

## **Secuenciación en ambientes job shop por medio de agentes inteligentes y minería de datos**

**Omar D. Castrillon<sup>1</sup>, William A. Sarache<sup>1</sup>, Jaime A. Giraldo<sup>1</sup>**

<sup>1</sup>Doctor en Ingeniería. Universidad Nacional de Colombia. Dpto. Ing. Industrial. [odcastrillong@unal.edu.co](mailto:odcastrillong@unal.edu.co), [wasarachec@unal.edu.co](mailto:wasarachec@unal.edu.co), [jaiagiraldog@unal.edu.co](mailto:jaiagiraldog@unal.edu.co) Grupo de Innovación y Desarrollo Tecnológico

### ***Resumen***

*El objetivo del presente trabajo<sup>77</sup>, es disminuir el tiempo de proceso (Makespan) y aumentar el tiempo de utilización de las máquinas, disminuyendo el tiempo de ocio (idle), en ambientes Job Shop, mediante el diseño de una metodología basada en agentes inteligentes y minería de datos. Este trabajo se desarrolla en dos fases: En la primera, se aborda la identificación y definición de una metodología predictiva para los procesos de secuenciación en ambientes Job Shop. En la segunda etapa, se demuestra la efectividad de este sistema, en los procesos tradicionales de programación de la producción. La investigación propuesta se desarrollo en una empresa del sector metalmecánico, donde por medio de la combinación de agentes inteligentes y minería de datos se mejora la programación de un pedido, logrando un disminuir considerablemente su respectivo tiempo total de proceso y tiempo total de ocio.*

**Palabras Claves:** Job Shop Scheduling, Fitness, Makespan, Idle.

### **1. Introducción**

En la inteligencia humana, es común el empleo de de todas aquellas técnicas, métodos o procedimientos inteligentes para realizar una tarea, que no son producto de un riguroso análisis formal, sino del conocimiento de un experto. Igualmente, algunas metodologías de alto nivel pueden ser diseñadas por medio de técnicas especializadas (agentes inteligentes y minería de datos) con el fin de controlar adaptativamente varias heurísticas, metaheurísticas y metodologías para solucionar un problema específico. (Santana 2004,)

---

<sup>77</sup> Este trabajo se deriva de la participación de sus autores en un proyecto de investigación financiado por La Vicerrectoría de Investigaciones de la Universidad Nacional de Colombia, con referencia 20201004562, titulado “Reducción en los plazos de fabricación en ambientes Job Shop con técnicas de Inteligencia Artificial”.

Actualmente, el empleo de técnicas de inteligencia artificial, en los procesos de secuenciación y programación de la producción en ambientes Job Shop no ha sido muy difundido (Santana, 2004); aunque existen algunas metodologías basadas en: Algoritmos genéticos(Akhilesh 2006, Monch 2006, Guo 2006), partículas inteligentes (Van 2006, Fatih 2006, Lian 2006), colonia de hormigas(Huang 2006) y sistemas inmunes (Zandieh 2006) entre otras. En esta área de la ingeniería, han permitido solucionar algunos de sus principales problemas como: Repartición de recursos, ineficiente asignación de máquinas, inadecuada ordenación y secuencia de los **I** lotes en cada una de las **J** máquinas, incumplimiento de plazos de entrega, inapropiada estimación de la demanda, difícil manejo de las ordenes de compra, mal control de inventarios, frecuentes acciones de empuje de trabajos, desequilibrio en la capacidad de los centros de trabajo **CT** e insatisfacción de las condiciones de calidad.( Miltenburg, 1995)

Si bien, existen múltiples metodologías para tratar de solucionar gran parte de los anteriores problemas, en su gran mayoría, estas técnicas son estáticas o de corto alcance y presentan problemas cuando el número de lotes (**I**) y máquinas (**J**), cambia considerablemente, pudiéndose decir que no existe ninguna técnica de solución general; incluso las técnicas de simulación informatizadas, son de difícil aplicación debido al altísimo número de posibles soluciones del problema ( $I^J$ ).

