Revista de Matemática: Teoría y Aplicaciones 2013 **20**(2) : 231–241
cimpa – ucr issn: 1409-2433

GENERADOR DE MODELOS DE SIMULACIÓN. APLICACIONES EN SCHEDULING

SIMULATION MODELS GENERATOR. APPLICATIONS IN SCHEDULING

Jaime Alberto Giraldo* Omar Danilo Castrillón†
William Ariel Sarache‡

Received: 18/Feb/2012; Revised: 24/Apr/2012;
Accepted: 16/May/2012

---

*Departamento de Ingeniería Industrial, Universidad Nacional de Colombia, Sede Manizales, Colombia. E-Mail: Jaiagiraldog@unal.edu.co

†Misma dirección que/*Same address as:* J.A. Giraldo. E-Mail: Odcastrillong@unal.edu.co

‡Misma dirección que/*Same address as:* J.A. Giraldo. E-Mail: Wasarachec@unal.edu.co

Resumen

El objetivo del presente trabajo es divulgar entre la comunidad académica un prototipo informático que genera automáticamente modelos de simulación de un sistema de producción industrial dado, en aras de facilitar el aprendizaje de algunos modelos determinísticos y estocásticos de *scheduling*. La arquitectura básica del prototipo informático se desarrolló en términos de las capas de presentación, procesamiento y almacenamiento. A través de la capa de procesamiento el prototipo genera un modelo de simulación que se guarda en la capa de almacenamiento y que posteriormente puede ser corrido con el fin de probar diversos modelos de *scheduling* del sistema de producción. La teoría de *scheduling* ha originado diferentes alternativas para generar las mejores soluciones a problemas de programación de producción y se propone con el desarrollo de este *software* probar algunas de estas alternativas mediante simulación computarizada. Entre las ventajas que la simulación trae consigo está la posibilidad de evaluar el impacto que las decisiones tomadas tendrán, para así tener un acercamiento a la realidad que permita evaluar las decisiones con vistas a tomar la más asertiva. Para la prueba del prototipo se utilizó como ejemplo el modelamiento de un sistema de producción con 9 máquinas y 5 trabajos, según una configuración *Job Shop*, probando tiempos de procesamiento y paradas de máquinas estocásticos con el fin de medir tasas de utilización de máquinas y tiempo promedio de los trabajos en el sistema, como medidas de desempeño del sistema. Esta prueba muestra la bondad del prototipo, al ahorrarle al usuario la construcción del modelo de simulación.

Palabras clave: simulación, scheduling, programación de producción, sistema de producción.

Abstract

The aim of this study is to disseminate the academic community a prototype computer that automatically generates simulation models of a given industrial production system, in order to facilitate the learning of some deterministic and stochastic scheduling. The basic architecture of the prototype computer developed in terms of presentation layers, processing and storage. Through the processing layer generates a prototype simulation model which is stored in the storage layer and subsequently can be run in order to test various models of production scheduling system. Scheduling theory has led to alternatives to generate the best solutions to problems of production scheduling and intends to develop this software to test some of these alternatives by computer simulation. Among the benefits that simulation brings is the ability to assess the impact that decisions

will, in order to have an approach to reality to evaluate decisions in order to take more assertive. To test prototype was used as the modeling example of a production system with 9 machines and 5 works as a job shop configuration, testing stops processing times and stochastic machine to measure rates of use of machines and time average jobs in the system, as measures of system performance. This test shows the goodness of the prototype, to save the user the simulation model building

Keywords: simulation, scheduling, production system.

Mathematics Subject Classification: 90B36.

1 Introducción

Las prioridades competitivas se hallan relacionadas con las características de los productos o servicios que los clientes consideran deseables, por esto el sistema de producción es un factor determinante para el desarrollo de la actividad empresarial [6].

Con la optimización del sistema productivo de las empresas se busca satisfacer el mercado objetivo al que se debe responder y la programación (*scheduling*) apropiada de dicho sistema es un factor determinante para la creación de atributos diferenciadores, ya que el mejoramiento de ésta permite el aprovechamiento de la capacidad instalada, reducción de costos, mayor velocidad de respuesta y flexibilidad en los ambientes de producción, lo que se traduce en un mejor servicio al cliente y el aseguramiento de la satisfacción [2].

