Available online at www.sciencedirect.com
ScienceDirect
Robotics and Computer-Integrated Manufacturing 24 (2008) 200–216
Robotics and Computer-Integrated Manufacturing
www.elsevier.com/locate/rcim

STEP-NC enabled on-line inspection in support of closed-loop machining

Fiona Zhao, Xun Xu*, Shane Xie

Department of Mechanical Engineering, University of Auckland, 20 Symonds Street, Auckland, New Zealand

Received 16 May 2006; received in revised form 21 August 2006; accepted 20 October 2006

Abstract

The object-oriented STEP-NC data model provides a seamless and integrated programming interface for on-machine (or also known as on-line or in-line) inspections as well as interoperable manufacturing. This paper proposes a STEP-NC data model for on-line inspections. A framework of STEP-NC enabled closed-loop machining is also presented. The aim is to achieve a fully closed computer-aided design, process planning, machining and inspection chain. A new version of STEP-NC Interpreter has been developed to implement the proposed framework. A case study is included to demonstrate the implementation.

Keywords: Machining; On-machine inspection; Closed-loop machining; STEP-NC

1. Introduction

Manufacturing is a collection of interrelated activities including product design and documentation, material selection, planning, production, quality assurance, management and marketing of goods [1]. Closed-loop machining (CLM) is a method to maximize efficiency of a machining process by maintaining a tight control in a manufacturing system. It is normally regarded as the highest level of CNC automation [2]. The history of CLM may date back 50 years ago when manual tolerance charts were introduced into the industry for dimensional control in machining processes [3]. It is regarded that reliable machines, robust processes, automatic data collection and continuous improvement of machining process are the necessary elements of a CLM system. A closed loop, which always includes CNC machines, may also include CAM, CAPP or even CAD. In this research, a closed loop with a limited scope (including CNC and CAM) is considered [2,4].

Inspection is an essential element in a closed-loop manufacturing system. It can gather data to achieve precise measurement and monitor machine tool's performance during the operation. In most CLM systems, coordinate measuring machine (CMM) is widely employed as a way of collecting the measured data from a machined part. However, there are problems in this type of CLM systems. These problems relate to time, compatibility and data modeling.

(1) Inspection using CMM is an off-line operation. It leads to increase in machining cycle time due to relocating workpiece between machines and CMMs.
(2) Inspection using CMM is typically a three stage activity: programming, execution of the program and evaluation of the results. These activities are often carried out on separate systems which lead to complex interface problems. Common exchange standards are not new to the metrology industry. The Dimensional Measurement Interface Standard (DMIS) [5], which is widely used for CMM data interoperability, allows programs to be shared between different pieces of

*Corresponding author. Tel.: +64 9 373 7599; fax: +64 9 373 7479.
E-mail address: x.xu@auckland.ac.nz (X. Xu).

0736-5845/$ - see front matter
doi:10.1016/j.rcim.2006.10.004

measurement equipment. It is designed primarily for the transfer of inspection programmes and results between CAD/Off-Line Programming (OLP) and CMMs. DMIS deals with tolerancing, but is limited in its ability to handle the geometry of the component as a whole. The problem of incompatible measurement and inspection software still exists [6,7].

(3) The inspection results cannot be easily incorporated into the current NC program, which is G-code based [4,8]. This is because it is a low level programming language. Geometrical information about the part being machined and inspected is not preserved.

On-machine inspection, which is also known as on-line inspection, offers an attractive alternative solution. On-machine inspection enables measurement taking, data collection as well as data feedback and process adjustment fully automated and integrated. With on-machine inspections, a part can be measured at the machine and corrected there to avoid relocation between the CNC and a CMM. Small samples can be made and then checked immediately. Problems such as overcompensation and under-compensation can be identified at an early stage. Incorporating on-machine inspection, the benefits of CLM are multiple [4]:

(1) CLM can reduce the reject rate by proofing fixture and part setups, automating offset adjustments, and monitoring the machine's "health" through machine self-checks;
(2) It can reduce part cost by promoting tooling standardization and reducing operator intervention;
(3) Inspection costs can be reduced through eliminating hard gauging, increasing the flexibility of measuring methods, avoiding cost of acquiring a CMM, and overcoming the CMM bottlenecking problems; and
(4) CLM can collect data about parts, processes and equipment for real time, adaptive control.

To summarize, utilizing CLM with on-machine inspection can increase reliability of a machine, provide a more controlled environment and reduce human operating errors, such as those concerning offset modification.

2. Data models for CLM

The common data model for CLM is G-code. G-code is a worldwide well-accepted standard for CNC machine tools. It can be extended for on-machine inspections. When this is the case, inspection and machining operations are characterized by a complex sequence of manual and automated activities based on various software applications and exchange formats. A number of irreversible data conversions (e.g. CAD/CAM data to G-code) with loss of accuracy and context hinder a seamless information flow. Lack of geometry information in the code leads to a uni-directional information flow. Changes made in an NC program after inspection cannot be fed back to the CAM system, as the context of single movement or switching instructions normally gets lost in a G-code NC program [9]. Therefore, the measurement results from inspections cannot be fed back to the CAM system directly. Manual reworking is unavoidable, and the manufacturing loop is not fully closed. Modern machining processes often require more information of the product data to be made available at the controller and to be used in a CLM environment. With geometrical information about a part available at the controller, the controller can carry out "positive" inspection operations. The advantage of CNC controller can be fully utilized. Shop floor experiences can be used to develop new way of machining the part [7,10].

The STEP-NC (ISO 14649) data model [11], in contrast, provides a higher level of information for manufacturing processes including the part geometry and tolerances, hence enables a bi-directional information flow. This is the data model used in proposed CLM framework. STEP-NC not only eliminates the costly and inefficient process of post processing, it also establishes a unified environment for the exchange of information between product design applications, manufacturing process planning, and the machine tool on the shop floor. By allowing the change and process information to be fed back to the planning level, a closed STEP-NC-based machining process chain with data feedback and consistent data on every level can be realized. STEP-NC for the first time offers a uniform data interface for a closed-loop from the planning of the machining operations, to machine tools, inspections and finally back to the production planning process [2,7,26].

The ISO 14649 data models describe the information requirement for CNC controllers and how to represent those within the larger view of product data defined by the combined set of ISO 10303 application protocols (AP) [18]. ISO 14649 Part 10 specifies the process data that is common for all NC machining processes. These data elements describe the interface between a CNC controller and the programming system (i.e. CAM system or shop-floor programming system). A STEP-NC program includes both geometric and technological information. It can be described using ISO 14649 Part 10 together with a technology-specific part (e.g. ISO 14649-Part 11 for milling, Part 111 for milling tools, Part 12 for turning, Part 121 for turning tools, and/or Part 16 for touch probing inspection) [12–17]. Part 10 of ISO 14649 provides the control structures for the sequence of program execution, mainly the sequence of Workingsteps and associated machine functions. The "machining_schema" entity defined in Part 10 contains the definition of data types which are generally relevant for

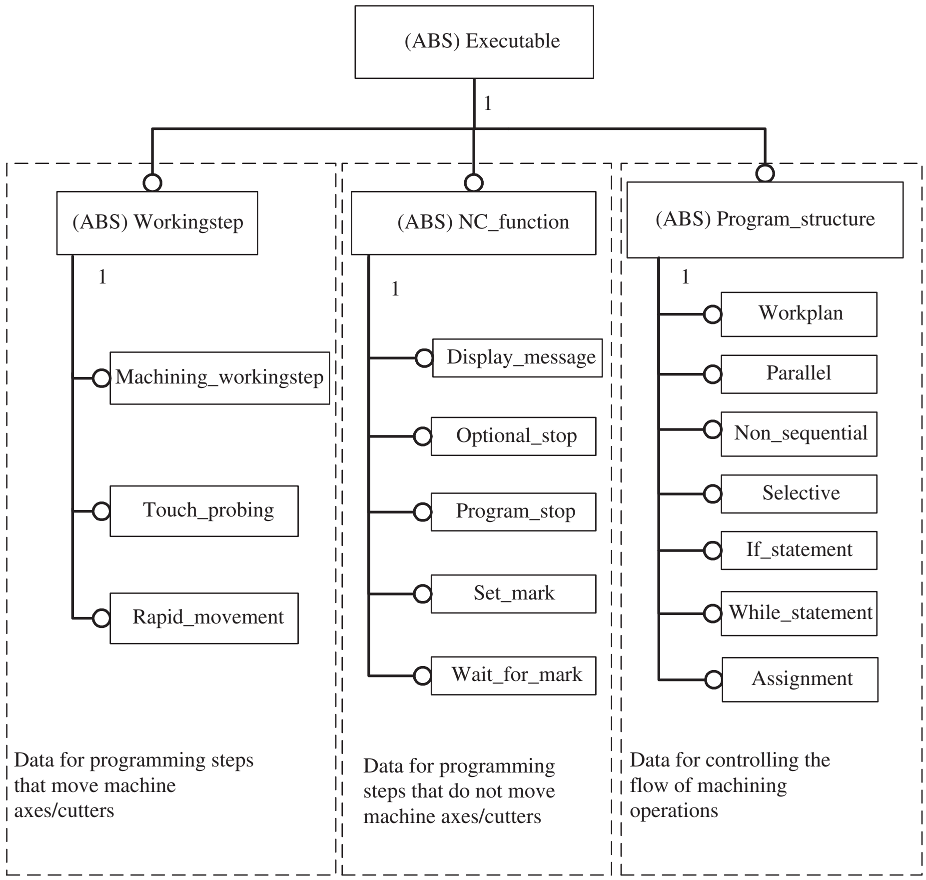

Fig. 1. Essential parts of STEP-NC data.

different technologies (e.g. milling, turning and grinding). Probing Workingstep is also defined in this part for inspection operations. However, the probing operations are specified in Part 16. The dimensional inspection data model is specified in ISO 10303 AP219 [18].

