases with increased  $T_R$ , having a shallow slope. The most important quantity from Figure 6 is how well *recall* matches the *requested recall* level. If the match is good, the predictions are accurate with respect to the uncertainty in the underlying data distribution, so lower precision can be tolerated.

## 5. Discussion

The neuro-cognitively inspired learning algorithms and representational paradigms described here have been remarkably successful in a variety of application domains. We have previously reported their use in a prototype program for port and littoral zone surveillance and automated scene understanding (Rhodes et al. 2006, 2007b). We also have

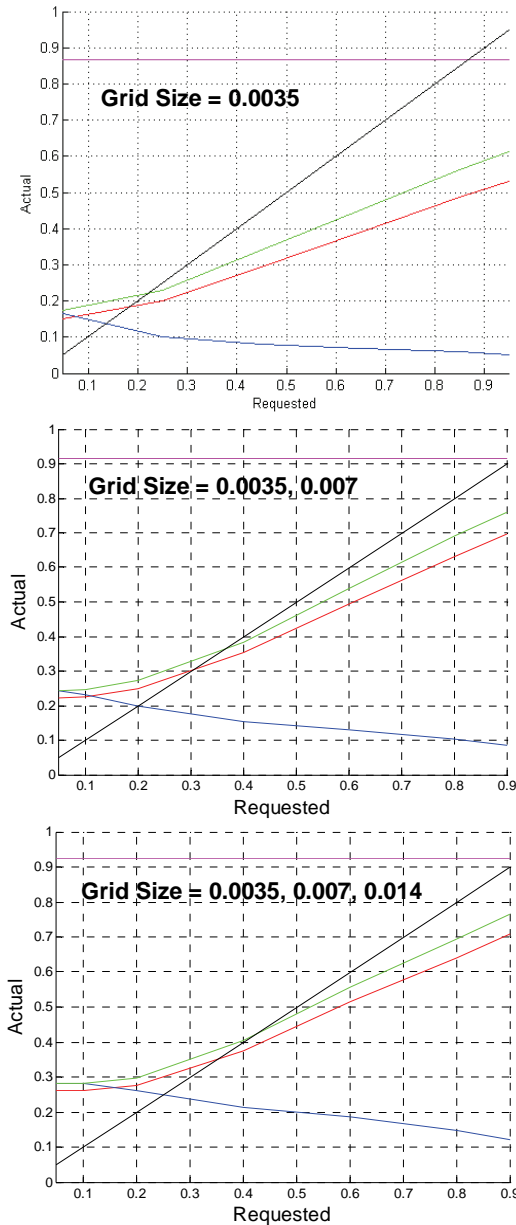


Fig. 6. Prediction results compiled across all zones. The top panel is based on a uniform grid; the middle and bottom panels are based on 2-scale and 3-scale grids respectively. The multi-scale grids had significantly better results on the metrics. *Recall* (red), *coverage* (magenta), *accuracy* (green) and *precision* (blue) are plotted vs. requested recall. The solid black line (slope=1) illustrates the recall performance of an ideal predictor for reference.



unreported success in other maritime domain awareness applications as well as land-based applications. The latter have been based on data from platforms such as surveillance towers and UAVs.

Learning-based track data analysis and exploitation as a surveillance and monitoring capability is an emerging new capability that becomes increasingly important as constraints on personnel clash with increasing needs for vigilant watchkeeping. These capabilities contribute to higher-level fusion situational awareness and assessment objectives. They also provide essential elements for automated scene understanding to shift operator focus from sensor monitoring and activity detection to assessment and response.

While having performed well in a variety of prototype level situations, our current effort represents first-generation technology. It is not yet mature enough for operational use. Each new application area produces new insight into the strengths and weaknesses of the algorithms and how they should be embedded into an overall system. Studying performance characteristics under a variety of circumstances enables the robustness and generality of the algorithmic components to be identified and enhanced. This also permits incorporation of situation specific functionality as needed to meet specific operational requirements. It is also often the case that insights gained from a new domain yield solutions that are beneficial across numerous domains.

## 6. Future research

Although the approaches described here have met with considerable success in a variety of domain applications, much remains to be done to produce a truly effective capability. For example, we have begun to move beyond the kinematic trajectory domain to address abnormality detection problems in other fields. Once we have multi-domain normalcy learning capabilities, it will be important to fuse across those domains in order to enhance anomaly detection. Consider, for example, a potential situation where a given activity pattern is considered normal in each of two domains judged independently, but determined to be deviant when the domains are jointly judged.

Other lines of pursuit include enhancing the flexibility of the contextually-sensitive aspect of our learning approach and refining the reinforcement learning approach used to incorporate operator feedback. In the former case, our current approach treats contexts in a discrete manner, proscribing capabilities such as mixing contexts to determine normalcy of current activity patterns or interpolating between contexts to account for previously unseen combinations of contextual conditions. As for reinforcement learning, enhancing the model refinement utility offered to operators is the key objective. Model fidelity and integrity need to be maintained while enabling user-specific insights and expertise to be incorporated via simple, intuitive interactions with the system. Moreover, potentially divergent interests of different users have to be accommodated in any tool in order for it to be useful in situations where multiple operators will be interacting with it.

## 7. Acknowledgements

The material presented here is based upon work supported by the AFOSR under Contract No. FA9550-06-C-0018.

## 8. References

- Alon, J., Sclaroff, S., Kollios, G., & Pavlovic, V. (2003). Discovering clusters in motion time-series data, *Proceedings of IEEE Computer Vision & Pattern Recognition*, Madison, WI, USA, June, 2003.
- Bomberger, N.A., Rhodes, B.J., Seibert, M., & Waxman, A.M. (2006). Associative learning of vessel motion patterns for maritime situation awareness, *Proceedings of the 9<sup>th</sup> International Conference on Information Fusion*, Florence, Italy, July, 2006.
- Burt, P.J. & Adelson, E.H. (1983). The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, COM-31, 532-540.
- Endsley, M. (1995). Towards a theory of situation awareness in dynamic systems. *Human Factors*, 37(1), 32-64.
- Hu, W., Tan, T., Wang, L., & Maybank, S. (2004a). A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, 34(3), 334-352.
- Hu, W., Xie, D., Tan, T., & Maybank, S. (2004b). Learning activity patterns using fuzzy self-organizing neural networks. *IEEE Transactions on Systems, Man & Cybernetics – Part B: Cybernetics*, 34(3), 1618-1626.
- Johnson, N. & Hogg, D. (1996). Learning the distribution of object trajectories for event recognition. *Image Vis. Computation*, 14(8), 609-615.
- Liao, T.W. (2005). Clustering of time series data – a survey. *Pattern Recognition*, 38(11), 1857-1874.
- Makris, D. & Ellis, T. (2005). Learning semantic scene models from observing activity in visual surveillance. *IEEE Transactions on Systems Man & Cybernetics – Part B: Cybernetics*, 35(3), 397-408.
- Owens, J. & Hunter, A. (2000). Application of the self-organising map to trajectory classification. *Proceedings of IEEE International Workshop Visual Surveillance*, 77-83.
- Piciarelli, C. & Foresti, G.L. (2006). On-line trajectory clustering for anomalous events detection. *Pattern Recognition Letters*, 27, 1835-1842.
- Rhodes, B.J. (2007). Taxonomic knowledge structure discovery from imagery-based data using the Neural Associative Incremental Learning (NAIL) algorithm. *Information Fusion*, 8, 295-315.
- Rhodes, B.J., Bomberger, N.A., & Zandipour, M. (2007a). Probabilistic associative learning of vessel motion patterns at multiple spatial scales for maritime situation awareness, *Proceedings of the 10<sup>th</sup> International Conference on Information Fusion*, Quebec City, Canada, July, 2007.
- Rhodes, B.J., Bomberger, N.A., Klamer, D., L'Italien, A.C., Stauffer, C., Creamer, W., Santos, B.C., & Seibert, M. (2007b). SeeCoast: Bringing automated scene understanding to US Coast Guard port surveillance. *Journal of Underwater Acoustics*.
- Rhodes, B.J., Bomberger, N.A., Seibert, M., & Waxman, A.M. (2006). SeeCoast: Automated port scene understanding facilitated by normalcy learning. In *Proceedings of IEEE MILCOM 2006 Military Communications Conference*, Washington, DC, USA, October, 2006.

Zandipour, M., Rhodes, B.J., & Bomberger, N.A. (2008). Probabilistic prediction of vessel motion at multiple spatial scales for maritime situation awareness, *Proceedings of the 11<sup>th</sup> International Conference on Information Fusion*, Cologne, Germany, June-July, 2008.

# A Biologically Based Framework for Distributed Sensory Fusion and Data Processing

Ferro M. and Pioggia G.  
*Interdepartmental Research Center "E. Piaggio"*  
*Faculty of Engineering, University of Pisa,*  
*Italy*

## 1. Introduction

The increasing complexity of the artificial implementations of biological systems poses issues in sensory feature extraction and fusion, drift compensation and pattern recognition, especially when high reliability is required [1, 2, 3, 4]. In particular, in order to achieve effective results, the pattern recognition system must be carefully designed. At present, these instruments often fail to give the expected results and research is under development. This happens for a series of concomitant causes, ranging from the measurements, to the limits relevant to instability and non-reproducibility of most existing sensors, up to the inappropriate use of the pattern recognition scheme, i.e. the perception of an odour/taste and its classification through the comparison with similar stimuli perceived in the past. Many techniques are used for this purpose, but recently, the processing architectures are often performed by models inspired by biology, such as genetic algorithms and Artificial Neural Networks (ANN) [4, 5, 6]. Enhancing the reliability of high-level processing systems represents the next critical step. Such architectures require high-efficiency interconnection and co-operation of several heterogeneous modules, i.e. control, data acquisition, data filtering, feature selection and pattern analysis. Heterogeneous techniques derived from chemometrics, neural networks, fuzzy-rules used to implement such tasks may introduce module interconnection and cooperation issues [7, 8]. It may not be reliable to establish a multi-channel communication among common artificial neural networks tools, feature extraction and selection processes, and acquisition and control systems. Moreover, high level interfaces often do not allow adapting of the architecture and/or the processes topology at run-time. As a result complex processing methods have to be designed. A real-time approach for data analysis requires the realization of interconnected modules which are capable to establish an efficient communication channel. In this way the application should be able to control all modules of the elaboration chain, including analysis protocol management and sensory and actuating interfaces.

The body is felt as a unity, with different qualities at different times and the brain mechanism that underlies the experience also comprises a unified system that acts as a whole and produces a neurosignature pattern of a whole body [9]. A distributed processing throughout many areas of the brain, comprises a widespread network of neurons that generates patterns, processes information that flows through it, and ultimately produces the

pattern that is felt as a whole body possessing a sense of self. The stream of neurosignature output with constantly varying patterns riding on the main signature pattern produces the feelings of the body-self with constantly changing perceptual and emotional qualities. The new breakthroughs made in the past few decades in material science in order to develop intelligent sensing materials built in compliance, non-linearity and softness allow to mimic the multi-component and bi-phasic nature of biological matter. Moreover, intelligent algorithms allow the transduction signals to be effectively reconstructed. In order to provide an artificial neural network architecture with the capabilities of processing, coding and fusing in real-time the distributed information continuously flowing from an artificial distributed sensing network, in this chapter a mammalian cortex inspired model is described.

According to the biological sensory systems, where environmental stimuli are deconstructed and then reconstructed in the brain to create perceptions [10], the presented architecture may help dealing with a dynamic and efficient management of multi-transducer data processing techniques, as well as serving as an initial step in the reconstruction of a fused image from its deconstructed features. The raw signals obtained from artificial implementations of biological systems can be preprocessed in order to extract relevant features. Features vectors constitute the dataset for sensory fusion and the pattern recognition processes. Fusion and processing are achieved by the homogeneous software framework where, in order to gain short-term priming in co-operation with other modules, artificial neural models and architectures inspired to the mammalian cortex [11, 12, 13, 14] are implemented. Artificial neuron models with high computational efficiency and biological accuracy are adopted to obtain a learning strategy able to avoid catastrophic interference [15, 16, 17] and to enable a selection of neuronal groups [18]. To take into account this theory the time variable in the learning task is used, so that neural groups may raise from a selection process.

The framework is able to manage at the same time transducer devices and data processing. Synchronization among modules and data flow is managed by the framework offering remarkable advantages in simulation of heterogeneous complex dynamic processes. Specific control processes, pattern recognition algorithms, sensory and actuating interfaces may be created inheriting from the framework base structures. The architecture is library-oriented rather than application-oriented and starting from the base models available in the framework core dedicated models for processes, maps and connections may be derived. Such a strategy permits the realisation of a user-defined environment able to automate the elaboration of cooperating processes. Heterogeneous processes will be able to communicate each other inside the framework as specified by the user. The framework architecture has been designed as a hierarchical structure whose root is a manager module. It is realised as a high-level container of generic modules and it represents the environment in which process modules and I/O filtering interfaces are placed. Modules may be grouped recursively in order to share common properties and functionalities of entity modules belonging to the same type. Communication channels are realised as connections through specific projection types that specify the connection topology. Connections are delegated to dispatch synchronization information and user-defined data. The filtering interface modules are able to drive the transducer hardware and to dispatch information to process modules. All base modules manage dynamic structures and are designed to maintain data consistency while the environment state may change. High level processes such as control processes and

pattern recognition algorithms are defined as application processes inside the framework. Such processes inherit properties and functionalities from the framework base structures, taking advantage of automation capabilities provided by the framework core. The framework allows to create a communication language between the framework core and the hardware architecture. This guarantees an increased flexibility thanks to the presence of interfaces performing the function of interpreters for the specific hardware and filters which specify the way the framework core senses and communicates the information.

## 2. A framework solution for high complex tasks

From a general point of view, a complex system consists of different modules cooperating in order to perform data acquisition from multiple sensors, data analysis through several techniques and data redirection to the actuator systems. The architecture here proposed addresses three main issues:

- acquisition from sensors: a protocol interface will be available to dispatch data coming from input systems; for each hardware sensory system the user will realize the software driver to filter the signals and to dispatch data to the framework core via the framework I/O interface.
- data processing: inside the framework core all the processes will be specified by the user; for each process the user will specify the algorithms, the connection topology between other processes and, optionally, the geometrical structure.
- actuator driving: a protocol interface will be available to dispatch data from the framework core; for each hardware actuating system the user will realize the software driver to filter the data and to dispatch the signals from the framework I/O interface to the actuating systems.

The design of a versatile instrument for data management and elaboration should be suitable for those systems which are equipped with distributed transducer devices, where a particular attention should be paid to inter-process communication. Applications of such instrument space from the simple elaboration of signals supplied by sensory and actuating networks, to pattern analysis and recognition techniques. Design specification included the ability to let the system to be able to operate in real-time. The realization of a framework able to perform a parallel device management should give to all the modules the ability to cooperate and the possibility to share data coming from sensory systems and directed to the actuating systems. The possibility to operate in real time imposes critical efficiency requirements to each single module. The framework design pays attention to the management and the synchronization of data and processes. Control modules and pattern recognition algorithms are defined as application processes inside the framework. The framework is realised as a software library in order to exploit the potentials of the computational algorithms and to enhance the performances of the processing techniques based on artificial neural networks. The architecture is able to manage at the same time transducer devices and data processing. Synchronization among modules and data flow is managed by the framework offering remarkable advantages in simulation of heterogeneous complex dynamic processes. Specific control processes, pattern recognition algorithms, sensory and actuating interfaces may be created inheriting from the framework base structures.

In order to exploit the potentials of the computational algorithms and to enhance the performances of control processing techniques, the framework is realised as a C++ software

library. In this way the library architecture is a re-programmable instrument available to the user to develop specific applications. It has been designed to be portable to any software platform in order to gain abstraction from the operating system. The framework however needs a low-level software layer to perform kernel re-building and low-level system calls. An Intel-based personal computer is actually being used and a commercial operative system grants the low-level communication.

### **2.1 I/O device communication**

The framework core and application processes are interfaced with the outside world through the framework I/O interface. This layer has been developed in order to act as a buffer for the flow of information coming in from the sensors and out to the actuators. With this strategy sensory fusion is gained enabling an abstraction with respect to the specific technology of the transducers used. Signals coming from the sensors are gathered in parallel and are encoded according to a standard protocol. The encoded information is received by a specific filter for each sensor, which then sorts them to framework I/O interface. For each actuating system a mirror image architecture has been reproduced with respect to the one described for the sensors. The information available in the framework I/O interface is encoded by a filter using the same standard protocol. A specific interface for each actuator pilots its specific hardware system. This architecture allows setting up a communication language between the framework core and sensory and actuating devices. This guarantees an increased flexibility thanks to the presence of interfaces performing the function of interpreters for the specific hardware and filters which specify the way the framework core senses and communicates the information. Fig. 2.1 shows the flow of information to and from the framework core. Communication channels are established as connections between application processes so that framework is able to perform a low-level inter-process communication. The domain of data flowing through connections and the flow chart of the application processes are user-defined.

### **2.2 Parallel distributed processing**

Synchronization among modules and data flow is managed by the framework offering remarkable advantages in simulation of heterogeneous complex dynamic processes. Specific control processes, pattern recognition algorithms, sensory and actuating interfaces may be created inheriting from the framework base structures, taking advantage of process automation provided by the framework core. A spatial definition of the entities involved in the framework can be supplied, making this information available to the control system for subsequent processing. To guarantee the execution of real-time applications an inner synchronization signal is provided from the framework core to the processes and to the framework I/O interface, enabling to gain time-space correlation. A dynamic geometrical representation can be visualised by a high efficiency 3D graphic interface, giving a support during experimental setup debug. Processes and connections are managed at run time and they can be manipulated under request. The presence of dynamic structures implies a configurable resource management, so the framework offers an optimised interface for enumeration and direct access requests.

### **2.3 Control and processing modules**

The framework architecture has been designed as a hierarchical structure whose root is a manager module (*3DWorld* ). It is realised as an high-level container of generic modules

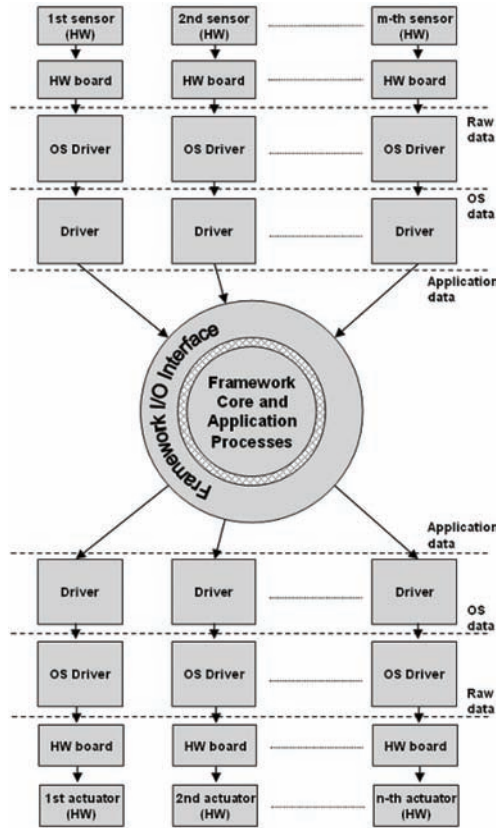


Fig. 2.1. Architecture of the framework for the parallel management of multiple elaboration processes. The transducer devices are synchronized and controlled through an appropriate I/O interface

representing the environment in which process modules ( $W$ ) and I/O filtering interfaces are placed ( $Drivers$ ). All these modules inherit low-level properties and functionalities from a base module ( $3DObject$ ) realised as an element able to populate the process environment. Virtual and pure-virtual functionality strategies have been applied to this base module in order to obtain an abstraction with respect to the generic application task. In this way the core is able to process user-defined functionalities without being reprogrammed. Moreover, modules may be grouped recursively ( $WGroup$ ) in order to share common properties and functionalities of entity modules belonging to the same type. Communication channels are realised as connections ( $WConnection$ ) through specific projection types that specify the connection topology. Connections are delegated to dispatch synchronization information and user-defined data ( $WConnectionSpec$ ). The filtering interface modules are able to drive the transducer hardware and to dispatch information to process modules. All base modules manage dynamic structures and they are designed to maintain data consistency while the environment state may change. This behaviour permits the execution of dynamic and real-time parallel distributed processing while synchronization and data flowing are managed



by the environment. All modules are realized as running processes while their control and synchronization is managed by the framework. The hierarchical and collaboration chart of the base structures is shown in Fig. 2.2.

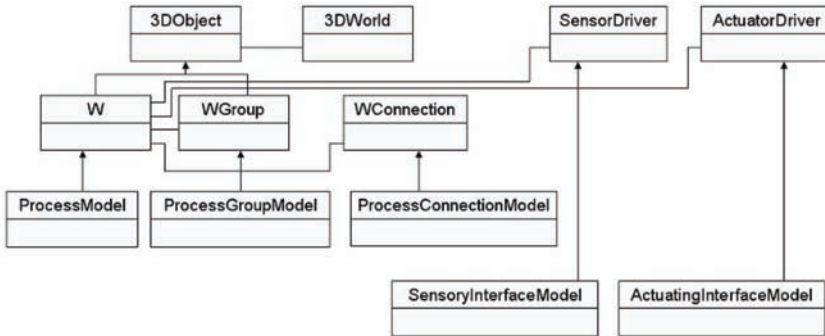


Fig. 2.2. Inheritance and collaboration diagram of the main modules of the framework core: I/O interfaces, communication channels, processes

### 3. Framework overview

Architecture implementation details will be showed in this section making use of the Unified Modelling Language (UML) representation. In Fig. 3.1 the legend is showed.

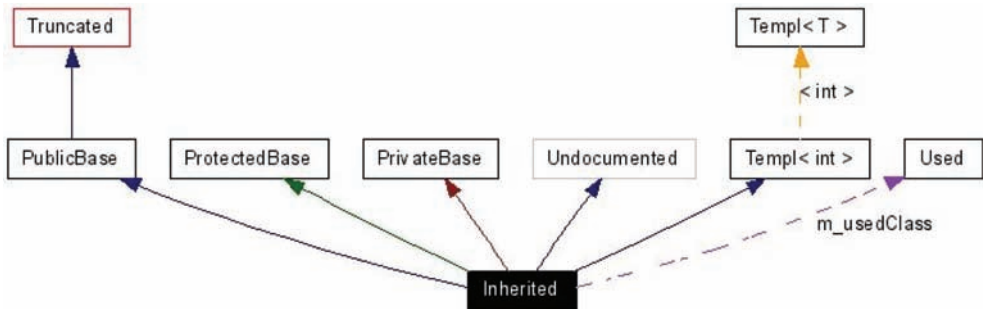


Fig. 3.1. Graph relationship legend

The boxes in the above graph have the following meaning:

- A filled black box represents the struct or class for which the graph is generated.
- A box with a black border denotes a documented struct or class.
- A box with a grey border denotes an undocumented struct or class.
- A box with a red border denotes a documented struct or class for which not all inheritance/containment relations are shown. A graph is truncated if it does not fit within the specified boundaries.

The arrows have the following meaning:

- A dark blue arrow is used to visualize a public inheritance relation between two classes.
- A dark green arrow is used for protected inheritance.
- A dark red arrow is used for private inheritance.

- A purple dashed arrow is used if a class is contained or used by another class. The arrow is labeled with the variable(s) through which the pointed class or struct is accessible.
- A yellow dashed arrow denotes a relation between a template instance and the template class it was instantiated from. The arrow is labeled with the template parameters of the instance.

The implementation will be described using technical object-oriented language terms. To avoid confusion main terms are resumed:

- an *object* is an synonymous word for a *class* or a *structure* and it defines a new data type.
- an *instance* of an object is a variable declared as object type.
- an object's property is called *member*.
- an object's function is called *method*.
- a *derived* object is a child of another object, and it inherits properties and functionalities from his father object, which is called *base* object.
- a *private* member or method is accessible only inside the object.
- a *protected* member or method is accessible both from the object and from a derived object.
- a *public* member or method is always accessible.
- while structures start implicitly with public definitions, classes start implicitly with private definitions.
- a *virtual* method defines a function that, if it is overridden in a derived object, cause the call to derived method even it is called on the base object.
- a *pure-virtual* method is undefined in the base object and, as a result, the base object can not be instantiated.
- an *abstract* object contains only pure-virtual methods.

### 3.1 Portability

Since the framework library is written using the C++ programming language, the software portability is guaranteed by the standard ANSI-C/C++ definitions. However the low-level interfaces depend on the particular libraries of the operative system. For this reason a low-level layer was defined to include all the dependencies for the specific platform. The low-level layer (*ARI\_Macro*) has been realised defining a set of operations for run-time memory management (allocation and deallocation), file I/O interface, log reports and window assert dialogs. All these operations are implemented as C++ macro and they will be used by the framework for all the low-level operative system interfacing.

Graphic User Interface (GUI) is not embedded in the framework in order to not slow down the application efficiency and to let the user to be able to choose his preferred tools. Since each of the framework objects provides functions to get information about the status and the output data, the GUI tools are developed as external modules that can be linked to the architecture. Main graphic output is guaranteed by OpenGL rendering, which libraries are available for many hardware platforms. If the user choose to use OpenGL support, then he must link OpenGL libraries to his application. Application GUI is actually supplied for Microsoft Windows operating system, including useful tools for layered graphs (*MGraph*) and OpenGL dialog windows (*glCDialog*). While data storage is already supplied by the framework, additional end-user tools are available for file tables (*TabData*) and database tables (*TabDataConn*) supporting MySQL, SQLServer and general ODBC drivers. Since all these tools are external they not compromise the architecture efficiency, making the user able to choose his appropriate strategy.



### 3.3 The base structure: 3DObject

The basic element of the framework is called *3DObject*. This is an element able to perform basic functions which are useful to many of the modules included in the framework.

*3DObject* properties include a label, a 3D position, a 3D radius and a set of user-defined flags. The label will be useful for log reports, application debug and for easy high-level object identification. The default label will be associated with the hexadecimal memory address of the object, which is unique during the execution of the application. Three-dimensional position and radius will be useful for graphic rendering under user request. Such values are initialised with null values, since the rendering functions are optional. Flags specify the way in which the object manager will process the object (see *3DWorld* ). By default no flag is set for a *3DObject*.

Several functions has been supplied for this base object in order to guarantee I/O transfers, 3D management and flags maintenance. Since process objects will be derived from *3DObject*, a set of virtual functions has been defined to perform process synchronization and update. The use of virtual functions guarantees the ability of the object manager to call redefined functions on derived objects without knowing them. User-defined processes have to follow the base virtual protocol, redefining the way in which synchronization and update operations are performed. In Fig. 3.3 the *3DObject* architecture is showed.

```

ARI_3DObject
#_position
#_radius
#_flags
#_name

+ARI_3DObject()
+Render()
+SetInput()
+Update()
+GetFlags()
+SetFlags()
+IsSetFlag()
+AddFlag()
+RemFlag()
+SetName()
+GetName()
+SetRadius()
+SetSphereRadius()
+GetRadius()
+SetPosition()
+GetPosition()
+FromFile()
+ToFile()
+FromBuffer()
+ToBuffer()
+GetBufferLen()

```

Fig. 3.3. Architecture of the *3DObject* structure. This is the base object used to populate the process environment managed by the *3DWorld* structure

Defined virtual functions are *Render*, *SetInput* and *Update*. Each of these functions will be called by the object manager only if the appropriate flag is set. Such strategy makes the user

to be able to specify virtual overrides for his specific processes. *Render* function will specify the way in which the object will be graphically rendered. OpenGL is the default renderer, permitting an high efficient three-dimensional output. Graphic rendering is useful for object status monitoring during application execution and debug. The definition of a rendering function does not reduce object efficiency since *Render* virtual function will be called only if screen repainting is needed and if the rendering flag has been set.

*SetInput* and *Update* functions specify the core process algorithms. The first method is needed to workaround the serial processing of all the objects. In fact the parallel execution of all the process objects will be serialized by the low-level CPU scheduling. The order of the execution of each process may compromise the effective result of the process network in the case of multiple cooperating processes. For this reason the object manager will first call the *SetInput* method over all the processes with the intent of buffering the actual output data over communication channels (see *WConnection*). After this buffering step, the *Update* method will be called over all the objects making use of the buffered connection data instead of the object actual output data. Such a strategy guarantees the independence from the process execution order. Default base method just defines the virtual processing and rendering protocol.

The same virtual strategy has been adopted for buffers and files data storage. A set of virtual functions has been defined to perform an object independent way for file I/O (*ToFile*, *FromFile*) using platform independent file operations. Such methods use buffer virtual functions (*ToBuffer*, *FromBuffer*, *GetBufferLen*) which will be specified by the user for each process. Default implementation of buffer virtual methods just defines the buffering I/O protocol.

### 3.4 The object manager: 3DWorld

Synchronization and update of all the running processes is managed by *3DWorld*. This structure is a collection of *3DObject*. Elements are organised in a dynamic list where the access order is often sequential. *3DWorld* provides the execution of specific methods for all the objects added during an initialisation step. Internally *3DWorld* manages the reference of each object and not the object itself. This strategy speeds up the enumeration of the elements, giving to the user the opportunity to override the manager behaviour and to gain the direct control of each single element. Elements may be added and removed at run-time (*AddObject*, *RemoveObject* ) while the execution of process virtual methods is managed accordingly to the active flags of each elements (*Render*, *SetInput*, *Update*).

### 3.5 The base process structure: W

The process base structure inherits properties and functionalities from *3DObject*. *W* is a generic transferring function, which is able to communicate with structures of the same base type. Communication channels are established as connections and the domains of input and output data are defined by the user. For this reason a pure-virtual strategy has been applied to this structure to take into account a processing method (*Process*) which is still not defined at this level. For this reason a *W* structure can not be instantiated. It only defines the process standard protocol and provides topology connection methods. *Render*, *SetInput* and *Update* flags are automatically activated in the initialization step. The architecture of *W* structure is shown in Fig. 3.4.

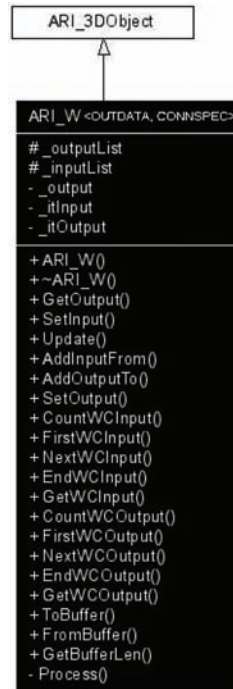


Fig. 3.4. Architecture of the *W* structure. It represents the base process model

*W* is a template structure where *OUTDATA* and *CONNSPEC* types are user-defined. While *OUTDATA* represents the output pattern, *CONNSPEC* contains the specification for each input connection. The *3DObject* virtual method *Update* is redefined to set the new output value accordingly to the result of the pure virtual method *Process*, which will be specified in derived process structures. Actual process output can be retrieved using *GetOutput* method. *W* provides methods to specify the connection topology of the single process (*AddInputFrom*, *AddOutputTo*) in respect to the other input and output processes. *SetInput* method is redefined to perform a scan over all the input connections and to transfer the actual output value of input process to the connection buffer (see *WConnection*). For sake of efficiency the starting point for both input and output connections is stored in each *W* structure. Since connections are usually browsed sequentially, enumeration methods are provided to perform high efficient navigation over input and output connections (*First*, *Next*, *End*). In derived structures the *Process* method has to be redefined to perform the core process algorithm, taking into account the specific data connection and the data output given by each input process, which are already buffered into the corresponding input connection. *Process* method is defined as private function, so it can not be directly called by the user since it is automatically managed by the framework during the update step. With this strategy the process, which core is still undefined, is able to manage the unknown information of the user-defined process.

Buffer I/O operation (*GetBufferLen*, *ToBuffer*, *FromBuffer*) are redefined to provide data storage for actual process output and for the actual specification values for all the input connections.

### 3.6 The connection between two processes: WConnection and WConnectionSpec

A connection between two process is realised by the *Wconnection* template structure. Since input connections for a given process are embedded inside the process itself, the template types are the same used for *W* structure. *WConnection* stores references of both source and destination processes. A field of *OUTDATA* type is stored to perform data buffering. In fact during the *SetInput* step on the destination process, the *GetOutput* method is called over the input process and the resulting value is stored inside the connection. While the *W* structure contains the references of both the first input connection and the first output connection, the next links to connections are stored into two dynamic lists inside the *WConnection* structure. In Fig. 3.5 an example of this strategy is shown.

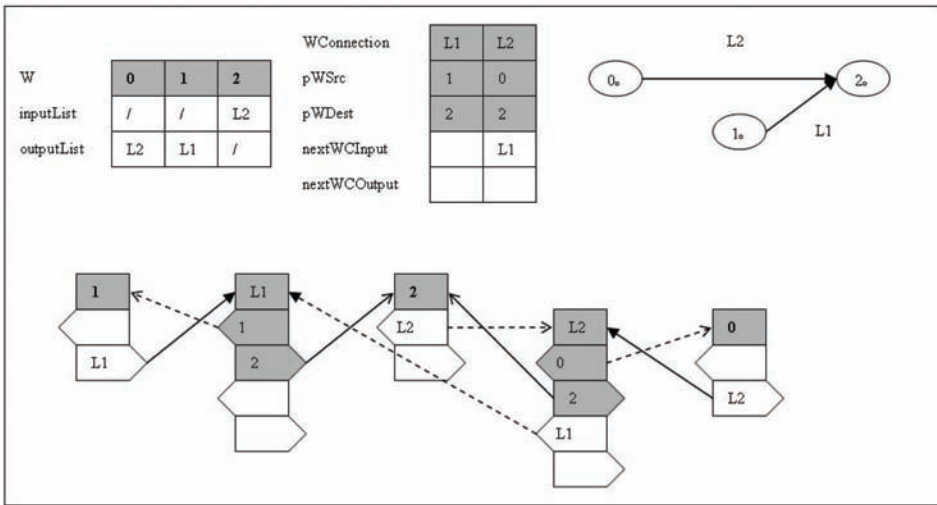


Fig. 3.5. Connection strategy example. Each process contains the references of both the first input connection and the first output connection. Each connection contains the references of both the source and the destination process, plus the references to both the next input connection and the next output connection

Inside the *WConnection* structure a field of *CONNSPEC* type is used to store the specific information needed by the destination process to evaluate the output data given by the source process. The base class for the *CONNSPEC* type is the *WConnectionSpec* structure. This structure is an abstract class where only a pure virtual method is defined (*Init*). This method will be automatically called during the creation of the connection to perform the initialisation of the *CONNSPEC* values on the basis of the user-defined parameters. This structure only defines the connection initialisation protocol and must be redefined in the derived structures.

### 3.7 Grouping processes: WGroup and WProjectionSpec

Processes may be grouped into a *WGroup* structure, which template arguments are the same of the *W* structure (*OUTDATA*, *CONNSPEC*). Since this structure only contains the references to several *W* structure, the user does not lose the control over each single process. Grouping processes results in a logical managing of several objects, which can be added and

removed at run-time (*Add*, *Remove* methods). *3DObject* is the base class for the *WGroup* structure, permitting the redefinition of the virtual functions for flag, synchronization and I/O management over all the grouped entities. *WGroup* contains its own flags, with the possibility to propagate flags to each grouped entity (*AddFlags*, *RemFlags*, *SetFlag*, *IsSetFlag*). With the same strategy adopted for the *3DWorld* structure, *WGroup* is able to propagate rendering and synchronization signals over grouped entities according to their active flags. By default these flags are activated only for the group and they are not propagated to the sub-entities. Since the *WGroup* structure has its own geometrical position and volume, a *Dispose* function is supplied for geometrically disposing all the grouped entities according to their volumes. Such information will be automatically taken into account during the framework rendering processing.

*WGroup* contains useful methods to create connections both to other groups and to other single processes. Since these methods (*AddInputFrom*, *AddOutputTo*) may involve more than one entity, connections are realised through projection specifications (*WProjectionSpec*). During the projection initialisation step, the framework looks for projection flags in order to perform the user-specific connection strategy. Currently *one-to-one*, *N-to-one*, *uniform random-to-N*, *sub-group-to-N* and *polar-random-to-N* flags are supported (see *WGroup.h* header file).

Since the entities inside a group are often browsed sequentially, high efficient iterators are defined also for this structure. I/O buffering operations are redefined to automatically join I/O buffering operations of each grouped entity. The *Wgroup* architecture is shown in Fig. 3.6.

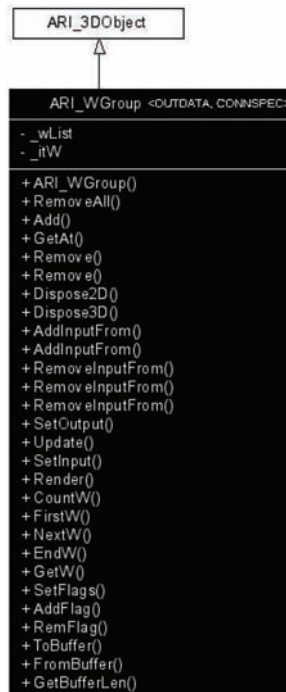


Fig. 3.6. Architecture of the *WGroup* structure. Many entities may be grouped together in order to create an high-level entity able to populate the process environment



### 3.8 SensorDriver and ActuatorDriver

The *SensorDriver* structure is the starting point for filtering input data coming from the generic input system. This structure directly derives from *3DObject* and uses *IFNeuron* and *IFNeuronGroup* structures. These two structures are specifically designed for artificial neural networks using the integrate-and-fire model, as it will be discussed later. However at this level they are used as buffering structures where the process functions (*Set-Input*, *Update*) are skipped by the framework. This choice, as it will be shown, does not cause lack of efficiency and of generality.

Since sensor information have to be available before then the framework update step, the *SensorDriver* objects have to be added into *3DWorld* by the user before than other process object. During the construction of the structure, only the *Set-Input* flag is set. *SetInput* method will be redefined for the specific hardware in derived structures, where the data filtering algorithm will be specified.

During the initialisation step (*InitDriver*) the user will specify the length of the sensor buffer data needed during the hardware acquisition step. At this point the driver just allocates the memory space in a static array, and it waits for the registration of the *IFNeuron* entities. These entities represent the objects where sensor data will be mapped. Such entities will be specified using the *Register* method, where the *SetInput* and *Update* flags are automatically removed from each registered entity. *SensorDriver* structure will be operative only when all of the needed mapping entities will be registered by the user. In order to speed up the acquisition process, the references of the *IFNeuron* objects are indexed during the registering phase. Such a strategy permits a direct memory access over all the registered entities. The user may choose to switch on/off the driver using the *SetPowerOn* method. The *SensorDriver* architecture is shown in Fig. 3.7. The *ActuatorDriver* structure follows a similar architecture.

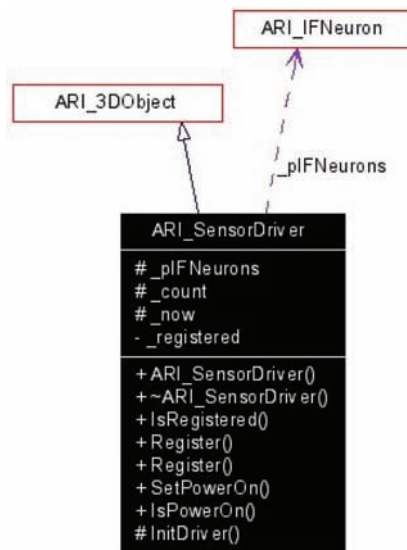


Fig. 3.7. Architecture of the SensorDriver structure. This object represents the base model for the interface of a generic input system

#### 4. Cortical-based artificial neural networks

The concept of artificial neural networks is to imitate the structure and workings of the human brain by means of mathematical models. Three basic qualities of the human brain form the foundations of most neural network models:

- knowledge is distributed over many neurons within the brain;
- neurons can communicate (locally) with one another;
- and the brain is adaptable.

The terminology with which neural networks are described is derived from these three qualities of the human brain, and is as follows:

- structure of the neuron;
- network topology;
- adaptation and learning rule.

The neurons, or processing units, which make up the neural network are single elements and consist principally of four components:

- a connection function;
- an input function;
- an activation or transfer function;
- an output function.

A neuron receives signal through several input connections. These are weighted at the input to a neuron by the connection function. The weights employed here define the coupling strength (synapses) of the respective connections and are established via a learning process, in the course of which they are modified according to given patterns and a learning rule. The input function compresses these weighted inputs into a scalar value, the so-called network activity at this neuron. Simple summation is generally employed here. In such cases, the network activity, which results from the connection function and the input function, is the weighted sum of the input values. The activation function determines a new activation status on the basis of the current network activity, if appropriate taking the previous status of the neuron into account. This new activation status is transmitted to the connecting structure of the network via the output function of the neuron, which is generally a linear function. By way of reference to biological neurons, the activation status at the output of a neuron is also known as the excitation of the neuron.

A process unit is of interest only as a unit of a network consisting primarily of homogeneous elements. In artificial neural nets, these elements are generally interconnected to form a rigid network structure, as a result of which the learning algorithm only rarely includes provision for the formation of new connections and the removal of old connections, such as occurs in biological systems. A layered connecting structure is generally employed, whereby the layer on which the input signals act is referred to as the input layer; the layer at which the results are collected is known as the output layer; and the layers located between these are known as hidden layers. The neurons are generally fully connected on a layer-by-layer basis. The number of layers often determines the performance of a network.

A distinction can be made between feedforward, lateral and feedback connections for the method of linking the different layers. Both feedforward and feedback connections over several layers are conceivable. The connecting structure and the choice of processing units determine the structure of a network. In order to carry out a data fusion and a classification, the network must be taught a task by presenting it with examples in a training phase. The

training phase normally proceeds as follows: random values are initially assigned for the weights of the neurons. Patterns from a training data record are then presented to the network and the weightings are adapted on the basis of the learning rule and training pattern until a convergence criterion, e.g. a defined error threshold, is attained. A test phase is then carried out, in which unknown test patterns are presented to the network to establish the extent to which the network has learnt the task in hand. Selection of the patterns for the training phase is a particularly important aspect. These patterns must describe the task as completely as possible, as in later use the network will only be able to provide good results for problems which it has learnt. This means that patterns must be selected which cover all classes and, where possible, describe the boundary ranges between the classes.

Most of these architectures are not able to proceed in new learning processes without losing memory of the past learning processes (catastrophic interference) [16, 17]. In order to overcome these issues models able to gain short-term priming in co-operation with other modules have been developed. In particular, hippocampus-based models operate a pattern separation avoiding the catastrophic interference [11, 12]. Input patterns are spread among different interconnected modules following the McCloskey and Cohen model [11] consisting of several interconnected two-dimensional self-organising maps of artificial neurons (Fig. 4.1). The Input Entorhinal Cortex and the Output Entorhinal Cortex maps represent respectively the input and the output of the net. Input and output maps have the same dimension in order to evaluate the activation and deactivation error by a one-to-one comparison of neuronal activity. Activation error represents the percentage of neurons that are firing in the Output Entorhinal Cortex and that are under threshold in the Input Entorhinal Cortex. Deactivation error represents the percentage of neurons that are under threshold in the Output Entorhinal Cortex and that are firing in the Input Entorhinal Cortex.

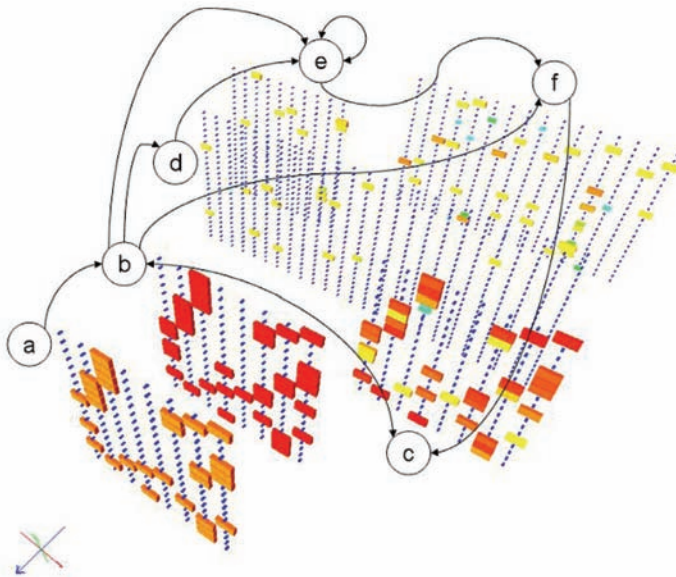


Fig. 4.1. Hippocampus model proposed by McCloskey and Cohen; a) Input pattern; b) Input Entorhinal Cortex; c) Output Entorhinal Cortex; d) Dentate Gyrus; e) CA3; f) CA1

#### 4.1 Learning strategy: selection of neuronal groups

The Theory of Neuronal Group Selection (TINGS) proposed by Edelman [18, 19], suggests a novel way for understanding and simulating neural networks. To take into account this theory we have to use the *time* variable in the learning task, so that neural groups may raise from a selection process. This strategy has been adopted by Izhikevich, who simulated a minimal neural network which is able to show the property of *polychronization* [20]. In such a network a correspondence between synaptic weights and axonal delays exists as a result of the neuron behaviour. One neuron can belong to many groups, which count is usually higher than the count of the neurons themselves. This guarantees a memory capability which is higher than the capability reached by the classical neuronal network. Such an architecture has been implemented into the framework here presented, giving the possibility to connect the neuronal groups to sensory and actuating systems. The advantage of the use of such an approach makes it possible to gain time-space correlation on input signals.

The classical approach in artificial neural networks simulation takes into account the modulation of the action potential rhythm as the only parameter for the information flowing to and from each neuron. Such a strategy seems to be in contrast with novel experimental results, since neurons are able to generate action potentials which are based on the input spike timings, with a precision up to one millisecond. The spike-timing synchrony is a natural effect that permits a neuron to be activated in correspondence of synchronous input spikes, while the neuronal activation of the post-synaptic neuron is negligible if pre-synaptic spikes arrive asynchronously to the target neuron. Axonal delays usually lie in the range [0.1, 44] milliseconds, depending on the type and location of the neuron inside the network. Such a property becomes an important feature for the selection of the neuronal groups as it is exposed by Edelman. In the artificial neural network model, the synaptic connections are modified according to the STDP rule. If a spike coming from an excitatory pre-synaptic neuron causes the firing of the post-synaptic neuron, the synaptic connection is reinforced since it gives the possibility to generate another spike in order to propagate the signal. Otherwise the synaptic connection is weakened. The values of the STDP parameters are chosen in order to permit a weakening that is greater than the reinforcement. Such a strategy permits the progressive removal of the unnecessary connections and the persistence of the connections between correlated neurons.

### 5. Implementation of artificial neurons: towards real-time data fusion and processing

The complexity of a biological neuron may be reduced by using several mathematical models. Each of these reproduces some of the functionalities of real neurons, such as the excitability in response to a specific input signal. The most accurate model for a biological neuron has been developed by Hodgkin and Huxley [13] and it is able to exactly reproduce the shape of the action potential of a neuron by taking into account the ionic currents. Besides this, the model is computationally expensive and it takes about 1200 FLOPs (Floating Point Operations) to simulate one millisecond of a single neuron activity. Several attempts have been made in order to reduce the mathematical complexity of this model. The most effective result has been obtained by the Morris-Lecar model [13], which is able to describe the oscillations of the muscular fibers of the giant squid and it is still close to the Hodgkin-Huxley model accuracy. Unfortunately the computational complexity is still high, since it takes about 600 FLOPs for one millisecond of neuron activity. Since these bottom-up

approaches are focused on the characterization of the biophysic properties of the cell membrane, a different approach has been adopted by Fitzhugh and Nagumo [13], taking into account the information of the nervous signal as a temporal distribution rather than an action potential shape. This top-down approach leads to the development of parametric differential equations with the aim to match them with experimental results. The Fitzhugh-Nagumo model, which takes about 72 FLOPS for one millisecond of neuron activity, is based on a variant of the Van Der Pol oscillator. Studies on the dynamics of non-linear systems showed a large variety of behaviours. Actually, the use of mathematical analogies seems to be the only way to simulate a large number of interconnected artificial neurons. For this reason the integrate-and-fire model (and its variant models) is the simpler and most used model for classification and prediction tasks in practical scenarios.

### 5.1 The integrate-and-fire model

The integrate-and-fire model is the simplest model of a spiking neuron that takes into account the dynamics of the input. The basis of the integrate-and-fire model is the simple compartmental model of a neuron. The equivalent electric schema is showed in Fig. 5.1.

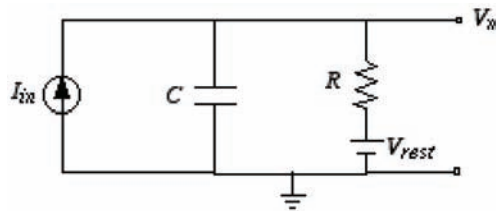


Fig. 5.1. The integrate-and-fire artificial neuron model: equivalent electric schema

The computational implementation of the integrate and fire model follows the schema showed in Fig. 5.2.

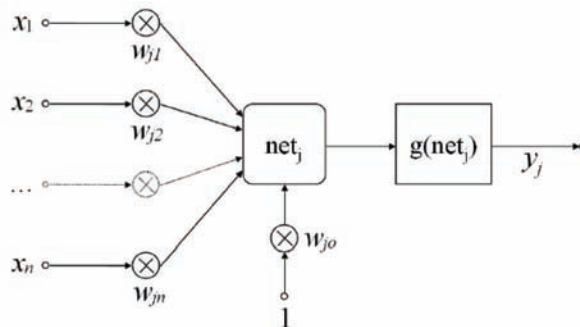


Fig. 5.2. The integrate-and-fire artificial neuron model: computational schema

An *IFNeuron* structure has been implemented in the framework as a running process directly deriving from the *W* structure. Template arguments have been specialised to obtain an *OUTDATA* as a real number (double precision floating point value) and a *CONNSPEC* as a *IFNeuronConnectionSpec* structure, which is shown in Fig. 5.3.

The connection structure for such a process uses a real number to manage the synaptic weight. The value may be initialised by the user or randomly chosen by the framework

according to the value initialisation parameters. A weight buffer value is needed for internal operations during supervised learning using the multi layer perceptron process, which will be discussed later. The *IFNeuron* structure defines the private virtual method *Process* in order to perform the weighted sum of signal coming from input connections. The result value is then filtered using the sigmoid function according to the integrate and fire model. The structure internally saves a value to speed up the delta-rule algorithm adopted during supervised learning. The I/O buffering operations simply manage internal members and recall the base class methods. The rendering function provides the graphic visualisation of the soma and of the input connections. The architecture is shown in Fig. 5.4.

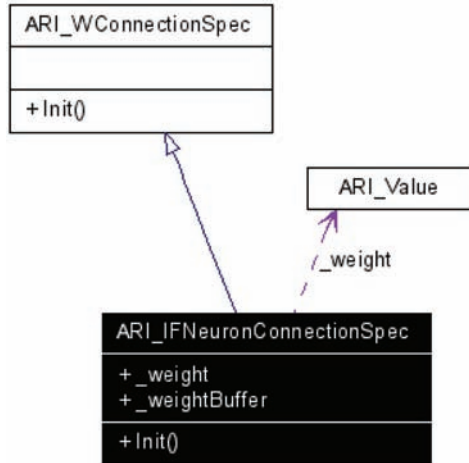


Fig. 5.3. The IFNeuronConnectionSpec structure

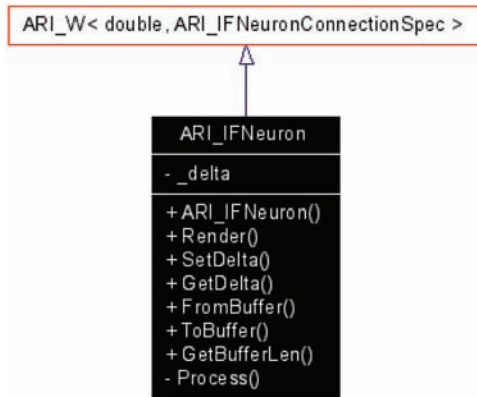


Fig. 5.4. Architecture of IFNeuron structure

The *IFNeuronGroup* structure, which represents a group of *IFNeurons*, has been derived from the *WGroup* base structure. The *IFNeuronGroup* structure will be used by high-level processes in order to perform supervised and unsupervised learning tasks based on the integrate and fire neuron model.

### 5.2 The leabra neuron model

The Leabra base model [11, 12] is a simplified version of the Hodgkin-Huxley model. Both models are shown in Table 1.

<b>Equations</b>	
$I = C_m \cdot \frac{\partial V}{\partial t} + \bar{g}_K \cdot n^4 \cdot (V - V_K) + \bar{g}_{Na} \cdot m^3 \cdot h \cdot (V - V_{Na}) + \bar{g}_l \cdot (V - V_l), V(0) = V_0$	
$\frac{dn}{dt} = \phi \cdot (\alpha_n \cdot (1 - n) - \beta_n), n(0) = n_0$	
$\frac{dm}{dt} = \phi \cdot (\alpha_m \cdot (1 - m) - \beta_m), m(0) = m_0$	
$\frac{dh}{dt} = \phi \cdot (\alpha_h \cdot (1 - h) - \beta_h), h(0) = h_0$	
<b>Hodgkin-Huxley</b>	$\phi = 3^{(7 - 6.3 \gamma 10)}$
<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>a) <math display="block">g_i(t) = (1 - dt_{max})g_i(t-1) + dt_{max} \left( \frac{1}{n_i} \sum_k s_k \frac{r_k}{\sum_p r_p} \frac{1}{\alpha_k} &lt; x_i w_{ij} &gt;_k + \beta \right)</math></p> <p>b) <math display="block">g_i^{\ominus} = \frac{g_i^* g_i (E_i - \ominus) + g_l g_l (E_l - \ominus)}{\ominus - E_i}</math></p> <p style="margin-left: 40px;"><math display="block">g_i(t) = g_i^{\ominus}[k+1] + q(g_i^{\ominus}[k] - g_i^{\ominus}[k+1])</math></p> </div> <div style="width: 45%;"> <p>c) <math display="block">V_m(t+1) = V_m(t) + dt_{max} [g_e(t) \bar{g}_e (E_e - V_m(t)) + g_i(t) \bar{g}_i (E_i - V_m(t)) + g_l(t) \bar{g}_l (E_l - V_m(t))]</math></p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="text-align: center;"> <math display="block">y_i = \frac{\gamma [V_m - \ominus]_+}{\gamma [V_m - \ominus]_+ + 1}</math> </div> <div style="text-align: center;"> <math display="block">y_j^*(x) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{z^2}{2\sigma}} y_j(z-x) dz</math> </div> </div> </div> </div>	
<h3 style="text-align: center;">Leabra Model</h3>	

Table 1. Top: Hodgkin and Huxley neuron model, based on chemical species. Bottom: Leabra model; a) Excitatory conductance; b) Inhibitory k-WTA function; c) Membrane potential; d) Activation function

Leabra uses a point neuron activation function that models the electrophysiological properties of real neurons, while simplifying their geometry to a single point. This function is nearly as simple computationally as the standard sigmoid activation function, but the more biologically-based implementation makes it considerably easier to model inhibitory competition, as described below. Further, using this function enables cognitive models to be more easily related to more physiologically detailed simulations, thereby facilitating bridge-building between biology and cognition.

Leabra uses a kWTA (k-Winners-Take-All) function to achieve inhibitory competition among units within a layer (area). The kWTA function computes a uniform level of inhibitory current for all units in the layer, such that the k+1th most excited unit within a

layer is generally below its firing threshold, while the k-th is typically above threshold. Activation dynamics similar to those produced by the kWTA function have been shown to result from simulated inhibitory interneurons that project both feedforward and feedback inhibition. Thus, although the kWTA function is somewhat biologically implausible in its implementation (e.g., requiring global information about activation states and using sorting mechanisms), it provides a computationally effective approximation to biologically plausible inhibitory dynamics. For learning, Leabra uses a combination of error-driven and Hebbian learning. Implementation diagrams are shown in Table 2.

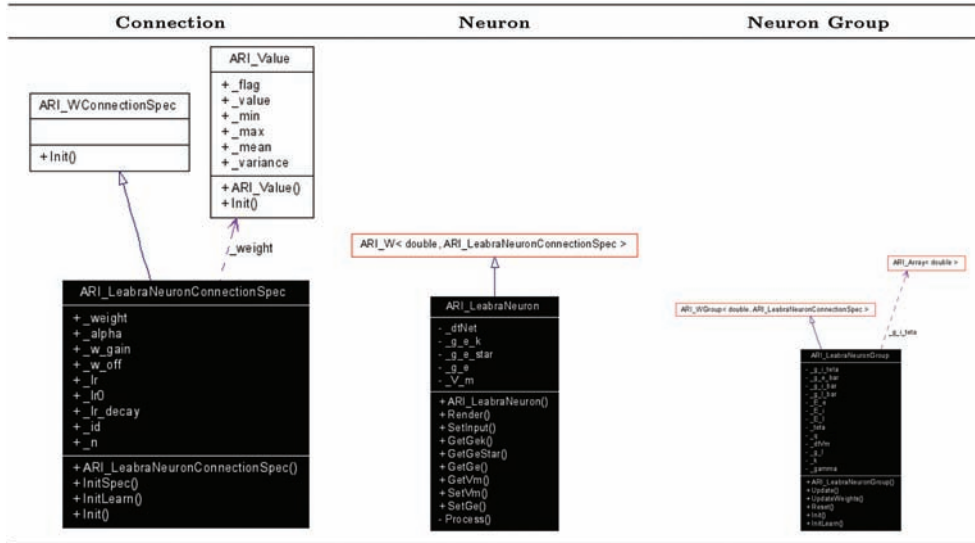


Table 2. Leabra model: inheritance and collaboration diagrams of structures for leabra model implementation

### 5.3 The Izhikevich artificial neuron

Izhikevich recently developed a simple model for an artificial neuron which is able to reproduce all the behaviours showed above [13]. The model takes 13 FLOPs for simulate one millisecond of neuron activity and it is based on a top-down approach, using two differential equation with four parameters. The introduction of axonal delays shows the possibility to create a neural network able to perform classification and prediction tasks. The connection of several maps follows and the Spike-Timing-Dependant Plasticity (STDP) rule, which permits the implementation of a real time learning rule based on signals which continuously flow from input systems. This architecture follows the theories of Edelman about the selection as the basis for the learning process.

