

Modelo de Programación Lineal Entera Mixta para la Programación de Sistemas tipo *Job-Shop Flexible* en Entornos *Make to Order*

Mixed Integer Lineal Programming Model to Schedule Flexible Job-Shop Systems in Make to Order Environments

DOI: <http://dx.doi.org/10.17981/ingecuc.13.2.2017.03>

Artículo de Investigación Científica. Fecha de Recepción: 28 de marzo de 2017. Fecha de Aceptación: 25 de mayo de 2017.

Sebastián Enrique Ortiz Gaitán

Escuela Colombiana de Ingeniería Julio Garavito. Bogotá D.C., Colombia

sebastian.ortiz-g@mail.escuelaing.edu.co

Carlos Rodrigo Ruiz Cruz

Escuela Colombiana de Ingeniería Julio Garavito. Bogotá D.C., Colombia

carlosr.ruiz@escuelaing.edu.co

Para citar este artículo:

S.E. Ortiz Gaitán y C.R. Ruiz Cruz. "Modelo de Programación Lineal Entera Mixta para la Programación de Sistemas tipo Job-Shop Flexible en Entornos Make to Order" INGE CUC, vol. 13, no. 2, pp. 18-34, 2017 DOI: <http://dx.doi.org/10.17981/ingecuc.13.2.2017.03>

Resumen-- Los sistemas de producción Job Shop (JS) se caracterizan porque los trabajos a ser procesados tienen diferentes rutas de producción. Una generalización de este tipo de sistemas es el Job Shop Flexible (FJS), en el cual se cuenta con más de una máquina por estación para realizar alguna de las operaciones. En este artículo se propone un modelo de programación lineal entera mixta para programar este tipo de sistemas en el cual la función objetivo es minimizar el número de trabajos tardíos. El modelo se aplica para la programación de producción de una pequeña empresa manufacturera. Los resultados muestran un desempeño razonable en términos del objetivo que se persigue y los tiempos computacionales para lograrlo. En instancias de hasta 14 trabajos el modelo permite obtener soluciones óptimas.

Palabras claves- Programación de producción, Job shop flexible, Programación lineal entera mixta, trabajos tardíos.

Abstract-- Job Shop (JS) production systems are characterized by different route of process of the jobs to be processed. A generalization of this type of systems is the Flexible Job Shop (FJS), in which there is more than one machine per station to perform some of the operations. In this article is proposed a mixed integer lineal programming model to schedule such systems in which objective function is to minimize the number of tardy jobs. The model is applied for scheduling a small manufacturing company production system. The results show a reasonable performance in terms of the objective pursued and the computational times to achieve it. In instances of up to 14 jobs the model allows to obtain optimal solutions.

Key Words— Production schedule, Flexible job shop, Mixed integer lineal programming, tardy jobs.

I. INTRODUCCIÓN

Las empresas que tienen definido un programa de producción están en capacidad de ofrecer un mayor cumplimiento con las fechas de entrega de los pedidos, optimizando el uso de recursos, aumentando los niveles de servicio y la satisfacción del cliente; y en consecuencia reduciendo costos, generando mayores utilidades.

El problema de programación de sistemas *job shop* (JS) es uno de los más importantes y difíciles en el campo de la programación de la producción. Este problema está definido por un conjunto de M máquinas y un conjunto de J trabajos, donde cada trabajo consiste de un número de operaciones sucesivas que se pueden desarrollar en una sola máquina. El objetivo consiste en definir la secuencia apropiada para la realización de las operaciones en las máquinas para optimizar el indicador de desempeño relevante en cada caso. Estos indicadores en general se enfocan en mejorar la productividad del sistema o el nivel de servicio del mismo. Una generalización de este tipo de problemas es el de programación de sistemas *job shop flexible* (FJS) que se diferencia en que cada operación puede desarrollarse en un conjunto dado de máquinas, por lo cual adicionalmente a la definición de la secuencia, debe asignarse cada operación a la máquina apropiada. Este tipo de problemas se acercan más a las situaciones que enfrentan sistemas productivos reales, por lo cual tienen mayor aplicación práctica.

Este artículo se encuentra organizado como se presenta a continuación. En la sección II se realiza una revisión literaria del problema expuesto. En la sección III se presentan las características de la empresa en la cual se desarrolló el proyecto. La sección IV describe el modelo matemático propuesto. En la sección V se presentan los resultados de la aplicación del modelo y finalmente, en la sección VI se exponen las conclusiones y recomendaciones.