Fig. 1 shows the essential part of data in a STEP-NC file. It contains sequenced executable manufacturing tasks. There are three types of executables, (a) Workingstep; (b) NC functions and (c) Program structure. A Workingstep is a manufacturing task that describes manufacturing operations. The NC functions fulfill most of the miscellaneous functions that M codes used to do. The Program structure is either a Workplan or conditional controls.

3. Related research

In the past 40–50 years, research work has been carried out in all aspects of CLM such as measurement technology, that is, different ways of carrying out inspection operations, condition monitoring and machine reliability. The intention has been to prevent the equipment from breaking down during the production, and facilitate robust process. However, most of the research is based on data models using G-code for data exchange with CNCs. Therefore, they fall short of achieving integration of inspection operation and machining process in a seamless, integrated process chain. With design, manufacturing and inspection consolidated on one platform, i.e. STEP (and STEP-NC), the above-mentioned problems can be solved. Limited work has been carried out using STEP and implementations to realize a truly CLM with bi-directional information flow, including feedback of inspection results to process planning.

Brecher et al. in the Laboratory for Machine Tools and Production Engineering (WZL) at Aachen University, Germany developed a system for a closed-loop process chain, which intergraded inspections into the STEP-NC information flow [19]. The research presents a system that supports milling a workpiece, inspection of several workpiece features and feeding back the measured results to the product model. Their research focused on the closure of a broad process chain by integrating inspection activities into the STEP-NC-based process chain and feeding the results of the manufacturing operation in terms of the obtained measurement data, back to process planning. Their inspection operations are not real-time because of the use of a CMM. Currently, no CMM is capable of interpreting STEP-NC information. The STEP-NC inspection data had to be converted into suitable format using for example, Dimensional Measuring Interface Standard (DMS) [20] and

Dimensional Markup Language (DML) [21]. In fact, on-machine probing can eliminate this conversion. STEP-NC Part 16 offers the necessary data structure for the exchange of Geometric Dimensioning and Tolerancing (GD&T) data as well as process information.

In the United States, NIST, Boeing, General Electrics, Unigraphics and other some industry partners worked on a STEP-enabled CLM using ISO 10303 AP 238 to support probing capabilities of CLM [22]. ISO 10303 AP 238 is the Application Interpreted Model (AIM) for STEP-NC Application Reference Model (ARM), which is effectively ISO 14649. The demonstration in May 2005 highlighted the use of probing results collected on a CNC machine to generate modified AP238 data. At the demonstration, offsets were coupled due to the possible_misalignment, and resident single-axis offsets could not be used to accomplish the full transformation. Thus, one AP238 program was used for the probing, a second AP-238 program specified the machining in nominal coordinates, and the STEP-NC converter generated NC code using the nominal AP238 program and the acquired transformation just prior to machining. Only Conformance Classes 1 (CC1) and 2 (CC2) of AP238 were experimented at this moment. Further development is to include CC3 and CC4. As AP238 is much larger than ISO14649, it takes more storage space than ISO14649 and is also more complex in programming.

Newman et al. from the Loughborough University, UK [23] developed an inspection framework for closing the inspection loop through integration of information across the CAx process chain. The major feature of the proposed STEP-compliant inspection framework is the inclusion of high-level and detailed information in terms of an inspection Workplan, Workingstep, and a mechanism to feedback inspection results across the total CAx process chain. STEP-NC (ISO 14649-16), DMIS and AP219 are used as the basis for representation of product and manufacturing models. This research still mainly focused on the utilization of CMMs. Hence it is not an on-machine inspection system.

All of the above-mentioned research did not support a real-time, closed-loop machining by way of integrating inspection operations with machining operations. In this research, a framework of STEP-NC enabled, real-time CLM system is proposed. Touch probes are utilized. ISO 14649 is used for inspection planning and machining operations.

4. Proposed research

4.1. STEP-NC data model for on-machine inspection

The Part 16 data model of ISO14649 enables a closed process chain by enabling the machining processes to be integrated with inspections so that automated, seamless data feedback of the measurement results to the product model can be achieved. This is considered important for process optimization and documentation.

Part 16 focuses on touch probing operations, which can either be executed with a touch probes on a CNC machine tool or a CMM. It provides an interface between a CNC controller and a programming system for inspection. In this research, only on-machine touch probes are considered. Fig. 2 is a simplified EXPRESS-G diagram describing the functionality of STEP-NC enabled on-machine inspection (in this study on-machine-probing) planning.

In Fig. 2, four types of probing operations are defined: workpiece_probing, workpiece_complete_probing, tool_probing and probing_workingstep. They are specified in Parts 10 and 16. Their EXPRESS schema definitions are as follows.

```
ENTITY touch_probing
  ABSTRACT SUPERTYPE OF (ONE OF (workpiece_probing,
  workpiece_complete_probing, tool_probing))
  SUBTYPE OF (Workingstep, OPERATION);
  measured_offset: nc_variable;
END_ENTITY;
```

Entity touch_probing is the supertype of workpiece_probing, workpiece_complete_probing, tool_probing, and probing_workingstep. It defines a single inspection task and is inserted into a sequence of executables including machining operations.

```
ENTITY workpiece_probing
  SUBTYPE OF (touch_probing);
  start_position:axis2_placement_3d;
  its_workpiece: workpiece;
  its_direction: direction;
  expected_value:toleranced_length_measure;
  its_probe: touch_probe;
END_ENTITY;
```