The model proposed by Izhikevich for the artificial neuron simulation shows the ability to reproduce the same accuracy of the Hodgkin and Huxley model. It can be resumed in the following relations:

$$\begin{cases} v' = 0.04v^2 + 5v + 140 - u + I \\ u' = a(bv - u) \end{cases}$$



A reset condition is needed:

$$\text{if } v \geq +30mV, \text{ then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases}$$

The four parameters ( $a$ ,  $b$ ,  $c$  and  $d$ ) are dimensionless values. The  $v$  variable represents the membrane potential of the neuron, while  $u$  keeps into account the activation of  $K^+$  ionic currents and the deactivation of the  $Na^+$  ionic currents. The  $I$  variable takes into account the synaptic currents and the bias currents as the input signal of the neuron. Depending on the values of the four parameters, the system may have a steady-state (which corresponds to a lack of activity in the neuron) and an unsteady-state (which corresponds to the presence of activity in the neuron). The reset condition is needed to perform the return of the system into the steady state after the neuron has fired. Table 3 shows the values of the four parameters in order to obtain the known neuron behaviours.

In order to implement a network able to use the polychronization feature as it is described above, a software module has been realised. An *IzhikevichNeuron* structure (see Fig. 5.5) has been implemented in the framework as a running process directly deriving from the *W* structure.

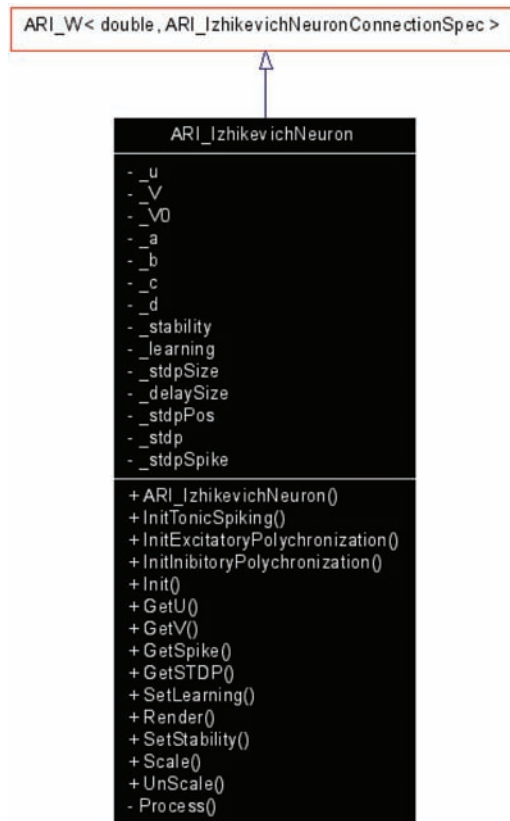


Fig. 5.5. The IzhikevichNeuron structure

Template arguments have been specialised to obtain an *OUTDATA* as a real number (double precision floating point value) which represents the membrane potential of the neuron, and a *CONNSPEC* as a *IzhikevichNeuronConnectionSpec* structure, which is shown in Fig. 5.6.

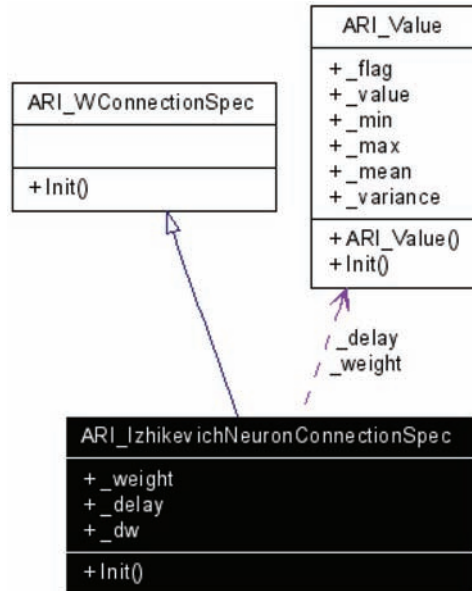


Fig. 5.6. The *IzhikevichNeuronConnectionSpec* structure

The connection structure for such a process uses real numbers to manage the synaptic weight and the synaptic channel delay. The values may be initialised by the user or randomly chosen by the framework according to the value initialisation parameters. A delta-weight value is needed for internal operations during the learning process, as it will be discussed later.

The *IzhikevichNeuron* structure is initialised using the *Init* method in order to setup the internal parameters (*a*, *b*, *c*, *d*) which specify the behaviour of the artificial neuron. Several initialisation wrapper methods are provided to use predefined behaviours as they are showed in Table 3. The STDP algorithm (Fig. ??) is implemented with a time-window of size equal to 1000 milliseconds. During this period the delta-weight values are updated according to the STDP rule, while weights are updated at the end of each period. During each period the structure traces the firing activity and the STDP status of the neuron, storing the information in two static arrays. The structure defines the private virtual method *Process* in order to perform the learning task. If the neuron is firing, the *Process* method reset the internal status (*u*, *v*) and the STDP value is reported to a value equal to 0.1. Otherwise the STDP value is decreased with a time-constant equal to 20 milliseconds. Subsequently the input connections are browsed to update input current, whose contribute depends on the thalamic input neurons and on those neurons who fired with a timing equal to the connection delay. According to the STDP rule, the STDP value of the post-synaptic neuron is increased if it fired synchronously with the the pre-synaptic neuorn, and it is decreased if

the pre-synaptic fire caused no firing in the post-synaptic neuron. Finally the status is updated following the Izhikevich model, and, if 1000 milliseconds are elapsed, the synaptic weights of the connections coming from the excitatory neurons are updated with the actual delta-weight values. During this step the weights are clamped within a convenient range and the delta-weight values are decreased with using a decay coefficient equal to 0.9.

	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>
A) Tonic Spiking	0.02	0.2	-65	6
B) Phasic Spiking	0.02	0.25	-65	6
C) Tonic Bursting	0.02	0.2	-50	2
D) Phasic Bursting	0.02	0.25	-55	0.05
E) Mixed Mode	0.02	0.2	-55	4
F) Spike frequency adaption	0.01	0.2	-65	8
G) Class 1 excitable	0.02	-0.1	-55	6
H) Class 2 excitable	0.02	0.26	-65	0
I) Spike latency	0.02	0.2	-65	6
J) Subthreshold oscillation	0.05	0.26	-60	0
K) Resonator	0.1	0.26	-60	-1
L) Integrator	0.02	-0.1	-55	6
M) Rebound Spike	0.03	0.25	-60	4
N) Rebound burst	0.03	0.25	-52	0
O) Threshold variability	0.03	0.25	-60	4
P) Bistability	0.1	0.26	-60	0
Q) Depolarizing after-potential	1	0.2	-60	-21
R) Accomodation	0.02	1	-55	4
S) Inhibition-induced spiking	-0.02	-1	-60	8
T) Inhibition-induced bursting	-0.026	-1	-45	-2

Table 3. Values of the four dimensionless parameters used to obtain the corresponding neuron behaviour.

The I/O buffering operations simply manage internal members and recall the base class methods. The rendering function provides the graphic visualisation of the soma and of the input connections.

The *IzhikevichNeuronGroup* structure (Fig. 5.7a), which represents a group of *IzhikevichNeurons*, has been derived from the *WGroup* base structure. The *IFNeuronGroup* structure will be used by high-level processes in order to perform the monitoring of the activity of the neurons during the learning and test tasks. Methods are provided to obtain the activation percentage (*GetActPerc*) and to retrieve the sub-group identification relating to a specified input pattern. A specific structure (*ARI\_ING\_Record*) has been realised to store the neuron reference and the activation time for each neuron belonging to the sub-group. Such records can be enumerated using the the iterator methods (*First*, *End*, *Next*, *Get*).

Moreover, an *IzhikevichMap* structure (Fig. 5.7b) has been derived from *W* base structure in order to speed-up the artificial neural group processing. This structure includes all the previous described structures, optimising the memory usage and computational efficiency.



Fig. 5.7. a) The IzhikevichNeuronGroup structure. b) The IzhikevichMap structure

A *ThalamicRandomSensorDriver* structure has been developed to train the architecture with random signals. Such signals are the basis of the cortico-thalamic interplay of neural assemblies and temporal chains in the cerebral cortex. A *Mic-SensorDriverStructure* has been used to test the architecture with audio signals. For such signals the power spectrum has been obtained using the *ARI\_FFT* structure. Both structures are showed in Fig. 5.8.

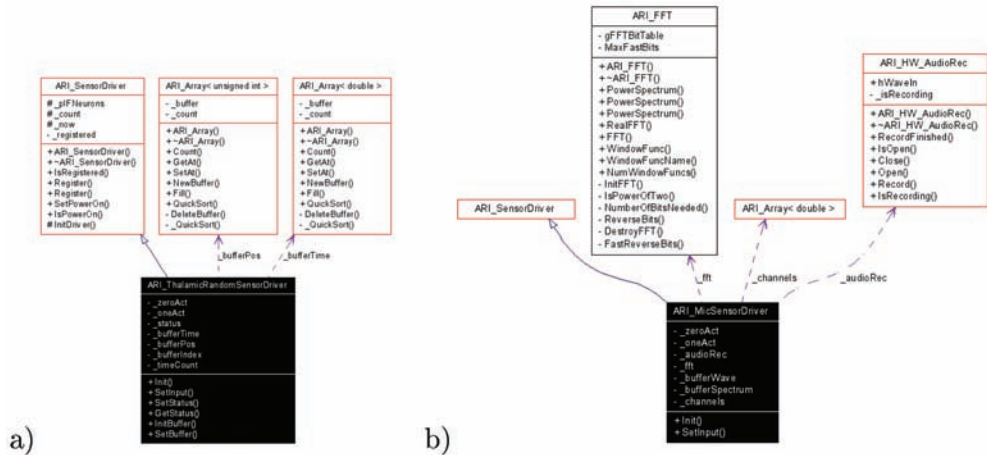


Fig. 5.8. a) The *ThalamicRandomSensorDriver* structure. b) The *MicSensorDriver* structure

## 6. Conclusions

In this work authors describe a high-efficiency architecture for parallel sensory fusion and real-time management of heterogeneous multi-transducers data processing. The interfaces with the external sensors and actuators, the specific control and processing methods and the data flowing through inner communication channels can be defined. For such entities the framework offers extendable structures, whose base implementation allows the realisation of high-efficiency data processing. Systems equipped with multiple transducers, tasks execution that are running as cooperative processes, off-line and real-time data acquisition and analysis tools, general stand alone applications represent some of the potential application areas.

A library-oriented interface was preferred to a user-oriented interface. Real-time analysis and actuation is gained for all the transducers and for all the running processes. Multi-process cooperation is possible thanks to a homogeneous communication language. The user can create extensions of new models of entities and processes. The data acquisition from sensor devices is granted by a protocol interface that is able to dispatch data coming from input systems. The data processing may be specified by the user inside the framework core. The actuator driving is granted by a protocol interface that is able to dispatch data from the framework core. Filters for sensory and actuating systems can be redefined according to the particular device technology; the efficiency of the filtering and buffering processes over the data coming from sensors and over the data directed to actuating devices is delegated to appropriate interfaces. The portability is allowed by a layered structure, an abstraction, and by the specification of the I/O drivers. A modular, reusable and object-

oriented architecture grants a parallel distributed processing, making the framework base architecture available to the researcher as a structured programming environment. Such features make the framework a solution for high-complex simulation tasks, representing a powerful instrument for the development of complex simulation tools operating as off-line and real-time applications.

## 7. References

- [1] Lee, K. & Schneeman, R., (2000). Distributed measurement and control based on the IEEE 1451 smart transducer interface standards. *IEEE Transactions on Instrumentation and Measurement*, Vol. 49, No. 3.
- [2] Steinberg, A.N. (2001). Data fusion system engineering. *IEEE Aerospace and Electronic Systems Magazine*, Vol. 16, No. 6.
- [3] Martin, C.; Schaffernicht, E.; Scheidig, A. & Gross, H.-M. (2006). Multi-modal sensor fusion using a probabilistic aggregation scheme for people detection and tracking. *Robotics and Autonomous Systems*, Vol. 54, No. 9, 721-728.
- [4] Li, C.; Heinemann, P. & Sherry, R. (2007). Neural network and Bayesian network fusion models to fuse electronic nose and surface acoustic wave sensor data for apple defect detection. *Sensors and Actuators B: Chemical*, Vol. 125, No 1, 301-310.
- [5] Pioggia, G.; Ferro, M. & Di Francesco, F. (2007). Towards a real-time transduction and classification of chemo-resistive sensor array signals. *IEEE Sensors Journal*, Vo. 7, No. 2, 237-244.
- [6] Ping, W.; Qingjun, L.; Wei, Z., Hua, C. & Ying, X. (2007). The design of biomimetic artificial nose and artificial tongue. *Sensors & Materials*, Vo. 19, No 5, 309-323.
- [7] Soria-Frisch, A.; Verschae, R. & Olano, A. (2007). Fuzzy fusion for skin detection. *Fuzzy Sets and Systems*, Vol. 158, No. 3, 325-336.
- [8] Härter, F.P. & Fraga de Campos Velho H. (2008). New approach to applying neural network in nonlinear dynamic model. *Applied Mathematical Modelling*, Vol. 32, No. 12, 2621-2633.
- [9] Damasio, A. (2000). *The feeling of what happens*. London: William Heinemann.
- [10] Driver, J. & Noesselt, T. (2008). Multisensory Interplay Reveals Crossmodal Influences on 'Sensory-Specific' Brain Regions, Neural Responses, and Judgments. *Neuron*, Vol. 57, No. 1, 11-23.
- [11] O'Reilly, R.C. & Munakata, Y. (2000). *Computational Explorations in Cognitive Neuroscience: Understanding the Mind by Simulating the Brain*. MIT Press: Cambridge.
- [12] O'Reilly, R.C. (2006). Biologically Based Computational Models of High-Level Cognition. *Science*, 314, 91-94.
- [13] Izhikevich, E.M. (2007). *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*. The MIT press.
- [14] Izhikevich, E.M. & Edelman, G.M. (2008). Large-Scale Model of Mammalian Thalamocortical Systems. *PNAS*, 105:3593-3598.
- [15] Whittington, M.A.; Traub, R.D.; Kopell, N.; Ermentrout, B. & Buhl, E.H. (2000). Inhibitionbased rhythms: experimental and mathematical observations on network dynamics. *Int. J. of Psychophysiol.*, Vol. 38, 315-336.
- [16] Sougnè, J.P. & French, R.M. (2002) Synfire Chains and Catastrophic Interference, *Proceedings of Annual Conference of Cognitive Science Society*.

- 
- [17] Norman K.A.; Newman E.L. & Perotte A.J. (2005). Methods for reducing interference in the Complementary Learning Systems model: Oscillating inhibition and autonomous memory rehearsal. *Neural Networks*, Vol. 18, No. 9, 1212-1228.
  - [18] Izhikevich, E.M.; Gally, J.A. & Edelman, G.M. (2004) Spike-Timing Dynamics of Neuronal Groups. *Cerebral Cortex*, Vol. 14, 933-944.
  - [19] Edelman, E. (1987). *Neural Darwinism: The Theory Of Neuronal Group Selection*. Basic Books.
  - [20] Izhikevich, E.M. (2006). Polychronization: Computation With Spikes. *Neural Computation*, Vol. 18, 245-282

# Agent Based Sensor and Data Fusion in Forest Fire Observer

Ljiljana Šerić, Darko Stipanichiev and Maja Štula  
*Faculty for Electrical Engineering, Machine Engineering and Naval Architecture  
Croatia*

## 1. Introduction

Although natural hazards have been present on earth forever, we still have not developed effective way of dealing with them, and because of that the early detection of natural hazards is still quite important task. Traditional methods based on human observers are mostly used, but they have shown ineffective because of a human observer's subjectivity.

Modern technologies, especially sensors technology provide tools for new techniques of natural hazard early detection. The sensor network technology is one of the most important technologies currently being investigated by scientist all around the world. The sensors, creating a sensor network, today could be deployed in nature on places where it was not possible to put sensors before, particularly because of today's development of wireless communication and miniature autonomous power supplies.

Collecting data from sensors creating a sensor network is one part of the research in this field, and another one is sensor network data utilization. The goal of sensor data interpretation, particularly using data fusion techniques, is to understand the sensor network surrounding and to create the image of the world around the sensors. In this paper this image of the world around the sensors will be call the scenario of the phenomenon.

Combining sensors for collecting data and advanced algorithms for data processing and interpretation, more advanced object called *the observer* could be designed. Combining several observers an observer network could be established, capable for better understanding what is going on in sensors surroundings. In such a way an advanced early warning system for initial phase natural hazards detection could be designed.

When done by humans, the process of recognition the phenomenon of interest based on fusion of sensory inputs is called *perception*. Our work was primarily inspired by formal theory of perception introduced in 1989 by Benett, Hoffman and Prakash (Bennett at al., 1989). This theory is used as a framework for formal description of data fusion processes in a observer network.

The observer network is a system consisting of its physical part, various sensors deployed directly in the environment integrated with appropriate communication components, and software part, intelligent mechanisms for sensor network data fusion and interpretation. Data fusion is used as a formal framework for combining data from different physical sensors of the same type, but also from different type of sensors. Data fusion main task is to obtain information of greater quality. The exact definition what 'greater quality' means is primarily application dependent (Wald, 1999).



Our main phenomenon of interest is forest fire. Forest fires represent a constant threat to ecological systems, infrastructure and human lives. Croatia belongs to countries with enhanced summer forest fire risk, particularly the Dalmatian coast and islands. Great efforts are therefore made to achieve early forest fire detection, which is traditionally based on human surveillance. Since 2003 we have intensively work at University of Split on development of automatic forest fire observer. The forest fire observer is the elementary node of forest observer network currently under development in Croatia, particularly in Croatian National Parks.

## 2. The observer theory

The observer theory is the formal theory of perception (Bennett at al., 1989). The theory is based on thesis that perception is process of inference and that it could be described mathematically using Bayes theorem. The goal of perception is to calculate the probability that specific scenario has occurred if sensory inputs are given; thus the process of perception can be described using Bayesian rule for conditioned probability (Bennett at al., 1996):

$$P(S | I) = \frac{P(I | S)P(S)}{P(I)} \quad (1)$$

where S is the scenario and I is the image, so the interpretation could be that the conditioned probability of occurrence of a scenario S, if the image I is given by sensory input, is equal probability of receiving the image I, if the scenario S happened, multiplied with probability of occurrence of S in the environment divided with probability of receiving the image I. Probability P(S) depends on the nature of the phenomenon the scenario describes, joined probability P(I|S) can be induced from sensory mapping functions, i.e. for image camera sensor it is perspective projection together with some limitations of the sensor and P(I) could be available from sensor information type.

According to the theory of perception in real situations this fraction is often uncertain, so more general form of Bayesian theory was used and presented using the formal foundation called the observer (Bennett at al., 1989). The observer is defined as a six-tuple:

$$O = (X, Y, E, S, \pi, \eta) \quad (2)$$

where X and Y are measurable spaces, E and S are subsets of X and Y respectively,  $\pi$  is a measurable surjective function and  $\eta$  conclusion kernel. Space X is a *configuration space* of the observer and E is a *configuration event* of the observer. Space X is a formal representation of those possible states of affair over which the configuration event E of the observer is defined. Y is an *observation space*, or *premises space*, of the observer. Space Y is a formal representation of the premises available to the observer for making inferences about occurrences of E. S is the *observation event*. Only points in S are premises of observer inferences which conclude that an instance of the configuration event E has occurred.  $\pi$  is a *perspective map*, the measurable surjective function from X to Y ( $\pi : X \rightarrow Y$ ) with  $\pi(E) = S$ .  $\eta$  is a *conclusion kernel* of the observer. For each point in the observation event  $s \in S$ ,  $\eta(s, \cdot)$  is a probability measure on E supported on  $(\pi^{-1}(s) \cap E)$ . This means that kernel  $\eta$  is a convenient way of assigning a probability measure on E to every point of S.

### 3. Forest fire observation and the forest fire observer

#### 3.1 Traditional approach

Forest fires are phenomena with devastating consequences. Preferred moment for fighting forest fires is during the initial fire stage. Automatic prediction of enhanced forest fire risk and early detection of forest fire in its initial stage is possible, but it requires a lot of visual, meteorological and historical data, detail information about the monitoring area, and quite complex processing procedures which include a lot of information fusion.

Today the traditional way of information fusion, having as a primarily goal the forest fire detection in early stage, is mostly based on human observers located in monitoring stations on monitoring spots. Human observer collects the data using natural senses, particularly vision, carefully observing his (or her) surroundings, but also reading the data from weather sensors displays and using past experiences and knowledge about particularly vulnerable and dangerous parts of landscape (for example because there is a railway or local road). In specific weather conditions the forest fire risk is enhanced and as a consequence, human observer attention is increased. But finally, the forest fire detection in its initial stage is done by vision. During the fire season a number of human observers, creating the human observers network, are usually responsible for early detection of forest fires in monitoring area. If the fire is spotted, the appropriate fire alarm is raised and the fire-fighters are activated.

#### 3.2 Observer network approach

The similar concept was used in our forest fire monitoring observer network designed as a semi-automatic system for raising fire alarms (Stipanicev & Hrasnik, 2007, Stipanicev et al. 2007a). Forest fire observer network was conceived as a system for real time data monitoring and information fusion having the same goal as the human observers – detection of forest fire in its initial stage. The system is semi – automatic, because the final decision for fire-fighters activation is done by human operator located in operation center. Now, using the forest fire observer network one operator can take care about larger area in comparison with human observers located on monitoring spots. In forest fire observer network the technical devices called forest fire sensors are located on monitoring spots while processing units and human operator are located in an operation center. The benefit of such system is not only that one operator can control larger area, but also his (her) working conditions in the operation center are much better and lower level of operator concentration is required because the system will warn the operator if any level of hazard is possible.

Forest fire detection in initial stage is based on environmental data collected with the sensor network deployed directly in the environment. Various sensors are responsible for sensing different aspects of reality connected with forest fire phenomenon. The same as in the case of human observers the most important sensor of forest fire observer network is the vision sensor and that is the pan/tilt/zoom controlled CCD camera sensitive in visible spectrum located on appropriate monitoring spot. Additional information are collected using a mini meteorological stations measuring temperature, humidity, wind speed and direction, atmospheric pressure, sun radiation, wetness, lightning discharges etc. An example of our forest fire sensors located in on observatory on mountain Mosor near Split and in Nature Park Vrana lake is shown in Figure 1.

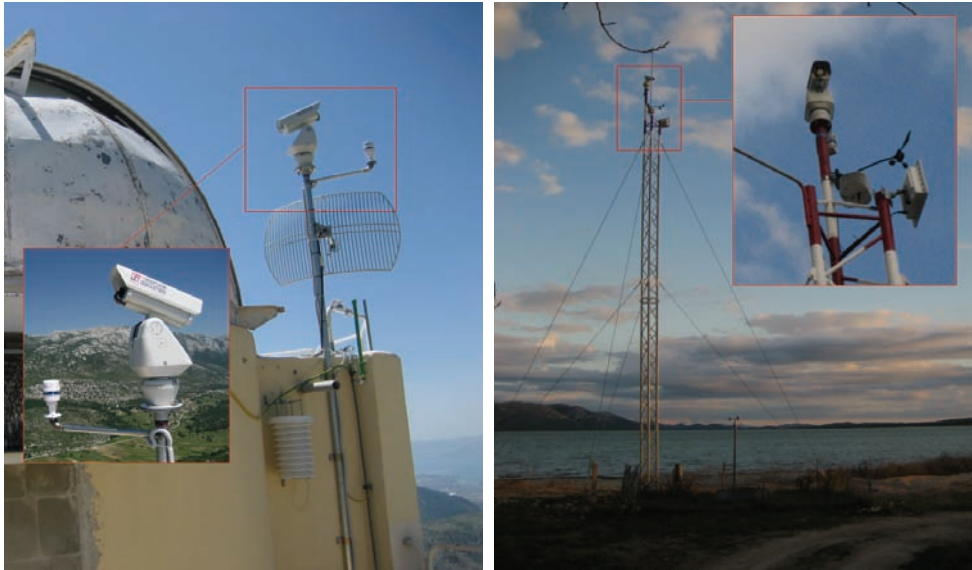


Fig. 1. Forest fire sensors on mountain Mosor near Split and in Nature Park Vrana lake

The main task of the forest fire observer is to collect all these data and to perform algorithms for data validation, data fusion and data interpretation, having as final result information about phenomena taking place in the environment, namely the forest fire in its initial stage. When performed by human observer, these actions are performed using complex processes of perception, so the inspiration to our work was the process of human perception based on the observer theory explained in short in Section 2. Let us here illustrate how observer theory could be applied to the forest fire observer.

Forest fire is a phenomenon of interest which may be recognized from sensory inputs. Simple forest fire observer consists of six entities as stated in equation (2).

The first part is the configuration space  $X$ . The space  $X$  is a set of all possible scenarios that can be recognized in environment. We call this set *scene configuration*. Some of those scenarios which could happened in our environment observed by forest fire observers are thunder, lightning, twister, fog, forest fire or simple sunny day phenomenon. Only those scenarios holding the occurrence of forest fire are collected in a configuration space  $E$ , which is a subset of set  $X$ . We refer to  $E$  the name the *phenomenon configuration*. So set  $E$  holds those scenarios where phenomenon of interest is identified, in our case only the forest fire including all its features, primarily the existence of smoke and flames.

The second part is the observation space  $Y$ . Sensors deployed in environment are used to map the input space  $X$  to the output space  $Y$ . We call the space  $Y$  the *scene observation*. The mapping function, or the sensors function is the perspective map  $\pi$ . Depending on the senses available, it can have only elements of orthogonal projection of three dimensional input places to two dimensional image if only a camera or image sensor is available, or it can include some measurements done by other sensors like meteorological sensors. Set  $Y$  holds images of scenarios in  $X$  with respect of this perspective map. The perspective map has to be injection, but not necessary surjection, so the set  $Y$  usually holds less elements then the set  $X$ . When mapping is done, all elements of  $X$  are mapped into elements of  $Y$ , and

those elements of  $X$  which are also elements of  $E$ , are mapped into elements of  $Y$  which are elements of  $S$ . The set  $S$  is a subset of  $Y$  holding those images of scenario where observer can identify phenomenon from  $E$ , and we name this set the *phenomenon observation*. So the conclusion could be: If an image from  $S$  is recognized by observer, the observer concludes that the phenomenon has occurred. But the problem is that the perspective map is not surjection, so sometimes happens that a scenario from  $E$  (fire) and a scenario from  $\neg E$  (not fire) could have the same image inside the set  $S$ . So if former scenario (not fire) happens, the observer can falsely conclude that phenomenon had happen. This situation is called the false alarm. The probability of the false alarm is given by the conclusion kernel  $\eta$ . The kernel  $\eta$  gives for each element of  $S$  the probability distribution supported on  $E$ , thus the conclusion kernel gives the final result of our observer – the probability that different scenarios from  $E$  really happen and belong to  $S$ . The illustration of forest fire observer is shown in Fig. 2.

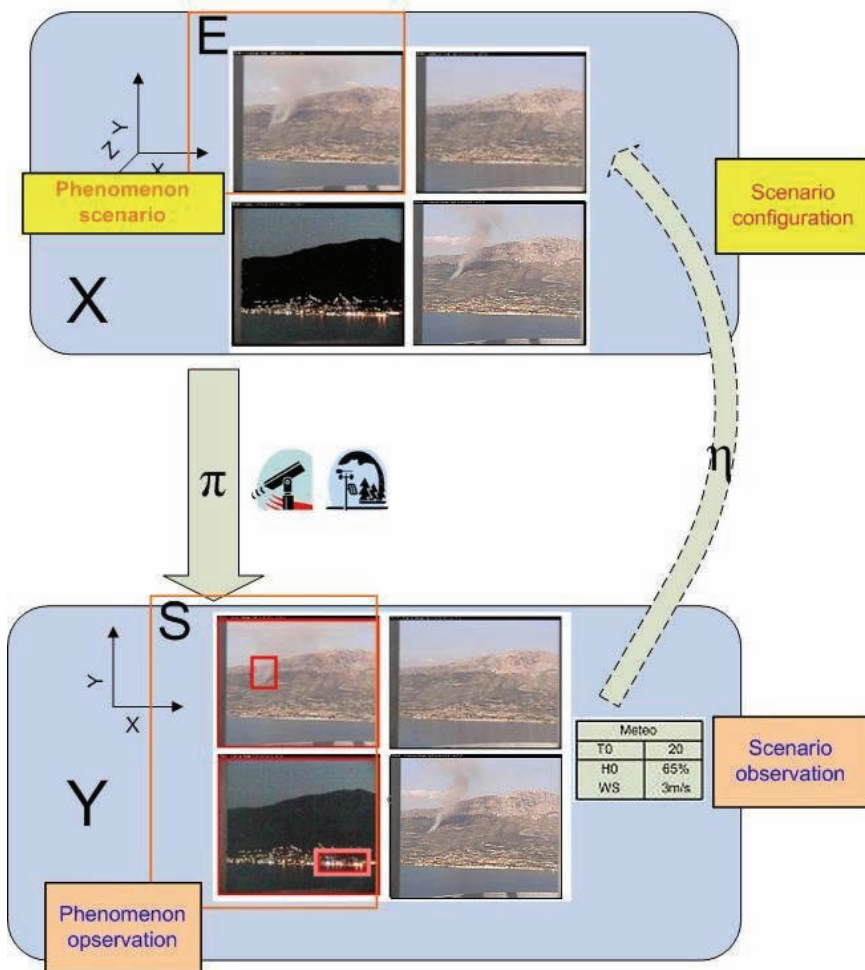


Fig. 2. Illustration of the forest fire observer

When a phenomenon  $x_1$  takes place in the environment and it is a forest fire, this event  $x_1$  is inside the subset  $E$  of  $X$ , so we can write  $x_1 \in E$  (photo in upper left corner of  $X$  in Fig.2). In observation space one observation event  $y_1$  corresponds to this configuration event (photo in upper left corner of  $Y$  in Fig.2). If  $y_1$  is inside  $S$  ( $y_1 \in S$ ), and the observer concludes that this observation event  $y_1$  was caused by the configuration event  $x_1$  it assigns to pair  $(\pi^{-1}(y_1), x_1)$  the biggest probability measure. The final observer conclusion is "There is a forest fire" and in decision diagram this situation is treated as a *hit* or *correct detection*.

The second case is for phenomenon  $x_2$ , which is not a forest fire ( $x_2 \notin E$ , photo in upper right corner of  $X$  in Fig.2). Let us suppose that the mapped data into observation space are also "clean", without any forest fire signs, so the corresponding observation event ( $y_2 \notin S$ , photo in upper right corner of  $Y$  in Fig.2). The final observer conclusion is "There is not a forest fire" and in decision diagram this situation is treated as *correct reject*.

The third case happens when phenomenon  $x_3$  which is outside the subset  $E$  ( $x_3 \notin E$ , photo in lower left corner of  $X$  in Fig.2) is mapped into an observation event  $y_3$  which is inside the subset  $S$  ( $y_3 \in S$ , photo in lower left corner of  $Y$  in Fig.2). This situation is called the *false alarm*, because the configuration event is not in  $E$ , but the observation event is in  $S$ . The conclusion kernel should be defined in such a way that these images are assigned lower probability on  $E$  than in the case of correct detection. The false forest fire alarm could be for example triggered by clouds, clouds shadows or mist.

The fourth possible combination is when configuration event  $x_4$  is in  $E$  ( $x_4 \in E$ , (photo in lower right corner of  $X$  in Fig.2) which means that there is a forest fire, but the observation event  $y_4$  is outside  $S$  ( $y_4 \notin S$ , photo in lower right corner of  $Y$  in Fig.2). This situation is called the *miss* case and it is the worst case, taking into question the efficiency of the observer. The phenomenon (forest fire) has happen, but the observer was not able to recognize it. The miss case should be avoided and if the observer is defined well then these situation should never happen.

Four possible decisions of forest fire observer are illustrated in the decision diagram in Fig. 3.

	$x_i \in E$	$x_i \notin E$
$y_i \in S$	True fire detection	False fire alarm
$y_i \notin S$	Fire false miss	True fire miss

Fig. 3. Decision diagram for the forest fire observer

The quality of the observer could be evaluated according to the number of false alarms and miss cases, and the waiting factor of these two possible decisions could not be the same. In good forest fire observer the set of miss cases has to be the zero set, and the cardinality of the set of false alarms has to be as close as possible to zero.

The collection of observers form the observer network. The observer network is responsible for phenomena detection in wider area. For example according to our analysis Split and Dalmatia County, which is the costal and island county in Croatia located in central part of Adriatic, having alltogether 14 045 km<sup>2</sup> and 4 572 km<sup>2</sup> on land, could be optimaly covered by the forest fire observer network having 56 forest fire sensors (Stipanicev et al. 2007b). Fig. 4 shows their locations and covering. The forest fire sensor on mountain Mosor shown in Fig.1 will be the part of this network, so it is marked in Fig.4. too.

#### 4. Observer network architecture

The process of human perception could be divided in two steps: the first one is self perception when observer examines if the readings form his (her) senses are valid (Linsay & Norman, 1977). If observer in self perception step concludes that there is something wrong with senses, then any other conclusion can not be valid. The second step is identification of scenario based on information from sensors. Having this in mind, we have defined the three layers observer network architecture. Similar architecture was proposed by EU EYES project (EYES, 2005).

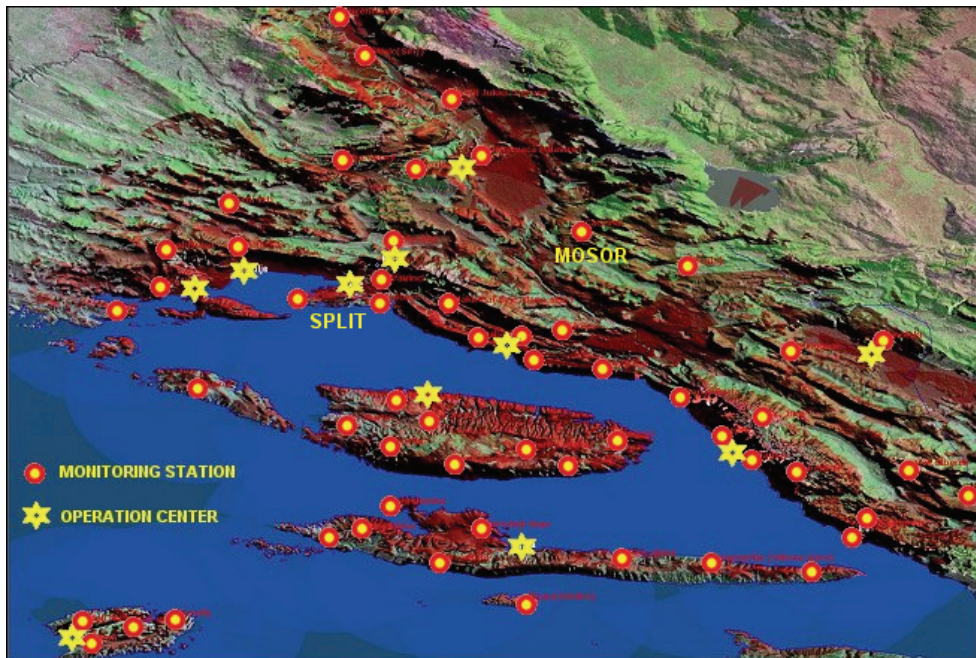


Fig. 4. Forest fire observer network of Split and Dalmatia County

This architecture is designed to be a unified framework for intelligent support of observer network. The framework covers processes of sensor network data collection, validation and implementation in a specific use case. In addition mechanisms for self observation and self reparation are included. The observer network architecture is shown in Fig.5.

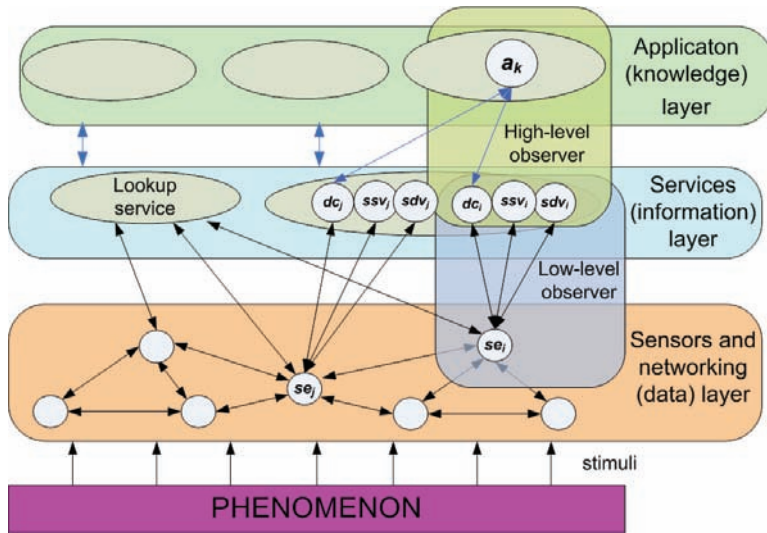


Fig. 5. Illustration of observer network architecture

The information flow is bottom up. Two types of observers has been defined – the low level observer or image fire observer (IFO) and the high level observer or decision fire observer (DFO), both of them composed of elements on two neighboring layers (Stipanicev et al., 2007c).

**Low-level observer** includes sensors on data layer and processing nodes on information layer. Some aspects of the environment are measured by sensors on data level and transferred as a raw data to nodes on information layer. The sensor  $se_i$  is used for mapping the scene configuration space  $X$  to first level of scene observation space  $Y$  with its own perspective map. On the information layer there are three nodes. The first one  $dc_i$  is responsible for transformation of raw input data to structured information used as an input of high-level observer. Other processing elements  $ssv_i$  and  $sdv_i$  are responsible for observer self perception, performing the validation of data integrity and testing the sensors functionality. Syntactic and semantic validation of data is performed based on the knowledge about the sensor's type and features, as well as their locations. Locations are important because data from other sensors in vicinity of the validated sensor are used during its semantic validation.

**High-level observer** includes several nodes on information layer and one processing node for each type of observation on the knowledge layer. Let us suppose that only forest fire detection is defined as an observation task. Node  $a_k$  is responsible for that on knowledge layer, but his decisions are based on information presented to him by various nodes on the information layer. To explain its functionality we will simplify the procedure of forest fire detection and said that it is based on three elements: motion detection, segmentation in RGB space and information is it raining or not. So, forest fire detection node  $a_k$  on knowledge layer has three corresponding nodes on information layer. Their elements  $dc_i$ ,  $dc_n$  and  $dc_j$  are responsible for preparation of appropriate input data. Data presented by nodes  $dc_i$  and  $dc_n$  are results of image analysis, and data presented by the third one  $dc_j$  is a result of moisture sensor row data interpretation. This means that both nodes  $dc_i$  and  $dc_n$  are connected with

the same sensor on the sensor layer and that is the video camera. The third node  $dc_j$  has a different sensor on the sensor layer and that is the moisture sensor. Both nodes  $dc_i$  and  $dc_n$  have the same raw data input (the same digital image), but they transform this input data in different type of output information. Node  $dc_i$  is a motion detector, so his output data, presented to forest fire decision maker  $a_k$  are regions on the image where motion has been detected. Node  $dc_i$  is responsible for RGB segmentation, so his output data presented to forest fire decision maker  $a_k$  are regions on the image where image red, green and blue components are inside predefined bands. The third node  $dc_j$  is the most simple one having moisture sensor data as input and binary information 1 (it is raining) or 0 (it is not raining) as an output to  $a_k$ . Now the task of forest fire decision maker is to conclude combining all input data is there a fire on the image or not and if there is a fire where it is located.

On application (knowledge) level it is possible to have various nodes and various applications for example not only forest fire detection, but fog detection, thunderstorm detection, intruder detection, etc. Each of them is connected with various nodes on the services (information) layer. We would like to emphasis that application nodes on knowledge layer make their decisions using data fusion and sensor fusion procedures. Their decisions are based on interpretation of data from various sensors, but also interpretation of data from the same sensor processed in different ways.

## 5. Multi agent architecture

The first part of our work was to define observer network organization, and the second one was its implementation in real life. Before choosing the best environment and architecture for implementation of observer network system we have stated a number of requests, primarily the system has to be modular, suitable to run on distributed environments and controlled through a number of user parameters. Last but not least was that the knowledge base has to be easily accessed and changeable.

Our final choice was the multi agent architecture configurable using database, knowledge base and properties files. Observer network realized in the form of multi-agent system consists of following parts:

- **Ontology** - defined to ease communication between intelligent agents, and between intelligent agents and external programs. Ontology adds meaning to the database slots.
- **Multi agent shell** - the core of the system, a shell consisting of agent bodies implementing the observer network functionality. During run time the number of living agents is determined according to the sensor network configuration and system properties written in system files.
- **Data base** - holds two kinds of tables: administrative tables with the system configuration and data tables with data information and alarms. Each level of observer network architecture has its own result written in its own table.
- **Knowledge base** - holds additional information, like the common sense rules, about sensors and sensor relative connections. Can be updated during run time, because the system has to be capable to learn.
- **Properties file** - tells the agents where to find database and external files.
- **External programs** - virtual sensors can be implemented in the form of formulae or in the form of external programs for more sophisticated calculations and analysis. These external programs can exist independent on the observer network system, but can be used by agent responsible for collecting virtual sensors data.



Multi agent system, incorporating the three layer architecture is responsible for sensor network data collection, verification, and forest fire detection. Intelligent agents are implemented using JADE - Java Agent Development Environment (JADE, 2008). Agents run Rete algorithm in knowledge based reasoning (Singh et al. 2004). The multi agent architecture of the system is shown in Fig. 6.

An agent of type *CollectorAgent* is responsible for data collection and archiving data into a database. *SyntacticObserverAgent* is triggered when data is stored into database with purpose of checking syntactic validity. *SemanticObserverAgent* input are sensor data from the same observer node. This agent first recognizes the same kinds of values on a same observer node, deployed for the redundancy, and notices the difference in them. Small differences are ignored, because although the sensors are covering the same location, differences in their exact location, elevation and orientation can cause different values. But larger outliers are rejected.

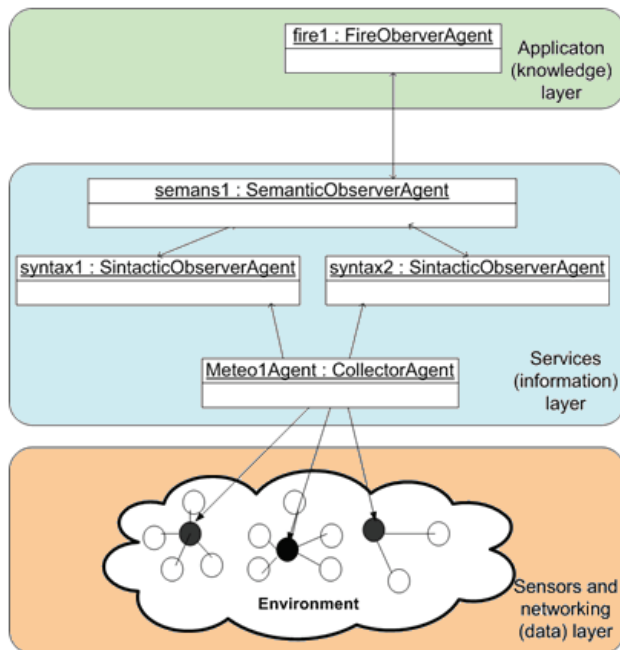


Fig. 6. Multi agent system of the observer network

### 5.1 Observer network ontology

As already stated in previous paragraph, ontology is defined to help the communication between different parts of the system, since those parts are not running in the same process, sometimes not even on the same computer. The ontology of observer networks provides contextual framework to messages that are exchanged between these parts.

Ontology of observer network can be divided in two parts - domain independent and domain dependent part. Domain independent part of ontology holds concepts and formalisms related to sensor and observer network configuration. The concepts of domain

independent part of the ontology can be linked to layers of three layer observer network architecture, as shown in Fig. 7.

We can consider the topmost concept to be the concept of location which can be compared to the observation station in traditional approach. The concept of location holds all relevant data about the physical location in the environment where the sensor node is situated. Usually one observer node holds one or more Data Server entities with sensing devices attached to it.

Domain dependant part of ontology holds the description of phenomenon taking place in the environment observed. In the case of forest fire observer, this concept holds the description how the forest fire is recognized using sensor values from the sensor network. The most important sensor in this case is the image sensor - video camera sensitive in visible spectra, but meteorological conditions can also be of use in recognition of forest fire and in rejection of false alarm from the image sensor.

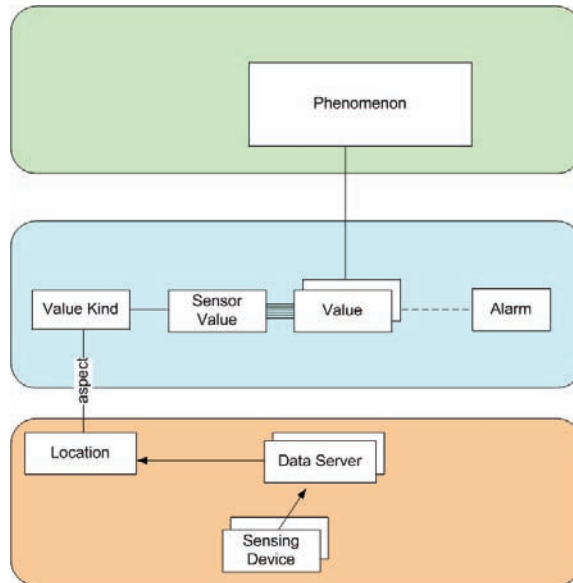


Fig. 7. Concepts of ontology distributed on observer network architecture

**5.2 Multi agent data fusion**

The whole observer network system performs data and information fusion with the goal of constructing the image that is most likely to correspond to the scenario taking place in the environment. Information about the configuration of sensor network and observer network, the knowledge about the phenomenon of interest and data from sensor network are input parts of this fusion and a fusion result is more sophisticated single information - image of scenario.

Following the formalism of observer, configuration space is the space consisting of every phenomenon that can take place in the environment. The configuration event is that phenomenon this observer is responsible for. The sensor network configures the perspective function for mapping different aspects of phenomenon into an observation space consisting

of data from sensor network and information about sensor network including observer environment. Observation event is the subset of the observation space where phenomenon of interest can be recognized. When an observation of the phenomenon happens, the observer network mechanisms decides which scenario from configuration space is most likely to be the true and that it has happened in the environment.

## 6. Conclusion

Inspired by the formal theory of perception and technology of sensor network we have introduced the idea of observer network as a reliable framework for data and information fusion. Our ideas have been successfully tested in the case of forest fire observer network. Observer network was implemented using multi-agent technology. A special multi agent shell was designed for this purpose having software system desirable features like modularity and flexibility.

The system was implemented in number of locations in Croatia under the name **iForestFire®** (iForestFire, 2008), particularly in national and nature parks as a small level observer networks. Two bigger observer networks are now in realization. The first one is in County of Istra where 22 forest fire sensors are already implemented and the second part dealing with multi-agent observer network is now in realization phase. Buzet subsystem is realized and shown in Fig.8. It is conceived of 5 forest fire sensors.

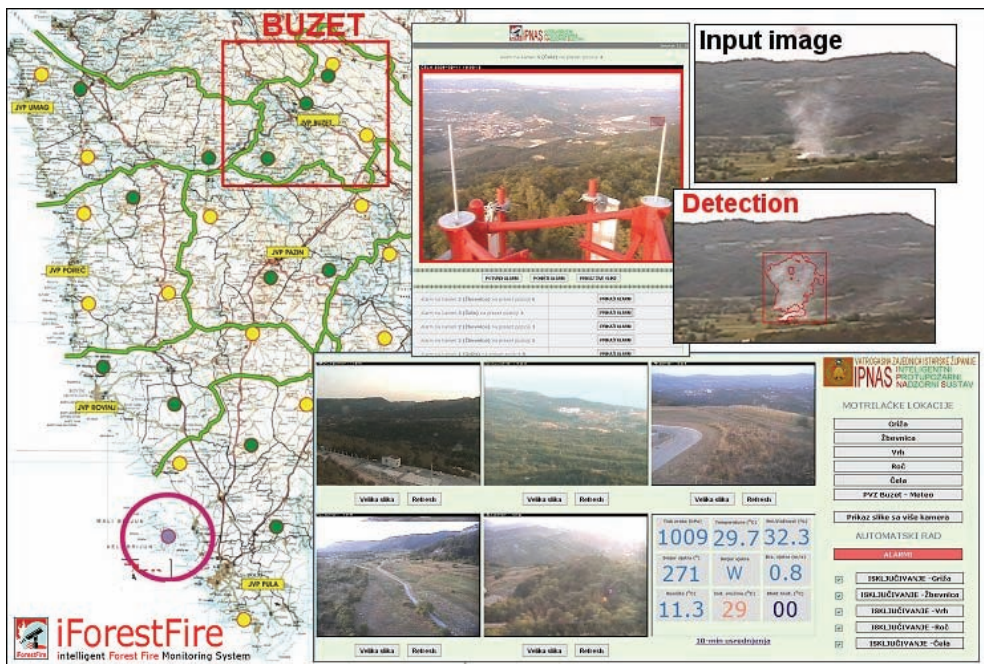


Fig. 8. Forest fire sensors location and various screen of Buzet forest observer network

The Split and Dalmatia County forest fire observer network shown in Fig.4. is also planned for realization.

Our case study was forest fire detection, so forest fire observer network was designed, but the same concept could be used for recognizing phenomenon of any other kind.

## 7. Acknowledgement

This work was supported by the Ministry of Science, Education and Sport of Republic Croatia under Grant 023-0232005-2003 "AgISEco-Agent-based intelligent environmental monitoring and protection systems", TP-03/0023-09 "System for early forest fire detection based on cameras in visible spectra" and Split and Dalmatia County authorities through a study „Holistic approach to forest fire protection in Split and Dalmatia County“. In this paper only sensor and data fusion principles of forest fire observer network are explain, but for overall system functionality detection algorithms are also important and they were developed and implemented by Damir Krstinić, and Kaja Radić was responsible for experimental systems realization.

## 8. References

- Benett, B.M.; Hoffman, D.D. and Prakash, C. (1989). Observer Mechanics – A Formal Theory of Perception”, Academic Press Inc., Public Domain
- Benett, B.M.; Hoffman, D.D.; Prakash, C. and Richman, S. (1996). Observer theory, Bayes theory, and psychophysics. In D. Knill & W. Richards (Eds.), Perception as Bayesian inference, Cambridge: Cambridge University Press, pp. 163-212. (UCI MBS Memo 93-12)
- Bodrožić, Lj., Stipaničev, D. & Krstinić, D. (2007) Data Fusion in Observer Networks, MASS 2007, The 4th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, October 8-11, 2007, Pisa, Italy
- EYES (2005) Energy Efficient Sensor Network, EU Research Project, 2002 - 2005, <http://www.eyes.eu.org>
- iForestFire (2008) Intelligent Forest Fire Monitoring System, [www.iforestfire.com](http://www.iforestfire.com)
- JADE (2008) Java Agent Development Framework - <http://jade.tilab.com>
- Lindsay, P. & Norman, D.A. (1977) Human Information Processing: An Introduction to Psychology, Harcourt; 2nd edition
- Singh, P., Minsky, M. & Eslick, I. (2004). Computing commonsense. BT Technology Journal, 22(4):201-210.
- Stipaničev, D. & Hrastnik, B. (2004) Holistic approach to forest fire protection in Split and Dalmatia County, study for Split-Dalmatia County (220 pages in Croatian)
- Stipaničev, D.; Vuko, T., Krstinić, D., Štula, M. & Bodrožić, Lj. (2006) Forest Fire Protection by Advanced Video Detection System - Croatian Experiences, Third TIEMS Workshop - Improvement of Disaster Management System, Trogir, Sept. 26 - 27, 2006., CD proceeding, 10 pages
- Stipaničev, D., Hrastnik, B. & Vujčić, R., (2007a) Holistic Approach to Forest Fire Protection in Split and Dalmatia County of Croatia, Wildfire 2007 Int. Conference, Sevilla, Spain, May 2007.
- Stipaničev, D.; Vuko, T. & Bodrožić, Lj. (2007b) Location Determination of Automatic Forest Fire Monitoring Stations based on AHP and GIS data, TIEMS (The International Emergency Management Society) 2007 Int. Conference, Trogir, June 2007. CD proceeding, 8 pages

- 
- Stipaničev, D.; Bodrožić, Lj. & Štula, M. (2007c) Environmental Intelligence based on Advanced Sensory Network, Proceeding of 14<sup>th</sup> Int. Conf. on Systems, Signals and Image Processing (IWSSIP 2007), Maribor, Slovenia, 27-30.6.2007.
- Wald L. (1999) Some terms of reference in data fusion, IEEE Transactions on Geosciences and Remote Sensing, 37 (3), 1190- 1193.