II. REVISIÓN LITERARIA

Los primeros modelos matemáticos utilizados para la programación de la producción datan de la década de 1960 [1]. El enfoque utilizado principalmente ha sido la formulación mediante el uso de programación entera mixta. Si bien los métodos exactos garantizan una solución óptima de los problemas no son necesariamente los más eficientes en cuanto a tiempo de procesamiento.

Se ha demostrado que este tipo de problemas son NP-Hard aun en problemas con trabajos que tienen tres operaciones y existen dos máquinas [2]. Esto hace que virtualmente sean imposibles de resolver de forma exacta en tiempos computacionales razonables. En general el objetivo de los modelos propuestos en la literatura es la minimización del makespan. Este indicador es apropiado cuando se busca mejorar la productividad del sistema ya que reduciendo el makespan, también se disminuyen los tiempos ociosos y el sistema queda disponible más pronto para atender nuevos trabajos. En cambio, cuando el interés está en mejorar los niveles de servicio al cliente el uso de otro tipo de indicadores tales como la tardanza máxima, tardanza total o número de trabajos tardíos, el número de referencias se reduce considerablemente. Algunas de las más relevantes se resumen en la tabla 1.

Tabla 1. Referencias bibliográficas para FJS minimizando variables asociadas al nivel de servicio

Referencia	Año	Objetivo del modelo	Método de solución
Scrigh et al. [3]	2004	Minimizar la tardanza total	Búsqueda Tabú
Wu y Weng [4]	2005	Minimizar tardanza y adelantos totales ponderados	Método multiagentes
Zribi et al. [5]	2006	Minimizar la tardanza total	Diferentes metaheurísticas
Tay y Ho [6]	2008	Minimizar Makespan, tardanza media y tiempo de flujo medio	Regla de despacho
Baykasoglu y Ozbakir [7]	2010	Minimizar tardanza media	Reglas de despacho
Thörnblad [8]	2011	Minimizar la tardanza total y maximizar la utilización de las células.	Exacto y heurístico
Chen et al. [9]	2012	Minimizar makespan, tardanza total y tiempo total de inactividad	Algoritmo genético
Lin et al. [10]	2012	Minimizar la tardanza total	Algoritmo evolucionario híbrido
Gomes et al. [11]	2013	Minimizar adelanto y tardanzas totales ponderados	Exacto
Huang et al. [12]	2013	Minimizar los costos asociados a tardanza y adelantos totales	Colonia de hormigas con dos feromonas
Mousakhani [13]	2013	Minimizar tardanza total	Algoritmo de búsqueda local
Sadrzadeh [14]	2013	Minimizar makespan y tardanza media	Diferentes metaheurísticas
Thörnblad et al. [15]	2013	Minimizar makespan y tardanza total	Exacto y heurístico
Calleja y Pastor [16]	2014	Minimizar la tardanza promedio	Reglas de despacho
Liu et al. [17]	2014	Minimizar consumo total de energía y la tardanza total ponderada	Algoritmo genético sin clasificación dominante
Na y Park [18]	2014	Minimizar tardanza total	Exacto
Gao et al. [19]	2016	Minimizar makespan, tardanza y adelanto promedios	Búsqueda de armonía discreta
Gatius [20]	2016	Minimizar los costos de energía, tardanza y adelanto totales	Diferentes metaheurísticas
Türkyilmaz y Bulkan [21]	2016	Minimizar la tardanza total	Algoritmo híbrido

Fuente: Autores

Es de particular interés el trabajo de Demir e Isleyen [22] en relación a los diferentes enfoques de formulación matemática de los problemas de FJS. En su trabajo destacan tres categorías de formulaciones matemáticas que

se diferencian principalmente por el tipo de variables utilizadas. Estas son:

- Variables indexadas por tiempo: Las variables de decisión son binarias y definen si en un periodo de tiempo se realiza una operación en una máquina determinada.
- Variables de precedencia: Las variables de decisión son binarias y definen si una operación determinada precede a otra en su procesamiento en la máquina asignada.
- Variables de secuencia-posición: Las variables de decisión son binarias y definen si una operación determinada es asignada a una posición en la secuencia de procesamiento de la máquina asignada.

