

Cóctel de heurísticas para resolver problemas difíciles



Albert Corominas-Subías
Alberto García-Villoria
Rafael Pastor-Moreno

Dr. Ingeniero Industrial
Dr. Automática Avanzada y Robótica
Dr. Ingeniero Industrial

UNIVERSIDAD POLITÉCNICA DE CATALUÑA (UPC). Instituto de Organización y Control de Sistemas Industriales (IOC). Dpto. Organización de Empresas. Avda. Diagonal, 647 – 08028 Barcelona. Tfno: +34 93 4010724. alberto.garcia-villoria@upc.edu

Recibido: 13/12/2011 • Aceptado: 11/04/2012

DOI: <http://dx.doi.org/10.6036/4508>

COCKTAIL OF HEURISTICS FOR SOLVING HARD PROBLEMS

ABSTRACT

- In this work we introduce the concept of cocktail of heuristics. It refers to the possibility of solving a problem by all available quick heuristics instead of the practice of using only the heuristic that has the best average performance. The reasons are that the computing time of running all quick heuristics is affordable, and the best solution for a certain application of the problem can be obtained by any heuristic, even by the heuristic with the worst average performance. Thus, when a new quick heuristic is developed, it is not so important its average performance but how much the performance of the cocktail compound of the available heuristics is improved by adding this new heuristic.
- **Key Words:** combinatorial optimisation, evaluation of heuristics.

RESUMEN

En este trabajo se introduce el concepto de cóctel de heurísticas. Se refiere a la posibilidad de resolver un problema con todas las heurísticas rápidas disponibles frente a la práctica de utilizar sólo la heurística que tiene el mejor rendimiento promedio. Las razones son que el tiempo de ejecución de todas las heurísticas rápidas es asequible, y que la mejor solución para cada caso particular del problema puede ser obtenida por cualquier heurística, incluso por la heurística con el peor rendimiento promedio. De esta forma, cuando se desarrolla una nueva heurística rápida, no es tan importante su rendimiento promedio sino hasta qué punto el rendimiento del cóctel de heurísticas disponibles mejora al añadirle la nueva heurística.

Palabras Clave: optimización combinatoria, evaluación de heurísticas.

1. INTRODUCCIÓN

En el ámbito de la Ingeniería Industrial, tanto en el diseño como en la gestión de sistemas, suelen suscitarse problemas de optimización combinatoria; es decir, problemas en que se trata de hallar una solución óptima, en relación con un criterio (función objetivo) especificado, entre un conjunto de soluciones discreto. Dado que este conjunto, aunque finito, puede tener un número ingente de soluciones, la resolución exacta puede exigir tiempos de cálculo inasumibles en los procesos de decisión empresarial. En estos casos se recurre a heurísticas, es decir, a procedimientos que proporcionan soluciones satisfactorias en tiempos de cálculo aceptables, a costa de no garantizar que se alcance la solución óptima. Muchos de los trabajos publicados en la literatura científica relativa a la optimización consisten en propuestas o revisiones de procedimientos heurísticos y es usual que su evaluación se base principalmente, para un tiempo de cálculo dado,

en la calidad media (es decir, en la media de los valores de la función que se desea optimizar) de las soluciones encontradas para un conjunto de referencia de casos concretos del problema; por ejemplo, en [1-3].

La idea implícita en esta forma de evaluación es que para resolver un caso concreto de un tipo de problema solamente se utilizará una heurística. Y ciertamente, cuando no se dispone de ninguna guía para estimar qué heurística es la más adecuada para una aplicación específica, parece razonable elegir la heurística que tiene la mejor valoración media.

Sin embargo, la práctica de resolver un problema con un único procedimiento heurístico entre los disponibles sólo se justifica cuando el tiempo de cálculo es *elevado*. En cambio, si el tiempo de cálculo es reducido, una práctica mejor sería la de ejecutar varias o incluso todas las heurísticas disponibles y, entre las soluciones obtenidas con cada una de las heurísticas, tomar la mejor solución encontrada. Téngase en cuenta que los adjetivos *elevado* y *reducido* referentes a los tiempos de cálculo son relativos al tiempo disponible para resolver el problema en una empresa determinada.

