

La búsqueda tabú contra los ordenamientos tradicionales aplicados al problema de la tardanza total

Martín Josué Castillo Montes¹, Adriana Mexicano Santoyo², Jesús Carlos Carmona Frausto³, Hilda Castillo Zacatelco⁴, Daniel Osvaldo Fernández Bonilla⁵

Resumen— El problema *single machine scheduling* es un problema de gran relevancia para las empresas de manufactura ya que para todas las empresas es vital entregar el producto a tiempo. Tradicionalmente, la mayoría de las empresas utilizan reglas de despacho tales como *Fist In Fist Out*, *Short Process Time* o *Earliest Due Date* para hacer más eficiente la programación de tareas, sin embargo, el uso de dichas reglas no siempre representa la mejor alternativa. El presente documento muestra una comparativa de la aplicación de las tres reglas de despacho previamente mencionadas y la metaheurística búsqueda tabú para resolver un conjunto de instancias de prueba bien conocido por la comunidad científica, del problema *single machine scheduling*. El objetivo es mostrar que una solución basada en métodos metaheurísticos expande en gran medida la posibilidad de encontrar mejores soluciones que las que pueden ser alcanzadas utilizando métodos tradicionales, sin sacrificar gran cantidad de tiempo.

Palabras clave—Búsqueda Tabú, tardanza total, calendarización de tareas.

Introducción

El *scheduling* (programación de tareas), consiste en generar una secuencia que permita minimizar algún parámetro de un conjunto de tareas a ejecutar, cada tarea cuenta con 2 características independientes, un tiempo de proceso, una fecha de entrega. Para los parámetros de eficiencia de producción, cómo tardanza y tiempo de flujo existen diferentes reglas de despacho, que buscan minimizar dichos parámetros un ejemplo serían: EDD (fecha más próxima de entrega) y SPT (tiempo más corto de proceso), dichas reglas pueden encontrar soluciones óptimas en ciertos casos, pero también pueden resultar ineficientes en otros, aunque lo más común es el uso de FIFO (primero en llegar primero en salir) en las empresas. Neelam Tyagi (2016) compara diferentes reglas de despacho para elegir la que minimice los trabajos tardíos, mostrando que EDD produce mejores resultados minimizando la tardanza, utilizándola cómo punto de partida para un algoritmo *Branch and Bound* demostrando cómo el uso de algoritmos más complejos permite explorar diferentes soluciones mediante el uso de la combinatoria, estas estrategias pueden obtener mejores resultados en comparación de las heurísticas simples cómo las reglas de despacho, en este trabajo se muestra una comparativa entre búsqueda tabú y diferentes reglas de despacho para mostrar la eficiencia entre los diferentes métodos.

El problema de la Tardanza Total

El problema de la tardanza total se consiste en un conjunto de n trabajos que se procesan uno a la vez en una sola máquina, dónde cada trabajo j ($j=1, \dots, n$), tiene disponibilidad en $t_j(0)$, tiempo de proceso p_j , peso w_j y fecha de vencimiento d_j puede calcularse para j el tiempo de terminación C_j y la tardanza $T = \max\{0, C_j - d_j\}$ (Graham et. Al. 1979). El objetivo es reducir la tardanza total, que es la sumatoria de las tardanzas dadas por los trabajos que se terminan después de su fecha de entrega o vencimiento, la función objetivo viene dada por la ecuación 1. C_j corresponde a la sumatoria de el tiempo de proceso de la tarea j más el tiempo de proceso de las tareas que le preceden.

$$f = \sum_{j=1}^n w_j [C_j - d_j]^+ \quad (1)$$

¹ Martín Josué Castillo Montes es alumno del Tecnológico Nacional de México / Instituto Tecnológico de Cd. Victoria, Tamaulipas. ingmc@outlook.com

² Adriana Mexicano Santoyo es profesora del Tecnológico Nacional de México / Instituto Tecnológico de Cd. Victoria, Tamaulipas. mexicanao@gmail.com (autor corresponsal)

³ Jesús Carlos Carmona Frausto es profesor del Tecnológico Nacional de México / Instituto Tecnológico de Cd. Victoria, Tamaulipas. jcarmonaфраusto@gmail.com

⁴ Hilda Castillo Zacatelco es profesora de la Benemérita Universidad Autónoma de Puebla. hildacz@gmail.com

⁵ Osvaldo Daniel Fernández Bonilla es profesor del Tecnológico Nacional de México / Instituto Tecnológico de Cd. Victoria, Tamaulipas. odfb81@gmail.com