# A Sensor Data Fusion Procedure for Environmental Monitoring Applications by a Configurable Network of Smart Web-Sensors

Claudio De Capua and Rosario Morello  
*Department of Computer Science and Electrical Technologies  
Mediterranea University of Reggio Calabria  
Italy*

## 1. Introduction

The present Chapter describes a methodical approach to the design of a monitoring process for environmental applications. The case study and experimental results concern the remote control of exposure levels to electromagnetic pollution. The main contribution focuses on the design of a configurable network of wireless and smart web-sensors and the development of a two-levels data fusion procedure. The aim of an environmental monitoring process is to provide qualified information on the investigated phenomenon, so to minimize possible errors or hazardous consequences for the exposed population. That objective requires not only an appropriate choice of the measurement system, but care has to be paid about the maintenance of instrumentation in order to assure a suitable metrological state. On the contrary measured data may be inconsistent and meaningless so to provide an erroneous knowledge about the monitored process, often that occurrence is cause of underestimated risks. Therefore the operating state of the measurement system has to be guaranteed with the passing of time, in order to get reliable data. So the matter requires to evaluate the measurement system performances, (Neely et al., 2000). Moreover data processing stage plays a crucial role when decisions have to be taken with reference to warning limits. In fact several laws and regulations have been issued to limit the exposure levels to environmental parameters responsible for pollution, in order to keep under control the status of our habitat and so to guarantee an appropriate quality of life. But such decisions are the result of a comparison between the fixed limits and the measured data, consequently the uncertainty contribution due to measurement process may be reason of wrong decisions if its effect is not taken into consideration. In that context, the considered application field needs methodical monitoring processes being able to characterize and manage in real time warning and risky situations so to reduce possible harms for the exposed population. In the urban centers today it is possible to characterize several electromagnetic pollution sources to different working frequencies, as in example the lines of power supply or the antennas of the radio-television service, (Ahlbom et al., 1998; IEC 50166-2, 1995). Medical studies would seem to point out a relationship between the continual exposure to high electromagnetic field levels and the onset of some typologies of cancer. The World Health Organization suggests prudence and the observance of the quality

standard levels. Therefore, environmental data and available information have to be timely processed by means of suitable procedures in order to guarantee specific requirements of accuracy and reliability, so to minimize errors and risks. Furthermore sensor data may be erroneous, inaccurate or incomplete because of malfunctions, anomalies, delays, unsuitable maintenance or limited range. In addition, typically, environmental monitoring requires a wide network of sensors displaced on a large area, such sensors have inevitably different metrological characteristics. So a huge amount of data has to be processed in short time according to information on measurement uncertainty, topographical data, operating state of the sensors and further knowledge. At this aim the authors propose a sensor data fusion procedure being able to merge such information in order to optimize the processing so to get information of greater quality. The developments of the research in the environmental monitoring field show several open problems to be treated. For example, distributed systems have to be dynamically configurable according to changing conditions of the surrounding environment, or to environmental and topographical information, (Lin & Gerla, 1997). The actual sensor networks do not provide effective solutions and the used architecture is not able to configure itself, (Bertocco et al., 2002; Hou et al., 2004; Lee & Song, 2006; Mahfuz & Ahmed, 2005; Saripalli et al., 2006; Tsujita et al., 2005). Moreover sampling plans have to be flexible according to the desired accuracy and specifications. Also population density distribution has to be considered, in fact in presence of alarm occurrences, it is possible to characterize the zones which need more attention and have higher intervention priority as the most populated ones. The present Chapter focuses attention on such matters in order to propose possible useful tools for environmental monitoring applications. In the first part of the Chapter the design of a configurable network of wireless and smart web-sensors is described. Since electromagnetic field is characterized by spatial and temporal variations, a distributed network has been designed. As a matter of fact, weather conditions, the presence of metallic objects and antennas of radio-mobile service may affect the trend of the electromagnetic field. Therefore sensors, displaced along a wide urban area, measure the electromagnetic field levels in high and low frequency ranges according to suitable sampling plans. Then a remote processing *Server* allows to fuse the available information so to evaluate the overcoming of the law limits and the occurrence of alert states. By General Packet Radio Service (*GPRS*) communication the network exchanges data and commands with a remote client, then data are stored on *WEB Pages*. Each measurement unit consists of an isotropic smart sensor, a *GPS* module and a *GPRS* modem. In order to optimize the configuration of the sensor network, an algorithm has been developed. In this way the network is configurable according to the needs, and the area is divided in local zones. The size of the partition depends on the accuracy and the resolution required for the monitoring, so environmental and topographical information is used to optimize the network configuration and the monitoring map. If the specifications and accuracy requirements change, it is possible to configure dynamically the sensor network and the sampling plans by a new partition of the area. The single sensor has the task to monitor a specific zone and, according to the monitoring map, sends data to an *ASP Web Page*. A *Server* acquires the measured electromagnetic field levels for each zone, with information on the state of the single sensor like its measurement uncertainty, its reliability and its operating state; data are protected by a password system. The second part of the Chapter describes a sensor data fusion procedure developed by the authors in order to qualify the data processing stage. A two-levels approach allows the progressive fusion of

the huge amount of data so to get first local and then global information about the exposure to electromagnetic field in the monitored area. In fact, each sensor allows to collect data in a specific zone, such information provides only a small 'image' of the whole area concerning a restricted spatial portion correlated with the near zones. The data fusion procedure merges those partial 'images' so to provide a view of the electromagnetic field behavior in the area. In this way the limitation of the single sensor is overcome and a more accurate knowledge on the field exposure is got. By data fusion, the initial large amount of information may be used to get a more accurate, consistent and meaningful result. Errors are so reduced, as much as possible, using the same redundancy of data. The main problem is to characterize optimal fusion rules being able to merge complementary information so to minimize the global probability of fault or error. At this purpose a Statistical Model estimates the sensor reliability curves and its operating state. In this way measured data, information on the sensor network reliability and measurement uncertainty are processed for a more efficient interpretation of the data and a more accurate representation of the monitored area than that one provided from the single sensor; the useful information is so maximized. Furthermore it is interesting to notice that typically sensors are in a continuous working state so they may go to faulty occurrences. In such circumstance, data fusion represents an effective tool to get accurate and efficient information from faulty measurements. The proposed procedure fuses data and correlated information so to get an improved accuracy. The *Server* runs the procedure in order to verify the compliance of the exposure levels to electromagnetic field with the limits established from law. A fuzzy algorithm allows to perform a first level of data fusion. Data of the single zone are processed by a decision-making algorithm in order to take local decisions concerning warning or alarm occurrences. The erroneous decision probabilities are estimated according to the statistical distribution of process and the measurement uncertainty, in this way the initial data amount is reduced and local information is got. A second level of data fusion allows to get a global information about the exposure to electromagnetic field in the whole area, in this way local decisions and environmental information are fused in order to get a more accurate image about the pollution status of the area. A report shows the global situation. Consequently, the zones which require more attention are characterized, and suitable corrective interventions can be planned. Quality indexes provide information on reliability and consistency of the results. The originality of the present contribution is due to the design of a configurable network of smart sensors and to a two-levels data fusion procedure compliant with the quality assurance requirements, (ISO 9001, 2000). Knowledge on measurement uncertainty and sensor reliability is merged in order to optimize the available information and obtain a better estimation about the monitored process by using the same data redundancy.

## **2. The sensor network design**

### **2.1 The configurable architecture**

In the present paragraph is described the design of the monitoring network used. The aim is to verify the compliance of the electromagnetic field levels with the limits established by laws, and to get global information about the field behaviour on the whole monitored area. Today networking and communication sectors have achieved remarkable developments in real-time applications, but the increasing complexity of systems and networks is cause of further problems concerning the management and maintenance of the used instrumentation. So not always the available network architecture is suitable for obtaining a



consistent and significant view of the observed phenomenon. As a result in order to guarantee reliability and accuracy, it is required an accurate choice of instrumentation, but also suitable procedures estimating its metrological state and measurement uncertainty. In this view, the authors propose the project of an original configurable network of wireless and smart web-sensors. Main features of the contribution regard the possibility to configure dynamically the network according to the needs and the topographical information. So the sampling plans, the partition of the area and the sensors layout are updated if requirements and specifications change. By a methodical approach, a distributed network has been projected so to monitor a wide urban centre (Reggio Calabria city, in the south of Italy). The network and the smart web-sensors have been projected according to the guidelines of the *IEEE 1451 Standard*, (IEEE 1451, 2001). The whole monitoring process has been optimized by innovative procedures being able to manage the network and maintain the sensors, so to check the operating state of the instrumentation and estimate the next calibration interval. The choice of a distributed architecture is due to the geographical extent. The single measurements units are displaced according to the monitoring map along the area. The size of the area to be monitored requires a complex and wide measurement network, so in order to reduce the computational burden of each sensing unit and to manage easily the network, an algorithm performs a geographical partitioning of the area. In this way the region is divided in several small local zones. The algorithm allows to make an efficient partition, in order to guarantee a suitable size for each zone according to the topographical information on the area and the population density distribution. The information collected in the single zone provide a meaningful view of the electromagnetic field trend, and the correlation among different measurements in the area is assured. Such constraint is an important requirement for the data fusion stage. The number  $N$  of the zones depends on the sampling specifications, and zones size is not necessarily equal. According to the desired accuracy and resolution of monitoring, the algorithm designs a specific partition of the area and so a new network configuration. A major/minor severity level of the sampling plan is therefore cause of a major/minor partition. The user can modify and configure the partition by setting some parameters, like the distribution of population density and the topographical data. The map of the area is acquired, and a Cartesian axes system is selected. According to the dimensions of the area, the number of sensors and the available resources, the user chooses the initial width of the grid to be applied. A first partition is so made on the area map, and a preliminary sampling plan is shown for each local zone. At first, the algorithm does a simple subdivision of the area which is partitioned by zones with equal size. A high number of zones would allow a more accurate monitoring, but it is also cause of high costs. So the matter requires a compromise between the desired resolution, the available resources and the tolerable costs. The best solution is therefore to characterize the zones which need a higher attention level, in fact it is possible to single out the "sensible zones" where the partition grid has to be thickened. In details, the cost constraints limit the maximum number of possible zones, so an opportune criterion has to be considered. Surely zones with greater population density require a more accurate monitoring. In fact in presence of an alarm occurrence or of high exposure levels to electromagnetic field, the risk for the exposed population is more high considering the possible impact on a greater number of people. Therefore the algorithm reduces the size of partition in the zones characterized by a high population density or by the presence of sensible targets like schools, orphanages, hospitals or known electromagnetic pollution sources. As a result, such zones are monitored with

more care. So the user provides the function of the population density  $pd(x,y)$  of the area, or specifies the site of the sensible targets or otherwise a generic zone of interest. Then automatically, the algorithm thickens the partition grid of the sensible zones, such zones are divided in further four sub-zones with equal size (see Fig. 1).

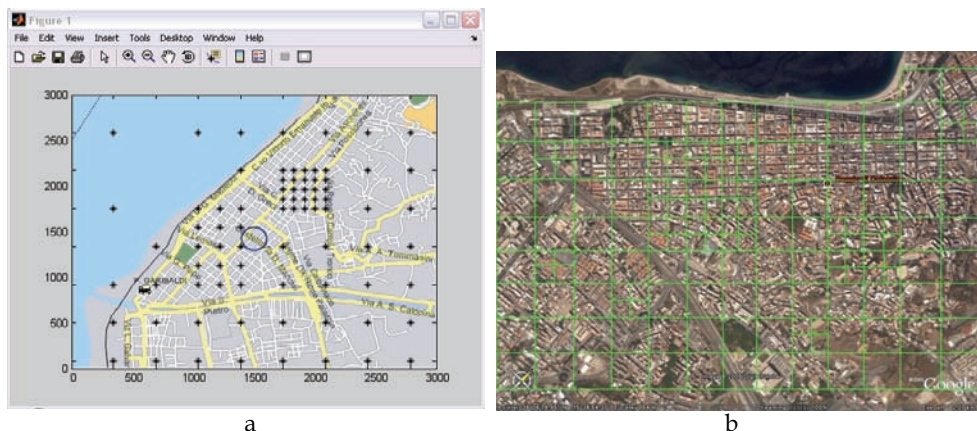


Fig. 1. (a) Partition tool. (b) Partition of the monitored area

According to topographical information about the area and specifications, the best configuration of the sensor network is got. The final partition allows to characterize the sampling map for each zone, while specific sampling plans are projected for the sub-zones in order to perform a more detailed monitoring with greater accuracy and resolution. Each sensor has to monitor a specific local zone with reference to a particular sampling plan. The sampling time, the refresh time of the monitoring map, the thickness of the grid, the number of samples and their locations depend on the desired confidence and severity level, on the tolerable costs, on the available resources and on the tolerable reliability for the monitoring. Usually a homogeneous sample is obtained by placing the monitoring points uniformly along the zone, but not always topographical data are compatible with this choice, so an accurate plan is necessary to assure a suitable reliability of collected information. The configurability features of the network allow so to adapt the sampling plans and the network configuration to environmental and topographical data. In this way the single zone is characterized by a sample of electromagnetic field levels measured in specific points, the representativeness of the sample is guaranteed by an experimental design. According to the local monitoring plan, each sensor moves along the associated zone and collects environmental data in the fixed points of interest. In this way the mobile architecture allows the measurement system to monitor a wide area with a reduced number of sensors. Network configuration, monitoring maps, number and size of the zones can be updated dynamically if the sampling specifications change, so a new partition is performed and new sensors may be added. The network security is guaranteed by a password access, thus only the authorized administrator can get access and manage the whole network and its configuration.

## 2.2 The wireless and smart web-sensors

If the electromagnetic field levels overcome the warning limits fixed by law, timely corrective actions are required in order to reduce possible risks for the population health. In

these circumstances only a real time monitoring process of the area allows to characterize such alarm events. So the matter demands the remote control of the used measurement instrumentation and a prompt data processing. The designed distributed network is composed of several wireless and smart web-sensors projected according to the guidelines of the *IEEE 1451 Standard*, (De Capua & Morello et al., 2004b; De Capua & Morello et al., 2005c). Each measurement system is an isotropic sensor sensitive to the electromagnetic field, it is equipped with a *GPRS modem* for wireless communication, and a *GPS module* for its localization. Three transducers, displaced orthogonally along the three axes of a Cartesian system by a plexiglas support, constitute the sensing unit. The single output is a  $\mu\text{V}\sim\text{mV}$  voltage proportional to the electromagnetic field applied perpendicularly to the chip surface. A suitable circuit has been projected in order to condition and amplify the voltage signals by an embedded auto-scale setting with noise compensation (see Fig. 2).

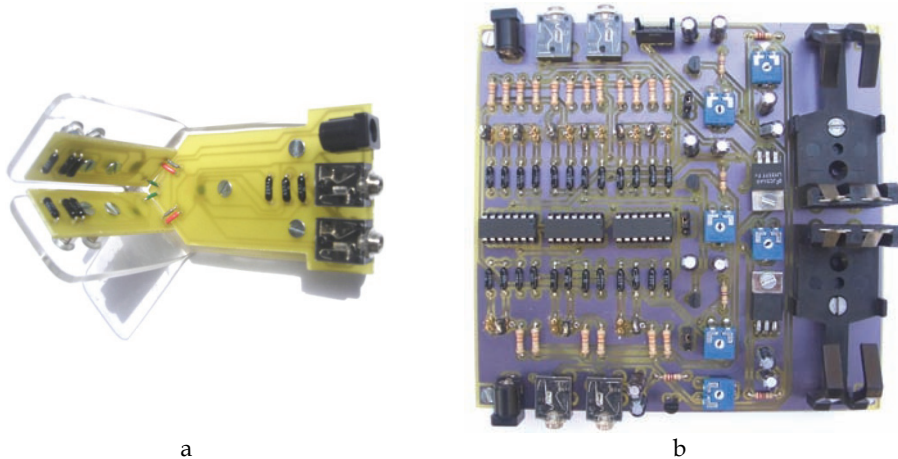


Fig. 2. (a) Electromagnetic Field Sensor. (b) Signal conditioning circuit

The system is able to configure automatically itself so to select the smaller available scale, in this way the smart sensor optimizes the accuracy and the resolution of measurement, reducing the measurement uncertainty of collected data. A *16 bits A/D converter* has been used to digitalize the three signals in order to make possible the communication between the several functional blocks of the sensor. A microcontroller architecture manages the exchange of data between the transducers, the *GPRS modem*, the *GPS module* and the internal memories. A *LCD display* shows the measured level of electromagnetic pollution. A *flash memory* stores metrological information about the sensor concerning its operating status, measurement uncertainty, calibration and reliability curves, next calibration interval, probability of faulty functioning, the actual sensor reliability, and the probabilities of erroneous decision relating to the decision making process. Such information is updated at the successive sensor calibration because of the inevitable decline of performances with time affecting any measurement system. It represents an important and essential knowledge for the data processing stage, in fact the data fusion procedure uses such information in order to optimize data and credibility of the results for a more accurate, complete and fault tolerant computing (for more details see Paragraph 3). Measured data are stored in a *Secure Digital (SD) Memory*. Then each sensor sends data (electromagnetic field levels, *GPS* position and

metrological information) to its own *Web Page* in *Active Server Pages (ASP)* format. In this way the single *Web Page* contains information about the pollution status of the associated zone, the sensor history and its operating state. Such data are accessible by password, only authorized users can read data and information for processing and statistical purposes. The *Server* acquires and processes data by the sensor data fusion procedure in order to test the electromagnetic field compliance with the exposure limits, (see Fig. 3).

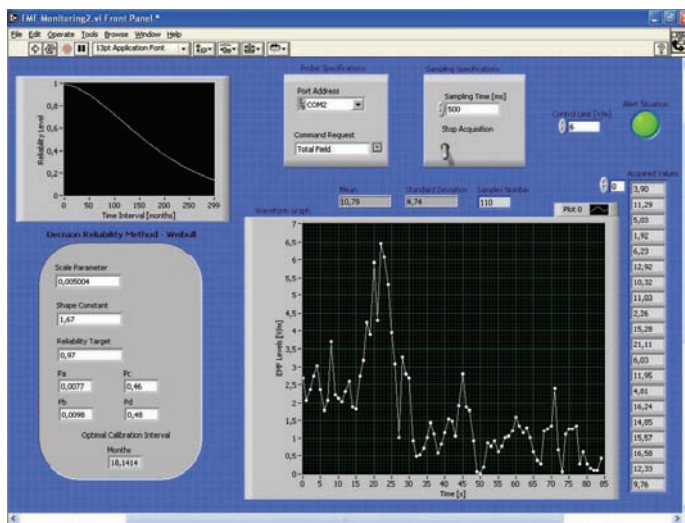


Fig. 3. Control panel of data acquisition and processing

Merely a free *WEB Page* is accessible to general public, it contains final reports and computing results on the pollution status of the monitored area. Wireless communication has been made possible by a programmable *GPRS* modem with *M2M* technology. Queries are performed by *AT commands*. In this way the administrator of network can get direct access to the single sensor by a *PIN number*. He may ask for specific measures or exchange information about the desired sampling specifications (sampling time and monitoring map).

### 3. The sensor data fusion approach

#### 3.1 Sensor reliability

Any measurement result is characterized by the estimated value of the measurand and the associated measurement uncertainty. The latter depends strongly on the metrological state of the used measurement system. Measurement uncertainty is a parameter characterizing the dispersion of the quantity values being attributed to the measurand by a measurement process, (IEC-ISO GUM, 1995; ISO/TS 17450-2, 2002). Typically the uncertainty of a generic measurement system changes inevitably with time. Therefore in order to guarantee reliable measures, the operating state of the used equipment and measurement systems has to be checked periodically by calibration, (ANSI/NCSL Z540-1, 1994; ISO/IEC 17025, 1999; UNI EN ISO 10012-1, 1993; UNI EN ISO 30012-1, 1993; UNI EN ISO 10012-2, 2001). In this view in the present paragraph the authors propose an original procedure being able to estimate the reliability and the performances of the sensors of the network, such an approach allows to

qualify and maximize the measured data. Clearly data with higher quality and accuracy are indispensable for a reliable processing. For that reason the developed procedure provides information about the credibility of the measured data and the actual reliability level of each sensor. Such knowledge is then used in order to characterize the integrity of the whole network and of the computing results. Thus if the reliability of the network or of the single sensor decreases below a fixed tolerable level, the sensor/sensors have to be calibrated. A proper maintenance of the used measurement instrumentation is an important prerequisite in order to guarantee the credibility of sensor data fusion. In details, by a Statistical Model the procedure estimates optimal calibration intervals for the single sensor. The purpose is to reduce the probability of an out-of-tolerance state during the maintenance time. So according to a desired reliability target, the next calibration time is estimated, assuring a suitable operating state for the sensor. About that matter, numerous models have been proposed by several authors, (Castrup & Johnson, 1994; Nunzi et al., 2004; Wyatt & Castrup, 1991). The present original approach bases oneself on evaluating the impact of the uncertainty associated with the calibration process on the estimation of the maintenance interval, (De Capua & Morello et al., 2005b). An erroneous analysis of the calibration results could be cause of unsuitable calibration intervals. Consequently it may be the reason for possible out-of-tolerance states during the maintenance time, so unreliable data would be processed. The basic problem concerns the decision, made during a calibration process, about the real operating state of a sensor. The used approach allows to characterize the decision reliability by the erroneous decision probabilities about the real state of the tested sensor. The estimated decision risk is subsequently used in order to optimize the reliability function of the sensor concerning its in-tolerance state. A sensor is in a tolerance operating state if its metrological characteristics are compliant with fixed tolerance limits in order to guarantee a tolerable measurement uncertainty. According to information on the last calibration process of the sensor, the Model estimates the erroneous decision probability  $\Pi_\beta$  concerning the occurrence to have taken a wrong decision about its in-tolerance state. In other words  $\Pi_\beta$  represents the probability to have supposed erroneously in the last calibration that the sensor was in an in-tolerance state, and so in accordance with the tolerance limits, because of the uncertainty associated with the calibration process, whereas in truth it was in an out-of-tolerance state, and so not compliant with the tolerance limits. If  $\delta$  is the tolerance limit of the tested parameter;  $x_m$  is the random variable representing the tested parameter;  $x$  is the random variable of the calibration process, its distribution around the expected parameter value represents the randomness of the measurement result, in other words its standard deviation is the standard uncertainty due to calibration process. We can assert that the tested parameter or the considered metrological characteristic is compliant with the tolerance limit, and consequently the sensor is in a tolerance state, if  $x_m \in [0, \delta]$ . So the erroneous decision probability associated with the sensor calibration is estimated by the equation:

$$\Pi_\beta = \int_{x_m \in [0, \delta]} F_x(\delta - x') \cdot f_{x_m}(x') dx' \quad (1)$$

Where  $F_x(-)$  is the Cumulative Distribution Function (CDF) associated with the variable  $x$ , while  $f_{x_m}(-)$  is the Probability Density Function (PDF) of the variable  $x_m$ .

The next step requires the definition of the sensor reliability curve  $R(t)$ , it can be estimated by the following expression based on the Weibull Model:

$$R(t) = R_o * e^{-(\lambda t)^\beta} \tag{2}$$

That function allows to evaluate the in-tolerance probability of the sensor in a precise moment  $t$  according to a desired quality target. In order to optimize the estimation of the reliability function, the parameter  $R_o$  has been set equal to  $1-I_{\beta}$ , in this way the erroneous decision probability weighs the information about the reliability level concerning the operating state of the sensor. In fact high probability values of erroneous decision are cause of low reliability levels for the sensor, because of a high probability to have supposed erroneously in the last calibration that the sensor was in an in-tolerance state. The parameters  $\lambda$  and  $\beta$  in the equation (2) are evaluated by a maximum likelihood estimation, in order to characterize the best fit function according to past information on previous calibrations of the considered sensor. In other words, the two parameters are settled in order to guarantee the best fit with the data history of the previous calibrations. In fact the reliability function can be sampled by historical time series of previous calibration data, by means of the ratio between the number of in-tolerance operating states and the total number of calibrations in a definite time  $t$ . In the Fig. 4 (a) the trend of reliability function is shown for different values of the two parameters. The Fig. 4 (b) shows the reliability curve estimated for a sensor of the network starting from data of six previous calibrations.

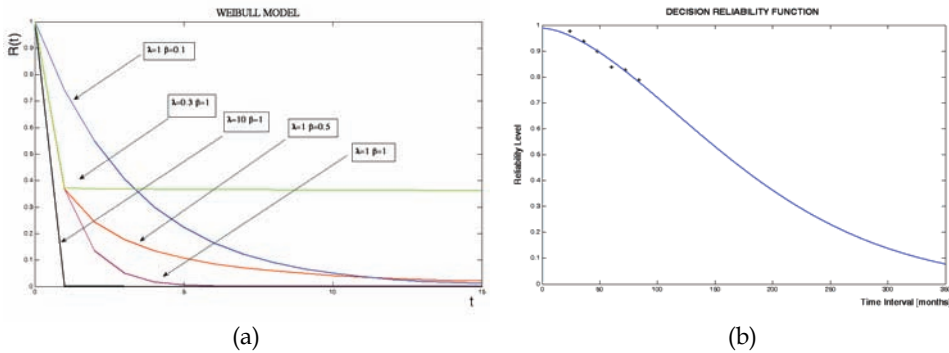


Fig. 4. (a) Weibull Model. (b) Sensor reliability curve

The reliability function allows to get information about the temporal evolution of the sensor performances. In this way it is possible to know how its measurement reliability decreases over time. So a low reliability level is sign of a sensor which may be faulty because of out-of-tolerance occurrences; differently a high reliability level is sign of a sensor with an appropriate in-tolerance state, and therefore the measured data are reliable. The reliability curve of each sensor is stored in its own internal *flash memory* and subsequently in the relative *Web Page*. In addition the procedure allows to estimate the next calibration interval  $t^*$  by fixing a desired maximum tolerable reliability  $R^*$  for the single sensor:

$$t^* = \frac{1}{\lambda} \beta \sqrt{\ln\left(\frac{R_o}{R^*}\right)} \tag{3}$$

In this way during the maintenance interval  $t^*$ , the sensor reliability  $R(t)$  will not decrease below the target level  $R^*$ , so to guarantee a suitable in-tolerance state. The described procedure has therefore two benefits: first it provides a suitable maintenance interval for the

single sensor in order to assure an appropriate operating state and therefore reliable data for the following fusion stage; finally the reliability function permits to evaluate the actual reliability of each sensor  $R_i(t')$  in a precise moment  $t'$ , such information is used during the processing stage in order to assure a fault tolerant fusion of data. In this way each zone is characterized by a value  $R_i(t')$  which is representative of the data reliability collected in the zone and consequently it gives a measure of the credibility of the fusion results.

### 3.2 The two-levels data fusion procedure

After the monitoring process each zone is characterized by a set of electromagnetic field levels collected according to the sampling plans. But information on the sensor reliability and on measurement uncertainty allows in addition to improve the available knowledge minimizing errors during the computation. Now data from different sensors have to be merged in order to optimize the information about possible alarm occurrences regarding pollution status of the monitored area. Sensor data fusion is the best solution to manage data and correlated knowledge from different sources which may be faulty. Moreover the sensors are distributed on a wide area to monitor a particularly complex phenomenon with temporal and spatial evolutions; besides a wireless sensor network is more complicate than a wired network because of communication errors, delay and topographical constraints. So each sensor has only a restricted view of the monitored phenomenon limited to the own local zone. Data fusion solves the limitation of the single source, and it is indispensable when a wide distributed network has to be used in order to get a more accurate view of the process behaviour in the whole area, (Linn et al., 1991; Wu et al., 2002). It allows a more efficient interpretation of data reducing the initial amount with less uncertainty and error than that obtained from a single source. The complexity of the matter needs not only to analyze and characterize these features, but also a careful analysis and integration of correlated information are necessary. The proposed approach consists in a two-levels data fusion, (De Capua & Morello et al., 2005a; De Capua & Morello et al., 2007). The procedure is able to improve the accuracy of data and to detect possible faulty states of the sensors. First for each zone, a fuzzy algorithm provides local decisions regarding alert occurrences due to the overcoming of the law limit, (De Capua & Morello et al., 2004a). In this way the initial amount of data is reduced by a preliminary fusion, and information on the measurement uncertainty is used to qualify the decision making process reducing errors and risks. Subsequently a final fusion of data and of correlated knowledge provides global information about the behaviour of electromagnetic field in the area so to characterize the general pollution status. In detail, after the partition of the area, each sensor must acquire a sample of  $N_i$  measures along the own local zone. Starting from the collected data and information on sensors performances, the *Server* processes electromagnetic field levels in order to single out possible hazardous events in the monitored area. For each measured value, the conformity with the exposure limit has to be evaluated. In fact in presence of warning situations, corrective interventions must be undertaken so to safeguard the population health. The problem requires suitable decision making rules in order to take reliable decisions. The matter concerns the choice of the more plausible decision about the conformity of measured data with the exposure limit. The uncertainty, which affects the measures, could be cause of possible wrong decisions, so a measured value may seem erroneously over the fixed limit, (Carbone et al., 2002; Castrup, 1995; Catelani et al., 1998; Zingales, 1996). Consequently the comparison between measured value and exposure limit

cannot be performed by a simple mathematical comparison. Qualified procedures are necessary to take account of costs and risks associated to erroneous decisions. It is possible to use the same data redundancy in order to improve the results accuracy, and correlated information as environmental and topographical knowledge may qualify the computation. The fuzzy algorithm is able to perform a decision making process on the acquired data. Each zone is characterized by a sample of  $N_i$  measures, which are representative of the electromagnetic field behavior according to the desired severity level of the monitoring. The procedure is performed in the single zone, to determine if an acquired measurement value is conformance or non-conformance with the specification limit. So starting from the statistical distribution correlated with the monitored process and the metrological characteristics of the measurement system, the algorithm takes a decision about the overcoming occurrence of the exposure limit for the single measured datum in the zone. The possible alternatives of decision are two:  $A_1$  indicates the conformance occurrence, that is the measured level is below the limit; while  $A_2$  indicates the non-conformance occurrence, in other words the level is beyond the specification limit. The information stored in the *flash memory* of the sensor, regarding its metrological operating status, is used in order to estimate the erroneous decision probabilities associated with the measurement process. So data are fused with information regarding the statistical distribution of process and the measurement uncertainty. Starting from the  $N_i$  measured values, the Statistical Model described in the previous Paragraph allows now to calculate the probability  $P_\alpha$ . It represents the probability to have supposed erroneously the measured value being above the exposure limit, because of the associated measurement uncertainty, when the measurand is really in conformity with the limit:

$$P_\alpha = \int_0^\delta [1 - F_x(\delta - x')] \cdot f_{x_m}(x') dx' \tag{4}$$

Similarly to the equation (1)  $\delta$  is the exposure limit;  $x_m$  is the random variable associated with the measured quantity; and  $x$  is the random variable associated with the measurement process, so its standard deviation represents the standard measurement uncertainty. With the same symbolism it is possible to define the erroneous decision probability  $P_\beta$ . Differently it represents the probability to have supposed erroneously the measured value being below the exposure limit, because of the associated measurement uncertainty, when the measurand is really in non-conformity with the limit:

$$P_\beta = \int_{x_m - [\delta, 0]} F_x(\delta - x') \cdot f_{x_m}(x') dx' \tag{5}$$

The reader has to notice that the equations (1) and (5) are equal, but the meaning of the expression is different. Because in the first equation the probability concerns the calibration process and so the decision regards the tolerance state of sensor. The second equation estimates the probability concerning the measurement process of the electromagnetic field levels and so the respective decision refers to the overcoming of the exposure limit. In the same way, the variables  $\delta$ ,  $x_m$  and  $x$  have a different connotation in the two equations being associated with the calibration and the measurement process respectively. The Fig. 5 shows, in example, the trend of the function  $P_\alpha(u, \delta)$ , where  $\sigma_y$  represents the standard deviation of the measured data. The function characterizes the relation between the erroneous decision probability and the measurement uncertainty  $u$  and the exposure limit  $\delta$ .



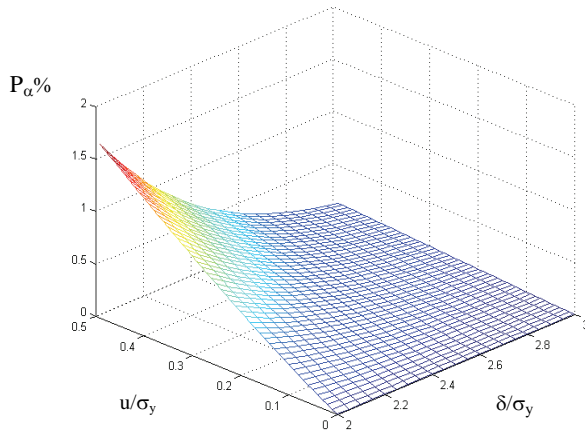


Fig. 5. Trend of  $P_\alpha(u, \delta)$

Increasing values of  $u$  are cause of more and more high values of the erroneous decision probability; while a restrictive limit  $\delta$  lead to a higher error probability. In order to explain the developed fuzzy decision making algorithm, we must start from the guidelines of the Standard EN ISO 14253-1, (ISO EN 14253-1, 1998). According to the Standard decisional rule it is possible to single out three zones when the measured value is put in comparison with the exposure limit: the conformance and non-conformance zones and the uncertainty range. So, if  $U$  is the expanded uncertainty, the measured value belongs to the conformance zone if it falls in the left interval  $[0 \delta-U]$ ; the right interval  $[\delta+U +\infty]$  represents the non-conformance zone; while the middle interval  $[\delta-U \delta+U]$  is the uncertainty range. The Fig. 6 shows the fuzzy decisional criterion developed according to the guidelines of the ISO Standard rule.

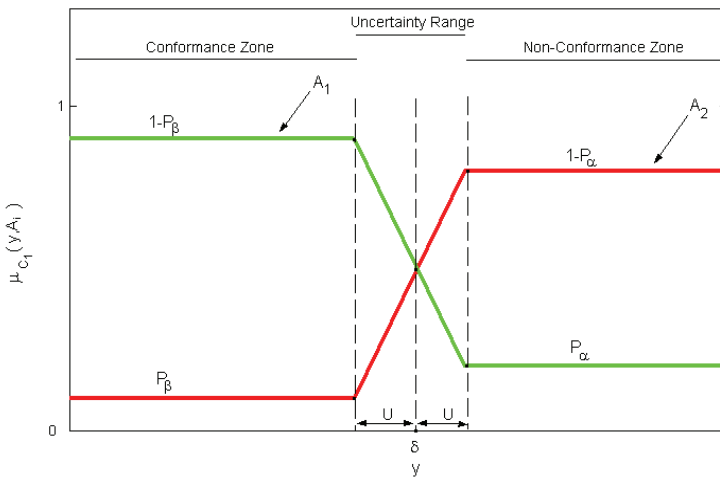


Fig. 6. Fuzzy decisional rule

Now in order to understand the criterion, it is necessary to analyze the previous ISO Standard. The rule of the ISO Standard is a simple decisional criterion: if the measured value belongs to the conformance or non-conformance zone then the measurand has to be considered respectively compliant or not compliant with the limit. Several gaps make such decisional rule not optimal. In the first instance, it is possible to point out that no decisional criterion is suggested for the uncertainty range, so if the measured value belongs to the interval  $[\delta-U \ \delta+U]$ , the ISO criterion does not provide any reliable decision. In the second instance, the decisional rule is lacking in information about the reliability or consistency of decision, so no knowledge is given about the decision goodness. It is possible to interpret such an approach as a binary one, so if the measured value belongs to the conformance zone, we could assign a credibility level equal to 1 to the conformity alternative  $A_1$ , while the non-conformity alternative  $A_2$  has a null credibility level; in fact according to the ISO Standard in this case the 'conformity' decision has to be taken. By a first analysis that choice is not always the most reliable, because the comparison does not take into account the measurement uncertainty influence. So also the 'non-conformity' alternative could be plausible with a credibility level being not null. In this view, a fuzzy approach may improve the decisional rule when data affected by uncertainty are available, (Sousa & Kaymak, 2001; Triantaphyllou & Chi-Tun, 1996). The proposed fuzzy algorithm improves the ISO Standard rule by fusing information about the measurement uncertainty and the erroneous decision probabilities. In fact the previous probability  $P_\beta$  allows to estimate the probability to have supposed erroneously the measured value in conformity with the exposure limit; since this probability is not necessarily null, it is possible to assert by a fuzzy approach that also the non-conformity alternative may be plausible. As a result if the measured value belongs to the conformance zone we can assign a credibility level equal to  $1-P_\beta$  to the alternative  $A_1$ , while the credibility level of the alternative  $A_2$  is not exactly equal to zero but equal to  $P_\beta$ , see Fig. 6. In this way the decisional criterion weighs up the occurrence of a possible wrong decision due to the measurement uncertainty. With the same reasoning, we can observe that the probability  $P_\alpha$  to have supposed erroneously the measured value in non-conformity with the limit, is not necessarily null. So if the measured value belongs to the non-conformance zone, it is reasonable to assign a credibility level equal to  $1-P_\alpha$  to the alternative  $A_2$ , and  $P_\alpha$  to the alternative  $A_1$ . The Fig. 6 shows the membership functions of the decisional criterion. For each decision alternative along the x-axis is represented the measured value, whereas the y-axis refers to the credibility assignment. In this way for each alternative, the function assigns a credibility level according to the measured value. The probabilities  $P_\alpha$  and  $P_\beta$  weigh the decisional criterion in order to guarantee the decision consistency. Therefore the credibility level depends on the erroneous decision probabilities and so on the measurement uncertainty. A second decisional criterion takes in account risks and costs associated with the available alternatives, so to improve the discernment properties of the algorithm. Practical visual tools provide information about the confidence level of decision, generating a graphical representation of the global reliability of the two possible alternatives  $A_1$  and  $A_2$ . Starting from the credibility levels, two fuzzy triangular sets show the global satisfaction of the decisional rules for the two alternatives (see Fig. 7). Each alternative is then described by a triangular set, which shows the alternative reliability. The single triangle is got by fixing the x-axis of the vertex equal to the previous credibility levels in Fig. 6 and its y-axis equal to 1. In this way the credibility levels of the membership functions are characterized by the most high level of reliability. By a fuzzy approach also the

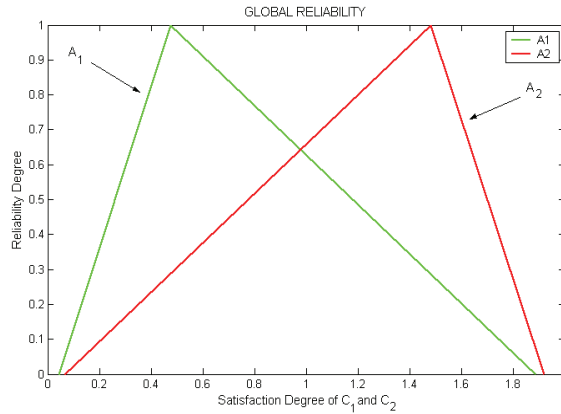


Fig. 7. Global reliability of the alternatives

near credibility levels are possible with a lower reliability degree, and it is visible by the decreasing trend of the triangle sides. A rapid visual analysis of the Fig. 7 allows to single out the best reliable alternative, in other words the alternative characterized by highest credibility levels. In the example reported in Fig. 7, the best alternative is  $A_2$ , because its triangle is on the right side with highest credibility levels. So the measured value has to be assumed non-compliant with the exposure limit. A quality index provides information on the decision reliability or decision consistency  $DC$ :

$$DC = \left( \frac{(p_{mi})_{\max} - (p_{mi})_{\min}}{2} \right) * (1 - ((p_{ri})_{\max} - (p_{mi})_{\max})) * 100 \tag{6}$$

where:

- i.  $(p_{mi})_{\max}$  is the vertex abscissa of the triangle associated with the chosen alternative;
- ii.  $(p_{mi})_{\min}$  is the vertex abscissa of the triangle associated with the rejected alternative;
- iii.  $(p_{ri})_{\max}$  is the abscissa of the right base value of the triangle associated with the chosen alternative.

The proposed fuzzy decision making algorithm allows to perform a first level of data fusion. So starting from the data collected in the single zone, each measured value is put in comparison with the exposure limit. The most reliable decision is taken about the conformity or non-conformity with the limit. The available information stored in the *flash memory* of each sensor is so used in order to minimize risks of possible decisional errors. The further knowledge on the reliability of sensor and the next calibration interval represents an useful information in order to evaluate the actual performances of the sensor and its operating status. The same computing is executed in each zone of the investigated area. This allows the data reduction in order to manage the computational load. In this way the  $i$ -th local zone is merely characterized by  $N_i$  decision results. In order to evaluate the pollution status of the single zone, the frequency of non-conformities  $f_i$  is estimated:

$$f_i = N_{i0} / N_i \tag{7}$$

where  $N_{i0}$  is the number of values which overcome the exposure limit, and  $N_i$  is the total size of the sample. In Fig. 8 the warning report of the local zones is shown. The report depicts the

pollution status in the area according to the made partition. The graduate color scale shows the different levels of pollution severity. The green color characterizes the zones with electromagnetic field levels in conformity with the exposure limit. The yellow color underlines the zones where warning situations are happened. The red color indicates alarm/alert events which may be cause of possible risks for the exposed population. The analysis of the pollution status in the area is an useful tool in order to single out the zones in which warning occurrences have happened. In this way it is possible to plan local corrective interventions, so to safeguard the involved population. Such information is not yet functional to characterize the risk state of the whole area, so an additional level of fusion is required in order to merge further information on topographical data and sensor reliability.

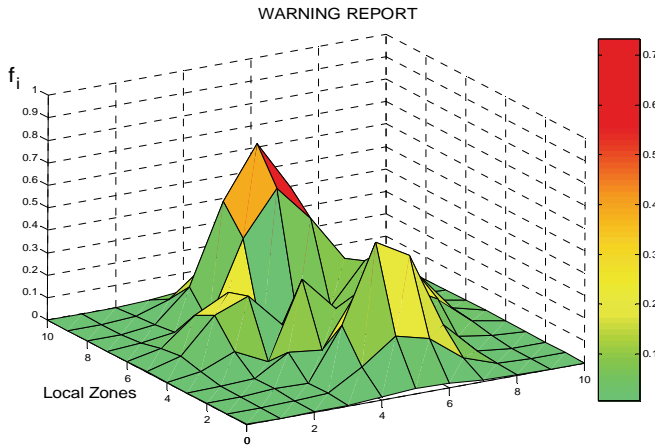


Fig. 8. Pollution status of the zones

The second level of data fusion has the task to fuse the obtained knowledge in order to get a view of the global pollution status of the area, so to plan general actions on the area characterizing the zones with higher priority of intervention. So information concerning the results of the decision making process in the single zone, the sensor reliability and topographical data are fused. The final result has the objective to improve the available knowledge turning the attention no more to the single zone but to the whole area. In this way the local results of each zone are weighed by the information concerning the reliability of the respective sensor. It is reasonable to observe that in presence of an alarm status, the zones, which have a higher priority of intervention, are those characterized by a higher reliability degree and a higher population density. So the zones with a higher priority of intervention may be singled out by means of the relative population density  $\sigma_{ri}$ :

$$\sigma_{ri} = \sigma_{li} / \sigma_t \tag{8}$$

where  $\sigma_i$  is the population density of the  $i$ -th zone, whereas  $\sigma_t$  is the total population density of the whole area. So high levels of  $\sigma_{ri}$  point out zones with high intervention priority, because, in presence of alarm situations, the impact on the population would be more disastrous. By the equation (2), it is possible to estimate the reliability  $R_i(t')$  of each sensor in a specific instant  $t'$ . This parameter represents a practical information about the consistency of the data fusion in the first level processing. So it can be used in order to get an indication

about the possibility of false alarm in the  $i$ -th zone due to a faulty functioning of the sensor. Now we can define a Loss Function  $L_i$  for the single zone by the following expression:

$$L_i = 100 * (f_i * R_i)^2 * (\sigma_{ri})^2 \tag{9}$$

This parameter is a percentage measure of the cost of non-quality, it shows how the non-quality of the monitored process is high, and so how the process is in an out-control situation. The Fig. 9 shows the trend of the Loss Function over the relative population density  $\sigma_{ri}$  and the reliability of sensor  $R_i$  weighed by the frequency of non-conformities  $f_i$ . The function represents an effective parameter evaluating the risk state of the  $i$ -th zone. A graduated color scale shows the risk level for the zone, where the red color characterizes the zones with higher risk.

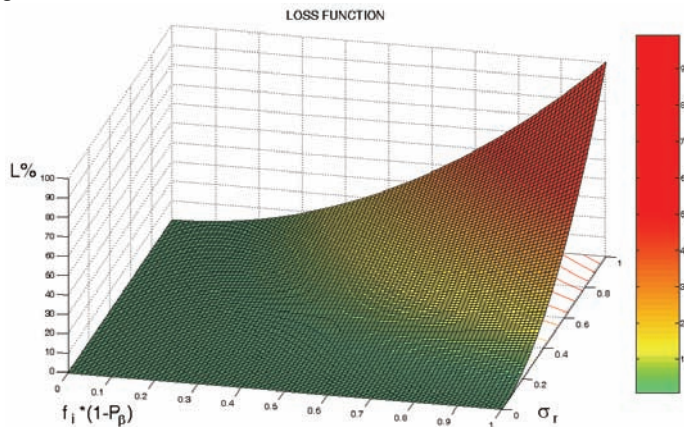


Fig. 9. Risk state of the  $i$ -th zone

The aim of that parameter is to characterize the risk of the zones in the whole area. So, assuming  $f_i$  to be constant, zones with high population density are characterized by high loss of quality. While the zones, which have a high probability of false alarm occurrences, have less weight in the definition of intervention plans. Finally a global index  $F$  provides information about the average frequency of the alarm occurrences in the  $N$  zones of the area:

$$F = \frac{\sum_{i=1}^N f_i * R_i}{N} \tag{10}$$

By a weighed sum of the non-conformities frequencies  $f_i$ , which are weighed by the respective sensor reliability  $R_i$ , the index provides a measure of the global alarm occurrence in the area. The final reports and computing results are stored in free-accessible *Web Pages*, so ordinary users can get knowledge about the environmental pollution of the area.

In conclusion, the first level of data fusion is functional to get restricted information about the local zone and the conformity of the electromagnetic field with the exposure limit fixed by laws. So local plans of intervention can be designed. A report shows the pollution status of the single zone. The knowledge of the measurement uncertainty allows to qualify the processing stage, in order to take reliable decisions during the comparison between

measured values and exposure limit. The used fuzzy algorithm permits to minimize the possible occurrence of errors. The second level of fusion has the task to fuse the previous information with topographical data and performances of sensor. The zones are put in comparison so that an outline of the global electromagnetic field behavior is got. The purpose is to design a global plan of intervention. In this way the zones with higher priority of intervention are singled out, and information about the risk for population health is obtained. Indexes and parameters provide knowledge about the reliability of the computing results and the risk state of each zone. Correlated information is used in order to guarantee a fault tolerant data fusion, so that zones with lower reliability levels, or in other words zones with a greater possibility of false alarm occurrences have less weight in the computing.

#### 4. Experimental results

The sensor data fusion procedure and the network of wireless and smart web-sensors have been tested in order to verify the consistency of the developed models. Experimental results have been obtained by a monitoring process carried out in a wide urban centre (Reggio Calabria city, in the south of Italy). Starting from the map of the area, the partition algorithm has subdivided the area in several local zones. According to the available resources, the topographical data and the desired accuracy for the monitoring, 30 local zones have been singled out. The initial partition grid has been thickened in two specific zones, because of the presence of sensible targets (schools and hospitals). So 8 new sensible zones have been got with a smaller size, they have required a more accurate monitoring, (see Fig. 10).



Fig. 10. Partition of the monitored area

In these zones the monitoring maps have been designed with more attention. The sampling plans for the remaining zones have been realized by an experimental design in order to optimize the choice of the specifications. In details, information on the severity level of monitoring, the desired accuracy and the population density distribution have been used in order to get for each zone the single monitoring map, the sampling frequency and the sample size. In this way the single zone has been characterized by a representative sample of electromagnetic field levels. The monitoring of the several zones has been executed in different temporal windows. In fact because of the limited number of the available

prototypes of the smart sensors, the same sensors have been used for monitoring different zones, though the sensor network design would expect the presence of a specific sensor for each zone. However that may be, that choice does not compromise the consistency of the monitoring. The Fig. 11 (a) shows the moment of data acquisition. The *Server* acquires for each zone the set of the measured values and information about the operating state of the sensor. In Fig. 11 (b) the data trend of a specific zone is shown.

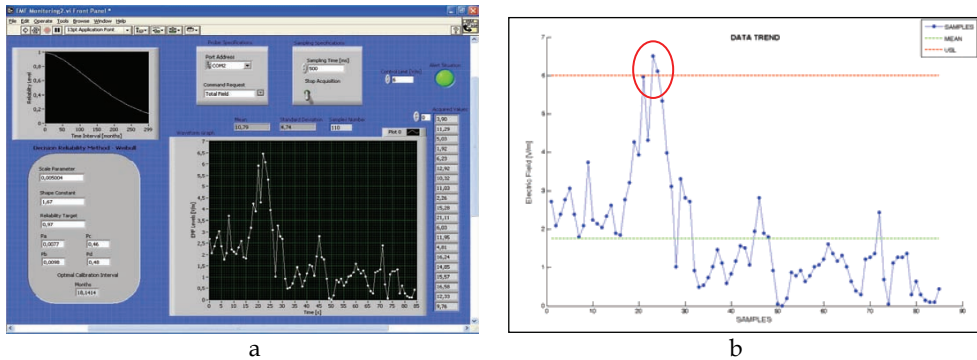


Fig. 11. (a) Data acquisition. (b) Electromagnetic field trend in the *i*-th zone

By a first analysis of the figure, it is possible to observe that some values (highlighted by a red oval line) would seem to overcome the exposure limit depicted by a dash red line. After the data acquisition, the *Server* runs the sensor data fusion procedure in order to compute data and information, (see Fig. 12).

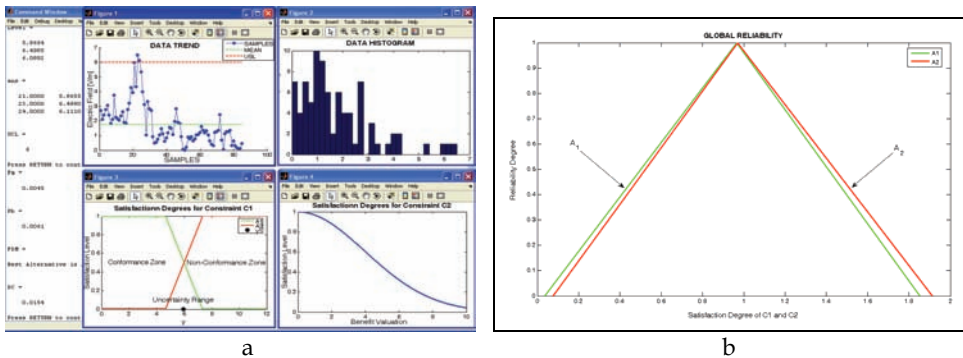


Fig. 12. (a) Processing stage. (b) Fuzzy triangular sets of the alternatives

In the Fig. 12, it is possible to see the several processing steps of the fuzzy decision making algorithm, with an example of fuzzy triangular sets for a specific measured value. In detail, considering the zone *n*°4 as an example, the measured electromagnetic field level  $y=5.96$  V/m would seem to not overcome the exposure limit  $\delta=6$  V/m, as it is shown in Fig. 11 (b). The estimated erroneous decision probabilities are  $P_{\alpha}=0.0045$  and  $P_{\beta}=0.0061$ . By observing the triangular sets, the decision making procedure has computed that the measured value overcomes the exposure limit with a decision consistency  $DC=1.5\%$ . In the same way, the procedure has been performed in each zone for all measured values. So each zone has been

characterized by  $N_i$  decision results. By calculating the frequency of non-conformities  $f_i$  for the single zone, the following report concerning the pollution status of the area has been got, (see Fig. 13).

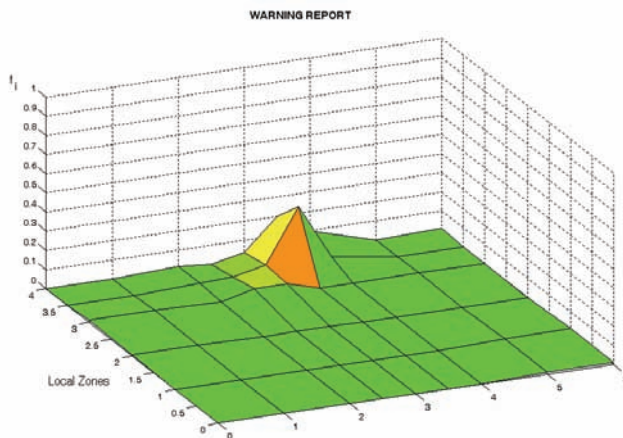


Fig. 13. Pollution status of the zones

By a simple analysis of the report it has been possible to detect the presence of three zones which require more attention. In fact the figure shows the presence of three zones with a different colouring; such zones are also highlighted in the Fig. 10 by a red contour. In the Table 1, the experimental results relating to the second level of data fusion are shown, in detail, as an example, the data of the previous three sensible zones are reported.

Zone #	Frequency of non- conformities $f_i \in [0 1]$	Sensor Reliability $R_i \in [0 1]$	Risk State $L_i\%$
...	...	...	...
7	0.2	0.9965	0.00089%
8	0.1	0.9961	0.00022%
9	0.34	0.996	0.0026%
...	...	...	...

Table 1. Experimental results

These zones are characterized by frequencies of non-conformities  $f_i$  above average, because of warning occurrences. The matter has required a more detailed monitoring in order to perform a more careful analysis. The investigation has made possible the identification of the sources of pollution in the zones. In fact the high exposure levels were caused mainly by antennas of radio-mobile service located in the vicinity. Only for such zones the monitoring campaign has therefore highlighted the necessity of local corrective actions in order to reduce the emission power of the sources, so to conform the electromagnetic field levels to the law's exposure limit. The average frequency of alarm occurrences  $F=0.03$  is rather low similarly to the risk state levels of the zones, so no global plan of intervention in the area is necessary but only local ones for the three examined zones. Finally, the estimated sensor reliability levels point out that the sensors are in an optimal operating state, therefore the probability of a faulty functioning and of false alarm is very low, assuring a suitable reliability for the results of the data fusion process.



## 5. Conclusion

In this Chapter an original approach to sensor data fusion for environmental monitoring applications has been proposed. Care has been paid for an innovative design of the distributed network of wireless and smart web-sensors with remote data processing. The architecture of the network is dynamically configurable by an algorithm according to the requirements and the accuracy desired for the monitoring. The sensors are able to measure the environmental electromagnetic field. A *GPRS* modem and a *GPS* module allow the single measuring unit to communicate remotely and to acquire information about its geographic location. In order to distribute the burden of measurement among several sensors, the monitored area (an urban centre) is divided in several local zones, where a set of sensors acquire a fixed number of data according to designed monitoring maps. Sampling time and location of the measurement points depend on topographical and environmental knowledge. The collected data are sent to *ASP Web Pages* accessible in reading only from identified users. The alone administrator has authorization to manage the whole network and to exchange information and commands with the sensors. Data and correlated information are remotely processed by an innovative data fusion procedure. The data fusion approach represents a suitable solution in order to manage a wide network and to process data from different sources. The developed procedure allows to minimize errors and faulty computing, by using information about the measurement uncertainty and the performances of the sensors. The data amount is processed and the same redundancy is used in order to increase the reliability and the accuracy of the results. A first level of data fusion provides information on alarm occurrences. So measured values are put in comparison with the exposure limit. The fuzzy decision making algorithm permits to qualify the comparison process minimizing possible occurrences of wrong decisions. A quality index values the consistency of the final decision alternative chosen. The estimation of the erroneous decision probabilities and the measurement uncertainty improve the computing results. A warning report shows the pollution status of the zones. Successively the results of the decisional process, information on sensor reliability and population density distribution are fused so to obtain a global view on the risk state in the whole area. The procedure is fault tolerant and permits the maximization of the useful information. In this way a more reliable result is got than that obtained from the single sensor, so greater efficacy and efficiency are achieved. The projected measurement process is a complete solution in support of monitoring, controlling and alarm reporting applications for environmental purposes.

## 6. References

- Ahlbom, A.; Bergqvist, U.; Bernhardt, JH.; Cesarini, JP. & al., (ICNIRP) (1998). Guidelines for Limiting Exposure to Time-Varying Electric, Magnetic, and Electromagnetic Fields (up to 300 GHz), *Health Physics*, Vol. 74, No. 4, pp. 494-522.
- ANSI/NCSL Z540-1, (1994). Calibration Laboratories and Measuring and Test Equipment-General Requirements.
- Bertocco, M.; Farias, M.; Offelli, C. & Sona, A. (2002). A Measurement System for the Evaluation of Environmental Electromagnetic Field, *Proceedings of 19<sup>th</sup> IEEE IMTC 2002*, pp. 1277-1281, Anchorage, AK, USA.

- Carbone, P.; Macii, D. & Petri, D. (2002). Management of Measurement Uncertainty for Effective Statistical Process Control, *Proceedings of 19<sup>th</sup> IEEE IMTC 2002*, Anchorage, AK, USA.
- Castrup, H. & Johnson, K.M. (1994). Techniques for Optimizing Calibration Intervals, *ASNE Test & Calibration Symposium*, December 1994, Arlington.
- Castrup, H. (1995). Uncertainty Analysis for Risk Management, *Proceedings on Measurement Science Conference*, January 1995, Anaheim, CA-USA.
- Catelani, M.; Iuculano, G. & Sartori, S. (1998). On the Quality of a Measurement Process: A Statistical Approach for the Variability Evaluation, *Proceedings of IEEE IMTC 1998*, pp.401-405, St.Paul.
- De Capua, C.; Morello, R. & De Falco, S. (2004a). A Fuzzy Decision Making Algorithm for Testing Conformance with Specifications, *Proceedings of XI Congress of SIGEF 2004*, ISBN 88-8296-146-X, November 2004, FALZEA Ed., Reggio Calabria-Messina, Italy.
- De Capua, C.; Morello, R. & Pasquino, N. (2004b). A Mobile Distributed Architecture for Measurement and Analysis of the Electromagnetic Field, *Proceedings of IEEE IMTC 2004*, Vol.3, pp. 2090-2094, ISBN 0780382498, May 2004, Como-Italy.
- De Capua, C.; Morello, R.; De Falco, S. & Liccardo, A. (2005a). A Sensor Data Fusion Approach for Remote Sensing and Processing of EM Field Levels with a Configurable Area Partitioning, *Proceedings of IEEE CIMSA 2005*, pp.14-18, ISBN: 0-7803-9026-3, July 2005, Giardini Naxos-Taormina, Italy.
- De Capua, C.; Morello, R.; De Falco, S. & Liccardo, A. (2005b). A Virtual Instrument for Estimation of Optimal Calibration Intervals by a Decision Reliability Approach, *Proceedings of IEEE VECIMS 2005*, pp.16-20, ISBN: 0-7803-9042-3, July 2005, Giardini Naxos-Taormina, Italy.
- De Capua, C.; Morello, R. & Liccardo, A. (2005c). Network of Smart Sensors for Remote Electromagnetic Field Monitoring with a Fuzzy Alarm Reporting, *Sensors & Transducers Magazine (S&T e-Digest)*, Vol.56, Issue 6, June 2005, pp. 319-325, ISSN: 1726-5479.
- De Capua, C.; Morello, R. & De Falco, S. (2007). The Design of a Web-Sensor Network for Environmental Monitoring and Remote Control Applications by Data Fusion Techniques, *Proceedings of 24<sup>TH</sup> IEEE IMTC 2007*, ISBN: 1-4244-05890-0/ISSN: 101-5281, May 2007, Warsaw, Poland.
- Hou, An S.; Lin, Chin E. & Gou, Youn Zong (2004). A Wireless Internet-Based Measurement Architecture for Air Quality Monitoring, *Proceedings of IEEE IMTC 2004*, Vol.3, pp. 1901-1906, ISBN 0780382498, May 2004, Como-Italy.
- IEC-ISO GUM, (1995). Guide to the expression of Uncertainty in Measurement, ISBN 92-67-10188-9.
- IEC 50166-2, (1995). Human Exposure to Electromagnetic Field. High Frequency (10 kHz to 300 GHz).
- IEEE 1451, (2001). Smart Transducer Interface Standards.
- ISO EN 14253-1, (1998). Geometrical Product Specifications (GPS) - Part 1- Decision rules for proving conformance or non-conformance with specifications.
- ISO/IEC 17025, (1999). General Requirements for the Competence of Testing and Calibration Laboratories.
- ISO/TS 17450-2, (2002). Geometrical Product Specifications (GPS) - General Concepts-Part 2: Operators and Uncertainties.