El objetivo de este trabajo es introducir y dar a conocer el concepto de cóctel de heurísticas; es decir, la utilización de un conjunto de procedimientos heurísticos para resolver un problema. Para ejemplificar su uso y, en especial, su utilidad utilizaremos algunas heurísticas existentes para el problema conocido como SALBP-1 (*Simple Assembly Line Balancing Problem*, de tipo 1), que consiste en determinar el número mínimo de estaciones de una línea de montaje.

2. HEURÍSTICAS DE ENSAYO Y MÉTODO

2.1 HEURÍSTICAS PARA EL SALBP-1

El SALBP-1 consiste en minimizar el número de estaciones de trabajo necesarias para no sobrepasar un determinado tiempo de ciclo en una línea de montaje, en la que debe llevarse a cabo un conjunto de tareas con relaciones de precedencia entre ellas y con tiempos de proceso conocidos. Para más detalles sobre el SALBP-1, remitimos, por ejemplo, a [4]; y para su formulación matemática, a [5].

Para ejemplificar el concepto de cóctel de heurísticas usaremos varias heurísticas clásicas de la literatura para resolver el SALBP-1. Todas estas heurísticas se basan en el algoritmo que se describe a continuación. Inicialmente, se abre una única estación en donde ninguna tarea ha sido aún asignada. A continuación, iterativamente, se asigna a la estación en curso, de acuerdo con una *regla de prioridad*, una tarea *candidata*. Una tarea es candidata si todas sus tareas precedentes ya han sido asignadas y si puede asignarse a la estación en curso sin sobrepasar el tiempo de ciclo límite. Si no hay tareas candidatas disponibles (pero todavía faltan tareas por asignar), la estación en curso se cierra y se abre la estación siguiente. El algoritmo termina cuando todas las tareas han sido asignadas.

Según la regla de prioridad utilizada para seleccionar

la siguiente tarea a asignar, obtendremos una heurística diferente. Se usarán 12 reglas de prioridad diferentes, *R1* a *R12*, existentes en la literatura (presentadas en el Apéndice), que dan lugar a 12 heurísticas que llamaremos *H1* a *H12* y que corresponden a usar, respectivamente, las reglas de prioridad *R1* a *R12*. Estas heurísticas han sido seleccionadas con el único fin de ilustrar la propuesta del cóctel de heurísticas y no para mejorar la resolución del SALBP-1, lo que no forma parte del objetivo de este estudio.

En www.assembly-line-balancing.de hay disponibles 269 casos de referencia del SALBP-1, ampliamente utilizados en la literatura (p.e., [5,6]), y que también serán usados en este estudio. Estos ejemplares presentan un amplio rango de valores del número de tareas (desde 7 hasta 297), tiempo de ciclo (desde 6 hasta 17067) y densidad de las precedencias entre tareas, caracterizada por un parámetro que puede tomar valores entre 0 y 100 denominado *order strength* (desde 22,49 hasta 83,47).

2.2 CÓCTEL DE HEURÍSTICAS

Para un problema combinatorio difícil de revolver (por ejemplo, el SALBP-1) usualmente existen en la literatura varias heurísticas para resolverlo. Entre todas éstas, frecuentemente se resuelve mediante la heurística con el mejor rendimiento promedio (entiéndase por rendimiento promedio la media de los valores de la función objetivo – para el SALBP-1, el número de estaciones – obtenidos al resolver diferentes casos particulares del problema).

Por otro lado, en la última década se ha extendido el uso de las denominadas hiperheurísticas [7], que aprovechan la diversidad de heurísticas para obtener mejores soluciones, incluso aunque algunas de estas heurísticas tengan un rendimiento promedio bajo. Las hiperheurísticas se basan en seleccionar dinámicamente las heurísticas disponibles para construir o modificar la solución en curso. La selección dinámica se realiza de forma *inteligente* mediante unas reglas de selección, que pueden ser más o menos complejas [8].