Para comprender la solución al problema se debe comprender el cálculo de la tardanza, la cual se calcula mediante la ecuación 1, a manera de ejemplo se muestra la solución de la siguiente instancia:

$$\begin{aligned} n &= 5 & d &= \{50, 45, 25, 22, 40\} \\ j &= \{1, 2, 3, 4, 5\} & w &= \{1, 1, 1, 1, 1\} \\ p &= \{10, 30, 15, 5, 20\} & t &= \{0, 0, 0, 0, 0\} \end{aligned}$$

La cual consta de 5 trabajos con su respectivo tiempo de proceso p y su fecha de entrega d , no se asigna penalización por lo que el peso w se asume en 1 y todas llegan en tiempo cero, es decir al mismo tiempo y puede atenderse cualquiera al inicio de la secuencia. La sustitución de la ecuación (1) para la secuencia sería:

$$\begin{aligned} f &= \{1 [10-50]^+\} + \{1 [40-45]^+\} + \{1 [55-25]^+\} + \{1 [60-22]^+\} + \{1 [80-40]^+\} \\ f &= +0 + 0 + 30 + 38 + 40 \\ f &= 108 \end{aligned}$$

Al sumar los valores positivos de los resultados anteriores se obtiene un valor de $f = 108$ lo cual representa el número de unidades de tiempo correspondiente al atraso de las tareas.

Trabajos Relacionados

En esta sección se muestran algunos de los trabajos relacionados con el problema de *scheduling*, al utilizar reglas de prioridad o despacho y búsqueda tabú que se han desarrollado para minimizar la tardanza. Así por ejemplo, Panwalkar et. Al. (1977), realizó una revisión de reglas de despacho con la finalidad de que sean utilizadas por investigadores y practicantes, listando más de 100 reglas en diferentes categorías clasificadas según su uso y en qué casos resultan aplicables para obtener buenos resultados. Tyagi (2016) realizó una comparativa de diferentes reglas de ordenamiento para ser utilizadas en un algoritmo *Branch and Bound*, con la finalidad de mejorar el tiempo de ejecución necesario para minimizar la tardanza total. Swamidass (2000) Explicó el uso de las diferentes reglas de prioridad y su objetivo de uso en la programación de tareas. Brucker (1998) presentó un libro sobre algoritmos de *scheduling* o programación de tareas, su trabajo cubre la clasificación de los problemas de programación de tareas, métodos de optimización combinatoria relevantes para los procedimientos de solución, algoritmos clásicos para resolver problemas para una sola máquina, máquinas paralelas etc. Laguna (1997) demostró métodos para resolver el problema *single machine scheduling*, en el cual utilizó búsqueda tabú para minimizar la tardanza total, utilizando diferentes heurísticas y mostrando una comparativa de los resultados obtenidos, en este trabajo se presenta una comparativa entre búsqueda tabú y 3 reglas de prioridad con la finalidad de mostrar su funcionamiento, limitantes y ventajas, para el problema de la tardanza total.

Solución por métodos tradicionales

A continuación se muestra a manera de ejemplo 3 ordenamientos por regla de prioridad con la instancia sintética mostrada en la sección anterior, para mostrar el comportamiento de cada uno para su comprensión y apreciar cómo modifican la tardanza y el tiempo de flujo, las reglas que se muestran son FIFO, SPT y EDD, las cuales son utilizadas por su facilidad de implementación.

First In First Out (FIFO)

El orden no se modifica y se atienden desde la tarea 1 hasta la tarea n , es el ordenamiento más común en cualquier sistema de producción.

Tabla 1 Ordenamiento FIFO

Trabajo (j)	Tiempo de proceso (p)	Fecha de entrega (d)	Tiempo de flujo (C)	Atraso (T)
1	10	50	10	0
2	30	45	40	0
3	15	25	55	30
4	5	22	60	38
5	20	40	80	40

Para poder calcular la tardanza se tiene que estimar el tiempo de flujo, que viene dado por el tiempo de proceso de la tarea j más el tiempo de proceso de las tareas que la preceden, para la tardanza se resta el tiempo de flujo C de la tarea j a su fecha de entrega d . En el cálculo de la tardanza total solo se consideran los valores positivos. Al sumar la tardanza de cada trabajo, se obtiene un valor de 108 de tardanza total, el cálculo se muestra en la sección previa.

Shortest Processing Time (SPT)

En la tabla 2 se aplica un ordenamiento SPT, que ordena de menor a mayor el tiempo de proceso, modificando el orden de los trabajos con la finalidad de reducir el tiempo de flujo. Al reducir el tiempo de flujo, se debe reducir la tardanza respecto a la fecha de entrega, lo cual puede apreciarse en las últimas 2 columnas de la tabla.

