

# EVALUACIÓN DEL DESEMPEÑO DE PROCESOS DE MANUFACTURA PROYECTANDO REDES DE PETRI HACIA REDES DE EVENTOS DISCRETOS

EDGAR CHACÓN\*, CARLOS PARRA\*\*

\**Laboratorio de Sistemas Discretos, Automatización e Integración (LASDAI)*  
*Av. Tulio Febres Cordero Mérida 5101 - Venezuela*  
*Facultad de Ingeniería, Universidad de los Andes*  
*Mérida, Venezuela*

\*\**Facultad de Ingenierías*  
*Universidad de Pamplona, Colombia*  
*Doctorado en Ciencias Aplicadas, Universidad de los Andes, Venezuela*

Emails: echacon@ula.ve, cparra@ula.ve

**Resumen**— Las Redes de Petri se utilizan para representar procesos industriales supervisados, y existen métodos formales para su validación. Por otra parte, los sistemas de eventos discretos se simulan por medio de redes de nodos interconectados que intercambian mensajes, y esta simulación proporciona información sobre el comportamiento y desempeño del sistema en estudio. Es conveniente para el estudio de un proceso industrial no sólo validar la lógica del proceso por medio de la Red de Petri, sino también evaluar su eficiencia en términos de la utilización de recursos y de otros indicadores. Este artículo propone un método para proyectar elementos de una Red de Petri en una red de componentes DEVS que intercambian mensajes, apoyado en un caso de estudio de un sistema de manufactura robotizado.

**Palavras-chave**— Redes de Petri, Simulación, Sistemas de Eventos Discretos.

## 1. Introducción

Este documento está ubicado dentro del estudio del Control Supervisorio de Sistemas Industriales, donde se plantea un método para evaluar el desempeño de los procesos de manufactura modelados con Redes de Petri por medio de la simulación de sistemas de eventos discretos. En detalle, utilizando una especificación común para modelar sistemas discretos, y proyectando la Red de Petri como una red de nodos, donde una o varias etapas del proceso industrial pueden verse como un conjunto de componentes que intercambian mensajes. Por medio de esta simulación, se podrían evaluar otros aspectos del sistema que no alcanzan a evaluarse utilizando solamente Redes de Petri. Uno de estos aspectos podría ser el tiempo libre que tiene un recurso sin utilizarse, otro podría ser el tiempo de permanencia en el sistema de un insumo, hasta que es transformado en producto. Las redes de Petri son una herramienta muy útil para validar el comportamiento de un modelo con respecto a la sincronización de sus actividades y al diseño de respuestas ante la ocurrencia de eventos, pero es difícil evaluar otros aspectos, como por ejemplo, cuando ocurrirá cada evento exactamente, o cual es la eficiencia del sistema en cuanto a utilización de los recursos.

El Formalismo DEVS (*Discrete-Event System Specification*) (Zeigler et al., 2000) es la especificación más general para describir sistemas discretos, abarcando Grafets, StateCharts, Redes de Petri y Redes de Eventos Discretos, entre otros. Si se tiene un proceso industrial modelado en Redes de Petri, se puede llevar a una representación a otros

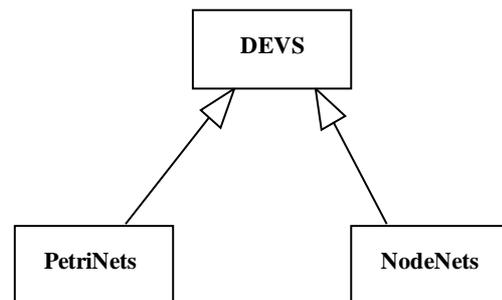


Figura 1: Jerarquía de clases utilizadas para representar procesos

sistemas de eventos discretos a partir de la especificación más general. Si se obtuviera una representación orientada a objetos de estos formalismos, tendrían la forma de la figura 1.

La simulación de sistemas de eventos discretos implementando una lista de eventos futuros (Law e Kelton, 2000) ofrece un complemento para medir la eficiencia del sistema en cuanto a la utilización de equipamiento, represamiento de insumos, probabilidad de ocurrencia de un evento en un tiempo dado, entre otros. Para explicar mejor la descripción anterior, es necesario apoyarse en un caso de estudio de un problema típico de manufactura.