La teoría del *scheduling* ha originado diferentes alternativas para generar las mejores soluciones a dicho problema de programación y se propone con el desarrollo de este *software* probar estas alternativas mediante simulación computarizada. Las ventajas que la simulación trae consigo es la posibilidad de evaluar el impacto que las decisiones tomadas tendrán, para así tener un acercamiento a la realidad que permita evaluar las decisiones con vista a tomar la más asertiva. La simulación permite tener un enfoque holístico de las situaciones teniendo en cuenta factores determinísticos y estocásticos [1].

Entre los objetivos principales de la función de *scheduling* con aplicación en sistemas de producción industrial [7], están:

- Definir las fechas de entrega para los clientes.
- Minimizar los pedidos atrasados.

- Minimizar el tiempo de entrega.
- Minimizar el tiempo en el sistema.
- Maximizar la utilización de equipos y operarios.
- Minimizar tiempos de parada y de cambio.
- Minimizar el inventario en proceso

La medición de estos objetivos se puede hacer con programas de simulación de eventos discretos tales como: Promodel, Flexsim, Simul8, etc. Específicamente Promodel permite esta medición promediándola a un número determinado de réplicas y diversos escenarios de experimentación. Además es posible tratar de encontrar óptimos mediante el empleo de programas como Optquest y Simrunner.

La medición de estos objetivos se puede hacer con programas de simulación de eventos discretos tales como: *Promodel, Flexsim, Simul8*, etc. Específicamente *Promodel* permite esta medición promediándola a un número determinado de réplicas y diversos escenarios de experimentación. Además es posible tratar de encontrar óptimos mediante el empleo de programas como *Optquest* y *Simrunner*.

De otra parte, la utilización de la programación de actividades se adentra más en el proceso al tener en cuenta aspectos sobre el conocimiento en detalle del cómo, el cuándo y el dónde de la fabricación de cada tipo de producto. Para resolver la gestión de dichos aspectos se utiliza entonces la *programación de operaciones*, denominada también *función de scheduling*, que tiene por objetivo: determinar qué operaciones se van a realizar sobre los distintos pedidos, durante cada momento del horizonte de planificación, en cada centro de trabajo, de forma que, con la capacidad disponible en cada uno de ellos, se cumplan las fechas de entrega planificadas, empleando el menor volumen de recursos en inventario posible. El horizonte de planificación dependerá de las características del proceso productivo y su entorno, pudiendo variar entre unas horas y varias semanas [3].

De una manera más directa en [5] se afirma que el *scheduling* se encarga de la asignación de recursos escasos a diferentes actividades, con el objetivo de optimizar una o más medidas de desempeño. Es claro además, que probar el desempeño de los modelos de *scheduling* con los sistemas reales de producción no es práctico en términos de tiempo y costo, por lo que se sugiere hacerlo previamente con modelos de simulación de dichos sistemas

de producción. Es por esto que el generador de modelos de simulación propuesto está dirigido a estudiantes de pregrado en ingeniería industrial y de producción con el objetivo que puedan incrementar su preparación para tratar problemáticas del mundo de la industria y de los servicios; y también para que las empresas tengan acceso directo a la utilización de métodos y herramientas que la universidad, como generadora de conocimiento, pueda facilitar.

2 Metodología

2.1 Representación matemática de los modelos de *scheduling*.

Sobre la notación de los modelos de *scheduling*, se precisa que n corresponde al número de trabajos y m al número de máquinas, y además se consideran finitos y computacionalmente tratables. Para pasar a conceptualizar los siguientes términos, debe saberse que i denota las máquinas y j los trabajos. Entonces:

Tiempo de procesamiento (p_{ij}): si un trabajo j requiere procesamiento en la máquina i, entonces p_{ij} representa el tiempo de procesamiento del trabajo j en la máquina i. Cuando el índice i es omitido, denota que el trabajo j solo es procesado en una máquina.

Fecha de liberación (r_j): es el momento en el cual un trabajo j puede empezar a procesarse.

La conocida notación de $\alpha/\beta/\gamma$ para *scheduling* es introducida por [4] con el fin de clasificar los problemas. Explícitamente: el campo α describe el ambiente de maquinado y contiene una simple entrada. El campo β provee detalles de las características del trabajo y las restricciones de programación, y puede estar compuesto de varias entradas (β_1, β_2, β_3, β_4, β_5, β_6, β_7, β_8). El campo γ contiene la función objetivo por optimizar.

