

# Planificación de la producción en talleres de cilindros de laminación mediante lógica borrosa\*

Ignacio González López<sup>1</sup>, Adenso Díaz Fernández<sup>2</sup>, Javier Tuya González<sup>3</sup>, María José Suárez Cabal<sup>4</sup>,

<sup>1</sup> Ingeniero Industrial (ETSII Gijón. Campus de Viesques. 33204 Gijón, ilopez@lsi.uniovi.es)

<sup>2</sup> Profesor Titular (ETSII Gijón. Campus de Viesques. 33204 Gijón, adenso@etsiig.uniovi.es)

<sup>3</sup> Profesor Titular (ETSII Gijón. Campus de Viesques. 33204 Gijón, tuya@etsiig.uniovi.es)

<sup>4</sup> Profesora Asociada (ETSII Gijón. Campus de Viesques. 33204 Gijón, cabal@correo.uniovi.es)

## RESUMEN

*El desarrollo de este modelo incorpora la lógica borrosa a la gestión del taller de cilindros de la factoría avilesina de Aceralia, con el objetivo primordial de agilizar el proceso de toma de decisiones en lo relativo a las prioridades de producción, desviando parte de la responsabilidad del maestro de taller hacia la dirección sin destinar más recursos que los necesarios en el momento de definición del sistema.*

### 1. Introducción.

Sectores industriales maduros en Europa, como la siderurgia, han sufrido serias correcciones en los últimos años para lograr mantenerse de forma competitiva en el mercado. Tras los ajustes de plantilla sufridos en la década de los ochenta, las empresas del sector siderometalúrgico han comenzado a invertir de manera importante en I+D buscando la mejora continua de sus procesos y métodos, adaptando sus sistemas logísticos y productivos a los requerimientos actuales. En concreto, el modelo del que se deriva este trabajo forma parte de un proyecto que incluye seis modelos jerarquizados que abarcan toda la gestión de actividades en el Departamento de Cilindros de Aceralia-Avilés. Son los siguientes (figura 1):

- Modelo de Inventario: Es el modelo que se describe en este artículo.
- Modelo de Secuenciación: Devuelve como salida las rutas que cada cilindro debe llevar a través del taller de forma que se optimice la capacidad productiva del mismo. Es el segundo nivel del modelo de inventario, al que complementa con mayor detalle.
- Modelo de Mantenimiento: Indica cuándo debe ser revisada cualquiera de las máquinas u otros componentes de las instalaciones del taller. Junto con los dos anteriores formaría el modelo total de planificación en el taller, incorporando al sistema las perturbaciones asociadas al mantenimiento preventivo.
- Modelo de Compras: Determina el momento en que han de realizarse los pedidos de cilindros y a qué proveedor, de forma que no sólo se realice con el menor coste, sino que se cumpla con los criterios de la política de compras del taller, teniendo en cuenta que los tiempos de entrega en este caso pueden ser superiores a 5 ó 6 meses.
- Modelo de Gestión de Ampuestas: Teniendo en cuenta la capacidad disponible y las necesidades de ampuestas, determina el plan de inspección de éstas.

---

\* Investigación financiada a través del proyecto europeo CECA-99-7210PR148

- Modelo de Emparejamiento de Cilindros: Normalmente los cilindros tienen asignada una pareja para trabajar. En caso de que uno de ellos sufra una incidencia crítica, el modelo trata de asignarle el más adecuado de los disponibles según distintos criterios.

Como se observa en la figura 1, el modelo 1 es el modelo central del sistema y en él se basará este artículo. Se activa por eventos relacionados con la producción o cambios de política, recibiendo información de unos modelos y activando a otros que se asocian a él.