### 1.1. Caso de estudio

El caso particular de producción que se quiere analizar consiste en una celda de producción cuyo

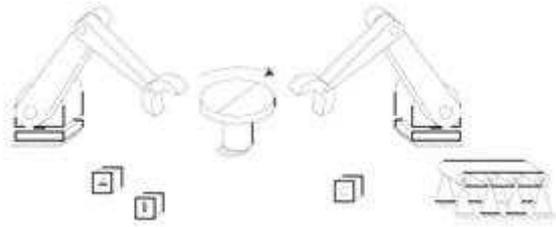


Figura 2: Esquema del proceso de fabricación

equipamiento lo conforman dos robots con instrumentos manipuladores, y una mesa giratoria. Los insumos son los elementos A, B y C. Una especificación del proceso de manufactura y sus restricciones se podría describir de la siguiente manera:

- El robot 1 toma un elemento A y lo coloca sobre su lado de la mesa, luego toma un elemento B, lo coloca sobre su lado de la mesa y ensambla las piezas, obteniendo el subproducto AB.
- La mesa gira para que el subproducto quede del lado del robot 2.
- El robot 2 toma un elemento C, lo coloca sobre su lado de la mesa y ensambla para obtener el producto terminado ABC.
- El robot 2 toma la pieza ABC ensamblada y la lleva a una cinta transportadora.
- Ambos robots sólo pueden tomar un elemento a la vez.
- El brazo 1 sólo coloca el elemento A, cuando su lado de la mesa está vacío, y sólo coloca el elemento B, luego de colocar el elemento A. La mesa solo gira cuando el robot 1 ensambló AB, y el lado del brazo 2 esta vacío. El brazo 2 sólo coloca el elemento C cuando la pieza AB está de su lado de la mesa.

Según varios autores, la automatización de sistemas de producción muestran dos aspectos muy importantes para optimizar y mejorar su producción: características físicas del sistema a automatizar, y el modo de operación del sistema automatizado (Chacón et al., 2004). La metodología que será utilizada para efectuar la evaluación del desempeño de este sistema se puede resumir de la siguiente manera:

1. Obtención de un modelo del comportamiento del proceso.
2. Traducción del modelo de comportamiento a una red de nodos que representan un sistema de eventos discretos.
3. Simulación numérica de la red de nodos como un sistema de colas.
4. Análisis de resultados.

## 2. MODELADO LÓGICO DEL PROCESO

Como en cualquier aplicación en control, en el proceso de automatización, se requiere de un modelo del sistema. Según la especificación descrita en la sección anterior, es evidente que el proceso presenta un carácter discreto. Para obtener el modelo de un proceso con esta característica es posible utilizar dos enfoques, o bien hacer una descripción por medio de autómatas o por medio de redes de Petri. Existen muchos argumentos para estudiar sistemas dinámicos utilizando redes de Petri (Murata, 1989), ya que pueden proporcionar información sobre la estructura y comportamiento del sistema, y puede utilizarse esta información para evaluar el sistema modelado y sugerir mejoras o cambios. Por las propiedades de funcionamiento de este proceso se ha escogido el método de las redes de Petri, debido a que permite determinar el estado del sistema de acuerdo a la posición de los mensajes (tokens) dentro de un grafo.

### 2.1. Red de Petri del proceso

Las redes de Petri son una herramienta gráfica que permite hacer modelado matemático, y son adecuadas en la descripción de sistemas concurrentes, bien sincrónicos o asincrónicos. Una red de Petri tiene cuatro elementos: un conjunto de lugares  $P$ , un conjunto de transiciones  $T$ , una función de entrada  $I$ , y una función de salida  $O$ , las cuales relacionan transiciones y lugares. Formalmente, se puede definir como una cuadrupla  $C$  (Peterson, 1981), siendo  $C = (P, T, I, O)$ .

Donde:

- $P = \{p_0, p_1, p_2, \dots, p_n\}$  conjunto de lugares.
- $T = \{t_0, t_1, p_2, \dots, t_m\}$  conjunto de transiciones.
- $I : T \rightarrow P^\infty$  es la función de entrada, que proyecta de transiciones a lugares.
- $O : P \rightarrow T^\infty$  un mapeo de transiciones a grupos de lugares.

La red de Petri que representa al proceso de producción que se está analizando se muestra en la figura 3. Cada elemento del proceso industrial está reflejado por un conjunto de lugares y transiciones de la red de Petri. Las interacciones entre ellos están representadas por los eventos que son habilitados por lugares y los que son activados por las transiciones. Incluso algunos lugares se encargan de inhibir la ocurrencia de ciertos eventos, como el supervisor impidiendo que la mesa gire cuando alguno de los robots esté ensamblando las piezas sobre ella.

