

**DISCRETE EVENTS SIMULATION OF INDUSTRIAL PROCESSES
DESCRIBED WITH PETRI NETWORKS****SIMULACION A EVENTOS DISCRETOS DE PROCESOS INDUSTRIALES
DESCRITOS CON REDES DE PETRI**

MSc. Hugo Fernando Castro Silva*, MSc. Carlos Arturo Parra Ortega**

Universidad de Pamplona

* Dpto. de Ingenierías Mecánica, Mecatrónica e Industrial

** Dpto. de Ingeniería Electrónica, Eléctrica, Sistemas y Telecomunicaciones

Ciudadela Universitaria. Pamplona, Norte de Santander, Colombia.

Tel.: 57-7-5685303, Fax: 57-7-5685303, Ext. 164.

E-mail: {hcastro, carapa}@unipamplona.edu.co

Abstract: A fundamental aspect of production systems is the measuring of performance. In order to obtain such measure, mathematical models are used, but when a production system is growing in complexity, both in number of products and used material, these models do not exist or it is difficult to derive it. For this reason production systems simulation is used. In this document a push-type production system is simulated, by using a discrete-event approach such as queues network, and real data provided by a pharmaceutical enterprise.

Resumen: Un aspecto fundamental de los sistemas de producción es la medida de su desempeño. Para obtener una medida, se recurre frecuentemente a modelos matemáticos, pero a medida que un sistema de producción es de cierta complejidad, tanto en la cantidad de productos como en los insumos que utiliza, estos modelos no existen o son difíciles de deducir. Por tal motivo se recurre a la simulación de sistemas de producción. En este documento se simula un sistema de producción de tipo *push*, utilizando un enfoque de eventos discretos como lo son las redes de colas, y a partir de datos reales proporcionados por una empresa del área farmacéutica.

Keywords: Push-Production Systems, Discrete-event Systems, Queues networks, Simulation.

1. INTRODUCCION

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 colas, donde una o varias etapas del proceso industrial pueden verse como un conjunto de recursos. Por medio de esta simulación, se podrían evaluar otros aspectos del sistema que no alcanzan a evaluarse utilizando Redes de Petri solamente. 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, y así sucesivamente.

El Formalismo DEVS (Zeigler, Praehofer y Kim, 2000) es la especificación más general para describir sistemas discretos, abarcando los Graficets, StateCharts, Redes de Petri y Redes de Colas, entre otros. Si se tiene un proceso industrial modelado en Redes de Petri, se puede llevar a una representación a otros sistemas de eventos discretos a partir de la especificación más general. Si se obtuviera una representación de objetos de estos formalismos, tendrían la siguiente forma:

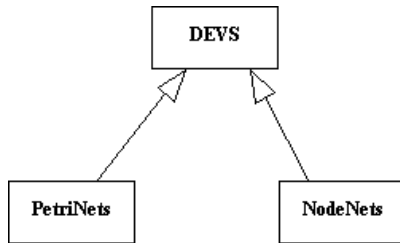


Fig. 1: Diagrama de clases para simular procesos

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. La simulación de sistemas de eventos discretos implementando una lista de eventos futuros (Law & 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 un sistema de producción conformado por dos máquinas M1 y M2, dos cintas transportadoras con una bandeja en cada una de ellas C1 y C2. Las piezas a ser procesadas son de tipo A y B. La secuencia de operaciones debe mantener las siguientes restricciones:

1. La secuencia de operaciones para una pieza de tipo A se inicia con su procesamiento en la máquina M1, una vez finaliza la operación se emplea la bandeja C1 para transportar la pieza

a la máquina M2, donde una vez procesada sale del sistema.

2. La secuencia de operaciones para la pieza de tipo B es la misma que para la pieza tipo A, solo que se utiliza la bandeja C2.
3. Las máquinas no pueden realizar consecutivamente dos operaciones sobre el mismo tipo de pieza, por lo tanto deben procesarse alternadamente.
4. Para que la máquina M1 quede libre una vez finalizado el proceso de una pieza tipo A es necesario que la bandeja C1 se encuentre al lado de la máquina M1 (posición superior). Para que esta misma máquina se libere luego de procesar una pieza de tipo B es necesario que la bandeja C2 se encuentre en la posición superior.