III. APLICACIÓN

El proyecto se desarrolló en una pequeña empresa de producción de empaques de caucho ubicada en la ciudad de Bogotá D.C., Colombia, y con más de 30 años en el mercado. Se especializa en la fabricación de empaques de caucho para uso industrial y automotriz. Sus clientes principales son empresas que fabrican partes y repuestos para electrodomésticos y para automóviles. La empresa desarrolla sus operaciones productivas un entorno bajo pedido (Make to Order, MTO) y con un sistema de producción enfocado hacia el proceso que se puede clasificar como una configuración de Job Shop Flexible dado que cada orden tiene una secuencia específica de procesamiento y en cada proceso se cuenta con varias máquinas que pueden ejecutar la tarea. La empresa objeto de estudio, no contaba con un procedimiento formal que le permitiera definir la programación de las órdenes de producción para satisfacer la demanda de los clientes. Básicamente este consistía programar de acuerdo al orden en el cual se recibían de los clientes, es decir, utilizando una regla de despacho FIFO (First in First Out), sin considerar el cumplimiento de las fechas de entrega convenidas. Esta situación motivó el desarrollo del proyecto con el fin de encontrar alternativas que permitieran mejorar su operación y de esta forma su productividad y competitividad.

La empresa fabrica sesenta referencias de producto, los cuales se clasifican en las siguientes familias: Tapones, grommets, mangueras, fundas, soportes, empaques cónicos, bridas, ruedas y empaques para fumigadoras. Por el tamaño de la empresa se reciben entre 3 y 7 pedidos de los clientes y en cada pedido se ordenan entre 1 y 3 productos diferentes, de tal forma el número de órdenes de producción podría estar entre 3 y 21.

Los pedidos de los clientes se reciben permanentemente, sin embargo, la programación de la producción se hace al inicio de la semana considerando las órdenes recibidas a la fecha. Los pedidos en general son diferentes en cuanto a los productos, especificaciones y cantidades. Una característica adicional muy importante para caracterizar el sistema es que cuando los pedidos se reciben se define, en conjunto con el cliente, una fecha de entrega para los mismos.

Los procesos productivos con los que cuenta la empresa son:

1. **Mezclado:** Para este proceso se emplea un molino, el cual se encarga de mezclar el caucho con aditivos que permiten garantizar las especificaciones del producto final. En este proceso se cuenta con 2 máquinas.
2. **Preformado:** Este proceso se realiza especialmente para tapones y mangueras. Consiste en tomar el caucho ya mezclado y consiste en dar una forma básica cilíndrica que facilite su procesamiento posterior. En este proceso se cuenta con 3 máquinas.
3. **Laminado:** Este proceso consiste en tomar el caucho mezclado y pasarlo por una laminadora, con el fin de crear láminas de diferentes grosores que permita la manipulación y conformado final de los productos. En este proceso se cuenta con 2 máquinas.
4. **Conformado metal-caucho:** Este proceso se realiza principalmente con las ruedas y soportes. Se realizan operaciones para adherir el caucho a las piezas de metal. En este proceso se cuenta con 1 máquina.
5. **Vulcanizado:** En este proceso se incluye el moldeado de la pieza y la vulcanización de la misma para mejorar las propiedades del caucho. En este proceso se cuenta con 3 máquinas.
6. **Rebabeado:** Consiste en remover los residuos resultantes después del vulcanizado. En este proceso se cuenta con 2 puestos de trabajo.

En la Fig. 1 se muestra el diagrama multi-producto que permite relacionar las familias de productos con los procesos productivos de la empresa.

En la Fig. 2 se presenta la distribución de planta actual de las áreas productivas de la empresa que permite observar la disposición de los procesos productivos y la cantidad de máquinas disponibles para cada uno de ellos.

IV. MODELO PROPUESTO

En esta sección se presenta el modelo propuesto en el cual se utiliza un enfoque de formulación matemática empleando variables de tipo secuencia-posición.

A. Supuestos del modelo

Se consideran los siguientes supuestos:

- Todas las máquinas están disponibles al inicio del programa
- Las máquinas pueden procesar un solo trabajo a la vez.
- En el flujo de procesamiento de los trabajos no se presenta recirculación, es decir, los trabajos pasan de una estación a la siguiente apenas vayan concluyendo sus respectivas operaciones, debido a que por la naturaleza del producto y de la materia prima utilizada no existe la posibilidad de reproceso.
- No se puede iniciar una operación de un trabajo sin haber terminado la operación anterior, o hasta que existan máquinas disponibles para realizar dicha operación, es decir, que no se puede ejecutar más de una operación de un mismo trabajo al tiempo.

