

Implementación de Redes de Petri para el Modelado y Simulación de un Proceso Industrial Mecatrónico

Xochipiltecatl Henry¹, Cano Ariana¹, Flores Irma¹, Carro Jorge¹, Rodríguez Julio², Santiago Samuel² y Mercado Eduardo²

(1)Universidad Politécnica de Tlaxcala, (2) Universidad Tecnológica de Huejotzingo

Resumen

El presente trabajo muestra el uso de Redes de Petri (PN) para el modelado y simulación de Sistemas Dinámicos de Eventos Discretos (DEDS). Esta herramienta matemática provee un ambiente uniforme para analizar un proceso industrial mecatrónico con dos fines, por un lado describir el proceso de un Sistema Flexible de Manufactura (FMS) compuesto por dos robots de seis grados de libertad y por el otro generar los algoritmos para la programación del Controlador Lógico Programable (PLC) que apoyará y sincronizará el FMS. La importancia de utilizar PN en el modelado permite observar los movimientos del FMS y obtener la programación del PLC de forma relativamente fácil.

El modelado por PN se hace por la técnica de árbol de cobertura, la simulación se realiza mediante el software HPsim.

Abstract

This paper demonstrates the use of Petri Nets (PN) for modeling and simulation of Discrete Event Dynamic Systems (DESD). This mathematical tool provides a uniform environment for analyzing an industrial process for two purposes, describe the process of a flexible manufacturing system consisting of two robots with six degrees of freedom and the other generates the algorithms for scheduling Robots and Programmable Logic Controller (PLC) to support and synchronize the FMS. The importance of using PN in modeling will allow for the PLC program with relative ease.

The PN modeling is done by the technique of tree cover, the simulation is performed using the software HPsim.

Palabras clave:

Redes de Petri, modelado, simulación, PLC.

1. Introducción

La mejora de procesos en la industria requiere de herramientas matemáticas y de software robusto y de fácil implementación. Al utilizar PN propuestas por Carl Adams Petri, desarrolladas entre los años 1960 a 1962, se logra acortar el tiempo de simulación y análisis de un FMS, las PN nos permiten eficientar el tiempo de modelación al observar el proceso como un sistema DEDS como lo menciona Murillo, 2008 [1], referenciando a W Scruben, 1992, G. Cansever 2006 y G. B. Lee 2004, Murillo, 2008 también hace referencia que las PN permiten predecir y mejorar rutas, sincronizar operaciones y tener una visión práctica de los requerimientos, Montoya-Torres, 2007 [2] abogan por la conceptualización de los modelos empleados, sin embargo la interrelación que se debe observar en el diseño y construcción de un sistema Mecatrónico permite utilizar PN y distinguir al sistema como un todo, Isermann, 2005[3].

El análisis con las PN como un sistema DEDS es posible dado que se puede representar un proceso como un sistema lineal de ecuaciones, esto permite fácilmente obtener una Matriz de incidencia y/o ecuación de estado, como lo describe S. Lee, 2004 [4]. Dicha ecuación permite encontrar la programación del PLC de manera natural. El proceso a modelar es un FMS que consta de dos robots ABB de seis grados de libertad y dos procesos de pintado.

En la primera parte de este trabajo se observará las definiciones de la PN, seguidas de sus propiedades, el diagrama obtenido y finalmente los resultados del trabajo.

2. Redes de Petri

Las PN son gráficos donde se muestran principalmente tres tipos de objetos, Lugares (P), Transiciones (T) y Arcos, donde son representados por círculos, cuadros o rectángulos y arcos respectivamente. Dos lugares pueden unirse mediante una transición y arcos que determinen el flujo de entrada o salida. A los lugares pueden agregarse tokens (puntos) que permiten observar la transición entre verdadero y falso, cuando un token se encuentra en un lugar se denomina que está marcado, los tokens nos permiten observar la ejecución de la transición y es posible agregar la cantidad de tokens que consideremos al modelar el sistema.

Murata 1989 referenciado por Wang [5] menciona que los lugares de entrada pueden observarse como datos, señales, condiciones, etc., los lugares de salida puede observarse como datos, señales, y conclusiones., las transiciones pueden representar eventos, una cláusula lógica, un retardo, procesar una señal, etc.

