

Programación de la producción en maquinas proporcionales con tiempos de preparación dependientes de la secuencia y fechas de entrega de trabajos

Julio Mario Daza-Escorcía

Universidad Autónoma del Caribe, Barranquilla, Colombia, juliomariodaza@hotmail.com

Mario Ferrer-Vásquez

Alfaisal University, Riyadh, KSA, mferrer@alfaisal.edu

Fabian Sanchez- Sanchez

Universidad Autónoma del Caribe, Barranquilla, Colombia, fabianchez@hotmail.com

RESUMEN

Este artículo presenta el diseño de una alternativa metaheurística para problemas multicriterio del tipo $Qm|r_j, s_j, k, d_j|Lex\{C_{max}, \sum_{j=1}^n C_j\}$. Para desarrollar el trabajo experimental propuesto se desarrollo un algoritmo metaheurístico de dos fases; i) en construir y ii) optimizar el scheduling. En la primera etapa se asignan los trabajos mediante la regla heurística de equilibrio (RHE). La segunda fase consiste en diversificar las soluciones obtenidas en la etapa previa mediante criterios de la metaheurística algoritmos genéticos (AG), luego intensificar estas a través de la técnica de optimización combinatoria búsqueda tabú (BT), logrando así escapar del óptimo local y generar un óptimo global. Esta alternativa se aplicó sobre instancias aleatorias y fue comparada con heurísticas similares utilizadas para solucionar el problema en mención, obteniendo así resultados satisfactorios.

Palabras claves: Programación de Operaciones, Maquinas Paralelas, Metaheurísticas.

ABSTRACT

This paper presents the design of an alternative metaheuristic for multi-type problems $Qm|r_j, s_j, k, d_j|Lex\{C_{max}, \sum_{j=1}^n C_j\}$. To develop the experimental work proposed metaheuristic algorithm development of two phases: i) to build and ii) optimize the scheduling. In the first stage are assigned jobs by balancing heuristic (RHE). The second phase is to diversify the solutions obtained in the previous stage using criteria metaheuristic genetic algorithms (GA), then enhance these through the combinatorial optimization technique tabu search (BT), thus escaping the local optimum and generate a global optimum. This alternative was applied to random instances and was compared with similar heuristics used to solve the problem in question, obtaining satisfactory results.

Keywords: Scheduling, Parallel Machine, Metaheuristics.

1. INTRODUCCIÓN

El presente trabajo propone una alternativa meta heurística para la solución del problema de secuenciación en maquinas paralelas proporcionales Qm con función objetivo considerada en un orden lexicográfico bicriterio $Lex\{C_{max}, \sum_{j=1}^n C_j\}$, donde la solución encontrada es optima para el criterio makespan C_{max} y mínima para el tiempo de flujo $\sum_{j=1}^n C_j$, las soluciones factibles generadas por esta alternativa están representadas por la asignación de los trabajos j en maquinas i , teniendo en cuenta restricciones concernientes a; el tiempo de liberación o disponibilidad r_j , el tiempo de alistamiento o ajuste s_{jk} y las fechas de entrega d_j . En particular, este

problema considerado en el contexto de teoría de optimización multicriterio puede formularse según (Graham et al, 1979) de la siguiente manera.

$$Q_m | r_j, s_{j,k}, d_j | Lex\{C_{\max}, \sum_{j=1}^J C_j\} (1)$$

Cuando se habla de máquinas paralelas, se distinguen tres casos generalizados en función de los tiempos de proceso de los trabajos en las máquinas. i) Cuando los tiempos de proceso de los trabajos son independientes de la máquina en la que se procesan, se habla de máquinas paralelas idénticas $p_{ij} = p_j$ para todo i . ii) Se denominan máquinas paralelas no relacionadas, dado un determinado par de trabajos y un determinado par de máquinas, no existe relación de proporcionalidad entre el tiempo de proceso de los dos trabajos en cada una de las dos máquinas (Cortés, García & Pastor, 2005). iii) Como máquinas paralelas proporcionales se entiende que, el tiempo de proceso de los trabajos depende de la velocidad de la máquina $p_{ij} = p_j/v_i$ donde p_j es por ejemplo un número de componentes en la operación a ser procesado, y v_i es el número de componentes tal que la máquina i pueda procesar por unidad de tiempo (Vincent & Billaut, 2006).