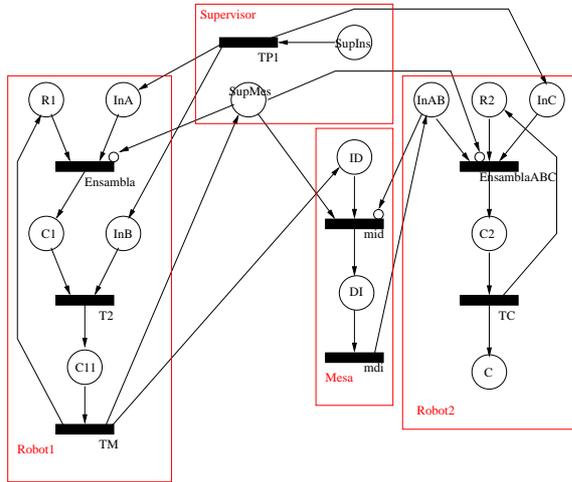


Figura 3: Red de Petri que representa al proceso de manufactura supervisado

## 2.2. El árbol de alcance

La evolución del proceso puede seguirse a través de los cambios que ocurren en los lugares cuando ocurre una transición. Un método habitual para hacer seguimiento a esta evolución es el del árbol de alcance, que consiste en representar todos los lugares de la red como un vector, asignándoles el valor de 0 o 1 si está presente o no un mensaje. En ciertas ocasiones, puede tener valores mayores, si en un lugar confluyen varios mensajes en la medida en que el sistema evoluciona. Los valores del vector cambian cada vez que ocurre una transición, ya que el mensaje se mueve de un lugar a otro, y así el sistema va cambiando de estado. Este método facilita un seguimiento sistemático y su implementación por medio de algún lenguaje de programación de alto nivel. Para el caso de estudio de los dos robots, el supervisor y la mesa, tenemos la siguiente estructura del vector de lugares, donde cada elemento toma el nombre del lugar representado:

$$[\text{SupIns} \text{SupMes} \text{R1} \text{InA} \text{C1} \text{InB} \text{C11} \text{ID} \text{DIR2} \text{InAB} \text{InC} \text{C2} \text{C}]$$

El estado inicial queda representado entonces por los siguientes valores:

$$[1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]^t$$

Que se interpretaría de la siguiente manera: el Supervisor va a producir una unidad de ABC, la mesa puede girar, el robot 1 está listo para trabajar, no hay insumo A, no hay insumo A en la mesa, no hay insumo B, no hay subproducto AB en la mesa, la mesa no ha girado, la mesa no ha retornado, el robot 2 está listo para trabajar, no insumo AB, no hay insumo C, los insumos y subproducto no están en la mesa, no hay producto ABC en la cinta transportadora. Cuando empiezan a ocurrir

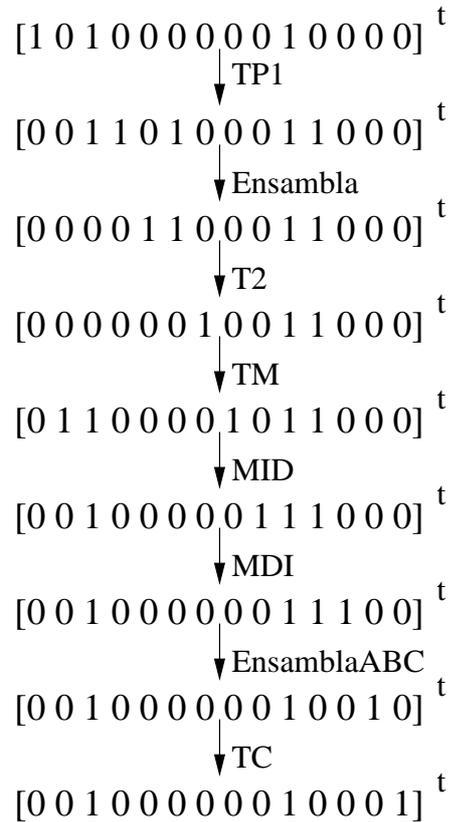


Figura 4: Árbol de alcance de la red de Petri que representa el proceso de ensamblaje de piezas.

los eventos, el sistema evoluciona de acuerdo al siguiente diagrama, que se muestra en la figura 5.