Nuestra propuesta del cóctel de heurísticas también hace uso de un conjunto de heurísticas disponibles, pero es más sencilla, conceptual y prácticamente, que las hiperheurísticas. Consiste en lo siguiente: dado un caso particular de un problema que se desea revolver, se le aplican todas las heurísticas rápidas disponibles, obteniendo tantas soluciones, no siempre distintas, como heurísticas aplicadas; y de entre ellas, naturalmente, se retiene la mejor (o una de las mejores, si hay empate). La razón es que no hay garantía de que sea la heurística con el mejor rendimiento promedio la que proporcione la mejor solución para un caso particular del problema y que, por consiguiente, una de las heurísticas componentes del cóctel, aun con peores resultados en promedio, puede ser la que obtenga la solución más satisfactoria. Una heurística puede ser mala para la mayoría de casos particulares, pero puede ser la más adecuada para casos particulares poco frecuentes, con características determinadas. Claro está que en ningún caso la adición de una heurística al cóctel puede empeorar la

calidad de la solución obtenida por el conjunto de heurísticas que componen el cóctel.

Es decir, la calidad de la solución obtenida aumenta monótonamente con el número de heurísticas incluidas en el cóctel. Sin embargo, hay que considerar también el incremento en el tiempo de cálculo y, sobre todo, la limitación en cuanto a las heurísticas de que la empresa disponga o para las que esté dispuesta a invertir el esfuerzo necesario para implementarlas o adquirirlas.

Un cóctel de heurísticas para la resolución del SALBP-1 podría estar compuesto, por ejemplo, por las heurísticas *H1* a *H12*. Estas heurísticas pueden ser consideradas muy rápidas, ya que cada una necesita menos de 8 milisegundos, en promedio, para resolver cada uno de los 269 casos de referencia del SALBP-1. Este tiempo de cálculo es insignificante en el contexto del proceso de diseño de la línea, sea cual sea la complejidad de ésta. Por tanto, el tiempo necesario para lanzar el cóctel de heurísticas es perfectamente asequible.

3. RESULTADOS

La experiencia computacional se realizó en un ordenador personal con Intel Core 2 Duo E8500 a 3.16 GHz con 3.5 GB de RAM. Las heurísticas fueron implementadas en Java, y ejecutadas bajo el entorno *Java SE Runtime Environment* 1.6.21 de Oracle.

La Tabla 1 muestra el número medio de estaciones de las soluciones obtenidas por cada heurística y por el cóctel de heurísticas, para los 269 casos de referencia.

Podemos observar que *H1* es la heurística que proporciona,

	H1	H2	H3	H4	H5	H6	
Número medio de estaciones	22,513	22,595	22,981	22,743	23,520	23,260	Cóctel
	H7	H8	H9	H10	H11	H12	22,290
	22,788	23,186	22,535	22,535	22,751	22,881	

Tabla 1: Número medio de estaciones

en promedio, mejores soluciones. Sin embargo, como era de esperar, el promedio del cóctel es menor, lo que significa que las otras heurísticas devuelven mejores soluciones que *H1* en algunos casos (específicamente en el 19,33% –52 casos– de los 269). Y aunque el tiempo de cálculo utilizado por el cóctel se multiplica alrededor de 12 con respecto a la ejecución de una sola heurística, sigue siendo insignificante en toda empresa para la resolución de SALBP-1.

La propuesta de usar todas las heurísticas rápidas disponibles en la resolución de un problema real de optimización combinatoria pensamos que es, de hecho, obvia. Sin embargo, lo más habitual en la literatura sobre procedimientos heurísticos es la evaluación que puede llamarse *competitiva*; es decir, aquella que compara los resultados de la nueva heurística con cada uno de los de las

otras existentes. Por otro lado, y de acuerdo con la propuesta del cóctel de heurísticas, es más interesante y útil evaluar una nueva heurística en función de su contribución al cóctel de las heurísticas existentes en la literatura; es decir, la mejora de los resultados obtenidos con el cóctel al incorporarle la nueva heurística. Téngase en cuenta que esta contribución no tiene una relación directa con el comportamiento promedio de la nueva heurística.

