

Política de despacho para evitar bloqueos en celdas de manufactura flexible del tipo multi-reentrante

Policy dispatching to avoid blockages in flexible manufacturing cells type multi-reentrant

Luis Diego Murillo-Soto¹

Fecha de recepción: 13 de febrero del 2013

Fecha de aprobación: 17 de junio del 2013

Murillo, L. Política de despacho para evitar bloqueos en celdas de manufactura flexible del tipo multi-reentrante. *Tecnología en Marcha*. Vol. 26, N° 4. Pág 42-51

¹ Escuela de Ingeniería Electromecánica, Instituto Tecnológico de Costa Rica. 159-7050 Cartago, Costa Rica. CRC. Tel: 506-2550-2366. Correo: lmurillo@tec.ac.cr.

Palabras claves

Política de despacho; controlador de eventos discretos; celda flexible de manufactura del tipo multi reentrantes; Red de Petri.

Resumen

Una Celda de Manufactura Flexible (CMF) se puede modelar usando la teoría pertinente a sistemas de eventos discretos. La CMF integra un conjunto de recursos, operaciones y materias primas que se combinan para la elaboración de productos. Si esos recursos no se controlan adecuadamente, la CMF se bloqueará. Con el fin de evitar dicho bloqueo, en este trabajo se presentan cuatro lineamientos para construir una política de asignación de recursos a partir del estudio básico de esperas circulares y prioridades de producción. Para elaborar una CMF teórica planteada por Peng y Zhou (2003), y desde los lineamientos propuestos, se construyó una política de despacho y se efectuaron simulaciones de la celda de manufactura con y sin controlador. Como resultado, la política de despacho logró realizar las asignaciones de recursos en forma adecuada para el ensamblaje de las partes.

Keywords

Dispatch policy; discrete event controller; flexible manufacturing cell; multi reentrant line; Petri nets.

Abstract

A Flexible Manufacturing Cell (FMC) can be modeled using relevant theory of discrete event systems. The FMC integrates a set of resources, operations and materials that was combined to product processing. If those resources are not adequately controlled, the FMC will be blocked. In order to avoid such blocking, this paper presents four guidelines to build a resource allocation policy from basic study of circular wait and production priorities. To develop a theoretical CMF raised by Peng and Zhou (2003), and from the proposed guidelines was constructed a dispatch policy. The policy was tested through simulations of the FMC, and as a result, the policy allocations were performed properly for the assembly of the parts.

Introducción

Para diseñar el controlador supervisor de una Celda de Manufactura Flexible (CMF) es necesario tener el conocimiento sobre la dinámica de los eventos del sistema productivo, así como habilidades en programación lineal para la búsqueda óptima del controlador. Estas características hacen que el diseño de controladores para CMF complejos, tal como en los sistemas denominados Free choice Multiple Re-entrant Flowlines (FMRF), esté solamente al alcance de estudiantes de posgrado especializados. Por tanto, se desarrolló un método para establecer una política de asignación de recursos a tareas que evite los bloqueos en el sistema, a partir de las esperas circulares y la prioridad de la producción.

Por otra parte, la política tiene la capacidad de ser trasladada en forma directa a un autómatas programable (PLC) o microcontrolador, por lo que su utilización para construir celdas de manufactura en nuestros ámbitos locales, usando equipos genéricos

de control y máquinas herramienta, representa una oportunidad de independencia tecnológica y de mejoramiento de nuestros sistemas productivos.

El presente trabajo se divide en seis partes. La segunda sección presenta el marco teórico, la tercera la descripción de la celda de manufactura, la cuarta sección los lineamientos propuestos, la quinta muestra la política de control y la sexta las simulaciones. Finalmente, la última contiene el análisis de resultados.

Marco Teórico

Redes de Petri

Una Red de Petri clásica se conciben como un grafo dirigido que posee dos tipos de nodos principales: los lugares representados por círculos y las transiciones representadas por barras rectangulares, (figura 1). Entre los nodos se ubican los arcos dirigidos, los cuales se encargan de unir las transiciones con los lugares y viceversa. Cada arco dirigido posee un

número que indica su peso; este peso determina la cantidad de marcas que consume o deposita en un nodo lugar. Este movimiento de marcas ocurre siempre y cuando se haya disparado una transición habilitada. Se entiende que los arcos dirigidos sin peso consumen o depositan una marca. Las marcas o tokens se representan en forma gráfica como puntos negros que se ubican dentro de cada nodo lugar.

De acuerdo con (Murata, 1989), la Red de Petri (RdP) se define como una quintupla:

$$\text{RdP} = (P, T, F, W, M_0) \quad (1)$$

Donde:

$P = \{p_1, p_2, \dots, p_m\}$ es un conjunto finito de lugares.

$T = \{t_1, t_2, \dots, t_n\}$ es un conjunto finito de transiciones.

$F \subseteq (P \times T) \cup (T \times P)$ es un conjunto de arcos dirigidos.

$W: F \rightarrow \{1, 2, 3, \dots\}$ es la función de pesos de los arcos.

$M_0: P \rightarrow \{1, 2, 3, \dots\}$ es el marcado inicial de la red.

$P \cap T = \emptyset$ y $P \cup T \neq \emptyset$.