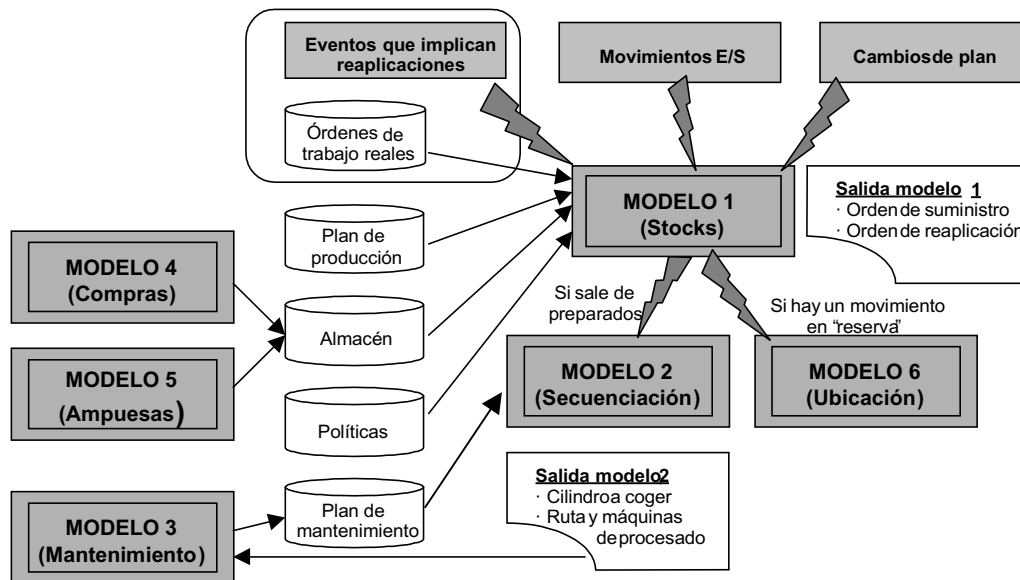


Figura 1: Esquema general de modelos del sistema

## 2. Planificación del taller.

El Taller de Cilindros de Aceralia-Avilés está dividido en dos áreas: equipos para trenes de laminación en caliente y en frío, con servicio para 10 trenes. En las cajas de los distintos trenes se necesitan cilindros de diversos tipos en función del producto final de laminación (“aca.tipo”). Existen en este momento más de 60 aca.tipos aunque la cifra varía según el momento y la cartera de clientes. Cada cierto tiempo los cilindros han de ser sustituidos y los retirados requieren mecanizado en el taller o en empresas exteriores antes de poder ser utilizados de nuevo. De esta forma, dado el gran número y variedad de cilindros que han de pasar por el taller, es necesario decidir el orden en que han de ser procesados, puesto que la disponibilidad de recursos es limitada y es importante tener el menor número posible de cilindros en circulación, consiguiendo de esta forma una disminución del stock con el consiguiente ahorro de espacio y dinero.

Las decisiones a tomar en el taller relacionadas con el modelo de inventario son de dos tipos:

- Procesado/suministro: Decidir cuál de los aca.tipos demandados en los trenes ha de ser mecanizado en primer lugar (establecer prioridades).
- Reaplicación: Decidir si cambiar el aca.tipo de un cilindro que ya está procesándose. Para ello deberán de alterarse las tareas pendientes así como actualizar la información en la base de datos del taller. Estas órdenes de reaplicación son soluciones de urgencia para solventar necesidades acuciantes de un determinado aca.tipo y evitar una ruptura del stock.

Hasta el momento de iniciarse el proyecto, estas decisiones eran tomadas por el maestro de taller basándose en su experiencia, estando en manos de los responsables las decisiones de carácter más agregado. La modernización del sistema contemplaba la utilización de la ingente cantidad de información relativa a los cilindros y las instalaciones recogida en la BD. Los responsables supervisarían para evitar errores, consistentes fundamentalmente en un sobredimensionamiento del stock de seguridad y en la utilización de reaplicaciones con una frecuencia superior a lo que sería deseable dado su carácter de “solución de urgencia”.

Puesto que no se podía suponer una calidad suficiente de la información de partida, fueron precisos varios meses de observación del proceso y de reuniones con los responsables del taller para conocer con exactitud los objetivos que deberían de ser cumplidos por este modelo. Esto, unido a un exhaustivo estudio de la base de datos, permitió la detección de aquellas variables que resultaban más relevantes en el proceso de toma de decisiones:

- Frecuencia de accidente: Debido a diversos problemas, existe un cierto número de cilindros que nunca llegan a laminar, y otros que proporcionan un rendimiento aceptable durante menos tiempo del normal. Estas incidencias no son igual de frecuentes en unos y otros aca.tipos.
- Importancia de tren/caja: Algunos trenes o cajas pueden ser más conflictivos en cuanto a incidencias o urgencias de pedidos.
- Desviación con la estimación de cilindros del aca.tipo a enviar a los trenes: Se utiliza cuando no existe suficiente información real suficiente, ponderando los resultados obtenidos.
- Desviación con el óptimo de cilindros en stock: Horas de taller necesarias para que el stock de cilindros de ese aca.tipo fuese óptimo.