Tabla 2 Ordenamiento SPT

Trabajo (j)	Tiempo de proceso (p)	Fecha de entrega (d)	Tiempo de flujo(C)	Atraso (T)
4	5	22	5	0
1	10	50	15	0
3	15	25	30	5
5	20	40	50	10
2	30	45	80	35

La notación expandida del cálculo de tardanza sería la siguiente:

$$f = \{1 [5-22]^+\} + \{1 [15-50]^+\} + \{1 [30-25]^+\} + \{1 [50-40]^+\} + \{1 [80-45]^+\}$$

$$f = 0 + 0 + 5 + 10 + 35$$

$$f = 50$$

Earliest Due Date (EDD)

En la tabla 3 se muestra el ordenamiento EDD, que da prioridad al trabajo con fecha más próxima de entrega, este orden reduce el atraso generado y el número de tareas atrasadas 3, en las últimas 2 columnas puede apreciarse cómo se modifican los valores de tiempo de flujo y el atraso total.

Tabla 3 Ordenamiento EDD

Trabajo (j)	Tiempo de proceso (p)	Fecha de entrega (d)	Tiempo de flujo(C)	Atraso (T)
4	5	22	5	0
3	15	25	20	0
5	20	40	40	0
2	30	45	70	25
1	10	50	80	30

La notación expandida del cálculo de tardanza sería la siguiente:

$$f = \{1 [5-22]^+\} + \{1 [20-25]^+\} + \{1 [40-40]^+\} + \{1 [70-45]^+\} + \{1 [80-50]^+\}$$

$$f = 0 + 0 + 0 + 25 + 30$$

$$f = 55$$

En el ordenamiento EDD, puede apreciarse que se reduce el número de tareas atrasadas a 2, aunque el valor de la sumatoria de los atrasos es mayor al de SPT. Por tanto, puede observarse la limitante de las reglas de despacho, dado que el valor de tardanza puede mejorarse al realizar intercambios de posiciones, mediante un método combinatorio.

Solución mediante búsqueda Tabú

La búsqueda tabú, es un método desarrollado por Fred Glover, el cual recuerda movimientos recientes y los prohíbe por cierto número de iteraciones, con la finalidad de evitar óptimos locales. Tomando en cuenta la tabla número 1 como solución inicial donde $f=108$, para minimizar el valor de f , se generan intercambios de posiciones por pares, creando un espacio de soluciones, donde en cada intercambio se calcula el valor de la función para esa secuencia. De acuerdo al valor obtenido de los intercambios se escogen los mejores valores como candidatos a la nueva solución, escogiendo el mejor de ellos.

En la tabla 4 se muestra el funcionamiento de la búsqueda tabú en 3 iteraciones, la estructura contenida en la primera columna contiene el espacio de soluciones; compuesto por los intercambios de posiciones del trabajo i por el j , la columna f corresponde al valor obtenido de la ecuación 1 y la columna "valor de movimiento" muestra el valor actual de la función objetivo f . La columna estructura Tabú de la tabla 4, contiene la estructura de memoria tabú; la estructura color verde corresponde a la secuencia de la solución obtenida que en la primera iteración que sería la secuencia 1, 4, 3, 2, 5. La estructura color morado, corresponde a los movimientos marcados como tabú, que en la primera iteración sería el par intercambiado 2-4. Estos intercambios realizados se prohíben por 3 iteraciones y

se decreta su periodo en 1 con cada iteración. La tercera estructura muestra los movimientos candidatos y su valor, en la primera iteración se observa que el menor valor se obtiene del intercambio 2-4 y representa una mejor solución reduciendo en 48 unidades el valor de la función objetivo. La tercera columna muestra el número de iteración y el valor de f obtenido, el cual en la tercera iteración resulta menor que cualquiera de los tres métodos observados previamente.

Tabla 4 funcionamiento de la Búsqueda Tabú

Espacio de soluciones				Estructura Tabú		Valor de la secuencia obtenida
Tareas		f	Valor de movimiento			
i	j					
1	2	108	0			Iteración 1 $f=60$
1	3	83	-25			
1	4	75	-33			
1	5	123	15			
2	3	88	-20			
2	4	60	-48			
2	5	83	-25			
3	4	98	-10			
3	5	118	10			
4	5	123	15			
Tareas		f	Valor de movimiento			Iteración 2 $f=50$
i	j					
1	5	60	0			
1	3	55	-5			
1	2	88	28			
1	5	73	13			
4	3	63	3			
4	2	108	48			
4	5	108	48			
3	2	75	15			
3	5	75	15			
2	5	50	-10			
Tareas		f	Valor de movimiento			Iteración 3 $f=45$
i	j					
1	4	50	0			
1	3	45	-5			
1	5	53	3			
1	2	98	48			
4	3	53	3			
4	5	83	33			
4	2	123	73			
3	5	60	10			
3	2	80	30			