Posibles ambientes de maquinado que soporta el prototipo, para a:

a) Máquina simple (1): hay solamente una máquina en el sistema.

b) Máquinas paralelas e idénticas (P_m): ambiente con m máquinas en paralelo e idénticas. Si m es omitido, ello significa que el número de máquinas es arbitrario.

c) *Job-shop* (J_m): es un ambiente con m máquinas, en el cual cada trabajo tiene su propia ruta de procesamiento por seguir. Los trabajos pueden visitar las máquinas más de una vez e incluso no visitar algunas.

d) *Flow-shop* (F_m): es un sistema con m máquinas en el cual estas son ordenadas linealmente y todos los trabajos siguen la misma ruta, desde la primera máquina hasta la última.

Características de los trabajos y las restricciones de programación (β) que soporta el prototipo informático:

a) *Preemptions* (pmtn): esta característica se refiere a si el procesamiento de un trabajo en una máquina puede ser interrumpido e incluso posteriormente terminado en otra máquina. Si la *preemption* es permitida, es incluido en el campo β, de otra manera, se excluye.

b) No espera (nwt): esta restricción es solamente para sistemas *flow shop* y consiste en que a los trabajos no les es permitido esperar entre dos máquinas sucesivas. Si nwt no es especificado en el campo β, la espera es permitida entre dos máquinas sucesivas.

c) Fechas de liberación (r_j): la fecha de liberación r_j del trabajo j es el momento más temprano en el cual el trabajo j puede empezar a ser procesado. Si este símbolo no está presente, significa que el procesamiento del trabajo j puede empezar en cualquier momento.

d) Restricciones del número de trabajos (nbr): si este símbolo está presente, entonces el número de trabajos es restringido; de lo contrario, el número de trabajos no es una restricción y está dado como un parámetro de entrada n.

Características de la función por optimizar (γ):

El objetivo por minimizar es siempre una función del tiempo de terminación de los trabajos. Con respecto a un programa, C_j denota el tiempo de terminación del trabajo j y d_j denota el plazo de entrega del trabajo j. Entonces, el retraso del trabajo j es definido como:

$$L_j = C_j - d_j.$$

La tardanza del trabajo j está definida como:

$$T_j = \max(L_j, 0).$$

2.2 Descripción del prototipo informático

Teniendo en cuenta su arquitectura (Figura 1), los niveles del sistema son el de presentación, el de procesamiento y el de almacenamiento, que representan sus características desde la interfaz con el usuario, su programación lógica y el almacenamiento de los parámetros de funcionamiento. En el nivel de presentación, se le permite al usuario fijar los parámetros de operación del sistema de producción y desplegar la simulación y las medidas de desempeño generadas. En el nivel de procesamiento, mediante código VBA (*Visual Basic for Aplications*) se dan las instrucciones necesarias que modifican propiedades y activan métodos de los objetos del modelo de simulación escrito en *ProModel* a partir de los parámetros seleccionados en la capa de presentación.

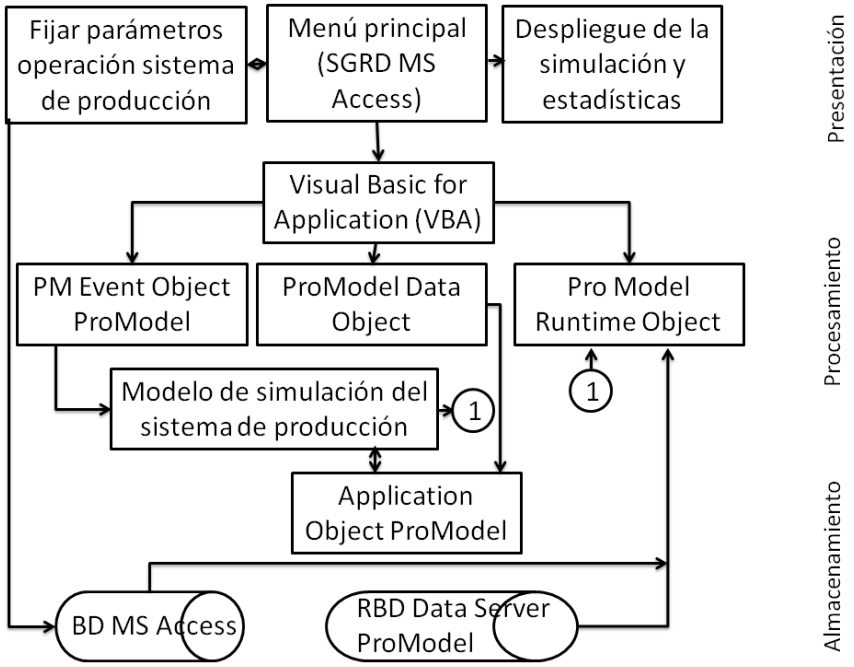

Figura 1: Arquitectura del prototipo informático.