De éstas, las dos primeras se hallan recogidas directamente en la base de datos, mientras que las otras necesitan ser calculadas. Para ello, se desarrolló un programa en lenguaje C que opera con distintas variables, siendo más de 20 las necesarias para calcular el óptimo de cilindros de cada aca.tipo y la desviación con el mismo (hasta el momento se suponía una cifra óptima sin tener en cuenta multitud de aspectos relevantes que podrían hacer que el óptimo real fuese mayor o menor que este valor). Se podría decir que en cuanto a planteamiento del problema, esta variable representaba el punto débil del sistema manual, mientras que un sistema fuzzy permitiría liberar de responsabilidad al maestro en la toma de decisiones semi-automatizándose el proceso (siempre precisará la supervisión del experto) con el conocimiento añadido de los responsables del taller.

### **3. Solución propuesta.**

Para este primer modelo se optó por la utilización de un sistema basado en la lógica difusa. Apoyándose en la teoría de conjuntos borrosos (en los que la pertenencia puede ser parcial), es una herramienta muy poderosa porque permite la simulación del conocimiento humano, que no se basa en exactitudes, sino en reglas de funcionamiento basadas en la experiencia y la observación de procesos [1]. Algunas otras ventajas que supone su utilización son su sencillez e intuitividad para el experto, su flexibilidad para adaptarse a los cambios sin grandes modificaciones y su tolerancia a la información imprecisa [2]. Es un sistema que no viene a sustituir al experto, sino a utilizar su conocimiento para ayudarlo en su tarea. Al basarse en el lenguaje natural, la aplicación del conocimiento del experto es prácticamente directa [3].

Como se apuntó en el apartado anterior, el modelo propuesto consta de dos bloques principales: el programa de obtención de datos desarrollado en C, y un sistema fuzzy que utiliza para cada aca.tipo sus cuatro valores de entrada para proporcionar un valor final que representa su prioridad de procesado en el taller.

El funcionamiento del modelo (figura 2) consiste básicamente en los siguientes pasos:

- Arranque: Ocurre por petición expresa del usuario (maestro de taller), periódicamente, o bien cada vez que algún cilindro sale del taller hacia alguno de los trenes (figura 1).
- Lectura y manipulación de datos: Lee y opera con múltiples datos de la BD, considerando las restricciones oportunas, para llegar a los cuatro datos por cada aca.tipo necesarios.
- Ejecución de la aplicación fuzzy: Para cada aca.tipo devuelve una cifra de prioridad (output).
- Orden de trabajo: Puede ser una orden de producción o de reaplicación.