Se utiliza la siguiente notación de puntos para referirse a nodos específicos en la red, por ejemplo sea p un nodo tipo lugar; la notación $\bullet p$ indica a los nodos anteriores a p , estos son los nodos tipo transición t . La notación $\bullet \bullet p$ denota los nodos tras anteriores que son tipo p . De igual forma aplica para los nodos siguientes al actual $p \bullet$ y a los sub siguientes $p \bullet \bullet$. Esta notación también es utilizada por los nodos tipo transacción, ver detalle en Murata, (1989), Lewis et al. (1998) y Huang, (2001).

Celdas de manufactura multireentrantas.

La celda de manufactura se visualiza como la interacción de un conjunto de recursos, que ejecutan secuencias de operaciones para producir distintos bienes a partir de partes específicas. Usando la notación de Bogdan et al. (2006), se puede definir la celda de manufactura flexible como la tupla:

$$\text{CMF} = (\Pi, J, R)$$

Donde:

$\Pi = \{A, B, C, \dots, k\}$ es el conjunto de partes a producir.

$J = J^A \cup J^B \cup J^C \cup \dots \cup J^k$ es el conjunto de operaciones sobre las k partes.

$J^k = \{J_1^k, J_2^k, J_3^k, \dots\}$ conjunto de operaciones secuenciales sobre la parte k .

$R = R_s \cup R_{ns}$ es el conjunto de recursos que desempeñan operaciones sobre las partes. Este conjunto está conformado por los subconjuntos de recursos compartidos y no compartidos.

$R_s = \{r_s : |J(r_s)| > 1\}$ es el conjunto de recursos que desempeñan más de una tarea.

$R_{ns} = \{r_{ns} : |J(r_{ns})| = 1\}$ es el conjunto de recursos que desempeñan sólo una tarea.

De vital importancia para la celdas de manufactura son los recursos que le brindan la característica de flexibilidad, es decir aquellos recursos que desempeñan más de una tarea distinta sobre las piezas Π , $r_s \subset R_s$. Dado que los recursos compartidos son los elementos de flexibilización, se dice que el flujo de las partes Π que se procesan está re-entrando al recurso al menos dos veces, de allí se deriva el nombre de celdas "Multiple Re-entrant Flowlines" (MRF). La figura 2, muestra que los flujos de las partes A, B, C ingresan varias veces a ser procesadas por el robot. Las definiciones formales de sistemas MRF se pueden encontrar en Huang et al. (1996) y Bogdan et al. (2004).

Existe un caso más general de celdas de manufactura llamadas "Free choice Multiple Re-entrant Flowlines" (FMRF). Estas celdas poseen las siguientes dos características:

- La celda tienen dos o más recursos, que pueden desempeñar la tarea J_i^k , y cada recurso puede desempeñar varias tareas, ya que $r_i \in R_s$.
- La celda puede presentar múltiples rutas de trabajo, estas rutas no son determinísticas y dependen del estado de ocupación de los recursos en la celda (Mirelles y Lewis, 2002, Ballal et al., 2007).

El modelado de una CMF usando las Redes de Petri se realiza usando los criterios que aparecen en Silva (1998), Zhou y Venkatesh (2000) y Hruz y Zhou (2007). Un ejemplo reciente de modelado y simulación de celda de manufactura flexible usando RdP se muestra en Murillo-Soto (2010).

Bloqueos y esperas circulares

El bloqueo es una condición lógica que surge en sistemas de manufactura por la mala asignación de recursos a tareas específicas (Zhou y Fanti, 2004). Se asume que las celdas de manufactura cumplen las condiciones lógicas necesarias para que exista un bloqueo tal y como lo plantean Bogdan et al. (2006) y Silberschatz y Galvin (1999) en sistemas operativos. A continuación se presentan las condiciones necesarias para el bloqueo:

- Tareas sin derecho a prioridad; una vez asignado un recurso r_i a la tarea J_i^k , el recurso no puede ser removido a otra tarea J_j^k , hasta que la tarea J_i^k haya finalizado.
- Exclusión mutua; un recurso r_i sólo puede desempeñar una tarea J_j^k a la vez.
- Mantener mientras espera; un proceso mantiene los recursos, hasta que pueda tener todos los recursos necesarios para ejecutar la tarea.
- Espera circular (EC); es el conjunto de recursos pertenecientes a R , que están en una cadena de relaciones de espera, y esta cadena de relaciones

es circular. La relación de espera entre dos recursos se denota como $(r_i \rightarrow r_j)$ y significa que la tarea ejecutada por r_i no inicia hasta que se libere el recurso r_j . Una espera circular simple (ECs) se denota como $(r_i \rightarrow r_j \rightarrow r_k \rightarrow r_p \rightarrow r_i)$, donde los recursos necesarios para la espera deben ser al menos dos. (Lewis, Gurel, Bogdan, Doganalp, & Pastravanu, 1998)

- No se consideran fallas en las máquinas de la celda, debido a que el paro forzado fomenta los bloqueos del sistema productivo.