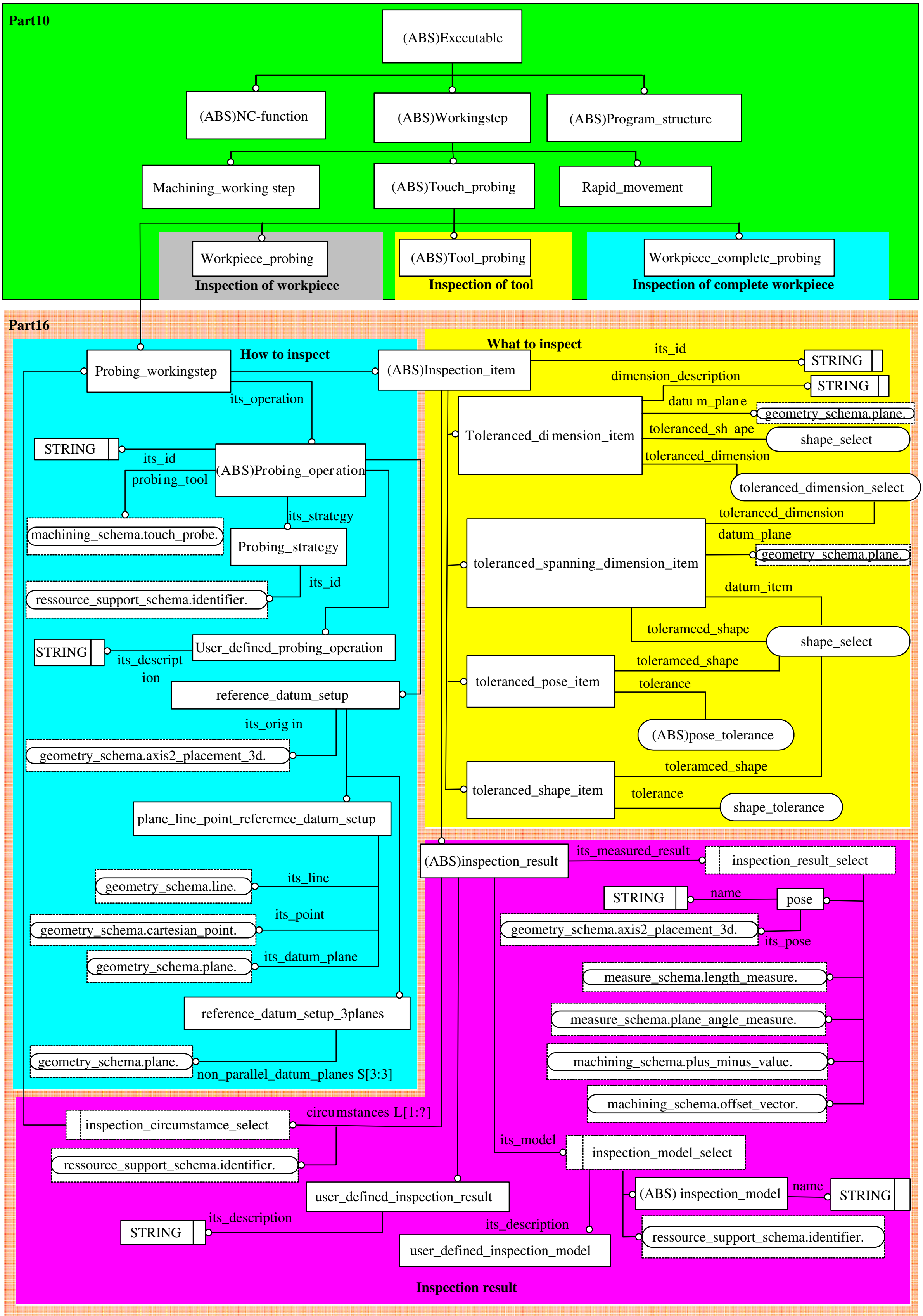

Fig. 2. EXPRESS-G diagram of on-machine inspection.

Entity workpiece_probing defines the probing of a dimension on a workpiece, with one axis movement from a starting location (start_position) to a nominal (expected) distance (expected_value). The starting location is known to be free of the workpiece along a direction (its_direction). The difference between the distance the actual probe travels and the nominal distance is recorded in the inspection result. This result can be used to determine possible corrective actions, such as re-machining, or to compute setup offsets.

```
ENTITY workpiece_complete_probing
  SUBTYPE OF (touch_probing);
  its_workpiece: workpiece;
  probing_distance: toleranced_length_measure;
  its_probe: touch_probe;
  computed_offset: offset_vector;
END_ENTITY;
```

Entity workpiece_complete_probing defines a complete measurement cycle at six locations of the workpiece. The six probing locations are determined by NC controller automatically based on the geometry of the workpiece. The translation and rotational offset of the workpiece is inspected and computed comparing with the design data and kept in computed_offset.

```
ENTITY tool_probing
  ABSTRACT SUPERTYPE OF (ONE OF (tool_length_probing, tool_radius_probing))
  SUBTYPE OF (touch_probing);
  offset: cartesian_point;
  max_wear:length_measure;
  its_tool:machining_tool;
END_ENTITY;
```

Entity tool_probing defines the probing of the length and width/diameter of a tool (its_tool). To do this, the tool starts its movement at a machine dependent start position. From that position the tool position is shifted to a fixed sensor position by an offset. Then the tool is moved in the direction of the sensor until coming to a contact. The difference between the fixed sensor position and the final contact position is the wear of the machining tool (max_wear). This is done in both longitudinal and perpendicular directions of the tool-axis.

```
ENTITY probing_workingstep
  SUBTYPE OF (touch_probing);
  its_operation: probing_operation;
  its_items: SET[1:?]OF inspection_item;
END_ENTITY;
```

Entity probing_workingstep defines a probing activity that is common for workpiece_probing, workpiece_complete_probing, and tool_probing. The probing activity has to be executed in order to inspect the given inspection items. It has two attributes: probing_operation and inspection_item. Their EXPRESS schemas are listed as follows.

```
ENTITY probing_operation
  ABSTRACT SUPERTYPE OF (user_defined_probing_operation);
  its_id:STRING;
  reference_datum:reference_datum_setup;
  its_strategy:OPTIONAL probing_strategy;
  probing_tool:touch_probe;
END_ENTITY;

ENTITY inspection_item
  ABSTRACT SUPERTYPE OF (ONEOF(toleranced_dimension_item, toleranced_spanning_dimension_
  item, toleranced_pose_item, toleranced_shape_item));
  its_id: STRING;
  toleranced_result: OPTIONAL inspection_result;
END_ENTITY;
```

```
ENTITY inspection_result
  ABSTRACT SUPERTYPE;
  its_measured_result: inspection_result_select;
  its_model: inspection_model_select;
  circumstances: LIST [1:?]OF inspection_circumstance_select;
END_ENTITY;
```

Entity probing_operation contains operational instructions and circumstances of the probing inspection operation (reference_datum and its_strategy). They describe how to carry out an inspection operation. Inspection_item can be considered as containing two parts: toleranced items and inspection result. Toleranced items denote the nominal values and tolerances of a workpiece. They make reference to one or several machining features on which the inspection operation is performed. Entity inspection_item describes the items to be inspected in the inspection operation, and provides a "container" to attach tolerances to geometrical elements. These toleranced elements can be attributes of features (e.g. diameter of a round hole), relations within one feature (e.g. distance between two sides of a pocket) or relations between two different items (e.g. distance of the centre-lines of two holes and perpendicularity of a hole with reference to a plane). Inspection results are stored in a container—inspection_result so that it can be fed back to update the machining operation.

In Fig. 2, all the attributes in these entities are defined by various schemas in ISO 10303, e.g. machining schema, geometry schema, measure schema and resource support schema. These schemas are the fundamental blocks of a STEP-NC data model. The information modeled by these schemas can be shared through a database without having to modify the underlying structure of the database. In ISO 14649, these schemas are treated as "referred schema"; hence re-defining in ISO 14649 is not necessary.

Dimensional inspection can occur at any stage of the life cycle of a product where checking for conformance with a design specification is required. In order to enable the exchange of inspection data between various systems, STEP AP219 information model—Dimensional Inspection Process Planning for CMMs—for dimensional inspection [18] was created. This application protocol specifies information requirements to manage dimensional inspection of solid parts or assemblies, i.e. administering, planning, and executing dimensional inspection as well as analyzing and archiving the results. One of the purposes of this research is to inspect without using CMMs. Therefore, AP219 is not considered.

A combined STEP schema (including ISO 14649-10, 11, 12 and 16) has been built. This schema defines both STEP-NC inspection data and machining data (milling and turning). As the inspection_result entity and inspection_model entity are Abstract Supertype, and they have no subtypes, new entities are created,

```
ENTITY user_defined_inspection_result
  SUBTYPE OF (inspection_result);
  its_description:STRING;
END_ENTITY;
ENTITY user_defined_inspection_model
  SUBTYPE OF (inspection_model);
  its_description:STRING;
END_ENTITY;
```

Entity user_defined_inspection_result is defined as a subtype of the abstract supertype inspection_result. It is designed to be the entity to store inspection results. The inspection result from the inspection operation is specified in this entity. The information of the inspection result includes the description of the measured feature and the circumstances in which the result is obtained. Entity user_defined_inspection_model is defined as a subtype of the abstract supertype inspection_model. This entity stores the specification(s) of the evaluation of the measured values.

The Express Engine 3.2 software [24] is used to debug the schema. Express Engine provides a STEP development environment in which STEP EXPRESS schema can be validated and compiled to the LISP language equivalent. It is also possible to build, check and validate data files based on the loaded EXPRESS schemata.

4.2. The framework for STEP-NC-enabled closed-loop machining

Use of ISO 14649-16 alongside other Parts in ISO 14649, makes it possible to consolidate machining and inspection operations in one single program by including inspection Workingsteps in a sequence of machining Workingsteps. However, when and how the inspection should be carried out are still tasks to be accomplished. This is regarded as "how-to-do" information and should be decided when the hardware specifications, e.g. machine tools and probes, become known. This is out of the scope of this research work.