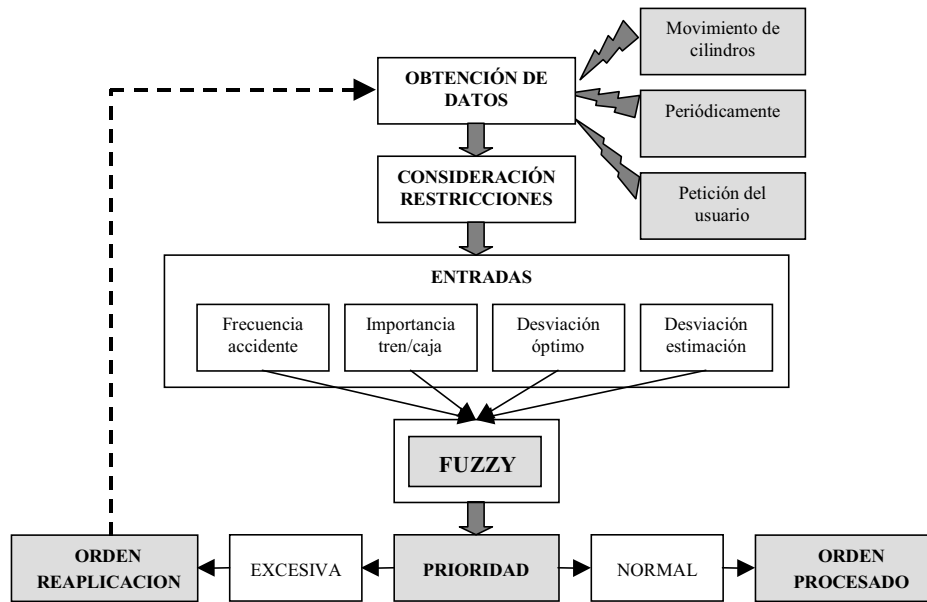


Figura 2: Esquema de funcionamiento del modelo

### 3.1. Obtención de datos.

Su aspecto más complejo es el relativo a la obtención del óptimo de cilindros de cada aca.tipo y el del número de cilindros disponibles de ese mismo aca.tipo, para obtener la desviación en horas de taller relacionando ambos valores y el tiempo de procesado. Respecto al óptimo, su cálculo requiere la contemplación de cuatro variables fundamentales:

- Histórico de laminación de los cilindros del aca.tipo
- Histórico de necesidades de cilindros de cada aca.tipo en los distintos trenes
- Tiempos de procesado en taller para cada aca.tipo
- Calendario de paradas programadas en los trenes.

### 3.2. Sistema Fuzzy.

Como en todo sistema fuzzy, es preciso definir unas funciones de pertenencia y unas reglas de funcionamiento del sistema. A través del paradigma de Mamdani [4], las entradas al sistema (4 por aca.tipo) se borrosifican y se relacionan entre sí según indican las 39 reglas definidas en el modelo para llegar a una solución borrosa final, la cual se convertirá en un valor exacto por el método de deborrosificación del centroide.

Las funciones de pertenencia permiten la utilización de lenguaje natural en las reglas, puesto que su función es agrupar el conjunto de valores posibles de una variable en un número limitado de conjuntos, lo suficientemente reducido para que puedan ser expresadas con lenguaje humano [5]. Su definición se llevó a cabo teniendo en cuenta los dominios de variación de las variables implicadas, así como la variación de actuación del maestro de taller en función de determinados valores de cada una de las variables. En la figura 3 se observan, a modo de ejemplo, las funciones de pertenencia asociadas a la variable “Desviación del óptimo” en horas de taller.

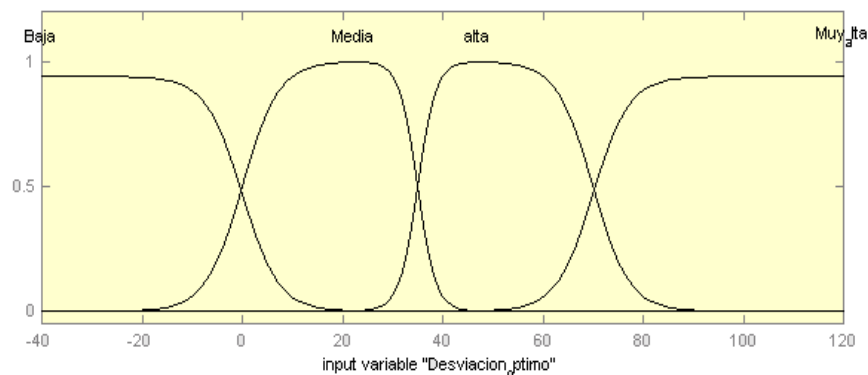


Figura 3: funciones de pertenencia de la variable “Desviación del óptimo”

Las reglas de funcionamiento del sistema son siempre más complejas de definir, puesto que desde la organización del taller se pretendía evitar que se simulase el comportamiento actual de las instalaciones, por lo que no se podía dar por válido el conocimiento del maestro desde un punto de vista final. Cualquier método de obtención de reglas de funcionamiento a partir de un conjunto de situaciones reales, como las redes neuronales, quedaba pues descartado. Fue necesario conocer de forma teórica cuál debería ser el funcionamiento final del taller, las situaciones críticas y los aspectos a evitar, para así tratar de construir un conjunto de reglas que se adaptasen a esa situación final. Puesto que se conocía la meta, con meticulosidad se fue comprobando cómo acercarse lo más posible a ella, con un laborioso proceso consistente en descartar determinadas reglas tras comprobar su redundancia o inutilidad, y construir otras que relacionaban variables individuales o conjuntos de 3 o 4 variables y que resultaban fundamentales para el funcionamiento del taller, siempre con la supervisión de los organizadores. Como muestra, se incluye la figura 4 con algunas de estas reglas en el editor del programa.