Para ilustrar esta última afirmación usaremos el siguiente ejemplo. Vamos a suponer que las heurísticas existentes para resolver el SALBP-1 son *H1* a *H10* (es decir, supondremos que se desconocen las heurísticas *H11* y *H12*). El número medio de estaciones de las soluciones que se obtienen con el cóctel de estas 10 heurísticas es de 22,297. Y supongamos, ahora, que desarrollamos 2 nuevas heurísticas, *H11* y *H12*. Según la forma competitiva usual de evaluación, diríamos que *H11* es mejor que *H12*, ya que el resultado medio de *H12* es peor que el de *H11* (22,881 vs 22,751, ver Tabla 1) y, así, *H12* ya la descartaríamos. Sin embargo, si se añade *H11* al cóctel existente se obtiene un número medio de estaciones de 22,297 (es decir, *H11* no contribuye en absoluto a que el cóctel obtenga mejores soluciones); en cambio, si se añade *H12* al cóctel, el número medio de estaciones se reduce a 22,290.

4. CONCLUSIONES

En este trabajo se presenta una metodología de resolución que hemos llamado *cóctel de heurísticas* y que se recomienda aplicar en muchas situaciones: cuando se dispone de varias heurísticas rápidas (respecto al tiempo disponible) para

un problema difícil de resolver, se propone usar todas las heurísticas en lugar de utilizar solamente la que tiene el mejor comportamiento medio. La razón obvia (aunque raramente aplicada) es que para un caso concreto del problema, la mejor solución no es necesariamente la obtenida con esta mejor heurística, sino que puede ser obtenida con otra de peor comportamiento medio. Sin embargo, la evaluación competitiva usual en la literatura puede inducir a la mala práctica de usar sólo la heurística de mejor comportamiento medio en vez del cóctel de heurísticas. El problema usado para ilustrar el cóctel de heurísticas es el SALBP-1, un problema bien conocido en entornos de líneas de producción. No obstante, el cóctel de heurísticas podría ser usado de igual forma en otros problemas que se presentan en el diseño y la gestión de las cadenas de suministro, como, entre muchos

otros, la carga de vehículos, el diseño de rutas, el corte de materiales o la programación de operaciones en una planta industrial, en un almacén o en un aeropuerto. En estos casos, para cada problema, el cóctel estaría formado por aquellas heurísticas rápidas disponibles para resolverlo.

Actualmente, dado un problema, el énfasis usual a la hora de diseñar una nueva heurística es que esta tenga el mejor rendimiento promedio posible. En cambio, a partir de este trabajo, una futura extensión interesante sería cómo diseñar, para cualquier problema, nuevas heurísticas que mejor se complementen con el cóctel de heurísticas existentes. Tal como se ha visto, la contribución de la nueva heurística al cóctel existente no siempre es proporcional al rendimiento promedio de esta.

5. AGRADECIMENTOS

El presente estudio ha sido desarrollado subvencionado por el Ministerio Español de Ciencia e Innovación bajo el proyecto ENE2010-15509. Los autores también desean agradecer a los evaluadores anónimos sus valiosos comentarios, que han ayudado a mejorar este artículo.

APÉNDICE

Las 12 reglas de prioridad se muestran en la tabla de abajo (*R1* [9], *R2* [10], *R3* [11] y *R4 – R12* [12]), donde *TC* es el tiempo de ciclo, *n* el número de tareas, *NS_i* el número de tareas sucesoras de la tarea *i*, *NS_i* el número de tareas sucesoras inmediatas de la tarea *i*, *S_i (P_i)* el conjunto de tareas sucesoras (predecessoras) a la tarea *i*, *t_i* el tiempo de proceso de la tarea *i*, y $\lceil x \rceil$ el operador que devuelve el menor entero que es igual o mayor que *x*.