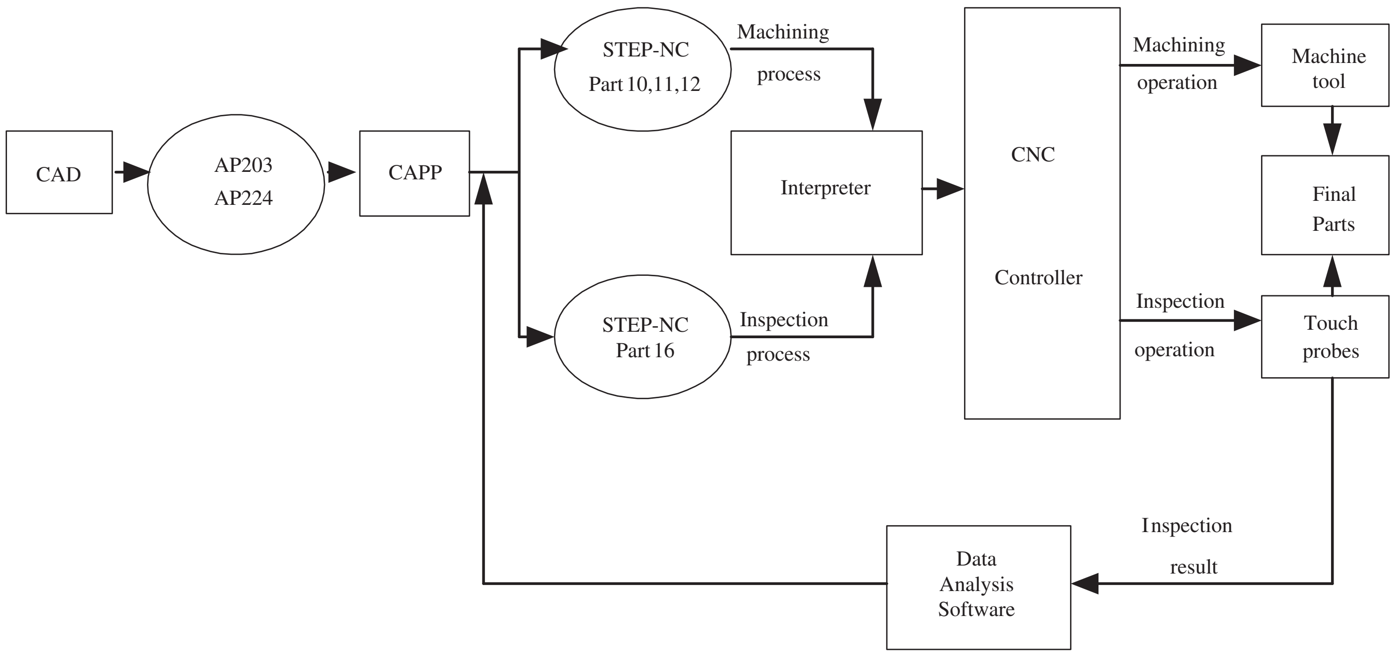

Fig. 3. STEP-NC enabled closed-loop machining process.

There are four types of inspections considered in this research:

Inspection Type I: Inspections that are carried out prior to the machining operations. This type of inspection is needed to provide the positional information of the workpiece to the CNC controller; hence the controller knows the actual location of the workpiece so that cutter paths can be planned/altered accordingly.

Inspection Type II: Inspections that are carried out at the end. This inspection is to make sure the planned operation has been successfully and accurately executed.

Inspection Type III: Inspections that are carried out during the machining process. This type of inspection is necessary to detect any abnormity on the workpiece or the tools, so that the controller can make changes in time. At what stage the inspection should be carried out during the machining process depends on the requirement of the workpiece. Parts that have very tight tolerances may need additional inspections in the machining processes, whereas parts that have looser tolerances may not need inspection at all.

Inspection Type IV: Inspections that are carried out for multiple setups. In this situation, inspections are necessary in each setup to provide accurate workpiece location information and new reference information after the previous machining operation.

The feature-based approach is adopted when representing inspection data. Whereas different types of features have been defined for design and manufacturing, they may not be utilized for inspection purposes. For example, a simple rectangular pocket in design may have been defined by removing a cuboid volume from a block and in NC machining the outer contour, the direction and the depth of the pocket are of interest. However, for pocket inspection, two pairs of apposing faces may be used as "inspection features". The data model proposed in this research can deal with both machining and inspection features. In the data model, attribute inspection_item provides a "container" to attach tolerances to geometrical elements by referencing features of the workpiece. That means each inspection_item links an explicit, tolerated geometrical element (e.g. toleranced length measure) to a specific parameterised or semantic attribute of a feature. For example, these tolerated geometrical elements can be attributes of features (e.g. diameter of a round hole), relations within one feature (e.g. distance between two sides of a pocket) or relations between two different items (e.g. distance of the centrelines of two holes). User and application-specific views can then filter the data according to the specific demands on the data and its representation. The inspection result as well as the circumstances in which the result was generated are stored in the inspection_result entity. This information is fed back to the STEP-NC data model to be compared with the required data so that necessary changes can be made to the subsequent machining operations. Fig. 3 shows the framework of this STEP-NC closed-loop machining process chain.

In the proposed closed-loop machining process, design work can be carried out in a CAD system (e.g. Pro/Engineer and SolidWorks), and saved in the format of STEP AP203. Features conforming to STEP AP224 are then constructed. The STEP-NC compliant CAPP software generates information for machining and inspection processes. This information is represented in a STEP-NC file (ISO 14649 Parts 10,11,12 and 16). This file contains the information about machining features, materials, Workingsteps, Workplan, tool path, machining operations, and inspections. This information is fed into a CNC controller for execution. During the machining process, depending on the design requirements, different inspections may need to be carried out. In each inspection operation, (1) the feature and dimension items that need to be measured are inspected; (2) the inspection results are then analyzed; (3) the analyzed results are compared with the required design specifications; (4) required changes are fed to the STEP-NC data model; and (5) new machining operations are generated by updating the original STEP-NC data model.

Each STEP-NC program may include machining Workingsteps as well as inspection Workingsteps. The inspection Workingsteps are inserted into the machining Workingstep list. During machining process, inspection Workingsteps are inserted at the place where inspections are necessary and then inspection operation is carried out before next machining Workingstep. Where to insert depends on the purpose of inspection. For example, if set-up information is required, an inspection operation before all machining Workingsteps is carried out. In this way, a real time, fully integrated closed-loop machining process is achieved.

5. Implementation

5.1. STEP-NC interpreter

The benefits of STEP-NC are obvious; but existing CNC machining centers still cannot accept STEP-NC data directly. An interpreter which "translates" STEP-NC data into the machining commands that a CNC can understand is a necessity. Fig. 4 shows the current way of using STEP-NC files in the machining process. In this research, a new version (Version 2.5) of the STEP-NC Interpreter is developed to interpret STEP-NC inspection data.

The first version of the STEP-NC Interpreter (Version 1.0) worked under UNIX environment. It was developed to process the first (milling) example in the STEP-NC ISO 14649-11 standard [25]. Canonical Machining Commands (CMCs) are generated as the output. The interpreter was then migrated to the Windows environment to become Version 2.0. Turning functions were later added to arrive at Interpreter Version 2.1. This research continues the work by adding inspection functions to the Interpreter, hence ISO 14649 Interpreter Version 2.5. It is intended to be able to process inspection operations together with machining operations.

ISO-14649 Part 16 schemas are used to build the data model for inspections. Only operations and features present in the first example of the STEP-NC ISO 14649-11 standard are considered, alongside some inspection operations that have been added. Table 1 shows the inspection operations and information this new version of Interpreter can handle. After parsing a Part 21 file, information about machining and inspection Workingsteps is extracted. This is currently demonstrated by Interpreter outputting a text file to show all the information extracted. The information is extracted through function execute_workingstep, where all Workingsteps are checked to see whether it is a machining_workingstep, rapid_movement or touch_probing. If it is a machining_workingstep, function execute_machining is called to carry out the machining operation. If it is a rapid_movement function, function execute_rapid is called. If it is a touch_probing, function execute_touch_probing is called (Fig. 5).

The touch_probing function is a key function in this version of the Interpreter. It calls other functions to carry out the interpreting process of all inspection data. The parameters required to perform inspection and machining of a feature are then extracted.