Se puede decir que la espera circular dentro de la celda está en bloqueo y por tanto la celda, si esta está esperando un evento que nunca ocurrirá. (Huang, 2001). Esto se conoce como bloqueo circular (BC). Por tanto, una forma de diseñar el controlador es a partir de un estudio estructural de la celda de manufactura, donde se detectan las esperas circulares y se evita que entren en BC.

Las esperas circulares se pueden obtener directamente por observación de una Red de Petri, o por álgebra de cadenas. Para mayor detalle ver Bogdan et al. (2004) y Chang et al. (1998). Existen métodos matriciales, que pueden ser vistos en Zhou y Fanti (2004) y Mirelles y Lewis (2002).

Para celdas de manufactura tipo MRF, la estrategia de control de los recursos se basa en evitar el bloqueo del sistema, buscando la mayor utilización de los recursos y estableciendo una cantidad máxima de tareas a ejecutar. Huang et al. (1996) establecen el concepto de sifones críticos (Sc) y se indica que limitando las tareas del sifón, se evita el bloqueo. A continuación los teoremas que propusieron estos autores:

Teorema 1: Teorema principal para evitar bloqueos. Sea C una espera circular y Sc su sifón crítico. Entonces, C está en bloqueo circular (BC) si y sólo si el sifón crítico Sc está vacío.

Teorema 2: Definición y construcción del sifón crítico

$$Sc = C \cap J(C)^+ = C + \sum_{i=0}^n \{ \bullet \{ \bullet r_{si} \} + \cap J \}$$

donde:

Sc es el conjunto de recursos y tareas de la celda que forma el sifón crítico.

C son los recursos que componen una espera circular.

J son las tareas de la celda de manufactura.

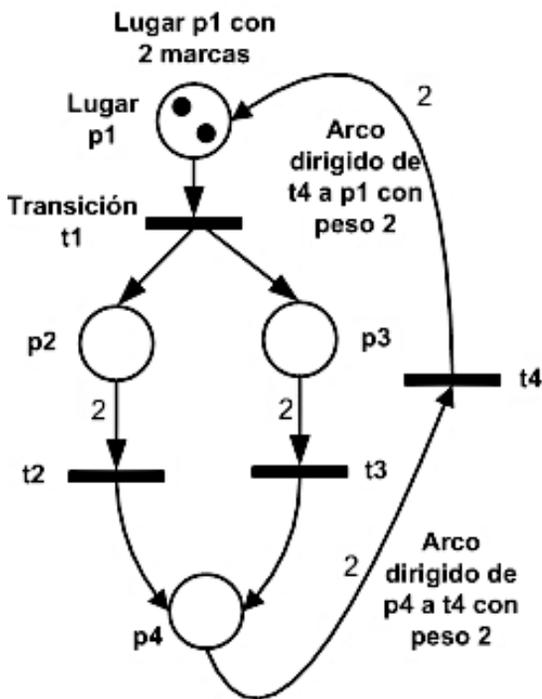


Figura 1. Red de Petri, con cuatro lugares, cuatro transiciones, cinco arcos dirigidos de peso uno y cuatro arcos dirigidos de peso dos.

- $J(C)_+$ son las tareas que realizan los recursos C , cuyos arcos de salida no apuntan a nodos ubicados dentro de la espera circular C .
- r_{si} son los recursos compartidos dentro de C .
- $\{\bullet r_{si}\}_+$ es el conjunto de transiciones cuyos arcos apuntan a r_{si} y no tienen arcos de entrada de otros recursos dentro de la espera circular C .
- $\bullet\{\bullet r_{si}\}_+$ es el conjunto de tareas cuyos arcos de salida apuntan a las transiciones definidas por $\{\bullet r_{si}\}_+$.

En el caso de las celdas de manufactura FMRF, tienen como característica la existencia de las esperas circulares cíclicas (ECC). Las ECC se conforman por la unión de ECs, la cual se realiza a través del llamado recurso crítico, que es un elemento dentro de los recursos compartidos R_s . De acuerdo con Huang (2001b) y Bogdan et al. (2006), el bloqueo se puede evitar si se cumple el siguiente teorema.

Teorema 3: Para un sistema irregular MRF, con esperas circulares llamadas C_1 y C_2 que forman una espera circular cíclica ECC con un recurso crítico, el bloqueo no ocurrirá si y sólo si el último recurso libre dentro de ECC no es el recurso crítico.

Celda de Manufactura

En Peng y Zhou (2003) se plantea una celda de manufactura que consta de dos bandas transportadoras, una máquina de control numérico (CNC), un centro de manufactura vertical (VMC), un robot (RB), un *buffer* o almacén de tamaño uno (BF), además del controlador lógico programable PLC que realiza la coordinación. La CMF debe procesar tres piezas que se llamarán A, B, C. La pieza A utiliza la máquina CNC y el robot RB, la pieza B utiliza la máquina VMC y el robot RB, la pieza C utiliza las máquinas CNC y VMC, el almacén BF y el robot RB. La idea es desarrollar un programa para el PLC que trabaje como controlador de alto nivel (supervisor) y coordine la asignación de los recursos para realizar las operaciones de la celda sin generar bloqueos.