La resolución del problema planteado resulta de gran interés científico por su complejidad algorítmica perteneciente al conjunto de problemas NP-Completo, es decir que no se ha encontrado un algoritmo polinómico para solucionarlo (Graham et al, 1979, Chen, 2004), además, se trata de una situación que aparece con frecuencia en entornos industriales, y asumiendo ciertas simplificaciones puede ser adaptado a diversas condiciones particulares.

En la literatura encontrada existen heurísticas que proponen soluciones acertadas para problemas como $Q_m | pmtn | Lex\{C_{\max}, \sum_{j=1}^J C_j\}$, véase [3], donde preemption *pmtn* se refiere a la capacidad del sistema productivo de detener una actividad ya iniciada en un recurso. Normalmente este tipo de restricción posee gran relevancia en sistemas on-line, en los que se requiere evaluar la posibilidad de detener un trabajo para diseñar una nueva secuencia (Narducci, 2004).

En esta investigación se cambiará la restricción *pmtn* por las mencionadas anteriormente, ec (1), permitiendo así generar un gran aporte en lo que respecta a la manera de abordar esta clase de problemas mediante metodologías de optimización combinatoria y reglas de equilibrio para la construcción inicial de secuencias.

2. PLANTEAMIENTO DE LA ALTERNATIVA

El algoritmo metaheurístico planteado para resolver el problema descrito en la sección anterior contempla dos fases, construcción y optimización; en la primera fase se origina la solución inicial mediante la regla heurística denominada “Regla Heurística de Equilibrio” con el objeto de acercar el proceso hasta una solución elite (Gestal, 2008). Posterior a esto se realiza la segunda fase denominada optimización, esta toma varios criterios encontrados en la literatura en cuanto a métodos metaheurísticos para optimización combinatoria se refiere, como los Algoritmos Genéticos (Daza, 2007) y la Búsqueda Tabú (Daza-Escorcía et al, 2009, Daza, 2007, Glover, 1997, Glover & Melian, 2003), con los que se logra obtener soluciones no dominadas en un tiempo polinomialmente razonable. La alternativa planteada es como se muestra en la figura 1.



Figure 1: Alternativa Metaheurística Diseñada

FASE 1: CONSTRUCCIÓN

En esta fase se plantea como lograr asignar los trabajos en las maquinas X_{ij} , teniendo en cuenta que se cumplan a cabalidad las restricciones planteadas previamente, logrando así mejorar lexicográficamente el makespan C_{max} y el tiempo de flujo $\sum_{j=1}^n C_j$; inicialmente se utiliza la regla EST (Earliest Start Time first (Pinedo, 2008), la cual ordena los trabajos con respecto a su fecha de disponibilidad o liberación, en caso de existir algún empate entre los trabajos se secuenciará de acuerdo a su fecha de entrega como describe la regla EDD (Earliest Due Date first (Pinedo, 2008). Una vez determinado el orden inicial de los trabajos a asignar se procede a determinar que maquina puede ocasionar que a futuro se genere un menor C_{max} . Para lograr esto se ideó una regla de asignación heurística denominada “Regla Heurística de Equilibrio RHE”, la cual nos permite programar las actividades sabiendo de antemano cuanto sería el Work In Process WIP_{ij} de la maquina i al asignar un trabajo determinado j teniendo en cuenta todas las restricciones y variables que afectan el modelo planteado véase ec (1), al conocer previamente este índice generamos un equilibrio, debido que al secuenciar en una iteración inicial un trabajo j en la maquina i con menor WIP_{ij} , en la iteración siguiente el trabajo $j+1$ preferirá ser asignado en otra máquina $i+1$ diferente a esta para minimizar así el C_{max} parcial e ir equilibrando tanto el scheduling como el trabajo en proceso de cada máquina. En caso de existir trabajos que no logren ser asignados se secuenciarán al final, para prevenir atrasar las actividades posteriores $j+n$, estos trabajos al programarse lo harán con respecto a la regla RHE la cual es como sigue.

$$X_{ij} = \left\{ WIP_{ij} = \max\{C_j, r_j\} + \frac{P_j}{v_i} + s_{j,k} \mid WIP_{ij} \leq \max\{C_i\} \right\}$$

FASE 2: OPTIMIZACIÓN

Tomando como base la solución generada anteriormente se realiza la segunda fase de la alternativa planteada denominada optimización, que se subdivide en diversificación e intensificación del algoritmo, para escapar del óptimo local y generar un óptimo global. Al diversificar tomamos el operador genético llamado mutación de la meta heurística Algoritmos Genéticos, el cual hace transformaciones unitarias, creando soluciones (individuo o cromosoma) novedosas de tal manera que en la nueva población exista una gran diversidad de estos, mediante esta mutación se altera aleatoriamente la información genética (genes) de una solución dando origen a un

individuo diferente, el cual permite que el algoritmo no quede atrapado en un valle profundo (Isao,1996). El proceso es como sigue.