- ISO 9001, (2000). Quality Systems-Model for Quality Assurance in Design, Development, Production, Installation and Servicing.
- Lee, K. & Song, E. (2006). A Wireless Environmental Monitoring System Based on the IEEE 1451.1 Standards, *Proceedings of 23rd IEEE IMTC 2006*, pp.1931-1936, ISBN: 0-7803-9360-0, April 2006, Sorrento, Italy.
- Lin, C.R. & Gerla, M. (1997). Adaptive clustering for mobile wireless networks, *IEEE Journal on Selected Areas in Comm.*, vol. 15, pp. 1265-1275.
- Linn, R.J.; Hall, D.L. & Llinas, J. (1991). A Survey of Multi-sensor Data Fusion Systems, *Proceedings on Data Structures and Target Classification SPIE*, Vol. 1470, pp. 13-29, Apr. 1991, Orlando, FL.
- Mahfuz, MU. & Ahmed, KM. (2005). A Review of Micro-nano-scale Wireless Sensor Networks for Environmental Protection: Prospects and Challenges, *Elsevier-Science and Technology of Advanced Materials*, No.6, pp.302-306.
- Neely, A.; Mills, J.; Platts, K.; Richards, H.; Gregory, K.; Bourne, M. & Kennerley, M. (2000). Performance Measurement System Design: Developing and Testing a Process-Based Approach, *International Journal of Operations & Production Management*, Vol. 20, No. 10, pp.1119-1145.
- Nunzi, E.; Panfilo, G.; Tavella, P.; Carbone, P. & Petri, D. (2004). Stochastic and Reactive Methods for the Determination of Optimal Calibration Intervals, *Proceedings of IEEE IMTC 2004*, ISBN 0780382498, May 2004, Como-Italy.
- Saripalli, P.; Amonette, J.; Rutz, F. & Gupta, N. (2006). Design of Sensor Networks for Long Term Monitoring of Geological Sequestration, *Elsevier-Energy Conversion & Management*, No.47, pp.1968-1974.
- Sousa, J. M. & Kaymak, U. (2001). Model Predictive Control Using Fuzzy Decision Functions, *Proc. of IEEE Transactions on Systems, Man and Cybernetics-PartB: Cybernetics*, Vol. 31, No. 1, February 2001.
- Triantaphyllou, E. & Chi-Tun, Lin (1996). Development and Evaluation of Five Fuzzy Multiattribute Decision-Making Methods, *International Journal of Approximate Reasoning*, Elsevier Science Inc., Vol. 14, No. 4, pp. 281-310.
- Tsujita, W.; Yoshino, A.; Ishida, H. & Moriizumi, T. (2005). Gas Sensor Network for air-pollution monitoring, *Elsevier-Sensors and Actuators*, No.110, pp.304-311.
- UNI EN ISO 10012-1, (1993). Quality Assurance Requirements for Measuring Equipment-Part1: Metrological Confirmation System for Measuring Equipment.
- UNI EN ISO 10012-2, (2001). Quality Assurance for Measuring Equipment-Part2: Guidelines for Control of Measurement Processes.
- UNI EN ISO 30012-1, (1993). Quality Assurance Requirements for Measuring Equipment- Metrological Confirmation System for Measuring Equipment.
- Wu, H.; Siegel, M.; Stiefelhagen, R. & Yang, J. (2002). Sensor Fusion using Dempster-Shafer Theory, *Proceedings of IEEE IMTC 2002*, Anchorage, AK, USA.
- Wyatt, D.W. & Castrup, H.T. (1991). Managing Calibration Intervals, *NCSL Annual Workshop & Symposium*, August 1991, Albuquerque.
- Zingales, G. (1996). The Evaluation of Uncertainties in Industrial Measurements for Quality, *Proceedings of IEEE IMTC 1996*, Brussels.

# Monitoring Changes in Operational Scenarios via Data Fusion in Sensor Networks

Papantoni-Kazakos, Dr. Titsa<sup>1</sup> and Burrell, Dr. Anthony<sup>2</sup>

<sup>1</sup>University of Colorado at Denver,

<sup>2</sup>Oklahoma State University

USA

## 1. Introduction

Sensor networks are designed to satisfy specific signal processing objectives, such as target recognition and identification, industrial quality control, community health sensing, multimedia systems & applications, etc. The satisfaction of each objective first requires careful stochastic modeling of the environment and deployment of the pertinent performance criterion; an optimal or best centralized (or coherent) signal processing procedure can be then determined, whose rate of convergence to the deployed performance criterion will be predictable. The centralized procedure utilizes unconstrained raw data, it is performed by a fusion center and attains the best possible convergence rate. In the distributed environment of sensor networks, however, the transmission of raw data to the fusion center induces a high communication cost (both in transmission power and bandwidth), where raw data are collected by local sensors. To reduce the communication cost, raw data are preprocessed by the local sensors. Reduced pertinent information is subsequently transmitted by the local sensors to possibly first cluster heads which in turn process the received information and transmit further reduced information to a fusion center. The fusion center executes the final steps of the now termed *decentralized* (or non-coherent) procedure for the satisfaction of the network objective. The arising issues here are: (a) The performance versus communication cost tradeoff arising when the centralized and the decentralized procedures are compared and (b) the effect of feedback on the convergence rate of the decentralized procedures.

We select as signal processing objective the monitoring of changes in operational scenarios. Such monitoring has numerous applications, including detection of anomalies in community health and industrial quality control. We propose, analyze and evaluate a sequential monitoring algorithm, including convergence, power and false alarm, as well as performance comparison with the corresponding centralized system.

The problem of detecting rapidly and accurately a change in the stochastic process that generates observation data has long history and numerous applications. The applications include industrial quality control, detection of edges in images, network quality control and traffic monitoring in data and sensor networks. The search for algorithms that detect changes in the underlying process which generates observation data has taken two distinct directions: Bayesian and non Bayesian. Since the assumed knowledge of prior probabilities in the Bayesian approaches is considered here unreasonable and unrealistic, we exclusively focus on non Bayesian solutions to the problem.

Motivated by the application of industrial quality control, (Page 1954) first proposed a sequential algorithm to detect a possible change from a given stationary and memoryless process to another given such process, where it was only assumed that such a change may occur randomly in time. (Lorden 1971) analyzed Page's algorithm and proved its asymptotic optimality in the sense of stopping time. (Bansal et al 1986) extended Page's algorithm for stationary and ergodic processes with memory and proved optimality in the sense of asymptotic expected stopping time. (Bansal et al 1989) also "robustified" the algorithm for resistance to data outliers. Burrell et al (1998) extended the algorithm in (Bansal et al 1986) to sequentially detect reoccurring possible changes within a given set of stochastic processes, and analyzed asymptotic performance. (Lai 1995) considered a scenario similar to that in (Burrell et al 1998). (Veeravalli et al 1993) adopted the algorithm of (Page 1954) and that of (Bansal et al 1986) to analyze the effect of a fusion center processing outputs from a set of distributed-data collecting sensors. Some modification of the latter scenario were considered by (Mei 2005). (Burrell et al 2004) considered a distributed modification of the algorithm presented in (Burrell et al 1998), to monitor traffic in sensor networks, where partial decisions from neighboring sensors are incorporated into the sequential algorithmic processing at each sensor.

In this chapter, we consider the existence of a fusion center which processes partial decisions by distributed local sensors, to make the final decision as to the change of the underlying data generating process. The processes model adopted is that of (Burrell et al 1998). Feedback from the fusion center to the sensors is implicit and utilized in the algorithmic steps of the overall system.

The chapter is organized as follows: In Section 2, the system model is presented. In Section 3, preliminaries about the basic algorithms deployed by a centralized system are presented. In Section 4, the algorithmic system is presented and analyzed and the comparison of its performance with that of the centralized system is discussed. In Section 5, numerical evaluation scenarios are included. In Section 6, some conclusions are drawn.

## 2. System model

We consider discrete-time processes and we let time start at zero. Let  $x_n^m$  denote the sequence  $x_1, \dots, x_n$  of  $n$  observations after time zero and let  $\{\mu_i; i = 0, 1, \dots, m-1\}$  denote the measures of  $m$  distinct and parametrically defined stochastic processes. The assumptions in the problem we consider are then as follows: the observation sequence is initially generated by the process  $\mu_0$ , while it is possible that a shift to any one of the  $m-1$  processes  $\mu_i; i = 1, \dots, m-1$  may occur at any point in time, where if a  $\mu_0 \rightarrow \mu_i$  shift occurs, then the process  $\mu_i$  remains active thereafter. The objective is to detect the occurrence of a  $\mu_0 \rightarrow \mu_i$  shift as accurately and as timely as possible, including the detection of the process  $\mu_i$  which  $\mu_0$  changed to. Let us denote by  $f_i; i = 0, 1, \dots, m-1$  density or probability functions induced by the processes  $\mu_i; i = 0, 1, \dots, m-1$  and let us denote by  $f_i(x_n | x_1^{n-1}); i = 0, 1, \dots, m-1$  conditional density or probability functions at  $x_n$ , given the sequence  $x_1^{n-1}$ .

In a centralized system, the problem objective is satisfied by a single processor which collects all the data and processes them sequentially via the algorithm in (Burrell et al 1998). Here, a decentralized system is considered, however, where  $M$  physically distributed processors collect local data, in conjunction with possible feedbacks from a fusion center. (See Figure 1). The  $M$  sensors are identical, placed in identical stochastic environments; that

is, possible changes of the local data generating processes occur simultaneously at all sensor sites. Each sensor deploys a sequential algorithm to detect a possible  $\mu_0 \rightarrow \mu_i, i = 0, 1, \dots, m-1$  change and transmits its local decisions to the fusion center. The fusion center makes the final decision as to a possible change in the data generating process, while it may be implicitly notifying the sensors as to its decision status at all times.

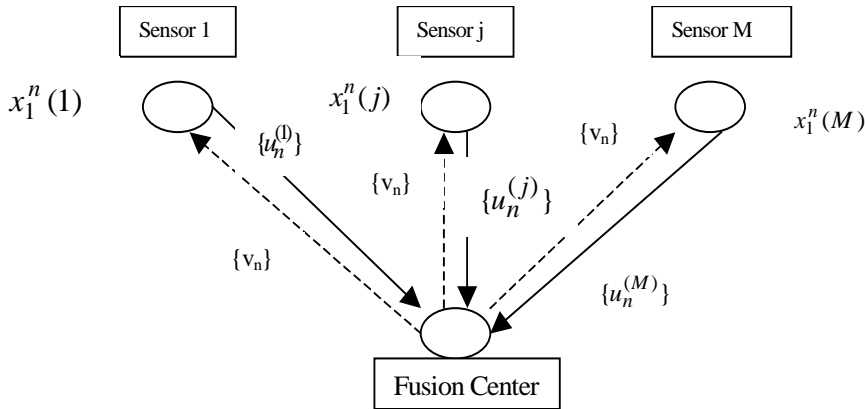


Fig. 1. Fusion Center

Let  $x_1^n(i)$  denote a  $n$ -dimensional local to sensor  $i$  data sequence. Let  $u_n^{(j)}$  denote the input of sensor  $j$  to the fusion center at time  $n$ . Note that sensor  $j$  does not transmit anything to the fusion center, until it makes a decision; if it makes a decision in favor of a  $\mu_0 \rightarrow \mu_i$  shift at time  $n$ , it then transmits  $u_n^{(j)} = i$ ; which also implies  $u_k^{(j)} = i$ ; for all  $k > n$ ; before  $n$  the fusion center simply deduces that  $u_l^{(j)} = 0, l \leq n-1$ , which means that sensor  $j$  has not yet decided that a change from  $\mu_0$  has occurred. Let  $v_n$  denote the feedback of the fusion center to the sensors, at time  $n$ . Then,  $v_n = 0$ ; for all  $n$  before a shift decision is made by the fusion center. Note, that the fusion center does not need to transmit any feedback to the sensors before it makes its shift decision: the sensors simply deduce then that  $v_n = 0$  during such periods. At the time when the fusion center makes its decision, it simply “orders” the sensors to stop their local processing.

### 3. Preliminaries

Let us assume that all the processes  $\mu_i; i = 0, 1, \dots, m-1$  are ergodic and stationary, where

$$L_{i,n} \stackrel{\Delta}{=} n^{-1} \log \frac{f_i(w_1^n)}{f_0(w_1^n)}$$

$$L_{0,n} \stackrel{\Delta}{=} n^{-1} \log \frac{f_0(w_1^n)}{f_i(w_1^n)}$$

$$I_{i,0} \stackrel{\Delta}{=} \lim_{n \rightarrow \infty} L_{i,n}$$

$$I_{0,i} \overset{\Delta}{=} \lim_{n \rightarrow \infty} L_{0,n}$$

and

$$p_{i,n}(v) \overset{\Delta}{=} P_{\mu_i}(L_{i,n} < v)$$

$$p_{0,n}(v') \overset{\Delta}{=} P_{\mu_0}(L_{0,n} < v')$$

and where the following conditions are satisfied, for all  $i = 1, \dots, m-1$ :  $I_{i0}$  and  $I_{0i}$  exist ( $I_{i0}, I_{0i} < \infty$ ) and

$$I_{i0} = E_{\mu_i} \{I_{i0}\} \quad \text{a.s. } (P_{\mu_i})$$

$$I_{0i} = E_{\mu_0} \{I_{0i}\} \quad \text{a.s. } (P_{\mu_0})$$

For  $v \in (0, I_{i0})$

$$\lim_{n \rightarrow \infty} n p_{i,n}(v) = 0 \quad \text{and} \quad \sum_{n \geq 1} p_{i,n} < \infty \tag{A}$$

For  $v' \in (0, I_{0i})$

$$\lim_{n \rightarrow \infty} n p_{0,n}(v') = 0 \quad \text{and} \quad \sum_{n \geq 1} p_{0,n} < \infty \tag{A'}$$

Then, denoting by  $x$  infinite generated sequences, we have from (Bansal et al 1986) the following results, regarding the centralized detection of a  $\mu_0$  to  $\mu_i$  shift :  
 Defining the stopping variable

$$N_{0i}^\delta(x) \overset{\Delta}{=} \inf \{ n : T_n^{0i}(x_1^n) \geq \delta \}$$

for

$$T_n^{0i}(x_1^n) \overset{\Delta}{=} \max_{1 \leq k \leq n+1} \left( \sum_{l=k}^n \log \frac{f_i(x_l | x_k^{l-1})}{f_0(x_l | x_k^{l-1})} \right)$$

we have that,

$$E_{\mu_0} \{N_{0i}^\delta(x)\} > \frac{\delta}{2} \quad \text{as } \delta \rightarrow \infty$$

$$\bar{E}_{\mu_i} \{N_{0i}^\delta(x)\} \sim \frac{\log \delta}{E_{\mu_i} \{I_{i0}\}} \quad \text{as } \delta \rightarrow \infty \tag{1}$$

If the  $\mu_i ; i = 0,1, \dots, m-1$  stochastic processes possess in addition Lai's (1977) mixing conditions, then the stopping variable  $\{N_{0i}^\delta(x)\}$  can be closely approximated by the following stopping variable  $N'_\delta(x)$  which possesses sequential properties while  $N_\delta(x)$  does not.

$$\begin{aligned}
 N_{0i}^{\delta}(x) &= \max_{0 \leq k \leq n} \left( \sum_{t=k+1}^n \log \frac{f_i(x_t | x_1^{t-1})}{f_0(x_t | x_1^{t-1})} \right) \\
 T_{n+1}^{0i}(x_1^{n+1}) &= \max \left( 0, T_{n+1}^{0i}(x_1^n) + \log \frac{f_i(x_{n+1} | x_1^n)}{f_0(x_{n+1} | x_1^n)} \right)
 \end{aligned} \tag{2}$$

where  $\log \frac{f_i(x_{n+1} | x_1^n)}{f_0(x_{n+1} | x_1^n)}$  represents the algorithmic updating step at time n+1.

Denoting  $I_{ij}$  as  $I_{0i}$ , when  $\mu_0$  and  $\mu_i$  are respectively substituted by  $\mu_i$  and  $\mu_j$ , and assuming that Lai’s (1977) mixing conditions hold, we have from Burrell et al (1998):

$$\text{As } \delta \rightarrow \infty, E\{N_{0i}^{\delta}(x) | \mu_j\} \sim \begin{cases} [I_{0j} - I_{ij}]^{-1} \log \delta & ; \text{if } I_{0j} \geq I_{ij} \\ \geq 2^{-1} \delta & ; \text{if } I_{0j} < I_{ij} \end{cases} \tag{3}$$

$$\text{As } \delta \rightarrow \infty, E\{N_{0j}^{\delta}(x) | \mu_j\} \sim I_{0j}^{-1} \log \delta : \begin{cases} < E\{N_{0i}^{\delta}(x) | \mu_j\} \sim [I_{0j} - I_{ij}]^{-1} \log \delta & ; \forall i \neq j : I_{0j} > I_{ij} \\ < E\{N_{0i}^{\delta}(x) | \mu_j\} \geq 2^{-1} \delta & ; \forall i \neq j : I_{0j} < I_{ij} \end{cases} \tag{4}$$

Expressions (1), (3) and (4) represent the asymptotic performance of the centralized system, where m-1 parallel algorithms as in (2) operate, with a common threshold  $\delta$ , and where the first algorithm to cross this threshold determines the system decision: if the  $\mu_0 \rightarrow \mu_k$  algorithm first crosses the threshold, then a  $\mu_0 \rightarrow \mu_k$  shift is decided and the algorithmic system stops.

### 4. The algorithmic system

We assume identical sensors collecting mutually independent local data. We denote by  $x_n(i)$  the  $n^{\text{th}}$  local datum at the  $i^{\text{th}}$  sensor. We denote by  $x_1^{n-1}(i)$  the (n-1)<sup>th</sup> dimensional data sequence collected locally at sensor i from time 1 to time n-1. The algorithms deployed by the sensors are identical, and utilize conditional densities or distributions. In addition to its local data, each sensor also utilizes the implicit fusion centers feedbacks  $\{v_k = 0\}_k$  throughout its operation. Let  $f_j(x_n(i), v_n = 0 | x_1^{n-1}(i), \{v_k = 0\}_{1 \leq k \leq n-1})$  denote the conditional density or distribution of the data at sensors i, given that the acting data process is  $\mu_j$ . It is clearly seen that the  $\{V_n\}$  sequence is a Markov Chain and that the data sequence  $X_1^n(i)$  is independent of  $\{V_n\}$ . We can thus write,

$$\begin{aligned}
 &f_j(x_n(i), v_n = 0 | x_1^{n-1}(i), \{v_k = 0\}_{1 \leq k \leq n-1}) = \\
 &= f_j(v_n = 0 | v_{n-1} = 0, x_1^n(i)) f_j(x_n(i) | x_1^{n-1}(i))
 \end{aligned} \tag{5}$$

We observe that the  $\{v_k\}$  sequence is based only on the  $\{u_k^{(i)}\}$  sequences of the sensor outputs rather than the data sequences collected by the sensors. We thus substitute



$f_j(v_n = 0 | v_{n-1} = 0, x_1^n(i))$  by  $f_j(v_n = 0 | v_{n-1} = 0, u_n^{(i)} = 0)$ . Since the sensors are identical we drop the index  $i$  and we write,

$$\begin{aligned} & f_j(x_n, v_n = 0 | x_1^{n-1}, \{v_k\}_{1 \leq k \leq n-1}) = \\ & f_j(v_n = 0 | v_{n-1}, u_n = 0) f_j(x_n | x_1^{n-1}) = \\ & \frac{f_j(v_n = 0 | u_n = 0)}{f_j(v_{n-1} = 0 | u_n = 0)} f_j(x_n | x_1^{n-1}) \end{aligned} \quad (6)$$

$$\begin{aligned} & \log \frac{f_j(x_n, v_n = 0 | x_1^{n-1}, \{v_k\}_{1 \leq k \leq n-1})}{f_0(x_n, v_n = 0 | x_1^{n-1}, \{v_k\}_{1 \leq k \leq n-1})} = \\ & \log \frac{f_j(x_n | x_1^{n-1})}{f_0(x_n | x_1^{n-1})} + \log \frac{f_j(v_n = 0 | u_n = 0)}{f_0(v_n = 0 | u_n = 0)} - \log \frac{f_j(v_{n-1} = 0 | u_n = 0)}{f_0(v_{n-1} = 0 | u_n = 0)} \end{aligned} \quad (7)$$

The expression in (7) represents the updating step of the  $\mu_0 \rightarrow \mu_j$  shift detecting algorithm in (2) at time  $n$ , at any one of the  $M$  sensors, where  $x_1^n$  are the locally collected data. As compared to the centralized scheme, the terms

$$\log \frac{f_j(v_n = 0 | u_n = 0)}{f_0(v_n = 0 | u_n = 0)} \quad \text{and} \quad -\log \frac{f_j(v_{n-1} = 0 | u_n = 0)}{f_0(v_{n-1} = 0 | u_n = 0)}$$

are added to the updating step. Due to the latter terms, the algorithmic systems across the different sensors are mutually dependent, while the locally collected data are mutually independent, instead.

At the Fusion Center,  $m-1$  parallel algorithms are operating, with a common threshold  $T$ , each monitoring a  $\mu_0 \rightarrow \mu_j$  possible shift, for  $j = 1, \dots, m-1$ . These algorithms utilize the vectors  $\bar{U}_n = [u_n^{(1)}, \dots, u_n^{(m)}]^T$ , where  $u_n^{(i)}$  is the output of sensor  $i$  at time  $n$ . If at time  $n$  the sensor has not made a decision yet, then  $u_n^{(i)} = 0$ . If at time  $n$  the sensor decides in favor of the process shift  $\mu_0 \rightarrow \mu_j$ , then  $u_n^{(i)} = j$ , and this value remains unchanged for all times after  $n$ .

Due to the above discussed evolution of the  $\{u_n^{(i)}\}$  outputs, it is clear that the process  $\{\bar{U}_n\}$  is a Markov Chain. Thus,  $f_j(\bar{U}_n | \bar{U}_k, 1 \leq k \leq n-1) = f_j(\bar{U}_n | \bar{U}_{n-1})$ , where, in addition, the conditional probability  $f_j(\bar{U}_n | \bar{U}_{n-1})$  is determined solely by the transitions of the zero-valued components of  $\bar{U}_{n-1}$ . In fact, due to the identical nature of the sensors,  $f_j(\bar{U}_n | \bar{U}_{n-1})$  is determined by the number of sensors whose algorithms are still running at time  $n-1$ , and among them, from the numbers which transition to the states  $u_n = 1, \dots, m-1$ , at time  $n$ . For sensor  $i$ , let us denote the variable  $d_n^{(i)}$  as,  $d_n^{(i)} = 0$  if  $u_n^{(i)} = 0$  and  $d_n^{(i)} = 1$  if  $u_n^{(i)} = 1, 2, \dots, m-1$ .

Let  $\bar{D}_n$  be the vector whose components are  $d_n^{(i)}; i=1,2,\dots,M$ . Then, we can first write  $f_j(\bar{U}_n|\bar{U}_{n-1}) = f_j(\bar{U}_n|\bar{D}_{n-1}) = f_j(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1}) f_j(\bar{D}_n|\bar{D}_{n-1})$  and,

$$\log \frac{f_j(\bar{U}_n|\bar{U}_{n-1})}{f_0(\bar{U}_n|\bar{U}_{n-1})} = \log \frac{f_j(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1})}{f_0(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1})} + \log \frac{f_j(\bar{D}_n|\bar{D}_{n-1})}{f_0(\bar{D}_n|\bar{D}_{n-1})} \tag{8}$$

From the discussion above, it is evident that the sufficient statistics for the term

$\log \frac{f_j(\bar{D}_n|\bar{D}_{n-1})}{f_0(\bar{D}_n|\bar{D}_{n-1})}$  are  $M^{-1} \sum_{i=1}^M d_n^{(i)}(1-d_{n-1}^{(i)})$  and  $M^{-1} \sum_{i=1}^n (1-d_n^{(i)})(1-d_{n-1}^{(i)})$ . As to the conditional

probability  $f_j(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1})$ ,  $j=0,1, \dots, m-1$ , it represents the probability of the number of sensors deciding in favor of the  $\mu_0 \rightarrow \mu_k; k=1,\dots,m-1$  shifts at time  $n$ , given that their algorithmic systems stop at time  $n$ ; this probability equals 1 if  $m=2$ , since then,  $\bar{D}_n = \bar{U}_n$ .

The sufficient statistics for the probability  $f_j(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1})$  and the term  $\log$

$f_j(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1}) / f_0(\bar{U}_n|\bar{D}_n, \bar{D}_{n-1})$  in (8) are,  $M^{-1} \sum_{i=1}^M d_n^{(i)}(1-d_{n-1}^{(i)})$  and  $M^{-1} \sum_{i=1}^M d_n^{(i)}(1-d_{n-1}^{(i)}) \prod_{\substack{l \leq l \leq m-1 \\ k \neq l}} (u_n^{(i)} - k); 1 \leq l \leq m$ . The expression in (8) represents the updating step

of the  $\mu_0 \rightarrow \mu_j$  shift detecting algorithm in (1) and (2) at time  $n$ , as implemented by the fusion center. Let now us denote,

$p_{j=1,\dots,m-1}^j$  : The probability that the algorithmic system at a sensor stops at time  $n$  ( the common threshold is first crossed at time  $n$ ), given that the data generating process is  $\mu_j$ .

$p_{k_j=1,\dots,m-1}^j$  : The probability that, given the data generating process  $\mu_j$ , the algorithmic system at a sensor stops at time  $n$ , where the  $\mu_0 \rightarrow \mu_k$  shift detecting algorithm is the one that first crosses the threshold at  $n$ .

$\beta_{j=1,\dots,m-1}^j$  : The probability that the algorithmic system at a sensor stops before or at time  $n$ , given that the data generating process is  $\mu_j$ .

$\alpha_n$ : The probability that the algorithmic system at a sensor stops before or at time  $n$ , given that the data generating process is  $\mu_0$ .

We note that  $p_n^j = p_{n_k}^j$  ; if  $m=2$ . Also,  $p_n^j = \beta_n^j - \beta_{n-1}^j$  ;  $j=1, \dots, m-1$  and  $p_n^0 = \alpha_n - \alpha_{n-1}$ .

We now express a theorem, whose proof is in the Appendix.

**Theorem 1:**

Let the probabilities  $f_j(v_n=0|u_n=0); j=1,\dots,m-1$  and  $f_0(v_n=0|u_n=0)$  be such that there exist constants  $n_0, c_j; j=1, \dots, m-1$  and  $c_0$  such that

$$f_j(v_n = 0 | u_n = 0) / f_j(v_{n-1} = 0 | u_{n-1} = 0) = c_j; j=1, \dots, m-1, \forall n > n_0$$

$$f_0(v_n = 0 | u_n = 0) / f_0(v_{n-1} = 0 | u_{n-1} = 0) = c_0; \forall n > n_0 \quad (9)$$

Then, the algorithmic systems across different scenarios are mutually independent for all  $n > n_0$ . In addition, if the  $\mu_j; j=0,1, \dots, m-1$  processes are stationary, ergodic and satisfying conditions (A) and (A') in Section III, as well as (Lai's 1977) mixing conditions, then the performances of the sensors algorithmic systems are asymptotically ( $n > n_0$ ) identical and as in (3) and (4). Finally, the updating step of the  $\mu_0 \rightarrow \mu_j$  shift detecting algorithm at the fusion center in (8) takes then the following form :

$$S_{FS}^{n,j} = U(m-2) \sum_{l=1}^{m-1} M^{-1} \sum_{i=1}^M d_n^{(i)} (1-d_{n-1}^{(i)}) \prod_{\substack{1 \leq k \leq m-1 \\ k \neq l}} (u_n^{(i)} - k) \log \frac{P_{nl}^j / P_n^j}{P_{nl}^0 / P_n^0} \\ + M^{-1} \sum_{i=1}^M d_n^{(i)} (1-d_{n-1}^{(i)}) \log \frac{P_n^j | (1-\beta_{n-1}^j)}{P_n^0 | (1-\alpha_{n-1})} + M^{-1} \sum_{i=1}^M (1-d_n^{(i)}) (1-d_{n-1}^{(i)}) \log \frac{(1-\beta_n^j) | (1-\beta_{n-1}^j)}{(1-\alpha_n) | (1-\alpha_{n-1})}; n > n_0 \quad (10)$$

$$; \text{ where } U(n) = \begin{cases} 1; n > 0 \\ 0; n \leq 0 \end{cases} \quad \square$$

The expected value of the updating step in (10), subject to the data generating process being  $\mu_i$  is found by straight substitution as follows:

$$E\{S_{FS}^{n,j} | \mu_i\} = U(m-2) \sum_{l=1}^{m-1} \frac{P_{nl}^j}{P_n^i} \log \frac{P_{nl}^j | P_n^j}{P_{nl}^0 | P_n^0} + \frac{P_n^i}{1-\beta_{n-1}^i} \log \frac{P_n^j | (1-\beta_{n-1}^j)}{P_n^0 | (1-\alpha_{n-1})} + \\ + \frac{(1-\beta_n^i)}{(1-\beta_{n-1}^i)} \log \frac{(1-\beta_n^j) | (1-\beta_{n-1}^j)}{(1-\alpha_n) | (1-\alpha_{n-1})}; n > n_0 \quad (11)$$

$$E\{S_{FS}^{n,j} | \mu_0\} = U(m-2) \sum_{l=1}^{m-1} \frac{P_{nl}^0}{P_n^0} \log \frac{P_{nl}^j | P_n^j}{P_{nl}^0 | P_n^0} + \frac{P_n^0}{(1-\alpha_{n-1})} \log \frac{P_n^j | (1-\beta_{n-1}^j)}{P_n^0 | (1-\alpha_{n-1})} + \\ + \frac{(1-\alpha_n)}{(1-\alpha_{n-1})} \log \frac{(1-\beta_n^j) | (1-\beta_{n-1}^j)}{(1-\alpha_n) | (1-\alpha_{n-1})}; n > n_0 \quad (12)$$

For asymptotically many sensors ( $M \rightarrow \infty$ ), the updating step in (10) converges to the expected values in (11) and (12), depending on the acting data generating process. Let  $K(p|q)$  denote the Kullback-Leibler information number of a Bernoulli trial with parameter  $p$ , with respect to a Bernoulli trial with parameter  $q$ . Let  $K(\{p_l\}_{1 \leq l \leq m-1} | \{q_l\}_{1 \leq l \leq m-1})$  denote the Kullback-Leibler number of a distribution with probabilities  $\{p_l\}_{1 \leq l \leq m-1}$ , with respect to a distribution with probabilities  $\{q_l\}_{1 \leq l \leq m-1}$ . Then, from expressions (11) and (12), we easily deduce the expressions (13) and (14) below.

$$\begin{aligned}
 E\{S_{FS}^{n,j} | \mu_i\} &= U(m-2) \left[ \mathbb{K} \left( \left\{ \frac{p_{nl}^i}{p_n^i} \right\}_{1 \leq l \leq m-1} \middle| \left\{ \frac{p_{nl}^0}{p_n^0} \right\}_{1 \leq l \leq m-1} \right) - \right. \\
 &\quad \left. - \mathbb{K} \left( \left\{ \frac{p_{nl}^i}{p_n^i} \right\}_{1 \leq l \leq m-1} \middle| \left\{ \frac{p_{nl}^j}{p_n^j} \right\}_{1 \leq l \leq m-1} \right) \right] + \\
 &\quad + \mathbb{K} \left( \frac{1-\beta_n^i}{1-\beta_{n-1}^i} \middle| \frac{1-\alpha_n}{1-\alpha_{n-1}} \right) - \mathbb{K} \left( \frac{1-\beta_n^i}{1-\beta_{n-1}^i} \middle| \frac{1-\beta_{n-1}^j}{1-\beta_{n-1}^j} \right) \tag{13}
 \end{aligned}$$

$$\begin{aligned}
 E\{S_{FS}^{n,j} | \mu_0\} &= -U(m-2) \mathbb{K} \left( \left\{ \frac{p_{nl}^0}{p_n^0} \right\}_{1 \leq l \leq m-1} \middle| \left\{ \frac{p_{nl}^j}{p_n^j} \right\}_{1 \leq l \leq m-1} \right) - \\
 &\quad - \mathbb{K} \left( \frac{1-\alpha_n}{1-\alpha_{n-1}} \middle| \frac{1-\beta_n^i}{1-\beta_{n-1}^i} \right) \tag{14}
 \end{aligned}$$

We note that the quantities  $\left\{ \frac{p_{nl}^i}{p_n^i} \right\}_{1 \leq l \leq m-1}$  ;  $i = 0, 1, \dots, m-1$ ,  $\frac{1-\beta_n^i}{1-\beta_{n-1}^i}$  ;  $i = 1, \dots, m-1$  and  $(1-\alpha_n) / (1-\alpha_{n-1})$  in expression (13) and (14) all represent performance metrics per single sensor. We now state a theorem whose proof is included in the Appendix.

**Theorem 2**

Let the sequences  $\left\{ \frac{p_{nl}^i}{p_n^i} \right\}_{1 \leq l \leq m-1}$  ;  $i = 0, 1, \dots, m-1$ ,  $\frac{1-\beta_n^i}{1-\beta_{n-1}^i}$  ;  $i = 1, \dots, m-1$  and  $(1-\alpha_n) / (1-\alpha_{n-1})$  converge asymptotically. Then, the algorithmic system at the fusion center has the following asymptotic performance characteristics, where  $N_{0j}^T$  denotes the stopping variable of the  $\mu_0 \rightarrow \mu_j$  shift monitoring algorithm in the system when the common threshold is T, and where  $E\{S_{FS}^j | \mu_k\} = \lim_{n \rightarrow \infty} E\{S_{FS}^{nj} | \mu_k\}$ .

$$\text{As } T \rightarrow \infty, \quad E\{N_{0j}^T | \mu_i\} \begin{cases} \sim E^{-1}\{S_{FS}^j | \mu_i\} \log T; & \text{if } E\{S_{FS}^j | \mu_i\} > 0 \\ \geq 2^{-1} T; & \text{if } E\{S_{FS}^j | \mu_i\} < 0 \end{cases} \tag{15}$$

$$\text{As } T \rightarrow \infty, \quad E\{N_{0j}^T | \mu_j\} < E\{N_{0i}^T | \mu_j\} \quad \forall i \neq j \tag{16}$$

In addition, the conditions in Theorem 1 for mutual independence across the various sensors hold, for asymptotically many sensors. □

From the results in Theorem 2, we clearly observe that the asymptotic performance of the algorithm deployed at the fusion center is determined by the performance of the algorithms deployed by the individual sensors, which are determined, in turn, by the Kullback-Leibler numbers among the various acting processes. Furthermore, each individual sensor may be viewed as a representation of a centralized system; thus, comparison between a decentralized and a centralized systems translates to comparison of the fusion center performance to that of

a single sensor. The asymptotic performance of the fusion center is controlled by the limits of the expectations in expressions (13) and (14), which are, in turn, determined by the limits of Kullback-Leibler numbers among power and false alarm quantities induced at a single sensor; the latter numbers are functions of the Kullback-Leibler numbers among the various acting processes. The asymptotic performance of a single sensor, on the other hand, is directly controlled by the Kullback-Leibler numbers among the acting processes. As the latter numbers increase, both sensor and fusion center performances enhance.

## 5. Numerical evaluations

In this section, we examine metrics for the non-asymptotic performance of the algorithms in the system. We first state the general experimental setup. Then, we present numerical results, for a specific scenario.

### 5.1 Experimental setup

In the construction of our experimental setup, we follow the steps listed below :

1. We select specific processes,  $\mu_1, \dots, \mu_{m-1}$ .
2. We construct the specific updating step for each of the parallel algorithms  $\mu_0 \rightarrow \mu_k; k = 1, \dots, m-1$  that are ran at each sensor, as per expression (7) in Section IV.
3. Via the construction in step 2, we compute numerically the quantities  $\{P_{0k}^i(n)\}$ ,  $\{\beta_{0k}^j(n)\}$  and  $\{\alpha_{0k}(n)\}$  in a recursive fashion, where :

$\{P_{0k}^j(n)\}$  : Given that the data generating process is  $\mu_j$ , the probability that the  $\mu_0 \rightarrow \mu_k$  monitoring algorithm crosses the threshold at n.

$$\alpha_{0k}(n) : \sum_{l=1}^n P_{0k}^0(l)$$

$$\beta_{0k}^j(n) : \sum_{l=1}^n P_{0k}^j(l), j = 1, \dots, m-1$$

4. Via the computed quantities in step 3, we compute the quantities  $P_{nk}^j, P_n^j, \alpha_n$  and  $\beta_n^j$  defined in Section IV, as follows :

$$P_{nk}^j = P_{0k}^j(n) \prod_{\substack{1 \leq l \leq m-1 \\ l \neq k}} (1 - \beta_{0l}^j(n)); j = 0, 1, \dots, m-1$$

$$P_n^j = \sum_{k=1}^{m-1} P_{nk}^j; j = 0, 1, \dots, m-1$$

$$\alpha_n = \sum_{l=1}^n P_l^0$$

$$\beta_n^j = \sum_{l=1}^n P_l^j; j = 1, \dots, m-1$$

5. The quantities computed in step 4 are used to compute the updating steps of the parallel algorithms ran by the Fusion Center, as the former are determined by expression (10) in Section IV.
6. The number of sensors is selected. Data are independently generated at each sensor by the same process  $\mu_l$ , where  $\mu_l$  is one of the processed selected in step 1. Given  $\mu_l$ , the overall system-sensors/fusion center - is simulated, where the system algorithmic thresholds have been a priori selected. The performance metrics computed are metrics at the Fusion Center. In particular, the computed metrics for each given  $\mu_l$  are:

$h_{k n}^l$  : Given that the data at the sensors are generated by the process  $\mu_l$ , the percentage of simulated runs that led the Fusion Center to decided at time n that  $\mu_k$  started acting.

$T_k^l$  : The average time to decided in favor of process  $\mu_k$ , given that the data generating process is  $\mu_l$ , where the decision is by the Fusion Center.

$r_{k n}^l$  : Given that the data generating process at the sensors is  $\mu_l$ , the probability that the Fusion Center decides in favor of process  $\mu_k$  before or at time n.

where,

$$r_{k n}^l = \sum_{p=1}^n h_{k n}^l \quad \text{and} \quad T_k^l = \sum_{p=1}^n n \cdot h_{k n}^l$$

m-1 sets of simulations are ran, each corresponding to one of the processes  $\mu_l$ ;  $l = 1, \dots, m-1$  that generate the actual data at each sensor.  $\square$

In step 6, we stated that the algorithm thresholds of the system are a priori selected. The methodology for this selection is as follows.

A. The thresholds across different sensors are identical, since the sensors are considered identical. Per sensor, we test a number of different thresholds,  $\delta_1, \dots, \delta_p$ . For each given threshold,  $\delta_i$ , we evaluate numerically the metrics  $\{\alpha_{0k}(n)\}$  and  $\{\beta_{0k}^k(n)\}_{1 \leq k \leq m-1}$ , where the latter metrics are defined in step 3. Given  $\delta_i$ , we plot the m-1 pairs of curves  $\{\alpha_{0k}(n), \beta_{0k}^k(n)\}_{1 \leq n \leq N}$  for some pre-selected N. We then decide on a value  $n_0$  and select a lower bound  $\beta_0$  for powers and a upper-bound  $\alpha_0$  for false alarms. We select as the operational algorithmic threshold, the minimum among the tested thresholds such that all powers at are above  $\beta_0$  and all false alarms at  $n_0$  are below  $\alpha_0$ . That is, operational selected thresholds attains :

$$\min_{1 \leq k \leq m-1} \beta_{0k}^k(n_0) \geq \beta_0 \quad \text{and} \quad \max_{1 \leq k \leq m-1} \alpha_{0k}(n_0) \leq \alpha_0 .$$

B. The thresholds for the Fusion Center are evaluated and selected, as in (A).

### 5.2 Specific simulation scenario

We selected homogeneous Poisson processes  $\mu_0, \dots, \mu_{m-1}$  with specific different rates. The simplification of the updating step in (7), Section 4, in this case, as well as the computation of the quantities  $\{P_{0k}^j(n)\}$ ,  $\{\beta_{0k}^j(n)\}$  and  $\{\alpha_{0k}(n)\}$  in Step 3, Section 5.1, was included in (Burrell et. al. 1998a) and is also included in the Appendix.

We specifically selected six homogenous Poisson processes,  $\mu_0, \mu_1, \mu_2, \mu_3, \mu_4, \mu_5$ , with corresponding rates per unit time :  $r_0 = 0.1, r_1 = 0.25, r_2 = 0.35, r_3 = 0.5, r_4 = 0.65, r_5 = 0.8$ . We tested several thresholds for the algorithm systems ran by the sensors, and finally selected a common threshold equal to 300. For the latter threshold, all induced powers attained values above 0.97 at time 200 and all false alarms remained below the value 0.005 at the same time. We simulated the overall system, for 30 and 50 sensors and for fusion center threshold values 10, 20, 100 and 300. To exemplify our results, we plot some power and false alarm curves in Figures 2 and 3 below. Specifically, in Figure 2, we plot the power and false alarm curves induced by the algorithm that monitors a change from Poisson rate 0.1 to Poisson rate 0.25 at the fusion center, when the number of sensors is 30 and the algorithmic threshold values are 10, 20, 100 and 300. In Figure 3, we plot the same curves when the

number of sensors is 50. From studying the two figures, we observe that, as the number of sensors increases from 30 to 50, low false alarm and high power are simultaneously attained for less than 100 data, when the threshold value at the fusion center is 10.

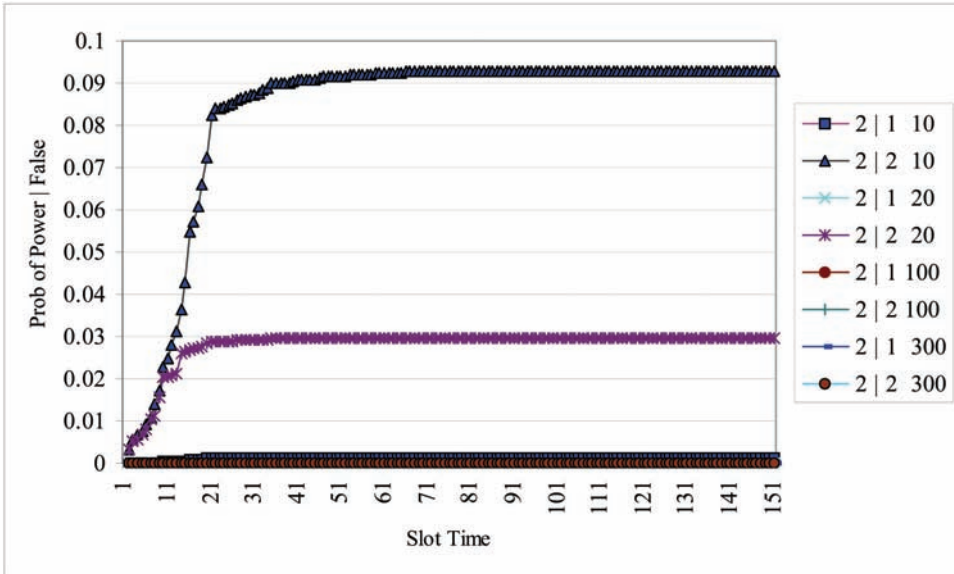


Fig. 2. Power & False Alarm Curves for 0.1 → 0.25 Monitoring Algorithm at the Fusion Center, for 30 Sensors. Legend: 2 | 1 : False Alarm 2 | 2 : Power

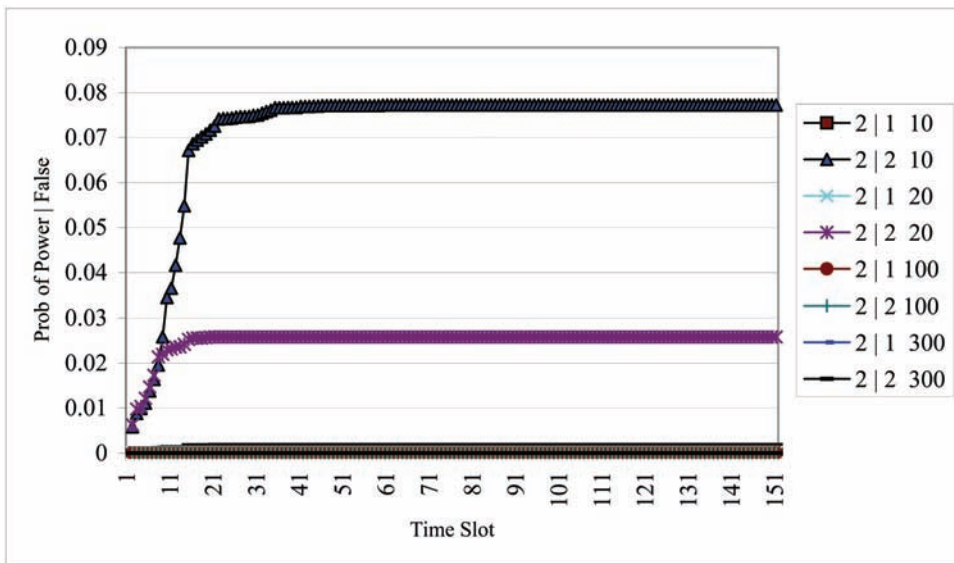


Fig. 3. Power & False Alarm Curves for 0.1 → 0.25 Monitoring Algorithm at the Fusion Center, for 50 Sensors. Legend: 2 | 1 : False Alarm 2 | 2 : Power

In Tables 1 and 2 below, we include our computed  $T_k^l$  values for 30 and 50 sensors respectively, for the Poisson model explained above, and for fusion center threshold values 10, 20, 100 and 300. From the values in the tables, we note that the system approaches its expected asymptotic performance, as the number of sensors increases from 30 to 50, and for fusion threshold value 10.

		k \ l	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 300	2		0	0	0	0	0.0008	0.0077	0
Sensors = 30	3		0	0.0037	0.0073	0.0527	0.1207	0.2117	0.0073
	4		514.26	3.8112	0.2629	0.4904	0.4103	0.2794	0.4904
	5		1290.03	36.1046	1.1842	0.8225	0.4612	0.2874	0.4612
	6		27.0723	13.2926	2.4171	1.311	0.5245	0.2861	0.2861

		k \ l	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 100	2		0	0.0003	0.0044	0.026	0.0541	0.1286	0.0003
Sensors = 30	3		0	0.0779	0.2877	0.4659	0.3382	0.2296	0.2877
	4		398.395	2.5276	0.7833	0.7369	0.3909	0.2342	0.7369
	5		1049.44	23.7907	2.892	1.1396	0.4044	0.2441	0.4044
	6		247.986	33.4387	4.1832	1.207	0.4336	0.2378	0.2378

		k \ l	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 20	2		0.741587	0.2708	0.662	0.5768	0.305	0.2103	0.2708
Sensors = 30	3		5.93519	7.8534	2.0764	0.927	0.3568	0.2206	2.0764
	4		344.015	7.3547	3.1896	0.9682	0.3356	0.2115	0.9682
	5		865.765	7.1572	3.2217	0.9998	0.3564	0.2233	0.3564
	6		380.923	11.6246	3.2076	0.9974	0.3569	0.2035	0.2035

		k \ l	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 10	2		9.00174	1.422	1.7969	0.8834	0.3555	0.2188	1.422
Sensors = 30	3		23.6324	7.3706	2.453	0.9198	0.3502	0.2176	2.453
	4		364.822	7.7948	3.3048	1.002	0.357	0.2125	1.002
	5		809.686	4.6126	2.8298	1.0032	0.3709	0.2204	0.3709
	6		379.928	4.4201	2.8206	0.9947	0.3622	0.2112	0.2112

Table 1.  $T_k^l$  Values at the Fusion Center for 30 Sensors

Legend:

- 1: Rate 0.1      2: Rate 0.25
- 3: Rate 0.35    4: Rate 0.5
- 5: Rate 0.65    6: Rate 0.8



		$k \setminus l$	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 300 Sensors 50	2		0	0	0	0	0	0.0013	0
	3		0	0	0.0037	0.017	0.0642	0.1898	0.0037
	4		493.968	3.2876	0.1555	0.2853	0.2617	0.2593	0.2853
	5		1783.71	43.4664	1.1773	0.6799	0.4089	0.282	0.4089
	6		11.1052	10.7968	2.069	1.1099	0.4433	0.2776	0.2776

		$k \setminus l$	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 100 Sensors 50	2		0	0	0.0001	0.0072	0.0539	0.1522	0
	3		0.972617	0.0393	0.1549	0.234	0.1958	0.207	0.1549
	4		545.46	1.6381	0.8906	0.6919	0.3144	0.2224	0.6919
	5		1247.48	20.5153	2.3331	0.7516	0.321	0.2155	0.321
	6		220.568	39.1443	4.8733	1.0118	0.3317	0.2109	0.2109

		$k \setminus l$	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 20 Sensors 50	2		1.19472	0.1307	0.3722	0.3605	0.2338	0.2041	0.1307
	3		5.97928	5.2579	1.3822	0.5765	0.2408	0.1975	1.3822
	4		382.187	7.9517	2.6174	0.6448	0.2468	0.2036	0.6448
	5		1012.35	5.0408	2.5545	0.7314	0.2501	0.21	0.2501
	6		396.057	8.7207	2.6216	0.7072	0.2418	0.1933	0.1933

		$k \setminus l$	1	2	3	4	5	6	Diagonal
T- Files									
Threshold 10 Sensors 50	2		4.34465	0.8419	1.1619	0.545	0.2322	0.2034	0.8419
	3		10.8702	4.293	1.7951	0.6618	0.2538	0.2085	1.7951
	4		362.561	6.9178	2.5261	0.6414	0.2377	0.1919	0.6414
	5		889.308	4.4505	2.1631	0.6447	0.2438	0.1974	0.2438
	6		469.714	3.7693	2.215	0.6622	0.2446	0.2073	0.2073

Table 2.  $T_k^l$  Values at the Fusion Center for 50 Sensors

Legend:

- 1: Rate 0.1    2: Rate 0.25
- 3: Rate 0.35   4: Rate 0.5
- 5: Rate 0.65   6: Rate 0.8

### 6. Conclusions

We studied a fusion center structure, for the detection of change in the underlying data generating process. We established the pertinent algorithms and stated conditions for the asymptotic optimality of the overall system. In general terms, we showed that the relevant algorithms converge in logarithmic time. We also established metrics for the study of non-asymptotic performance and presented numerical results for a specific Poisson data traffic scenario.

## 7. References:

- Bansal, R. K., and Papantoni-Kazakos, P., An Algorithm for Detecting a Change in Stochastic Process, *IEEE Trans. Inform. Th.*, March 1986, IT-32, pp. 227-235.
- Bansal, R. K., and Papantoni-Kazakos, P., Outlier Resistant Algorithms for Detecting a Change In Stochastic Process, *IEEE Transactions on Information Theory*, May 1989, Vol. 35, pp. 521-535.
- Burrell, A. T., Makrakis, D., Papantoni-Kazakos, P., Traffic Monitoring for Capacity Allocation of Multimedia Traffic in ATM Broadband Networks, *Telecommunications Systems*, Vol 9, 1998, pp. 173-206.
- Burrell, A.T. and Papantoni-Kazakos, P., Extended Sequential Algorithms for Detecting Changes in Acting Stochastic Processes, *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 28, No. 5, Sept. 1998, pp. 703-710.
- Burrell, A. and P. Papantoni, A Distributed Traffic Monitoring Algorithm for Sensor Networks, *CISS 2004*, March 17-19, Princeton University, Princeton, NJ.
- Lai, T. L., Convergence Rates and r-Quick Versions of the Strong Law for Stationary Mixing Sequences, *Annals of Probability*, Vol. 5, 1977, pp. 693-706.
- Lai, T. L., Sequential Change-Point Detection in Quality Control and Dynamic Systems, *J. Roy Statistics Soc. Ser. B*, Vol. 57, 1995, pp. 613-658.
- Lorden, G., Procedures for Reacting to a Change in Distributions, *Annals of Mathematical Statistics*, Vol. 42, pp. 1897-1908, 1971.
- Mei, Y., Information Bounds and Quickest Change Detection in Decentralized Decisions Systems, *IEEE Transactions on Information. Theory*, IT 51-7, July 2005, pp. 2669-2681.
- Page, E. S., Continuous Inspection Schemes, *Biometrika*, Vol. 41, pp. 100-115, 1954.
- Veeravalli, V. V., Basar, T. and Poor, H. V., Decentralized Sequential Detection with a Fusion Center Performing the Sequential Test, *IEEE Transaction on Information Theory*, Vol. 39, no. 2, pp.433-442, March 1993.

## APPENDIX:

### Proof of Theorem 1

When the conditions in (9) hold, the two last terms in the updating step in (7) reduce to a constant, for all  $n > n_0$ . The sensor algorithmic system becomes then identical to that of a centralized system, when no implicit feedback from the fusion center exists. The latter sensor systems are mutually independent, since the local data are. The performance of these independent systems are then as in (Bansal et al 1989) and (Burrell et al 1998). The expression in (10) is derived from (8) in a straight forward fashion, via the mutual independence of the sensors and the sufficient statistics at the fusion center.

### Proof of Theorem 2

When the sequences in the Theorem converge, the Markov Chain  $\left\{ \bar{U}_n \right\}$  becomes asymptotically stationary, and the algorithmic system is optimal in the sense of (Bansal et al 1989); expression (15) is a direct consequence of this optimality. A direct inspection of expression (13) leads to the conclusion expressed by (16) in a straight forward fashion. When the number of sensors is asymptotically large, the updating steps of the algorithms at the fusion converge to the expected values in (13) and (14). Via the assumptions in the Theorem, the latter values converge asymptotically to constants. The latter fact leads directly to the satisfaction of the assumptions in Theorem 1.

The Computation of Useful Probabilities

In this part of the Appendix, we express the useful quantities needed in the simulation scenario of Section 5.2.

Consider the algorithm which monitors a change from  $\mu_0$  to  $\mu_k$ , where the process  $\mu_0$  and  $\mu_k$  homogeneous Poisson, with respective rates  $r_0$  to  $r_k$ . Then, by dividing both threshold and the updating step of the monitoring algorithm by the scaling factor  $|\ln(r_k / r_0)|$ , we obtain the following simplified form of the updating step in (7), Section 4, for

$$\text{sgn } x \stackrel{\Delta}{=} \begin{cases} 1 & ; x > 0 \\ -1 & ; x < 0 \end{cases} : (m \cdot s - t) \text{sgn} \left( \ln \frac{r_k}{r_0} \right)$$

where

$m$  : number of arrivals within a time unit.

$$t, s : \frac{r_k - r_0}{\ln(r_k / r_0)} \approx \frac{t}{s} < 1, \quad t, s, \text{ natural numbers.}$$

Let us denote by  $v$  the (without loss in generality) integer threshold value of the  $\mu_0 \rightarrow \mu_k$  monitoring algorithm. If  $V_0$  is an integer common threshold for the algorithmic system, then

$$v = \left\lfloor \frac{V_0}{(\ln(r_k / r_0))} \right\rfloor. \text{ Let us define :}$$

$P_{0k}^j(y, n)$  : Given that all data are generated by the Poisson process with rate  $r_j$ , the probability that at time  $n$  the  $\mu_0 \rightarrow \mu_k$  monitoring algorithm has the value  $y$  and it has not crossed its threshold before or at  $n-1$ .

We can then express the time/value evolution of the probabilities  $\{P_{0k}^j(y, n)\}$  as follows:

$$\begin{aligned} P_{0k}^j(0, 0) &= 1 \\ n \geq 1 ; P_{0k}^j(0, n) &= \sum_{x=0}^{\min(t, v-1)} P_{0k}^j(x, n-1) e^{-r_j} \\ ; \quad n \geq 1 \text{ and } v > 1+t \\ \quad 1 \leq y \leq v-1-t & ; P_{0k}^j(y, n) = \sum_{m=0}^{\lfloor \frac{y+t}{s} \rfloor} P_{0k}^j(y-ms+t, n-1) e^{-r_j} \frac{(r_j)^m}{m!} \quad (A.1) \\ ; \quad n \geq 1 \\ \quad v-t \leq y \leq v-1 & ; P_{0k}^j(y, n) = \sum_{m=\lfloor \frac{y+t-v+1}{s} \rfloor}^{\lfloor \frac{y+t}{s} \rfloor} P_{0k}^j(y-ms+t, n-1) e^{-r_j} \frac{(r_j)^m}{m!} \end{aligned}$$

Then, the quantities  $P_{0k}^j(n)$ ,  $\beta_{0k}^j(n)$  and  $\alpha_{0k}(n)$  are computed as follows :

$$P_{0k}^j(n) = \sum_{y=0}^{v-1} P_{0k}^j(y, n-1) \left\{ 1 - \sum_{0 \leq m \leq \lfloor \frac{v-y+t}{s} \rfloor - 1} e^{-r_j} \frac{(r_j)^m}{m!} \right\} \quad (A.2)$$

$$\alpha_{0k}(n) = \sum_{l \leq n} P_{0k}^0(l) \quad \text{and} \quad \beta_{0k}^j(n) = \sum_{l \leq n} P_{0k}^j(l) \quad (A.3)$$

# Elements of Sequential Detection with Applications to Sensor Networks

Stefano Marano and Vincenzo Matta

*Department of Information and Electrical Engineering (DIIIE), University of Salerno,  
Italy*

## 1. Introduction

In this contribution the basic elements of sequential detection theory are presented and some applications to sensor networks are then addressed. Sequential detection was basically introduced by A. Wald in 1947 and deals with hypothesis testing problems assuming that the number of observations made available to the detector is virtually unbounded, hence not fixed in advance as it is the case of the more classical and well-known detection paradigms. From a conceptual view point, a sequential decision procedure is stopped when the specific realization of the data available to the detector is sufficiently informative to make a decision that satisfies prescribed error probability bounds. To compare with the Neyman-Pearson paradigm, note that for this latter the false alarm probability is prescribed, and the goal is to achieve the best detection probability, given a fixed number of samples available. In Wald's hypothesis test, instead, we are given both the false alarm and detection probabilities, and the attempt is to make a decision, compatible with those performance levels, using the minimum number of samples. Sequential tests outperform the classical decision procedures. This notwithstanding, they are less known to non specialists, and perhaps less used in practical implementations. In part, this is due to the fact that sequential tests are less easy to be analyzed when the data are non independent and/or non identically distributed, although recent advances in that direction are available.