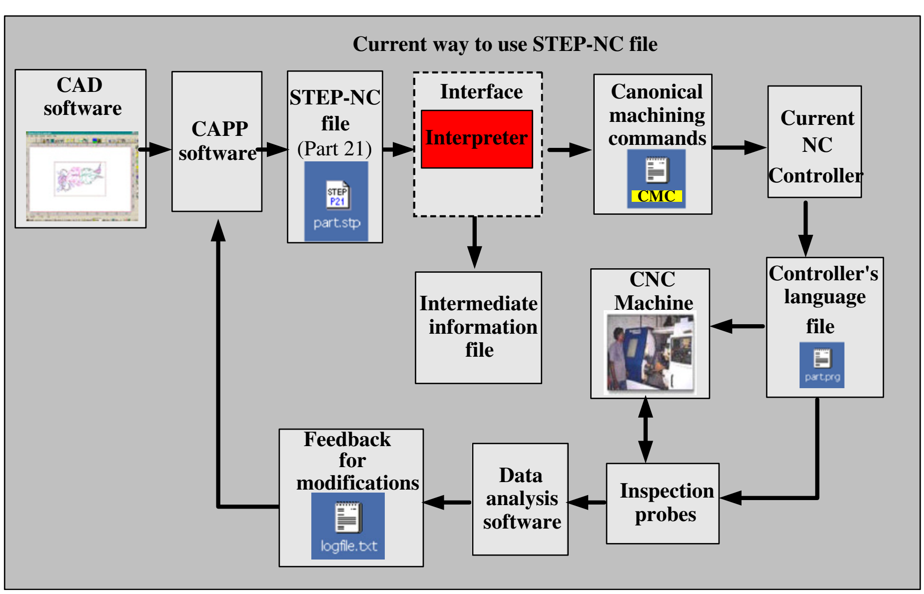

Fig. 4. Current way of using STEP-NC in a machining process.

Table 1
Inspection operation added to STEP-NC Interpreter Version 2.5

probing_workingstep	probing_operation user_defined_probing_operation
inspection_item	toleranced_dimension_item toleranced_spanning_dimension_item toleranced_pose_item toleranced_shape_item toleranced_length_measure
inspection_result	user_defined_inspection_result inspection_model_select inspection_result_select inspection_circumstance_select

5.2. Case study

A case study is carried out to demonstrate the proposed data model and the feasibility of the proposed framework. The STEP-NC data information must be transferred to a CNC controller and be understood by the controller so that the commands can be carried out for machining and inspection (Fig. 6).

The case study is based on the example in ISO 14649-11, "enriched" with some inspections. There are five machining Workingsteps in the original file. They are for milling the top surface of the workpiece; drilling and reaming the hole; and rough and finish milling the pocket.

The following set of codes shows the first machining Workingstep.

```
#10 = MACHINING_WORKINGSTEP('WS FINISH PLANAR FACE1',#62,#16,#19,$);
#19 =  PLANE_FINISH_MILLING($,$,'FINISH PLANAR FACE1',10.000,$,#39,#40,#41,$,#60,#61,
#42,2.500,$);
#39 = MILLING_CUTTING_TOOL('MILL 18MM',#29,(#125),80.000,$,$);
  #29 = TAPERED_ENDMILL(#30,4,.RIGHT.,.F.,$,$);
    #30 = MILLING_TOOL_DIMENSION(18.000, $,$, 29.0, 0.0, $,$);
  #125 = CUTTING_COMPONENT(80.000,$,$,$,$);
#40 = MILLING_TECHNOLOGY(0.040,.TCP.,$,-12.000,$,.F.,.F.,.F.,$);
#42 = BIDIRECTIONAL(0.05, .T., #43, .LEFT., $);
  #43 = DIRECTION('STRATEGY PLANAR FACE1: 1.DIRECTION',(0.000,1.000,0.000));
#60 = PLUNGE_TOOLAXIS($);
#61 = PLUNGE_TOOLAXIS($);
/*********************************************************/
#16 = PLANAR_FACE('PLANAR FACE1',#4,(#19),#77,#63,#24,#25,$,());
  #24 = LINEAR_PATH($,#54,#55);
    #54 = TOLERANCED_LENGTH_MEASURE(120.000,#56);
    #55 = DIRECTION('COURSE OF TRAVEL DIRECTION',(0.000,1.000,0.000));
  #25 = LINEAR_PROFILE($,#57);
    #57 = NUMERIC_PARAMETER('PROFILE LENGTH',100.000,'MM');
#63 = PLANE('PLANAR FACE1-DEPTH PLANE',#80);
  #80 = AXIS2_PLACEMENT_3D('PLANAR FACE1',#107,#108,#109);
    #107 = CARTESIAN_POINT('PLANAR FACE1:DEPTH ',(0.000,0.000,-5.000));
    #108 = DIRECTION(' AXIS ',(0.000,0.000,1.000));
    #109 = DIRECTION(' REF_DIRECTION',(1.000,0.000,0.000));
#77 = AXIS2_PLACEMENT_3D('PLANAR FACE1',#104,#105,#106);
  #104 = CARTESIAN_POINT('PLANAR FACE1:LOCATION ',(0.000,0.000,55.000));
  #105 = DIRECTION(' AXIS ',(0.000,0.000,1.000));
  #106 = DIRECTION(' REF_DIRECTION',(1.000,0.000,0.000));
/*****************************************************/
```

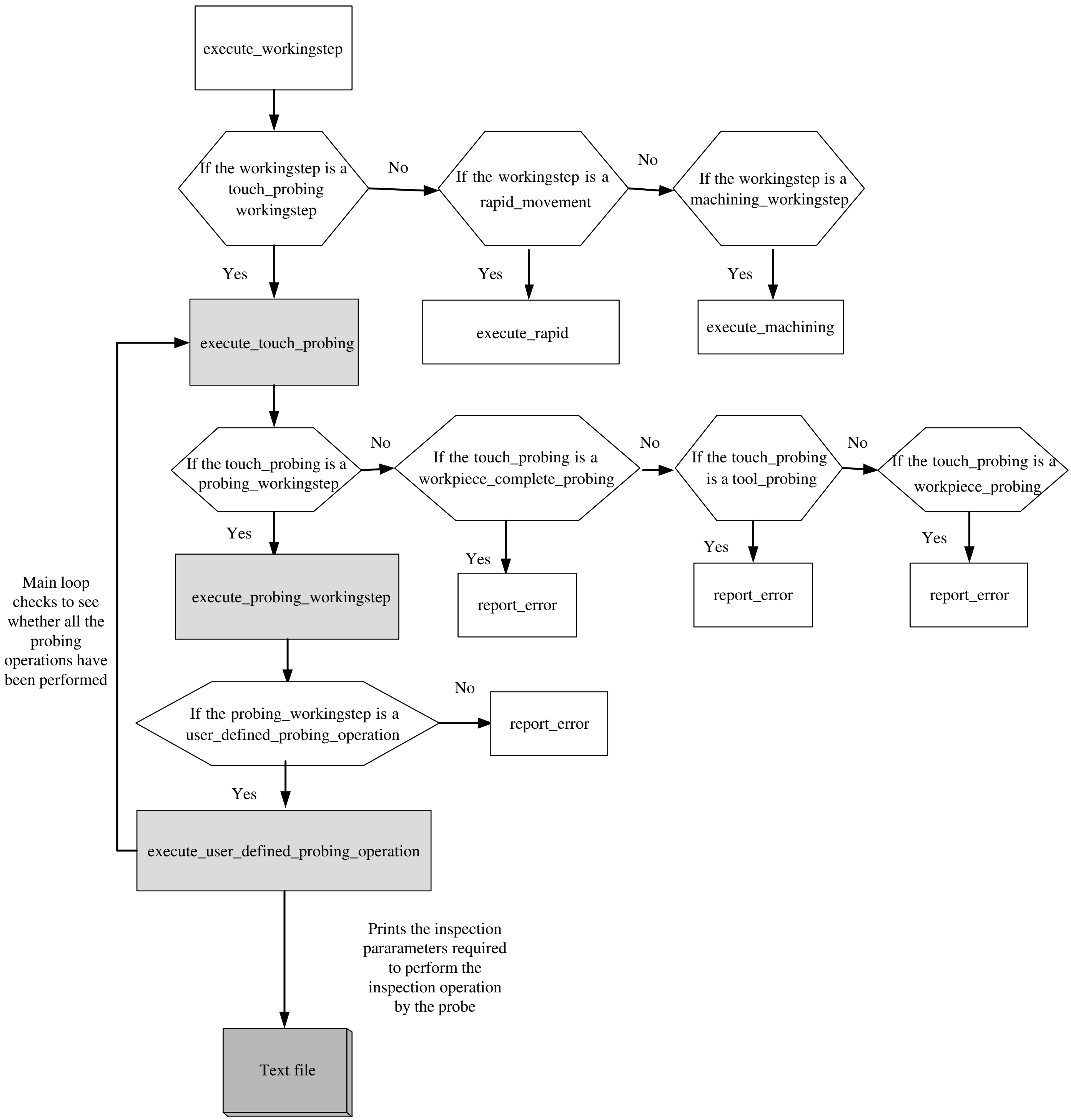

Fig. 5. Flowchart of the processes in function execute_workigstep.

Note that this data file follows ISO 10303-21 specifications to give a clear text encoding for instance data described in EXPRESS. Fig. 7 shows the structure of a Part 21.