Actualmente, el gran dinamismo de las técnicas de inteligencia artificial, hace que estas surjan como una alternativa a este problema, dado que establecen nuevas y mejores soluciones a partir de soluciones ya existentes, lo que permite una gran versatilidad en la solución de esta clase de problemas (Domínguez, 2001).

En general, la inteligencia artificial ha permitido solucionar una gran variedad de problemas tales como: Optimización de cualquier función calculable independientemente de que sea analítica o digital; solución del clásico problema del viajero, planificación de horarios de clase; programación del transporte público; programación de aterrizajes de aviones; carga de talleres, análisis de colas en problemas de programación dinámica. ( Azaron 2006, Yong 2006, PaulWatson 2006), vehículos inteligentes (Armingol 2007) y otras diversas áreas (Chen 2008).

## 2. METODOLOGIA

Para la solución de un problema Job Shop Scheduling (*JSSP*), se han escrito diversas aplicaciones relacionadas con la inteligencia artificial. Sin embargo, en este apartado y como se esbozó en la introducción, se propone una nueva metodología basada en agentes inteligentes y minería de datos, con el fin de mejorar la solución de esta clase de problemas. La metodología propuesta para esta investigación, empieza por considerar algunos supuestos o restricciones:

### Supuestos

- Cada operación, una vez iniciada, debe ser completada antes que otra operación pueda ser iniciada en la máquina.
- La posibilidad que un trabajo pueda requerir procesamiento doble en la misma máquina no es permitida. Además, cada trabajo es procesado sobre cada máquina.
- Cada trabajo debe completarse hasta su terminación.

- Cada tiempo de preparación es una secuencia independiente, es decir, el tiempo tomado para ajustar una máquina para un trabajo es independiente del último trabajo procesado.
- El tiempo para mover un trabajo entre máquinas no es considerado.
- El inventario en proceso es permitido. Esto es, los trabajos pueden esperar a que la próxima máquina este libre.
- Las máquinas opcionales en el procesamiento de un trabajo no es permitido.
- Las máquinas solo pueden realizar una operación a la vez.
- Las máquinas nunca paran (por ejemplo por fallos) y están disponibles a través del periodo de programación.
- Las restricciones tecnológicas son conocidas e inmutables en el avance del programa.

**Paso 1:** Tomando como referencia algunos escritos, Thompson (1963), un problema Job Shop  $N \times M$ , puede ser representado, por una estructura: Maquina, Tiempo de proceso. En la Tabla 1, se ilustra un proceso Job Shop  $3 \times 3$ .

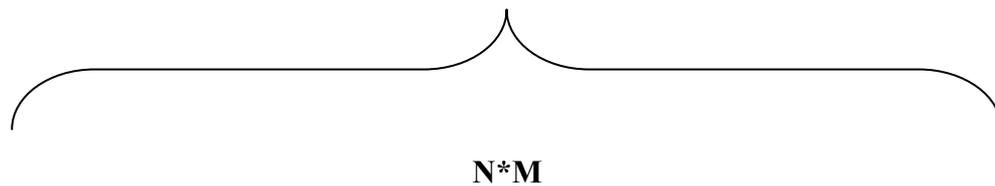
**Tabla 1.** Representación de un problema JSS  $3 \times 3$ . Thompson (1963)

C.Trabajo	Operación		
	1	2	3
1	M1;Tp11	M1;Tp12	M1;Tp13
2	M2;Tp21	M2;Tp22	M2;Tp23
3	M3;Tp31	M3;Tp32	M3;Tp33

**Paso 2:** El proceso Job Shop  $N \times M$ , debe ser codificado dentro de un arreglo de  $N * M$  elementos; por ejemplo: para un proceso Job Shop de  $3 \times 3$ , el siguiente vector es una codificación del mismo: ( **1,3,2** ,1,3,2, **1,2,3** ), donde cada elemento dentro del arreglo corresponde a un trabajo (Koonce, 2000). Los tres primeros elementos del arreglo representan la secuenciación del primer centro de trabajo *CT*, los elementos entre la posición 4 y la posición 6, representan la secuenciación del segundo centro de trabajo y los últimos tres elementos representan la secuenciación del tercer centro de trabajo.