Proceso	Familia de producto								
	Tapones	Grommets	Mangueras	Fundas	Soportes	Empaques Cónicos	Bridas	Ruedas	Empaques para Fumigadoras
Mezclado	1	1	1	1	1	1	1	1	1
Preformado	2		2						
Laminado		2		2		2	2		
Conformado Metal-Caucho					2			2	2
Vulcanizado	3	3	3	3	3	3	3	3	3
Rebado	4	4	4	4	4	4	4	4	4

Fig. 1. Diagrama multi-producto para las familias de productos de la empresa.
Fuente: Autores

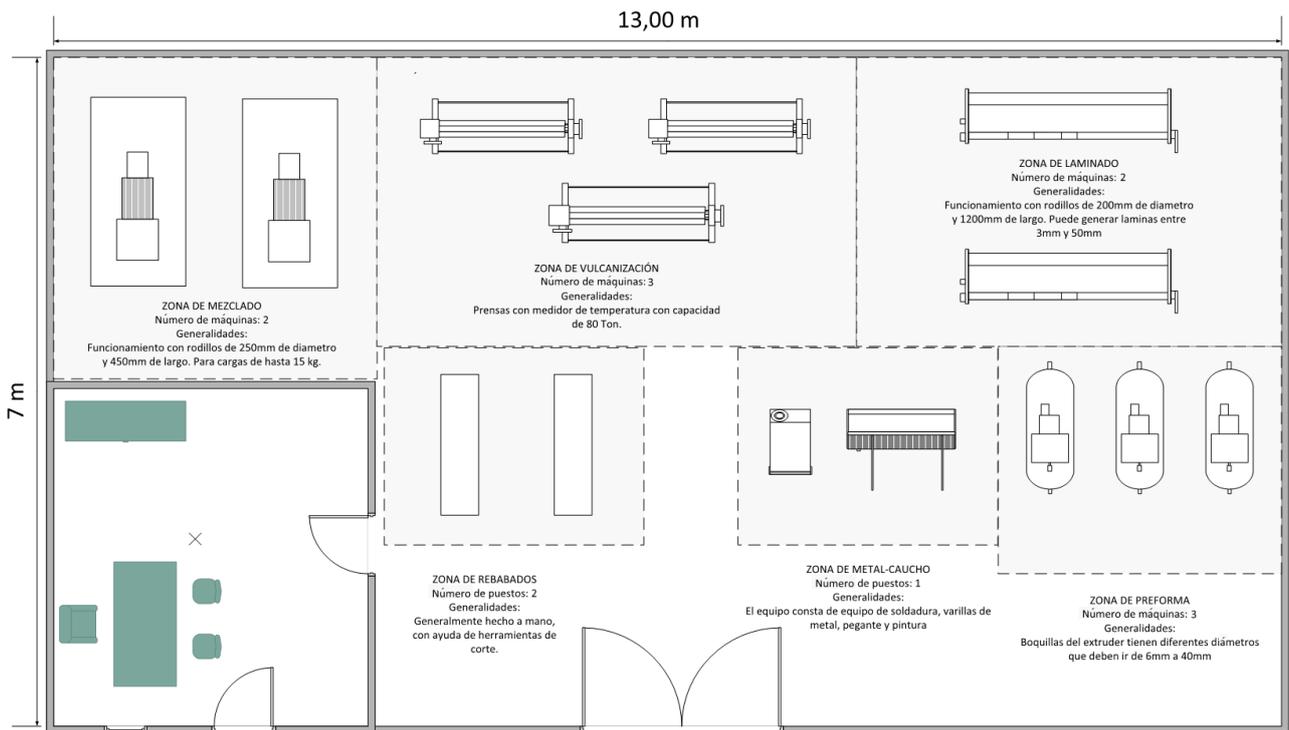


Fig. 2. Distribución en planta de la empresa.
Fuente: Autores

- Siempre hay personal disponible para realizar cada operación.
- Los tiempos de transporte entre operaciones y tiempos de alistamiento de máquinas son despreciables.
- No existen paradas de emergencia ni necesidades de mantenimiento a lo largo del proceso
- Todos los trabajos tienen la misma importancia.
- Todos los trabajos están disponibles al inicio del programa.
- Las máquinas son homogéneas, es decir, el tiempo de procesamiento del trabajo no depende de la máquina seleccionada.