2.1 Representación gráfica

Una PN es una 5-tupla $N = (P, T, I, O, M_0)$

Donde:

- 1.- $P = \{p_1, p_2, \dots, p_m\}$ es un número finito de lugares.
- 2.- $T = \{t_1, t_2, \dots, t_n\}$ es un número finito de transiciones
 $P \cup T \neq \emptyset$, y $P \cap T = \emptyset$
3. $I: (P \times T) \rightarrow \mathbb{N}$, es una función de entrada que define directamente arcos de lugares a transiciones donde $\mathbb{N}$ es un número entero no negativo.
- 4.- $O: (P \times T) \rightarrow \mathbb{N}$, es una función de salida que define directamente los arcos de transiciones a lugares.
- 5.- $M_0: P \rightarrow \mathbb{N}$ es una marca inicial.

Así un arco dirigido de un lugar (p_j) a una transición (t_i) define que (p_j) será un lugar de entrada denotado por $I(t_i, p_j)=1$, de otra forma, un arco dirigido de una transición (t_i) a un lugar (p_j) define que (p_j) será un lugar de salida denotado por $O(t_i, p_j)=1$, si $I(t_i, p_j)=k$ o bien $O(t_i, p_j)=k$ entonces k significa que son paralelos estos arcos y lugares.

La representación gráfica está dada por una marca inicial $M_0 = (1 \ 0)^T$ en un tiempo donde t_1 no está activo representado por la figura 2.1 (a), se activa t_1 resultado una nueva marca M_1 figura 2.1 (b), resultando $M_1 = (0 \ 1)^T$

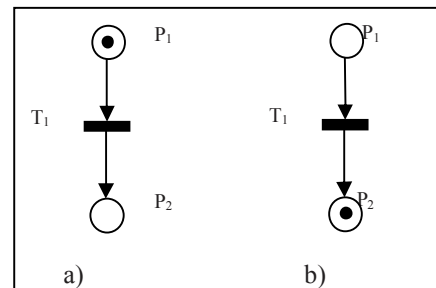


Fig. 1 Disparo de t_1 en una PN

2.2 Reglas de transición

El disparo sucede por las reglas de disparo y de permiso.

Regla de permiso: Una transición t es habilitada si en cada lugar (p) de t contiene al menos un numero igual de tokens al que se esta conectando

Regla de disparo:

- a) Una transición t puede o no dispararse dependiendo de otra interpretación adicional
- b) Un disparo de una transición habilitada t mueve el token de cada lugar (p) con el peso de cada arco que está conectando de p a t , esto es que se deposita el numero de tokens en el lugar (p) de salida dependiendo el peso del arco generando una nueva marca.

El disparo es representado por:

$$M'(p) = M(p) - I(t,p) + O(t,p) \dots (1)$$

para cualquier p en P .

2.3 Representación de actividades de la FMS

La ventaja modelar por PN es observar las características del proceso. Esto es posible al observar las características de las PN: secuencia, concurrencia, sincronización y prioridades, que nos permiten tomar decisiones sobre tiempos, rutas y rediseño del proceso.

Las PN permiten reducir el tiempo de ciclo de la producción, buscando los cuellos de botella, estaciones de trabajo verificando limitantes de tiempo.

Las propiedades de las PN: por comportamiento (estado inicial) y estructural (topología o estructura de red), nos permitirán observar que el modelado del sistema cumpla con el comportamiento de un sistema esto es que sea asequible, acotable, conservable, alcanzable

controlable vivaz y persistente como lo menciona Murata, 1989.

2.4 Sistemas de control de la FMS

El principal control de la FMS será el PLC, funcionando como esclavos los robots pintadores. Los robots realizan una serie de secuencias, y cuando los sensores el PLC determina las acciones a realizar. Los robots poseen su propio centro de toma de decisiones sin embargo es el PLC quien coordina las actividades.

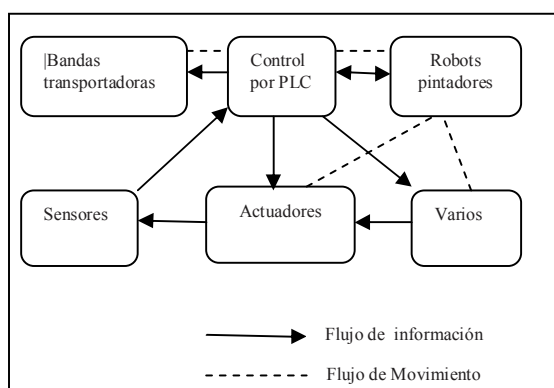


Fig. 2. Flujo de información y movimientos.

3. Ejemplo de aplicación

La FMS consta de dos brazos robots ABB con seis grados de libertad encargados de realizar el pintado de las piezas, el primero tarda ocho segundos en hacer el acabado y el segundo solo uno, para ello se apoyan de bandas de rodamiento, pistolas de pintado, y dos mesas giratorias.