La distribución física de las máquinas de la CMF se muestra la figura 2a, la secuencia de tareas de la CMF se detalla a través del Cuadro I, la interacción entre máquinas, secuencia de tareas y flujo de partes se muestran en la figura 2b. Se observa que la celda es del tipo FMRF ya que la piezas tipo C entran múltiples veces al robot y además existen dos rutas en conflicto después de la CNC.

Cuadro I. Símbolos de las tareas que ejecuta la celda de manufactura.

Símbolos y Tareas	
J_1^A	Banda 1 transporta la pieza A
J_2^A	Robot carga piezas A en la máquina CNC
J_3^A	CNC procesa la pieza A
J_4^A	Robot descarga piezas A en la banda transportadora 2
J_5^A	Banda 2 transporta la pieza A
J_1^B	Banda 1 transporta la pieza B
J_2^B	Robot carga piezas A en la máquina VMC
J_3^B	VMC procesa la pieza B
J_4^B	Robot descarga piezas B en la banda transportadora 2
J_5^B	Banda 2 transporta la pieza B
J_1^C	Banda 1 transporta la pieza C
J_2^C	Robot carga piezas C en la máquina CNC
J_3^C	CNC procesa la pieza C
J_4^C	Robot descarga piezas C y carga en Buffer
J_5^C	Robot almacena pieza C en Buffer
J_6^C	Robot descarga buffer y carga piezas C en la VMC
J_7^C	Robot carga piezas C en la VMC
J_8^C	VMC procesa la pieza C
J_9^C	Robot descarga piezas C en la banda transportadora 2
J_{10}^C	Banda 2 transporta la pieza C

Metodología para la construcción de una política de despacho

Las reglas que a continuación se presentan, se basan en la interpretación de los teoremas expuestos en la sección 2.2. En la literatura dichos teoremas son utilizados para construir controladores a partir de lenguajes de alto nivel como Matlab o LabView en donde se programan distintas operaciones matriciales para obtener con controlador matricial (Mireles & Lewis, 2001); sin embargo, aplicar tales teoremas de forma directa a un PLC no resulta sencillo en muchos casos.

Las siguientes reglas permiten construir un algoritmo de supervisión y control para una RdP. Las reglas buscan crear y colocar nodos de control en la red, con con la finalidad de administrar el flujo del mercado, esto se logra controlando marcas en los nodos de control de acuerdo al marcado de las esperas circulares.

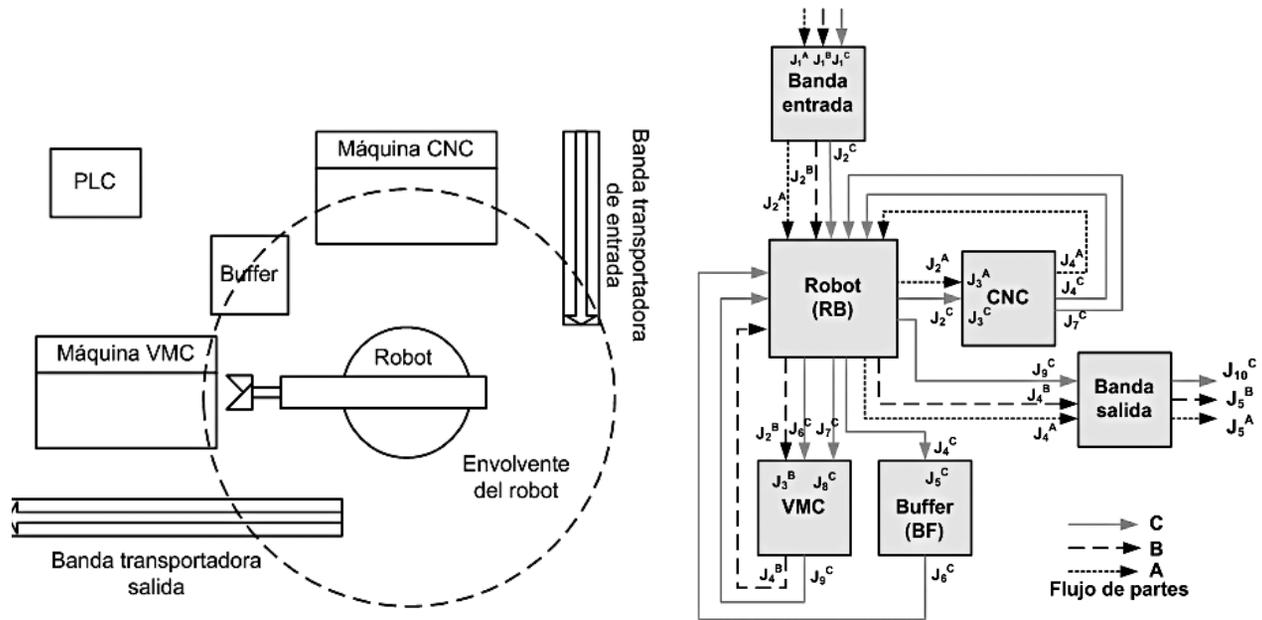


Figura 2. a) Distribución física de las máquinas. b) Flujo de las partes dentro de la celda.

Regla 1: Establecer nodos de control.

