

# Sistema multi-agente para control de manufactura flexible basado en la teoría de restricciones

Jorge Martínez Muñoz<sup>1</sup>, J. Juan Guerra Vásquez<sup>1</sup>

<sup>1</sup> Centro de Investigación en Computación del Instituto Politécnico Nacional  
{george, jguerrav}@correo.cic.ipn.mx

**Resumen.** Los Sistemas Multi-Agente, representan parte de los esfuerzos recientes en la solución de problemas que requieren planificación emergente y distribuida entre los participantes. En este trabajo se presenta el diseño e implementación de un sistema orientado al control flexible de manufactura y basado en la Teoría de Restricciones. Dicha Teoría plantea una nueva forma de concebir a la empresa y en especial, a la parte de manufactura. Se presentan además, resultados obtenidos de la experimentación en una plataforma abierta para desarrollo de agentes (JADE).

## 1 Introducción

Las tendencias actuales en el área de manufactura han dejado la producción masiva para dar lugar a la producción personalizada. Las empresas deben adaptarse a la demanda especializada de cada consumidor. Para resolver esta situación, se requiere de sistemas con capacidad de adaptación dinámica y flexible. En el presente trabajo se describe la implementación de la Teoría de Restricciones (TOC) en un Sistema Multi-Agente (SMA) orientado al control flexible de manufactura. Con ello se pretende experimentar con la fusión de dos ramas del conocimiento, Administración de la Producción e Inteligencia Artificial, en la solución de un problema: calendarización dinámica de partes/componentes en la manufactura de un producto.

## 2 Definición del problema

El problema se toma de un caso de estudio general propuesto por Jennings en [1]. Se trata de un sistema de manufactura que comprende un conjunto de máquinas (M1-M9) para transformación o ensamble de componentes. El grupo de máquinas es alimentado por tres almacenes de materia prima. Con respecto al problema original, se aplicaron dos modificaciones: cuando una máquina falla, el componente que se encontraba procesando es desechado en lugar de mandarlo a reproceso y se agregó explícitamente un nuevo tipo de almacén para productos terminados. La configuración del problema se muestra en la Figura 1.

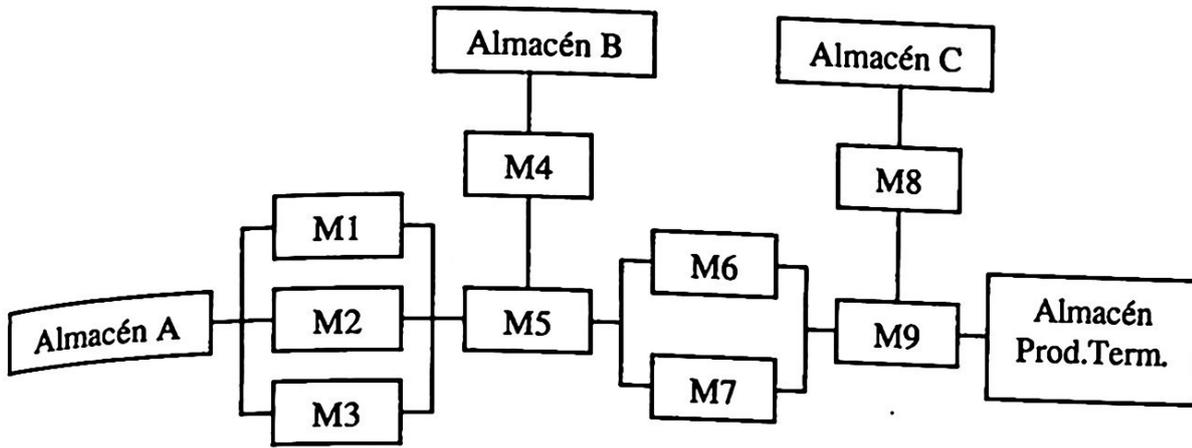


Fig. 1. Sistema Flexible de Manufactura.

### 3 Teoría de restricciones

Esta teoría se propuso como una alternativa cuya intención principal es la contemplación de una empresa, no como componentes aislados que colaboran entre sí, sino como un todo que debe ser optimizado. Una de las aplicaciones en un entorno similar al problema que aquí se presenta está en [3].