5	2	60	10		
---	---	----	----	--	--

Resultados Experimentales

En esta sección se muestran resultados de prueba obtenidos al aplicar reglas de ordenamiento y búsqueda tabú al conjunto de instancias WT40, para el problema *single machine scheduling*, las instancias fueron extraídas de la OR-Lib (Beasley, s.f.). Las pruebas se realizaron en un equipo portátil con procesador AMD A8, 4GB de RAM utilizando una rutina programada en C++.

El conjunto WT40 consta de 125 instancias que agrupan 40 tareas cada una, las cuales utilizan diferentes valores de tiempo de proceso C , fecha de entrega d y además un peso w asignado a cada tarea. Las variaciones en los valores propios de cada trabajo, resulta en que ciertas reglas se comporten mejor en ciertas instancias por lo que la regla EDD no resulta adecuada en todos los casos especialmente cuando varía la distribución de los valores de fecha de entrega, tiempo de proceso y el peso.

En la tabla 5 se muestra un extracto de 10 instancias extraídas del conjunto WT40, La columna OPT muestra los valores óptimos, éstos se encuentran en un archivo adjunto a los que contienen las instancias (Beasley, s.f.). Las columnas FIFO, SPT y EDD, muestran los valores obtenidos para cada ordenamiento, la columna Tabú, tiene los valores obtenidos por el algoritmo Tabú, a la derecha de cada ordenamiento: FIFO, SPT y EDD se anexó otra columna dónde se muestra la diferencia entre el valor obtenido por el algoritmo correspondiente respecto al valor óptimo.

Tabla 5 Resultados WT40

OPT	FIFO	Diferencia vs OPT	SPT	Diferencia vs OPT	EDD	Diferencia vs OPT	Tabú	Diferencia vs OPT
990	12006	11016	4650	3660	4030	3040	990	0
6955	47616	40661	26491	19536	23691	16736	7232	277
6324	46420	40096	20099	13775	33547	27223	7217	893
6865	39586	32721	17549	10684	23032	16167	6882	17
16225	44572	28347	32629	16404	34596	18371	16250	25
9737	58317	48580	30049	20312	44178	34441	10017	280
17465	79327	61862	34386	16921	67845	50380	18321	856
19312	66049	46737	45255	25943	50511	31199	19512	200
29256	105394	76138	41628	12372	75119	45863	29464	208
14377	71689	57312	27579	13202	37395	23018	14746	369

En la gráfica de la Figura 1 se muestran los valores obtenidos (eje “Tardanza Total”) de 10 instancias del conjunto WT40, cuando se aplicaron los ordenamientos FIFO, SPT, EDD y la búsqueda tabú a las diferentes instancias (eje “No de instancias de 40 trabajos”).

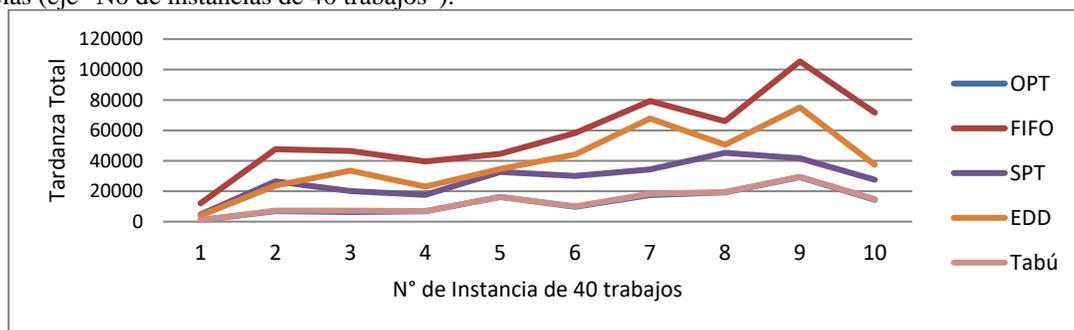


Figura 1 Valores de tardanza total obtenida por cada ordenamiento

Respecto a los valores óptimos puede verse que existe mayor diferencia entre las reglas de ordenamiento FIFO, STP y EDD en relación a la búsqueda tabú. Así mismo, puede observarse que a pesar de que se espera que EDD sea mejor que FIFO y STP, en ciertos casos EDD no obtiene el mejor resultado, esto es debido a la distribución de la tardanza y el tiempo de terminación así como el de los pesos asignados. Por lo tanto resulta una mejor opción el uso de un algoritmo de optimización combinatoria para obtener buenos resultados. La diferencia entre el valor óptimo y

el encontrado por cada algoritmo pueden observarse en el gráfico de la Figura 2 dónde el eje vertical muestra la diferencia existente entre el ordenamiento y el óptimo, mientras que el eje “instancias” corresponde a las 125 instancias del conjunto WT40.