De la Red de Petri de la celda, por cada nodo que representa un recurso, que sus transiciones posteriores estén en conflicto, se establecerán lugares de control que tenga un arco apuntando a la transición. El nodo lugar de control será llamado $u_{D_{ki}}$ donde k es el tipo de parte y el índice i representa el número consecutivo de la tarea asociada a la parte k . Entonces, los nodos de control $u_{D_{ki}}$ y las transiciones en conflicto se establecen como:

$$\exists u_{D_{ki}} \forall x_c, u_{D_{ki}} = \bullet x_c$$

$$x_c = \{x \in T \mid \bullet x \cap R \neq \emptyset \wedge |(\bullet x) \bullet| > 1\} \quad (4)$$

Regla 2: Establecer marcados mínimos para las EC:

El marcado de una espera circular simple C_i , debe ser mayor a cero para que no exista bloqueo.

$$m(C_i) > 0 \quad (5)$$

Cuando la espera circular es compuesta, el marcado debe ser siempre mayor a uno para que no exista bloqueo.

$$m(C_i) > 1 \quad (6)$$

Si el marcado de C_i llega a ser uno, y si el último recurso disponible es el recurso compartido, entonces este se podrá utilizar si ejecuta una tarea que libera al menos un recurso dentro de la misma espera circular; haciendo que el marcado de C_i cumpla la ecuación (5) una vez finalizada la tarea $J(r_s)$.

$$IF(m(C_i) = m(r_s) = 1)$$

$$THEN J(r_s) \Rightarrow m(C_i) = 0 \quad (7)$$

Si el marcado de una espera circular compuesta C_i , llega a ser dos, entonces los recursos disponibles deberán pertenecer a la misma espera circular simple C_i . Y sólo se utilizará en la ejecución de tareas que usen los recursos dentro de la espera circular disponible.

$$IF \left(\frac{(m(C_i) = 2 \Rightarrow m(C_i) > 1)}{\wedge m(r_{ns_{C_i}}) = 0} \right) THEN J(C_i) \quad (8)$$

Regla 3: Establecer prioridades entre tareas secuenciales dentro de una misma espera circular

Para la espera circular C_i , si existen tareas previas $\bullet \bullet (J_l^A(r_s))$, $\bullet \bullet (J_m^A(r_s))$ a las tareas $J_l^A(r_s)$, $J_m^A(r_s)$, donde la tarea m es posterior a la tarea l , entonces

se asignará el recurso a la tarea I, para cumplir las ecuaciones (5),(7).

$$\begin{aligned}
 & \text{IF} \left(\begin{array}{l} m(C_i) > 1 \\ \wedge m(\bullet\bullet(J_i^A(r_s))) = 1 \\ \wedge m(\bullet\bullet(J_m^A(r_s))) = 1 \end{array} \right) \\
 & \text{THEN } u_{D_{A_i}} = 1 \Rightarrow J_i^A(r_s) = 1 \\
 & \text{IF} \left(\begin{array}{l} m(C_i) = m(r_s) = 1 \\ \wedge m(\bullet\bullet(J_i^A(r_s))) = 1 \\ \wedge m(\bullet\bullet(J_m^A(r_s))) = 1 \end{array} \right) \\
 & \text{THEN } u_{D_{A_m}} = 1 \Rightarrow J_m^A(r_s) = 1 \quad (9)
 \end{aligned}$$

Donde: $(\bullet\bullet(J_i^\Pi(r_s))) \cap J \neq \emptyset$

Regla 4: Establecer prioridades de producción, cuando dos o mas productos compiten por un recurso

Cuando dos o más tareas de piezas distintas compiten por el mismo recurso, se asignará prioridad de acuerdo a la demanda de producción. En el siguiente

seudocódigo, la prioridad la poseen la piezas tipo A, luego B y por último C.

$$\begin{aligned}
 & \text{IF} \left(\begin{array}{l} m(C_j) > 1 \\ \wedge m(\bullet\bullet(J_i^A(r_s))) = 1 \\ \wedge \left(\begin{array}{l} m(\bullet\bullet(J_m^B(r_s))) \\ \vee m(\bullet\bullet(J_n^C(r_s))) \end{array} \right) = 1 \end{array} \right) \\
 & \text{THEN } u_{D_{A_i}} = 1 \Rightarrow J_i^A(r_s) = 1 \\
 & \text{ELSE IF} \left(\begin{array}{l} m(C_j) > 1 \\ \wedge m(\bullet\bullet(J_m^B(r_s))) = 1 \\ \wedge m(\bullet\bullet(J_n^C(r_s))) = 1 \end{array} \right) \\
 & \text{THEN } u_{D_{B_m}} = 1 \Rightarrow J_m^B(r_s) = 1 \\
 & \text{ELSE IF} \left(\begin{array}{l} m(C_j) > 1 \\ \wedge m(\bullet\bullet(J_n^C(r_s))) = 1 \end{array} \right) \\
 & \text{THEN } u_{D_{C_n}} = 1 \Rightarrow J_n^C(r_s) = 1 \quad (10)
 \end{aligned}$$

Política de despacho

el primer paso para aplicar las reglas propuestas es determinar las esperas circulares de la celda de