- Se escoge al azar un trabajo j , donde; $0 < j < n_trabajos+1$.
- Se escoge al azar un corrimiento C , donde; $0 < C < n_trabajos+1$.
- Se corre el trabajo j escogido C posiciones.

Para cada solución (individuo o cromosoma) generado por la mutación construida inicialmente se determina su correspondiente vecindario para intensificar así la búsqueda, tanto los vecindario como la evaluación de las soluciones generadas por mutación se realizan mediante principios de las metaheurística Búsqueda Tabú. Esta evaluación tiene en cuenta dos criterios de aspiración, aspiración por objetivo y por defecto. El proceso es como sigue (Daza, 2007, Daza-Escorcía et al 2009).

- Se escoge un trabajo j , donde; $0 < j < n_trabajos+1$.
- Se escoge un corrimiento $C=1$.
- Se corre el trabajo j escogido C posiciones.
- Se evalúa que la solución vecina generada es la candidata a mutar nuevamente en $Iter=t+1$ si $f(x_{ij}) < C_{max}$, (criterio de aspiración por objetivo), las demás soluciones mueren.
- Si ninguna solución vecina mejora el C_{max} , se selecciona la solución que más se acerco (criterio de aspiración por defecto), las demás soluciones mueren.

Para todos los casos la regla de evaluación que nos dice si una solución sobrevive y es mutada nuevamente es la regla RHE, véase ec (2).

FASE 2: ANÁLISIS Y EVALUACIÓN DEL DESEMPEÑO

Este algoritmo fue desarrollo en el lenguaje C++ en entorno Borland C++ Builder 6.0 Enterprise Edition y comparado con la aplicación Legin v2.4 (Pinedo,2008), arrojando excelentes resultados en cuanto a la calidad de la solución y la convergencia de esta en cada iteración tal como se observa en las tablas 1 y 2.

Tabla 1: Instancias Evaluadas

CARACTERÍSTICAS DE LAS INSTANCIAS EVALUADAS						
Trabajos j	Alternativa (RHE+AG+BT)			Legin 2.4 (SB)		
	Makespan	Tiempo de Flujo	Trabajos Atrasados	Makespan	Tiempo de Flujo	Trabajos Atrasados
5	41	115	2	41	135	3
10	89	504	7	89	567	9
20	103	1069	13	99	1260	18
25	132	1744	16	129	2080	24
30	166	2482	19	164	3188	29

Para hacer posible la comparación de la alternativa algorítmica propuesta con la herramienta comercial Legin se relajo el problema en estudio definiendo para cada instancia la variable v_i con valores iguales para cada máquina.

Con el objetivo de evaluar el desempeño de la metodología propuesta en esta investigación, comparamos su desempeño contra el procedimiento de búsqueda local Shifting Bottleneck, SB (para tal actividad se utilizo el software Legin). Los resultados de las 5 instancias evaluadas, pueden ser observados en la tabla 2.

Tabla 2: Resultados

CALIDAD DE LOS RESULTADOS			
Evaluación del Desempeño			
Makespan	Tiempo de Flujo	Trabajos Atrasados	Instancias
	0%	15%	33%
	15%	22%	1
	11%	28%	2
	15%	33%	3
	16%	33%	4
	22%	34%	5

La alternativa propuesta presentó un comportamiento estable en cuanto a la pertinencia y calidad de la solución. Esto implica que mediante la aplicación de un procedimiento heurístico combinado en dos fases, se garantiza una solución con mejor calidad que la originada por la heurística Shifting Bottleneck. La figura dos ilustra una solución obtenida con la herramienta desarrollada en la investigación.

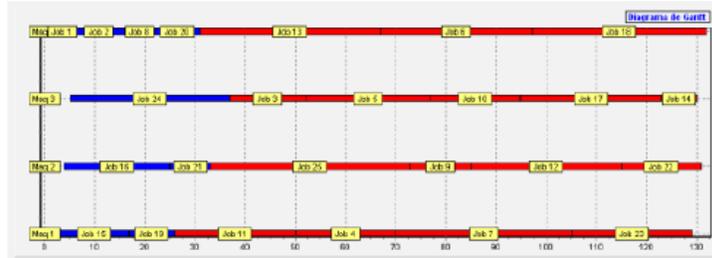


Figure 1: Solución Obtenida por QmScheduler

Cabe destacar el resultado de la quinta instancia, la cual presentó un desempeño sobresaliente con la alternativa propuesta, lo que permite fortalecer la validez y pertinencia del procedimiento. Es notable también, el poco tiempo de corrida empleado por nuestra propuesta para la obtención de resultados con una calidad superior a los obtenidos en periodos similares de tiempo, por la heurística que se utilizó como comparación.