However, and this is relevant for the present work, there exist many applicative scenarios where the sequential paradigm naturally arises as a suitable framework. This is the case, for instance, of certain sensor network architectures where a mobile agent sequentially queries the nodes of the network, in order to retrieve data or local inferences stored at those sensors. Here the mobile nature of the agent implies that the data are made available to it in a sequential fashion. In addition, the typical dimensions of certain sensor networks make the assumption of an unbounded stream of data available for sampling, a reasonable mathematical model of the physical scenario.

This work is made of two parts. The first presents the theoretical background and tools of sequential detection, while the second addresses some practical applications. In presenting the basic elements of sequential analysis we are neither exhaustive nor mathematically advanced, beyond the typical tools of an electrical engineer. Rather, we hope to collect the main results for easy reference, and basic understanding of the applicative case studies. In discussing the applications to the sensor network, again, we have no pretence at all of

exhaustiveness. Rather, we focus on some recent applications of sequential sampling to sensor networks, as investigated by our research group.

## 2. Basic elements of sequential detection

### 2.1 Martingales

The modern theory of martingales is due to Doob [2] that still remains the basic reference. Below, we only present the basic concepts and results. A random process  $\{X_n\}_{n=1}^{\infty}$  is a martingale if, for  $n = 1, 2, \dots$ , we have

$$E[|X_n|] < \infty, \quad \text{and} \quad E[X_{n+1}|X_1, X_2, \dots, X_n] = X_n. \quad (1)$$

The term martingale was introduced in France to denote a gambling scheme in which the gambler doubles his bet at each step of the play, until he finally wins. More in general, a martingale is intended as a betting scheme designed to improve one's fortune. In effect, if we interpret  $X_n$  as the fortune of the gambler at step  $n$ , the above definition of martingale random process states that the gambler's fortune on the next play is, on the average, the same of his current fortune, irrespectively of the previous history; therefore, the martingale models a fair game [3-5].

The perhaps simplest examples of martingales are the sum  $X_n = \sum_{i=1}^n W_i$  of independent random variables  $W_i$ 's satisfying the first of (1) and having zero mean, and the product  $X_n = \prod_{i=1}^n W_i$  of independent random variables  $W_i$ 's with unit mean (again, if the first of (1) is satisfied).

A slightly general definition of martingale is as follows. The random process  $\{X_n\}_{n=1}^{\infty}$  is a martingale with respect to the random process  $\{Y_n\}_{n=1}^{\infty}$  if, for  $n = 1, 2, \dots$ ,

$$E[|X_n|] < \infty, \quad \text{and} \quad E[X_{n+1}|Y_1, Y_2, \dots, Y_n] = X_n. \quad (2)$$

With the above definitions, the so-called Doob martingale can be introduced by considering an arbitrary random process  $\{Y_n\}_{n=1}^{\infty}$ , and a random variable  $X$  with  $E[|X|] < \infty$ . In fact, the process

$$X_n = E[X|Y_1, Y_2, \dots, Y_n] \quad (3)$$

is easily shown to be a martingale with respect to  $\{Y_n\}_{n=1}^{\infty}$ . Doob's martingale has important applications in various fields, including estimation theory since the sequence of optimal (in the mean square sense) estimators of a random variable  $X$ , given observations  $Y_1, \dots, Y_n$ , is the conditional mean  $E[X | Y_1, \dots, Y_n]$  which, therefore, is nothing but a Doob martingale.

In the sequential sampling framework that we are interested in, the concept of stopping time is key. A random variable  $N$  taking values in  $\{1, 2, \dots, \infty\}$  is a *random time* for the process  $\{X_n\}_{n=1}^{\infty}$ , if the event  $\{N = n\}$  is determined by  $X_1, X_2, \dots, X_n$ . This means that we can decide if  $N = n$  or  $N \neq n$ , by only observing the process  $X_i$  up to time  $n$ , while the samples  $X_{n+1}, X_{n+2}, \dots$  are irrelevant for that. A random time  $N$  is called *stopping time* for the process  $\{X_n\}_{n=1}^{\infty}$  if  $\Pr\{N < \infty\} = 1$ . Therefore,  $N$  is a stopping time for  $\{X_n\}$  if the event  $\{N = n\}$  conditioned on knowing the past of the process  $\{X_n\}_{n=1}^{\infty}$ , does not depend upon what  $\{X_n\}$  does in the future. Indeed, a more general formal definition of stopping time can be given just in terms conditional independence, which is relevant in cases where the sequence of random variables  $\{X_n\}_{n=1}^{\infty}$  is only one actor of a more general probabilistic experiment [6].

Consider now a stopping time for the process  $\{X_n\}_{n=1}^\infty$ , and define the associated *stopped process* [5]: for  $n = 1, 2, \dots$ ,

$$\bar{X}_n = \begin{cases} X_n & \text{if } n \leq N \\ X_N & \text{if } n > N \end{cases} \tag{4}$$

It can be shown that, if  $\{X_n\}_{n=1}^\infty$  is a martingale, then the associated stopped process  $\{\bar{X}_n\}_{n=1}^\infty$  is a martingale too.

By definition of martingale, taking the expectation to both the sides of the second equation in (1), we have  $E[X_{n+1}] = E[X_n]$ ,  $\forall n \geq 1$ , and consequently,

$$E[X_n] = E[X_1]$$

for all  $n$ . Since  $\{\bar{X}_n\}_{n=1}^\infty$  is a martingale too, we also have  $E[\bar{X}_n] = E[\bar{X}_1] = E[X_1]$  (note that always  $\bar{X}_1 = X_1$ ). Since the stopping time is finite with probability one, the original process  $\{X_n\}_{n=1}^\infty$  will be eventually stopped, that is to say, there must exist a (sufficiently large) value of  $n$  such that  $\bar{X}_n = X_N$ . From that value of  $n$  on, the stopped process remains constant, implying that  $\lim_n \bar{X}_n = X_N$ . Taking the expectation

$$E[\lim_n \bar{X}_n] = E[X_N].$$

If we could exchange the limit with the statistical expectation operator, we would get the following important result

$$E[X_N] = E[\lim_n \bar{X}_n] = \lim_n E[\bar{X}_n] = E[X_1], \tag{5}$$

Indeed, under appropriate technical conditions the above exchange is legitimate and this result is known as the *martingale optional stopping theorem*, which states the following (see e.g., [4, 5]): Equation (5) holds true, provided that at least one of the following conditions is met

- the random variables  $\{\bar{X}_n\}$  are uniformly bounded;
- the stopping time  $N$  is bounded, i.e.,  $\Pr\{N \leq k\} = 1$ , for some  $k \geq 1$ ;
- $E[N] < \infty$  and  $E[|X_{n+1} - X_n| \mid X_1, \dots, X_n] < k < \infty$ , for some  $k \geq 1$ .

### 2.2 Sequential probability ratio test

With the concepts introduced above, we can now elaborate on the likelihood ratio to derive the basic design formulas for the sequential test proposed by Wald. Let us consider the binary test between two simple hypotheses:

$$\begin{aligned} \mathcal{H}_0 : Y_i &\sim f_0(y), \\ \mathcal{H}_1 : Y_i &\sim f_1(y), \end{aligned} \tag{6}$$

where  $f_{0,1}(y)$  are two known probability density functions of  $Y_i$ , under hypothesis 0, 1, and where the data  $Y_i, i = 1, 2, \dots$  are iid (independent, identically distributed), for simplicity. As distinct feature, the number of such observations is not determined in advance but it can be virtually unbounded. Wald's test, also known as SPRT (Sequential Probability Ratio Test) prescribes to proceed as follows. Let  $\Lambda_n$  be the likelihood ratio pertaining to the above

statistical test, using the first  $n$  samples  $Y_1, \dots, Y_n$  available, that is:  $\Lambda_n = \prod_{i=1}^n f_1(y_i)/f_0(y_i)$ . We have [1]

$$\Lambda_n \begin{cases} \geq \gamma_1 & \text{decide } \mathcal{H}_1 \text{ and stop testing ,} \\ \leq \gamma_0 & \text{decide } \mathcal{H}_0 \text{ and stop testing ,} \\ \text{otherwise} & \text{take the } (n+1)^{\text{th}} \text{ sample (i.e., increase } n\text{).} \end{cases} \quad (7)$$

As it is clear from this formulation, the actual number of samples processed in order to make a decision is not fixed and is instead a random quantity, whose actual value will depend on the specific realization of the observation process  $\{Y_n\}_{n=1}^{\infty}$ . The detection and false alarm probabilities of the test are defined as usual:

$$P_d = \Pr\{\text{the decision is for } \mathcal{H}_1 \mid \text{the actual state of the nature is } \mathcal{H}_1\}, \quad (8)$$

$$P_f = \Pr\{\text{the decision is for } \mathcal{H}_1 \mid \text{the actual state of the nature is } \mathcal{H}_0\}, \quad (9)$$

and there is an amazing simple relationship relating the pair  $(P_d, P_f)$  to the thresholds of the test  $(\gamma_0, \gamma_1)$ , as we shall promptly see.

Let us assume that hypothesis  $\mathcal{H}_0$  is actually in force. Given the observations  $Y_1, \dots, Y_n$  let us consider the random process build upon the likelihood ratio

$$\Lambda_n = \frac{\prod_{i=1}^n f_1(y_i)}{\prod_{i=1}^n f_0(y_i)}. \quad (10)$$

Under mild regularity conditions (i.e., assuming  $E[|\Lambda_n| \mid \mathcal{H}_0] < \infty, \forall n$ ), the process  $\{\Lambda_n\}_{n=1}^{\infty}$  is a martingale with respect to the observation process  $\{Y_n\}_{n=1}^{\infty}$ . Indeed, we have

$$\begin{aligned} E[\Lambda_{n+1} | Y_1, Y_2, \dots, Y_n] &= E\left[\frac{\prod_{i=1}^{n+1} f_1(y_i)}{\prod_{i=1}^{n+1} f_0(y_i)} \mid Y_1, Y_2, \dots, Y_n\right] \\ &= E\left[\frac{f_1(y_{n+1})}{f_0(y_{n+1})} \frac{\prod_{i=1}^n f_1(y_i)}{\prod_{i=1}^n f_0(y_i)} \mid Y_1, Y_2, \dots, Y_n\right] \\ &= E\left[\frac{f_1(y_{n+1})}{f_0(y_{n+1})}\right] \Lambda_n = \Lambda_n. \end{aligned} \quad (11)$$

Let us define the random time (with  $\gamma_0 < 0$  and  $\gamma_1 > 0$ )

$$N = \min\{n \geq 1 : \Lambda_n \leq e^{\gamma_0} \text{ or } \Lambda_n \geq e^{\gamma_1}\} \quad (12)$$

and assume that  $\Pr\{N < \infty\} = 1$ , i.e., that  $N$  is actually a stopping time for the process  $\{Y_n\}_{n=1}^{\infty}$ . The associated stopped process can be defined as in (4), and we can invoke (under suitable regularity conditions, see above) the martingale optional stopping theorem, yielding

$$\begin{aligned} E[\Lambda_N] &= E[\Lambda_N \mid \Lambda_N \leq e^{\gamma_0}] \Pr\{\Lambda_N \leq e^{\gamma_0}\} + E[\Lambda_N \mid \Lambda_N \geq e^{\gamma_1}] \Pr\{\Lambda_N \geq e^{\gamma_1}\} \\ &= E[\Lambda_N \mid \Lambda_N \leq e^{\gamma_0}] (1 - P_f) + E[\Lambda_N \mid \Lambda_N \geq e^{\gamma_1}] P_f = E[\Lambda_1] = 1. \end{aligned} \quad (13)$$

Neglecting the *excess over the boundaries* (this is also known as Wald's approximation), yields

$$E[\Lambda_N | \Lambda_N \leq e^{\gamma_0}] \approx e^{\gamma_0}, \tag{14}$$

$$E[\Lambda_N | \Lambda_N \geq e^{\gamma_1}] \approx e^{\gamma_1}, \tag{15}$$

that underly many of the approximate design formulas for sequential detectors. We therefore get

$$e^{\gamma_0} (1 - P_f) + e^{\gamma_1} P_f \approx 1. \tag{16}$$

Reasoning in the same way under hypothesis  $\mathcal{H}_1$ , but using as martingale the the inverse of the likelihood ratio  $1/\Lambda_n$ , we get

$$e^{-\gamma_0} (1 - P_d) + e^{-\gamma_1} P_d \approx 1. \tag{17}$$

Putting together eqs. (16) and (17), immediately yields<sup>1</sup>

$$\gamma_1 \approx \log \frac{P_d}{P_f}, \tag{18}$$

$$\gamma_0 \approx \log \frac{1 - P_d}{1 - P_f}. \tag{19}$$

We reiterate that the approximation involved follows from having neglected the excess over the boundaries<sup>2</sup>.

Equations (18) and (19) relate the error probability of the SPRT to the thresholds and are therefore used to set the thresholds of the test, given prescribed performance level. Intriguingly, note that we can set the thresholds without knowing the statistics of the observations; clearly, the likelihood ratio does depend on these statistics.

Once that the desired error probabilities have been fixed, and that the thresholds are accordingly set, the main performance figure of the Wald’s test is the average sample number (ASN)  $E[N]$ . This is the expected number of samples needed to make a decision, and is also referred to as the (averaged) decision delay. To characterize  $E[N]$  let us start from what is known in the literature as *Wald’s identity*. Let  $W_i$ ,  $i = 1, 2, \dots$  be a sequence of iid random variables, and let  $X_n = \sum_{i=1}^n W_i$  their cumulative sum. Also, let

$$N = \min\{n \geq 1 : X_n \leq \gamma_0 \text{ or } X_n \geq \gamma_1\}. \tag{20}$$

Introducing the semi-invariant moment generating function of the random variable  $W_i$ , that is  $\Gamma(r) = \log E[e^{rW_i}]$ , and assuming that  $\Gamma(r)$  is finite in an open interval  $\Omega$  around the origin  $r = 0$ , we have that, for  $r \in \Omega$ ,

<sup>1</sup> Logarithms are to natural base.

<sup>2</sup> The approximation involved in the above relationships is often accurate for practical purposes, but one can also resort to certain bounds. Specifically, let  $P_{da}$  and  $P_{fa}$  the *actual* detection and false alarm probabilities that one gets by setting the thresholds according to eqs. (18) and (19) in which the *nominal* values  $P_d$  and  $P_f$  of those probability are used. Then, see e.g. [7]:

$$1 - P_{da} \leq \frac{1 - P_d}{1 - P_f}, \quad P_{fa} \leq \frac{P_f}{P_d}.$$



$$\Gamma(r) = \log \mathbf{E}[e^{rW_i}] \quad (21)$$

Differentiating both sides of Wald's identity (21) and evaluating the result at  $r = 0$ , the so-called *Wald's equality* is obtained:

$$\mathbf{E}[X_N] = \mathbf{E}[N] \mathbf{E}[W]. \quad (22)$$

(Clearly, the above equality can be also proved more directly, without resorting to Wald's identity, see, e.g., [5].) On the other hand, the second derivative evaluated at  $r = 0$  gives

$$\mathbf{E}[X_N] = \mathbf{E}[N] \text{VAR}[W], \quad (23)$$

which is useful in sequential analysis whenever  $\mathbf{E}[W] = 0$ , in which case Wald's equality is of limited use.

Suppose now that the true hypothesis is  $\mathcal{H}_0$ , and let us replace the generic  $X_n$  with  $L_n$ , the *log-likelihood ratio* of hypothesis test (6):

$$L_n = \sum_{i=1}^n \log \frac{f_1(y_i)}{f_0(y_i)}. \quad (24)$$

With this assumption, it should be clear that the stopping time  $N$  in eq. (20) is exactly that defined in (12), and we are in fact faced with the earlier discussed sequential test. Clearly,  $L_n$  is the cumulative sum of the sequence of iid random variables  $\log [f_1(y_i)/f_0(y_i)]$ , whence application of Wald's equality yields

$$\mathbf{E}[N | \mathcal{H}_0] = \frac{\mathbf{E}[L_N | \mathcal{H}_0]}{\mathbf{E}\left[\log \frac{f_1(y_i)}{f_0(y_i)} | \mathcal{H}_0\right]}. \quad (25)$$

The numerator can be expanded by conditioning, and then approximated by Wald's approximations:

$$\begin{aligned} \mathbf{E}[L_N | \mathcal{H}_0] &= \mathbf{E}[L_N | L_N \leq \gamma_0, \mathcal{H}_0] \Pr\{L_N \leq \gamma_0 | \mathcal{H}_0\} \\ &+ \mathbf{E}[L_N | L_N \geq \gamma_1, \mathcal{H}_0] \Pr\{L_N \geq \gamma_1 | \mathcal{H}_0\} \\ &\approx \gamma_0 \Pr\{L_N \leq \gamma_0 | \mathcal{H}_0\} + \gamma_1 \Pr\{L_N \geq \gamma_1 | \mathcal{H}_0\} \\ &= \gamma_0 (1 - P_f) + \gamma_1 P_f \\ &\approx (1 - P_f) \log \frac{1 - P_d}{1 - P_f} + P_f \log \frac{P_d}{P_f}, \end{aligned} \quad (26)$$

where the last approximation follows from eqs. (18) and (19).

Defining the binary divergence (measured in nats)  $D_b(\alpha | \beta)$  between two probability mass functions (pmf)  $[\alpha, 1 - \alpha]$  and  $[\beta, 1 - \beta]$  as [8]

$$D_b(\alpha | \beta) = \alpha \log \frac{\alpha}{\beta} + (1 - \alpha) \log \frac{1 - \alpha}{1 - \beta},$$

the numerator of (25) can be expressed (within the stated approximation) as follows

$$E[L_N | \mathcal{H}_0] \approx -D_b(P_f || P_d).$$

The divergence between two arbitrary probability density functions (pdfs)  $f_0(y)$  and  $f_1(y)$  is defined as [8]

$$D(f_0 || f_1) = \int f_0(y) \log \frac{f_0(y)}{f_1(y)} dy$$

implying that the denominator of (25) is  $-D(f_0 || f_1)$ . We finally get  $E[N | \mathcal{H}_0] = \frac{D_b(P_f || P_d)}{D(f_0 || f_1)}$ . Elaborating exactly in the same way, assuming  $\mathcal{H}_1$  true, we can compute  $E[N | \mathcal{H}_1]$ . Therefore, the final result is

$$E[N | \mathcal{H}_0] \approx \frac{D_b(P_f || P_d)}{D(f_0 || f_1)}, \tag{27}$$

$$E[N | \mathcal{H}_1] \approx \frac{D_b(P_d || P_f)}{D(f_1 || f_0)}. \tag{28}$$

Roughly speaking, the above approximations become tighter and tighter for small error probabilities  $P_f, 1 - P_d \ll 1$ , namely, when the no decision region between the thresholds is large:  $\gamma_0 \ll 1$  and  $\gamma_1 \gg 1$ . Otherwise stated, the approximations are fair when the average number of samples collected is large enough.

It is worth noting how Wald popularized these formulas in 1947 in a form that did not involve the divergences, and in fact the divergence was defined by Kullback [9] in the context of information theory, a theory born just one year later, in 1948, with the work by Shannon.

### 2.3 Sequential detection with general test statistics

The fundamental results presented in the previous section trace the route for implementing the SPRT, as well as for computing simple approximations for performance evaluation. Of course, since the pioneering work by Wald, these results have been extended in many different directions, including, among many other, the case of dependent and non-identically distributed observations [10], sequential tests with arbitrary detection statistics [6], asymptotic results for vanishing signal-to-noise ratio [11], refined approximations for the excesses over boundaries [12], and so on. An exhaustive review of these concepts is clearly beyond the scope of the present work. In this section, we limit ourselves to consider a setting which slightly extends the classical SPRT framework. The mathematical results, which are perhaps less intuitive than the classical Wald’s formulas, turn out to be useful from an engineering perspective, as we next show in the sections devoted to sensor network applications.

Let  $T_n = \sum_{i=1}^n t(Y_i)$ , with  $Y_i$  iid random variables, and with  $t(\cdot)$  being a certain transformation, in general different from the log-likelihood ratio. We consider the case that a sequential test is implemented, based upon the above  $T_n$ . More specifically, let  $N$  be the smallest  $n$  for which either  $T_n \geq \gamma_1$  or  $T_n \leq \gamma_0$ . For concreteness, we assume that the  $t(Y_i)$  has positive expectation under  $\mathcal{H}_1$  and negative expectation under  $\mathcal{H}_0$ , and that the two thresholds accordingly obey  $\gamma_0 < 0 < \gamma_1$ . This problem can be cast in the more general framework of random walks with two thresholds [6, 13, 14].

In order to assess the test performances, we need setting the thresholds in order to meet the prescribed false alarm and detection probabilities. To fix ideas, let us consider the case that  $E[t(Y_i)] < 0$ , that is, for our purposes, we are under  $\mathcal{H}_0$ . Assuming that the semi-invariant moment generating function  $\Gamma(r)$  of the random variable  $t(Y_i)$  is finite in an open interval  $\Omega$  around the origin, and that has a root  $r^* > 0$ , we can use Wald's identity with  $r = r^*$ , that is,  $E[e^{r^* T_N}] = 1$ , which is equivalent to

$$E[e^{r^* T_N} | T_N \leq \gamma_0] \Pr\{T_N \leq \gamma_0\} + E[e^{r^* T_N} | T_N \geq \gamma_1] \Pr\{T_N \geq \gamma_1\} = 1 \quad (29)$$

Note that, if  $T_n$  is the log-likelihood ratio,  $r^* = 1$ , and eq. (29) essentially translates into eq. (13), which, incidentally, was obtained by resorting to the theory of martingales. Now, different bounds and approximations can be derived from (29). One option is to neglect the overshoots, as previously done for the log-likelihood ratio, obtaining

$$e^{r^* \gamma_0} (1 - \Pr\{T_N \geq \gamma_1\}) + e^{r^* \gamma_1} \Pr\{T_N \geq \gamma_1\} \approx 1, \quad (30)$$

which is the counterpart of eq. (16).

The above technique gives also a direct way to derive upper bounds on the threshold crossing probabilities. Indeed, conditional on  $T_N \geq \gamma_1$ ,  $e^{r^* T_N} \geq e^{r^* \gamma_1}$ , and being all terms on the right-hand-side of eq. (29) non-negative, we easily have

$$\Pr\{T_N \geq \gamma_1\} \leq e^{-r^* \gamma_1}. \quad (31)$$

A positive threshold crossing can be considered, under the assumption that  $E[t(Y_i)] < 0$ , as an event becoming rare as  $\gamma_1$  grows, and this is consistent with the obtained *exponential* bound. All the reasoning can be applied to the random walk with  $E[t(Y_i)] > 0$  (that is, under  $\mathcal{H}_1$ ), obtaining the two formulas:

$$e^{-|r^{**}| \gamma_0} (1 - \Pr\{T_N \geq \gamma_1\}) + e^{-|r^{**}| \gamma_1} \Pr\{T_N \geq \gamma_1\} \approx 1, \quad (32)$$

and

$$\Pr\{T_N \leq \gamma_0\} \leq e^{-|r^{**}| \gamma_0},$$

where the non-zero root of  $\Gamma(r)$  is now negative, and is denoted by  $r^{**}$ . We note explicitly that these exponential bounds do not work for the case that the random walk is zero-mean. Before concluding this section, we would like to report another useful tool, which extends the previous results to the characterization of the joint distribution of the stopping time  $N$  and the barriers. Assuming again that  $E[t(Y_i)] < 0$  and  $\Gamma(r^*) = 0$  for some  $r^* > 0$ , the following two bounds can be derived [6]:

$$\Pr\{N \geq n, T_N \geq \gamma_1\} \leq \begin{cases} e^{-r \gamma_1 + \Gamma(r)}; & \text{for } n > \gamma_1 / \Gamma'(r^*), \quad r : \Gamma'(r) = \gamma_1 / r, \\ e^{-r^* \gamma_1}; & \text{for } n \leq \gamma_1 / \Gamma'(r^*), \end{cases}$$

and

$$\Pr\{N \leq n, T_N \geq \gamma_1\} \leq \begin{cases} e^{-r\gamma_1+n\Gamma(r)}; & \text{for } n < \gamma_1/\Gamma'(r^*), \quad r : \Gamma'(r) = \gamma_1/r, \\ e^{-r^*\gamma_1}; & \text{for } n \geq \gamma_1/\Gamma'(r^*) \quad . \end{cases}$$

In some sense, the above bounds furnish an interpretation of the value  $n^* = \gamma_1/\Gamma'(r^*)$  as the typical value of the stopping time  $N$  conditional on the upper threshold crossing. This concludes our survey of the basic tools and results of sequential detection. In the following, we apply and extend these results by addressing several case studies.

### 3. Selected applications

We now present some applications of the sequential sampling theory, whose basic elements have been summarized in the previous sections; the applications are selected from the authors' recent works on the subject, which are somehow related to sensor networks. In the first example (Sect. 3.1) the decentralized architecture of the sensor network is key, and we take a genuine cross-layer perspective of the whole system that merges the detection layer with the (many-to-one) communication layer. The last two examples (Sects. 3.3 and 3.2) focus on the signal processing at the sensor level designed for improving the detection performances of the fusion center, and are therefore exploitable even in certain non decentralized systems. Whenever appropriate, for easy reference, we try to maintain the notation as close as possible to that of the original works to which we refer for more general discussion, in-depth description, and for many technicalities which are deliberately neglected in this presentation.

#### 3.1 SENMA detection with censoring nodes [15]

Suppose that a WSN designed for solving a binary hypothesis test is made of many tiny remote units uniformly deployed over the surveyed area, and of a Mobile Agent (MA) having the role of fusion center. The remote units sense the environment and collect data relevant to the detection task, while the MA travels across the network domain and sequentially polls the sensors. Indeed, in the SENMA (Sensor Network with Mobile Agents) architecture proposed in [16], see also [17–19], at each successive MA's *snapshot* the nodes falling within its field of view are queried for delivering their data. Oppositely to the intrinsic nature of the remote units, the MA can be a very reliable device with large power capabilities and adequate communication/computational properties. In addition, its mobile nature greatly simplifies the sensors/MA communication tasks, thus making the SENMA architecture particularly suited for many practical applications being scalable, robust and simple to implement. In addition, as one might expect, the more important advantage of the SENMA over alternative network structures (*e.g.*, ad-hoc system) is in terms of energy saving for communications, a key parameter for sensor networks.

The MA collects the data delivered from the sensors and, as soon as a new observation is made available to it, this is included in the computed detection statistic. We assume that such a statistic is the cumulative sum of the log-likelihood ratios, resulting in the Wald's SPRT [1], discussed previously in this work.

The specific viewpoint taken in [15] is that the remote units do not necessarily deliver their data to the MA when they fall in its field of view. In order to further economize the energy burden, a censoring protocol is implemented [20–23]. Data are delivered only if they are

sufficiently informative: the sensor transmission is inhibited if the the locally computed log-likelihood of the measured data does not exceed (in modulus) a certain threshold level. In this way a communication session is activated, and the correspondent energy is spent by the sensor, only if the local observation is expected to contribute in a meaningful way to the final decision. Otherwise, such data will be never received by the MA and do not play any role in building the final statistics. A trade off clearly emerges between detection performances and energy consumption.

### 3.1.1 Network performances

Let us suppose that data collected by the remote units of the network are  $M$ -vectors of iid (independent and identically distributed) observations, and that different nodes observe iid data, as well. If we label with an index  $n = 1, 2, \dots$ , the (virtually, infinitely many) remote units, the basic hypothesis test under study is as follows

$$\begin{aligned} \mathcal{H}_0 : \mathbf{x}_n &= \mathbf{w}_n, \\ \mathcal{H}_1 : \mathbf{x}_n &= \mathbf{w}_n + \mu \mathbf{1}, \end{aligned} \quad (33)$$

where  $\mathbf{1}$  represents a vector of all 1s. The vectors  $\mathbf{w}_n = [w_{n1}, w_{n2}, \dots, w_{nM}]$  have iid components picked from a continuous random variable, whose probability density function (pdf) is  $\phi(w)$ , and is here assumed to be an even function with domain the whole real axis. The known parameter  $\mu$  rules the amount of shift in mean that distinguishes the two alternative hypotheses.

Denoting by  $x_{nm}$  the  $m$ th observation taken at the  $n$ th sensor, the local log-likelihood is

$$L(\mathbf{x}_n) = \sum_{m=1}^M \log \frac{\phi(x_{nm} - \mu)}{\phi(x_{nm})},$$

and, in absence of censoring, the SPRT would be (see eq.(7))

$$\sum_{n=1}^j L(\mathbf{x}_n) \begin{cases} \geq \gamma_1 & \text{choose } \mathcal{H}_1, \\ \leq \gamma_0 & \text{choose } \mathcal{H}_0, \\ \text{otherwise} & \text{take another sample (increase } j). \end{cases} \quad (34)$$

Due to the censoring, when polled by the MA, the  $n$ th sensor of the network actually delivers data only if  $L(\mathbf{x}_n) \notin [-\delta, \delta]$ , where  $\delta$  is the level of censoring. The delivering probability can be found to be

$$p_t(\delta) = \Pr[L(\mathbf{x}_n) \notin [-\delta, \delta]] = 1 - F_i(\delta) + F_i(-\delta) = F_1(-\delta) + F_0(-\delta), \quad (35)$$

where  $F_i(y)$  is the CDF of the log-likelihood  $L(\mathbf{x}_n)$ , under hypothesis  $i = 0, 1$ .

Let  $N_t$  be the random number of sensors that actually deliver data to the MA, as opposed to  $N_v$ , the number nodes encountered by the MA in its travel across the surveyed area. Denoting by  $\mathcal{I}(\cdot)$  the indicator function, we have  $N_t = \sum_{n=1}^{N_v} \mathcal{I}\{L(\mathbf{x}_n) \notin [-\delta, \delta]\}$ , where  $N_v$  is a valid stopping time [6], so that Wald's equality (22) yields

$$\mathbb{E}[N_t] = \mathbb{E}[N_v] \mathbb{E}[\mathcal{I}\{L(\mathbf{x}_n) \notin [-\delta, \delta]\}] = \mathbb{E}[N_v] p_t(\delta). \quad (36)$$

It makes sense to adopt  $N_t$  as a proxy of the energy consumption, and  $N_v$  as a proxy of detection performance in terms of detection delay for achieving a desired level of error probabilities. Therefore, the above relationship emphasize the trade off between detection performances (more data yield better performances) and energy saving, tuned by the censoring level  $\delta$ .

We need now to introduce the following quantity:

$$s(\delta) = \frac{1}{p_t(\delta)} \int_{|\xi|>\delta} f_1(\xi) \log \frac{f_1(\xi)}{f_0(\xi)} d\xi = \frac{1}{p_t(\delta)} \int_{|\xi|>\delta} \xi f_1(\xi) d\xi. \quad (37)$$

Denoting by  $P_d$  and  $P_f$  the desired detection and false alarm probabilities set at the design stage, see eqs. (8) and (9), in [15] the framework of sequential analysis earlier discussed is exploited, to show that (via proper modification of the the techniques leading to eqs. (27) and (28)):

$$\mathbb{E}[N_t|\mathcal{H}_0] = \frac{D_b(P_f||P_d)}{s(\delta)}, \quad \mathbb{E}[N_t|\mathcal{H}_1] = \frac{D_b(P_d||P_f)}{s(\delta)},$$

and

$$\mathbb{E}[N_v|\mathcal{H}_0] = \frac{D_b(P_f||P_d)}{p_t(\delta)s(\delta)}, \quad \mathbb{E}[N_v|\mathcal{H}_1] = \frac{D_b(P_d||P_f)}{p_t(\delta)s(\delta)}.$$

It can be also shown that  $s(\delta)$  is monotonically increasing in  $\delta$ , while the product  $p_t(\delta)s(\delta)$  monotonically decreases with  $\delta$ . This implies that the larger is  $\delta$ , the more energy the network saves but the larger is the detection delay.

### 3.1.2 Optimization

To compare our censored system with respect to the absence of censoring, let us define  $\mathbb{E}[N]$  as the average number of sensors resulting from assuming  $\delta = 0$  (no censoring). The following quantities can be introduced

$$\eta_t = \frac{\mathbb{E}[N_t|\mathcal{H}_0]}{\mathbb{E}[N|\mathcal{H}_0]} = \frac{\mathbb{E}[N_t|\mathcal{H}_1]}{\mathbb{E}[N|\mathcal{H}_1]} = \frac{s(0)}{s(\delta)} \quad (38)$$

and

$$\eta_v = \frac{\mathbb{E}[N_v|\mathcal{H}_0]}{\mathbb{E}[N|\mathcal{H}_0]} = \frac{\mathbb{E}[N_v|\mathcal{H}_1]}{\mathbb{E}[N|\mathcal{H}_1]} = \frac{\eta_t}{p_t(\delta)}. \quad (39)$$

Now, let us consider the single-sample local log-likelihood  $L(x_n)$  and let us denote by  $a$  and  $b^2$  its mean and variance:

$$a = \mathbb{E}[L(x_n)|\mathcal{H}_1] = -\mathbb{E}[L(x_n)|\mathcal{H}_0] \quad (40)$$

$$b^2 = \text{VAR}[L(x_n)|\mathcal{H}_1] = \text{VAR}[L(x_n)|\mathcal{H}_0]. \quad (41)$$

Also let

$$\Delta = \frac{\delta}{b\sqrt{M}}, \quad \rho = \frac{a\sqrt{M}}{b}. \quad (42)$$

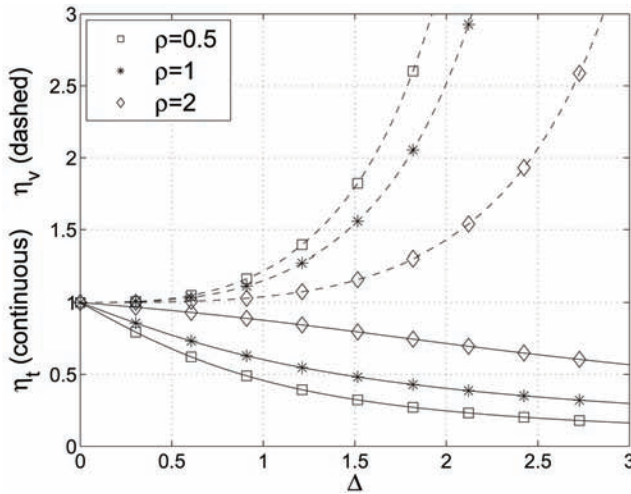


Fig. 1. Performance of the designed censored system,  $\eta_t$  and  $\eta_v$ , as function of the parameter  $\Delta$  for different values of  $\rho$ .

An approximate analysis providing amenable formulas for system performances can be made by using the Central Limit Theorem, see [15], yielding:

$$\eta_t \approx \left( 1 - \frac{1}{\sqrt{2\pi\rho^2}} \frac{e^{-\frac{(\Delta+\rho)^2}{2}} - e^{-\frac{(\Delta-\rho)^2}{2}}}{Q(\Delta+\rho) + Q(\Delta-\rho)} \right)^{-1}, \quad (43)$$

$$\eta_v \approx \frac{\eta_t}{Q(\Delta+\rho) + Q(\Delta-\rho)}, \quad (44)$$

where  $Q(\cdot)$  is the standard Gaussian complementary CDF.

By drawing  $\eta_t$  and  $\eta_v$  as functions of  $\Delta$  for different values of  $\rho$ , we get the curves for the system optimization, as depicted in Fig. 1, from which the desired system operative point in terms of detection delay and sensors energy consumption can be decided. Figure 2 provides the same information and insight of Fig. 1.

An interesting behavior is also observed when the sensors, provided that their observations are informative enough, can only send to the MA the *hard* decisions (*i.e.*, a binary value) taken at a local level. In this case, different from the previous case that the censored log-likelihood are transmitted, it is possible to prove that an optimal censoring level exists, minimizing the detection delay. Examples of applications, as well as detailed discussions of the above aspects, are addressed in [15], to which the reader is referred for details.

### 3.2 Pre-processing at sensor level for detection after transmission over noisy channels

Suppose that the sensors of a network are connected by dedicate channels (parallel architecture) to a fusion center, *i.e.*, some unit devoted to the task of data fusion, and assume

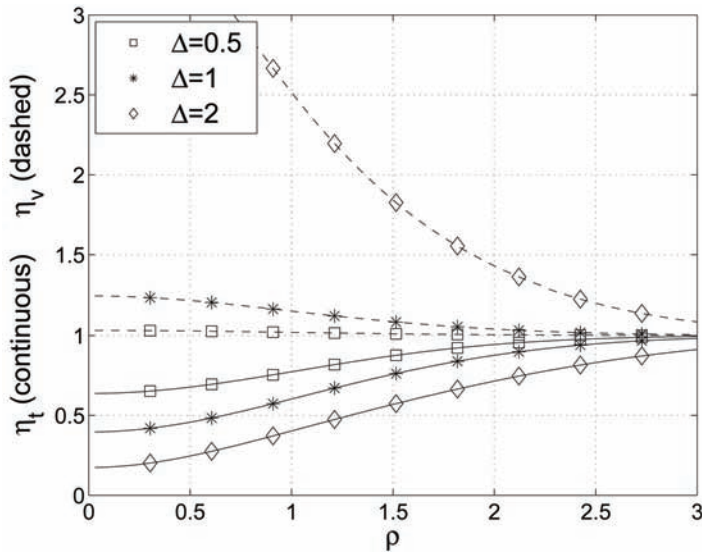


Fig. 2. Performance of the designed censored system,  $\eta_t$  and  $\eta_v$ , as function of the parameter  $\rho$  for different values of  $\Delta$ .

also that such channels are noisy. The issue is to understand if some processing of the data measured at the sensor would increase the detection capabilities of the fusion center. Specifically, we are faced with a detection problem in which remotely observed data are delivered to a fusion center through a certain channel. The fusion center is designed to decide between two mutually exclusive statistical hypotheses, basing its decision upon the received data whose statistical distribution is determined by the underlying hypothesis. Should the observations made at the remote sensor be somehow processed before delivering them over the channel?

We assume that the fusion center implements a sequential test. Motivated by eqs. (27) and (28), it makes sense to choose as a measure of the detection performances the divergence between the distributions under the two hypotheses since this directly impacts the average sample number, *i.e.*, the detection delay. Therefore, the above question can be rephrased in terms of divergence: can we increase the divergence at the output of a noisy channel, by elaborating on its input? It is obvious that, if the channel were ideal (noiseless) the answer is certainly negative in view of the data processing inequality. On the other hand, for noisy channels, the answer is in some case affirmative. Let us limit the following discussion to the case where the noisy channel has *binary* input and output alphabets, and let us model the sought sensor processing as a further channel having as input the original measured data and whose output are the transformed data, to be sent over the physical noisy channel. Formally, we consider the statistical test

$$\begin{aligned} \mathcal{H}_0 &: I \sim p, \\ \mathcal{H}_1 &: I \sim q, \end{aligned} \tag{45}$$

where  $p$  and  $q$  are two arbitrary pmfs (column vectors) with alphabet  $\{1, 2\}$ , that rule the random variable  $I$  modeling the (iid) sensor observations. The sensor delivers data to the



fusion center by a discrete memoryless channel,  $J \rightarrow K$ , whose input may be different from  $I$ . Indeed a possible sensor processing may take place, which is also modeled as a channel  $I \rightarrow J$ . The physical channel is

$$\mathbf{C} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} \quad (46)$$

where  $C_{kj} = \Pr\{K = k \mid J = j\}$ ,  $k, j = 1, 2$ , while the processing channel is

$$\mathbf{H} = \begin{pmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{pmatrix} \quad (47)$$

where  $H_{ji} = \Pr\{J = j \mid I = i\}$ ,  $j, i = 1, 2$ . A convenient, self-explaining, notation is as follows

$$I \xrightarrow{\mathbf{H}} J \xrightarrow{\mathbf{C}} K, \quad (48)$$

where  $\mathbf{C}$  is given, while the task is to find a matrix  $\mathbf{H}$  that maximizes the detection performance at the remote site that observes  $K$ .

Let  $x$  and  $y$  be the pmfs of  $J$  under the two hypotheses, and, similarly, let  $w$  and  $z$  be the pmfs of  $K$ . We have:

$$x = \mathbf{H}p, \quad y = \mathbf{H}q, \quad w = \mathbf{C}\mathbf{H}p, \quad z = \mathbf{C}\mathbf{H}q. \quad (49)$$

The described problem can be cast in the form of an optimization

$$\max_{\mathbf{H}} D(\mathbf{C}\mathbf{H}p \parallel \mathbf{C}\mathbf{H}q), \quad (50)$$

and the following claim can be proved by elementary convex analysis tools: only the following four matrices are candidates for solving the posed optimization problem

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \\ \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}.$$

Clearly, the last two matrices should be ignored since they both lead to zero divergence in terms of the variable  $K$ . Given that the first matrix is the identity, the only possibility for improving the detection performance based on  $K$  is to try with the upper-right matrix, that is to say, to try with a symbol flipping: if  $I = 1$  is observed, then  $J = 2$  is presented at the input of the physical channel  $\mathbf{C}$ , and vice versa.

To elaborate, let us assume that the original detection problem is "difficult", in the sense that the hypotheses  $p$  and  $q$  are very close each other. For instance:

$$q_1 = p_1 + \epsilon, \quad q_2 = p_2 - \epsilon, \quad (51)$$

where  $|\epsilon|$  is small enough. Expanding in series around  $\epsilon = 0$ , we can find

$$\begin{aligned}
 D(\mathbf{C}p \parallel \mathbf{C}q) &\approx \epsilon^2/2 (1 - C_{11} - C_{22})^2 \\
 &\quad [(1 - C_{11} - C_{22})p_1 + C_{22}]^{-1} \\
 &\quad [1 - C_{22} - (1 - C_{11} - C_{22})p_1]^{-1}, \\
 D(\mathbf{C}Fp \parallel \mathbf{C}Fq) &\approx \epsilon^2/2 (1 - C_{11} - C_{22})^2 \\
 &\quad [(1 - C_{11} - C_{22})p_1 + C_{11}]^{-1} \\
 &\quad [1 - C_{11} - (1 - C_{11} - C_{22})p_1]^{-1}.
 \end{aligned}$$

If we want that  $D(\mathbf{C}Fp \parallel \mathbf{C}Fq) > D(\mathbf{C}p \parallel \mathbf{C}q)$  we must have that

$$C_{11}(1 - C_{11})(2p_1 - 1) > C_{22}(1 - C_{22})(2p_1 - 1).$$

The conclusion is that the sensor pre-processing improves the final detection performances if and only if

$$\begin{aligned}
 \text{either} \quad & C_{11}(1 - C_{11}) > C_{22}(1 - C_{22}) \text{ and } p_1 > 1/2, \\
 \text{or} \quad & C_{11}(1 - C_{11}) < C_{22}(1 - C_{22}) \text{ and } p_1 < 1/2.
 \end{aligned} \tag{52}$$

We are currently working on extensions of the topic here briefly described, with a more general formulation, including non binary observations and channels. For the time being, it is important to emphasize that for *binary* and *symmetric* physical channels (i.e.,  $C_{11} = C_{22}$ ) there is no way of improving the performances. We have evidences, however, that for non binary cases the problem exhibits much more structure and provides more useful insights from a practical perspective. What remains true in more general settings, however, is that the optimal pre-processing is deterministic, in the sense that given the input  $I$  the output  $J$  can be determined with probability one, a circumstance with a precise physical meaning.

### 3.3 Noise enhanced sequential detectors [24]

Let us consider a *fully decentralized* sensor network without fusion center, designed for an inference task. In this typical architecture, each node senses the environment and collects data about a phenomenon to be monitored, think for instance of a binary detection problem where the challenge is to decide which of two possible statistical distributions actually rules the observations. The lack of fusion center is remediated by suitable inter-node communication protocols that allow the system to exchange data up to make the final decision. These data, due to often unavoidable physical constraints, are here assumed to be some nonlinear transformation  $t(\cdot)$  of the original observations.

Specifically, the  $i^{th}$  node of the network computes  $t(X_i)$ , where  $X_i$  is the sensed sample, and delivers such a value to one of its neighbors, say node  $j$ . This latter computes  $t(X_i) + t(X_j)$  and delivers that to node  $k$ , and so on. The decision process is sequential: as soon as the value computed at some node exceeds given thresholds the decision is taken, and the task is terminated.

With this model in mind, motivated by recent advances in noise enhanced and stochastic resonance detection, in [24] the question is posed if adding a “noise” sample, say  $W_i$ , to the measurement made at node  $i$  before computing the nonlinearity  $t(\cdot)$ , could provide any benefit in terms of the final detection performances. This at first glance counterintuitive question may have (surprisingly?) a positive answer: there exist cases in which adding noise is beneficial!

### 3.3.1 Problem formalization

Let us consider first the original shift-in-mean binary hypothesis test:

$$\begin{aligned} \mathcal{H}_0 &: X_i \sim f_X(x + A), \\ \mathcal{H}_1 &: X_i \sim f_X(x - A). \end{aligned}$$

Here  $i = 1, 2, \dots, \infty$ , represents the sensor number,  $X_i$  is the observation made at node  $i$  (observations are iid), and the pdf  $f_X(x)$  is an even function. According to formulation (7), the SPRT for this problem would be

$$\sum_{i=1}^n \log \frac{f_X(x_i - A)}{f_X(x_i + A)} \begin{cases} \geq \gamma_1 & \Rightarrow \text{decide } \mathcal{H}_1, \\ \leq -\gamma_0 & \Rightarrow \text{decide } \mathcal{H}_0, \\ \text{otherwise} & \Rightarrow \text{go ahead,} \end{cases} \quad (53)$$

but, in many cases of interest, implementing such an optimal SPRT is unfeasible [25]. Therefore, we consider sub-optimal sequential detectors and, as explained before, we also contaminate the original observations with iid noise: this latter effect amounts to consider the noise contaminated observables  $Y_i = X_i + W_i$ , in place of the original  $X_i$ . The noise density is also assumed even-symmetric:  $f_W(w) = f_W(-w)$ , and that of the contaminated samples becomes the convolution of the two:  $f_Y(y) = f_X(x) * f_W(w)$ .

The said sub-optimality of the detector amounts to work with a decision statistic in the form:

$$T_n = \sum_{i=1}^n t(y_i), \quad (54)$$

where  $t(y)$  is a bounded and non-decreasing odd function. To simplify the analysis, let the error probabilities  $1 - P_d$  and  $P_f$  of the sequential test, see eqs. (8) and (9), be equal, and denote by  $P_e$  such value. We have

$$T_n \begin{cases} \geq \gamma & \text{decide } \mathcal{H}_1 \text{ and stop testing,} \\ \leq -\gamma & \text{decide } \mathcal{H}_0 \text{ and stop testing,} \\ \text{otherwise} & \text{take the } (n + 1)^{\text{th}} \text{ sample (i.e., increase } n), \end{cases}$$

where the thresholds  $\gamma$  and  $-\gamma$  are symmetric as consequence of the assumed problem symmetries.

The above test is not a standard SPRT, since  $T_n$  is not the log-likelihood. Therefore, the system performances can be obtained as discussed in Sect. 2.3. First, Wald's equality (25), under hypothesis  $\mathcal{H}_1$ , yields, in the regime of small  $P_e$ ,

$$E[N] \approx \frac{\gamma}{E[t(X + W) | \mathcal{H}_1]}. \quad (55)$$

Furthermore, the exponential bound (31) is used:

$$P_e \leq e^{-\gamma r^*}, \quad (56)$$

where  $r^*$  solves

$$\mathbf{E} \left[ e^{r^* t(X+W)} | \mathcal{H}_0 \right] = 1. \tag{57}$$

The denominator of (55) can be written

$$\mathbf{E}[t(X + W) | \mathcal{H}_1] = \iint f_X(x - A) f_W(w) t(x + w) dx dw,$$

and, introducing the function

$$h_1(w) = \int f_X(x - A) t(x + w) dx, \tag{58}$$

yields

$$\mathbf{E}[t(X + W) | \mathcal{H}_1] = E_W[h_1(W)].$$

Similarly, we have

$$\mathbf{E} \left[ e^{r t(X+W)} | \mathcal{H}_0 \right] = \iint f_X(x + A) f_W(w) e^{t(x+w)r} dx dw,$$

which, introducing the function

$$h_2(w, r) = \int f_X(x + A) e^{t(x+w)r} dx, \tag{59}$$

yields

$$\mathbf{E} \left[ e^{r t(X+W)} | \mathcal{H}_0 \right] = E_W[h_2(W, r)].$$

The two above functions allow us to write compact formulas for the system analysis. Indeed, in [24] it is found that

$$\begin{aligned} \mathbf{E}[N] &\approx \frac{\gamma}{E_W[h_1(W)]}, \\ P_e &\approx e^{-\gamma r^*}, \quad r^* : E_W[h_2(W, r^*)] = 1. \end{aligned} \tag{60}$$

The optimal performance-enhancing noise density  $f_W(w)$  must minimize the expected sample number, without increasing the error probability. Using our approximations and bounds, this amounts to

$$\max_{f_W: r^* \geq r_0^*} E_W[h_1(W)], \tag{61}$$

where  $r_0^*$  corresponds to the noise-free case.

### 3.3.2 Example: sign detector

Assume now that the nonlinearity is the sign function, which amounts to one-bit quantized observations

$$t(x) = \text{sign}(x).$$

The performance functions  $h_1(w)$  and  $h_2(w, r)$  can be found to be

$$h_1(w) = 2F_X(A + w) - 1 \quad (62)$$

$$h_2(w, r) = e^{-r} F_X(A - w) + e^r [1 - F_X(A - w)], \quad (63)$$

where  $F_X(\cdot)$  is the cumulative distribution function of the random variable  $X$ . By introducing the quantity

$$p = \Pr[X + W > 0; \mathcal{H}_1] = E_W[F_X(A + W)], \quad (64)$$

we finally get

$$E_W[h_1(W)] = 2p - 1, \quad (65)$$

$$\begin{aligned} E_W[h_2(W, r)] &= e^{-r} E_W[F_X(A + W)] + e^r E_W[1 - F_X(A + W)] \\ &= e^r - 2p \sinh(r). \end{aligned} \quad (66)$$

Enforcing  $E_W[h_2(W, r)] = 1$  yields as solution

$$r^* = \log \frac{p}{1-p}. \quad (67)$$

Equation (61) requires a constrained maximization of  $E_W[h_1(W)] = 2p - 1$ , where the constraint is  $r^* = \log \frac{p}{1-p} \geq r_0^*$ , with  $r_0^*$  being the value of  $r^*$  corresponding to the absence of injected noise. However this can be shown to be equivalent to its unconstrained counterpart, which amounts to simply find the maximum achievable value of  $p$ . Remarkably, it is also possible to prove that the optimizing noise density can be chosen in the class of the *coin flipping* distributions:

$$f_W(w) = \frac{1}{2} [\delta(w - w_0) + \delta(w + w_0)].$$

We report now the evidences of some numerical simulations aimed at checking the goodness of the found formulas, as well as the potential benefits of adding noise to the observations for detection purposes. We choose for the observation density a mixture of Gaussians:

$$f_X(x) = \frac{1}{2\sqrt{2\pi}} \left[ e^{-\frac{(x-\mu/2)^2}{2}} + e^{-\frac{(x+\mu/2)^2}{2}} \right].$$

Moreover, we assume that the detection threshold  $\gamma$  is fixed in such a way to yield, in absence of noise, an error probability  $P_{e0} \approx 10^{-2}$ .

In Fig. 3, top plot, the ASN is displayed as function of different injected noise depth  $w_0$ . The theoretical formulas reasonably match the simulation points, and an optimal value of  $w_0$  minimizing the sample number is clearly present. To get the complete perspective, the actual error probability  $P_e$  is also displayed in the bottom plot of the same figure. It can be seen that, for any  $w_0$ , while the ASN decreases, the error probability is kept below the value  $P_{e0}$ . Remarkably, in correspondence of the optimal  $w_0$ , the actual  $P_e$  is in effect orders of magnitude smaller than the design value  $10^{-2}$ .

## 4. Conclusions

In many instances of Wireless Sensor Networks (WSNs) designed for detection purposes, the fusion center is a mobile device that sequentially queries the nodes of the network. Such

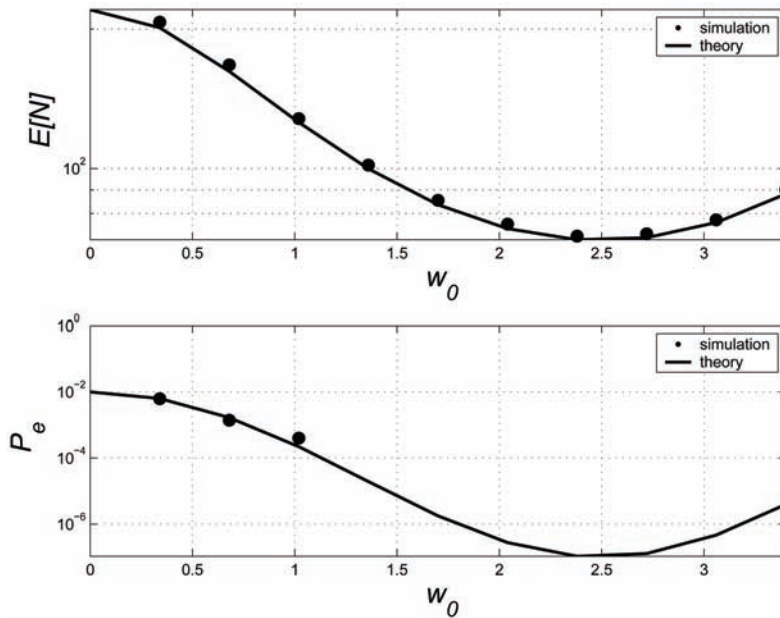


Fig. 3. Sequential detector with fixed threshold  $\gamma$ , for the sign nonlinearity. The expected sample number (top panel) and the error probability (bottom panel) of the proposed sequential detector are displayed as function of the injected noise depth  $w_0$ . The simulation parameters are  $A = 1$ ,  $\mu = 5$ , and  $P_{e0} = 10^{-2}$ . Simulation points are obtained by  $10^4$  Monte Carlo trials and, in bottom panel, we draw only the simulation points that do not fall under the accuracy of the numerical procedure.

a sequential architecture, known as the SENMA structure, fits well the sequential detection paradigms, *i.e.*, the SPRT and its variants. Aside from the SENMA scenarios, the typical tools of sequential detection are exploited much more in general, in various guises, in a variety of WSN applications. This paper provides a succinct introduction to sequential analysis and presents several examples of applications to detection problems.

The main aim of this paper is to introduce the reader to the very powerful tool of sequential analysis, providing the basic insights and useful entry points to some topical literature. The specific issues presented in the first part of this work are selected in order to provide the necessary theoretical background for the applicative examples discussed in the second part of the paper. These latter examples, on the other hand, reflect the authors' recent research on the subject. As a consequence, neither the theory nor the applications are exhaustive or complete in any sense. However, the paper is rather self-consistent and can help in gain a first understanding of the topic.

### 5. References

[1] A. Wald, *Sequential Analysis*. New York: Dover, 1947.  
 [2] J. Doob, *Stochastic Processes*. New York: John Wiley & Sons, 1953.  
 [3] G. Shafer and V. Vovk, *Probability and finance. It's only a game*. New York: John Wiley & Sons, 2001.