Un esquema simplificado que muestra el anterior proceso se muestra en la figura 2, se considera simplificado porque no se muestran detalles sobre como se proporcionarán las piezas a la máquina, o el mecanismo de coordinación entre máquina y la cinta. Obviamente existe una red de comunicación de campo, para transmitir señales y recibir comandos de los controladores.

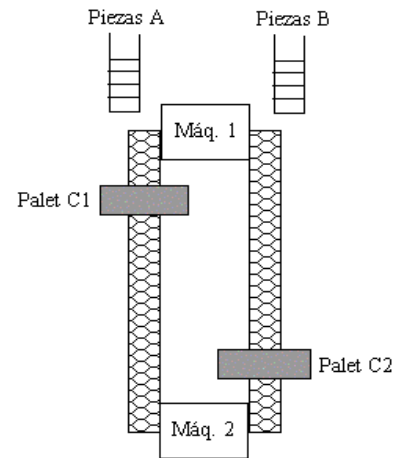


Fig. 2: Esquema del proceso de fabricación

La principal característica del sistema es que los recursos son compartidos (las dos máquinas), y bajo ciertas condiciones en la secuencia de producción, pueden operar simultáneamente.

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 de la alternativa de automatización se puede resumir de la siguiente manera:

- Aplicación de una arquitectura de referencia para automatización.
- Obtención de un modelo del comportamiento del proceso.
- Simulación numérica de este modelo como un sistema de eventos discretos.

2. MODELADO LOGICO DEL PROCESO

Como en cualquier aplicación en control, en el proceso de automatización, se requiere de un modelo del sistema. Según la descripción del funcionamiento y las restricciones dada 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, y su desempeño puede simularse como un conjunto de objetos intercambiando mensajes (Fishwick, 1994).

2.1 Red de Petri del proceso industrial.

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, 1984), siendo $C = (P, T, I, O)$.

Donde:

- $P = \{p_0, p_1, p_2, \dots, p_n\}$ conjunto de lugares, $n > 0$.

- $T = \{t_0, t_1, t_2, \dots, t_m\}$ conjunto de transiciones, $m > 0$.
- $I: T \rightarrow P^\infty$ es la función de entrada, que proyecta de transiciones a lugares.
- $O: P \rightarrow T^\infty$ un mapeo de transiciones a grupos de lugares.

Vale la pena destacar que existen otras definiciones formales equivalentes a la mencionada anteriormente.

La descripción de eventos discretos por redes de petri se fundamenta en los conceptos de eventos y condiciones. Los eventos son asociaciones que ocurren en el sistema; y las condiciones son descripciones lógicas del estado de varias partes del sistema.

Para que los eventos ocurran deben existir ciertas condiciones, llamadas precondiciones. Cuando ocurre un evento estas precondiciones generalmente cambian para dar lugar a otro conjunto de condiciones, llamadas post-condiciones. Así, las post-condiciones de un evento pueden ser precondiciones de otro evento.

Para construir una red de petri el modelador debe estimar el dominio de interpretación del sistema físico etiquetando lugares y transiciones. Así, por medio de identificar los eventos y las condiciones del sistema, se puede modelar un sistema a eventos discretos de la siguiente manera:

1. Cada evento se representa como una transición.
2. Los lugares representan las condiciones.
3. Las precondiciones de un evento son los lugares de entrada a la transición de tal evento.
4. Las post-condiciones de un evento son los lugares de salida de la transición que representa tal evento.

La red de petri que representa al proceso de producción que se está analizando se muestra en la figura 4. 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.