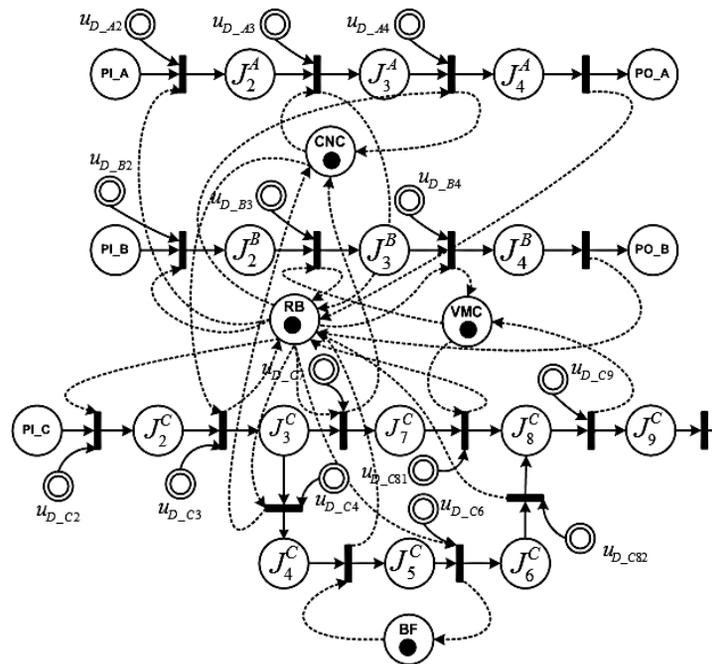


Figura 3. Red de Petri con los nodos lugar de control. Los nodos de control se representan como círculos dobles y cada uno tienen un arco dirigido a un nodo transición en conflicto x_c .

manufactura descrita anteriormente. Las esperas circulares simples y cíclicas se obtuvieron de la observación directa de RdP de la figura 3.

$$EC = \begin{pmatrix} \text{TBI} & \text{TB2} & \text{RB} & \text{CNC} & \text{VMC} & \text{Bf} & \\ 0 & 0 & 1 & 1 & 0 & 0 & C_1 \\ 0 & 0 & 1 & 0 & 1 & 0 & C_2 \\ 0 & 0 & 1 & 1 & 1 & 0 & C_3 \\ 0 & 0 & 1 & 0 & 0 & 1 & C_4 \\ 0 & 0 & 1 & 1 & 0 & 1 & C_5 \\ 0 & 0 & 1 & 0 & 1 & 1 & C_6 \\ 0 & 0 & 1 & 1 & 1 & 1 & C_7 \end{pmatrix} \quad (1)$$

El siguiente algoritmo coloca marcas en los lugares de control, con la finalidad de evitar los bloqueos de la red, y a su vez buscando maximizar los productos procesados. A cada nodo de control se asignará un "token", siguiendo los lineamientos anteriores. El pseudocódigo utiliza las esperas circulares simples para manejar los bloqueos de la red. Observe que la RdP de la figura 3, se controla llevando control de las esperas circulares simples, las cuales son: $\{C_1, C_2, C_4\}$. Las esperas circulares del CMF se muestra en ecuación (11).

```

WHILE (true)

//Código para el manejo del marcado de los nodos
de control

    uD_A2=0, uD_A3=0, uD_A4=0
    uD_B2=0, uD_B3=0, uD_B4=0
    uD_C2=0, uD_C3=0, uD_C4=0
    uD_C6=0, uD_C7=0, uD_C81=0
    uD_C82=0, uD_C9=0,

    IF (m(C1)>1 ∧ m(••(J2A(RB)))> 0) THEN uD_A2=1
    IF(m(CNC)=1 ∧ m(••(J3A(CNC)))=1) THEN uD_A3=1
    IF (m(RB)=1 ∧ m(••(J4A(RB)))=1) THEN uD_A4=1
    IF (m(C2)>1 ∧ m(••(J2B(RB)))> 0) THEN uD_B2=1
    IF (m(VMC)=1 ∧ m(••(J3B(VMC)))=1) THEN uD_B3=1
    IF (m(RB)=1 ∧ m(••(J4B(RB)))=1) THEN uD_B4=1
    IF (m(C1)>1 ∧ m(••(J2C(RB)))> 0) THEN uD_C2=1
    IF(m(CNC)=1 ∧ m(••(J3C(CNC)))=1) THEN uD_C3=1
    IF (m(C4)>1 ∧ m(VMC)=0 ∧ m(••(J4C(RB)))=1)
        THEN uD_C4=1

```

```

    IF (m(C2)>1 ∧ m(••(J6C(RB)))=1) THEN uD_C6=1
    IF (m(C2)>1 ∧ m(••(J6C(RB)))=1) THEN uD_C7=1
    IF (m(VMC)=1 ∧ m(••(J8C(VMC)))=1) THEN uD_C81=1
    IF (m(VMC)=1 ∧ m(••(J8C(VMC)))=1) THEN uD_C82=1
    IF (m(RB)=1 ∧ m(••(J9C(RB)))=1) THEN uD_C9=1

//Código para despacho de tareas
FOR k = A to C
    FOR i = 1 to SizeOfSequence(Jk)
        IF (uD_ki=1 ∧ uD_ki•=1) THEN Jik(ri)=1
    END FOR
END FOR
ENDWHILE

```

Donde: $u_{D_{ki}} \bullet \cap T \neq \emptyset$, se refiere a la habilitación del nodo transición k_i .