## 3. PROYECCIÓN DE LA RED DE PETRI A UNA RED DE EVENTOS DISCRETOS

Los modelos que representan a sistemas de eventos discretos dinámicos, estocásticos y discretos, donde las variables de estado cambian de valor en instantes no predecibles de tiempo. Estos instantes de tiempo corresponden a un evento, por lo tanto un evento se define como *una acción instantánea que puede cambiar el estado del sistema* (Guasch et al., 2005). Según investigadores de la simulación de eventos discretos (Banks et al., 2000), los elementos de interés de los modelos de eventos discretos son las *actividades*, que son tareas que se desarrollan en el sistema, delimitadas por dos eventos: uno que la inicia y uno que la termina. Otro tipo de elemento son las *Entidades*, que son los objetos que circulan por el sistema. Pueden ser recursos que utiliza el sistema (permanentes) u objetos que se procesan en el sistema (temporales). Estas entidades tienen *atributos* que son las propiedades de los recursos y de los objetos temporales del sistema.

### 3.1. Generalidades de las redes de nodos que procesan eventos discretos

Existen muchas formas de implementar un sistema basado en DEVS para simular fenómenos de diversa naturaleza. Un ejemplo de lenguaje de simulación basado en DEVS es Glider (CESIMO e IEAC, 1996), donde se representa un sistema de eventos discreto por medio de una red de nodos intercambiando mensajes. Un resumen de las características de los tipos de nodos predefinidos se muestra en la siguiente tabla 1.

Cuadro 1: Tipos de nodos que implementa el lenguaje de simulación Glider, basado en DEVS.

Nodo	Descripción
Input	Generan mensajes que entran al sistema para ser procesados, enviándolos a otros nodos.
Line	Reorganiza una cola de eventos, según la disciplina que se quiera implementar.
Gate	Detiene o permite que los mensajes circulen por la red.
Resource	Simula los recursos utilizados por los mensajes, pueden representar también etapas en un proceso.
Decisión	Se utiliza para dirigir mensajes entre nodos (enrutamiento).
Exit	Destruye los mensajes. Simula las entidades que abandonan el sistema.
Continuous	Resuelve sistemas de ecuaciones diferenciales de primer orden.
Autonomous	De propósito general, maneja variables globales, envía mensajes.

### 3.2. Proyección de la red de Petri hacia un sistema estocástico de eventos discretos.

Considerando que las redes de Petri son una herramienta útil para evaluar la lógica de procesos industriales concurrentes, y que las redes de colas permiten la evaluación del desempeño global del sistema, es necesario disponer de algún método que permita obtener tanto la lógica del proceso como la medida del desempeño. Por medio del formalismo DEVS se puede lograr la traducción de una forma de representación a otra, de una manera sistemática. Un método general puede tener la siguiente forma:

1. Tomar la representación formal de la Red de Petri. A partir de los lugares de la red, clasi-

ficar:

- Lugares que representen insumos físicos
  - Lugares que sean indicadores de recurso o máquina disponible
  - Etapas dentro del proceso
  - Lugares que correspondan a las acciones del supervisor
2. Llevar esa representación formal hacia la especificación DEVS general. Uno o más lugares de la Red de Petri pueden corresponder a uno o más componentes DEVS básicos, con varias entradas y varias salidas.
  3. De la especificación DEVS llevar a la representación de la red de nodos que procesan eventos discretos. Estos eventos discretos circulan a través de la red como si fueran mensajes que pasan de un componente a otro. Estos nodos pueden ser de tipo: entrada (origen de eventos), recurso del sistema, salida del sistema, y nodos que regulan el paso de mensajes. También se agrega un nodo especial que tendrá la función de coordinar las activaciones de los demás nodos y la sincronización de los mensajes que representan al material que es transformado para obtener un producto terminado.

La proyección de una Red de Petri a una Red de Nodos se puede resumir así:

- Lugar que representa a un insumo físico resulta en un nodo de tipo entrada (input).
- Equipamiento disponible resulta en un nodo con variables globales, que activa los eventos de los nodos de entrada.
- Las etapas del proceso resulta en un nodo de tipo recurso (resource).
- Los lugares que llevan a cabo la coordinación del proceso se proyectan hacia un nodo con variables globales, y uno o varios nodos que representan compuertas (gates) para regular el paso de mensajes.