**Paso 3:** Tomando como referencia *K* reglas de prioridad (Por ejemplo: Operación más corta *OMC*, operación más larga *OML*, menor tiempo restante *MTR*, menor fecha de entrega *MFE*,

etc.) (Domínguez, 2001) se construye una matriz de números, con dimensión  $(K*L, N*M)$ , Donde, N representa el número de trabajos, M representa el número de centros de trabajo, K el Nro de reglas de prioridad objeto de análisis Y L algunas de las posibles combinaciones que se pueden establecer con los N pedidos en cada uno de los CT. Cada uno de los elementos, de cada una de las filas de la anterior matriz, corresponden a un trabajo; como se expresó en el paso 2. Los N primeros elementos de la fila representan la secuenciación del primer centro de trabajo, los elementos comprendidos entre la posición  $N+1$  y la posición  $2*N$ , representan la secuenciación del segundo centro de trabajo y así sucesivamente. La matriz resultante, generara los datos que serán analizados mediante la minería de datos.



**Tabla 2.** Matriz resultante para aplicar la minería de datos.

Operación	Ct1	Ct1	Ct1	Ct2	Ct2	Ct2	Ct3	Ct3	Ct3
OMC									
OML									
MTR	Secuenciación en el CT1 con la regla MTR			Secuenciación en el CT2 con la regla MTR			Secuenciación en el CT3 con la regla MTR		
MFE									
OMC									

→ **K\*L**

**Paso 4:** Para cada una de las  $K*L$  filas de la matriz, establecidos en el paso anterior, se debe definir un diagrama de Gant, el cual establezca el orden de los procesos en el tiempo, en cada uno de los diferentes centros de trabajo. Establecido el anterior diagrama se procede a evaluar cada una de las diferentes filas de la matriz, con el fin de calcular los tiempos totales de proceso (Makespan) y los tiempos totales de ocio (Idle), bajo las siguientes funciones de cálculo (Fitness):

$$Fitness_{makespan} = \frac{1}{\min(\max_{1 \leq i \leq N} (\max_{1 \leq j \leq M} (P_{ij})))}$$

(1)

$$Fitness_{idle} = \frac{1}{\min \sum_{j=1}^m f_j}$$

(2)

El objetivo fundamental, es maximizar las dos funciones Fitness. Donde  $N$ , representa el número de trabajos.  $M$ , representa el número de máquinas,  $P_{ij}$  es el tiempo de procesamiento del trabajo  $i$ , en la máquina  $j$  y  $f_j$ , es el tiempo total ocio de la maquina  $j$ .

**Nota:** Los anteriores pasos, sólo constituyen la preparación de los datos, con el fin de poder emplear algunas de las técnicas más usadas en la minería de datos. El objetivo fundamental de esta metodología será predecir el comportamiento del sistema de información, al identificar en los nuevos trabajos, la mejor regla de prioridad que debe ser aplicada, según el comportamiento previo de los datos. Para esto se procede de la siguiente forma:

**Paso 5 :** Usando herramientas de consulta, se realiza un análisis de los datos (Minería de datos con la información de la tabla 2). Una consulta SQL a un conjunto de datos, permite rescatar algunos aspectos visibles como la distribución de las funciones Makespan e Idle, promedio, variación, desviación. Lo cual hace factible identificar la regla de prioridad más útil; para la empresa objeto de estudio.