Para resolver el caso de estudio, se aplicaron únicamente aquellos conceptos de la Teoría de Restricciones con mayor relación al tipo de problema. En un sistema de manufactura se presentan fenómenos que afectan la operación de la misma. Por una parte se tienen los eventos dependientes y son aquellos procesos que por la arquitectura de la línea, no pueden operar a menos que reciban la salida de procesos anteriores a ellos; por otro lado, están las fluctuaciones estadísticas, como ejemplo se pueden mencionar las máquinas M3, M4 y M8 del caso de estudio. Ellas tres tienen capacidad de realizar dos tipos distintos de operaciones, de antemano es difícil pronosticar cuál operación ejecutarán, lo cual afecta el promedio de tiempo en línea que acumulen cada una de ellas. Cualquier retraso en un proceso anterior se acumula en los procesos subsecuentes.

Es importante aprender a distinguir entre dos tipos de recursos: Cuellos de Botella y No Cuellos de Botella. Ambos están definidos con respecto a su capacidad, el primero de ellos es aquel cuya capacidad es igual o menor a la demanda que hay de él, mientras que el segundo tiene capacidad sobrada según la demanda. Los Cuellos de Botella indican cuándo alimentar al sistema con más inventario.

Típicamente, las empresas cometen el error de balancear la capacidad con la demanda. Lo importante es balancear el flujo de producto por la línea de producción con la demanda del mercado. El hecho de mantener a las máquinas trabajando todo el tiempo no es señal de eficiencia, finalmente, lo único que se logra es acumulación de inventario en los almacenes de productos en proceso. De aquí se obtiene la consideración más importante en el modelo desarrollado: detectar por medio de un análisis cuál es la máquina Cuello de Botella, que a su vez trabajará como el calendarizador de nuevos productos. No importa cuán rápido trabajen las otras

máquinas, eventualmente todas se verán obligadas a sincronizar sus tiempos de operación con los números que maneja el Cuello de Botella.

#### 4 Diseño del Sistema Multi-Agente

Cada elemento presente en el modelo de producción estudiado, está representado por un agente. Para cada almacén de materia prima (A, B, C), existe un agente que corresponde y que alimenta de manera inicial al sistema. Por su parte, cada una de las materias primas y las partes resultantes de su respectivo ensamblaje, representadas por agentes Componentes; la meta de estos últimos es llegar hasta el almacén de producto terminado.

Adicionalmente existe un agente en el sistema por cada máquina presente en el modelo, en este caso se identificaron nueve agentes máquina (M1-M9). La función de estos agentes es la de prestar sus servicios a los agentes parte/componente que los soliciten. Dichos servicios consisten de las operaciones que son capaces de realizar, en un modelo de producción algunas máquinas pueden tener operaciones en común. La máquina más lenta de acuerdo a los parámetros que definen sus operaciones, define el cuello de botella.

El almacén de producto terminado también tiene su representante. En el sistema este agente cumple la función de llevar el inventario de productos exitosamente procesados, así como de informar la demanda de los mismos de acuerdo a criterios preestablecidos (p.ej. prioridades de manufactura).

El proceso de producción inicia con la liberación de una materia prima (parte/componente) por parte de algún almacén. Esta es una tarea periódica en función de la velocidad de operación del cuello de botella. Al ingresar la materia prima al sistema, solicita al almacén de producto terminado, le informe cuál producto falta. A continuación, por medio de un servicio externo, se le proporciona una secuencia de operaciones que integran el proceso tecnológico aplicable al componente.

Cuando en alguna parte del proceso se requiere de un ensamble, el agente parte/componente solicita al almacén correspondiente la creación de una pareja de componentes. Con ello, al nacer la pareja, ya tiene conocimiento del producto que formará e inmediatamente solicita información de las operaciones que requiere para formar dicho producto. Ambos agentes componentes negocian con las máquinas respectivas para sincronizar su arribo simultáneo al punto de ensamble. Cuando la pareja de agentes es ensamblada, muere y da lugar al nacimiento de un nuevo agente compuesto con las características de ambos. Si un agente componente que integra la pareja es desechado durante el procesamiento de una máquina, éste notifica la falla al otro agente perteneciente a su pareja y muere. El sobreviviente solicita a su vez al almacén respectivo, la creación de una refacción con las mismas características que el elemento fallido.

## 5 Implementación

El sistema se implementó en el entorno para desarrollo de agentes JADE [4]; dicha plataforma está basada en el lenguaje de programación Java. Se codificaron cuatro tipos de agentes: Almacén de Materia Prima (Warehouse), Almacén de Producto Terminado (FPWarehouse), Máquina (Machine) y Parte/Componente (Component). Adicionalmente, se desarrollaron dos agentes de utilería. El primero es una máquina de inferencias (WrapperProlog) que informa a los componentes el proceso tecnológico por el cual deben pasar. Se trata de un *wrapper* al intérprete SWI-Prolog. El segundo consiste de un servicio para presentación de resultados, análisis estadístico y gráfico en tiempo real (WrapperGraph). Éste se implementó por medio de un agente *wrapper* a la hoja electrónica de cálculo SCalc de OpenOffice [5].