Simulación

La política de control para la celda de manufactura, se simuló en el software Petri.Net, desarrollado por la Universidad de Zagreb. Se procedió a ejecutar mil pasos de simulación, la llegada de las piezas tipo A, B y C se seleccionó en forma aleatoria para buscar una mayor diversidad de secuencias de piezas de entrada a la celda. Se observó numerosos conflictos en la solicitud de los recursos. El oscilograma de la simulación sin controlador se muestra en la figura 4; por ejemplo el robot (fila RB), en los pasos de simulación 5, 7, 25, 27, 29, 31 muestran valores negativos, esto indica que se solicita el recurso más veces de lo disponible. Además, el marcado en los nodos de salida se muestra con valores bajos y acotados desde los primeros pasos de simulación, esto es indicativo que la CMF está bloqueada, por ejemplo el nodo PO_A indica que se elaboraron solamente dos productos tipo A, PO_B cuatro tipo B y PO_C seis tipo C respectivamente.

Por otra parte, cuando se simula la RdP con el controlador, el oscilograma de la figura 5 muestra que no existieron ni conflictos ni bloqueos. Los nodos de salida muestran muestra que se procesaron 62, 89 y 49 productos tipo A, B, C respectivamente.

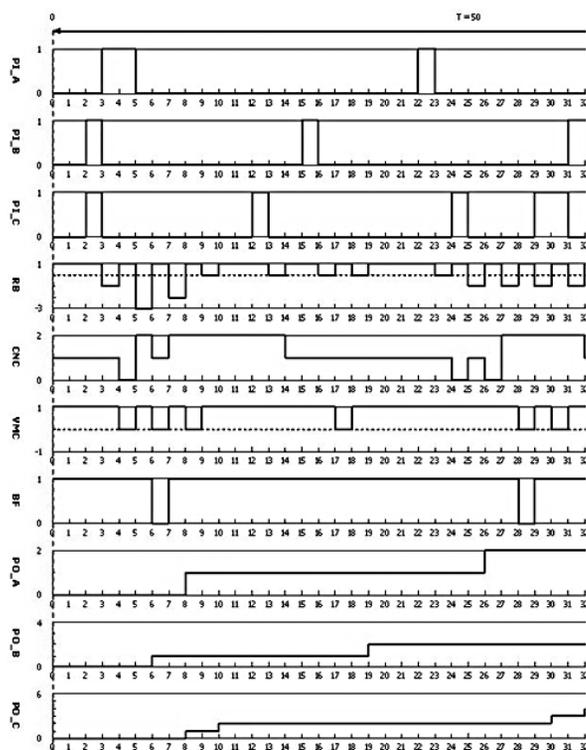


Figura 4. Oscilograma de las colas de entrada y salida de las piezas A, B, C, Así como, la utilización de las máquinas y buffer de la celda.

Analisis de resultados

Se desarrolló un algoritmo de control, en función de los lineamientos propuestos. El algoritmo resolvió las situaciones de conflicto y bloqueo, no fue posible evaluar su rendimiento en el tiempo, ya que el simulador utilizado (Petri.Net) no soportó las RdP temporizadas de libre escogencia, por tanto el algoritmo de control fue probado únicamente con la simulación de los eventos, por lo que se requiere probar la generalidad de las reglas de control, en otras celdas, así como probar la validez de las reglas cuando existen duraciones de tareas predefinidas.

Por otra parte, se requiere determinar la mejor propuesta de algoritmo, ya que este algoritmo no es el único. Existen otros cinco posibles algoritmos de control de acuerdo a la prioridad asignada a las piezas, ya sea: ACB, BAC, BCA, CAB, CBA. La escogencia de la prioridad se asocia al método de producción seleccionado, "push" o "pull":

- Tasa de llegada $\lambda_{i_{II}}$ o tasa de salida $\lambda_{o_{II}}$ de cada una de las piezas respectivamente.
- Duración de las tareas para el procesamiento de las piezas, $J^{\Pi} = \{J^A, J^B, J^C\}$.

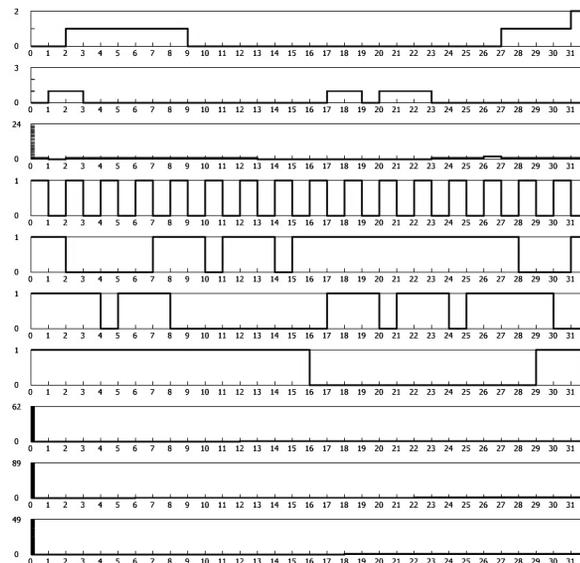


Figura 5. Oscilograma para la RdP controlada con el algoritmo propuesto. Se observa que los conflictos y bloqueos se han eliminado.