### 3.3. El ejemplo del caso de estudio.

Para proceder a realizar su simulación, es necesario entender y analizar el sistema. El desarrollo de las definiciones propuestas por Banks (Banks et al., 2000) aparece a continuación, aplicándose al sistema objeto de estudio.

- Entidades: pueden ser los recursos con que cuenta el sistema para desarrollar su misión, como lo son los dos robots y la mesa giratoria; y el material que circula por el sistema, en este caso los insumos A, B y C, productos intermedios AB, y producto final ABC.

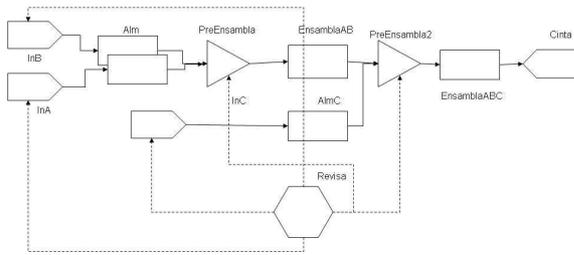


Figura 5: Red de Nodos para simular el sistema de los dos robots

- Atributos: son atributos el tiempo promedio que le toma ensamblar las piezas a los robots. Este tiempo se representa por medio de una variable aleatoria. Para los insumos y subproductos que están circulando, el número de orden es importante, ya que el insumo A1 solo puede ensamblarse con el insumo B1, para dar origen al subproducto AB1, el cual solo se puede ensamblar con el insumo C1.
- Actividades: toda acción que represente un período de tiempo es una actividad. El proceso de ensamblado de los insumos A y B, la colocación de los insumos A y B en la mesa, del lado del robot 1, la colocación de insumos C y subproducto AB en la mesa, del lado del robot 2 y el ensamble de los insumos C con AB son las actividades que ocurren en este sistema.
- Eventos: los eventos considerados son: cada vez que llega un insumo A o B a la mesa, previa orden del supervisor. El proceso de ensamble de A y B, el giro de la mesa para colocar AB en el lado del robot 2, el proceso de ensamble de AB con C, y la colocación del insumo C en la mesa.
- Variables de estado: la cantidad de producto terminado colocado en la cinta transportadora, el estado de los robots: desocupado, ensamblando. El estado de la mesa: puede girar o no puede girar. Estas variables se pueden proyectar hacia el vector que contiene los lugares de la red de Petri.

Una vez determinados los elementos del sistema, y basándose en la Red de Petri del proceso, se diseñó una red de nodos que representan a las entidades mencionadas previamente. Para obtener la red de nodos, se aplicaron los pasos que se describieron en la sección anterior, y se muestra el diagrama de la red en la figura 5.

- InA: nodo Input, representa al insumo A cuando el robot 1 lo puede tomar.
- InB: igual que el anterior, para el insumo B.

- Alm: es un nodo de tipo Resource, representa la puesta en la mesa de los insumos A y/o B, del lado más cercano al robot1.
- PreEnsambla: nodo de tipo Gate, simula la sincronización de los insumos A y B para su ensamblado.
- EnsamblaAB: nodo de tipo Resource, simula el proceso de ensamblado de los insumos A y B para obtener el producto intermedio AB.
- InC: nodo Input, representa al insumo C, cuando el robot 2 lo puede tomar.
- AlmC: es un nodo de tipo Resource, y representa la puesta en la mesa del insumo C, más cercano al robot 2.
- PreEnsambla2: nodo de tipo Gate, simula la sincronización de los insumos AB y C para su ensamblado.
- EnsamblaABC: nodo de tipo Resource, simula el proceso de ensamblado de los insumos AB y C para obtener el producto final ABC.
- Cinta: nodo de tipo Exit, los mensajes que llegan a este nodo representan la puesta del producto ABC en la cinta transportadora para su despacho.
- Revisa: es un nodo de tipo Autónomo, que en este caso representa a un objeto que está chequeando el proceso y permitiendo/inhabilitando el paso de insumos, a manera de un supervisor.

#### 4. SIMULACIÓN COMO UN SISTEMA DE EVENTOS DISCRETOS

El diagrama de la red de nodos expuesto en la figura anterior se codificó con el lenguaje Glider, desarrollado en la Universidad de los Andes, el cual fue desarrollado para simular sistemas de eventos discretos y continuos, como una agrupación de subsistemas que intercambian información de diferentes maneras, frecuentemente como paso de mensajes entre nodos. Las siguientes condiciones iniciales y parámetros del sistema se mencionan a continuación. El modelo se programó para solicitarle al usuario cuantas unidades de los insumos A, B y C estaban disponibles para el proceso de manufactura. De la simulación se obtuvieron una traza y tres resúmenes de resultados.