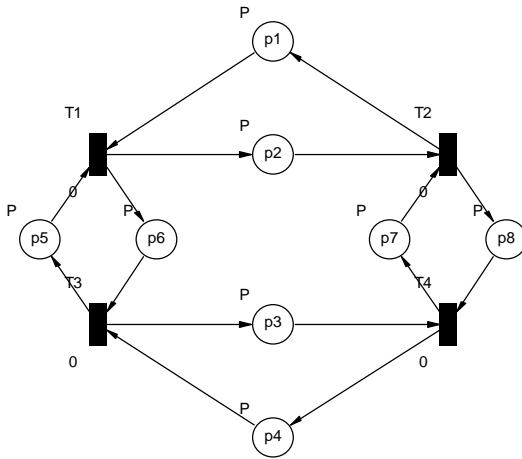


Fig. 3: Red de Petri del proceso industrial

2.2 Lugares y transiciones asociados al proceso

Los lugares indican el estado en que se encuentran los componentes físicos del sistema de producción, y las transiciones indican la ocurrencia de eventos que modifican un estado, o tareas que se deben efectuar para alcanzar un estado determinado.

La siguiente tabla describe los lugares de la red, que están asociados a condiciones de operación de las máquinas y recursos físicos del sistema productivo.

Tabla 1. Lugares del sistema de producción.

Nombre	Descripción
P1	Palet C1 en posición superior.
P2	Palet C1 en posición inferior.
P3	Palet C2 en posición superior.
P4	Palet C2 en posición inferior.
P5	Máquina M1 procesando pieza tipo A
P6	Máquina M1 procesando pieza tipo B
P7	Máquina M2 procesando pieza tipo B
P8	Máquina M2 procesando pieza tipo A

La tabla 2 describe las transiciones asociadas a eventos ocurridos durante la ejecución del proceso industrial.

La ocurrencia de estos eventos requiere que se cumplan las condiciones necesarias para habilitar dichos eventos. A su vez, cada vez que ocurre un evento se generan nuevas condiciones en el proceso que permitirán eventos posteriores.

Tabla 2. Transiciones asociadas a la ejecución del proceso.

Nombre	Descripción
T1	Máquina 1 termina de procesar pieza tipo A, empieza descenso de palet C1 y Máquina 1 empieza a procesar pieza tipo B
T2	Máquina 2 termina de procesar pieza tipo B, palet C1 llega a su destino y máquina 2 empieza a procesar pieza tipo A
T3	Máquina 1 termina de procesar pieza tipo B, asciende palet C2, Máquina 1 empieza a procesar pieza tipo A.
T4	Máquina 2 termina de procesar pieza tipo A, llega palet C2 a posición superior, empieza a descender palet C2 y Máquina 2 empieza a procesar pieza tipo B

2.3 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 valores enteros de acuerdo a la cantidad de mensajes o marcas presentes en cada lugar.

Los valores del vector cambian cada vez que ocurre una transición, ya que la marca 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. Incluso puede proyectarse a lenguajes de bajo nivel, y así obtener la implementación de un control supervisorio basado en redes de Petri. La fórmula de evolución de este árbol de alcance tiene la siguiente forma:

$$\mu_i = \mu_{i-1} + A_{m \times n} * T_{n \times 1} \quad (1)$$

Donde μ_i es el i-ésimo vector de marcación. A es la matriz de incidencia de la red de Petri, T es el vector de transiciones que se van a disparar. Para nuestro caso de estudio de las dos máquinas y las posiciones de los palets, tenemos la siguiente estructura del vector de lugares:

$$\mu = [P1 \ P2 \ P3 \ P4 \ P5 \ P6 \ P7 \ P8]^t \quad (2)$$

Debe haber una marcación inicial, o sea unas condiciones en que el sistema empieza a operar en el tiempo $t = 0$. Supongamos que el vector de marcación inicial tiene la siguiente forma:

$$\mu_0 = [1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1 \ 0]^t \quad (3)$$

Que se interpretaría de la siguiente manera: El palet C1 se encuentra en la posición superior, y el palet C2 se encuentra en la posición inferior. La máquina 1 está procesando la pieza A y la máquina 2 está procesando la pieza B.