Figura 4: Interacción de reglas del sistema

Como software para el diseño del sistema borroso se utilizó el toolbox de Matlab (versiones posteriores a la 5.0). Está especialmente indicado pues resulta muy sencillo de comprender por las partes implicadas en el proceso aún sin conocimientos de lógica difusa. Son especialmente útiles los visores de superficie (figura 5), que permiten que los responsables supervisen la efectividad de determinadas reglas sin profundizar en otros aspectos más que en comprobar si mejoran la salida conseguida con respecto a la que existía anteriormente. El visor de reglas, usado conjuntamente con el de superficie, facilita la comprobación de cuál de ellas es responsable de una discontinuidad en la tendencia de la salida.

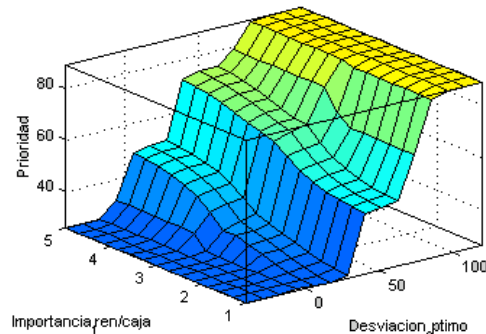


Figura 5: Visor de superficie del toolbox de Matlab relacionando, en este caso, la prioridad con la importancia de tren y la desviación del óptimo

## 4. Implementación del sistema.

### 4.1. Arquitectura física.

Todos los modelos, y por tanto el de inventario, deben ser incluidos dentro del sistema informático existente en el Departamento del Taller de Cilindros. En la figura 6 se reflejan las partes que intervienen en la aplicación y que proporcionan y obtienen información del sistema. Como se puede observar, la estructura comunica a través de la red corporativa un ordenador central (donde se encuentra el Sistema de Información, S.I., del Taller de Cilindros) y equipos pertenecientes al Taller de Cilindros, a los Trenes de Laminación y a Producción.

Cada parte aporta distinta información al modelo de inventarios:

- Departamento del Taller de Cilindros: Proporciona información de cilindros, estado actual de cada uno, disponibilidad de unidades, prioridad de procesado, ...
- Trenes de laminación: Toneladas laminadas por cada cilindro, hora de entrada y salida de un cilindro a laminar, paradas de los trenes previstas, averías en el tren debidas a cilindros, ...
- Producción: Información de posibles pedidos de materiales próximos, ...

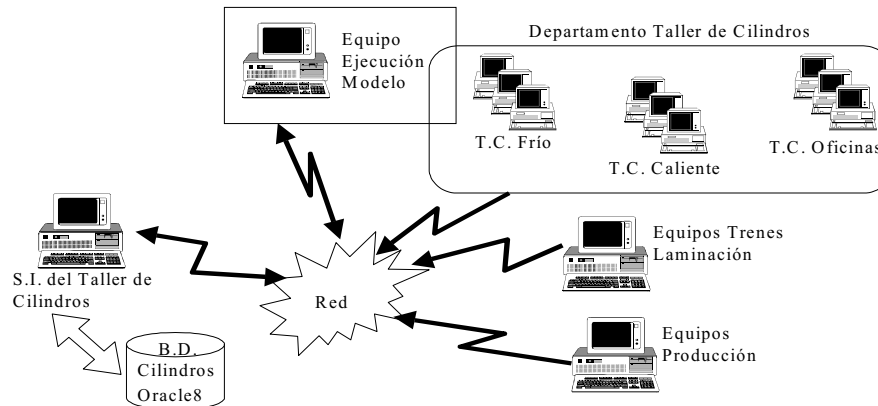


Figura 6: Arquitectura física del sistema

El módulo desarrollado se encuentra instalado en el ordenador central junto con el S.I. y se invocará desde equipos situados en el Taller de Cilindros. Podrá obtener la información actualizada on-line cuando sea necesaria y visualizará los resultados producidos por el modelo.