##### 4.1. Análisis de la traza de eventos

Se realizó una traza corta, donde se manufacturaron un total de seis piezas. La siguiente tabla muestra un fragmento de la misma, y luego se resumen todos los eventos que ocurrieron, ordenados por tiempo. Los eventos que ocurrieron durante la simulación se resumen en la tabla 2

Cuadro 2: Eventos ocurridos durante la simulación.

Tiempo	Evento
0	Activación de paso de insumos A y B
1	Insumo A habilitado
1	Insumo B habilitado
1.997	El robot1 toma el insumo A
2.166	El robot1 toma el insumo B
3.845	Se ensambló y se obtuvo AB y la mesa giró, el robot 2 tomó AB
5	Insumo C habilitado
6.056	El robot2 toma el insumo C
8.255	Se ensambló y se obtuvo ABC, el robot2 colocó en la cinta a AB.
...	...
49.142	Se ensambló y se obtuvo ABC, el robot2 colocó en la cinta a AB.
50	Revisión del sistema, se termina la simulación porque no hay insumos a procesar

#### 4.2. Análisis de resultados

Para las posteriores simulaciones, se hicieron ocho replicaciones, con una producción de 25 unidades ensambladas. Revisando los resultados replicaciones, se observa que producir 25 unidades toma entre 208 y 219 unidades de tiempo, no se presentan colas en ninguno de los recursos, y esto se debe a que el nodo que revisa (el supervisor) no habilita paso de nuevos insumos hasta que la mesa está disponible. El robot 1 permanece libre unas 177 unidades de tiempo como promedio, y esto se debe a que debe esperar a que los insumos le lleguen, a tenerlos a ambos en la mesa, y luego esperar que el robot 2 complete su tarea. Por su parte, el robot 2 permanece libre unas 163 unidades de tiempo como promedio, por la misma razón. A un insumo le toma como promedio unas 6.2 unidades de tiempo ser procesado y a partir de el obtenerse un producto intermedio o un producto final.

## 5. Conclusiones

Recopilando las experiencias adquiridas durante el desarrollo de esta aplicación y su posterior documentación, se presentan las siguientes conclusiones:

- Las redes de Petri como herramienta de modelado matemático permite establecer de una manera relativamente sencilla la supervisión de procesos de manufactura determinados por la ocurrencia de eventos discretos. Aunque requiere de complementarse de otros métodos de modelado para evaluar el comportamiento de los eventos discretos en el tiempo.

po.

- La simulación estadística para medir la eficiencia de un proceso industrial requiere de la utilización de modelos basados en redes de nodos, y estos derivan del formalismo DEVS. Dado que las redes de Petri son un caso particular de DEVS, se puede establecer un mapeo entre las redes de Petri y las redes de nodos
- El complemento de redes de Petri con simulación de eventos discretos permite diseñar y analizar de una manera mas completa sistemas de manufactura integrados, de manera que luego pueda facilitarse su implantación en una planta industrial, conociendo de antemano el comportamiento que va a presentar.

## Referencias

- Banks, J., Carson, J., Nelson, B. e Nicol, D. (2000). *Discrete-Event System Simulation*, third edn, Prentice-Hall.
- CESIMO e IEAC (1996). *GLIDER Reference Manual*, Universidad de los Andes.
- Chacón, E., Besembel, I. e DeSarrazin, G. (2004). *Automatización Industrial*, Universidad de los Andes.
- Guasch, A., Piera, M., Casanovas, J. e Figueras, J. (2005). *Modelado y Simulación. Aplicación a procesos logísticos de fabricación y servicios*, Alfaomega.
- Law, A. e Kelton, D. (2000). *Simulation Modeling and Analysis.*, third edn, McGraw-Hill.
- Murata, T. (1989). *Petri Nets, Properties, Analysis and Applications*, Proceedings of the IEEE, Vol 77 No. 4.
- Peterson, J. (1981). *Petri Net theory and the modeling of systems*, Prentice-Hall.
- Zeigler, B., Praehofer, H. e Kim, T. G. (2000). *Theory of Modeling and Simulation*, second edn, Academic Press.