Su funcionamiento es de la siguiente manera: existe a la entrada una banda transportadora que alimenta de material al robot 1, cuando llega el material al robot 1, este procede a realizar el pintado para ello tiene que accionar la mesa giratoria y abrir la válvula de la pintura, después de ocho segundos, se cierra la válvula, se frena la mesa y la pieza circula mediante la banda transportadora hacia el segundo robot de pintado, una vez que llega la pieza al robot 2 se procede a girar la mesa dos y abrir la válvula de pintura esto durante un segundo, terminando el tiempo se procede a cerrar la válvula, frenar la mesa giratoria y hacer avanzar a la pieza que tiene una duración de un segundo para finalmente salir por la banda transportadora de salida.

La siguiente tabla muestra las actividades y transiciones simuladas mediante PN.

Lugar	Descripción
P1	Banda de entrada
P2	Carga Pieza
P3	Sensor posición
P4	Disposición robot1
P5	Giro Base
P6	Apertura Válvula
P7	Robot 1
P8	Freno Base
P9	Cierra Válvula
P10	Avance Banda
P11	Posición final Banda
P12	Giro base 2
P13	Apertura Válvula 2
P14	Robot 2
P15	Disposición robot 2
P16	Freno Base 2
Transición	Descripción
T1	Procesamiento señales
T2	Procesamiento señales
T3	Tiempo pintado (retardo)
T4	Procesamiento señales
T5	Retardo
T6	Procesamiento señal
T7	Tiempo pintado 2 (retardo)
T8	Procesamiento de señal
T9	Procesamiento de señal

Tabla 1 Actividades y tiempos en la PN.

3.1 Simulación

Pardo 2001 [6] menciona que si bien existen diferentes trabajos donde se realiza simulación de las PN, son los mismos autores quienes construyen e implementan estos programas. En este trabajo se utilizó software desarrollado por distintos institutos de investigación, universidades y empresas, entre ellos DiaGen-eclipse- ver. Galileo, HPsim 1.1, Snoopy 2.0, TINA 2.9.6, y MATLAB 7.1 utilizando el Modelo de Calidad (MOSCA+) para evaluar software de simulación de eventos discretos propuesto por Rincón 2001 [7], el comparativo se muestra en la tabla 2.

	Funcionalidad	Usabilidad	Eficiencia
DiaGen	√		
HPsim		√	√
Snoopy	√		√
TINA	√		√
MATLAB	√		√

Tabla 2 Comparativo MOSCA por categorías de cinco tipos de software de simulación

En la siguiente figura 3 se muestra el resultado de la simulación mediante el software HPsim, se inician con tres tokens que serán el número de objetos iniciales a ser pintados en los dos procesos. Los robots están enmarcado en el centro del dibujo, de ellos se observa la sincronización y los tiempos de espera.

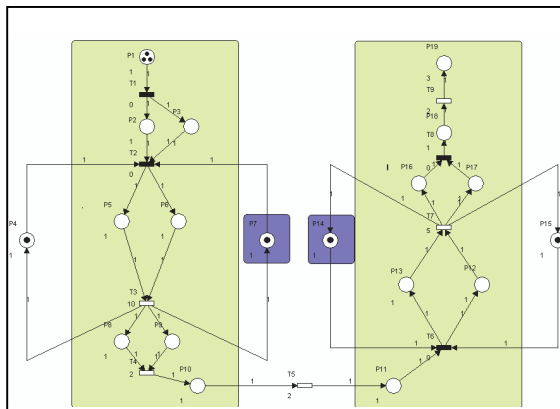


Fig. 3 Disparo de t_0 en una PN Secuencial

Otra forma de observar el comportamiento del dinámico de la PN es mediante la construcción de una Matriz binaria llamada matriz de incidencia que representa las relaciones que existen entre los lugares, transiciones y los tokens, estos últimos se representan como enteros negativos para mostrar las entradas y las salidas con enteros positivos.

Así tenemos que la matriz de incidencia es:

```
( 1 0 0 0 0 0 0 0 0 )
(-1 1 0 0 0 0 0 0 0 )
(-1 1 0 0 0 0 0 0 0 )
( 0 1 -1 0 0 0 0 0 0 )
( 0 -1 1 0 0 0 0 0 0 )
( 0 -1 1 0 0 0 0 0 0 )
( 0 1 -1 0 0 0 0 0 0 )
( 0 0 -1 1 0 0 0 0 0 )
( 0 0 -1 1 0 0 0 0 0 )
( 0 0 0 -1 1 0 0 0 0 )
( 0 0 0 0 -1 1 0 0 0 )
```

```
( 0 0 0 0 0 -1 1 0 0 )
( 0 0 0 0 0 -1 1 0 0 )
( 0 0 0 0 0 1 -1 0 0 )
( 0 0 0 0 0 1 -1 0 0 )
( 0 0 0 0 0 0 -1 1 0 )
( 0 0 0 0 0 0 -1 1 0 )
( 0 0 0 0 0 0 0 -1 1 )
( 0 0 0 0 0 0 0 -1 1 )
```

Esta matriz permite hacer el cambio durante la transformación a lenguaje escalera.