En el nivel de almacenamiento se emplea una base de datos MS- Access (manipulable por el usuario), que almacena los valores de los parámetros de operación del sistema para poder generar a partir de estos el modelo de simulación.

	M1	M2	M3	M4	M5	M6	M7	M8	M9
J1	T(2,2,5) min	x	T(2,2,3) min	x	T(0,2,2) min	T(0,1,2) min	x	T(3,4,8) min	x
J2	T(1,2,3) min	T(3,4,8) min	x x	T(1,2,3) min	x x	x x	T(1,2,3) min	T(3,5,7) min	x x
J3	x	T(3,5,7) min	x	T(5,6,6) min	T(3,4,8) min	T(1,3,6) min	T(3,4,8) min	T(1,2,3) min	T(1,2,3) min
J4	x	0	T(1,2,3) min	x	T(3,5,7) min	x	T(3,5,7) min	x	x
J5	x	T(2,2,5)	x	x	0	T(1,2,3)	x	x	T(0,2,3)

Tabla 1: Tiempos de procesamiento para 5 trabajos en 9 máquinas.

3 Experimentación

Con el fin de probar el prototipo informático, se uso como ejemplo un sistema de producción *Job Shop* con nueve máquinas ($m = 9$) y cinco trabajos ($n = 5$), cada uno con una ruta de procesamiento diferente según lo ilustra la Figura 2.

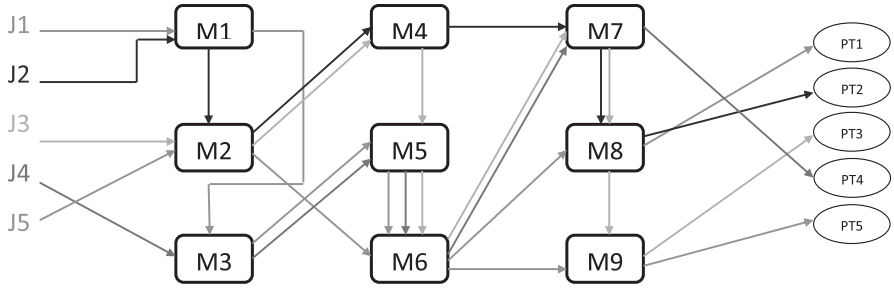

Figura 2: Representación de sistema de producción *Job Shop* con 9 máquinas y 5 trabajos.

La Tabla 1 muestra los tiempos de procesamiento de cada trabajo en cada máquina. Estos tiempos son aleatorios que para el caso siguen una distribución triangular con parámetros: mínimo, moda y máximo dados en minutos. Si un trabajo no se procesa en una máquina, su tiempo es 0 (cero).

Las Figuras 3(a,b,c) muestran la interface del prototipo informático que permite la entrada de datos de las máquinas y algunas restricciones

como paradas de máquina por reloj y por volumen de producción. Las Figuras 3(d,e,f) muestran la interface del prototipo informático que permite la entrada de datos de los trabajos, tiempos de procesamiento y fechas de liberación.

Número	Nombre	Capacidad	Unidades	Estadística
1	M1	1	1	0
2	M2	1	1	0
3	M3	1	1	0
4	M4	1	1	0
5	M5	1	1	0
6	M6	1	1	0
7	M7	1	1	0
8	M8	1	1	0
9	M9	1	1	0

(a) Datos de máquinas.

Número 1

Máquina M1

Paradas reloj

Frecuencia	PrimeraVez	Prioridad	Duración	Programado
30 MIN	0		6 MIN	
100			N[5,1] MIN	

(b) Datos de paradas de máquina 1 por reloj.

Número 1

Máquina M1

Paradas por volumen de producción

Frecuencia	PrimeraVez	Duración
50 MIN	0	6 MIN
100		N[5,1] MIN

(c) Datos de paradas de máquina 1 por producción.

Número	Nombre	Velocidad	Costo Inicial
1	J1	50	0
2	J2	50	0
3	J3	50	0
4	J4	50	0
5	J5	50	0

(d) Datos de trabajos.