The Header file includes the information about file description, file name, authors, complimentary level, and file schema. The Data section contains all the data of a particular application. Each line is one instance. For example:

```
#107 = CARTESIAN_POINT('PLANAR FACE1:DEPTH ',(0.000,0.000,-5.000));
```

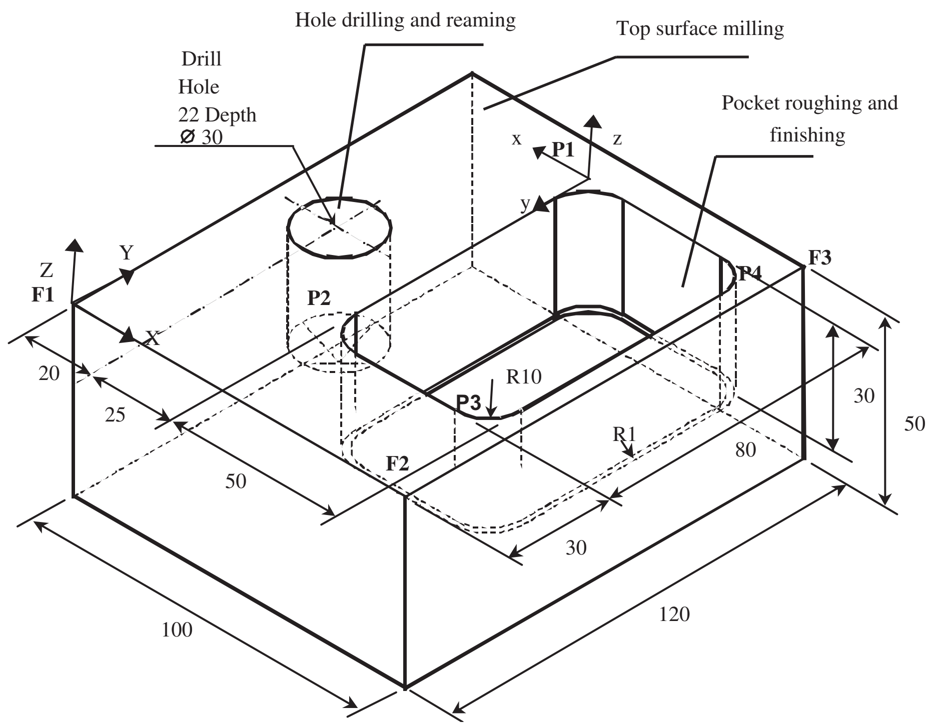

Fig. 6. Example workpiece.

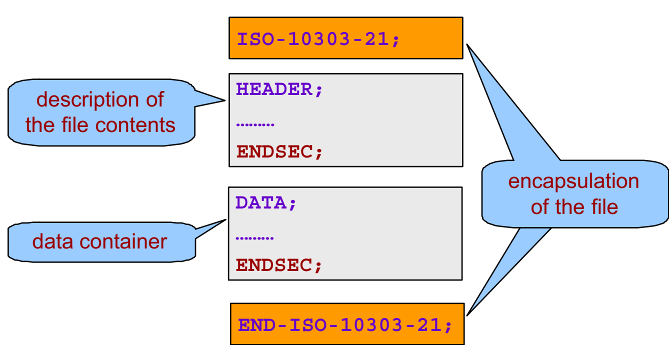

Fig. 7. Structure of a Part 21 file.

#107 is a unique unsigned integer indicating the line number. It may contain maximum 9 digits. CARTESIAN_POINT is the Entity name defined in the EXPRESS schema. The contexts in the bracket are the attribute values, which can be of different types.

In the above code, the Workingstep defines a face (finish) milling operation (#19). The milling cutting tool is defined in lines #39, #29, #30 and #125. The strategy and technology for this milling operation are defined in lines #40, #42 and #43. The approach strategy and the retract strategy are defined in Lines #60 and #61. The section from line #16 to #106 defines the plane feature information of this milling operation.

Four inspection Workingsteps are added to the file, each for a different purpose.

(1) Inspection Workingstep for setup. This Workingstep is to provide necessary information for set-up, e.g. the size and shape of the workpiece.
(2) Inspection Workingstep carried out after the first machining Workingstep (top surface milling). This Workingstep is to check out the dimensional tolerance of the top surface.
(3) Inspection Workingstep checking the true position of the bottom of the pocket. This is done after the pocket rough milling Workingstep. The purpose is to obtain the accurate finishing allowances for the finishing operation.

(4) Inspection Workingstep to check the dimensions of the finished pocket. The inspected items include depth, width, and the length of the pocket.

The following code shows the first inspection Workingstep inserted after the first milling Workingstep.

```
#200 = PROBING_WORKINGSTEP('SETUP INSPECTION',$,$,#62,$,#201,(#202,#203,#204));
#201 = USER_DEFINED_PROBING_OPERATION('SETUP PROBING',#205,$,#206,'SETUP PROBING');
  #205 = REFERENCE_DATUM_SETUP(#71);
  #206 = TOUCH_PROBE('NUM 01');
#202 = TOLERANCED_DIMENSION_ITEM('WORKPIECE DEPTH',#207,'workpiece depth',#210,$,$);
#210 = TOLERANCED_LENGTH_MEASURE(50.000,#56);
#203 = TOLERANCED_DIMENSION_ITEM('WORKPIECE WIDTH',#208,'workpiece length',#211,$,$);
#211 = TOLERANCED_LENGTH_MEASURE(120.000,#56);
#204 = TOLERANCED_DIMENSION_ITEM('WORKPIECE LEGNTH',#209,'workpiece width',#212,$,$);
#212 = TOLERANCED_LENGTH_MEASURE(100.000,#56);
#207 = USER_DEFINED_INSPECTION_RESULT (0.000,'INSPECTION RESULT OF SETUP PROBING',
(#200),'workpiece depth');
#208 = USER_DEFINED_INSPECTION_RESULT (0.000,'INSPECTION RESULT OF SETUP PROBING',
(#200),'workpiece length');
#209 = USER_DEFINED_INSPECTION_RESULT (0.000,'INSPECTION RESULT OF SETUP PROBING',
(#200),'workpiece width');
```

Line #200 defines the probing_workingstep for setup. Lines #205 and #206 define the reference datum and the probing tool of this probing operation. Lines #202, #203 and #204 define the features to be inspected. They are the depth, width and length of the workpiece respectively. Lines #207–#209 describe and record the inspection results.

Due to the insertion of the inspection Workingsteps, the top level Workingstep list is changed. This is shown in Fig. 7 by the thicker arrows. Entity inspection_item contains the toleranced item and provides a container (entity inspection_result) for storing the inspection result and the circumstances for the inspection activity. The measured results are kept in the container and updated when further inspections are carried out. The circumstances entity as shown in Fig. 2 is an inspection operation. It refers to the probing_workingstep entity. In this way, inspection result is recorded after each of the inspection Workingsteps and analyzed by data analysis software. These analyzed results are fed back to certain machining Workingsteps for them to be updated. As shown in Fig. 8, the information of the machined workpiece can be fed back to achieve the closed-loop.

The modified Part 21 file was manually constructed and checked against the accompanying schemas using Express Engine. After starting up the interpreter, the Part 21 file is first read into the memory and its semantics are checked. Then, the Part 21 file is stored in the secondary storage in a binary format which is easier for the computer to process. The interpreter reads in this binary data file and processes it to generate the CMCs. At the current stage of this research, only an intermediate file (Appendix A) is generated. This file shows all the inspection information that has been extracted from the Part21 file. The case study demonstrated the correctness of the proposed data model through its output file. Further research will utilize the information present in this output to undertake inspection planning as well as machining process planning.

6. Conclusions

The advantages of on-machine inspection are obvious. It has the potential to reduce part rejects, part cost and inspection costs. It also enables real-time, closed-loop control. However, the existing data model for the machining process (G-code) is an obstacle to achieving these advantages. With the new CNC data model, ISO 14649, inspection tasks can be performed on several inspection systems and the resulting measurement data can be consolidated, which allows a unified data management and company-wide evaluation of product data. In the proposed framework for STEP-NC enabled closed-loop machining process, no data conversions are needed; hence data loss is minimized, both production time and cost are reduced. With STEP-NC data, geometry, e.g. machining features, is reused to define inspection tasks. The results of the process-integrated inspection tasks are used for adjustment of positions as well as for data acquisition to decide optimal machining operations or for quality assurance. The results are finally used for planning subsequent manufacturing operations, for example to compensate system errors. Furthermore, based on the integrated STEP data, changes made to the shape of the workpiece for example can automatically be taken into account for the provision of a new measuring strategy and/or procedure.