Resultados

Se obtiene de forma sencilla la simulación de un FMS mediante PN y se realiza su simulación por medio de HPsim, para observar posibles errores o puntos muertos (deadlocks) que puedan ocurrir en las transiciones. Esto es de suma importancia para el diseño Mecatrónico donde los sistemas mecánicos son influenciados por los sistemas electrónicos y viceversa, donde debe observarse el diseño como un sistema orgánico Isermann 2003 [3]

Se reconoce la necesidad de utilizar una metodología para transformar la PN en lenguaje escalera, se estudia actualmente las propuestas de Peng, 2004 [8], G. Cansever, 2006 [9] y Chirn – McFarlane 2000 [10] para aplicarlas al entorno real basándonos en la norma IEC 61131-3.

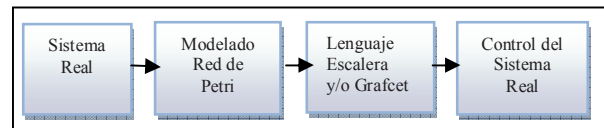


Fig. 4. Secuencia de actividades para convertir de PN a lenguaje PLC.

Trabajo Futuro

Se propondrá una metodología para desarrollar este tipo de sistemas donde se permita identificar cuando aplicar el método gráfico y hasta qué punto el método algebraico.

Siguiendo los trabajos de Mina 2002 [11], se comprobará la eficiencia del software DiaGen Eclipse versión galileo la conversión de PN a lenguaje PLC, verificando los resultados para algoritmos de control

Referencias

- [1] Murillo L. “Redes de Petri: Modelado e implementación de algoritmos para autómatas programables” Revista Tecnología en Marcha, Vol. 21, N.º 4, Octubre-Diciembre 2008, pp. 102-125
- [2] Montoya-Torres, J. R. Modelisation Conceptuelle D'une Unite de Fabrication Microelectronique, Revista EIA, ISSN 1794-1237 Numero 7, p. 9-24. Junio 2007
- [3] Isermann R, Mechatronic Systems Fundamentals, Ed. Springer; London, first edition, 2005.
- [4] Lee J. S., Hsu P. L.. “A improved evaluation of ladder logic diagrams and Petri nets for sequence controller design in manufacturing systems”. International Journal of Advanced Manufacturing Technology. Vol. 24, No. 3-4, pp 279-287. August 2004
- [5] J. Wang . “Timed Petri Nets, Theory and application”. Kluwer Academic Publishers, USA, 1era. 1998.
- [6] Pardo, M.E., Izquierdo. J.M. Obregón, O. Ejemplos de modelado y control de procesos mediante redes de petri “ Revista ingeniería electrónica, automática, y comunicaciones, Vol. XXII, No.3 2001
- [7] Rincón G., Pérez M, Hernández, S. Modelo de Calidad (MOSCA+) para evaluar software de simulación de eventos discretos
- [8] S. S. Peng, M. C. Zhou. “Sensor bases stage Petri nets modeling of PLC logic programs for discrete control design”. *International Journal of Production Research*, Vol. 41, No. 3, pp 629-644, 2003.
- [9] G. Cansever, I. B. Kucukdemiral. “A New approach to supervisor design with sequencial control Petri nets using minimization technique for discrete event systems”. *International Journal of Advanced Manufacturing Technology*. Vol. 29, No. 11-12, pp 1267-1277. August 2006.
- [10] J-L Chim, J-L. McFarlane ,D. C. Petri Nets based design of ladder logic diagrams, *Control 2000* Institute for Manufacturing University of Cambridge pag- 4-6, 2000
- de Procesos y Sistemas, Universidad Simón Bolívar Caracas – Venezuela
- [11] Mina, M. Frey, G. Visula PLC- Programing using Signal Interpreted Petri Nets. Proceedings of the American control, conference 2002 (ACC2002), anchorage Alaska, pp, 5019- 5024, May 2002
- [12] Dunlop , S. Jones, B., McFarlane, D, Petri Net Based Design and Diagnosis for an Assembly Cell Proceedings of SafeProcess 97, Hull, England, 1997