**Paso 6:** Dado que identificar una regla de prioridad una vez, no basta, es necesario emplear un agente inteligente, el cual periódicamente monitoree el entorno y actualice la información de la tabla 2 (la nueva secuenciación de los pedidos en los diferentes CT), con el fin de calcular, los nuevos valores de las funciones MakeSpan e Idle, y determinar si es necesario cambiar la regla de prioridad que actualmente se emplea.

*Experimentación – Caso de aplicación*

Esta metodología se realizó en una empresa con 10 centros de trabajo, perteneciente al sector metalmecánico y dedicada principalmente a la fabricación de los siguientes productos: Tornillos, grapas, ferrules, niples, amarras, pernos, ganchos de anclaje, espárragos, varillas roscadas y tres diferentes clases de porta focos, que varían fundamentalmente en sus dimensiones, siendo este último producto y sus tres presentaciones (Porta focos tipo I, Tipo II, Tipo III ) el objeto de análisis de este trabajo.

El problema fundamental se centra en una inadecuada asignación de cargas y por consiguiente una ineficiente utilización de los recursos, ocasionando sobrecostos e incumplimiento de pedidos.

Los tiempos (en segundos) necesarios para el desarrollo de cada operación (Porta focos I, Porta focos II, Porta focos III) en cada uno de los diferentes centros de trabajo, son ilustrados entre las Tablas 3 y 5:

**Tabla 3.** Principales tiempos para producción de los porta focos en Centro de trabajo I.

CT I	Prom Porta focos I	Desv Porta focos I	Prom Porta focos II	Desv Porta focos II	Prom Porta focos III	Desv Porta focos III
<b>Doble troquelado de la pieza</b>	<b>1.48</b>	<b>0.6</b>	<b>1.8</b>	<b>0.6</b>	<b>2..3</b>	<b>0.6</b>

**Tabla 4.** Principales tiempos para producción de los porta focos en el centro de trabajo II.

CT II	Prom Porta focos I	Desv Porta focos I	Prom Porta focos II	Desv Porta focos II	Prom Porta focos III	Desv Porta focos III
<b>Total proceso</b>	<b>22</b>	<b>3.14</b>	<b>24.8</b>	<b>3.14</b>	<b>26.9</b>	<b>3.14</b>

**Tabla 5.** Principales tiempos para el roscado de los porta focos en el centro de trabajo III.

CT3 III	Prom Porta focos	Desv. Porta	Prom Porta focos	Desv Porta	Prom Porta focos	Desv Porta

	I	focos I	II	focos II	III	focos III
<b>Total proceso</b>	<b>4.02</b>	<b>0.47</b>	<b>6.02</b>	<b>0.53</b>	<b>8.02</b>	<b>0.57</b>

Experimentalmente, se puede demostrar que el tiempo de proceso de una operación en un centro de trabajo, es una función de probabilidad normal con media y desviación estándar conocidas. Finalmente, se considera que el porta focos II, tiene un tiempo menor de entrega que el porta focos I y que este a su vez, tiene un tiempo de entrega menor que el porta focos III.

### 3. RESULTADOS Y DISCUSIONES.

**Paso 1.** La representación del problema JSS para la construcción de los porta focos tipo I, II y III en una estructura máquina y tiempo proceso; como la que se ilustró en la Tabla 1, generó la siguiente matriz (Tabla 6):

**Tabla 6.** Representación de las operaciones del Portafoco

C. Trabajo	Operación		
	1	2	3
1	1;1.48	1;1.8	1;2.3
2	2;22	2;24.8	2;26.9
3	3;4.02	3;6.02	3;8.02

**Paso 2 .** Como resultado de este paso se estableció el vector (1,2,3,1, 2,3,1,2,3). Donde los tres primeros elementos del arreglo representan la secuenciación del primer centro de trabajo, los elementos entre la posición 4 y la posición 6, representan la secuenciación del segundo centro de trabajo y los últimos tres elementos representan la secuenciación del tercer centro de trabajo.