B. Conjuntos, índices, parámetros y variables de decisión

Conjuntos

J : conjunto de trabajos ($i \in J = \{1, \dots, n\}$)
 J_i : conjunto de operaciones ($j \in J_i = \{1, \dots, n_i\}$)
 \mathcal{K} : conjunto de máquinas ($k \in \mathcal{K} = \{1, \dots, m\}$)
 \mathcal{L} : conjunto de posiciones ($l \in \mathcal{L} = \{1, \dots, s_k\}$)

Parámetros:

n : número total de trabajos
 m : número total de máquinas
 a_{kij} : $\begin{cases} 1, & \text{si la operación } j \text{ del trabajo } i \text{ se hace en la} \\ & \text{máquina } k \\ 0, & \text{en caso contrario} \end{cases}$
 p_{ij} : tiempo de proceso de la operación j del trabajo i
 M : número muy grande
 d_i : fecha de entrega del trabajo i

Variables de decisión:

x_{ijkl} : $\begin{cases} 1, & \text{si la operación } j \text{ del trabajo } i \text{ se hace en la} \\ & \text{máquina } k \text{ en la posición } l \\ 0, & \text{en caso contrario} \end{cases}$
 v_{ijk} : $\begin{cases} 1, & \text{si la operación } j \text{ del trabajo } i \text{ se hace en la} \\ & \text{máquina } k \\ 0, & \text{en caso contrario} \end{cases}$
 TM_{kl} : tiempo de inicio de la máquina k en la posición l
 ti_{ij} : tiempo de inicio de la operación j del trabajo i
 C_i : tiempo requerido para completar el trabajo i
 T_i : tardanza del trabajo i
 U_i : $\begin{cases} 1, & \text{si } c_i - d_i > 0 \\ 0, & \text{en caso contrario} \end{cases}$
 s_k : número de operaciones asignadas a la máquina k

C. Formulación del modelo

Dados los conjuntos, parámetros y variables definidos previamente se propone el siguiente modelo matemático cuya función objetivo es minimizar el número total de trabajos tardíos:

$$\text{Min } z = \sum_i U_i \quad (1)$$

Sujeto a:

$$C_i \geq ti_{ij} + p_{ij} \quad \forall_i \quad (2)$$

$$T_i \geq C_i - d_i \quad \forall_i \quad (3)$$

$$T_i \leq U_i * M \quad \forall_i \quad (4)$$

$$ti_{ij} + p_{ij} \leq ti_{i,j+1} \quad \forall_i, \forall_j = 1 \dots J_i - 1 \quad (5)$$

$$TM_{kl} + p_{ij} * x_{ijkl} \leq TM_{kl+1} \quad \forall_i, j, k, \forall l = 1, \dots, s_k - 1 \quad (6)$$

$$TM_{kl} \leq ti_{ij} + (1 - x_{ijkl}) * M \quad \forall_i, j, k, l \quad (7)$$

$$TM_{kl} + (1 - x_{ijkl}) * M \geq ti_{ij} \quad \forall_i, j, k, l \quad (8)$$

$$v_{ijk} \leq a_{kij} \quad \forall_i, j, k \quad (9)$$

$$\sum_i \sum_j x_{ijkl} = 1 \quad \forall_k, l \quad (10)$$

$$\sum_k v_{ijk} = 1 \quad \forall_i, j \quad (11)$$

$$\sum_l x_{ijkl} = v_{ijk} \quad \forall_i, j, k \quad (12)$$

$$TM_{kl} \geq 0 \quad \forall_k, l \quad (13)$$

$$ti_{ij} \geq 0 \quad \forall_i, j \quad (14)$$

$$C_i, T_i \geq 0 \quad \forall_i \quad (15)$$

$$s_k \geq 0 \quad \forall_k \quad (16)$$

$$x_{ijkl} \in \{0,1\} \quad \forall_i, j, k, l \quad (17)$$

$$v_{ijk} \in \{0,1\} \quad \forall_i, j, k \quad (18)$$

$$U_i \in \{0,1\} \quad \forall_i \quad (19)$$

La función objetivo (1) busca minimizar el número de trabajos tardíos, definidos estos como aquellos trabajos que son terminados después de la fecha de entrega establecida. Esta consideración es el aporte más innovador del modelo debido a que esta medida de desempeño no se considera normalmente en la literatura, en particular para este tipo de sistemas productivos, y resulta de suma importancia en los entornos de manufactura MTO debido a que en estos se definen fechas de entrega y las penalizaciones por el incumplimiento están asociadas a el tiempo de incumplimiento, o como se considera en este modelo, por el solo incumplimiento, ya que los clientes pueden no recibir la orden debido a que no se entrega en la fecha prometida. La restricción (2) define el tiempo requerido para completar cada trabajo. La restricción (3) establece la tardanza de cada trabajo. La restricción (4) considera un trabajo como tardío si tiene tardanza. La restricción (5) asegura que se respete la precedencia entre las operaciones. La restricción (6) asegura que cada máquina solo pueda procesar una operación a la vez. Las restricciones (7) y (8) aseguran que una operación no puede iniciar hasta que la máquina a la cual fue asignada esté disponible y hasta que su precedente no haya sido terminada. La restricción (9) establece la relación entre las máquinas asignadas y las disponibles para cada operación a asignar. La restricción (10) asegura que cada operación de cada trabajo solo pueda ser asignada a una máquina y en una posición. Las restricciones (11) y (12) aseguran que cada operación efectivamente se procese en una máquina y en la posición que fue asignada. Las restricciones (13)-(16) garantizan la no negatividad de las variables de decisión. Las restricciones (17)-(19) definen las variables como correspondientes como binarias.