Finalmente, es necesario que el controlador de la celda pueda manejar la ocurrencia de las faltas y las fallas, esto debido a que estos eventos modifican la dinámica del sistema y si el controlador no modifica sus acciones de control, estas no evitarán que la CMF se bloquee. Por tanto, se requiere incorporar reglas para manejo de faltas y fallas, así como incorporar políticas de mantenimiento.

Reconocimientos

Se agradece a la Vicerrectora de Investigación y Extensión, del Instituto Tecnológico de Costa Rica por el apoyo brindado al proyecto VIE 5402-1341-1101.

Referencias

- Ballal, P., Lewis, F., Mireles, J. & Sreenath, K. (2007). Deadlock avoidance for free choice multi-reentrant flow lines: Critical siphons and critical subsystems. *IEEE Proc. Mediterranean Conf. Control & Automation*.
- Ballal, P.M. (2008). *Decision and Control of Distributed Cooperative Systems*. Ph.D Thesis, University of Texas, Arlington.
- Bogdan, S., Lewis, F.L., Kovacic, Z., Gurel, A. & Stajdohar, M. (Sept. 2002). An Implementation of Matrix-Based Supervisory Control of Flexible Manufacturing Systems. *IEEE Transactions on Control Systems Technology* 10(5): 709-725.

- Bogdan, S., Kovacic, Z., Smolic-Rocak, N. & Birgmajer, B. (Dic. 2004). A matrix approach to an FMS control design. *IEEE Robotics & Automation Magazine* 11(4): 92-109.
- Bogdan, S., Lewis, F.L., Kovacic, Z. & Mireles, J. (2006). *Manufacturing System Control Design: A Matrix-based Approach*. London: Springer-Verlag.
- Chang, T.C., Wysk, R.A. & Wang, H.P. (1998). *Computer Aided Manufacturing*. New Jersey: Prentice Hall.
- Groover, M. (1987). *Automation, Production System and Computer Integrated Manufacturing*. New Jersey: Prentice-Hall.
- Hruz, B. & Zhou, M. (2007). *Modelling and Control of Discrete-event Dynamic Systems - (Advanced textbooks in control and signal processing)*. London: Springer-Verlag.
- Huang, H.H., Lewis, F.L. & Tacconi, D. (1996). Deadlock analysis using a new matrix-based controller for reentrant flow line design. *Proceedings IEEE IECON 22nd International Conference* (pp. 463-468). Taiwán: IEEE Industrial Electronics, Control, and Instrumentation.
- Huang, H.H. (2001a). Matrix Controller Design and Deadlock Analysis of Automated Manufacturing Systems. Part 1: Designing the Matrix-Based Controller. *The International Journal of Advanced Manufacturing Technology* 18(6): 434-447.
- Huang, H.H. (2001b). Matrix Controller Design and Deadlock Analysis of Automated Manufacturing Systems. Part 2: Deadlock Avoidance Policy. *The International Journal of Advanced Manufacturing Technology* 18(7): 490-501.
- Lewis, F.L., Gurel, A., Bogdan, S., Doganalp, A. & Pastravanu, O.C. (1998). Analysis of Deadlock and Circular Waits Using a Matrix Model for Flexible Manufacturing Systems. *Automatica* 34(9): 1083-1100.
- Mireles, J. & Lewis, F.L. (Dic. 2001). Intelligent Material Handling: Development and Implementation of Matrix-Based Discrete-Event Controller. *IEEE Transactions on Industrial Electronics* 48(6): 1087-1097.
- Mireles, J. & Lewis, F.L. (2002). Deadlock analysis and routing on free-choice multipart reentrant flow lines using a matrix-based discrete event controller. *Proceedings of the 41st IEEE Conference on Decision and Control 1*, pp. 793-798.
- Murata, T. (1989). Petri nets: Properties, Analysis and Application. *Proceeding of the IEEE* 77(4): 541-580.
- Murillo-Soto, L.D. (2010). Simulación de un sistema de manufactura flexible con redes de Petri coloreadas. *Tecnología en Marcha* 23(1): 47-62.
- Peng, S. & Zhou, M. (2003). Sensor-based stage Petri net modelling of PLC logic programs for Discrete-event control design. *International Journal of Production Research* 41(3): 629-644.
- Silberschatz, A. & Galvin, P.C. (1999). *Sistemas Operativos*. 5 ed. México: Addison Wesley.
- Silva, M. (1998). Petri nets and Production Systems. LNCS: *Lecture on Petri Nets Application II* (1492): 85-124.
- Tacconi, D.A. & Lewis, F.L. (Oct. 1997). A new matrix model for discrete event systems: application to simulation. *IEEE Control Systems Magazine* 17(5): 62-71.
- Zhou, M. & Venkatesh, K. (2000). *Modeling, Simulation and Control of Manufacturing Systems: A Petri nets Approach*. New Jersey: World Scientific.
- Zhou, M. & Fanti, M.P. (Eds.). (2004). *Deadlock Resolution in Computer Integrated Systems*. New York: Merceel Dekker.