**Paso 3.** Con el fin de construir la respectiva matriz de números, con dimensión (4\*6,3\*3), se toma como referencia las siguientes reglas de prioridad: Operación más corta, operación más larga, menor tiempo restante, menor fecha de entrega. Como resultado de este proceso se obtiene la siguiente matriz de datos (Tabla 7):

**Tabla 7.** Matriz de números empresa objeto de análisis

Operación	Ct1	Ct1	Ct1	Ct2	Ct2	Ct2	Ct3	Ct3	Ct3
OMC	1	2	3	1	2	3	1	2	3
OML	3	2	1	3	2	1	3	2	1
MTR	1	2	3	1	2	3	1	2	3
MFE	2	1	3	2	1	3	2	1	3
OMC	1	2	3	1	2	3	1	2	3
OML	3	2	1	3	2	1	3	2	1
MTR	1	2	3	1	2	3	1	2	3
MFE	2	1	3	2	1	3	2	1	3
OMC	3	2	1	1	2	3	1	2	3
OML	1	2	3	3	2	1	3	2	1
MTR	1	2	3	1	2	3	1	2	3
MFE	2	1	3	2	1	3	2	1	3
OMC	1	2	3	3	2	1	1	2	3
OML	3	2	1	1	2	3	3	2	1
MTR	1	2	3	1	2	3	1	2	3
MFE	2	1	3	2	1	3	2	1	3
OMC	2	1	3	1	2	3	1	2	3
OML	3	1	2	3	2	1	3	2	1
MTR	1	2	3	1	2	3	1	2	3
MFE	2	1	3	2	1	3	2	1	3
OMC	1	2	3	1	2	3	1	2	3
OML	3	2	1	3	2	1	3	2	1
MTR	1	2	3	1	2	3	1	2	3
MFE	2	1	3	2	1	3	2	1	3

**Paso 4.** Una vez establecidos, los respectivos diagramas de Gant para cada una de las filas de la matriz anterior y considerando ligeras variaciones en los tiempos de procesos, se obtienen como resultado los siguientes tiempos totales de proceso (Makespan) y de ocio (Idle) en cada una de las secuencias anteriores. ver (Tabla 8):

**Tabla 8.** Matriz de tiempos de procesos y tiempos muertos.

Oper	Ct1	Ct1	Ct1	Ct2	Ct2	Ct2	Ct3	Ct3	Ct3	Fitness Idle	Fitness Makespan
OMC	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
OML	3	2	1	3	2	1	3	2	1	<b>0,03174603</b>	<b>0,01315789</b>
MTR	1	2	3	1	2	3	1	2	3	<b>0,03996803</b>	<b>0,01340662</b>
MFE	2	1	3	2	1	3	2	1	3	<b>0,02242152</b>	<b>0,01324503</b>
OMC	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
OML	3	2	1	3	2	1	3	2	1	<b>0,03174603</b>	<b>0,01315789</b>
MTR	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
MFE	2	1	3	2	1	3	2	1	3	<b>0,02242152</b>	<b>0,01324503</b>
OMC	1	2	3	1	2	3	1	2	3	<b>0,04001601</b>	<b>0,01340662</b>
OML	3	2	1	3	2	1	3	2	1	<b>0,03205128</b>	<b>0,01321004</b>
MTR	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
MFE	2	1	3	2	1	3	2	1	3	<b>0,02227171</b>	<b>0,01319261</b>
OMC	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
OML	3	2	1	3	2	1	3	2	1	<b>0,03174603</b>	<b>0,01315789</b>
MTR	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
MFE	2	1	3	2	1	3	2	1	3	<b>0,02242152</b>	<b>0,01324503</b>
OMC	1	2	3	1	2	3	1	2	3	<b>0,04011231</b>	<b>0,01341742</b>
OML	3	2	1	3	2	1	3	2	1	<b>0,03144654</b>	<b>0,01310616</b>