V. RESULTADOS

Para el procesamiento del modelo se empleó el software GAMS versión 23.5.6 usando CPLEX y empleando un computador con procesador de 3.2 GHz y 4 GB de RAM. El tiempo máximo para obtener una solución óptima se definió en 3600 segundos.

Para las pruebas de desempeño del modelo se seleccionaron instancias con datos históricos de la empresa en las cuales el número de órdenes de producción (i , de acuerdo a la notación usada en el modelo) estuvo entre 3 y 20. El número de operaciones por orden y el número de máquinas (j y k , de acuerdo a la notación usada en el modelo) son 4 y 13 respectivamente debido a que como se muestra en la Fig. 1 todos los productos requieren 4 operaciones en su fabricación, y se cuenta con 13 máquinas en la planta. De las instancias seleccionadas se logró obtener el óptimo, en un tiempo computacional menor a 3600 segundos, en instancias de menos de 14 órdenes de producción. Para instancias más grandes solo se pueden obtener soluciones factibles en razón a que el FJS es un problema NP-Hard por su naturaleza combinatoria lo que causa que el número de variables y restricciones crezcan de forma exponencial. En la tabla 2 se presenta una comparación de los resultados obtenidos para las instancias seleccionadas.

Tabla 2. Resultados de la aplicación del modelo

i	Tiempo CPU (s)	Total de trabajos tardíos	Tipo de solución
3	0.04	0	Óptima
4	0.11	0	Óptima
5	0.9	0	Óptima
6	5.2	1	Óptima
7	18	2	Óptima
8	42	2	Óptima
9	75	1	Óptima
10	100	2	Óptima
11	180	2	Óptima
12	580	3	Óptima
14	1200	3	Óptima
16	3600	6	Factible
17	3600	5	Factible
18	3600	7	Factible
20	3600	6	Factible

Fuente: Autores

En la Fig. 3 se presenta el resultado de una instancia ficticia (por motivos de confidencialidad no se presentan datos reales) con 4 trabajos y fechas de entrega de 113, 87, 104 y 129 unidades de tiempo respectivamente para cada trabajo. La solución del modelo garantiza 0 trabajos tardíos.

VI. CONCLUSIONES Y RECOMENDACIONES

El modelo propuesto permite a la empresa desarrollar mejores programas de producción en relación a minimizar el impacto de los incumplimientos en las fechas de entrega de las órdenes recibidas a los clientes. De igual manera se

debe resaltar que obtener estos resultados se logra en tiempos computacionales razonables, que antes del desarrollo del proyecto se obtenían de manera empírica y manual. Aunque los tiempos computacionales son relativamente pequeños sigue siendo necesaria la captura de los datos y la interpretación de los mismos, pero estos tiempos representan un costo bajo en relación a los ahorros que se logran gracias a la disminución de los costos asociados a los incumplimientos. El modelo permite obtener soluciones óptimas en instancias pequeñas e intermedias, en relación al número de órdenes, en instancias de mayor tamaño puede generar soluciones factibles debido a que por la naturaleza del problema los tiempos computacionales necesarios para obtener el óptimo pueden ser imprácticos. Para el caso de la empresa la implementación ha resultado viable debido al bajo número de órdenes de producción que maneja, pero en empresas más grandes puede resultar poco práctico su uso, por lo cual una línea futura de investigación es el desarrollo de otros métodos de solución tales como meta heurísticas que permitan manejar instancias de mayor tamaño en tiempos computacionales razonables.

Un resultado derivado de la implementación es que en la práctica los tiempos de entrega de los pedidos a los clientes puede realizarse previa negociación con el cliente, pero teniendo en cuenta que la fecha solicitada por ellos puede ser verificada a través de una corrida del modelo, con lo cual se puede llegar a fechas de entrega realistas y satisfactorias para ambas partes, con la consecuencia de que la disminución del número de trabajos tardíos puede ser mucho mayor gracias a este pre procesamiento.