Para comunicarse entre sí, los agentes usan protocolos definidos por los estándares FIPA[6], específicamente: FIPA-Request, FIPA-Query y FIPA-ContractNet; éste último es de uso intensivo en la negociación entre componentes que desean ser procesados y máquinas que ofrecen el tipo de operación requerida.

La implementación se enfocó fuertemente en el aspecto de que las interacciones entre agentes se basan en la solicitud y prestación de servicios, es decir, cada agente tiene la capacidad de solicitar y prestar servicios a los demás agentes del sistema.

La figura 2 muestra información adicional sobre el tipo de hardware y sistemas operativos que dieron soporte a la implementación y experimentación.

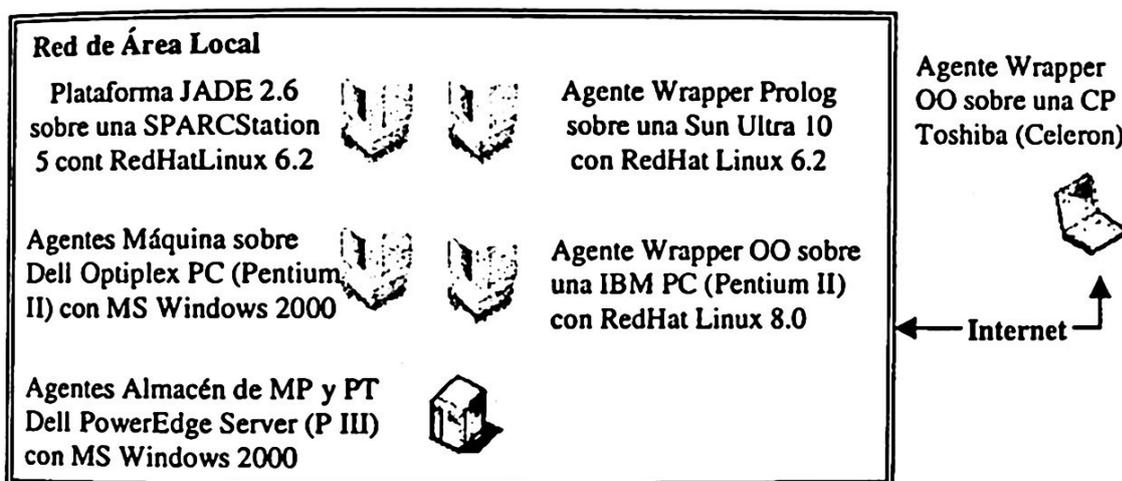


Fig.2. Diagrama de despliegue para la implementación y experimentos realizados.

El agente que inicia la simulación y se encarga de controlar la calendarización de la producción es el agente máquina Cuello de Botella. Este agente solicita al almacén correspondiente, la liberación de un agente parte/componente que desata el proceso para la generación de un producto: esta actividad es llevada a cabo periódicamente en función a la velocidad de operación del cuello de botella

Una vez que el agente *Component* (A,B,C) ha sido liberado por un agente *Warehouse*, procede a definir cuál es el producto final del que deberá formar parte. Esta decisión se toma considerando criterios tales como una producción equilibrada o

bien productos ordenados por prioridad. Para lo anterior, el agente *Component* solicita dicha información al agente almacén de productos terminados (*FPWarehouse*), que es responsable de suministrar esa información (artículo que necesita ser producido). Una vez que el agente *Component* tiene la información del producto a formar, necesita conocer la operación que requiere para poder transformándose en un producto terminado. Esto se logra solicitando el servicio inferencia al agente *wrapper* respectivo. La respuesta que obtiene es una lista de operaciones que deben ser aplicadas sobre el agente parte/componente. continuación se presentan una serie de negociaciones entre componentes y máquinas bajo el protocolo Fipa-Contract-Net; cada componente elige a la máquina cuya propuesta represente el menor tiempo de espera para ingresar a procesamiento.

Otra operación que el agente componente realiza, es preguntar al agente *wrapper* de software de inferencia, con cuál tipo de agente parte/componente debe ser ensamblado para generar un componente compuesto. Con esta información se solicita al almacén correspondiente la liberación del componente requerido y se estima el tiempo que tardarán los integrantes de la pareja en llegar al punto de ensamblaje. Cuando uno de los dos agentes pueda llegar antes que su pareja al punto de ensamblaje, procede a renegociar su ingreso a procesamiento con la(s) máquina(s) intermedia(s) a fin de que ambos arriben al mismo tiempo a dicho punto.