Oper	Ct1	Ct1	Ct1	Ct2	Ct2	Ct2	Ct3	Ct3	Ct3	Fitness Idle	Fitness Makespan
MTR	1	2	3	1	2	3	1	2	3	<b>0,03996803</b>	<b>0,01340662</b>
MFE	2	1	3	2	1	3	2	1	3	<b>0,02262443</b>	<b>0,01317523</b>
OMC	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>
OML	3	2	1	3	2	1	3	2	1	<b>0,03174603</b>	<b>0,01315789</b>
MTR	1	2	3	1	2	3	1	2	3	<b>0,0400641</b>	<b>0,01341202</b>

**Paso 5.** Una vez realizado el análisis de minería de datos, se encontró en el promedio, calculado sobre los resultados obtenidos por una misma regla de prioridad, la función que mejor caracterizaba los valores Makespan e Idle (Tabla 9):

**Tabla 9** Promedio Makespan e Idle

Operación	Promedio Fitness Idle	Promedio Fitness Makespan
OMC	0,01341202	0,0400641
<b>OML</b>	<b>0,01315789</b>	<b>0,03174603</b>
<b>MTR</b>	<b>0,01341022</b>	<b>0,04003203</b>
<b>MFE</b>	<b>0,01322576</b>	<b>0,02247191</b>

Como consecuencia de este análisis, se supone que la regla de prioridad más útil, para la empresa objeto de estudio, es la OMC.

**Paso 6.** Al igual que lo expresado en la metodología, identificar una regla de prioridad una vez no basta, por esto se debe emplear un agente inteligente, el cual periódicamente monitoree el entorno (los nuevos pedidos). Como resultado de este proceso, se obtiene la nueva regla de prioridad, en este caso MTR.

#### *DISCUSIONES.*

- En las diversas revisiones literarias (Acero 2003 ), se puede establecer que el diseño de hiperheurísticas para la solución de problemas de secuenciación en ambientes job shop no

ha sido muy difundido, sin embargo es notorio el empleo de algunas heurísticas, con las cuales se han obtenido resultados similares a los de este trabajo.

- Resultados similares, se han logrado con heurísticas diseñadas bajo el concepto de lógica difusa. Estas heurísticas permiten suponer una fecha de entrega de un trabajo, en un intervalo de tiempo determinado ( $d_{i1}$ ,  $d_{i2}$ ), en el cual el grado de satisfacción para el trabajo  $j_i$  es una función decreciente, según la tardanza del trabajo  $T_i$ , donde está última representa una función creciente según el incumplimiento del trabajo.
- En este artículo se emplea la misma regla de prioridad para secuenciar cada uno de los diferentes centros de trabajo. Sin embargo esto no es lo más usual; por ende, en futuros trabajos que se están realizando, se considera el pronóstico de una regla de prioridad diferente, para cada centro de trabajo. Igualmente se estima emplear clasificadores bayesianos, con el fin de mejorar los sistemas de predicción.

#### 4. CONCLUSIONES

- Una de las principales ventajas de esta metodología es que no es necesario programar constantemente la producción en una empresa, evitando retrasos innecesarios. La programación de las máquinas (regla de prioridad empleada) solo se cambia cuando el agente inteligente así lo determina.
- Los agentes inteligentes y la minería de datos constituyen una excelente técnica para solucionar los procesos de secuenciación en ambientes JSSP, en el cual se encuentra valores cercanos al óptimo. Si estas técnicas se aplican de forma combinada, como se ilustró en este artículo, es posible mejorar el proceso de secuenciación de pedidos para un centro de trabajo, optimizando el empleo de las reglas de prioridad y por consiguiente su secuenciación.
- Las técnicas de inteligencia artificial propuestas en este trabajo, para la solución de un problema JSSP, pueden ser aplicadas a otros problemas industriales como: Enrutamiento de vehículos, aprendizaje en redes, diseño de circuitos lógicos combinatorios, enrutamiento de paquetes en redes de comunicación y de forma general en la solución de cualquier problema NP- Duro, multi objetivo.
- La desviación estándar, que se considero en los tiempos de proceso, permitió una simulación más real del sistema objeto de análisis, sin embargo esta no tuvo una incidencia muy significativa en la secuenciación de los diferentes centros de trabajo.