El modelo se convierte en una aproximación a problemas en entornos MTO en los cuales los indicadores de desempeño asociados a tardanzas tienen mayor relevancia que los usados tradicionalmente en los modelos de optimización que usan por lo general el makespan como objetivo a minimizar. Este modelo puede servir como base para la formulación y solución de otro tipo de problemas de programación de producción, tanto teóricos como prácticos.

Otra posible línea futura de investigación es la inclusión de objetivos que apunten al mejoramiento de la productividad del sistema productivo, tales como el makespan o el tiempo total de terminación, a través de modelos multiobjetivo. De igual forma se pueden incluir en el modelo restricciones asociadas a tiempos de parada, programadas y no programadas, ventanas de tiempo para la entrega, entre otras, así como la consideración de tiempos de procesamiento estocásticos.

FINANCIACIÓN

Artículo derivado del proyecto de investigación titulado: "Diseño de modelo para la programación de producción para una empresa de empaques de caucho". Financiado por la Escuela Colombiana de Ingeniería Julio Garavito. Fecha de Inicio: 2015, Fecha de Finalización 2016.

Proceso	Máquina	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100	105	110	115		
Mezclado	M1	i3j1					i4j1																			
	M2	i2j1					i1j1																			
Preformado	M3											i1j2														
	M4																									
	M5																									
Laminado	M6					i2j2																				
	M7											i4j2														
Conformado Metal-Caucho	M8					i3j2																				
Vulcanizado	M9																i4j3									
	M10																		i1j3							
	M11											i2j3		i3j3												
Rebabado	M12												i2j4								i1j4					
	M13														i3j4					i4j4						

Fig. 3 Diagrama de Gantt del programa de producción de una instancia ficticia.
Fuente: Autores

REFERENCIAS

[1] C. Özgüven, Y. Yavuz, and L. Özbakir, "Mixed integer goal programming models for the flexible job-shop scheduling problems with separable and non-separable sequence dependent setup times," *Appl. Math. Model.*, vol. 36, no. 2, pp. 846–858, 2012. Disponible: <http://dx.doi.org/10.1016/j.apm.2011.07.037>

[2] M. Mastrolilli and L. M. Gambardella, "Effective Neighborhood Functions for the Flexible Job Shop Problem!," *J. Sched.*, vol. 3, no. 1, pp. 3–20, 1999. Disponible: [http://dx.doi.org/10.1002/\(SICI\)1099-1425\(200001/02\)3:1<3::AID-JOS32>3.0.CO;2-Y](http://dx.doi.org/10.1002/(SICI)1099-1425(200001/02)3:1<3::AID-JOS32>3.0.CO;2-Y)

[3] C. R. Scrich, V. A. Armentano, and M. Laguna, "Tardiness Minimization in a Flexible Job Shop: A Tabu Search Approach," *J. Intell. Manuf.*, vol. 15, no. 1, pp. 103–115, 2004. Disponible: <http://dx.doi.org/10.1023/B:JIMS.0000010078.30713.e9>

[4] Z. Wu and M. X. Weng, "Multiagent scheduling method with earliness and tardiness objectives in flexible job shops," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 35, no. 2, pp. 293–301, 2005. Disponible: <https://doi.org/10.1109/TSMCB.2004.842412>

[5] N. Zribi, A. El Kamel, and P. Borne, "Total tardiness in a flexible job-shop," in *IMACS Multiconference on "Computational Engineering in Systems Applications"*, CESA, 2006, pp. 1543–1549. Disponible: <https://doi.org/10.1109/CESA.2006.4281882>

[6] J. C. Tay and N. B. Ho, "Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems," *Comput. Ind. Eng.*, vol. 54, no. 3, pp. 453–473, 2008. Disponible: <https://doi.org/10.1016/j.cie.2007.08.008>

[7] A. Baykasoglu and L. Özbakir, "Analyzing the effect of dispatching rules on the scheduling performance through grammar based flexible scheduling system," *Int. J. Prod. Econ.*, vol. 124, no. 2, pp. 369–381, 2010. Disponible: <https://doi.org/10.1016/j.ijpe.2009.11.032>

[8] K. Thörnblad, "On the optimization of schedules of a multitask production cell," 2011. Disponible: <http://publications.lib.chalmers.se/records/fulltext/144904.pdf>