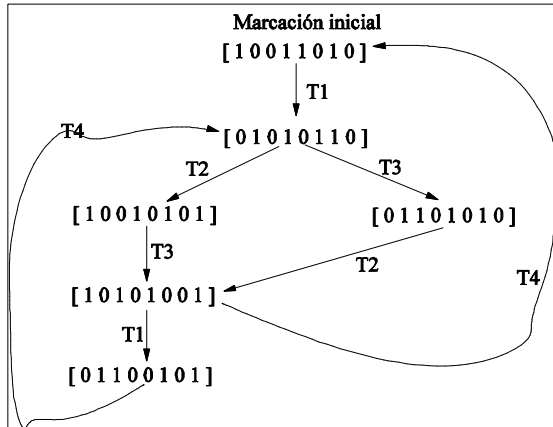


Fig. 4: Arbol de alcance

3. PROYECCION HACIA UNA RED DE COLAS

3.1 Generalidades

Los modelos que representan a sistemas de eventos discretos son 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, 2004). Resumiendo a varios autores, los elementos de interés de los modelos de eventos discretos son:

- Actividades: tareas que se desarrollan en el sistema, delimitadas por dos eventos: uno que la inicia y uno que la termina.
- Entidades: 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).
- Atributos: son las propiedades de los recursos y de los objetos temporales del sistema.

3.2. 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

& 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.

Tabla 3. Tipos de nodos que implementa el lenguaje de simulación Glider, basado en DEVS.

Nodo tipo	Función
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.

3.3 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:

I. Tomar la representación formal de la Red de Petri.

I.1. A partir de los lugares de la red, clasificar:

- 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

II. 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.

III. 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.

3.4. 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, 1999) aparece a continuación, aplicándose al sistema objeto de estudio.

Entidades. Se pueden dividir en dos: 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.

Atributos: Se pueden considerar como atributos el tiempo promedio que le toma ensamblar las piezas. Para el robot 1, podemos considerar el tiempo que toma ensamblar las piezas A y B como una variable que se distribuye como una variable aleatoria gaussiana¹, con una media y desviación determinada. Lo mismo podemos considerar para el robot 2, el cual tiene que ensamblar el subproducto AB con la pieza C. Para los insumos y subproductos que están circulando, el número de

orden es importante, ya que el insumo A #1 solo puede ensamblarse con el insumo B #1, para dar origen al subproducto AB #1, el cual solo se puede ensamblar con el insumo C #1.

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 de la figura 3, se diseñó una red de nodos que representan a las entidades mencionadas previamente, donde cada nodo representa o bien una entidad o un evento. 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 siguiente figura.

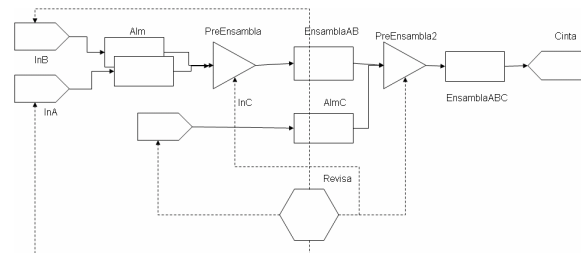


Fig. 5: Diagrama de la red de nodos

3.5. Descripción de los nodos

- InA: es un nodo de tipo Input, el mensaje que genera representa al insumo A cuando se da la orden de que se coloque en la mesa para que trabaje el robot 1.

¹ Normal truncada, es decir, no admite valores negativos.