## 5. BIBLIOGRAFÍA

Acero R. (2003) Aplicación de una heurística de búsqueda tabú en un problema de programación de tareas en líneas flexibles de manufactura. Santa fe de Bogota: Universidad de los andes.

Akhilesh K, Anuj P, et All. (2006). Psycho-Clonal algorithm based approach to solve continuous flow shop scheduling problem. *Expert Systems with Applications* 31 (2006) 504–514. Ed. Elsevier.

Armingol Jose M, de la Escalera A. (2007). Intelligent vehicle based on visual information. *Robotics and Autonomous Systems* 55 904–916. El Sevier

Azaron A. Katagiri H. (2.006) Longest path analysis in networks of queues: Dynamic scheduling problems. *European Journal of Operational Research* 174 132–149. Ed. Elsevier.

Chen Toly. (2008). An intelligent mechanism for lot output time prediction and achievability evaluation in a wafer fab. *Computers & Industrial Engineering* 54 77–94. El sevier.

Domínguez M. (2001). Dirección de Operaciones. J. A. Aspectos tácticos y operativos en la dirección de servicios. Mac Graw Hill. Madrid.

Fatih M., Y. C. Liang. (2006) A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem. *European Journal of Operational Research* Ed. Elsevier.

Guo Z., W. Wong. (2006) Mathematical model and genetic optimization for the job shop scheduling problem in a mixed- and multi-product assembly environment: A case study based on the apparel industry. *Computers & Industrial Engineering* 50 202–219. Ed Elsevier.

Huang K., C. Liao. (2006) Ant colony optimization combined with taboo search for the job shop scheduling problem. *Computers & Operations Research*. Ed. Elsevier.

Koonce D.A. (2000). Using data mining to find patterns in genetic algorithm solutions to a job shop schedule. *Computer & Industrial Engineering* 38 361 -374.

Lian Z, Gu X. (2006) A similar particle swarm optimization algorithm for permutation flowshop scheduling to minimize makespan. *Applied Mathematics and Computation* 175 773-785. Ed. Elsevier.

Miltenburg J. (1.995). Estrategia de fabricación productivity. Press Pórtland, Oregon.

Monch L., R. Schabacker. (2006). Genetic algorithm-based subproblem solution procedures for a modified shifting bottleneck heuristic for complex job shops. *European Journal of Operational Research* . Ed. Elsevier.

PaulWatson J. Howeb A. (2006). Deconstructing Nowicki and Smutnicki's i-TSAB tabu search algorithm for the job-shop scheduling problem. *Computers & Operations Research* 33 2623–2644. Ed Elsevier.

Santana J. B, C. C. Rodríguez, otros. (2004). Metaheurística. una revisión actualizada. Pág. 5. Universidad de la laguna. Grupo de computación inteligente. España.

Thompson G. L. & Muth, J. F.(1963). Industrial scheduling, Englewood Cliffs, NJ: Prentice Hall.

Van F, Engelbrecht (2006). A study of particle swarm optimization particle trajectories. Information Sciences 176 937 -971. Ed. Elsevier.

Yong C, Li P. (2006) A tabu search algorithm with a new neighborhood structure for the job shop scheduling problem. Computers & Operations Research. Ed Elsevier.

Zandieh M, Fatemi S.M. (2006). An immune algorithm approach to hybrid flow shops scheduling with sequence-dependent setup times. Applied Mathematics and Computation 180 111–127. Ed Elsevier.