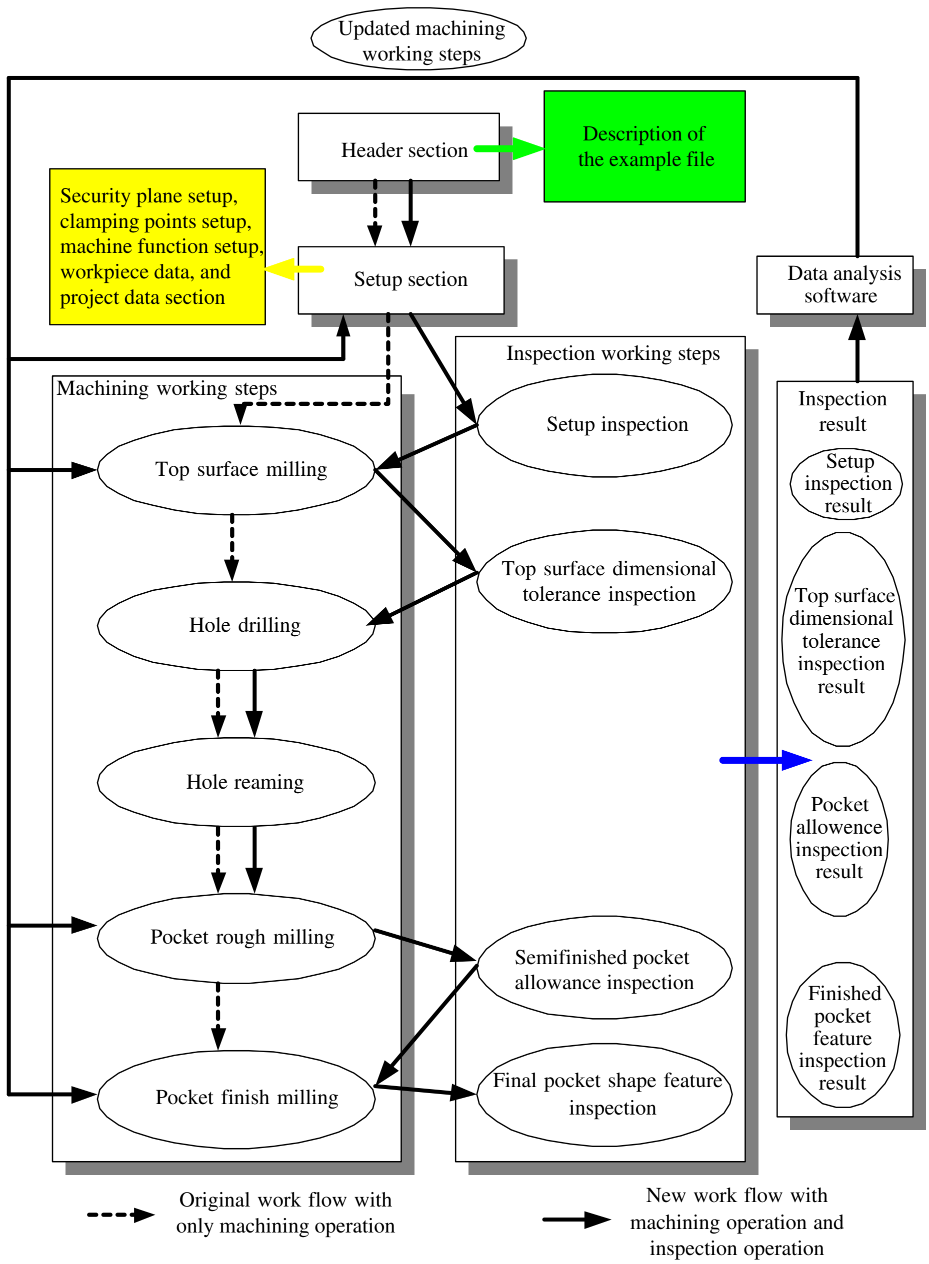

Fig. 8. Comparison of the original and the modified example file.

The proposed framework also defines a way of relating measurement results with machining features and operations. This is made possible through use of a STEP-NC data model with which design data can be embedded and measurement can be carried out in a more informed and controlled manner. This is one of the fundamental differences between the traditional so-called in-cycle gauging and the system proposed herein. Once analyzed and post-processed, inspection result can be fed back to the machining processes in real time to update the machining parameters. This could support a real-time closed-loop machining process to be integrated with on-line inspections. This combined schema developed in this research can be used for applications other than the example in the case study. For example, it also caters for turning operations. The extended Part 21 file with inspection operation inserted can also be used for applications, e.g. a genuine STEP-NC compliant inspection system.

It has to be pointed out that this paper only proposes a data model for on-line inspection. The follow-on research work includes inspection process planning and inspection data analysis. The former involves similar tasks as machining process planning except that the cutters are being replaced by the probes and the cutter path is replaced by the inspection path. The latter involves employment of control theories for processing the measured data so as to give useful instructions to the following machining operations as to how to correct the errors if any.