- [4] S. Karlin and H. M. Taylor, *A first Course in Stochastic Processes*, 2nd ed. San Diego, CA: Academic Press, 1975.
- [5] S. Ross, *Stochastic Processes*, 2nd ed. New York: John Wiley & Sons, Inc., 1996.
- [6] R. G. Gallager, *Discrete Stochastic Processes*. Kluwer Academic Publisher, 1996.
- [7] H. V. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1988.
- [8] T. Cover and J. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, 1991.
- [9] S. Kullback, *Information Theory and Statistics*. New York: Dover, 1968.
- [10] A. G. Tartakovsky, "Sequential composite hypothesis testing with dependent nonstationary observations," *Problems of Information Transmission*, vol. 17, pp. 18–28, 1981.
- [11] T. L. Lai, "Pitman efficiencies of sequential tests and uniform limit theorems in nonparametric statistics," *The Annals of Statistics*, vol. 6, no. 5, pp. 1027–1047, 1978.
- [12] A. G. Tartakovsky and I. A. Ivanova, "Approximations in sequential rules for testing composite hypotheses and their accuracy in the problem of signal detection from postdetector data," *Prob. Inform. Transm.*, vol. 28, no. 1, pp. 55–64, Jan.-Mar. 1992.
- [13] W. Feller, *An Introduction to Probability and Its Applications*, Volume 2. New York: John Wiley & Sons, 1971.
- [14] D. R. Cox and H. D. Miller, *The theory of stochastic processes*. Chapman & Hall, 1964.
- [15] P. Addesso, S. Marano, and V. Matta, "Sequential sampling in sensor networks for detection with censoring nodes," *IEEE Trans. Signal Processing*, vol. 55, no. 11, pp. 5497–5505, Nov. 2007.
- [16] L. Tong, Q. Zhao, and S. Adireddy, "Sensor networks with mobile agents," in *Proceedings of MILCOM 2003*, Boston MA, Oct. 2003.
- [17] B. M. Sadler, "Fundamentals of energy-constrained sensor network systems," *IEEE Aerosp. Electron. Syst. Mag*, vol. 20, no. 8, Aug. 2005.
- [18] S. Appadwedula, V. Veeravalli, and D. Jones, "Energy-efficient detection in sensor networks," *IEEE J. Select. Areas Commun.*, vol. 23, no. 4, pp. 693–702, Apr. 2005.
- [19] —, "Energy-efficient detection in sensor networks," in *Proceedings of Intern. Conf. on Information Fusion (FUSION 2003)*, Cairns, Queensland, Australia, July 2003, pp. 764–771.
- [20] C. Rago, P. Willett, and Y. Bar-Shalom, "Censoring sensors: a low-communication-rate scheme for distributed detection," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 2, pp. 554–568, Apr. 1996.
- [21] S. Appadwedula, V. Veeravalli, and D. Jones, "Robust and locally-optimum decentralized detection with censoring sensors," in *Proceedings of Intern. Conf. on Information Fusion (FUSION 2002)*, Annapolis, MD, USA, July 2002, pp. 56–63.
- [22] N. Patwari, I. Hero, A.O., and B. Sadler, "Hierarchical censoring sensors for change detection," in *IEEE Workshop on Statistical Signal Processing*, Sept. 27-Oct. 29 2003, pp. 21–24.
- [23] P. Willett and L. Tong, "One aspect to cross-layer design in sensor networks," in *Proceedings of MILCOM 2004*, Monterey, CA, USA, Oct. 2004, pp. 688–693.
- [24] M. Guerriero, S. Marano, V. Matta, and P. Willett, "Stochastic resonance in sequential detectors," *IEEE Trans. Signal Processing*, in print.
- [25] H. Chen, P. K. Varshney, S. M. Kay, and J. H. Michels, "Theory of the stochastic resonance effect in signal detection: Part I-Fixed detectors," *IEEE Trans. Signal Processing*, vol. 55, no. 7, pp. 3172–3184, July 2007.

# Parameter Estimation Over Noisy Communication Channels in Distributed Sensor Networks

Thakshila Wimalajeewa<sup>1</sup>, Sudharman K. Jayaweera<sup>1</sup> and Carlos Mosquera<sup>2</sup>

<sup>1</sup>*Dept. of Electrical and Computer Engineering, University of New Mexico,*

<sup>2</sup>*Dept. de Teoría de la Señal y Comunicaciones, Universidad de Vigo,*

<sup>1</sup>*USA*

<sup>2</sup>*Spain*

## 1. Introduction

With the recent development of low-cost, low-power, multi-functional sensor nodes, sensor networks have become an attractive emerging technology in a wide variety of applications including, but not limited to, military surveillance, civilian, industrial and environmental monitoring [1]–[5]. In most of these new applications sensor nodes are capable not only of sensing but also of data processing, wireless communications and networking. It can be argued that it is their ability of ad hoc wireless networking that has attracted much interest to wireless sensor networks in recent years.

A typical sensor network may consist of a large number of spatially distributed nodes to make a decision on a Parameter of Interest (PoI). This can be detection, estimation or tracking of a target or multiple targets. Once the network is deployed, the network resources, such as node power and communication bandwidth, are limited in many situations. This is due to the fact that reinstalling and recharging the batteries might be difficult, or even impossible, once the network is deployed. A common question arising in such networks is how to effectively combine the information from all the nodes in the network to arrive at a final decision while consuming the resources in an optimum way. In a distributed sensor network, the distributed nodes make observations of PoI and process them locally to make a summary of their observations. The final decision is usually made by combining these locally processed data.

Once local decisions are made at each individual sensor node, the natural questions are how to combine the local decisions and where the final decision is made. When there is a possibility that the sensor network can have a central node (generally called as the fusion center) with relatively high processing power compared to distributed nodes, the summary of the local observations can be sent to the fusion center. The fusion center combines all local decisions in an optimum way to arrive at a final decision in what is known as the centralized architecture. The disadvantage of such a system is that if there is a failure in the fusion center, the whole network fails. On the other hand, in some applications, it might be of interest that nodes communicate with each other to reach at a final decision without



depending on a central node. In this set-up the node that makes the final decision may change over time due to the dynamic nature of the sensor network and the PoI. This may lead to a more robust architecture compared to that with centralized architecture.

Irrespective of the data fusion architecture, the local information from sensor nodes needs to be shared over a communication channel that, in general, can undergo both path loss attenuation and multipath fading. As a result, the received signal at a destination node, be it another distributed node or a central fusion center, is corrupted by both multiplicative and additive noise. The performance of the final decision will depend not only on measurement noise at the distributed nodes but also on channel quality of communication links.

The performance of resource-constrained wireless sensor networks with communication and measurement noise has been addressed by many authors in different contexts. For example, performance of the sensor networks under power and bandwidth constraints are analyzed in [6]–[22] and [10], [11], [23]–[36], respectively. Collaborative signal processing, including sequential communication, is addressed in [15], [37]–[42].

In this chapter we address the problem of multi-sensor data fusion over noisy communication channels. The objective of the sensor network is to estimate a deterministic parameter. Distributed nodes make noisy observations of the PoI. Each node generates either an amplified version of its own observation or a quantized message based on its own observation, and shares it with other nodes over a wireless channel. The final decision can be made either at a central node (fusion center) or fully distributively. In the case of centralized architecture, the locally processed messages can be sent to the fusion center over a set of orthogonal channels or a multiple-access channel in which nodes share a common communication channel. In the fully distributive architecture, there is no explicit central fusion center and nodes communicate with each other to arrive at a final decision. There are several variations of this architecture: in one setting, nodes may communicate sequentially with neighbors to sequentially update an estimator (or a sufficient statistic for the parameter). The final decision can be declared by any node during this sequential updating process. On the other hand, it might sometimes be of interest for all nodes in the network to arrive at a common final decision. This leads to a distributed consensus estimation problem. Note that, here all nodes communicate with each other in contrast to the sequential communication architecture above until they reach an agreement.

The rest of this chapter is organized as follows: Section 2 formulates the problem of parameter estimation over noisy communication channels in a distributed sensor network. Section 2-A presents the assumed observation model. The ideal centralized estimation is reviewed in Section 2-B. Ideal estimation is, of course, not possible in a wireless sensor network since communication is over a noisy channel and the network is constrained by available communication resources. By sharing only a summary of the observations with each other, the scarce communication resources can be efficiently utilized. Local processing schemes to achieve this goal are discussed in Section 2-C.

Section 3 focuses on centralized estimation architecture with noisy communication channels between distributed nodes and the fusion center. Estimation performance with orthogonal channels is discussed in Section 3-A and that with non-orthogonal communication channels is discussed in Section 3-B.

Sections 4 and 5 discuss the distributed estimation performance in a sensor network with collaborative information processing. Section 4 considers the distributed sensor network architecture with sequential communication where inter-sensor communication links are

assumed to be noisy. In Section 5 collaborative estimation with distributed consensus is addressed. Here, the nodes are allowed to communicate with a set of other nodes that are considered as their neighbors. Sections 5-A and 5-B address static parameters whereas section 5-C considers, time-varying parameters. Finally chapter summary is given in section 6.

## 2. Data fusion problem in a wireless sensor network

Throughout, we consider a spatially distributed sensor network that is deployed to estimate a PoI. It is natural to expect that a final decision be obtained by combining the information from different nodes. In a distributed sensor network, nodes share summaries of their observations over noisy communication channels. Since network resources, in particular the node power and the communication bandwidth, are scarce it is important that the observations at each node are locally processed to reduce the observation to a concise summary. The final decision can then be made based on these local outputs that nodes share with each others and/or with a fusion center.

### A. Multi-sensor observation model

We consider a situation in which multiple sensors observe a PoI. When these nodes form a sensor network, the final decision can be made in either a centralized or distributed way. In the centralized architecture, each node sends a summary of its observations to a central node called a fusion center. There is no inter-node communication. The fusion center combines all received information in an effective way to arrive at a final decision. In the distributed decision-making architecture, on the other hand, the nodes collaborate with each other to arrive at a final decision distributively, without the aid of a central fusion node.

Irrespective of the architecture, communication between sensors and the fusion center, or among sensors, is over a noisy channel. Thus, the information sent sees distortion due to both additive as well as multiplicative noise. The multiplicative noise is due to path loss attenuation and multipath fading encountered, for example, in a wireless channel. In this section, we first consider the centralized architecture as shown in Fig. 1. The distributed architecture is covered in Sections 4 and 5.

Consider a spatially distributed network of  $n$  sensors. Let us assume that the network is to estimate, in general, a vector parameter  $\Theta$  where  $\Theta$  is a  $p$ -vector. The observation at each node is related to the parameter  $\Theta$  that we wish to estimate via the following observation model;

$$z_k(t) = f_k(\Theta) + v_k(t),$$

where  $z_k(t)$  is the observation at the  $k$ -th node at time  $t$ ,  $f_k : \mathbb{R}^p \rightarrow \mathbb{R}$  is a function of the parameter vector  $\Theta$  (in general, non-linear) and  $v_k(t)$  is the additive observation noise at node  $k$ . In the special case of linear observation model, the joint observation vector at  $n$  nodes at time  $t$  can be written as

$$\mathbf{z}(t) = \mathbf{B}\Theta + \mathbf{v}(t), \tag{1}$$

where  $\mathbf{B}$  is an  $n \times p$  (known) matrix and  $\mathbf{v}$  is the observation noise vector having a zero mean and a covariance matrix of  $\Sigma_v$ . In this chapter we focus mainly on scalar parameter

estimation (where we assume  $p = 1$ ) although the techniques developed and the results can easily be extended to vector parameter estimation. For a scalar parameter, the observation vector (1) formed by observations at all  $n$  nodes reduces to,

$$\mathbf{z} = \theta \mathbf{e} + \mathbf{v}, \quad (2)$$

where we have suppressed the timing index  $t$  and  $\mathbf{e}$  is the  $n$ -vector of all ones.

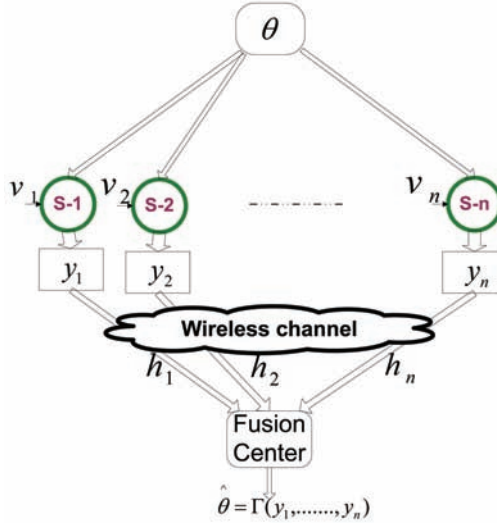


Fig. 1. Distributed estimation with a central fusion center

### B. Ideal centralized data fusion

When local observation vector  $\mathbf{z}$  is directly available at the fusion center, the problem is termed the ideal centralized data fusion. The optimal final estimator and its mean-squared error performance are summarized in the following lemma:

*Lemma 1:* [43] *When the observation vector (2) is available at the fusion center, the best linear unbiased estimator (BLUE) for the scalar parameter  $\theta$  is given by*

$$\hat{\theta}(\mathbf{z}) = \frac{\mathbf{e}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{z}}{\mathbf{e}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{e}}, \quad (3)$$

where  $x^T$  denotes the transpose of  $x$ . The corresponding mean squared error (MSE) achieved by (3) is

$$MSE = \mathbb{E}\{|\hat{\theta} - \theta|^2\} = (\mathbf{e}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{e})^{-1}, \quad (4)$$

where  $\mathbb{E}\{\cdot\}$  denotes the mathematical expectation. Further, if the local observations are i.i.d., so that  $\Sigma_{\mathbf{v}} = \sigma_v^2 \mathbf{I}$ , where  $\mathbf{I}$  is the  $n \times n$  identity matrix, the estimator in (3) simplifies to the sample mean of the observations,

$$\hat{\theta}(\mathbf{z}) = \frac{1}{n} \sum_{k=1}^n z_k,$$

with the corresponding MSE in (4) simplified to

$$MSE = \frac{\sigma_v^2}{n}.$$

Since communication from distributed nodes to the fusion center is over noisy channels, in practice signals transmitted by the distributed nodes undergo distortion. Hence, direct access from a distant fusion center to the exact observations at distributed nodes may not be possible. However, the lemma 1 will serve as a benchmark for other schemes that we will discuss in this chapter.

### C. Local processing at sensor nodes

To facilitate efficient utilization of node and network resources, each node in a sensor network locally processes its observation to generate a useful summary. The transmitted signal at the  $k$ -th node is then given by  $y_k = g_k(z_k)$ . In the following we consider two local processing schemes:

1) *Amplify-and-Forward (AF) local processing*: In many practical situations where sensor observations are corrupted by additive noise, the amplify and forward strategy has been shown to perform well. In this method, each node directly amplifies its observation and sends it to the fusion center. The transmitted signal from node  $k$  is

$$y_k = g_k z_k,$$

where  $g_k$  is the amplifying gain at the  $k$ -th node. In order to save the node power, it is important to select the amplification gain  $g_k$  for  $k = 1, \dots, n$  appropriately depending on the other network parameters such as channel quality and observation quality, etc.. If nodes are operated at the same power level, sometimes it may lead to an unnecessary usage of the network power especially when observation qualities of nodes and channel qualities are not the same for all nodes. Therefore, choosing  $g_k$ 's in a meaningful way is an important issue to be addressed in designing resource-constrained sensor networks. This problem is discussed in section 3.

With AF local processing, the received signal vector at the fusion center with noiseless communication is given by

$$\mathbf{r} = \mathbf{G}\mathbf{z}, \quad (5)$$

where  $\mathbf{G} = \text{diag}(g_1, \dots, g_n)$  is the channel gain matrix. Then the Best Linear Unbiased Estimator and its corresponding mean squared error is given by the following lemma:

*Lemma 2: [34], [43] If the received signal at the fusion center is as given in (5), then the BLUE estimator based on the received signal vector is given by*

$$\hat{\theta}(\mathbf{r}) = \frac{\mathbf{e}^T \boldsymbol{\Sigma}_v^{-1} \mathbf{G}^{-1} \mathbf{r}}{\mathbf{e}^T \boldsymbol{\Sigma}_v^{-1} \mathbf{e}},$$

and the corresponding MSE is

$$MSE = \mathbb{E}\{|\hat{\theta} - \theta|^2\} = (\mathbf{e}^T \boldsymbol{\Sigma}_v^{-1} \mathbf{e})^{-1}.$$

Further, if the local observations are i.i.d., so that  $\boldsymbol{\Sigma}_v = \sigma_v^2 \mathbf{I}$ , the MSE simplifies to

$$MSE = \frac{\sigma_v^2}{n}.$$



Fig. 2. Probabilistic quantization

2) *Quantized local processing*: To save node energy and communication bandwidth, sensors can compress their observations before transmitting to the fusion center. In this set up, local nodes quantize their observations to generate finite-range messages  $m_k(z_k)$  where each  $m_k$  is represented by  $L_k$  number of bits [9]. Based on the quantized messages received from nodes, the fusion center computes the final estimator

$$\hat{\theta} = \Gamma(\hat{m}_1, \hat{m}_2, \dots, \hat{m}_n)$$

where  $\hat{m}_k$ 's are the corrupted versions of quantized messages  $m_k$ 's received at the fusion center and  $\Gamma(\cdot)$  is the final estimator mapping.

There are several quantization schemes proposed in the literature each having its own advantages and disadvantages [9], [22], [21]. For simplicity, throughout this chapter we concentrate on the universal decentralized quantization scheme given in [9]. According to this scheme, each node locally quantizes its own observation  $z_k$  into a discrete message  $m_k(z_k, L_k)$  of  $L_k$  bits. Due to the lack of knowledge of probability density function (pdf) of noise, the quantizer  $Q_k: z_k \rightarrow m_k(z_k, L_k)$  at local nodes is designed to be a uniform randomized quantizer [9]. To that end, suppose the observation range of each sensor is  $[-W, W]$  where  $W$  is a known parameter determined by the physical properties of the sensor nodes. At each node the range  $[-W, W]$  is divided into  $2^{L_k}-1$  intervals of length  $\Delta_k = 2W / (2^{L_k} - 1)$  each as shown in Fig. 2. The quantizer  $Q_k$  rounds-off  $z_k$  to the nearest endpoint of one of these intervals in a probabilistic manner. For example, suppose,  $i\Delta_k \leq z_k < (i+1)\Delta_k$  where  $-2^{L_k-1} \leq i \leq 2^{L_k-1}$ . Then  $Q_k$  will quantize  $z_k$  into  $m_k(z_k, L_k)$  so that

$$P\{m_k(z_k, L_k) = i\Delta_k\} = 1 - r,$$

and

$$P\{m_k(z_k, L_k) = (i+1)\Delta_k\} = r,$$

where  $r \equiv (z_k - i\Delta_k) / \Delta_k \in [0, 1]$ . Note that the quantizer noise  $q_k(z_k, L_k) = m_k(z_k, L_k) - z_k$  is then a Bernoulli random variable taking values of  $q_k(z_k, L_k) = -r\Delta_k$  and  $q_k(z_k, L_k) = (1 - r)\Delta_k$  with probabilities

$$P\{q_k(z_k, L_k) = -r\Delta_k\} = 1 - r,$$

and

$$P\{q_k(z_k, L_k) = (1 - r)\Delta_k\} = r.$$

With this local processing scheme the quantized message at node  $k$  can be expressed as

$$m_k(z_k, L_k) = z_k + q_k = \theta + v_k + q_k; \text{ for } k = 1, \dots, n, \quad (6)$$

where we have made use of (2). Note that the quantization noise  $q_k$  and the observation noise  $v_k$  in (6) will be assumed to be independent. Moreover,  $q_k$  is independent across sensors since quantization is performed locally at each sensor.

It can be easily shown that  $m_k(z_k, L_k)$  is an unbiased estimator of  $\theta$  so that  $\mathbb{E}\{m_k\} = \theta$  with the MSE (which is, in fact, the variance of the estimator) upper bounded as

$$V_k(m_k) \leq \frac{W^2}{(2^{L_k} - 1)^2} + \sigma_v^2 = \delta_k^2 + \sigma_v^2 \quad \text{for } k = 1, \dots, n, \tag{7}$$

where  $\delta_k^2 = \frac{W^2}{(2^{L_k} - 1)^2}$ . Hereafter we use the short-hand notation  $m_k$  to denote  $m_k(z_k, L_k)$ , so that the transmitted signal  $y_k$  at node  $k$  is  $y_k = m_k$  for  $k = 1, \dots, n$ .

With quantized processing, the received signal vector at the fusion center, with noiseless communication is

$$\mathbf{r} = \mathbf{m} = \theta \mathbf{e} + \mathbf{v} + \mathbf{q} \tag{8}$$

where  $\mathbf{q} = [q_1, \dots, q_n]^T$  is the quantization noise vector and  $\mathbf{m} = [m_1, \dots, m_n]^T$ . The BLUE estimator at the fusion center and its performance are characterized in lemma 3 below.

*Lemma 3: [9] The BLUE estimator based on the received signal in (8) is given by*

$$\hat{\theta}(\mathbf{m}) = \frac{\mathbf{e}^T (\Sigma_v + \Sigma_q)^{-1} \mathbf{m}}{\mathbf{e}^T (\Sigma_v + \Sigma_q)^{-1} \mathbf{e}},$$

where  $\Sigma_q = \text{diag}(\delta_1^2, \dots, \delta_n^2)$ . An upper bound for the MSE of above estimator can be found to be (using(7))

$$MSE = \mathbb{E}\{|\hat{\theta} - \theta|^2\} \leq (\mathbf{e}^T (\Sigma_v + \Sigma_q)^{-1} \mathbf{e})^{-1}. \tag{9}$$

When local observations are i.i.d. the MSE upper bound (9) can be further simplified as

$$MSE \leq \left( \sum_{k=1}^n \frac{1}{\sigma_v^2 + \delta_k^2} \right)^{-1}. \tag{10}$$

Of course, in practice the above ideal estimators cannot be realized due to imperfect communications between distributed nodes and the fusion center. These imperfections can be due to multiplicative noise (caused by channel fading and path loss attenuation) and additive noise at the receiver. When the sensor system has to conform with resource constraints on node power and communication bandwidth, it is important to consider the minimum achievable error performance taking into account these channel imperfections. Parameter estimation under imperfect communications in a distributed sensor network is discussed in the next section.

### 3. Optimal decision fusion over noisy communication channels

The performance of a final estimator when locally processed data are transmitted to the destination over a noiseless channel was discussed in the latter part of Section 2. In this section we discuss the final estimator performance at a fusion center in the presence of noisy communication channels from distributed nodes to the fusion center. In the following we

consider two communication schemes where sensors transmit data over orthogonal or non-orthogonal channels.

#### A. Communication over orthogonal channels

When locally processed sensor data are transmitted through orthogonal channels (either TDMA, FDMA or CDMA), the received signal vector at the fusion center can be written as

$$\mathbf{r} = \mathbf{H}_c \mathbf{u} + \mathbf{w}, \quad (11)$$

where  $\mathbf{H}_c = \text{diag}(h_1, \dots, h_n)$  are the fading coefficients of each channel and  $\mathbf{w}$  is the receiver noise vector with mean zero and the covariance matrix  $\Sigma_w$ . Note that in (11) we have assumed flat fading channels between sensors and the fusion center which can be a reasonable assumption in certain WSN's but not all. When the channels are selective one can modify (11) by using a tapped-delay line model. The statistics of  $h_k$  is determined by the type of fading distributions. Throughout this chapter we will assume that  $h_k$ 's are Rayleigh distributed.

1) *AF local processing*: With AF local processing and orthogonal communication channels, the received signal vector at the fusion center is given by

$$\begin{aligned} \mathbf{r} &= \mathbf{H}_c \mathbf{G} \mathbf{z} + \mathbf{w} \\ &= \mathbf{H}_c \mathbf{G} \mathbf{e} \theta + \mathbf{n}, \end{aligned} \quad (12)$$

where  $\mathbf{n} = \mathbf{H}_c \mathbf{G} \mathbf{v} + \mathbf{w}$  is the effective noise vector at the fusion center with mean zero and covariance matrix  $\Sigma_n = \mathbf{H}_c \mathbf{G} \Sigma_v \mathbf{G} \mathbf{H}_c + \Sigma_w$ , assuming that the receiver noise and the node observation noise are independent. In the following lemma we summarize the optimal estimator at the fusion center based on the received signal (12) and its performance:

*Lemma 4*: [34] *If the fusion center has the knowledge of channel fading coefficients, the BLUE estimator and the MSE based on the received signal (12) is given by*

$$\hat{\theta}(\mathbf{r}) = \frac{\mathbf{e}^T \mathbf{G} \mathbf{H}_c \Sigma_n^{-1} \mathbf{r}}{\mathbf{e}^T \mathbf{G} \mathbf{H}_c \Sigma_n^{-1} \mathbf{H}_c \mathbf{G} \mathbf{e}}, \quad (13)$$

and

$$MSE = (\mathbf{e}^T \mathbf{G} \mathbf{H}_c \Sigma_n^{-1} \mathbf{H}_c \mathbf{G} \mathbf{e})^{-1}. \quad (14)$$

*In the special case when local observations and the receiver noise are both i.i.d. such that  $\Sigma_v = \sigma_v^2 \mathbf{I}$  and  $\Sigma_w = \sigma_w^2 \mathbf{I}$ , the BLUE estimator (13) and the MSE (14) further simplify to*

$$\hat{\theta}(\mathbf{r}) = \frac{\sum_{k=1}^n \frac{g_k h_k r_k}{\sigma_v^2 h_k^2 g_k^2 + \sigma_w^2}}{\sum_{k=1}^n \frac{h_k^2 g_k^2}{\sigma_v^2 h_k^2 g_k^2 + \sigma_w^2}}, \quad (15)$$

and

$$MSE = \left( \sum_{k=1}^n \frac{h_k^2 g_k^2}{\sigma_v^2 h_k^2 g_k^2 + \sigma_w^2} \right)^{-1}, \quad (16)$$

where  $\sigma_w^2$  is the receiver noise power.

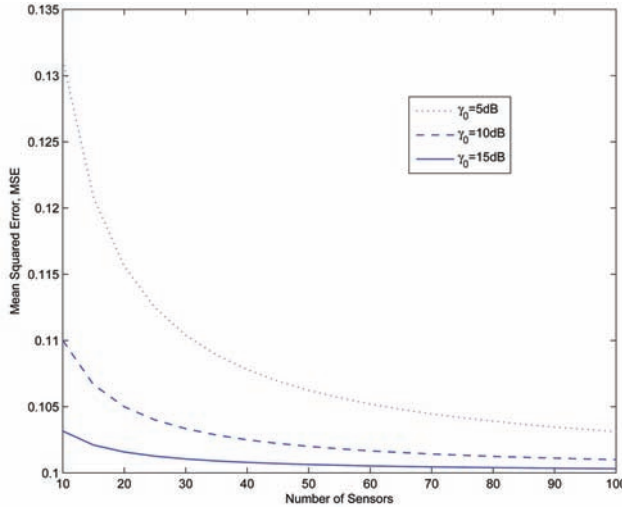


Fig. 3. Mean squared error performance vs. number of nodes: The total network power is constant.

The performance of the BLUE estimator (15) is shown in Figs. 3 and 4 given that the total power in the network is constant. Note that in the both Figs. 3 and 4 the node power is the same at each sensor, so that  $g_k = g$  for  $k = 1, \dots, n$  and each channel gain is unity (i.e.  $h_k=1$  for all  $k$ ). Hence, if total network power is  $P_T$  then the individual node power is given by  $g^2 = P_T/n$ . In this case, the MSE in (16) is further simplified to  $MSE = \frac{\sigma_v^2}{n} + \frac{\sigma_w^2}{P_T}$ . The local SNR,  $\gamma_0$  is defined as  $P_s/\sigma_v^2$  where  $P_s$  is the average power at local nodes. In the simulations we have let  $P_s = 1$ . It can be seen that when either the number of sensors or the total network power is increased, the performance of the BLUE estimator is floored: i.e.  $\lim_{n \rightarrow \infty} MSE \approx \frac{\sigma_v^2}{P_T}$  and  $\lim_{P_T \rightarrow \infty} MSE \approx \frac{\sigma_v^2}{n}$ . The first of these limits is illustrated in Fig. 3 for a constant total network power, as parameterized by the local observation SNR  $\gamma_0$ . It is seen from Figs. 3 and 4 that when local SNR is high the system shows better performance which intuitively makes sense. From Fig. 4, it can be seen that in the region of low local SNR, the performance of the system can be improved by increasing the number of nodes. But in high local SNR region, increasing the number of nodes may not affect the final performance much since ultimately the performance is limited by the channel quality between nodes and the fusion center.

Allocating equal power for all nodes may not result in the best performance since all nodes may not have the same quality observations or communication channels. This is particularly true when one considers channel fading. Let us consider the power allocation among nodes such that the network consumes the minimum possible energy to achieve a desired performance. The optimization problem can be formulated as

$$\min_{g_k \geq 0, k=1, \dots, n} \sum_{k=1}^n g_k^2 \text{ such that } MSE \leq \epsilon_1,$$

where  $\epsilon_1$  is the required MSE threshold at the fusion center. If we assume that the local observations are independent, the optimization problem can be rewritten as



$$\min_{g_k \geq 0, k=1, \dots, n} \sum_{k=1}^n g_k^2 \text{ such that } \epsilon'_1 - \sum_{k=1}^n \frac{h_k^2 g_k^2}{\sigma_v^2 h_k^2 g_k^2 + \sigma_w^2} \leq 0, \tag{17}$$

where we have defined  $\epsilon'_1 = \frac{1}{\epsilon_1}$ . The optimal power allocation strategy is stated in the following lemma assuming that the channel state information (CSI) is available at the distributed nodes.

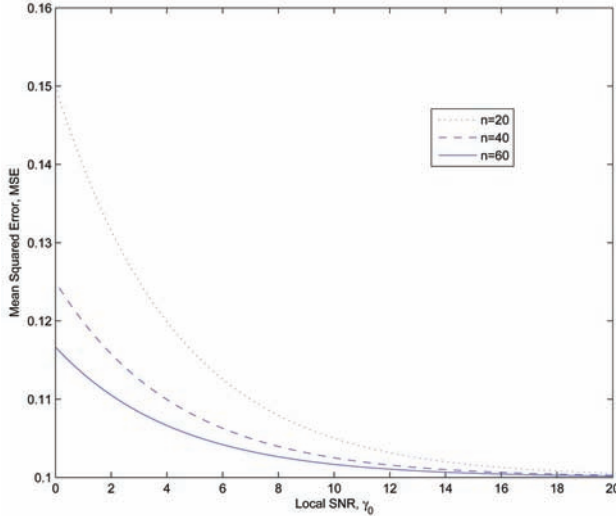


Fig. 4. Performance of mean squared error vs. local SNR,  $\gamma_0$ ; The total network power is constant.

Lemma 5: [34] When local observations are i.i.d., the optimal power allocation solution to (17) is given by

$$g_k^{*2} = \begin{cases} \frac{\sigma_w^2}{h_k^2 \sigma_v^2} \left[ \frac{h_k \sum_{j=1}^{K_1} \frac{1}{h_j}}{(K_1 - \epsilon'_1 \sigma_v^2)} - 1 \right] & ; \text{if } k < K_1 \text{ and } n > \epsilon'_1 \sigma_v^2 \\ 0 & ; \text{if } k > K_1 \text{ and } n > \epsilon'_1 \sigma_v^2 \\ \text{infeasible} & ; \text{if } n < \epsilon'_1 \sigma_v^2 \end{cases} \tag{18}$$

where assuming, without loss of generality,  $h_1 \geq h_2 \geq \dots \geq h_n$ ,  $K_1$  is found such that  $s_1(K_1) < 1$  and  $s_1(K_1 + 1) \geq 1$  for  $1 \leq K_1 \leq n$  where  $s_1(k) = \frac{(k - \epsilon'_1 \sigma_v^2)}{h_k \sum_{j=1}^k \frac{1}{h_j}}$  for  $1 \leq k \leq n$ .

Lemma 5 says that the optimal power at each node depends on its observation quality, channel quality and the required MSE threshold at the fusion center. Note that letting  $\sqrt{\lambda_0} = \sigma_w \frac{\sum_{k=1}^{K_1} \frac{1}{h_k}}{K_1 - \epsilon'_1 \sigma_v^2}$ , for  $s_1(k) - 1 < 0$  and  $n > \epsilon'_1 \sigma_v^2$ , the optimal  $g_k^{*2}$  can be written as  $g_k^{*2} = \frac{\sigma_w^2}{h_k^2 \sigma_v^2} \left( \frac{h_k \sqrt{\lambda_0}}{\sigma_w} - 1 \right)$ . Hence, when CSI is available at distributed nodes, each node can determine its power using  $\sqrt{\lambda_0}$  as a side information that is provided by the fusion center.

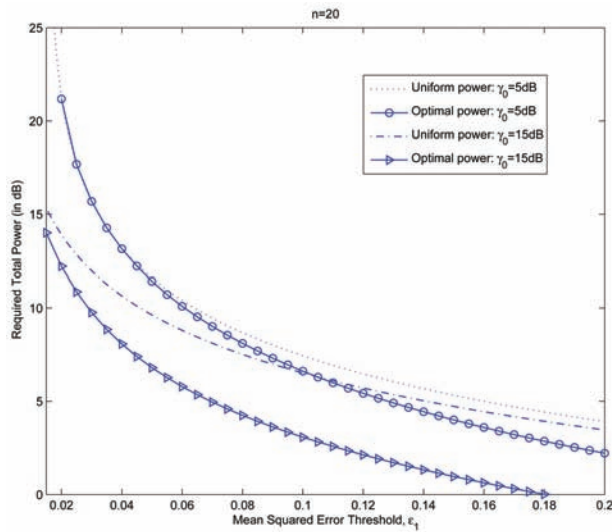


Fig. 5. Optimal power allocation scheme vs. uniform power allocation scheme: The required optimal power to achieve a given MSE of  $\epsilon_1$  as given in (18) is shown in the figure parameterized by local SNR  $\gamma_0$  for  $n=20$  is shown. The comparison of the required uniform power to achieve the same MSE threshold is illustrated.

Figure 5 shows the performance of the optimal power allocation scheme (18) in achieving a desired MSE performance at the fusion center. Figure 5 assumes that fading coefficients are drawn from a Rayleigh distribution with unity mean. It is seen that allocating power optimally as in (18) gives a significant power saving over the uniform power allocation only when either the local observation SNR is high or when the required MSE at the fusion center is not significantly low. This is not surprising since if local observation SNR is high node estimators are good enough on their own and thus perhaps collecting the local estimators from only those nodes with very good fading coefficients can save total power while also meeting the MSE requirement at the fusion center. Moreover, when the MSE required at the fusion center is not very demanding, we may meet it by only collecting local estimators of few nodes (and turning others off), so that the optimal power allocation, may lead to better power savings over the uniform power allocation scheme.

2) *Quantized local processing*: Recall that with the quantization scheme presented in Section 2-C2, an upper bound for the MSE at the fusion center is given by (9) when the communication between the sensors and the fusion center is noiseless. When discrete messages  $m_k$ 's are transmitted over noisy communication channels, however, bit errors may occur in a resource constrained network with a finite power.

Let us assume that the discrete messages are transmitted over a noisy channel where bit errors occur due to imperfect communication. Let  $\hat{m}_k$  and  $p_b^k$  be the decoded message at the fusion center corresponding to the transmitted message  $m_k$  from the  $k$ -th node and the associate bit error probability, respectively. To compute the resulting MSE of the estimator  $\hat{\theta}$  at the fusion center based on the decoded messages  $\{\hat{m}_1, \dots, \hat{m}_n\}$ , the bit errors caused by the channel should be taken into account. For the quantization scheme presented in Section

2-C2, a complete analysis of the resulting MSE at the fusion center with noisy channels is given in [9]. According to [9], for i.i.d. local observations an upper bound for the MSE, when the messages are transmitted over a memoryless binary symmetric channel is given by the following lemma:

*Lemma 6:* [9] *If the bit error rates from node  $k$  is  $p_b^k$ , then the MSE achieved by the fusion center based on the decoded messages  $\{\hat{m}_1, \dots, \hat{m}_n\}$  is upper bounded by*

$$MSE \leq (1 + p_0)^2 \left( \sum_{k=1}^n \frac{1}{\sigma_v^2 + \delta_k^2} \right)^{-1}, \quad (19)$$

where  $\delta_k^2 = \frac{W^2}{(2^{L_k} - 1)^2}$  and  $p_0 > 0$  satisfies the following condition:

$$p_0 \geq \frac{8W}{\sigma_v} \sqrt{\frac{np_b^k}{3}} \text{ for all } k = 1, \dots, n.$$

By comparing (19) with (10) it is observed that the achievable MSE with imperfect communication deviates by that with noiseless communication by a constant factor.

Let the communication channel between node  $k$  and the fusion center undergo path loss attenuation  $a_k$  proportional to  $d_k^\alpha$  where  $d_k$  is the transmission distance from node  $k$  to fusion center and  $\alpha$  is the path loss attenuation index. Assuming that node  $k$  sends  $L_k$  bits using quadrature amplitude modulation (QAM) with constellation size  $2^{L_k}$ , at a bit error probability of  $p_b^k$  the transmission power spent by node  $k$  is  $P_k = B_s E_k$ , where  $B_s$  is the transmission symbol rate and  $E_k$  is the transmission energy per symbol, given by,

$$E_k = \varsigma_k a_k \left( \ln \frac{2}{p_b^k} \right) (2^{L_k} - 1), \quad (20)$$

with  $\varsigma_k = 2N_f N_0 G_d$  where  $N_f$  is the receiver noise figure,  $N_0$  is the single sided thermal noise spectral density and  $G_d$  is a system constant [9].

It can be easily seen from (20) that  $L_k = \log \left( 1 + \frac{P_k}{B_s \varsigma_k a_k \left( \ln \frac{2}{p_b^k} \right)} \right)$ . Thus, to determine the optimal number of bits  $L_k$  to be allocated to node  $k$  in order to meet a desired MSE performance at the fusion center while minimizing the total network power, [9] solves the following optimization problem (assuming  $\varsigma_k$ ,  $B_s$  and  $p_b^k$  are the same for all nodes):

$$\min \|\mathbf{P}\|_2 \text{ such that } MSE \leq \epsilon_2, \quad (21)$$

where  $\|\mathbf{P}\|_2 = \sqrt{\sum_{k=1}^n P_k}$  is the  $L^2$ -norm of the power vector  $\mathbf{P} = [P_1, \dots, P_n]^T$ ,  $\epsilon_2$  is the desired MSE threshold at the fusion center and the MSE is as given by (19). The optimal number of bits  $L_k^*$  to quantize the observations at node  $k$ , that is given by the solution to (21), are characterized in the following lemma.

*Lemma 7:* [9] *The optimal number of bits used to quantize the observations at the  $k$ -th node found by solving (21) is*

$$L_k^* = \begin{cases} \log \left( 1 + \frac{W}{\sigma_v} \sqrt{\frac{\eta_0}{a_k} - 1} \right) & ; \text{if } k \leq K_2 \text{ and } n > \frac{\sigma_v^2}{\epsilon_2} \\ 0 & ; \text{if } k \geq K_2 + 1 \text{ and } n > \frac{\sigma_v^2}{\epsilon_2} \\ \text{infeasible} & ; \text{if } n < \frac{\sigma_v^2}{\epsilon_2} \end{cases}$$

where  $\eta_0 = \left( \frac{K_2}{\sigma_v^2} - \frac{1}{\epsilon_2} \right)^{-1} \sum_{k=1}^{K_2} \frac{a_k}{\sigma_v^2}$ ,  $\epsilon_2' = \frac{\epsilon_2}{(1+p_0)^2}$  and assuming, without loss of generality,  $a_1 \leq a_2 \leq \dots \leq a_n$ ,  $K_2$  is found such that  $s_2(K_2) < 1$  and  $s_2(K_2 + 1) \geq 1$  for  $1 \leq K_2 \leq n$  where  $s_2(k) = a_k \left( \frac{k}{\sigma_v^2} - \frac{1}{\epsilon_2} \right) \left( \sum_{j=1}^k \frac{a_j}{\sigma_v^2} \right)^{-1}$ . Then the optimal transmission power at the  $k$ -th node is given by

$$P_k^* = \varsigma_k B_s \left( \ln \frac{2}{p_b} \right) \frac{W h_k}{\sigma_v} \sqrt{\left( \frac{\eta_0}{a_k} - 1 \right)^+},$$

where  $(x)^+$  equals to zero if  $x < 0$  and equals to  $x$  otherwise.

Note that again the optimal power at node  $k$  is determined by the observation quality, channel quality and the required MSE threshold as was the case with AF local processing we saw in lemma 5. Figure 6 shows the number of sensors that are active in the network to achieve a desired MSE threshold at the fusion center. In Fig. 6, the network size  $n = 1000$  and  $\alpha = 2$ . The distance from node  $k$  to fusion center,  $d_k$ , is drawn from a uniform distribution on  $[1, 2]$ . It is observed that when the required MSE threshold increases the number of active sensors decreases greatly. That is, the network discards the observations at nodes with poor observation and channel quality. This is similar to what we observed in Fig. 5 earlier.

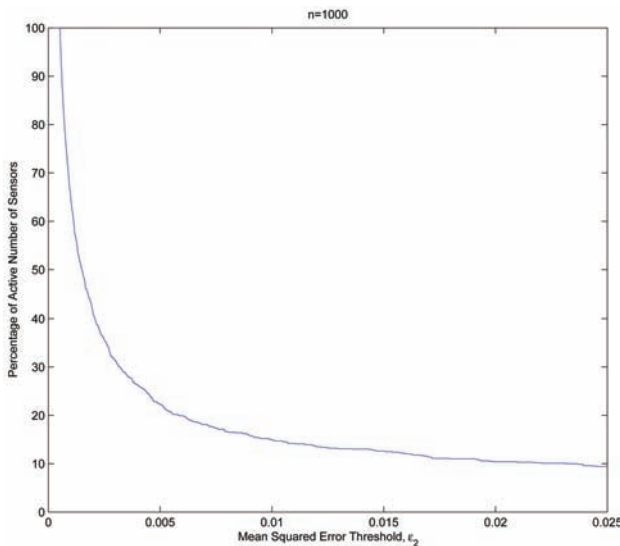


Fig. 6. Number of active sensors in the network according to the optimal power allocation scheme given by lemma 7. The number of total sensors in the network is  $n = 1000$  and  $\alpha = 2$

Figure 7 shows the energy saving due to the optimal power allocation scheme given in lemma 7 compared to the uniform power allocation scheme. Clearly Fig. 7 shows that significant energy savings are possible by optimally selecting number of bits, especially at moderate levels of desired MSE at the fusion center.

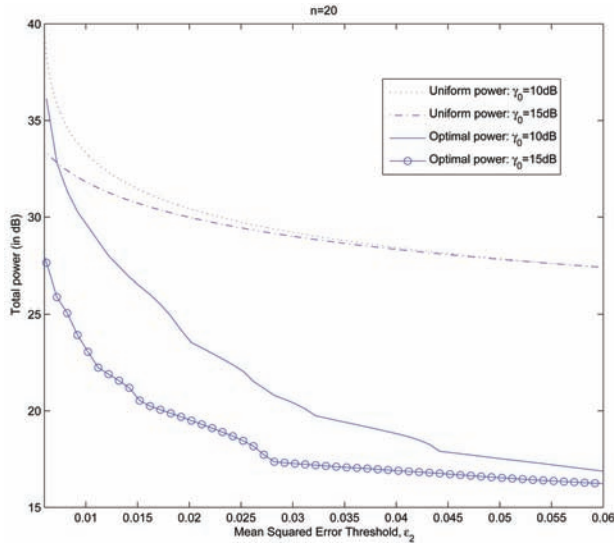


Fig. 7. Performance of optimal power allocation scheme given in lemma 7 vs. uniform power allocation scheme: network size  $n = 20$

### B. Communication over multiple access channels

One of the disadvantages of using orthogonal channels to transmit local decisions is the large bandwidth consumption as the number of distributed nodes  $n$  increases. An alternative is to allow multiple sensor nodes to share a common channel. Such multiple access communication (MAC) in bandwidth constrained wireless sensor networks has been investigated in, among others, [10], [11], [17], [24], [25], [34], [36], [44]. For example, in [24], [25], [44] the authors proposed a type based multiple-access communication in which sensors transmit according to the type of their observation in a shared channel where the *type* is as defined in [45]. An analysis of both orthogonal and MAC channels for distributed detection in a sensor network was presented in [44]. MAC with correlated observations was considered in [34] and [46]. The use of CDMA signaling in distributed detection of deterministic and Gaussian signals under strict power constraints was presented in [10] and [11], respectively. When all sensor nodes communicate with the fusion center coherently, with amplify-and-forward local processing the estimator performance can be improved compared to that of orthogonal communication due to the coherent beam-forming gain [47], [46]. Performance of MAC communication with asynchronous transmissions was discussed in [48].

In the following we consider the form and performance of the final estimator at the fusion center when communications from distributed nodes to the fusion center is over noisy multiple-access channels. Assuming perfect synchronization among sensor transmissions, the received signal at the fusion center over a MAC can be written as

$$r = \sum_{k=1}^n h_k y_k + w,$$

where  $w$  is the receiver noise with zero mean and variance of  $\sigma_w^2$  and  $h_k$  is the channel fading coefficient from node  $k$  to the fusion center, as defined earlier. For the AF local processing, substituting  $y_k = g_k z_k$ , the resulting received signal is given by

$$r = \sum_{k=1}^n h_k g_k z_k + w. \quad (22)$$

Fusion center computes the final estimator based on the received coherent signal  $r$ . The resulting BLUE estimator and its performance is given by the following lemma.

*Lemma 8: [34] The BLUE estimator and the resulting MSE based on the received signal (22) can be shown to be*

$$\hat{\theta}(r) = \frac{r}{\sum_{k=1}^n h_k g_k},$$

and

$$MSE = \frac{\mathbf{e}^T \mathbf{G} \mathbf{H}_c \Sigma_v \mathbf{H}_c \mathbf{G} \mathbf{e} + \sigma_w^2}{(\mathbf{e}^T \mathbf{H}_c \mathbf{G} \mathbf{e})^2}.$$

With i.i.d. local observations the MSE simplifies to

$$MSE = \frac{\sigma_v^2 \sum_{k=1}^n h_k^2 g_k^2 + \sigma_w^2}{(\sum_{k=1}^n h_k g_k)^2}.$$

The MSE performance of the BLUE estimator under both orthogonal and multiple-access channels, with i.i.d. observations, is depicted in Fig. 8 as a function of total network power. Figure 8 assumes equal node powers and unity channel gains. Moreover, MAC communication is assumed to be perfectly synchronized among nodes. As seen from Fig. 8, when total network power is small, the MAC communication leads to a better MSE performance compared to that with orthogonal communication. But as total network power increases both schemes converge to the same performance level. This is because when the network can afford a large transmission power, irrespective of the communication scheme the overall estimator performance is only limited by the local observation quality and the effects of additive/multiplicative channel noise is mitigated by the large gain in the transmission. However, when a practical sensor network is power-constrained the MAC communication may be able to provide a much better performance over that of the orthogonal transmissions when nodes are perfectly synchronized.

Figure 8 assumes equal transmission powers at all nodes. However, when the fusion center needs to achieve only a target estimator quality, say an MSE of  $\epsilon_3$ , one can consider non-uniform power allocations such that,

$$\min_{g_k \geq 0, k=1, \dots, n} \sum_{k=1}^n g_k^2 \text{ such that } MSE \leq \epsilon_3, \quad (23)$$

where MSE is as given in lemma 8. When the observations are i.i.d., a tractable analytical solution for the above optimization problem was given in [34] that is stated in the following lemma.

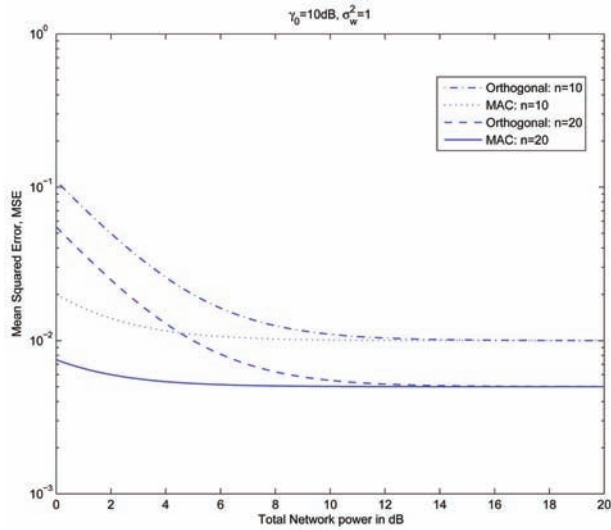


Fig. 8. Mean squared error performance vs. total network power for different network sizes

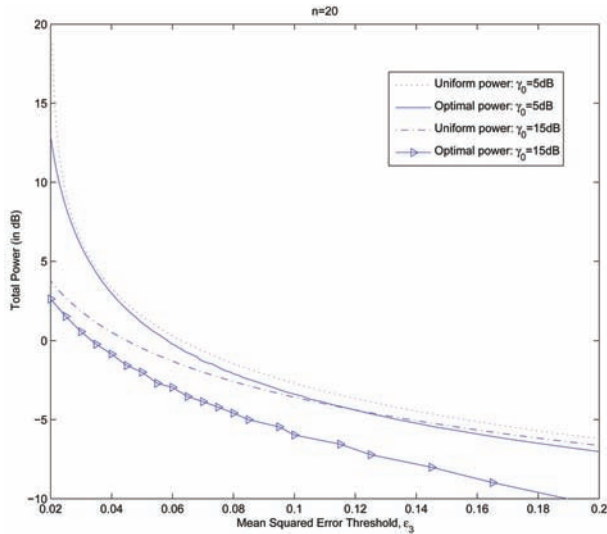


Fig. 9. Performance of optimal power allocation scheme vs. uniform power allocation scheme

Lemma 9: With i.i.d. local observations, the optimal power at node  $k$ ,  $g_k^{*2}$  that solves the optimization problem in (23) with the MSE as given in lemma 8 is  $g_k^{*2} = \frac{\mu^2}{4} \frac{h_k^2}{(1+\eta_0 h_k^2)^2}$  for  $k = 1, 2, \dots, n$ , where  $\eta_0$  and  $\mu$  can be found numerically by solving the equations  $\sum_{k=1}^n \frac{\eta_0 h_k^2}{(1+\eta_0 h_k^2)} = \frac{1}{\epsilon_3}$  and  $\mu = 2 \frac{\sigma_w}{\sigma_v} \left( \frac{1}{\eta_0^2 \epsilon_3} - \sum_{k=1}^n \frac{h_k^4}{(1+\eta_0 h_k^2)^2} \right)^{-\frac{1}{2}}$  where  $\epsilon_3' = \frac{\epsilon_3}{\sigma_v^2}$ .

It is observed that  $\eta_0$  has a feasible solution only when  $n > \frac{\sigma_w^2}{\epsilon_3}$  [34]. The total power spent by the network with the above optimal power allocation scheme is given by  $P_{total} = \sum_{k=1}^n g_k^{*2} = \frac{\sigma_w^2}{\sigma_v^2} \eta_0$ .

Figure 9 shows the performance of the optimal power allocation scheme compared to that of uniform power allocation scheme for a network size of  $n = 20$ . Again, the optimal power scheduling scheme has a significant performance gain over the uniform power allocation scheme especially when local SNR is high or the required MSE threshold at the fusion center is moderate, similar to what was observed in Section 3-A in the case of orthogonal communication.

### C. Effects of synchronization errors on MAC

To achieve coherent gain with MAC transmissions, it is important that the sensor transmissions are synchronized. For this discussion on node synchronization, we assume, i.i.d observations and AF local processing. Analysis would remain essentially the same for other network models as well.

To achieve synchronization in node transmissions, one may assume that there is a master-node (that can be taken as the fusion center itself, for simplicity) that broadcasts the carrier and timing signals to the distributed nodes [47]. Suppose that the  $k$ -th node is located at a distance of  $d_k + \delta_k$  from the fusion center, for  $k = 1, 2, \dots, n$ , where  $d_k$  and  $\delta_k$  are the nominal distance and the sensor placement error of the  $k$ -th node, respectively. The fusion center broadcasts a carrier signal  $\cos(2\pi f_0 t)$  where  $f_0$  is the carrier frequency. The received carrier signal at the  $k$ -th node is a noisy version of  $\cos(2\pi f_0 t + \psi_k + \psi_{ek})$  where  $\psi_k = \frac{2\pi f_0 d_k}{c}$  and  $\psi_{ek} = \frac{2\pi f_0 \delta_k}{c}$ . Each node employs a Phase Locked Loop (PLL) to lock onto the carrier. If each node precompensates for the difference in their nominal distances  $d_k$ , by transmitting its locally processed and modulated observation with a proper delay and phase shift  $\psi_k$ , then the received signal at the fusion center is corrupted only by the phase shift due to the sensor placement error  $\delta_k$ . Considering only the phase shift due to this sensor placement error, the received signal at the fusion center is given by  $r = \sum_{k=1}^n h_k g_k z_k \cos(\psi_{ek}) + w$ , assuming AF local processing at sensor nodes. In the following lemma we assume that the placement error  $\delta_k$  is Gaussian with zero mean and variance  $\sigma_\delta^2$ .

*Lemma 10:* [34] Assuming that  $\sigma_\delta^2 \ll \iota_0$  where  $\iota_0 = \frac{c}{f_0}$ , so that phase error  $\psi_{ek} \sim \mathcal{N}(0, \sigma_\psi^2)$  where  $\sigma_\psi^2$  is small, the BLUE estimator at the fusion center when local observations are i.i.d. is

$\hat{\theta}(r) = r \left( e^{-\frac{\sigma_\psi^2}{2}} \sum_{k=1}^n h_k g_k \right)^{-1}$ . The resulting MSE with the phase synchronization errors is

$$MSE = \frac{\sigma_v^2 \sum_{k=1}^n h_k^2 g_k^2 + e^{\sigma_\psi^2} \sigma_w^2}{\left( \sum_{k=1}^n h_k g_k \right)^2}. \quad (24)$$

Figure 10 shows the MSE performance (24) of a sensor system in the presence of phase synchronization errors. It can be seen that the performance is robust against synchronization errors as long as the variance of the phase error  $\sigma_\psi^2$  is sufficiently small.



#### 4. Sequential communication

In Section 3 it was assumed that the final decision on the PoI is made at a central fusion center, and all nodes were to send their locally processed data to this fusion center. In a distributed network, however, it might be desirable in some applications that the final decision be made fully distributively without depending on a central node. To achieve this goal, nodes may communicate with each other to reach a final decision albeit at the cost of inter-sensor communications. One such architecture of distributed estimation is to communicate with nodes sequentially until the desired performance is reached.

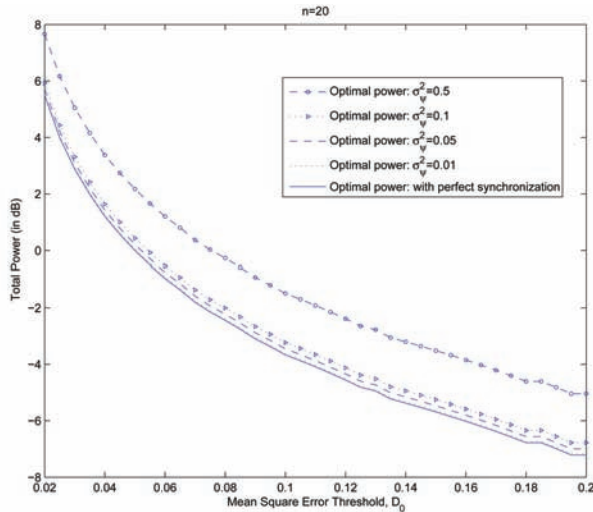


Fig. 10. Performance of optimal power allocation scheme with synchronization errors:  $n = 20$ ,  $\eta_0 = 10dB$

The basic problem of sequential detection for statistical hypotheses was first formulated by Wald in [49] who derived the sequential probability ratio test (SPRT). Analysis of SPRT and its comparison with fixed sample-size test for centralized detection of a constant signal was given in [50]. The decentralized version of the binary sequential detection problem was addressed by [41] and a more general formulation of the distributed sequential detection problem was presented in [40]. Decentralized sequential detection problem with multiple hypotheses was considered in [51] and [52].

Distributed estimation with sequential communication in wireless sensor networks has been addressed by several authors in recent years. For example, in [37]–[39], [53] information driven approaches for distributed sequential estimation of a source location have been investigated. In this architecture only one node communicates with another at a given time and the final decision can be made at any node once a required performance level is reached. However, it is worth mentioning that most of these are concerned with random parameters. To be consistent with our analysis in section 3, we, on the other hand, will consider non-random parameter estimation with sequential estimation, as considered, for example, in [42]. Thus, in the following, we consider the distributed sequential estimation of a non-random parameter where each node makes a local decision based on its own observation and the decision from the previous node.

A. Distributed sequential estimation

Let us assume the same linear observation model (2) as in sections 2 and 3:

$$z_k = \theta + v_k, \text{ for } k = 1, \dots, n. \tag{25}$$

Let  $\hat{\theta}_{k-1}$  be the local estimator at the  $(k - 1)$ -th node. The  $(k - 1)$ -th node sends its estimator to the  $k$ -th node over a noisy channel. Then the effective observation vector at node  $k$  is [42]

$$\mathbf{z}_k = \begin{bmatrix} z_k \\ y_k \end{bmatrix} = \begin{bmatrix} \theta + v_k \\ \hat{\theta}_{k-1} + w_k \end{bmatrix} \text{ for } k = 2, \dots, n, \tag{26}$$

with  $\mathbf{z}_1 = z_1$  where  $w_k$  is the channel noise at inter-sensor communication link from node  $(k - 1)$  to node  $k$  and, as before,  $v_k$  is the observation noise at the  $k$ -th node. Both  $\mathbf{v}$  and  $\mathbf{w}$  are assumed to be zero mean with covariance matrices  $\Sigma_v$  and  $\Sigma_w$ , respectively. Throughout this section we assume that the channel noise  $\{w_k\}_{k=1}^n$  is independent with the covariance matrix  $\Sigma_w = \text{diag}(\sigma_{w_1}^2, \dots, \sigma_{w_n}^2)$ . Moreover the observation noise  $v_k$  and the channel noise  $w_k$  are assumed to be independent of each other. Given the effective observation vector  $\mathbf{z}_k$  the node  $k$  computes the BLUE estimator of parameter  $\theta$ . In the following we consider the cases of independent and correlated observations, separately.