- InB: es un nodo de tipo Input, el mensaje que genera representa al insumo B cuando se da la orden de que se coloque en la mesa para que trabaje el robot 2.
- Alm: es un nodo de tipo Resource, tiene capacidad 2 y 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. El robot 1 toma ambos insumos.
- EnsamblaAB: nodo de tipo Resource, simula el proceso de ensamblado de los insumos A y B para obtener el producto intermedio AB. Tiene una duración que se distribuye estadísticamente como una normal truncada.
- InC: es un nodo de tipo Input, el mensaje que genera representa al insumo C, cuando se da la orden de que se coloque en la mesa para que la trabaje el robot 2.
- 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. El mensaje que representa al insumo C ya se ha colocado en la mesa, proveniente del nodo AlmC, y se ubica junto con el producto intermedio AB, que proviene del nodo EnsamblaAB.
- EnsamblaABC: nodo de tipo Resource, simula el proceso de ensamblado de los insumos AB y C para obtener el producto final ABC. La duración de este proceso se distribuye como una variable normal truncada.
- 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. SIMULACION ESTOCASTICA DEL SISTEMA A EVENTOS DISCRETOS

El diagrama de la red de colas 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 fueron:

Tiempo:	Evento:
0	Activación de paso de insumos A y B
1	Insumo A habilitado
1	Insumo B habilitado
1	Revisión del sistema
1.997	El robot1 toma el insumo B
2	Revisión del sistema
2.166	El robot1 toma el insumo A
3	Revisión del sistema
3.845	Se ensambló y se obtuvo AB y la mesa giró, el robot 2 tomó AB
4	Revisión del sistema
5	Insumo C habilitado
5	Revisión del sistema
6	Revisión del sistema
6.056	El robot2 toma el insumo C
7	Revisión del sistema
8	Revisión del sistema
8.255	Se ensambló y se obtuvo ABC, el robot2 colocó en la cinta a AB.
9	Revisión del sistema
10	Insumo A habilitado
10	Insumo B habilitado
10	Revisión del sistema
10.979	El robot1 toma el insumo B
11.050	El robot1 toma el insumo A
.....	
49	Revisión del sistema
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 tres replicaciones, con una producción de 25 unidades ensambladas. Un resultado típico arrojado por Glider tiene la siguiente forma:

```
Basic Experiment
Time 219.00 Time Stat. 219.00 Replication 2
```

```

12/10/2005 13h 21m 36s
Elapsed time 0h 0m 6.990s
Nod/Ind Ant/Li #Ent Lgth Max Mean Dev MaxSt
MeanSt Dev T.Free
INA
 1 # Gen. 25
INB
 1 # Gen. 25
ALM
 1 EL 25 0 1 0.0 0.0 0.0 0.0 0.0 219.000
  IL 25 0 1 0.11283 0.31639 1.19301 0.94376
0.31636 194.290
 2 EL 25 0 1 0.0 0.0 0.0 0.0 0.0 219.000
  IL 25 0 1 0.11586 0.32006 1.18756 0.96979
0.32117 193.626
PREENSAMBLA
 1 EL 50 0 2 0.01419 0.11829 0.36358 0.06217
0.09491 215.891
ENSAMBLAAB
 1 EL 25 0 1 0.0 0.0 0.0 0.0 0.0 219.000
  IL 25 0 1 0.17131 0.37678 1.85583 1.44009
0.48057 181.483
INC
 1 # Gen. 25
ALMC
 1 EL 25 0 1 0.0 0.0 0.0 0.0 0.0 219.000
  IL 25 0 1 0.11534 0.31943 1.09894 0.96810
0.29509 193.741
PREENSAMBLA2
 1 EL 50 0 2 0.27921 0.44861 2.94854 1.22292
1.23424 157.854
ENSAMBLAABC
 1 EL 25 0 1 0.0 0.0 0.0 0.0 0.0 219.000
  IL 25 0 1 0.24525 0.43023 2.82321 2.07252
0.61863 165.291
SALEABC
 1 EL 25 0 1 0.0 0.0 0.0 0.0 0.0 219.000
Time in System: Mean 7.15875 Dev. 0.32629 Max.
7.89281 Min. 6.28330
REVISA

```

Revisando las replicaciones, observamos 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 esperara 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 enuncian las siguientes conclusiones:

Las redes de Petri como herramienta de modelado matemático permiten establecer de una manera relativamente sencilla la supervisión de procesos de manufactura determinados por la ocurrencia de eventos discretos. Aunque su síntesis es más difícil a medida que aumenta la complejidad del sistema a controlar.