Trabajo J1 J1

Máquina	Tiempo Proceso	% Defectuosos	Preemptions delTrabajo	Prioridad	Regla Enrutamiento
M1	T[2,2,5]sec	0			Disponible
M2	0	0			Disponible
M3	T[2,2,3]sec	0			Disponible
M4	0	0			Disponible
M5	T[0,2,2]sec	0			Disponible
M6	T[0,1,2]sec	0			Disponible
M7	T[0,1,2]sec	0			Disponible

(e) Tiempos de procesamiento de trabajo 1.

Trabajo	Locación	Cantidad	Primer Tiempo	Ocurrencias	Frecuencia
J1	Alm1	1	0	100	N[9,2]MIN
J2	Alm1	1	0	100	N[8,1]
J3	Alm1	1	0	100	N[10,2]
J4	Alm1	1	0	100	N[9,1]

(f) Datos de liberación de los trabajos al sistema.

Figura 3: Interface del prototipo informático.

4 Resultados

En las Figuras 4 se puede observar parte del código del modelo de simulación generado en formato Promodel.

```
Formatted Listing of Model:
*****************************************
Time units: Minutes
Distance units: Meters
*****************************************
Locations
*****************************************
Name Cap Units Starts Rules
-----------------------------------------
M1 1 1 Time series Oldest, , Firts
M2 1 1 Time series Oldest, , Firts
M3 1 1 Time series Oldest, , Firts
M4 1 1 Time series Oldest, , Firts
M5 1 1 Time series Oldest, , Firts
M6 1 1 Time series Oldest, , Firts
M7 1 1 Time series Oldest, , Firts
M8 1 1 Time series Oldest, , Firts
M9 1 1 Time series Oldest, , Firts
Alm1 Infinite 1 Time series Oldest, ,
Alm2 Infinite 1 Time series Oldest, ,
Alm3 Infinite 1 Time series Oldest, ,
Alm4 Infinite 1 Time series Oldest, ,
Alm5 Infinite 1 Time series Oldest, ,
Alm6 Infinite 1 Time series Oldest, ,
Alm7 Infinite 1 Time series Oldest, ,
Alm8 Infinite 1 Time series Oldest, ,
Alm9 Infinite 1 Time series Oldest, ,


************************************************
Clock downtimes for locations
************************************************

Loc Frequency First Time Priority Scheduled Disable logic
------------------------------------------------
M1 N(100,3)MIN 0 100 Yes No wait N(5,1)MIN
M2 N(105,5)MIN 0 200 Yes No wait N(6,1.5)
M3 N(103,2)MIN 0 150 Yes No wait N(5,0.8)MIN
M4 N(100,2.9)MIN 0 600 Yes No wait N(5,1)MIN
M5 N(98,4)MIN 0 460 Yes No wait N(5,1)MIN
M6 N(101,3)MIN 0 800 Yes No wait N(4.9,2)
M7 N(149,2)MIN 0 850 Yes No wait N(5,0.7)MIN
M8 N(193,1)MIN 0 800 Yes No wait N(5.3,1.3)MIN
M9 N(202,3)MIN 0 999 Yes No wait N(5.8,2)MIN
************************************************
usage downtimes for locations
************************************************
Loc Frequency First Time Priority Logic
------------------------------------------------
M1 N(100,3)MIN 0 100 wait N(6,2)MIN
M2 N(105,5)MIN 0 200 wait N(5,1)MIN
M3 N(103,2)MIN 0 150 wait N(4.9,2)MIN
M4 N(100,2.9)MIN 0 600 wait N(5.3,1.9)MIN
M5 N(98,4)MIN 0 460 wait N(5,1)MIN
M6 N(101,3)MIN 0 800 wait N(4,0.9)
M7 N(149,2)MIN 0 850 wait N(5,1)MIN
M8 N(193,1)MIN 0 800 wait N(6,2)MIN
M9 N(202,3)MIN 0 999 Wait N(4.4,0.8)MIN
```

Figura 4: Parte del código del modelo de simulación.

En la Tabla 2 se muestra las tasas de utilización de las máquinas obtenidas al correr el modelo de simulación a 10 réplicas incluyendo 500 trabajos liberados en el sistema (100 por cada tipo de trabajo).