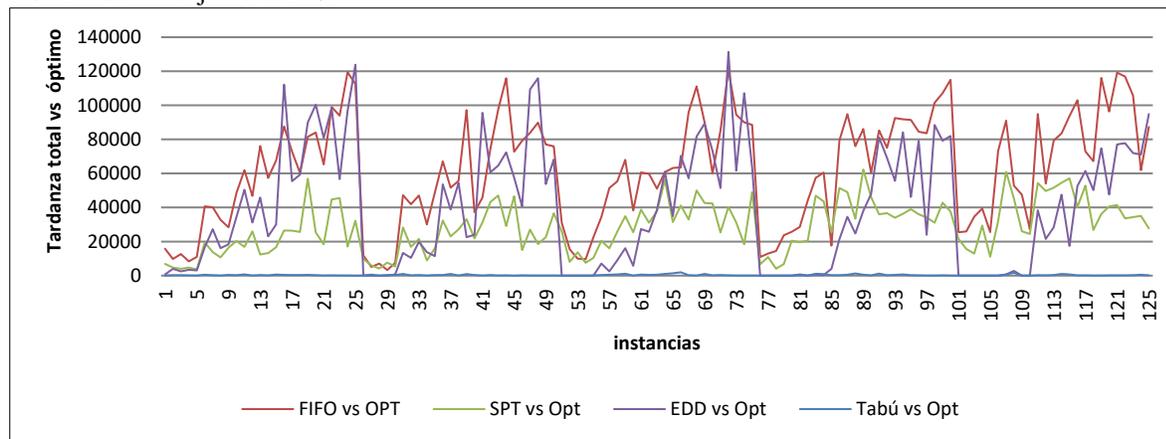


Figura 2 Resultados obtenidos del conjunto de instancias WT40 de 40 trabajos

Conclusiones

En el presente artículo se mostró cómo el uso de técnicas heurísticas resulta una mejor opción para resolver el problema de la tardanza total. Durante la experimentación se observó claramente cómo se obtienen mejores resultados respecto al uso de técnicas tradicionales como las reglas de despacho FIFO, STP y EDD. Lo anterior es debido a que la heurística hace uso de estrategias como el uso de memoria reciente, intercambio por pares y el uso del mejor resultado encontrado por la heurística en el espacio de soluciones, lo cual se realiza de manera eficiente en cuestión de milésimas de segundos ofreciendo un buen calidad en los resultados alcanzados. La aplicación de métodos heurísticos, hoy en día resulta beneficioso para las empresas de manufactura ya que lo que tradicionalmente era considerado bueno, ahora puede ser mejorado y con ello aumentar la confianza del cliente y volver eficiente la producción.

Bibliografía

- Tyagi, Neelam & Tripathi, R. & Chandramouli, A. (2016). Single Machine Scheduling Model with Total Tardiness Problem. *Indian Journal of Science and Technology*. 9. 10.17485/ijst/2016/v9i37/97527.
- Graham, R. L.; Lawler, E. L.; Lenstra, J.K.; Rinnooy Kan, A.H.G. (1979). "Optimization and Approximation in Deterministic Sequencing and Scheduling: a Survey" (PDF). *Proceedings of the Advanced Research Institute on Discrete Optimization and Systems Applications of the Systems Science Panel of NATO and of the Discrete Optimization Symposium*. Elsevier. pp. (5) 287–326.
- Panwalkar, S. S., and Wafik Iskander. "A Survey of Scheduling Rules." *Operations Research*, vol. 25, no. 1, 1977, pp. 45–61. JSTOR, www.jstor.org/stable/169546.
- Swamidass (2000). *Priority Scheduling Rules*. In: *Encyclopedia of Production and Manufacturing Management*. Springer, Boston, MA
- P. Brucker (1998): *Scheduling algorithms*, 2nd edition, Springer, Heidelberg
- Laguna Manuel, "Tabu search methods for a single machine scheduling", *Journal of Intelligent Manufacturing* 2(2), p 63-73, Abril 1991
- J E Beasley. (s.f), OR- Library. <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/wtinfo.html>.