Existen herramientas para validar redes de Petri, aunque su simulación estadística para medir su eficiencia requiere de la utilización de modelos de simulación basados en redes de colas. En particular, para medir el grado de utilización de los recursos dentro de un sistema, y así poder sugerir una optimización posterior del sistema.

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., Nicol, D. *Discrete-event system simulation*. Editorial Prentice-hall. Upper saddle river, 2000.
- Beamon, Benita M. (1998): "Supply Chain Design and Analysis: Models and Methods". *International Journal of Production Economics* 20 (3): 281-294.
- Cesimo & IEAC. *Glider Reference Manual*. Universidad de los Andes, Mérida. 1996.
- Chacón, E., De Sarrazin, G., and Besembel, I. *Automatización industrial*. (Universidad de Los Andes) Mérida. 2004.
- ENIAC Organization (2002): "Supply Chain in Action". <http://www.eniac.com/notesupp.htm> (10 de diciembre de 2005).
- Fishwick, P. *Simulation model design and execution*. Editorial Prentice-Hall. Englewood cliffs, 1995.
- Gaonkar, R. y Viswanadham, N. (2005): "Strategic and Collaborative Planning in Internet-Enabled Supply Chain Networks Producing Multigeneration Products". *IIIE Transaction on Engineering Management* 2 (1): 54-55.
- Guasch, A., Piera, M., Casanovas, J., Figueras, J. *Modelado y Simulación. Aplicación a procesos logísticos de fabricación y servicios*. Editorial Alfaomega. Ciudad de México. 2005.

- Handfield, R. B. y Nichols, E. L. (1999): "Introduction to Supply Chain Management". Englewood Cliffs. New Jersey 07458, USA. Huish
- Lambert, D. M.; Cooper, M. C. y Pagh, J. D. (1998): "Supply Chain Management: Implementation issues and research opportunities". *Logistic Management* 9 (2): 1-19.
- Lee, Y. H. y Kim, S. H. (2002): "Production-distribution planning in supply chain considering capacity constraints". *Computers & Industrial Engineering* 43: 169-190.
- Marbet, V. A. y Venkataramana, M. A. (1998): "Special research focus on Supply chain linkages: Challenges for design and management in the 21st century". *Decision Science* 29 (3): 537-552.
- Murata, T., *Petri Nets, properties, análisis and applications*. Proceedings of the IEEE, vol 77, No. 4. April 1989.
- Peterson, J. *Petri Net Theory and the modeling of systems*. Editorial Prentice-Hall. Englewood cliffs. 1981.
- Ricyt-Fapesp. (2004): "Estrategias metodológicas y experiencias recientes de medición del impacto social de la ciencia y tecnología". *Relatoría General*. Taller de San Pablo, Brasil – 3 y 4 de agosto: 1-6.
- Rosenblatt, M. y Lee, H. L. (1985): "Improving profitability with Quantity Discounts under Fixed Demand". *IIE Transactions* 17 (4): 388-395.
- Rozembert, D. (2000): "Cadena de suministros - Adiós a los inventarios". Information Week México.
http://www.infoweek.com.mx/articulo.php?id_articulo=324 (3 noviembre 2004).
- Sasson, R. R. (2005): "La cadena de suministro". Monografías en Línea.
<http://www.monografias.com/trabajos31/cadena-suministros/cadena-suministros.shtml> (15 de septiembre 2006).
- Simchi-Levi, D.; Kamisky, P. y Simchi-Levi, E. (2000): *Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies*. McGraw-Hill International Edition, Singapore.
- Sudhindra, K. y Vinayak, R. (2004): *E-Manufacturing & SAP-Creating Responsive Shop Floor in the Supply Chain*. *Wipro Technologies*: 1-14.
- Vachon, S. y Klassen, R. D. (2002): *An Exploratory Investigation of the Effects of Supply Chain Complexity Performance*. *IIIE Transaction* 49 (3): 218-230.