	ModeloBase.MOD [Normal Run-Rep.1]							
Name	Schedule Time[HR]	Capacity	Total Entries	Avg Time Per Entry [MIN]	Avg Contents	Maximun Contents	Current Contents	% Utilización
M1	23,76	1,00	500,00	0,99	0,35	1.00	0,00	34,84
M2	23,65	1,00	500,00	2,63	0,93	1.00	0,00	92,63
M3	23,95	1,00	500,00	0,87	0,30	1.00	0,00	30,25
M4	23,85	1,00	500,00	1,53	0,53	1.00	0,00	53,39
M5	23,90	1,00	500,00	2,32	8,81	1.00	0,00	80,97
M6	23,89	1,00	500,00	1,23	0,43	1.00	0,00	42,82
M7	24,16	1,00	500,00	2,39	0,82	1.00	0,00	82,34
M8	24,44	1,00	500,00	2,38	0,81	1.00	0,00	81,24
M9	24,41	1,00	500,00	0,74	0,25	1.00	0,00	25,27
Alm1	25,11	999999,00	500,00	251,36	83,41	409,00	0,00	0,01
Alm2	25,11	999999,00	500,00	442,14	146,72	302,00	0,00	0,01
Alm3	25,11	999999,00	500,00	0,29	0,10	4,00	0,00	0,00
Alm4	25,11	999999,00	500,00	1,82	0,61	6,00	0,00	0,00
Alm5	25,11	999999,00	500,00	9,34	3,10	8,00	0,00	0,00
Alm6	25,11	999999,00	500,00	0,73	0,24	4,00	0,00	0,00
Alm7	25,11	999999,00	500,00	5,37	1,78	9,00	0,00	0,00
Alm8	25,11	999999,00	500,00	2,53	0,84	6,00	0,00	0,00
Alm9	25,11	999999,00	500,00	0,62	0,21	4,00	0,00	0,00

Tabla 2: Tasas de utilizacion de máquinas.

En la Tabla 3 se muestran los tiempos en el sistema, espera, operación, movimiento y bloqueo de los trabajos obtenidos al correr el modelo de simulación a 10 réplicas incluyendo 500 trabajos liberados en el sistema (100 por cada tipo de trabajo).

Finalmente se espera que la función objetivo sea evaluada en términos

	ModeloBase.MOD [Normal Run-Rep.1]						
Name	Total Exits	Current Qty In System	Avg Time Per Entry [MIN]	Avg Time In Move Logic [MIN]	Avg Time Waiting [MIN]	Avg Time In Operation [MIN]	Avg Time Blocked [MIN]
J1	100,00	0,00	722,89	0,00	0,00	13,54	709,34
J2	100,00	0,00	728,00	0,00	0,00	17,26	710,73
J3	100,00	0,00	732,98	0,00	0,00	30,46	702,52
J4	100,00	0,00	733,38	0,00	0,00	12,96	720,42
J5	100,00	0,00	735,04	0,00	0,00	6,99	728,04

Tabla 3: Tiempos en el sistema, de espera, operación, movimiento y bloqueo de los trabajos.

de: tasa de utilización de las máquinas, tiempo promedio de permanencia de un trabajo en el sistema, tiempo promedio de espera de los trabajos, etc.

Referencias

[1] Castrillón, O.D.; Giraldo, J.A.; Sarache, W.A. (2009) *Técnicas de Programación de la Producción. Aplicaciones en Ambientes Job Shop.* Unibiblos. Bogotá, Colombia.

[2] Castrillón, O.D.; Giraldo, J.A.; Sarache, W.A. (2010) *Técnicas Inteligentes y Estocásticas en Scheduling. Un Enfoque en la Producción y las Operaciones.* Unibiblos, Bogotá, Colombia.

[3] Domínguez, J.A.; Álvarez A.; Domínguez M.A.; García, S.; Ruiz, A. (1995) *Dirección de Operaciones. Aspectos Estratégicos en la Producción y los Servicios.* Mac Graw Hill, Madrid.

[4] Graham, R.L.; Lawler, E.L.; Lenstra, J.K.; Rinooy Kan, A.H.G. (1979) Optimization and aproximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics* **5**: 287–326.

[5] Leung, J.; Pinedo, L. (2003) "Minimizing total completion time on parallel machines with deadline constraints", *SIAM Journal Computing* **32**(5): 1370–1388

[6] Miltenburg, J. (2005) *Manufacturing Strategy. How to Formulate and Implement a Winning Plan.* Productivity Press, Portly OR.

[7] Russell, R.; Taylor III, B. (2003) *Operations Management*, 4th Ed. Prentice Hall, New Jersey.