[9] J. C. Chen, C.-C. Wu, C.-W. Chen, and K.-H. Chen, "Flexible job shop scheduling with parallel machines using Genetic Algorithm and Grouping Genetic Algorithm," *Expert Syst. Appl.*, vol. 39, no. 11, pp. 10016–10021, 2012. Disponible: <https://doi.org/10.1016/j.eswa.2012.01.211>

[10] L. Lin, M. Gen, Y. Liang, and K. Ohno, "A hybrid EA for reactive flexible job-shop scheduling," in *Procedia Computer Science*, 2012, vol. 12, pp. 110–115. Disponible: <https://doi.org/10.1016/j.procs.2012.09.039>

[11] M. C. Gomes, A. P. Barbosa-Póvoa, and A. Q. Novais, "Reactive scheduling in a make-to-order flexible job shop with re-entrant process and assembly: a mathematical programming approach," *Int. J. Prod. Res.*, vol. 51, no. 17, pp. 5120–5141, 2013. Disponible: <http://dx.doi.org/10.1080/00207543.2013.793428>

[12] R.-H. Huang, C.-L. Yang, and W.-C. Cheng, "Flexible job shop scheduling with due window—a two-pheromone ant colony approach," *Int. J. Prod. Econ.*, vol. 141, no. 2, pp. 685–697, 2013. Disponible: <https://doi.org/10.1016/j.ijpe.2012.10.011>

[13] M. Mousakhani, "Sequence-dependent setup time flexible job shop scheduling problem to minimise total tardiness," *Int. J. Prod. Res.*, vol. 51, no. 12, pp. 3476–3487, 2013. Disponible: <http://dx.doi.org/10.1080/00207543.2012.746480>

[14] A. Sadrzadeh, "Development of Both the AIS and PSO for Solving the Flexible Job Shop Scheduling Problem," *Arab. J. Sci. Eng.*, vol. 38, no. 12, pp. 3593–3604, Dec. 2013. Disponible: <http://doi.org/10.1007/s13369-013-0625-y>

[15] K. Thörnblad, A.-B. Strömberg, M. Patriksson, and T. Almgren, "An efficient algorithm for solving the flexible job shop scheduling problem," in *25th NOFOMA conference proceedings, June 3-5 2013, Göteborg, Sweden*, 2013. Disponible: http://publications.lib.chalmers.se/records/fulltext/181796/local_181796.pdf

[16] G. Calleja and R. Pastor, "A dispatching algorithm for flexible job-shop scheduling with transfer batches: an industrial application," *Prod. Plan. Control*, vol. 25, no. 2, pp. 93–109, 2014. Disponible: <http://dx.doi.org/10.1080/09537287.2013.782846>

[17] Y. Liu, H. Dong, N. Lohse, S. Petrovic, and N. Gindy, "An investigation into minimizing total energy consumption and total weighted tardiness in job shops," *J. Clean. Prod.*, vol. 65, pp. 87–96, 2014. Disponible: <https://doi.org/10.1016/j.jclepro.2013.07.060>

[18] H. Na and J. Park, "Multi-level job scheduling in a flexible job shop environment," *Int. J. Prod. Res.*, vol. 52, no. 13, pp. 3877–3887, 2014. Disponible: <http://dx.doi.org/10.1080/00207543.2013.848487>

[19] K. Z. Gao, P. N. Suganthan, Q. K. Pan, T. J. Chua, T. X. Cai, and C. S. Chong, "Discrete harmony search algorithm for flexible job shop scheduling problem with multiple objectives," *J. Intell. Manuf.*, vol. 27, no. 2, pp. 363–374, Apr. 2016. Disponible: <http://doi.org/10.1007/s10845-014-0869-8>

[20] O. Faura Gatiús, "Flexible Job-Shop Problem minimizando los costes en función de la fecha de entrega y el instante de realización," *Universitat Politècnica de Catalunya*, 2016. Disponible: <http://hdl.handle.net/2117/98534>

[21] A. Türkyılmaz and S. Bulkan, "A hybrid algorithm for total tardiness minimisation in flexible job shop: genetic algorithm with parallel VNS execution," *Int. J. Prod. Res.*, vol. 53, no. 6, pp. 1832–1848, Mar. 2015. Disponible: <http://dx.doi.org/10.1080/00207543.2014.962113>

[22] Y. Demir and S. Kürşat İşleyen, "Evaluation of mathematical models for flexible job-shop scheduling problems," *Appl. Math. Model.*, vol. 37, no. 3, pp. 977–988, 2013. Disponible: <https://doi.org/10.1016/j.apm.2012.03.020>