FASE 2: CONCLUSIONES

Se ha presentado un modelo original para la minimización de la función objetivo referida anteriormente, en un entorno de máquinas paralelas proporcionales, abordadas mediante metodologías de optimización combinatoria como lo es la Búsqueda Tabú y los Algoritmos Genéticos, consiguiendo una solución factible al problema y la cual permite concluir que la aplicación combinada de procedimientos heurísticos y meta heurísticos, que implemente un proceso iterativo aleatorizado, sin la necesidad de ajustes de parámetros, puede presentar un comportamiento homogéneo y confiable ante diversas instancias de situaciones problemáticas reales de programación de operaciones en entornos industriales que necesiten soluciones asertivas en cortos periodos de tiempo.

Con el propósito de mejorar el desempeño del algoritmo se propone el empleo de otras técnicas de generación de soluciones iniciales que reemplacen los mecanismos de construcción con base a las reglas heurísticas, EST y EDD, como las de generación aleatoria con mejoramiento de soluciones a través de alguna heurística de búsqueda local. Así, como la utilización de diferentes metodologías de optimización combinatoria que generen nuevas estructuras de espacios de búsqueda, por ejemplo la Búsqueda Dispersa o los Sistemas de Colonias de Hormigas que incluyan colonias homogéneas y heterogéneas divididas según el tipo de estructura de espacio de búsqueda al que apliquen.

REFERENCIAS

- [1] R.L. Graham, E.L. Lawer, J.K. Lenstra and A.H.G. Rinnoy Kan, Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5:287-326 (1979).
- [2] R. Cortés, J. García y R. Pastor, “Programación de trabajos en máquinas paralelas con velocidad dependiente de la asignación de recursos limitados”, in: IX Congreso de Ingeniería de Organización, Gijón, 2 (2005).

- [3] T. Vincent and J.C. Billaut, Multicriteria scheduling, theory, models and algorithms, Springer, Ed. I., 2005, Ed. II, (2006).
- [4] Z. Chen, “Simultaneous job scheduling and resource allocation on parallel machines”, in: Annals of Operations Research, vol. 129, no. 1-4, 135-153 (2004).
- [5] F. Narducci, Programación de talleres intermitentes flexibles, por medio de la heurística del margen de tolerancia, Tesis de Maestría, Universidad del Norte, p33, (2004).
- [6] J.M. Daza Escorcía, J.R. Montoya y F. Narducci, Resolución del problema de enrutamiento de vehículos con limitaciones de capacidad utilizando un procedimiento metaheurístico de dos fases, Revista EIA, ISSN 1794-1237, N°. 12, 2009, págs. 23-38.
- [7] M.Gestal, Introducción a los algoritmos genéticos, Notas de Investigación, Universidad de la Coruña, Departamento de Tecnologías de la Información y las Comunicaciones (2008), 16 pp, (<http://sabia.tic.udc.es/~mgestal>).
- [8] J. Daza, Alternativa meta heurística de dos fases para el problema de ruteo de vehículos capacitados, Tesis de Pregrado, Corporación Universitaria de la Costa, p63, (2007).
- [9] Fred Glover and Manuel Laguna, tabu search, Kluwer Academic Publishers, 1997, 408 pp, ISBN 0-7923-9965-X. [10] F.Glover y B. Melián, Tabú Search, Revista Iberoamericana de Inteligencia Artificial. No.19 (2003), pp. 29-48 ISSN: 1137-3601. © AEPIA (<http://www.aepia.org/revista>).
- [11] Pinedo Michael, Scheduling theory, algorithms, and systems, Springer Science, 2008, 671 pp, ISBN 0-3877-78935-X.
- [12] O.Isao, A genetic algorithm for Job Shop problem using Job-based Order Crossover: Revista IEEE 2902-3, 547-552 (1996)..

Autorización y Renuncia

Los autores autorizan a LACCEI para publicar el escrito en las memorias de la conferencia. LACCEI o los editores no son responsables ni por el contenido ni por las implicaciones de lo que esta expresado en el escrito