Después de que dos agentes parte/componente han sido ensamblados, mueren, lo que permite crear un nuevo tipo de agente parte/componente con las características de los anteriores (ej. mueren A y B para formar un componente AB). El nuevo agente conserva como objetivo formar el mismo producto que sus integrantes originales; al igual que ellos, requiere de informarse cuáles son las operaciones que deben ser ejecutadas sobre él y en consecuencia, debe negociar con las máquinas respectivas.

Las máquinas pueden fallar, cuando eso sucede el agente parte/componente que está siendo procesado en el momento de la falla, se desecha y muere. En el sistema cuando un agente parte/componente es notificado de la falla en su procesamiento, notifica la situación a su agente pareja. Éste último solicita la creación de un nuevo compañero con las características del anterior y procede a esperarlo en el búffer de salida de la máquina mas cercana al punto de ensamblaje. Aquí se presentan dos situaciones, el primero es que la creación del nuevo agente pareja (refacción) no repercute en el rendimiento del sistema y pueda crearse y procesarse sin problemas hasta ensamblarse con su respectiva pareja. El segundo caso es que la creación de dicho agente de refacción sí repercute en la planificación de las operaciones de la máquina cuello de botella, lo cual implicará que el agente parte/componente que esta en algún buffer espere por más tiempo ser ensamblado. Éste segundo caso se presenta generalmente cuando el agente parte/componente desechado se encontraba próximo a formar un producto terminado.

## 6 Experimentos y resultados

Los experimentos se enfocaron sobre dos objetivos principales: estudiar el comportamiento del sistema bajo el modelo de administración de producción propuesto y, evaluar el desempeño de la implementación sobre la plataforma seleccionada (JADE).

Con respecto al primer punto, los experimentos comprendieron los agentes para nueve máquinas (M1-M9), tres almacenes de materia prima (A, B, C) y un almacén de producto terminado (FPWH). Se consideraron un total de nueve operaciones en el proceso de producción (O1-O9). Las máquinas M1 y M2 realizaron las operaciones O1 y O2 respectivamente, la máquina M3 las mismas operaciones que M1 y M2; a M4 le correspondieron las operaciones O3 y O4. Las máquinas M6 y M7 desempeñaron la misma operación (O6), mientras que M8 tuvo la capacidad de desempeñar las operaciones O7 y O8. Las máquinas que llevaron a cabo las operaciones de ensamblaje O5 y O9 fueron M5 y M9 respectivamente.

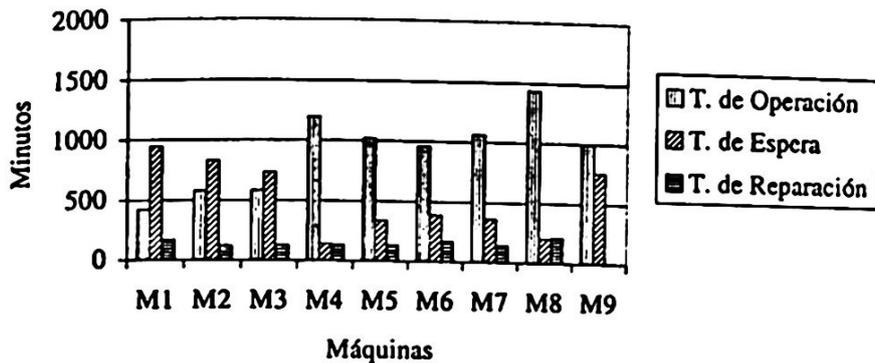


Fig. 3. Operación de las máquinas.

La figura 3 muestra los resultados correspondientes a los tiempos de operación de las máquinas. M1, M2, M3 presentan un tiempo de espera mayor al tiempo de operación debido a que tienen operaciones comunes y el trabajo de las tres en conjunto satisfacen los requerimientos del tiempo simulado. Algo semejante se presenta con las máquinas M6 y M7. M8 presenta un considerable tiempo de espera, lo cual es inadmisibles según la Teoría de Restricciones. M8 debería ser la máquina que nunca se detiene, a menos que se deba a una falla. Se detectó que fallas en la ejecución en algunas máquinas, inciden sensiblemente en los tiempos del recurso cuello de botella, sobre todo cuando el componente desechado involucra componentes B o AB. El tiempo de espera en M8 es también consecuencia de los procesos tecnológicos seleccionados para el caso de estudio.