1) *Independent observations:* When local observations are independent,  $\Sigma_v = \text{diag}(\sigma_1^2, \dots, \sigma_n^2)$ . The following lemma from [42] summarizes the BLUE estimator and its performance at the  $k$ -th node for independent observations.

Lemma 11: [42] *When observation noise is independent the BLUE estimator at the  $k$ -th node is given by*

$$\hat{\theta}_k(\mathbf{z}_k) = \frac{P_{k-1} + \sigma_{w_k}^2}{G(k)} z_k + \frac{\sigma_k^2}{G(k)} y_k, \text{ for } k = 2, \dots, n, \tag{27}$$

where  $G(k) = P_{k-1} + \sigma_{w_k}^2 + \sigma_k^2$  and  $P_k$  is the MSE at  $k$ -th node which can be shown as

$$P_k = P_{k-1} - \frac{(P_{k-1} + \sigma_{w_k}^2)P_{k-1} - \sigma_{w_k}^2 \sigma_k^2}{G(k)}, \text{ for } k = 2, \dots, n. \tag{28}$$

For  $k=1$ , we have

$$\hat{\theta}_1(\mathbf{z}_1) = z_1, \tag{29}$$

with corresponding MSE of

$$P_1 = \sigma_1^2. \tag{30}$$

It is seen from lemma 11 that the BLUE estimator at node  $k$  can be determined by observation vector at node  $k$  along with the variance of the estimator  $P_{k-1}$  at the previous node. It is also interesting to see that as the inter-sensor communication link noise vanishes,

$$\lim_{\sigma_{w_k}^2 \rightarrow 0} P_k \approx \frac{P_{k-1}}{1 + \frac{P_{k-1}}{\sigma_k^2}}, \text{ for } k = 2, \dots, n. \tag{31}$$

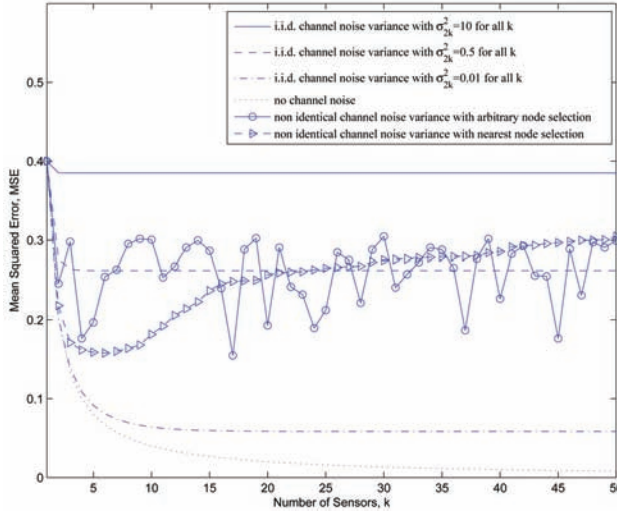


Fig. 11. Mean squared error at the  $k$ -th node: The observations are assumed to be i.i.d. with  $\sigma_k^2 = \sigma_0^2 = 0.4$  for all  $k$ . Three instances of channel noise variance  $\sigma_{w_k}^2$  are considered where case 1:  $\sigma_{w_k}^2$  is i.i.d. with values of 0.01, 0.5 and 10; case 2:  $\sigma_{w_k}^2$ 's are drawn from a uniform distribution in  $[0, 1]$ ; case 3: no channel noise, i.e.  $\sigma_{w_k}^2 = 0$  for all  $k$

In other words, when the inter-sensor communication links are very good we have  $P_k < P_{k-1}$  for  $k = 2, \dots, n$  irrespective of how the next node is selected. That implies that, sending the  $(k-1)$ -th node's decision to the  $k$ -th node always improves the MSE performance. On the other hand, if the quality of inter-sensor communication links is poor, we get

$$\lim_{\sigma_{w_k}^2 \rightarrow \infty} P_k \approx \sigma_k^2, \text{ for } k = 2, \dots, n. \quad (32)$$

(32) implies that when the quality of inter-sensor communication links is poor, the MSE at the  $k$ -th node is not affected by the  $(k-1)$ -th node's decision. Therefore, in that case sequential communication will not improve the MSE performance. In fact, we can see that the above sequential estimation process gives improved performance only when the following condition is satisfied for the observation quality:

*Lemma 12:* [42] *If the current node's MSE is  $P_k$ , then  $P_k \leq P_{k-1}$  if and only if the observation quality at the  $k$ -th node satisfies the following condition:*

$$\sigma_k^2 \leq \left(1 + \frac{P_{k-1}}{\sigma_{w_k}^2}\right) P_{k-1}. \quad (33)$$

For i.i.d. observations and channel noise (such that  $\Sigma_{\mathbf{v}} = \sigma_0^2 \mathbf{I}$  and  $\Sigma_{\mathbf{w}} = \sigma_w^2 \mathbf{I}$ ) it can be shown that  $P_k \leq P_{k-1}$  for all  $k \geq 1$  [42]. In this case, the MSE at the  $k$ -th node also simplifies to

$$P_k = \frac{\sigma_0^2(P_{k-1} + \sigma_w^2)}{P_{k-1} + \sigma_w^2 + \sigma_0^2}. \quad (34)$$

As  $k \rightarrow \infty$ , the asymptotic variance converges to  $P_k = P_{k-1} = P_\infty$ . Then from (34), it can be shown that [42]

$$P_\infty = \frac{\sigma_w^2}{2} \left( \sqrt{1 + \frac{4\sigma_0^2}{\sigma_w^2}} - 1 \right), \tag{35}$$

and we have  $P_\infty \leq \sigma_0^2$ . It is also of interest to see that

$$\text{if } \sigma_w^2 \gg \sigma_0^2 : \quad P_\infty \approx \sigma_0^2 \tag{36}$$

$$\text{if } \sigma_w^2 \ll \sigma_0^2 : \quad P_\infty \approx \sigma_w \sigma_0 \left( 1 - \frac{\sigma_w}{2\sigma_0} \right). \tag{37}$$

Figure 11 shows the performance of the distributed sequential estimator for i.i.d. observation noise (i.e.  $\sigma_k^2 = \sigma_0^2$  for  $k = 1, \dots, n$ ).

In Fig. 11 we have let  $\sigma_0^2 = 0.4$ . If the channel noise variance is also i.i.d. so that  $\sigma_{w_k}^2 = \sigma_w^2$  [42] has shown that  $P_k \leq P_{k-1}$  for all  $k \geq 1$  which is illustrated in Fig. 11 as well. However, it is seen that in this case, when the channel noise variance  $\sigma_w^2$  increases the MSE is limited by the observation noise variance  $\sigma_0^2$  as predicted by (32). When channel noise variances are non-identical, Fig. 11 depicts the MSE performance at node  $k$  with two different schemes for the next node selection. In one scheme, next node is selected randomly and in the other scheme the next node is chosen to be the node with the minimum distance to the current node (note that here we are assuming that  $\sigma_{w_k}^2$ 's are in ascending order with  $k$ ). As can be seen from Fig. 11, with random node selection, whenever the condition (33) in lemma 12 is satisfied, we have  $P_k < P_{k-1}$ . On the other hand, when the next node is selected to be the one at the minimum distance, it is seen from Fig. 11 that after a certain number of nodes, the MSE starts to monotonically increase. Therefore, with this scheme it is important to identify the node at which the MSE is minimum, and terminate the sequential updating process at the particular node.

Fig. 11 only shows the MSE performance when the observations are i.i.d.. However, if the observations are not identical, it might be of interest to perform an information driven distributed sequential estimation process, in which the sequence of nodes are selected to capture the highest information gain as well as with the lowest communication cost.

2) *Correlated observations*: When the observations are correlated, the covariance matrix of the effective received signal  $\mathbf{z}_k$  (26) at  $k$ -th node can be written as,

$$\Sigma_{\mathbf{z}_k} = \begin{bmatrix} \sigma_k^2 & r_{k,k-1} \\ r_{k,k-1} & \sigma_{w_k}^2 \end{bmatrix}, \text{ for } m = 1, \dots, k, \tag{38}$$

where  $r_{k,k-m} \triangleq \mathbb{E}\{v_k \tilde{\theta}_{k-m}\}$  with  $\tilde{\theta}_k = \hat{\theta}(\mathbf{z}_k) - \theta$ . With the covariance matrix in (38), the BLUE estimator and MSE are summarized in the following lemma.

*Lemma 13*: [42] *When local observations are correlated with the covariance matrix in (38), the BLUE estimator at  $k$ -th node is given as*

$$\hat{\theta}_k(\mathbf{z}_k) = \frac{P_{k-1} + \sigma_{w_k}^2 - r_{k,k-1}}{G'(k)} z_k + \frac{\sigma_k^2 - r_{k,k-1}}{G'(k)} w_k, \tag{39}$$

where  $G'(k) = P_{k-1} + \sigma_k^2 + \sigma_{w_k}^2 - 2r_{k,k-1}$  and the corresponding MSE  $P_k$  at  $k$ -th node is

$$P_k = \frac{\sigma_k^2(P_{k-1} + \sigma_{w_k}^2) - r_{k,k-1}^2}{G'(k)}, \quad (40)$$

with  $P_1 = \sigma_1^2$  and  $r_{k,k-1}$  can be computed at node  $k$  using following recursion formula:

$$r_{k,k-1} = \frac{\sigma_{k-1}^2 - r_{k-1,k-2}}{G'(k-1)} r_{k,k-2} + \frac{P_{k-2} + \sigma_{w_{k-1}}^2 - r_{k-1,k-2}}{G'(k-1)} \mathbb{E}\{v_k v_{k-1}\}. \quad (41)$$

In general, computing  $r_{k,k-1}$  from (41) needs a recursion that spans over all  $r_{j,j-m}$  for  $m = 1, \dots, j$  for each  $j = k, k-1, \dots, 1$ . In the special case where  $v_k$  is wide sense stationary with identical variance  $\sigma_0^2$  such that  $\mathbb{E}\{v_k v_{k-m}\} = \rho^m \sigma_0^2$ , [42] has shown that the recursion in (41) can be computed based at node  $k$  only on that is computed at node  $k-1$ . Then the distributed sequential algorithm can be summarized as [42],

1) Initialization

$$\begin{aligned} \hat{\theta}_1(\mathbf{z}_1) &= z_1 \\ P_1 &= \sigma_0^2 \text{ and } r_{2,1} = \rho \sigma_0^2 \end{aligned}$$

2) For  $k = 2, 3, \dots$

$$\begin{aligned} G'(k) &= P_{k-1} + \sigma_0^2 + \sigma_{w_k}^2 - 2r_{k,k-1} \\ \hat{\theta}_k(\mathbf{z}_k) &= \frac{P_{k-1} + \sigma_{w_k}^2 - r_{k,k-1}}{G'(k)} z_k + \frac{\sigma_0^2 - r_{k,k-1}}{G'(k)} w_k \\ F(k) &= \frac{\sigma_{w_k}^2 + P_{k-1} - \sigma_0^2}{G'(k)} \\ P_k &= \frac{\sigma_0^2(P_{k-1} + \sigma_{w_k}^2) - r_{k,k-1}^2}{G'(k)} \\ r_{k,k-1} &= \frac{\rho}{2} [(1 + F(k))\sigma_0^2 + (1 - F(k))r_{k,k-1}] \end{aligned}$$

The above algorithm implies that the sequential estimation process for correlated observations at node  $k$  can be performed with information received from node  $k-1$ , ( $\hat{\theta}_{k-1}(\mathbf{z}_{k-1})$ ,  $P_{k-1}$  and  $r_{k,k-1}$ ) and its own information.

Figure 12 shows the MSE performance at node  $k$  with the number of nodes assuming i.i.d. channel noise with  $\sigma_w^2 = 1$ . It is seen that, when the observation noise is highly correlated, the sequential estimation process does not give significant performance which intuitively makes sense.

## 5. Distributed collaborative data fusion with consensus

The specific topology of the network becomes an issue if all the participant nodes must collaborate to improve on their individual estimates, without a central fusion center that collects and processes all measurements as assumed in Section III. Still it is possible to

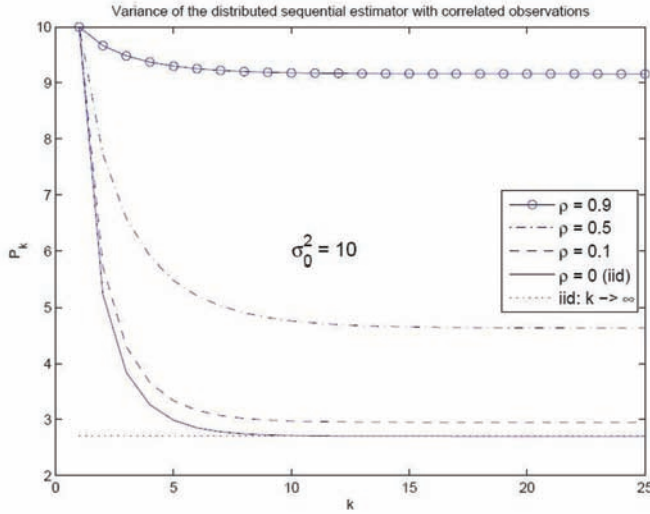


Fig. 12. MSE performance at  $k$ -th node with correlated observations

obtain distributively a function of the observation vector such as the BLUE estimator, as we will detail later. In this section we assume that information is exchanged only locally with neighbors, and can reach distant nodes through an iterative process.

For a proper characterization of the collaboration in such a network, it is necessary to make use of analytical tools which describe the topology of a sensor net. Thus we model the sensor network as a graph  $G = (V, E)$ , with nodes (sensors)  $v_k \in V$  and edges  $e_{kj} \in E$  if there is a path from node  $v_k$  to node  $v_j$ . Note that a path exists in the network if transmissions from node  $v_k$  reaches node  $v_j$ . The elements of the adjacency matrix  $\mathcal{A}$  are defined as  $[\mathcal{A}]_{kj} = 1$  if  $e_{jk} \in E$ , otherwise  $[\mathcal{A}]_{kj} = 0$ . If there is a sequence of edges to go from any node  $k$  to any other node  $j$  the graph is said to be connected. The degree matrix  $\mathbf{D}$  of graph  $G$  is a diagonal matrix such that  $[\mathbf{D}]_{kk}$  is equal to the number of connections entering node  $k$ . With that, the Laplacian matrix  $\mathbf{L}$  is defined as  $\mathbf{L} = \mathbf{D} - \mathcal{A}$ . Specifically, the elements  $[\mathbf{L}]_{kj}$  of the Laplacian matrix  $\mathbf{L}$  are defined as

$$[\mathbf{L}]_{kj} = \begin{cases} -1, & \text{if } e_{jk} \in E \\ [\mathbf{D}]_{kk}, & \text{if } k = j. \end{cases} \tag{42}$$

It turns out that the eigenvalues  $\{\lambda_k\}_{k=1}^n$  of  $\mathbf{L}$  contain important information about the topology of the graph  $G$ . In fact, if they are ordered as  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ , we always have that  $\lambda_1 = 0$  and  $\lambda_2 > 0$  for a connected graph (that is, a graph in which there exists a path connecting any two nodes). This second eigenvalue  $\lambda_2$  is known as the algebraic connectivity of the graph, and its value plays a major role in the speed at which information can be diffused through the network [55]. The corresponding eigenvectors are denoted by  $\mathbf{u}_k$ , with  $\mathbf{u}_1 = \mathbf{e}$  where, as before,  $\mathbf{e}$  is the  $n$ -vector of all ones. As an illustration, the four-node network in Fig. 13 has the following associated Laplacian matrix:

$$\mathbf{L} = \begin{pmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & -1 & 2 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}. \quad (43)$$

The connection between nodes  $k$  and  $j$  is symmetric if whenever node  $k$  sends data to node  $j$  it can also get data from node  $j$ . If all the connections were symmetric, the corresponding Laplacian matrix of the graph in Fig. 13 would look like as follows:

$$\mathbf{L} = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix}. \quad (44)$$

Note that in this case the Laplacian matrix is also symmetric. In the first case the eigenvalues of  $\mathbf{L}$  are given by  $\{0, 1.5 \pm 0.866j, 2\}$ , whereas in the latter case, they are  $\{0, 2, 4, 4\}$ . Clearly the network is more strongly connected in the second case, and this fact is reflected in the magnitude of the second eigenvalue or, in other words, in the algebraic connectivity.

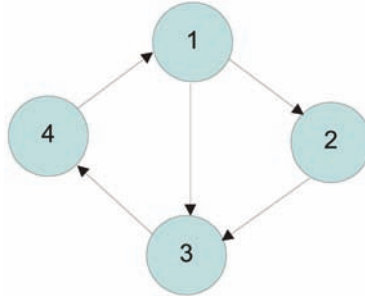


Fig. 13. Four node graph

#### A. Distributed estimation of static parameters

We consider again the same observation model (2) at distributed nodes. However, now all sensors attempt to share information with everyone else in order to update each of their local estimators. Moreover, all sensors wish to agree on a common estimate for  $\theta$ , in what is known as distributed consensus estimation. This cannot be achieved in one shot, however, due to the constraints imposed by the network topology, since sensors can only access information from other nodes corresponding to the non-zero entries in  $\mathbf{L}$  after one exchange, in  $\mathbf{L}^2$  after two exchanges, and so on. Nodes will have to iteratively keep updating and exchanging their local estimators to reach a consensus. Let  $\mathbf{z}(i)$  denote the information vector of all nodes after the  $i$ -th information exchange with  $\mathbf{z}(0) = \mathbf{z}$  where  $\mathbf{z}$  is as given in (2). If, for simplicity, we first consider ideal noiseless links, the following lemma shows how to lead all sensors to the same estimate.

Lemma 14: [55] *The recursion*

$$\mathbf{z}(i+1) = (\mathbf{I} - \gamma\mathbf{L})\mathbf{z}(i), \quad \text{for } i = 0, 1, 2, \dots \quad (45)$$

converges to the vector  $(\mathbf{u}^T \mathbf{z})\mathbf{e}$ , where  $\mathbf{u} = [u_1, u_2, \dots, u_n]^T$  denotes the left eigenvector of  $\mathbf{L}$  associated with the eigenvalue 0 normalized as  $\sum_{k=1}^n u_k = 1$ , provided that  $0 < \gamma < 1/\max_k [\mathbf{D}]_{kk}$ .

According to the recursion (45), at each iteration, the current estimates are exchanged with nodes defined by the active links of the Laplacian matrix  $\mathbf{L}$ , and updated after weighting the neighbors contributions by  $\gamma$ . Thus, an agreement on the value  $\mathbf{u}^T \mathbf{z}$  can be reached after convergence<sup>1</sup>. The convergence speed is a function of both  $\gamma$  and the algebraic connectivity of the graph. Hence sensor local estimates will converge faster in more densely connected networks.

In many situations, however, it might be of interest to ensure that the network converges to an arbitrary final consensus value of  $\mathbf{c}^T \mathbf{z}$ . We can indeed modify (45) to achieve this, so that the final asymptotic consensus estimator is given by  $(\mathbf{c}^T \mathbf{z})\mathbf{e}$ .

Lemma 15: *The recursion*

$$\mathbf{z}(i+1) = (\mathbf{I} - \gamma\mathbf{C}^{-1}\mathbf{U}\mathbf{L})\mathbf{z}(i), \quad i = 0, 1, 2, \dots \quad (46)$$

converges to the vector  $(\mathbf{c}^T \mathbf{z})\mathbf{e}$ , where  $\mathbf{C}$  and  $\mathbf{U}$  are diagonal matrices such that  $\mathbf{C} = \text{diag}(\mathbf{c})$  and  $\mathbf{U} = \text{diag}(\mathbf{u})$ .

Note that the local exchange of values (peer-to-peer) serves to improve the individual estimates at nodes even not directly connected with each other, as the information is diffused through the network at each iteration.

#### B. Robust consensus schemes

Clearly, the assumption of ideal noiseless links is not valid in practice: exchanges contain additive noise, and it turns out that the recursion (45) does not converge to a consensus [56], unless some provision is taken<sup>2</sup>. One possible remedy is to use a decreasing sequence of positive steps  $\gamma(i)$  as suggested in [57]:

$$\mathbf{z}(i+1) = (\mathbf{I} - \gamma(i)\mathbf{C}^{-1}\mathbf{U}\mathbf{L})\mathbf{z}(i) + \text{Diag}\{(\mathbf{I} - \gamma(i)\mathbf{C}^{-1}\mathbf{U}\mathbf{L})\mathbf{W}(i)\}, \quad i = 0, 1, 2, \dots \quad (47)$$

where  $\text{Diag}(A)$  denotes the vector formed by diagonal elements of  $A$  and  $\text{Diag}\{(\mathbf{I} - \gamma(i)\mathbf{C}^{-1}\mathbf{U}\mathbf{L})\mathbf{W}(i)\}$  in (47) accounts for the channel noise leaking into the sensors. Note that the  $n \times n$  matrix  $\mathbf{W}(i)$  contains the links noise values  $w_{kj}(i)$ , as illustrated in Fig. 14 (if two different nodes are not connected, the corresponding value in  $\mathbf{W}(i)$  is irrelevant).

Lemma 16: [57] Asymptotic consensus is achieved in (47) as long as the positive sequence  $\{\gamma(i)\}$  is such that  $\gamma(i) < 1/\max_k [D]_{kk}$  for all  $i$  and satisfies the following conditions:

$$\sum_{i=0}^{\infty} \gamma(i) = \infty \quad \text{and} \quad \sum_{i=0}^{\infty} \gamma^2(i) < \infty. \quad (48)$$

<sup>1</sup> If the network is symmetric, that is,  $\mathbf{L} = \mathbf{L}^t$ , then  $\mathbf{u} = \mathbf{e}/n$ , and the consensus value is the average of the initial observations.

<sup>2</sup> If the noise is correlated with the exchanged values, as it is the case for quantization noise, the strategies to follow can be different from those presented here, which are more general.



In the noiseless case the asymptotic consensus value is that corresponding to the constant step of previous section, which turns out to be independent of the convergence speed. This is no longer the case in the presence of noise. In fact, the stepsize sequence  $\{\gamma(i)\}$  should be designed to speed up convergence while minimizing the impact of channel noise, which will be proportional to the sequence-energy  $\sum_{i=0}^{\infty} \gamma^2(i)$  [58]. For instance, at each step each sensor could compute the BLUE estimator by combining what it receives from its neighbors with its own value, taking into account the exchange noise statistics in the analysis. Invariably, as the exchanges progress the estimates at different nodes become correlated even for initially independent observations. Hence the nodes should be able to estimate the degree of correlation among them for proper combining. On the other hand, it is also possible to use filtering to fight noise (see e.g. [59]) although this causes strict consensus not be achievable.

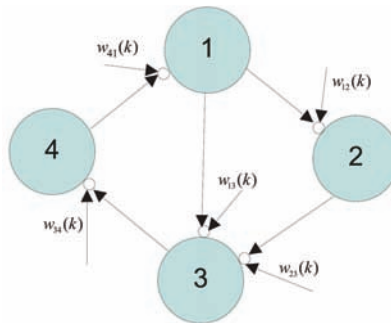


Fig. 14. Four node graph with noisy links.

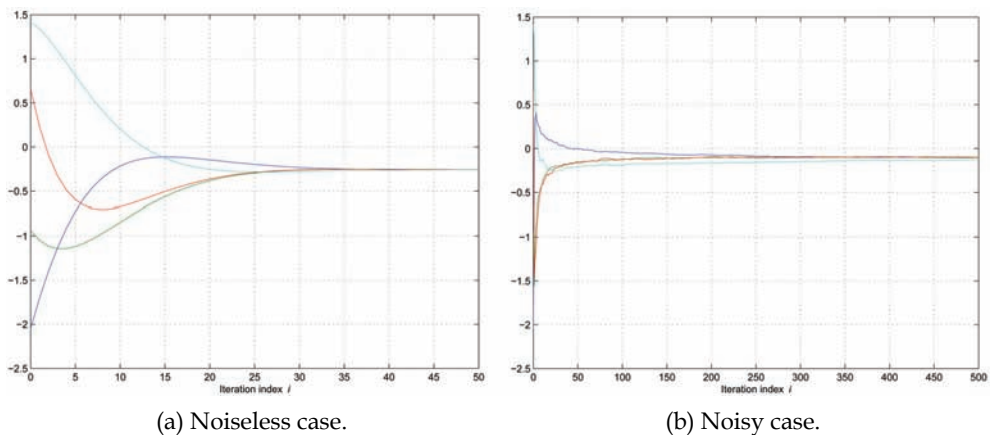


Fig. 15. Convergence curves towards consensus: (a) noiseless links,  $\gamma = 0.1$ , (b) noisy links,  $\gamma(i) = 0.5/i$ ,  $i = 1, 2, \dots$ ,  $\sigma_w^2 = 0.1$ .

Figure 15 shows the evolution of the values in the four node network shown in Figs. 13 and 14, respectively. Clearly, consensus is achieved even in the noisy case, although the asymptotic estimate differs from the ideal case due to the noise. Observe the different horizontal axis scaling used in the two figures, due to the slower convergence in the noisy case.

C. Cooperative tracking: distributed Kalman filters

Dynamic parameter estimation, including target tracking, using wireless sensor networks has been considered by many authors in the literature. Common methods used in dynamic parameter estimation (or target tracking) include Kalman filtering, Bayesian estimation methods, particle filtering and their variants. Kalman filtering for distributed parameter estimation or tracking has been studied by many authors, [60]-[65]. When the state-space model is nonlinear or non-Gaussian, authors in [66]-[70] have proposed the use of particle filters for the dynamic parameter estimation. In this section we discuss how cooperative tracking can be achieved when inter-node communication is possible. In this case, we assume that each node observes an underlying state which evolves, for instance, according to a classical Gauss-Markov model:

$$x(t + 1) = F(t)x(t) + u(t). \tag{49}$$

We denote the observation at node  $k$  at time instant  $t$  by  $z_k(t)$ , where

$$z_k(t) = H_k(t)x(t) + v_k(t), k = 1, 2, \dots, n \text{ and } t = 1, 2, \dots \tag{50}$$

If a centralized observer collects observations from all nodes, it could run a Kalman filter, which is the optimum MSE estimator provided that the noise processes  $u(t)$  and  $v_k(t)$ ,  $k = 1, \dots, n$ , are jointly Gaussian (otherwise, optimality is still true only among linear estimators). The process noise  $u(t)$  and observation noise  $\mathbf{v}(t)$  are assumed to be white, independent and have covariances  $\Sigma_u(t)$  and  $\Sigma_v(t)$  respectively, with

$$\Sigma_v(t) \doteq \begin{pmatrix} \Sigma_{v(1)}(t) & 0 & \dots & 0 \\ 0 & \Sigma_{v(2)}(t) & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \Sigma_{v(n)}(t) \end{pmatrix}. \tag{51}$$

However, if a central fusion center is not available, nodes can benefit from cooperation with neighbors. The degree of improvement as well as the performance degradation with respect to that of a central fusion center will depend on how close to an agreement the network can go before the next state update in (49). This statement can be made more clear after the formulation of the *information form* centralized Kalman filter for the recursive update of the linear estimator  $\hat{x}(t)$  of  $x(t)$  based on the observations up to time instant  $t$  [71]:

$$\begin{aligned} P^{-1}(t)\hat{x}(t) &= F^{-1}(t)(1 - K(t))P^{-1}(t - 1)\hat{x}(t - 1) \\ &+ \mathbf{H}^T(t)\Sigma_v^{-1}(t)\mathbf{z}(t) \end{aligned} \tag{52}$$

with

$$\mathbf{z}(t) \doteq \begin{bmatrix} z_1(t) & z_2(t) & \dots & z_n(t) \end{bmatrix}^T, \tag{53}$$

$$\mathbf{H}(t) \doteq \begin{bmatrix} H_1(t) & H_2(t) & \dots & H_n(t) \end{bmatrix}^T, \tag{54}$$

and where  $P(t) = \mathbb{E}\{(x(t) - \hat{x}(t))^2\}$  is computed through the following updates:

$$R(t) = \Sigma_u^{-1}(t) + F^{-1}(t)P^{-1}(t-1)F^{-1}(t), \quad (55)$$

$$K(t) = F^{-1}(t)P^{-1}(t-1)F^{-1}(t)R^{-1}(t), \quad (56)$$

$$P^{-1}(t) = F^{-1}(t)P^{-1}(t-1)F^{-1}(t) + \mathbf{H}^T(t)\Sigma_v^{-1}(t)\mathbf{H}(t) - K(t)R(t)K(t), \quad (57)$$

for an initial uncertainty  $P(-1)$ . As observed in [61] the above equations show that each sensor would be able to emulate a central fusion center should it be able to compute the following quantities:

$$\mathbf{H}^T(t)\Sigma_v^{-1}(t)\mathbf{z}(t) = \sum_{k=1}^n H_k(t)z_k(t)/\Sigma_{v(k)}(t) \quad (58)$$

$$\mathbf{H}^T(t)\Sigma_v^{-1}(t)\mathbf{H}(t) = \sum_{k=1}^n H_k^2(t)/\Sigma_{v(k)}(t) \quad (59)$$

As stated above, any linear combination of the initial sensor values can be asymptotically achieved in a distributed form via collaborative exchanges of local information, although in a time-varying setting the number of available exchanges will determine the level of alignment of the quantities (58) and (59). As expected, if the number of exchanges without channel noise goes to infinity, the performance of the network approaches that of a centralized observer [72]. Clearly, this requires the exchange of the observations as well as associated signal-to-noise ratios.

As an illustration, let us consider in detail the setting with one hop exchanges in which each sensor has access to the (noisy) observations from its closest neighbors exclusively. From (58) and (59) we see that nodes must exchange  $H_k(t)z_k(t)/\Sigma_{v(k)}(t)$  and  $H_k^2(t)/\Sigma_{v(k)}(t)$ , so the combined observations can be written as

$$\bar{\mathbf{z}}(t) = \bar{\mathbf{H}}(t)x(t) + \bar{\mathbf{v}}(t) = \mathbf{A} \begin{pmatrix} H_1(t)z_1(t)/\Sigma_{v(1)}(t) \\ \vdots \\ H_n(t)z_n(t)/\Sigma_{v(n)}(t) \end{pmatrix} + \text{Diag}\{\mathbf{A}\mathbf{W}(t)\} \quad (60)$$

where we have defined the extended observation vector as

$$\bar{\mathbf{z}}(t) \doteq \begin{bmatrix} \bar{z}_1(t) & \bar{z}_2(t) & \cdots & \bar{z}_n(t) \end{bmatrix}^T. \quad (61)$$

If the weighting matrix is chosen, for simplicity, as  $\mathbf{A} = \mathbf{I}\gamma\mathbf{L}$ , although more sophisticated weighting coefficients can also be used, then the  $k$ -th element of the  $n$ -vector  $\bar{\mathbf{H}}(t)$  and the covariance matrix of the vector noise  $\bar{\mathbf{v}}(t)$  in (60) are given in the following lemma.

Lemma 17: The state-space model at the  $k$ -th node when merging observations as in (60) is given by

$$x(t + 1) = F(t)x(t) + u(t) \tag{62}$$

$$\bar{z}_k(t) = \bar{H}_k(t)x(t) + \bar{v}_k(t), \tag{63}$$

for  $k = 1, \dots, n$ , with

$$\bar{H}_k(t) = \frac{H_k^2(t)}{\Sigma_{\bar{v}(k)}(t)} - \gamma \sum_{j=1}^n \frac{H_j^2(t)}{\Sigma_{\bar{v}(j)}(t)} [\mathbf{A}]_{kj}. \tag{64}$$

In addition, if the noise power  $\sigma_w^2$  is the same for all communication links, then

$$\Sigma_{\bar{v}}(t) = \mathbf{A} \begin{pmatrix} H_1^2(t)/\Sigma_{v(1)}(t) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & H_n^2(t)/\Sigma_{v(n)}(t) \end{pmatrix} \mathbf{A}^T + \gamma^2 \sigma_w^2 \mathbf{D}.$$

It is important to note that the covariance matrix of the vector noise  $\bar{\mathbf{v}}(t)$  containing the  $n$  components  $\bar{v}_k(t)$  in (17) is now a function of the original observation noise statistics, the weighting matrix used, and the communication noise. Each node can iterate the corresponding Kalman filter provided that the communication of  $H_k^2(t)/\Sigma_{v(k)}(t)$  is noiseless (Note that, in practice, these values may change slowly making robust communication easier). As a consequence, after each measurement the corresponding values are exchanged with neighbors to assist the Kalman updating at each node. The weight  $\gamma$  should be chosen as a function of the channel noise, with a low value for highly unreliable channels [73].

It is also possible to exchange state estimates instead of the described observations merging, or combine both if enough bandwidth is available. In any case, the number of exchanges will determine the performance of the distributed Kalman filter.

## 6. Chapter summary

In this chapter, we discussed the fusion performance of estimation of a scalar parameter, be it static or dynamic by a distributed sensor network in the presence communication noise. The distributed nodes in the network make observations on a PoI and make local decisions based on the observations. Then by communicating the local summaries over wireless channels (to a central node or to their local neighbors), a final estimator is obtained.

In section 2, the multi-sensor data fusion problem was formulated and two types of local processing schemes were discussed. Then these locally processed data was transmitted over wireless channels to a central node or shared among each other. The communication noise was allowed to be multiplicative (due to path loss attenuation and the multi-path fading) or additive (due to receiver noise). In section 3, the data fusion was analyzed when nodes communicate with a central node that forms the final estimator.

When there is no central fusion center available in the network, nodes may communicate with each other to improve their local estimators by combining them with those of their neighbors. Sections 4 and 5 addressed this type of sensor nets in two different contexts.

Section 4 discussed the distributed sequential estimation in which nodes communicate sequentially to update their estimator and the final decision can be made at any distributed node. In Section 5, the distributed estimation with consensus was considered where all nodes try to reach a final decision that agrees with each other via inter-node communication among neighbors.

## 7. References

- [1] R. S. Smith and F. Y. Hadaegh, "Distributed estimation, communication and control for deep space formations," *IET control Theory and Applications*, vol. 1, no. 2, pp. 445 – 451, March 2007.
- [2] A. Speranzon, C. Fischione, and K. H. Johansson, "A distributed estimation algorithm for tracking over wireless sensor networks," in *IEEE Int. Conf. on Communications (ICC)*, June 2007, pp. 3088 – 3093.
- [3] R. Nowak, U. Mitra, and R. Willett, "Estimating inhomogeneous fields using wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 999–1006, Aug. 2004.
- [4] H. Wang, J. Elson, L. Girod, D. Estrin, and K. Yao, "Target classification and localization in habitat monitoring," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Hong kong, April 2003.
- [5] E. Biagioni and K. Bridges, "The application of remote sensor technology to assist the recovery of rare and endangered species," *Int. Journ. of High Performance Computing Applications*, vol. 16, no. 3, pp. 112–121, Fall 2002.
- [6] A. Krasnopeev, J.-J. Xiao, and Z.-Q. Luo, "Minimum energy decentralized estimation in a wireless sensor network with correlated sensor noises," *EURASIP Journal on Wireless Commun. and Networking*, vol. 2005, no. 4, pp. 473–482, 2005.
- [7] J. F. Chamberland and V. V. Veeravalli, "Asymptotic results for decentralized detection in power constrained wireless sensor networks," *IEEE J. Select. Areas Commun.*, vol. 22, no. 6, pp. 1007–1015, Aug. 2004.
- [8] S. Appadwedula, V. V. Veeravalli, and D. L. Jones, "Energy efficient detection in sensor networks," *IEEE J. Select. Areas Commun.*, vol. 23, no. 4, pp. 693–702, Apr 2005.
- [9] J.-J. Xiao, S. Cui, Z.-Q. Luo, and A. J. Goldsmith, "Joint estimation in sensor networks under energy constraints," in *Proc. IEEE first conf. Sensor and Ad Hoc commun. and Networks*, Santa Clara, CA, Oct. 2004.
- [10] S. K. Jayaweera, "Large system decentralized detection performance under communication constraints," *IEEE Commun. Lett.*, vol. 9, pp. 769–771, Sep. 2005.
- [11] – –, "Bayesian fusion performance and system optimization in distributed stochastic Gaussian signal detection under communication constraints," *IEEE Trans. Signal Processing.*, vol. 55, no. 4, pp. 1238–1250, April. 2007.
- [12] J. F. Chamberland and V. V. Veeravalli, "The impact of fading on decentralized detection in power constrained wireless sensor networks," in *Proc. Acoust., Speech, Signal Processing 2004. (ICASSP '04)*, vol. 3, Montreal, Canada, May. 2004, pp. 837–840.
- [13] X. Zhang, H. V. Poor, and M. Chiang, "Optimal power allocation for distributed detection in wireless sensor networks," *IEEE Trans. Signal Processing.*, vol. 56, no. 9, pp. 4124–4140, Sept. 2008.

- [14] I. Bahceci and A. K. Khandani, "Energy-efficient estimation of correlated data in wireless sensor networks," in *40th Annual Conf. on Information Sciences and Systems (CISS)*, Princeton, NJ, March 2006, pp. 973–978.
- [15] A. Speranzon, C. Fischione, and K. H. Johansson, "Distributed and collaborative estimation over wireless sensor networks," in *Proc. 45th IEEE Int. Conf. on Decision and Control (CDC)*, San Diego, CA, Dec. 2006, pp. 1025–1030.
- [16] T. Wimalajeewa and S. K. Jayaweera, "Optimal power scheduling for correlated data fusion in wireless sensor networks via constrained PSO," *IEEE Trans. on Wireless Commun.*, vol. 7, no. 9, pp. 3608–3618, Sept. 2008.
- [17] S. K. Jayaweera and K. Altarazi, "Large system decision fusion performance in inhomogeneous sensor networks," in *IEEE Int. Conf. on Advanced Video and Signal based Surveillance (AVSS)*, Sydney, Australia, Nov. 2006.
- [18] S. Marano, V. Matta, and P. Willett, "Distributed estimation in large wireless sensor networks via a locally optimum approach," *IEEE Trans. on Signal Processing*, vol. 56, no. 2, pp. 748–756, Feb. 2008.
- [19] I. D. Schizas, G. B. Giannakis, and Z.-Q. Luo, "Distributed estimation using reduced-dimensionality sensor observations," *IEEE Trans. on Signal Processing*, vol. 55, no. 55, pp. 4284 – 4299, Aug. 2007.
- [20] Q. Tian and E. J. Coyle, "Optimal distributed estimation in clustered sensor networks," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, Toulouse, France, May 2006.
- [21] V. Megalooikonomou and Y. Yesha, "Quantizer design for distributed estimation with communication constraints and unknown observation statistics," *IEEE Trans. Communications*, vol. 48, no. 2, pp. 181–184, Feb. 2000.
- [22] K. Zhang and X. R. Li, "Optimal sensor data quantization for best linear unbiased estimation fusion," in *Proc. 43rd IEEE Int. Conf. on Decision and Control (CDC)*, Atlantis, Paradise Island, Bahamas, Dec. 2004, pp. 2656–2661.
- [23] W. Li and H. Dai, "Distributed detection in wireless sensor networks using a multiple access channel," *IEEE Trans. Signal Processing.*, vol. 55, no. 3, pp. 822–833, March 2007.
- [24] G. Mergen and L. Tong, "Type based estimation over multiaccess channels," *IEEE Trans. Signal Processing.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.
- [25] G. Mergen, V. Naware, and L. Tong, "Asymptotic detection performance of type-based multiple access over multiaccess fading channels," *IEEE Trans. Signal Processing.*, vol. 55, no. 3, pp. 1081–1092, March 2007.
- [26] S. Marano, V. Matta, L. Tong, and P. Willett, "A likelihood-based multiple access for estimation in sensor networks," *IEEE Trans. on Signal Processing*, vol. 55, no. 11, pp. 5155–5166, Nov. 2007.
- [27] T. C. Aysal and K. E. Barner, "Blind decentralized estimation for bandwidth constrained wireless sensor networks," *IEEE Trans. on Wireless Commun.*, vol. 7, no. 5, pp. 1466–1471, May 2008.
- [28] C. Thejaswi and R. K. Patro, "Distributed detection in bandwidth constrained wireless sensor networks," in *IEEE Radio and Wireless Symposium (RWS)*, San Diego, CA, Jan. 2006, pp. 95–98.

- [29] Z.-Q. Luo and J.-J. Xiao, "Universal decentralized estimation in a bandwidth constrained sensor network," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 2210–2219, Jun 2005.
- [30] Z.-Q. Luo, "An isotropic universal decentralized estimation scheme for a bandwidth constrained ad hoc sensor network," *IEEE Journal on Selected areas in Commun.*, vol. 23, no. 4, pp. 735–744, April 2005.
- [31] A. Ribeiro and G. B. Giannakis, "Bandwidth-constrained distributed estimation for wireless sensor networks- part II: Unknown probability density function," *IEEE Trans. on Signal Processing*, vol. 54, no. 7, pp. 2784–2796, July 2006.
- [32] — —, "Bandwidth-constrained distributed estimation for wireless sensor networks-part I: Gaussian case," *IEEE Trans. on Signal Processing*, vol. 54, no. 3, pp. 1131–1143, Mar. 2006.
- [33] S. Cui, J.-J. Xiao, A. J. Goldsmith, Z.-Q. Luo, and H. V. Poor, "Estimation diversity and energy efficiency in distributed sensing," *IEEE Trans. on Signal Processing*, vol. 55, no. 9, pp. 4683 – 4695, Sept. 2007.
- [34] T. Wimalajeewa and S. K. Jayaweera, "Power efficient distributed estimation in a bandlimited wireless sensor network," in *41st Asilomar Conference on Signals, Systems and Computers (ACSSC)*, Pacific Grove, CA, Nov. 2007, pp. 2156–2160.
- [35] P. Ishwar, R. Puri, K. Ramchandran, and S. S. Pradhan, "On rate-constrained distributed estimation in unreliable sensor networks," *IEEE Jour. on Selected Areas of Communications*, vol. 23, no. 4, pp. 765 – 775, April 2005.
- [36] T. M. Duman and M. Salehi, "Optimal quantization for distributed estimation via a multipleaccess channel," in *IEEE Int. Symp. on Inform. Theory (ISIT)*, Whistler, Canada, Sept. 1995.
- [37] T. Zhao and A. Nehorai, "Distributed sequential Bayesian estimation of a diffusive source in wireless sensor networks," *IEEE Trans. Signal Processing.*, vol. 55, no. 4, pp. 1511–1524, April 2007.
- [38] M. Chu, H. Haussecker, and F. Zhao, "Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks," *Int. J. High-Perform. Comput. Applicat.*, vol. 16, no. 3, pp. 90–110, 2002.
- [39] T. Zhao and A. Nehorai, "Information-driven distributed maximum likelihood estimation based on Gauss-Newton method in wireless sensor networks," *IEEE Trans. on Signal Processing*, vol. 55, no. 9, pp. 4669–4682, Sept. 2007.
- [40] H. R. Hashemi and I. B. Rhodes, "Decentralized sequential detection," *IEEE Trans. Inform. Theory.*, vol. 35, no. 3, pp. 509–520, May 1989.
- [41] D. Teneketzis and Y. C. Ho, "The decentralized Wald problem," *Inform. Computation*, vol. 73, no. 1, pp. 23–44, 1987.
- [42] S. K. Jayaweera and C. Mosquera, "Distributed sequential estimation with noisy, correlated observations," *IEEE Signal Processing Letters*, July 2008 accepted.
- [43] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. New Jersey, USA: Prentice Hall PTR, 1998.
- [44] K. Liu and A. Sayeed, "Optimal distributed detection strategies for wireless sensor networks," in *42nd annual Allerton conf. commun., control and computing*, Monticello, IL, Oct. 2004.
- [45] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley and Sons, Inc., NY, 2006.

- [46] K. Altarazi, S. K. Jayaweera, and V. Aravinthan, "Performance of decentralized detection in a resource-constrained sensor network," in *39th Annual Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, Nov. 2005.
- [47] G. Barriac, R. Mudumbai, and U. Madhow, "Distributed beamforming for information transfer in sensor networks," in *Third Int. Symp. on Information Processing in Sensor Networks (IPSN 2004)*, vol. 26-27, Berkeley, CA, April 2004, pp. 81-88.
- [48] Y. Lin, B. Chen, and L. Tong, "Distributed detection over multiple access channels," in *IEEE International Conference on, Acoustics, Speech and Signal Processing, (ICASSP)*, Honolulu, Hawaii, USA, April 2007, pp. 541-544.
- [49] A. Wald, "Sequential tests of statistical hypotheses," *The Annals of Mathematical Statistics*, vol. 9, no. 2, pp. 117-186, June 1945.
- [50] H. V. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1994.
- [51] V. Dragalin, A. G. Tartakovsky, and V. V. Veeravalli., "Multihypothesis sequential probability ratio tests, part I: Asymptotic optimality," *IEEE Trans. on Info. Theory*, vol. 45, no. 7, pp. 2448-2462, Nov. 1999.
- [52] — —, "Multihypothesis sequential probability ratio tests, part II: Accurate asymptotic expansions for the expected sample size," *IEEE Trans. on Info. Theory*, vol. 46, no. 4, pp. 1366-1383, July 2000.
- [53] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 61-72, March 2002.
- [54] A. H. Sayed and C. G. Lopes, "Distributed recursive least-squares strategy over adaptive networks," in *Proc. 40th Asilomar Conf. on Signals, Systems and Computers (ACSSC)*, Pacific Grove, CA, 2006, pp. 233-237.
- [55] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215-233, Jan. 2007.
- [56] L. Xiao, S. Boyd, and S.-J. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, vol. 67, no. 1, pp. 33-46, Jan 2007.
- [57] Y. Hatano and M. Mesbahi, "Agreement over random networks," *IEEE Trans. on Automatic Control*, vol. 50, no. 11, pp. 1867-1872, Nov. 2005.
- [58] C. Mosquera, R. López-Valcarce, and S. Jayaweera, "Distributed Estimation with Noisy Exchanges," in *9th IEEE Int. Workshop on Signal Processing Advances in Wireless Communications, Recife, Brazil*, July 2008.
- [59] S. Barbarossa and G. Scutari, "Decentralized maximum-likelihood estimation for sensor networks composed of nonlinearly coupled dynamical systems," *IEEE Trans. on Signal Processing*, vol. 55, no. 7-Part 1, pp. 3456 - 3470, July 2007.
- [60] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *46th IEEE Conf. on Decision and Control (CDC)*, New Orleans, LA, USA, Dec. 2007, pp. 5492 - 5498.
- [61] — —, "Distributed Kalman Filter with Embedded Consensus Filters," in *Proc. of 44th IEEE Conf. on Decision and Control (CDC)*, Sevilla, Spain, Dec 2005, pp. 8179-8184.
- [62] D. P. Spanos, R. O. Saber, and R. M. Murray, "Approximate distributed Kalman filtering in sensor networks with quantifiable performance," in *Proc. 4th Int. Symp. Information Processing in Sensor Networks (IPSN)*, Los Angeles, CA, USA, Apr. 2005, pp. 133-139.



- [63] U. A. Khan and J. F. Moura, "Distributed kalman filters in sensor networks: Bipartite fusion graphs," in *14th Workshop on Statistical Signal Processing (SSP)*, Madison, WI, USA, Aug. 2007, pp. 700 – 704.
- [64] C.-J. Ran and Z.-L. Deng, "Two average weighted measurement fusion Kalman filtering algorithms in sensor networks," in *7th World Congress on Intelligent Control and Automation (WCICA)*, Chongqing, China, June 2008, pp. 2387–2391.
- [65] S. Kirti and A. Scaglione, "Scalable distributed Kalman filtering through consensus," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, NV, USA, March-April 2008, pp. 2725–2728.
- [66] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
- [67] M. Coates, "Distributed particle filters for sensor networks," in *3rd Int. Symp. on Information Processing in Sensor Networks (IPSN)*, April 2004, pp. 99 – 107.
- [68] D. Gu, J. Sun, Z. Hu, and H. Li, "Consensus based distributed particle filter in sensor networks," in *Int. Conf. on Information and Automation (ICIA)*, Hunan, China, June 2008, pp. 302–307.
- [69] P. M. Djuric, M. Vemula, and M. F. Bugallo, "Target tracking by particle filtering in binary sensor networks," *IEEE Trans. on Signal Processing*, vol. 56, no. 6, pp. 2229–2238, June 2008.
- [70] A. Ahmad and M. R. Gani, "Distributed estimation for nonlinear stochastic signals over sensor networks," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, NV, USA, March-April 2008, pp. 3425 – 3428.
- [71] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Prentice Hall, 2000.
- [72] I. Schizas, G. B. Giannakis, A. Ribeiro, and S. I. Roumeliotis, "Consensus in Ad Hoc WSNs with Noisy Links - Part II: Distributed Estimation and Smoothing of Random Signals," *IEEE Trans. on Signal Processing*, vol. 56, no. 4, pp. 1650–1666, April 2008.
- [73] C. Mosquera and S. Jayaweera, "Entangled Kalman Filters for Cooperative Estimation," in *5th IEEE Sensor Array and Multichannel Signal Processing Workshop, Darmstadt (Germany)*, 2008.

# Monte Carlo Methods for Node Self-Localization and Nonlinear Target Tracking in Wireless Sensor Networks

Joaquín Míguez, Luis Arnaiz and Antonio Artés-Rodríguez  
*Department of Signal Theory and Communications*  
*Universidad Carlos III de Madrid*  
*Spain*

## 1. Introduction

Most applications of wireless sensor networks (WSN) rely on the accurate localization of the network nodes [Patwari et al., 2005]. In particular, for network-based navigation and tracking applications it is usually assumed that the sensors, and possibly any data fusion centers (DFCs) in charge of processing the data collected by the network, are placed at *a priori* known locations. Alternatively, when the number of nodes is too large, WSNs are usually equipped with beacons that can be used as a reference to locate the remaining nodes [Sun et al., 2005]. In both scenarios, the accuracy of node localization depends on some external system that must provide the position of either the whole set of nodes or, at least, the beacons [Patwari et al., 2005]. Although beacon-free network designs are feasible [Sun et al., 2005, Ihler et al., 2005, Fang et al., 2005, Vemula et al., 2006], they usually involve complicated and energy-consuming local communications among nodes which should, ideally, be very simple.

In this paper, we address the problem of tracking a maneuvering target that moves along a region monitored by a WSN whose nodes, including both the sensors and the DFCs, are located at unknown positions. Therefore, the target trajectory, its velocity and all node locations must be estimated jointly, without assuming the availability of beacons. We advocate an approach that consists of three stages: initialization of the WSN, target and node tracking, and data fusion. At initialization, the network collects a set of data related to the distances among nodes. These data can be obtained in a number of ways, but here we assume that each sensor node is able to detect, with a certain probability of error, other nodes located nearby and transmit this information to the DFCs. These data are then used by the DFCs to acquire initial estimates of the node positions. An effective tool to perform this computation is the accelerated random search (ARS) method of [Appel et al., 2003], possibly complemented with an iterated importances sampling procedure [Cappé et al., 2004] to produce a random population of node positions approximately distributed according to their posterior probability distribution given the available data. This approach is appealing because it couples naturally with the algorithms in the tracking phase.

We propose to carry out target tracking by means of sequential Monte Carlo (SMC) methods, also known as particle filters (PFs) [Doucet et al., 2000, 2001, Crisan & Doucet,

2002, Djurić et al., 2003, Ristić et al., 2004, Bolić et al., 2005], which recursively track the target position and velocity, as well as improve node positioning, with the generation of new data by the WSN. We should remark that the treatment of unknown (random) fixed parameters, such as the sensor positions are in our framework, using PFs is still an open problem. We propose two algorithms that tackle this difficulty. The first one is based on the auxiliary particle filtering (APF) methodology [Pitt & Shephard, 2001, Liu & West, 2001] and the second one follows the density-assisted strategy of [Djurić et al., 2004].

The data fusion stage deals with the combination of the outputs produced by different DFCs in order to produce improved estimates of both the target state and the WSN node locations. Again, we investigate two approaches. Both of them are aimed at the coherent combination of the estimates produced by individual DFCs but differ in the amount of information that they use and the requirements imposed on the WSN communication capabilities. The most complex technique theoretically yields asymptotically optimal Bayesian estimation of the target state and the node locations (hence, optimal fusion) with distributed computations, but at the expense of making *all* data available to *all* DFCs. It is based on the parallelization of PFs as addressed in [Bolić et al., 2005, Míguez, 2007].

The remaining of the paper is organized as follows. After a brief comment on notation, Section 2 is devoted to a formal description of the system model. A general outline of the proposed scheme is given in Section 3. Sections 4, 5 and 6 are devoted to the procedures proposed for initialization, tracking and data fusion, respectively. In Section 7 we show some illustrative simulation results. Finally, the main results are summarized in Section 8.

### 1.1 Notation

Scalar magnitudes are denoted as regular letters, e.g.,  $x, N$ . Vectors and matrices are denoted as lower-case and upper-case bold-face letters, respectively, e.g., vector  $x$  and matrix  $\mathbf{X}$ . We use  $p(\cdot)$  to denote the probability density function (pdf) of a random magnitude. This is an argument-wise notation, i.e.,  $p(x)$  denotes the pdf of  $x$  and  $p(y)$  is the pdf of  $y$ , possibly different. The conditional pdf of  $x$  given the observation of  $y$  is written as  $p(x|y)$ . Sets are denoted using calligraphic letters, e.g.,  $\mathcal{Q}$ . Specific sets built from sequences of elements are denoted by appropriate subscripts, e.g.,  $x_{1:N} = \{x_1, \dots, x_N\}$ .

## 2. System model

We assume that the target moves along a 2-dimensional region  $\mathcal{C} \subseteq \mathbb{C}^2$  (a compact subset of the complex plane) according to the linear model Gustafsson02

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{u}_t, \quad t \in \mathbb{N} \quad (1)$$

where  $\mathbf{x}_t = [r_t, v_t]^\top \in \mathbb{C}^2$  is the target state, which includes its position and its velocity at time  $t$  ( $r_t$  and  $v_t$ , respectively);  $\mathbf{A} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$  is a transition matrix that depends on the observation period,  $T$ , and

$$\mathbf{u}_t = [u_{r,t}, u_{v,t}]^\top \sim CN(\mathbf{u}_t | 0, \mathbf{C}_u) \quad (2)$$

is a complex Gaussian noise term, with zero mean and known covariance matrix

$$\mathbf{C}_u = \sigma_u^2 \begin{bmatrix} \frac{1}{4}T^4 & 0 \\ 0 & T^2 \end{bmatrix}, \tag{3}$$

that accounts for unknown acceleration forces. The initial target state,  $x_0$ , has a known prior probability density function (pdf),  $p(\mathbf{x}_0) = p(r_0)p(v_0)$ , and we assume  $p(v_0) = CN(0, \sigma_{v,0}^2)$ , i.e., the prior pdf of the velocity random process is complex Gaussian with zero mean and variance,  $\sigma_{v,0}^2$ .

The network consists of  $N_s$  sensors and  $N_c$  DFCs. Sensors are located at random unknown positions  $s_{1:N_s} = \{s_1, s_2, \dots, s_{N_s}\}$ ,  $s_i \in \mathbb{C}$ , with independent and identical uniform prior pdf's,  $p(s_i) = U(\mathcal{C})$ ,  $i = 1, \dots, N_s$ , on the 2-dimensional region monitored by the WSN. During the network startup, each sensor detects any other nodes located within a certain range,  $S_u > 0$ . In particular, the  $n$ -th sensor builds up an  $N_s \times 1$  vector of decisions,  $\mathbf{b}_n = [b_{n,1}, \dots, b_{n,N_s}]^T$ , where (deterministically)  $b_{n,n} = 1$  while  $b_{n,k} \in \{1, 0\}$ ,  $n \neq k$ , is a binary random variable with probability mass function (pmf)

$$p(b_{n,k} = 1 | s_{1:N_s}) = p_d(d_{n,k}^s, S_u), \tag{4}$$

where  $d_{n,k}^s = |s_n - s_k|$  is the distance between the  $n$ -th and  $k$ -th sensors and  $p_d(\cdot, \cdot)$  is the function that yields the probability of detection. At time 0, these decisions are broadcast to the DFCs and we collect them altogether in the  $N_s \times N_s$  matrix  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_{N_s}]$  for notational convenience.

The locations of the  $N_c$  DFCs are denoted as  $c_1, \dots, c_{N_c}$ , with  $c_i \in \mathcal{C} \ \forall i$ . By convention, the first DFC is assumed to be located at the origin of the monitored region, i.e.,  $c_1 = \mathbf{0}$ . The positions of the remaining DFCs are assumed random and unknown, with complex Gaussian prior pdf's  $p(c_i) = CN(c_i | \mu_i^c, \sigma_c^2)$ ,  $i = 2, \dots, N_c$ . The physical implication of this model is that DFCs are deployed at locations which are only roughly known. The variance  $\sigma_c^2$  indicates the uncertainty in this prior knowledge.