Appendix A. : The intermediate output of STEP-NC Interpreter

```
workplan:       SETUP INSPECTION
                FIRST MACHINING WORKINGSTEP-TOP SURFACE MILLING
                TOP SURFACE INSPECTION
                SECOND MACHINING WORKINGSTEP-HOLE DRILLING
                THIRD MACHINING WORKINGSTEP-HOLE REAMING
                FOURTH MACHINING WORKINGSTEP-POCKET ROUGHING
                SEMIFINISHED POCKET INSPECTION
                FIFTH MACHINING WORKINGSTEP-POCKET FINISH MILLING
                FINISHED POCKET INSPECTION

  // The main workplan of the whole machining: process. It is consist of five machining-
workings teps and four inspection workingsteps. The order of these workingsteps
represents the order of the machining and inspection order. //
  ----------------------Inspection operation starts----------------
  The ID of the probing_workingstep: SETUP INSPECTION
    probing_operation ID: SETUP PROBING
    probing_tool ID: NUM 01
    user_defined_probing_operation DESCRIPTION: SETUP PROBING OF THE RAW WORKPIECE
    toleranced_dimension_item ID: WORKPIECE DEPTH
    toleranced_dimension_item DESCRIPTION: THE DEPTH OF THE RAW WORKPIECE
    the dimension of the item needs to be inspected: (50.000)
    tolerance value of this dimens ion:(0.30, 0.30, 3)
    user_defined_inspection result DESCRIPTION: THE REAL DEPTH OF THE RAW WORKPIECE
  //Setup inspection of the workpiece depth//
    toleranced_dimension_item ID: WORKPIECE LENGTH
    toleranced_dimension_item DESCRIPTION: THE LENGTH OF THE RAW WORKPIECE
    the dimension of the item needs to be inspected: (120.000)
    tolerance value of this dimens ion:(0.30, 0.30, 3)
    user_defined_inspection_result DESCRIPTION is: THE REAL LENGTH OF THE RAW WORKPIECE
  //Setup inspection of the workpiece length//
    toleranced_dimension_item_ID: WORKPIECE WIDTH
    toleranced_dimension_it em DESCRIPTION: THE WIDTH OF THE RAW WORKPIECE
    the dimension of the item needs to be inspected: (100.000)
    tolerance value of this dimens ion:(0.30, 0.30, 3)
    user defined inspection result DESCRIPTION: THE REAL WIDTH OF THE RAW WORKPIECE
  //Setup inspection of the workpiece width//
  --------------------Machining workinstep ID--------------------------

  The machining workingstep ID: FIRST MACHINING WORKINGSTEP-TOP SURFACE MILLING
  // Machining workingstep (top surface milling) starts from here. Interpreter generates
CMCs//
  ----------------------Inspection operation starts---------------------
  The ID of the probing_workingstep: TOP SURFACE INSPECTION
    Probing_operation ID: TOP SURFACE PROBING
    Probing_tool ID: NUM 01
    user_defined_probing_operation DESCRIPTION: TOP SURFACE PROBING AFTER FIRST MACHIN-
ING WORKINGSTEP
    toleranced_dimension_item ID: WORKPIECE DEPTH
    toleranced_dimension_item DESCRIPTION: THE REAL DEPTH OF THE WORKPIECE AFTER THE
FIRST MACHINING WORKINGSTEP
    the dimension of the item needs to be inspected: (50.000)
    tolerance value of this dimension:(0.05,0.03,3)
    user defined inspection result DESCRIPTION: THE REAL DIMENSION OF THE TOP SURFACE AFTER
MACHINING
```

```
  // Inspection of top surface after the first machining workingstep//
  --------------------Machining workinstep ID--------------------------

  The machining workingstep ID: SECOND MACHINING WORKINGSTEP-HOLE DRILLING
  // Machining workingstep (hole drilling) starts from here . Interpreter generates
CMCs//

  ---------------------------Machining workinstep ID-------------------
  The machining workingstep ID: THIRD MACHINING WORKINGSTEP-HOLE REAMING
  // Machining workingstep (hole reaming) starts from here . Interpreter gen erates CMCs//
  -----------------------Machining workinstep ID-----------------------

  The machining workingstep ID: FOURTH MACHINING WORKINGSTEP-POCKET ROUGHING
  -----------------------Inspection operation starts--------------------

  The ID of the probing_workingstep: SEMIFINISHED POCKET INSPECTION
    probing_operation ID: SEMIFINISHED POCKET BOTTOM PROBING
    probing_tool ID: NUM 01
    user defined probing operation DESCRIPTION: SEMIFINISHED POCKET BOTTOM PLANE PROBING
AFTER THE FOURTH MACHINIG WORKINGSTEP
    toleranced_dimension_item ID: POCKET DEPTH
    toleranced_dimension_item DESCRIPTION: THE REAL POCKET DEPTH AFTER THE FOURTH
MACHINING WORKINGSTEP
    the dimension of the item needs to be inspected: (29.500)
    tolerance value of this dimension:(0.30, 0.30, 3)
    user_defined_inspection_result DESCRIPTION: THE REAL DIMENSION OF THE POCKET DEPTH
AFTER MACHINING
  // Inspection of the semifinished pocket after the fourth machining workings tep//
  -----------------------Machining workinstep ID-----------------------
  The machining workingstep ID: FIFTH MACHINING WORKINGSTEP-POCKET FINISHE MILLING
  // Machining workingstep (pocket finish milling) starts from here. Interpreter generates
CMCs//

  -----------------------Inspection operation starts--------------------
  The ID of the probing_workingstep: FINISHED POCKET INSPECTION
    probing_operation ID: FINISHED POCKET PROBING
    probing_tool ID: NUM 01
    user_defined_probing_operation DESCRIPTION: FINISHED POCKET FEATURE PROBING AFTER
ALL MACHINING WORKINGSTEPS
    toleranced_dimension)_item ID: POCKET WIDTH
    toleranced_dimension_item DESCRIPTION: THE REAL POCKET WIDTH AFTER MACHINING
    the dimension of the item needs to be inspected: (50.000)
    tolerance value of this dimens ion:(0.05, 0.03, 3)
    user_defined_inspection_result DESCRIPTION: THE REAL WIDTH OF THE FINISHED POCKET
AFTER MACHINING
  // Inspection of the finished pocket width//
    toleranced_dimension_item ID: POCKET LENGTH
    toleranced_dimension_item DESCRI PTION : THE REAL POCKET LENGTH AFTER MACHINING
    the dimension of the item needs to be inspected: (80.000)
    tolerance value of this dimens ion:(0.05,0.03,3)
    toleranced_dimension_item DESCRIPTION is: THE REAL POCKET WIDTH AFTER MACHINING
    the dimension of the item needs to be inspected is: (50.0000)
    tolerance value of this dimension is: (0. 05, 0.03, 3)
    user_defined_inspection_result DESCRIPTION is:THE REAL WIDTH OF THE FINISHED POCKET
AFTER MACHINING
    toleranced_dimension_item ID is: POCKET LENGTH
    toleranced_dimension_item DESCRIPTION is:THE REAL POCKET LENGTH AFTER MACHINING
```

```
    the dimension of the item needs to be inspected is: (80.0000)
    tolerance value of this dimension is:(0.05, 0.03, 3)
    user_defined_inspection_result DESCRIPTION is: THE REAL LENGTH OF THE FINISHED POCKET
AFTER MACHINING
    toleranced_dimension_item ID is: POCKET DEPTH
    toleranced_dimension_item DESCRIPTION is:THE REAL POCKET DEPTH AFTER MACHINING
    the dimension of the item needs to be inspected is: (30.0000)
    tolerance value of this dimension is:(0. 05, 0.03, 3)
    user_defined_inspection_result DESCRIPTION is:THE REAL DEPTH OF THE FINISHED POCKET
AFTER MACHINING
  -----------------------End-----------------------
```

References

[1] Chongstitvatana P. Overview of Computer-integrated-manufacturing (CIM). Lecture Notes. Thailand: Chulalongkorn University, Bangkok; 2005.
[2] Kennedy A, Ford DG. From manual grinding to CNC automation—a major step forward for the rod mill. Laser Metrol Machine Performance 2003;6:579–88.
[3] Cheraghi SH, Chen X, Twomey JM, Arupathi R. A closed-loop process analysis and control system for machining parts. Int J Prod Res 1999;37(6):1353–68.
[4] Jesse C. Process controlled manufacturing at Pratt & Whitney. Seattle, WA, USA: Aerospace manufacturing technology conference and exposition; 2001.
[5] ANSI/CAM-I 104.0-2001, Part 1. DMIS 4.0 Standard, Part 1, 2001.
[6] Genest DH, Improving measurement system compatibility. Quality Digest Mag 2001. 〈http://www.qualitydigest.com/april01/html/measurement.html〉
[7] Legge DI. Off-line programming of coordinate measuring machines. Luleå University of Technology, ISSN 0280-8242/HLU-TH-L-1996/19-L—SE/NR. 1996:19.
[8] International Organization of Standard, ISO 6983/1: Numerical control of machines—program format and definition of address words—part 1: data format for positioning, line and contouring control systems, 1st ed. 1982.
[9] Xu XW, He Q. Striving for a total integration of CAD, CAPP, CAM and CNC. Robot Compu-Integrated Manuf 2000;20:101–9.
[10] Kwon Y, Tseng T-L, Ertekin Y. Characterization of closed-loop measurement accuracy in precision CNC milling. Robot Compu-Integrated Manuf 2006;22(4):288–96.
[11] International Organization for Standardization, ISO 14649-1. Data model for computerized numerical controllers: part 1 overview and fundamental principles. ISO TC 184/SC 1, 04 July 2002.
[12] International Organization for Standardization, ISO 14649-16. Data model for computerized numerical controllers—part 16: data for touch probing based inspection. ISO TC 184/SC 1, 12 April 2004.
[13] International Organization for Standardization, ISO 14649-121. Data model for computerized numerical controllers: part 121—Tools for turning. ISO TC 184/SC 1, February 28 February 2003
[14] International Organization for Standardization, ISO 14649-12. Data model for computerized numerical controllers: part 12: process data for turning. ISO TC 184/SC 1, 13 May 2003.
[15] International Organization for Standardization, ISO 14649-10. Data model for computerized numerical controllers: part 10—general process data. ISO TC 184/SC 1, 17 February 2004.
[16] International Organization for Standardization, ISO 14649-111:2004. Industrial automation systems and integration—physical device control—data model for computerized numerical controllers—part 111: tools for milling machines. ISO TC 184/SC 1, 17 February 2004.
[17] International Organization for Standardization, ISO 14649-11:2003. Industrial automation systems and integration—physical device control—data model for computerized numerical controllers—part 11: process data for milling ISO TC 184/SC 1, 23 May 2003.
[18] International Organization of Standard, 2004, ISO/WD10303-219 Industrial automation systems and integration—product data representation and exchange—part 219: application protocol: Dimensional inspection information exchange.
[19] Brecher C, Wolf J, Vitr M. Closed-loop CAPP/CAM/CNC process chain based on STEP and STEP-NC inspection tasks. Int J Comput Integrated Manuf 2006;19(6):570–80.
[20] ANSI/CAM-I 104.0-2001. Part 1, DMIS 4.0 Standard, Part 1, 2001, 〈http://www.cami.org/storelistitem.cfm?itemnumber=8〉
[21] DML specification committee. DML—Dimensional Markup Language Version 2.0, 2004, 〈http://www.dmlspec.org〉
[22] Hardwick M. STEP-NC probing demonstration. EASTEC 2005 exposition and conference, West Springfield, MA, USA, 24–26 May 2005. 〈http://www.isd.mel.nist.gov/projects/stepnc/〉
[23] Ali L, Newman ST, Petzing J. Development of a STEP-compliant inspection framework for discrete components. Proc Inst Mech Eng, Part B: J Eng Manuf 219;7:557–63.
[24] Open Source Technology Group. Express Engine, 2005, 〈http://sourceforge.net/project/showfiles.php?group_id=24297〉
[25] Kramer TR, Proctor F, Xu X, Michaloski JL. Run-time interpretation of STEP-NC: implementation and performance. Int Compu Integrated Manuf 2006;19(6):495–507.
[26] Sharke P. On-machine inspecting. USA: Mechanical Engineering, ASME; 2005. p.30.