El tiempo de espera de una máquina está constituido por retrasos acumulados en fallas de máquinas que le anteceden y el tiempo que pierde al ajustarse a la velocidad del cuello de botella.

Se observó que la calendarización del proceso de producción en función a la velocidad del cuello de botella, permite contar con un sistema fluido y con un mínimo de inventario en proceso. Éste se da como una consecuencia de fallas en las máquinas, esto es, un componente fallido implica que su compañero exitoso se quedará en espera del arribo de una nueva refacción.

El segundo tipo de experimentos se dividió en tres configuraciones: todos los agentes ejecutándose en una misma computadora (el servidor Dell PowerEdge), los agentes distribuidos en una red local (Figura 2) y el agente wrapper OpenOffice ejecutándose en una computadora portátil desde otro país. Para poder presentar los resultados al usuario, este agente se comunica de manera intensiva con el resto del

sistema. Por ello se decidió realizar dos experimentos para cada configuración, con sin el agente wrapper.

En total se simularon seis, doce y veinticuatro horas de operación que se pueden apreciar en la Figura 4. Cuando el agente wrapper está ausente, los tiempos que tarda la simulación en terminar son muy similares. Sin embargo, si el agente wrapper OpenOffice se encuentra presente, llama la atención que los experimentos en red local tardaron menos tiempo en ejecutarse que los experimentos centralizados. Aquí tiene una consecuencia natural de la distribución de tareas. Finalmente, se tiene que experimento con la computadora portátil en otro país, requirió de mayor tiempo que las otras dos modalidades para completarse.

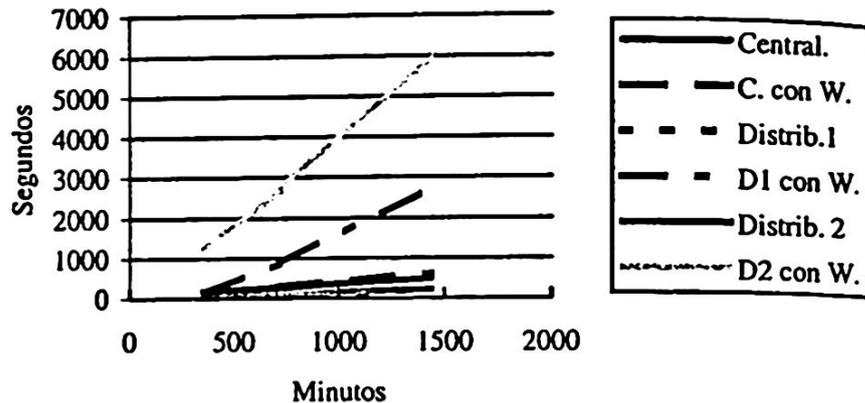


Fig. 4. Relación Tiempo Simulado - Tiempo Real de los experimentos.

## 7 Conclusiones

En apariencia, la Teoría de Restricciones parece una idea contrapuesta a la filosofía de los Sistemas Multi-Agente. En esta última, se insiste en la minimización de dependencias entre las entidades participantes en el sistema con el fin de proteger la autonomía. Sin embargo, no hay garantía de que el comportamiento global, resultado de las interacciones, sea el mejor que se pudiera esperar. En el desarrollo del presente trabajo, se mantiene la necesidad de un planteamiento general bajo el cual los Agentes colaboren (o compitan), sin obligarlos a establecer dependencias que impidan el funcionamiento autónomo.

El motor de las interacciones entre los agentes es la prestación y solicitud de servicios que en un momento dado necesitan, también resalta el uso de distintos protocolos de comunicación en función al tipo de agente con que pretenden interactuar.

Finalmente, dos de las mejoras que se proponen para extender la funcionalidad del modelo son: minimizar el tiempo de espera para el cuello de botella de acuerdo a las estrategias de control y detectar de manera automática a dicho recurso. Actualmente se le especifica de manera explícita al sistema, cuál es la máquina más lenta de todo el grupo.

## Referencias

Jennings, NR. On agent-based software engineering. *Artificial Intelligence*, 117: 277-296, (2000).

Goldratt, Eliyahu M. *La Meta*. Ediciones Castillo. (2002).

Baker A.D., A Survey of Factory Control Algorithms which Can be Implemented in a Multi-Agent Heterarchy: Dispatching, Scheduling, and Pull, *J. of Manufacturing Systems*, 17(4): 297-320, (1998).

Java Agent DEvelopment Framework <<http://jade.cselt.it>>

OpenOffice Suite <<http://www.openoffice.org>>

Foundation for Intelligent Physical Agents <<http://www.fipa.org>>