During the normal operation of the network, the  $n$ -th sensor periodically measures some distance-dependent physical magnitude related to the target. The measurement obtained by the  $n$ -th sensor at discrete-time  $t \geq 1$  is denoted as  $y_{n,t} = f_s(d_{n,t}, \mathcal{E}_{n,t}^y)$ , where  $d_{n,t} = |r_t - s_n|$  is the distance between the target and the sensor,  $\mathcal{E}_{n,t}^y$  is a random perturbation with known pdf and  $f_s(\cdot, \cdot)$  is the measurement function. We assume that not every sensor necessarily transmits its observation,  $y_{n,t}$ , at every time. Indeed, it is often convenient (in order to reduce energy consumption) that only a subset of sensors become active and transmit their measurements. The local decision of a sensor to transmit its data or

not depends on the comparison of the measurement,  $y_{n,t}$ , with some reference value,  $S_y$ . We also introduce a certain probability of transmission failure,  $\beta$ . A failure can be caused, e.g., by a strong interference in the channel that prevents adequate reception of the communication signal at the DFC. Thus, at time  $t$  only an  $N_t \times 1$  vector of observations,  $\mathbf{y}_t = [y_{\kappa(1),t}, \dots, y_{\kappa(N_t),t}]^T$ , where  $0 \leq N_t \leq N_s$  and  $\kappa(i) \in \{1, \dots, N_s\}$ ,  $\forall i$ , is effectively broadcast to the DFCs (note that all DFCs collect the same data from the sensors). We assume that the likelihood  $p(\mathbf{y}_t | r_t, s_{1:N_s}, c_{1:N_c})$  can be evaluated up to a proportionality constant.

Each DFC has the capability to extract some distance-related magnitude from the communication signals transmitted by the sensors. For simplicity, we consider the same type of measurement carried out at the sensors, hence the  $n$ -th DFC also has available, at time  $t \geq 0$ , the  $N_t \times 1$  data vector  $\mathbf{z}_{n,t} = [z_{\kappa(1),n,t}, \dots, z_{\kappa(N_t),n,t}]^T$ , where  $\mathbf{z}_{i,n,t} = f_s(d_{i,n,t}^c, \varepsilon_{i,n,t}^z)$ ,  $d_{i,n,t}^c = |s_i - c_n|$  and  $\varepsilon_{i,n,t}^z$  is a random perturbation with known pdf, so that the likelihood  $p(z_{n,t} | s_{1:N_s}, c_n)$  can be computed. Note that  $\mathbf{z}_{n,0}$  is defined (unlike  $y_0$ ), and has dimension  $N_0 = N_s$ , because during the network startup all sensors broadcast signals to the DFCs.

We assume that the DFCs are equipped with communication devices more sophisticated than those at the sensor nodes and, as a consequence, it is feasible to exchange data among the DFCs. In particular, during network startup one DFC collects a set of  $N_c(N_c - 1)$  observations,  $q_{i,n} = f_s(d_{i,n}^0, \varepsilon_{i,n}^0)$ ,  $i, n \in \{1, \dots, N_c\}$  (but  $i \neq n$ ), where  $d_{i,n}^0 = |c_i - c_n|$  and  $\varepsilon_{i,n}^0$  is a random perturbation with known pdf, so that  $p(q_{i,n} | c_{1:N_c})$  can be evaluated. For conciseness, we define the set  $\mathcal{Q} = \{q_{i,n}\}_{i,n \in \{1, \dots, N_c\}}^{i \neq n}$ . Moreover, during normal operation of the WSN, each DFC may receive sufficient information from the other fusion nodes to build the  $N_t \times N_c$  matrix of observations  $\mathbf{Z}_t = [\mathbf{z}_{1,t}, \dots, \mathbf{z}_{N_c,t}]$ . Essentially, this means that the DFCs may be capable of sharing data.

The goal is to jointly estimate the target states  $\mathbf{x}_{0:t} = \{\mathbf{x}_0, \dots, \mathbf{x}_t\}$ , the sensor locations,  $s_{1:N_s}$  and the unknown DFC positions,  $c_{2:N_c}$ , from the decisions in  $B$ , the data in  $\mathcal{Q}$  and the sequences of observation vectors  $\mathbf{y}_{1:t} = \{\mathbf{y}_1, \dots, \mathbf{y}_t\}$  and  $\mathbf{Z}_{0:t} = \{\mathbf{z}_0, \dots, \mathbf{z}_t\}$ .

### 3. Proposed scheme

The proposed method consists of three stages, that we outline below.

**Stage 1. Initialization:** Using the data generated during the network startup, at this stage we compute maximum *a posteriori* (MAP) estimates of the DFC locations,  $c_{2:N_c}$ , that will be kept fixed during the WSN operation (including the tracking and fusion stages). We also compute marginally MAP estimates of the sensor locations,  $s_{1:N_s}$ . The latter point estimates are employed to initialize an iterated importance sampling procedure that generates a

population of candidate positions at time  $t = 0$ , denoted  $s_{0:1:N_s}^{(i)}$ ,  $i = 1, \dots, M$ , approximately distributed according to the posterior pdf  $p(s_{1:N_s} | \mathcal{Q}, \mathbf{B}, \mathbf{Z}_0, c_{1:N_c})$ . This posterior sample is needed to start the tracking algorithms in stage 2.

**Stage 2. Tracking:** We investigate two techniques. The first one is an APF algorithm for state estimation in dynamic systems with unknown fixed parameters derived according to [Liu & West, 2001]. A special feature of this technique is that it adaptively approximates the high dimensional posterior pdf of the sensor positions using a sequence of kernel mixtures.

The second algorithm relies on the density-assisted (DA) approach of [Djurić et al., 2004] to approximate the posterior density of the sensor positions using a parametric family of pdf's. An advantage of this procedure is that it enables the analytical integration of the target velocity, thus reducing the sampling space dimension [Chen & Liu, 2000].

**Stage 3. Fusion of estimates:** Both the APF and the DA tracking algorithms can either be run independently in separate DFCs (an using only the data available at each DFC) or be designed as a single global algorithm (that needs to use all the data available through all DFCs) implemented in a distributed way. The computational complexity is similar in both cases but the second approach imposes stringent communication requirements on the network.

If the tracking algorithms are run independently, using different observation sets at each DFC, we propose a mechanism for fusing the resulting  $N_c$  estimates that combines them *coherently* into a single point estimate. We must note that estimates produced by different DFCs with different data cannot be simply averaged because the available observations (both  $\mathbf{y}_{1:t}$  and  $\mathbf{Z}_{0:t}$ ) are insensitive to the angles between the target and the sensors or the sensors and the DFCs. Therefore, it is necessary to take one track estimate as a reference (e.g., the one produced by DFC 1) and adjust the others *by means of rotations only* to minimize the mismatch.

If the tracking algorithms use the same set of observations in all DFCs, then it is possible to apply methods for the implementation of a single PF whose computations are distributed (parallelized) among the available DFCs. This is done by applying the techniques in [Bolić et al., 2005] and [Míguez, 2007]. With this approach each DFC transmits its  $\mathbf{z}_{n,t}$ ,  $n \in \{1, \dots, N_c\}$ , to all other DFCs but, in exchange, it theoretically guarantees an asymptotically optimal Bayesian estimation of the target state and sensor positions by a simple linear combination of the estimates produced by the  $N_c$  DFCs.

## 4. Initialization

### 4.1 Point estimation

As a first step, we obtain point estimates of the node locations, including the DFCs,  $c_{2:N_c}$  (note that we assume  $c_1=0$ ), and the sensors,  $s_{1:N_s}$ , given the information available at time  $t=0$ . An accurate estimation of the DFC positions is of utmost importance, since they will be kept fixed in subsequent stages. Therefore, we propose to compute MAP estimates of the locations  $c_{2:N_c}$  by solving the nonlinear optimization problem

$$\hat{c}_{2:N_c} = \arg \max_{c_{2:N_c}} \{p(c_{2:N_c} | \mathcal{Q}, \mathbf{Z}_0)\}$$

$$= \arg \max_{c_{2:N_c}} \left\{ \prod_{i \neq k} p(q_{i,k} | c_i, c_k) \prod_{j=2}^{N_c} CN(c_j | \mu_j^c, \sigma_c^2) \right\}. \quad (5)$$

Similarly, marginal MAP estimates of the sensor locations, conditional on  $c_{2:N_c} = \hat{c}_{2:N_c}$ , is achieved by solving

$$\begin{aligned} \hat{s}_\ell &= \arg \max_{s_\ell} \left\{ p(s_\ell | Z_0, \hat{c}_{2:N_c}) \right\} \\ &= \arg \max_{s_\ell} \left\{ \prod_{n=1}^{N_c} P(z_{\ell,n,0} | s_\ell, \hat{c}_n) \right\}, \end{aligned} \quad (6)$$

for  $\ell = 1, \dots, N_s$  and  $\hat{c}_1 = c_1 = 0$  is *a priori* known. We point out that addressing the estimation of each sensor individually we neglect the information in matrix  $\mathbf{B}$ , but the dimensionality reduction accounts for this loss (searching for a global MAP solution in  $\mathcal{C}^{N_s}$  turns out practically much harder).

Problems (5) and (6) do not have closed form solutions in general. However, they can be numerically solved, with high accuracy, using the accelerated random search (ARS) algorithm [Appel et al., 2003]. ARS is a Monte Carlo technique for global optimization that enjoys a fast convergence rate and a simple implementation. The algorithm is described, for a general setup, in Table 1.

<p><b>Problem:</b> <math>\hat{\alpha} = \arg \max_{\alpha \in \mathcal{A}} g(\alpha)</math> for some function <math>g</math>.</p> <p>Denote: <math>R_{min} &gt; 0</math>, the “minimum radius”; <math>R_{max} &gt; R_{min}</math>, the “maximum radius”; <math>R_{max} \geq R_n \geq R_{min}</math> the radius at the <math>n</math>-th iteration; <math>\nu &gt; 1</math> the “contraction” factor; <math>\alpha_n</math> the solution obtained after the <math>n</math>-th iteration; and</p> <p><math>\mathcal{B}_n = \{\tilde{\alpha} \in \mathcal{A} : \ \tilde{\alpha} - \alpha_n\ _2 &lt; R_n\}</math>,</p> <p>where <math>\ \cdot\ _2</math> indicates the Euclidean norm.</p> <p><b>Algorithm:</b> given <math>R_n</math> and <math>\alpha_n</math>,</p> <ol style="list-style-type: none"> <li>(1) Draw <math>\tilde{\alpha} \sim U(\mathcal{B}_n)</math>.</li> <li>(2) If <math>g(\tilde{\alpha}) &gt; g(\alpha_n)</math> then <math>\alpha_{n+1} = \tilde{\alpha}</math> and <math>R_{n+1} = R_{max}</math>, else <math>\alpha_{n+1} = \alpha_n</math> and <math>R_{n+1} = R_n/\nu</math>.</li> <li>(3) If <math>R_{n+1} &lt; R_{min}</math>, then <math>R_{n+1} = R_{max}</math>.</li> <li>(4) Go back to (1)</li> </ol>
---

Table 1. Iterative ARS algorithm for a maximization problem. Parameter  $\alpha$  is possibly multidimensional (typically,  $\alpha \in \mathcal{C}^n$ ). The algorithm is usually stopped after a given number of iterations without going changing  $\alpha_n$ .

We assume the DFC positions known for all subsequent derivations, i.e., we treat  $\hat{c}_{2:N_c}$  as the true values and skip them from notation for conciseness.

## 4.2 Population Monte Carlo

The PFs applied in the tracking stage need a sample of sensor positions drawn from the posterior pdf at time  $t=0$ ,  $p(s_{1:N_s} | \mathbf{B}, \mathbf{Z}_0)$  in order to start running. To generate this initial population, we propose an iterated importance sampling<sup>1</sup> method called PMC [Cappé et al., 2004]. In the first iteration, particles are drawn from independent complex Gaussian proposals built from the ARS estimates and a fixed variance,  $\sigma_s^2(0)$ , i.e.,

$$s_n^{(i)}(0) \sim CN(s_n | \hat{s}_n, \sigma_s^2(0)), \quad n = 1, \dots, N_s, \quad i = 1, \dots, M, \quad (7)$$

with weights

$$w^{(i)}(0) = \frac{p(\mathbf{Z}_0 | s_{1:N_s}^{(i)}(0))p(\mathbf{B} | s_{1:N_s}^{(i)}(0))}{\prod_{n=1}^{N_s} CN(s_n^{(i)}(0) | \hat{s}_n, \sigma_s^2(0))}. \quad (8)$$

After the  $(k-1)$ -th iteration, the weighted particles are  $\Omega_{k-1}^{pmc} = \{s_{1:N_s}^{(i)}(k-1), w^{(i)}(k-1)\}_{i=1}^M$  and importance sampling for the  $k$ -th iteration is performed as

$$s_n^{(i)}(k) \sim CN(s_n | \bar{s}_n^{(i)}(k-1), \sigma_{s,n}^2(k-1)), \quad (9)$$

where

$$\begin{aligned} \bar{s}_n(k-1) &= \sum_{i=1}^M w^{(i)}(k-1) s_n^{(i)}(k-1), \\ \bar{s}_n^{(i)}(k-1) &= a s_n^{(i)}(k-1) + (1-a) \bar{s}_n(k-1), \quad \text{and} \\ \sigma_{s,n}^2(k-1) &= (1-a^2) \sum_{i=1}^M w^{(i)}(k-1) |s_n^{(i)}(k-1) - \bar{s}_n(k-1)|^2 \end{aligned}$$

for some  $0 < a < 1$ , i.e., we build the  $(k-1)$ -th kernel approximation of  $p(s_{1:N_s} | \mathbf{Z}_0, \mathbf{B})$  with shrinkage [Liu & West, 2001] for variance reduction. The corresponding weights are

$$w^{(i)}(k) = \frac{p(\mathbf{Z}_0 | s_{1:N_s}^{(i)}(k))p(\mathbf{B} | s_{1:N_s}^{(i)}(k))}{\prod_{n=1}^{N_s} CN(s_n^{(i)}(k) | \bar{s}_n^{(i)}(k-1), \sigma_{s,n}^2(k-1))}. \quad (10)$$

If the algorithm is iterated  $N$  times, we obtain a sample of equally-weighted particles  $\{s_{0,1:N_s}^{(i)}\}_{i=1}^M$ , with approximate pdf  $p(s_{1:N_s} | \mathbf{Z}_0, \mathbf{B})$  by resampling from  $\Omega_N^{pmc}$ .

<sup>1</sup>See, e.g., [DeGroot & Schervish, 2002] for a brief review of the importance sampling principle.



## 5. Tracking

In the second stage, the aim is to track the sequence of target states,  $\mathbf{x}_t$ , and improve the estimation of the sensor positions,  $s_{1:N_s}$ , recursively, as new observations  $\mathbf{y}_t$  and  $\mathbf{Z}_t$  are received. We introduce two PFs that attain this goal. We assume that each DFC runs a PF, but the algorithms are derived for the scenario where all data at time  $t$ , including both  $\mathbf{y}_t$  and  $\mathbf{Z}_t = [\mathbf{z}_{1,t}, \dots, \mathbf{z}_{N_c,t}]$ , are available for the algorithm. This means that  $\mathbf{z}_{n,t}$  must be transmitted from DFC  $n$ , where it is collected, to all other DFCs. Note, however, that we can derive the proposed algorithms in the alternative scenario in which only  $\mathbf{z}_{n,t}$  is available at the  $n$ -th DFC by simply substituting  $\mathbf{Z}_{0:t}$  by  $\mathbf{z}_{n,0:t}$  in the proposed procedures. Indeed, we will assess the performance of the PFs in the two scenarios in Section 7.

### 5.1 Auxiliary particle filter

As a first approach, we propose to use an APF algorithm based on [Liu & West, 2001]. The APF is a recursive algorithm that generates a sequence of discrete probability measures, denoted  $\Omega_t = \{(\mathbf{x}_t, s_{1:N_s}, w_t^{(i)})_{i=1}^M$ , that approximate the posterior pdf's of the unknowns, i.e., for  $n \in \{1, \dots, N_c\}$ ,

$$p(\mathbf{x}_t, s_{1:N_s} | \mathbf{y}_{1:t}, \mathbf{Z}_{0:t}, \mathbf{B}) \approx \sum_{i=1}^M \delta_i(\mathbf{x}_t, s_{1:N_s}) w_t^{(i)}, \quad (11)$$

where  $\delta_i(\mathbf{x}_t, s_{1:N_s})$  is a delta measure centered at  $\{\mathbf{x}_t^{(i)}, s_{1:N_s}^{(i)}\}$ . The samples,  $\mathbf{x}_t^{(i)}$  and  $s_{1:N_s}^{(i)}$ , for  $i = 1, \dots, M$ , are called particles and the importance weights  $w_t^{(i)}$  are normalized to yield  $\sum_{i=1}^M w_t^{(i)} = 1$ . When at time  $t$  observations  $\mathbf{y}_t$  and  $\mathbf{Z}_t$  become available,  $\Omega_t$  is recursively computed from  $\Omega_{t-1}$  as described in Table 2.

The proposed APF algorithm is based on the relationship

$$\begin{aligned} p(\mathbf{x}_t, s_{1:N_s} | \mathbf{y}_{1:t}, \mathbf{Z}_{0:t}, \mathbf{B}) &\propto p(\mathbf{y}_t, \mathbf{Z}_t | \mathbf{x}_t, s_{1:N_s}) p(\mathbf{x}_t | s_{1:N_s}, \mathbf{y}_{1:t-1}) \\ &\times p(s_{1:N_s} | \mathbf{y}_{1:t-1}, \mathbf{Z}_{0:t-1}, \mathbf{B}) \end{aligned} \quad (12)$$

and the approximations

$$p_M(\mathbf{x}_t | s_{1:N_s}, \mathbf{y}_{1:t-1}) = \sum_{i=1}^M p(\mathbf{x}_t | \mathbf{x}_{t-1}^{(i)}) \delta_i(s_{1:N_s}) w_{t-1}^{(i)} \quad (13)$$

$$p_M(s_{1:N_s} | \mathbf{y}_{1:t-1}, \mathbf{Z}_{0:t-1}, \mathbf{B}) = \sum_{i=1}^M w_{t-1}^{(i)} K_i(s_{1:N_s}), \quad (14)$$

where  $K_i(\cdot)$  is a symmetric kernel. For the latter, we have chosen

$$K_i(s_{1:N_s}) = CN(s_{1:N_s} | \boldsymbol{\mu}_{t-1}^{(i)}, h^2 \boldsymbol{\Sigma}_{t-1}) \quad (15)$$

where

$$\boldsymbol{\mu}_{t-1}^{(i)} = [s_{t-1,1}^{-(i)}, \dots, s_{t-1,N_s}^{-(i)}]^\top \quad (16)$$

$$\boldsymbol{\Sigma}_{t-1} = \text{diag}\{\sigma_{t-1,1}^2, \dots, \sigma_{t-1,N_s}^2\} \quad (17)$$

and  $h \in (0,1)$  is a bandwidth factor. The kernel modes are calculated as

$$\bar{s}_{t-1,k}^{-(i)} = a s_{t-1,k}^{(i)} + (1-a) \bar{s}_{t-1,k}, \quad (18)$$

for  $a = \sqrt{1-h^2}$  and

$$\bar{s}_{t-1,k} = \sum_{k=1}^M w_{t-1}^{(i)} s_{t-1,k}^{(i)}. \quad (19)$$

<p><math>t = 0</math>: Draw <math>\mathbf{x}_0^{(i)} \sim p(\mathbf{x}_0)</math> and <math>s_{0,1:N_s}^{(i)} \sim p(s_{1:N_s}   \mathbf{B}, \mathbf{Z}_0)</math></p> <p><math>t</math>: Given <math>\Omega_{t-1} = \left\{ (\mathbf{x}_{t-1}, s_{t-1,1:N_s})^{(i)}, w_{t-1}^{(i)} \right\}_{i=1}^M</math>:</p> <ol style="list-style-type: none"> <li>(1) Compute <math>\tilde{\mathbf{x}}_t^{(i)} = \mathbf{A} \mathbf{x}_{t-1}^{(i)}, i = 1, \dots, M</math>.</li> <li>(2) Draw indices <math>\ell^{(i)} \sim q_t(\ell), i = 1, \dots, M</math>, where <math>q_t(\ell) \propto w_{t-1}^{(\ell)} p(\mathbf{y}_t, \mathbf{Z}_t   \tilde{\mathbf{x}}_t^{(\ell)}, s_{t-1,1:N_s}^{(\ell)})</math>.</li> <li>(3) Draw <math>s_{t,1:N_s}^{(i)} \sim q_t(s_{1:N_s}   \ell^{(i)})</math>, where <math>q_t(s_{1:N_s}   \ell^{(i)}) = N(s_{1:N_s}   \boldsymbol{\mu}_{t-1}^{(\ell^{(i)})}, h^2 \boldsymbol{\Sigma}_{t-1})</math>, for <math>i = 1, \dots, M</math>.</li> <li>(4) Draw target states <math>\mathbf{x}_t^{(i)} \sim p(\mathbf{x}_t   \mathbf{x}_{t-1}^{(\ell^{(i)})}), i = 1, \dots, M</math>, and build the trajectory <math>\mathbf{x}_{0:t}^{(i)} = \{\mathbf{x}_{0:t-1}^{(\ell^{(i)})}, \mathbf{x}_t^{(i)}\}</math>.</li> <li>(5) Update the weights, for <math>i = 1, \dots, M</math>, <math>w_t^{(i)} \propto \frac{p(\mathbf{y}_t, \mathbf{Z}_t   \mathbf{x}_t^{(i)}, s_{t,1:N_s}^{(i)})}{p(\mathbf{y}_t, \mathbf{Z}_t   (\tilde{\mathbf{x}}_t, s_{t-1,1:N_s})^{(\ell^{(i)})})}</math>.</li> <li>(6) MMSE estimation: <math>\hat{\mathbf{x}}_t = \sum_{i=1}^M \mathbf{x}_t^{(i)} w_t^{(i)}</math> and <math>\hat{s}_{1:N_s} = \sum_{i=1}^M s_{1:N_s}^{(i)} w_t^{(i)}</math>.</li> </ol>
---

Table 2. Liu and West's APF algorithm for joint estimation of the target trajectory,  $x_{0:t}$ , and the fixed node locations,  $s_{1:N_s}$ , from the observations available at time  $t$ .

The variances, in turn, are found as

$$\sigma_{k,t-1}^2 = \sum_{l=1}^M w_{t-1}^{(l)} \left( s_{t-1,k}^{(l)} - \bar{s}_{t-1,k} \right)^2. \quad (20)$$

This choice of  $\boldsymbol{\mu}_{t-1}^{(i)}$  and  $\boldsymbol{\Sigma}_{t-1}$  ensures that the mean and marginal variance of every fixed parameter given by the kernel approximation (14) is equal to the corresponding mean and marginal variance given by the weights [Liu & West, 2001].

One difficulty with the approximations (13) and (14) is that they involve mixtures of a typically large number ( $M$ ) of pdf's. We avoid this limitation by incorporating a discrete auxiliary random variable  $\ell \in \{1, \dots, M\}$  that indicates the terms in (13) and (14) to be selected. In particular, we define

$$p(x_t, s_{1:N_s}, \ell | \mathbf{y}_{1:t}, \mathbf{Z}_{0:t}, \mathbf{B}) \propto p(y_t, Z_t | x_t, s_{1:N_s}) p(x_t | x_{t-1}^{(\ell)}) w_{t-1}^{(\ell)} K_\ell(s_{1:N_s}). \quad (21)$$

Using (21) we can easily draw particles and compute weights by applying the principle of importance sampling (IS) [DeGroot & Schervish, 2002]. In particular, we define a suitable importance function, or proposal pdf,

$$q_t(x_t, s_{1:N_s}, \ell) = q_t(\ell) q_t(s_{1:N_s} | \ell) p(x_t | x_{t-1}^{(\ell)}) \quad (22)$$

(see Table 2 for the details) that we use for drawing new particles and then update the weights as

$$w_t^{(i)} \propto \frac{p((\mathbf{x}_t, s_{t,1:N_s}, \ell)^{(i)} | \mathbf{y}_{1:t}, \mathbf{Z}_{0:t}, \mathbf{B})}{q_t((\mathbf{x}_t, s_{t,1:N_s}, \ell)^{(i)})}, i = 1, \dots, M. \quad (23)$$

The auxiliary variables are discarded before proceeding to time  $t+1$ .

We finally note that, given  $\Omega_t$ , it is straightforward to produce estimates of the target trajectory and the node locations. In particular, we can approximate the minimum mean square error (MMSE) estimate of  $\mathbf{x}_t$  or  $s_{1:N_s}$  at time  $t$  by simply computing the weighted mean of the particles in  $\Omega_t$ , as shown also in Table 2.

## 5.2 Mixture Kalman filter

For convenience of exposition, let us begin with the case in which the locations of the sensors,  $s_{1:N_s}$ , are known. If we aim at the Bayesian estimation of the sequence of target positions  $r_{0:t}$  conditional on the observations  $\mathbf{y}_{1:t}$  (given  $c_{1:N_c}$  and  $s_{1:N_s}$ ,  $\mathcal{Q}$ ,  $\mathbf{B}$  and  $\mathbf{Z}_{0:t}$  are not relevant for the estimation problem), all statistical information is contained in the posterior pdf  $p(r_{0:t} | \mathbf{y}_{1:t}, s_{1:N_s})$ , which can be approximated by means of a particle filter. Specifically, the dynamic model (1) is linear in  $r_t$  conditional on  $v_t$ , hence the recursive decomposition

$$p(r_{0:t} | \mathbf{y}_{1:t}, s_{1:N_s}) \propto p(\mathbf{y}_t | r_t, s_{1:N_s}) p(r_t | r_{0:t-1}) p(r_{0:t-1} | \mathbf{y}_{1:t-1}, s_{1:N_s}) \quad (24)$$

enables the application of the mixture Kalman filter (MKF) technique [Chen & Liu, 2000] to build a point-mass approximation of the posterior pdf,

$$p(r_{0:t-1} | \mathbf{y}_{1:t-1}, s_{1:N_s}) \approx \sum_{i=1}^M w_{t-1}^{(i)} \delta_i(r_{0:t-1}), \quad (25)$$

where  $\delta_i$  is the delta measure centered at  $r_{0:t-1}^{(i)}$  and  $\{w_{t-1}^{(i)}\}_{i=1}^M$  are normalized importance weights [Doucet et al., 2000]. Given the set  $\Omega_{t-1} = \{r_{0:t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^M$ , we can apply the sequential importance sampling (SIS) [Doucet et al., 2000]. algorithm to recursively compute  $\Omega_t$ . For  $i = 1, \dots, M$ , the following steps are recursively applied:

1. Importance sampling: Draw  $r_t^{(i)} \sim p(r_t | r_{0:t-1}^{(i)})$ .
2. Weight update:

$$w_t^{(i)*} = w_{t-1} p(\mathbf{y}_t | r_t^{(i)}, s_{1:N_s}) \quad \text{and} \quad w_t^{(i)} = w_t^{(i)*} / \sum_{k=1}^M w_t^{(k)*}. \quad (26)$$

Resampling steps must also be applied (although not necessarily at each  $t$ ) to avoid weight degeneracy [Doucet et al., 2000]. Since the likelihood  $p(\mathbf{y}_t | r_t^{(i)}, s_{1:N_s})$  can be computed, by assumption of the model, and several methods are available for resampling, the only difficulty in the application of this algorithm is sampling from the prior  $p(r_t | r_{0:t-1})$ . The latter can be obtained from the Kalman filter (KF) equations [Kalman, 1960]. To be specific, let us note that the pair of equations jointly given by (1),

$$v_t = v_{t-1} + u_{v,t} \quad (27)$$

$$\Delta_t = r_t - r_{t-1} = T v_{t-1} + u_{r,t}, \quad (28)$$

where  $[u_{r,t}, u_{v,t}]^T = \mathbf{u}_t$  in (1), form a linear-Gaussian system. Hence, the posterior pdf of  $v_t$ , given a specific sequence  $r_{0:t}$  is complex Gaussian,  $p(v_t | \Delta_{1:t}, r_0) = CN(v_t | \mu_t^v, \sigma_{v,t}^2)$ , with posterior mean and variance,  $\mu_t^v$  and  $\sigma_{v,t}^2$ , respectively, that can be recursively computed using the KF recursion [Haykin, 2001],

$$\sigma_{v,t|t-1}^2 = \sigma_{v,t-1}^2 + T^2, \quad (29)$$

$$g_t = T^2 \sigma_{v,t|t-1}^2 + \frac{1}{4} T^4, \quad (30)$$

$$\mu_t^v = \mu_{t-1}^v + \sigma_{v,t|t-1}^2 \frac{T(\Delta_t - T\mu_{t-1}^v)}{g_t}, \quad (31)$$

$$\sigma_{v,t}^2 = \sigma_{v,t|t-1}^2 \left( 1 - \sigma_{v,t|t-1}^2 \frac{T^2}{g_t} \right), \quad (32)$$

where  $g_t$  is the Kalman gain and  $\sigma_{v,t|t-1}^2$  is the variance of  $v_t$  conditional on  $\Delta_{1:t-1}$ . Moreover, the normalization constant of  $p(v_t | \Delta_{1:t}, r_0)$  is  $p(\Delta_t | \Delta_{1:t-1}, r_0) = p(r_t - r_{t-1} | r_{0:t-1})$ , which is also complex Gaussian and can be analytically found. In our specific model we obtain

$$p(r_t | r_{0:t-1}) = CN(r_t | r_{t-1} + T\mu_{t-1}^v, \sigma_{r,t|t-1}^2), \quad (33)$$

where  $\sigma_{r,t|t-1}^2 = T^2\sigma_{v,t-1}^2 + \frac{5}{4}T^4$ .

Therefore, the outlined MKF algorithm can be implemented using a bank of  $M$  KFs, one per particle. Given  $r_{0:t-1}^{(i)}$ , it is possible to (recursively and analytically) obtain  $p(r_t | r_{0:t-1}^{(i)})$ , which can be easily sampled to draw  $r_t^{(i)}$  in the importance sampling step. Moreover, both  $r_t$  and  $v_t$  can be estimated (in the MMSE sense),

$$\hat{r}_t = \sum_{i=1}^M w_t^{(i)} r_t^{(i)}, \quad \hat{v}_t = \sum_{i=1}^M w_t^{(i)} \mu_t^{(i)}, \quad (34)$$

by combining the outputs of the KF's, hence the name MKF. This methodology was applied to generic tracking problems in [Gustafsson et al., 2002].

Unfortunately, the standard MKF algorithm does not provide means to properly handle the unknown sensor positions  $s_{1:N_s}$ . To overcome this limitation, we resort to the DA-PF scheme of [Djurić et al., 2004]. The basic probabilistic relationship that we exploit to derive the new algorithm is obtained by means of the Bayes theorem and the repeated decomposition of conditional probabilities, namely

$$\begin{aligned} p(r_{0:t}, s_{1:N_s} | \mathbf{y}_{1:t}, \mathbf{Z}_{0:t}, \mathbf{B}) &\propto p(\mathbf{y}_t | r_t, s_{1:N_s}) p(\mathbf{Z}_t | s_{1:N_s}) p(r_t | r_{0:t-1}) \\ &\times \rho_t(s_{1:N_s}) p(r_{0:t-1} | \mathbf{y}_{1:t-1}, \mathbf{Z}_{0:t-1}, \mathbf{B}) \\ &= \prod_{k=1}^t p(\mathbf{y}_k | r_k, s_{1:N_s}) p(\mathbf{Z}_k | s_{1:N_s}) p(r_k | r_{0:k-1}) \\ &\times p(r_0) p(s_{1:N_s} | \mathbf{Z}_0, \mathbf{B}), \end{aligned} \quad (35)$$

$$\times p(r_0) p(s_{1:N_s} | \mathbf{Z}_0, \mathbf{B}), \quad (36)$$

where

$$\rho_t(s_{1:N_s}) = p(s_{1:N_s} | r_{0:t-1}, \mathbf{y}_{1:t-1}, \mathbf{Z}_{0:t-1}, \mathbf{B}) \quad (37)$$

is the posterior pdf of  $s_{1:N_s}$  at time  $t-1$ .

Assume that we are able to draw samples from  $\rho_t(s_{1:N_s})$ . Then, (35) shows that a SMC algorithm can be used to recursively approximate  $p(r_{0:t}, s_{1:N_s} | \mathbf{y}_{1:t}, \mathbf{Z}_{0:t}, \mathbf{B})$ . Indeed, if at time  $t-1$  the set of particles  $\Omega_{t-1} = \{r_{0:t-1}^{(i)}, s_{t-1,1:N_s}^{(i)}, w_t^{(i)}\}_{i=1}^M$  is available, then we can compute a point-mass approximation of the last factor in (35),

$$p_M(r_{0t-1} | \mathbf{y}_{1:t-1}, \mathbf{Z}_{0:t-1}, \mathbf{B}) = \int \sum_{i=1}^M w_{t-1}^{(i)} \delta_i(r_{0t-1}, s_{1:N_s}) ds_{1:N_s} \quad (38)$$

$$= \sum_{i=1}^M w_{t-1}^{(i)} \delta_i(r_{0t-1}), \quad (39)$$

where the integrand in (38) is the approximation of  $p(r_{0t-1}, s_{1:N_s} | \mathbf{y}_{1:t-1}, \mathbf{Z}_{0:t-1}, \mathbf{B})$  built from  $\Omega_{t-1}$ . Eq. (39) implies that we can start from the set  $\Omega_{t-1} = \{r_{0t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^M$  and exploit (35) to build  $\Omega_t$  via the MKF algorithm. Specifically,

$$r_t^{(i)} \sim p(r_t | r_{0t-1}^{(i)}) \quad (40)$$

$$s_{t,1:N_s}^{(i)} \sim \rho_t(s_{1:N_s}) \quad (41)$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} p(\mathbf{y}_t | r_t^{(i)}, s_{t,1:N_s}^{(i)}) p(\mathbf{Z}_t | s_{t,1:N_s}^{(i)}). \quad (42)$$

Moreover, (36) explicitly shows that, in order to start the recursion, we need to draw initial populations not only from the prior  $p(r_0)$ , but also from the posterior  $p(s_{1:N_s} | \mathbf{Z}_0, \mathbf{B})$ .

MMSE estimates of  $r_t$  and  $v_t$  are computed in the same way shown by Eq. (34), while  $\hat{s}_{t,1:N_s} = \sum_{i=1}^M s_{t,1:N_s}^{(i)} w_t^{(i)}$ .

To apply the MKF algorithm (40)-(42) we only need to specify how to approximate  $\rho_t(s_{1:N_s})$ . One of the simplest choices is to assume a Gaussian distribution built from the particles and weights at time  $t-1$ , i.e.,

$$\rho_t(s_{1:N_s}) \approx \prod_{n=1}^{N_s} CN(s_n | \bar{s}_{t-1,n}, \sigma_{t-1,s,n}^2), \quad (43)$$

where  $\bar{s}_{t-1,n} = \sum_{i=1}^M w_{t-1}^{(i)} s_{t-1,n}^{(i)}$  and  $\sigma_{t-1,s,n}^2 = \sum_{i=1}^M w_{t-1}^{(i)} (s_{t-1,n}^{(i)} - \bar{s}_{t-1,n})^2$ . This approximation is easy to sample and still provides an acceptable performance, as will be numerically shown in Section 7.

## 6. Fusion

### 6.1 Point estimation

Assume that the tracking algorithms, either the APF or the DA-MKF, run independently in the separate DFCs, using the same sequence of sensor observations<sup>2</sup>,  $\mathbf{y}_{0:t}$ , but different DFC

<sup>2</sup>This is not a requirement for fusion based on point estimation. The same technique could be applied if each DFC collected a different sequence  $\mathbf{y}_{n,1:t}$ ,  $n = 1, \dots, N_c$ .

observations,  $\mathbf{z}_{n,0:t}$ ,  $n = 1, \dots, N_c$ , instead of the whole set  $\mathbf{Z}_{0:t}$ . At any time  $t$ , the PF can yield an MMSE point estimate of the target and sensor positions. Let  $\hat{r}_{0:t}(n)$  and  $\hat{s}_{1:N_s}(n)$  denote the estimates computed at the  $n$ -th DFC. We wish to combine all available estimates coherently to obtain an improved estimator. However, we must take into account the possibility that  $n$ -th DFC estimates may be rotated around the location  $c_{n_t}$  as a consequence of the insensitivity of the observations  $\mathbf{y}_t$  and  $\mathbf{z}_{n,t}$  to the angle between  $r_t$  and  $s_k$ ,  $k=1, \dots, N_s$ , and the angle between  $s_k$  and  $c_n$ , respectively.

In order to correct any possible rotation and guarantee the computation of a coherent average of the available estimates, we propose the following fusion rule for the overall target and sensor position estimator,

$$\hat{r}_t = \frac{1}{N_c} \sum_{n=1}^{N_c} \hat{c}_n + (\hat{r}_t(n) - \hat{c}_n) e^{j\phi_n} \quad (44)$$

$$\hat{s}_{t,k} = \frac{1}{N_c} \sum_{n=1}^{N_c} \hat{c}_n + (\hat{s}_{k,t}(n) - \hat{c}_n) e^{j\phi_n}, \quad (45)$$

where  $k=1, \dots, N_s$  and the correction angles  $\phi_1, \dots, \phi_{N_c}$  are selected to minimize the mismatch with respect to the estimates produced by DFC 1, which is (arbitrarily) chosen as a reference. Specifically,

$$\begin{aligned} \phi_n = \arg \min_{\phi \in [0, 2\pi)} & \left\{ \sum_{i=1}^{N_s} \left| \hat{s}_i(1) - \left[ \hat{c}_n + (\hat{s}_i(n) - \hat{c}_n) e^{j\phi} \right] \right|^2 \right. \\ & \left. + \sum_{k=1-L_t}^t \left| \hat{r}_k(1) - \left[ \hat{c}_n + (\hat{r}_k(n) - \hat{c}_n) e^{j\phi} \right] \right|^2 \right\}, \quad (46) \end{aligned}$$

where  $L_t > 0$  is a delay lag and  $j = \sqrt{-1}$ . Problem (46) can be solved using the ARS algorithm described in Section 4.

## 6.2 Distributed implementation of PFs

In this section we describe how a centralized PF can be implemented in a distributed manner, such that each DFC updates a distinct subset of particles and generates individual estimates that can subsequently be combined optimally. For this approach to be formally sound, we require that all observations (in particular, the sequence  $\mathbf{Z}_{0:t}$ ) be available to all DFCs. If this is not the case, the fusion technique described here can still be used, but it becomes an approximation and optimality cannot be claimed.

We propose to use the resampling with non-proportional allocation (RNA) method of [Bolić et al., 2005, Míguez, 2007] 7 to distribute the MKF tracking algorithm over the  $N_c$  DFCs. Let us split the overall particle set  $\Omega_t$  into  $N_c$  subsets, one per DFC, denoted as

$\Omega_{n,t} = \{r_{0:t}^{(n,i)}, s_{t,1:N_s}^{(n,i)}, w_t^{(n,i)}, W_t^{(n)}\}_{i=1}^{M_n}$ ,  $n = 1, \dots, N_c$ , and such that  $\sum_n M_n = M$ . The

weights in  $\Omega_{n,t}$  are normalized locally, i.e.,  $\sum_{i=1}^{M_n} W_t^{(n,i)} = 1$ , and the sum of the unnormalized weights,  $W_t^{(n)*} = \sum_{i=1}^{M_n} W_t^{(n,i)*}$  are also kept in order to assess the relative value of each subset (the subsets  $\Omega_{n,t}$  are not equally “good” in general).

In the basic RNA scheme, an independent MKF algorithm (40)-(41) is run for each subset  $\Omega_{n,t}$  (i.e., for each DFC). This means that resampling is carried out locally (using any desired method) at each DFC and estimates are also computed locally,

$$\hat{r}_t(n) = \sum_{i=1}^{M_n} W_t^{(n,i)} r_t^{(n,i)} \quad \text{and} \quad \hat{s}_{1:N_s}(n) = \sum_{i=1}^{M_n} W_t^{(n,i)} s_{t,1:N_s}^{(n,i)}. \quad (47)$$

Optimal fusion is performed by combining the local estimates according to the sum-weights,  $W_t^{(n)*}$ , i.e., global MMSE estimates are computed as

$$\hat{r}_t = \frac{\sum_{n=1}^{N_c} W_t^{(n)*} \hat{r}_t(n)}{\sum_{k=1}^{N_c} W_t^{(k)*}} \quad \text{and} \quad \hat{s}_{1:N_s} = \frac{\sum_{n=1}^{N_c} W_t^{(n)*} \hat{s}_{1:N_s}(n)}{\sum_{k=1}^{N_c} W_t^{(k)*}}, \quad (48)$$

in such a way that only the local estimates and the sum-weights need to be transmitted to the DFC in charge of the fusion stage.

One limitation of this approach is that when the subset sizes,  $M_n, n=1, \dots, N_c$ , are not large enough, some particle filters may get relatively impoverished [Míguez,2007], i.e., it may eventually happen that, for some  $n$ ,  $W_t^{(n)*} \ll W_t^{(k)*}$ , for all  $k \neq n$ . In such a case, the corresponding  $n$ -th DFC becomes ‘useless’, since its local estimates are essentially irrelevant for the computation of the global estimates. A solution to this phenomenon (equivalent to the weight degeneracy in standard particle filters [Doucet et al., 2000]) is to periodically perform a local exchange (LE) of a small number of particles between pairs of DFCs. We propose a simple implementation of LE in which  $L < \min_n \{M_n\}$  particles from DFC  $n$  are transmitted to DFC  $n+1$ , for  $n=1, \dots, N_c - 1$  and  $L$  particles from DFC  $N_c$  are transmitted to DFC 1, i.e., particles are exchanged in a ring configuration.

### 7. Simulations

In order to provide illustrative numerical results, we have particularized the model of Section 2 to a network of power-aware sensors. Specifically, the measurement function  $f_s(\cdot; \cdot)$  has the form

$$f_s(d, \varepsilon) = 10 \log_{10} \left( \frac{1}{d^2} + \eta \right) + \varepsilon, \quad (dB) \quad (49)$$

where  $\eta = 10^{-6}$  accounts for the sensitivity of the measurement device (-60 dB). The  $n$ -th sensor transmits its measurement,  $y_{n,t}$ , only if it corresponds to a distance  $d_{n,t} < S_y = 50$



m (i.e.,  $y_{n,t} > -33.97$  dB) and otherwise remains silent. A transmission failure can also occur, with probability  $\beta = 10^{-4}$ . The observational noise,  $\varepsilon$ , is zero mean Gaussian but, depending on whether the power observation is carried out at a sensor node or at a DFC node, its variance is assumed different. In particular  $\varepsilon_{n,t}^y \sim N(0,2)$ , for sensors, and, for DFCs,  $\varepsilon_{i,n,t}^z$  and  $\varepsilon_{i,n}^0$  are identically distributed according to the Gaussian pdf  $N(0,10^{-2})$ . Therefore, the likelihoods, namely,

$$p(y_t | r_t^{(m)}, s_{1:N_s}^{(m)}) = \prod_{l=1}^{N_t} p(y_{\kappa(l),t} | r_t^{(m)}, s_{\kappa(l)}^{(m)}) \quad (50)$$

$$p(\mathbf{Z}_t | s_{1:N_s}^{(m)}, c_1, \hat{c}_{2:N_c}) = \prod_{i=1, n=1}^{N_t, N_c} p(z_{\kappa(i),n,t} | s_{t,\kappa(i)}^{(m)}, \hat{c}_n) \quad (51)$$

are Gaussian with known mean and variance.

At time zero, the sensors detect all other nodes which are closer than  $S_u=50$  m. Since observations are obtained from function  $f_s(\cdot, \cdot)$ , with the parameters already described for the sensors, the probability of detection is

$$p_d(d_{n,k}^s, S_u) = \Phi_N \left( \frac{\bar{P}_{n,k} - P_u}{\sqrt{10^{-2}}} \right), \quad (52)$$

where  $\Phi_N(\cdot)$  is the standard (zero mean, unit variance) Gaussian cumulative distribution function,  $\bar{P}_{n,k} = f_s(d_{n,k}^s, 0)$ ,  $P_u = f_s(S_u, 0) = -33.97$  dB and  $10^{-2}$  is the variance of the observational noise.

The state priors are  $p(r_0) = CN(r_0 | 0, 5)$  and  $p(v_0) = CN(v_0 | 0, 10^{-2})$  and the state equation parameters are  $T = \frac{1}{2}$  s and  $\sigma_u^2 = \frac{1}{5}$ . There are  $N_c = 4$  DFCs and  $N_s = 23$  sensors in the network. We assume  $c_1 = 0$ , while the others have complex Gaussian priors with equal variance  $\sigma_c^2 = 25$  and means  $\mu_2^c = -50 + j35$ ,  $\mu_3^c = 45 - j37$  and  $\mu_4^c = 36 + j45$  (where  $j = \sqrt{-1}$ ), respectively. This prior pdf's are used to randomly draw initial estimates of  $c_{2:4}$  which are used as inputs to the ARS algorithm that solves (5), the other parameters being  $\nu = 2$ ,  $R_{max} = 15$ ,  $R_{min} = 10^{-4}$ . The ARS algorithm for problem (6) receives as inputs a sensor position drawn from  $U(\mathcal{C})$  (where  $\mathcal{C}$  is the square centered at 0 with sides of length 160 m),  $R_{max} = 200$ ,  $r_{min} = 10^{-4}$  and  $\nu = 2$ . The ARS procedures are iterated 3000 times for (5) and 300 times for each (6). The estimates  $\hat{s}_{1:N_s}$  are then used to build the first proposal pdf in the PMC procedure. The corresponding variance is  $\sigma_s^2(0) = \frac{1}{2}$  and the subsequent proposals are computed by shrinkage, with parameter  $a = 0.7$ . We iterate the PMC algorithm 200 times with 1600 particles.

Resampling, via the RNA scheme, is performed every 5 time steps of the tracking algorithms. The latter are run with  $M=1600$  particles and each DFC is assigned  $M_n = M/N_c = 400$  particles (for  $n=1,2,3,4$ ). We assume a local exchange of particles every 4 resampling steps, with  $L=8$  particles being transmitted from DFC  $n$  to DFC  $n+1$  and from DFC  $N_c$  to DFC 1.

Figure 1 shows an example of estimation of the DFC and sensor positions,  $c_{2:N_c}$  and  $s_{1:N_s}$ , respectively, using the available data at time  $t=0$  and the ARS algorithm. It is observed that MAP estimation using the observations in the initialization stage can be very accurate. However, the simulation results also illustrate the ambiguity in position estimation that arises due to the insensitivity of the available measurements to rotations. The plot shows that the estimation of node locations near the origin (where angle errors have little effect) is clearly more accurate than for nodes placed far away from the 0 point, where rotation errors cause an apparent shift of the estimates with respect to the true values.

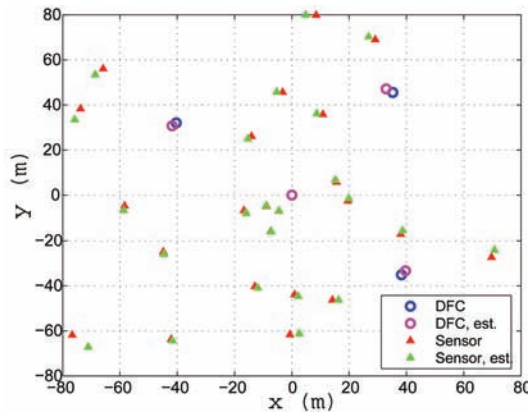


Fig. 1. Example of node MAP position estimation at the initialization stage using the ARS algorithm. The phenomenon of rotation ambiguity (due to the angle insensitivity of the available measurements) can be clearly observed for nodes located far from the 0 (origin) point.

Next we turn attention to the performance of the schemes that use independent PFs at separate DFCs and employ the proposed angle correction method to fuse the  $N_c$  available estimates coherently, as described in Section 6.1. We recall that, in this scenario, the observations available to the PF in the  $n$ -th DFC during the tracking stage are  $y_{1:t}$  and  $z_{n,1:t}$ . The ARS algorithm that approximates the solutions (correction angles) in problem 46 is iterated 100 times, with initial values  $\phi_{n>1} = 0$  and time lag  $L_t=20$  (for  $t>20$ , and  $L_t=t$  otherwise).

Figure 2 (left) shows the mean absolute target position error attained with the APF and MKF algorithms in this scenario. These results are the average of 500 independent simulation trials. The APF technique turns out to be clearly superior and outperforms the MKF method by  $\approx 0.6$  m of accuracy. This means that the improved importance function employed by the APF algorithm (which takes into account  $\mathbf{y}_t$  and  $\mathbf{z}_{n,t}$  when drawing the particles at time  $t$ )

provides an error reduction superior to the analytical integration of the velocity process carried out by the MKF method. We must remark that the position errors shown in this plot are corrected to remove the insensitivity, inherent to the proposed model, to rotations of the complete system (including the target and the nodes) around point 0, where DFC 1 is assumed to be located.

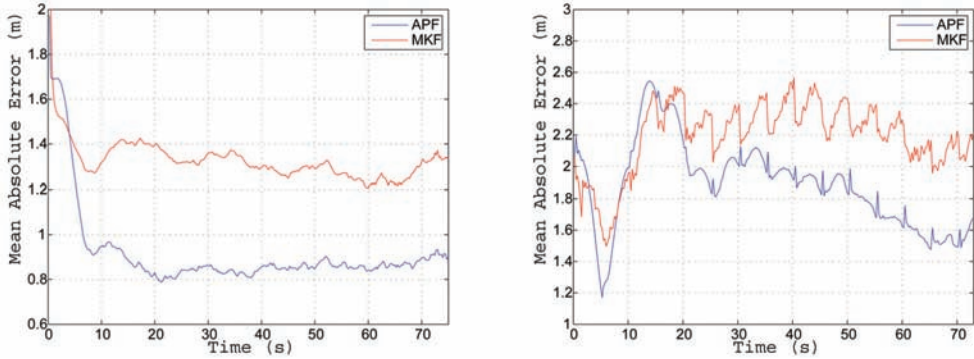


Fig. 2. Averaged absolute error in the estimation of the target position, measured in meters (m). *Left*: Tracking is carried out using independent APF and DA-MKF algorithms at each DFC and performing fusion by coherent combination of point estimates. *Right*: Tracking is carried out using a distributed implementation of the APF and DA-MKF algorithms over the  $N_c=4$  DFCs, by means of the RNA technique.

Figure 3 (left) depicts the mean absolute error in the estimation of the sensor positions attained by the schemes built around the APF and MKF algorithms. Again, these results are the average of 500 independent simulation trials. In this case, more accurate results are obtained with the MKF algorithm. This means that the density-assisted approach to the estimation of fixed parameters in the MKF scheme is more efficient than the adaptive kernel approximation employed by the APF procedure. The difference, however, is small ( $\approx 0.06$  m) and the superior sampling efficiency of the proposal function in the APF technique obviously has a dominant role in the overall performance of the tracker.

Figures 2 (right) and 3 (right) show the results, for the absolute errors in the estimation of the target and sensors positions, obtained when we apply the RNA algorithm described in Section 6.2. The curves have been obtained by averaging the results of 500 independent simulation trials. In this scenario, the different DFCs cooperate to share the observations  $\mathbf{z}_{n,t}$ ,  $n=1, \dots, N_c$ , hence the matrix  $\mathbf{Z}_t$  is available for all DFCs at time  $t$ . Again, the APF tracker turns out more efficient than the MKF in turns of target positioning, whereas the MKF yields better estimates of the sensor locations. Both plots show a clear 'step' shape. The reason is the sudden improvement in the estimates that occurs when a local exchange of particles is carried out as a part of the RNA procedure. This suggests the need to increase the number of exchanged particles ( $L=8$  for this set of simulations) or even the number of particles assigned to each DFC, which seems too small to exploit the potential of the RNA scheme. Indeed, the accuracy of positioning when using coherent point estimation is superior (for the APF, there is an advantage of  $\approx 0.8$  meters in absolute error). This fact should be attributed to the averaging of the angle error that is carried out when selecting the correction angles  $\phi_{2:N_c}$  in Eq. (46). Nevertheless, our simulation results (not shown) indicate that RNA-based schemes

always outperform point-estimation methods when the system runs for  $\approx 300$  time steps or more.

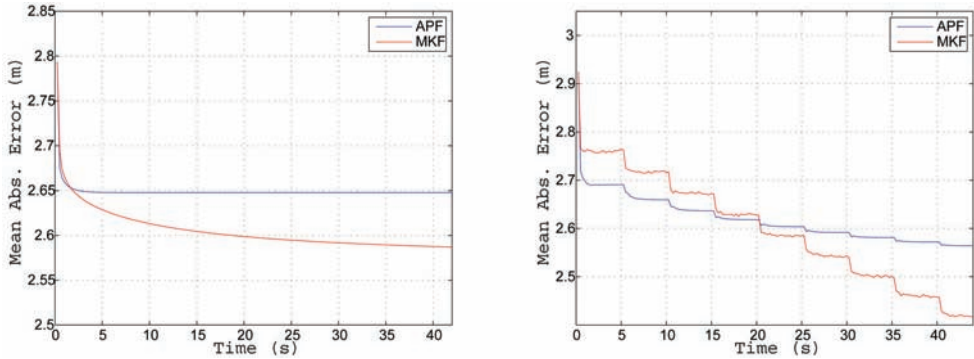


Fig. 3. Averaged absolute error in the estimation of the sensor positions, measured in meters/second (m). *Left*: Estimation is carried out using independent APF and DA-MKF algorithms at each DFC and performing fusion by coherent combination of point estimates. *Right*: Estimation is carried out using a distributed implementation of the APF and DA-MKF algorithms over the  $N_c=4$  DFCs, by means of the RNA technique.

### 8. Summary

We have proposed a novel scheme for joint node localization and target tracking in wireless sensor networks using Monte Carlo methods. The proposed approach does not require the aid of beacons in order to locate the network nodes. Instead, it resorts to a novel combination of Monte Carlo optimization and iterated sampling procedures in order to generate an initial population of node locations with sufficient quality. Starting from this population, we have described novel particle filtering algorithms that recursively track the target position *and* sequentially generate new samples of node positions, as new data become available, in order to improve node positioning. For networks equipped with more than one data fusion center, we have also proposed two schemes that enable the combination of the estimates obtained by different fusion centers, possibly using different data. Computer simulation results have also been presented to illustrate the performance of the proposed techniques.

### 9. Acknowledgements

This work has been partially supported by the Ministry of Science and Innovation of Spain (project MONIN, ref. TEC-2006-13514-C02-01/TCM), the Ministry of Industry, Tourism and Commerce of Spain (project TIMI, ref. CENIT-2002-2007) and the Autonomous Community of Madrid (project PROMULTIDIS-CM, ref. S-0505/TIC/0233).

### 10. References

M. J. Appel, R. Labarre, and D. Radulovic. On accelerated random search. *SIAM Journal of Optimization*, 14(3):708–730, 2003.

- M. Bolić, P. M. Djurić, and S. Hong. Resampling algorithms and architectures for distributed particle filters. *IEEE Transactions Signal Processing*, 53(7):2442–2450, July 2005.
- O. Cappé, A. Gullin, J. M. Marin, and C. P. Robert. Population monte carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.
- R. Chen and J. S. Liu. Mixture Kalman filters. *Journal of the Royal Statistics Society B*, 62:493–508, 2000.
- D. Crisan and A. Doucet. A survey of convergence results on particle filtering. *IEEE Transactions Signal Processing*, 50(3):736–746, March 2002.
- M. H. DeGroot and M. J. Schervish. *Probability and Statistics*, 3rd ed. Addison- Wesley, New York, 2002.
- P. M. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Míguez. Particle filtering. *IEEE Signal Processing Magazine*, 20(5):19–38, September 2003.
- P. M. Djurić, M. F. Bugallo, and J. Míguez. Density assisted particle filters for state and parameter estimation. In *Proceedings of the 29th IEEE ICASSP*, May 2004.
- A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo Sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.
- A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer, New York (USA), 2001.
- L. Fang, W. Du, and P. Ning. A beacon-less location discovery scheme for wireless sensor networks. In *Proceedings of INFOCOM*, volume 1, pages 161–171, March 2005.
- F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund. Particle filters for positioning, navigation and tracking. *IEEE Transactions Signal Processing*, 50(2):425–437, February 2002.
- S. Haykin. *Adaptive Filter Theory*, 4th Edition. Prentice Hall, Information and System Sciences Series, 2001.
- A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky. Nonparametric belief propagation for self-localization of sensor networks. *IEEE Transactions on Selected Areas in Communications*, 23(4):809–819, April 2005.
- R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45, 1960.
- J. Liu and M. West. Combined parameter and state estimation in simulation-based filtering. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*, chapter 10, pages 197–223. Springer, 2001.
- J. Míguez. Analysis of selection methods for cost-reference particle filtering with applications to maneuvering target tracking and dynamic optimization. *Digital Signal Processing*, 17:787–807, 2007.
- N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal. Locating the nodes. *IEEE Signal Processing Magazine*, 22(4):54–69, July 2005.
- M. K. Pitt and N. Shephard. Auxiliary variable based particle filters. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*, chapter 13, pages 273–293. Springer, 2001.
- B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter*. Artech House, Boston, 2004.
- G. Sun, J. Chen, W. Guo, and K. J. R. Liu. Signal processing techniques in networkaided positioning. *IEEE Signal Processing Magazine*, 22(4):12–23, July 2005.
- M. Vemula, M.F. Bugallo, and P.M. Djurić. Fusion of information for sensor selflocalization by a Monte Carlo method. In *Proceedings of ICIF*, July 2006.