#### 4.2. Implementación del modelo.

El desarrollo del modelo se llevó a cabo en varias fases. La primera consistió en distribuir la funcionalidad en varios módulos para ser analizados, implementados y probados de forma independiente, para lo que se realizaron las siguientes tareas:

- Implementación del sistema fuzzy con el toolbox de Matlab.
- Creación de una réplica de la base de datos del taller.
- Desarrollo de un módulo de obtención de información mediante consultas SQL.
- Implementación de los algoritmos para el tratamiento de la información, y generación de los fichero legibles desde el módulo fuzzy.
- Procesamiento de los ficheros generados por el módulo fuzzy para volcar la información generada a la BD, también mediante consultas SQL con ayuda del lenguaje PL/SQL.

En la segunda fase, se unen las partes desarrolladas anteriormente en un único bloque que lea información, la gestione y almacene resultados en la BD. Se han utilizado (figura 7):

- ProC/C++: Herramienta proporcionada por Oracle mediante la cual a partir de ficheros donde se desarrollan funciones tipo "C" con sentencias SQL y PL/SQL intercaladas, genera nuevas funciones C que pueden ser compiladas con Microsoft Visual C++.
- Función "stand-alone" compilable también con Visual C++ que implementa el módulo fuzzy. Es capaz de interpretar los ficheros ".FIS" (que crea el toolbox de Matlab con información del sistema) y los datos de entrada, y proporcionar la salida del módulo fuzzy, convirtiéndose el Matlab en una herramienta de diseño, no indispensable para el funcionamiento del modelo.

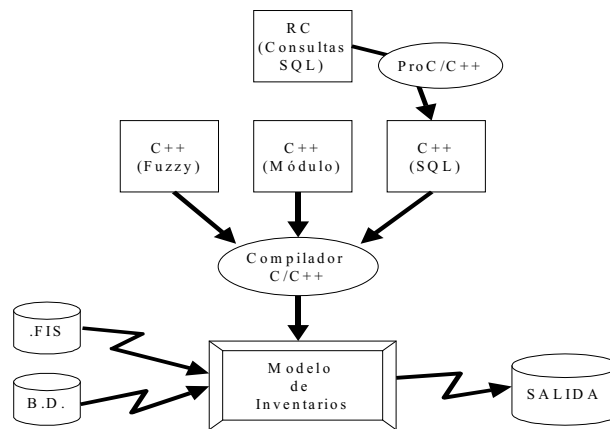


Figura 6: Diseño del modelo

Una vez que todos los módulos están en código C fueron compilados en un único programa que implementa el Módulo de Inventarios. Finalmente, se modificó la estructura de la BD para recoger la nueva información generada, y en el proceso de pruebas del modelo se realizaron los últimos ajustes a los parámetros de estimación y del módulo fuzzy.

## 5. Conclusiones

Con el sistema borroso se ha conseguido simular el conocimiento y la forma de actuar del experto, contando no sólo con la experiencia en el puesto de trabajo del maestro, sino las directrices y objetivos expresados por los responsables del taller, corrigiendo muchos de los errores y tendencias habituales del taller y liberando parte del tiempo del personal implicado. En este momento, el modelo se está implantando en las instalaciones tras haberse comprobado su eficiencia en la réplica de la base de datos.

## Referencias

- [1] Burnham, K.J., Haas, O.C.L. y James, D.J.G, (2000), "Intelligent systems for optimisation and control", *Kybernetes*, Vol. 29, No. 5/6, pp. 716-732.
- [2] López González, E. y Rodríguez Fernández, M. A., (2000), "Genetic optimisation of a fuzzy distribution model", *International Journal of Physical Distribution & Logistics Management*, Vol. 30, No. 7/8, pp. 681-696.
- [3] Zadeh, L. (1965), "Fuzzy Sets", *Information and Control*, Vol. 8, pp. 338-357.
- [4] Mamdani, E.H. y Assilian, S. (1975), "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, Vol. 7, No. 1, pp. 1-13.
- [5] Abbod, M.F., Linkens D.A. y Browne, (2000) "A blackboard software architecture for integrated intelligent control systems", *Kybernetes*, Vol. 29, No. 7/8, pp. 999-1015.