6. BIBLIOGRAFÍA

[1] Hartmann, S., Kolisch, R. "Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem". *European Journal of Operational Research*. 2000, vol 127, p. 394-407. [http://dx.doi.org/10.1016/S0377-2217\(99\)00485-3](http://dx.doi.org/10.1016/S0377-2217(99)00485-3)

[2] Aldowaisan, T., Allahverdi, A. "New heuristics for m-machine no-wait flowshop to minimize total completion time". *Omega*. 2004, vol 32, p. 345-352. <http://dx.doi.org/10.1016/j.omega.2004.01.004>

[3] Vallada, E., Ruiz, R., Gerardo, M. "Minimising total tardiness in the m-machine flowshop problem: A review and evaluation of heuristics and metaheuristics". *Computers & Operations Research*, vol 35, p. 1350-1373. <http://dx.doi.org/10.1016/j.cor.2006.08.016>

[4] Scholl, A. *Balancing and sequencing of assembly lines*. 2ª edición. Heidelberg: Physica-Verlag, 1999, ISBN: 978-3790811803.

[5] Pastor, R., Ferrer, L. "An improved mathematical program to solve the simple assembly line balancing problem". *International Journal of Production Research*. 2009, vol 47, p. 2943-2959. <http://dx.doi.org/10.1080/00207540701713832>

[6] Pastor, R., García-Villoria, A., Corominas, A. "Comparing ways of breaking symmetries in mathematical models for SALBP-1". *Assembly Automation*. 2011, vol 31, p. 385-387. <http://dx.doi.org/10.1108/01445151111172970>

[7] Burke, E.K., Kendall, G., Newall, J., Hart, E., Ross, P., Schulenburg, S. "Hyper-heuristics: An Emerging Direction in Modern Search Technology", Capítulo 16 en *Handbook of Metaheuristics*. 1ª edición. Eds. Glover & Kochenberger, Kluwer Academic Publishers, 2003, pp. 457-474, ISBN: 1-4020-7263-5.

[8] Burke, E.K., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., Woodward, J.R. "A Classification of Hyper-heuristic Approaches", Capítulo 15 en *Handbook of Metaheuristics*. 2ª edición. Eds. Gendreau & Potvin, Springer, 2010, pp. 449-468, ISBN: 978-1-4419-1663-1.

[9] Helgeson, W.B., Birnie, D.P. "Assembly line balancing using Ranked Positional Weight Techniques". *Journal of Industrial Engineering*. 1961, vol 12, p. 394-398.

[10] Moodie, C.L., Young, H.H. "A heuristic method of assembly line balancing for assumptions of constant or variable work element times". *Journal of Industrial Engineering*. 1965, vol 16, p. 23-29.

[11] Tonge, F.M. *A heuristic program for assembly line balancing*. Englewood Cliffs: Prentice-Hall Press, 1961.

[12] Talbot, F.B., Patterson, J.H., 1984, Gehrlein, W.V. "A comparative evaluation of heuristic line balancing techniques". *Management Science*. 1986, vol 32, p. 430-454.

R1. Máximo peso posicional	$PP_i = t_i + \sum_{i' \in S_i} t_{i'}$	R7. Mínima holgura	$CS_i - CI_i$
R2. Máximo tiempo de tarea	t_i	R8. Mínimo número de tarea	i
R3. Máximo número de tareas sucesoras inmediatas	NSI_i	R9. Mínima cota superior dividida por el número de tareas sucesoras	$\frac{CS_i}{NS_i + 1}$
R4. Máximo número de tareas sucesoras	NS_i	R10. Máximo tiempo de tarea dividido por la cota superior	$\frac{t_i}{CS_i}$
R5. Mínima cota inferior	$CI_i = \left\lceil \frac{t_i + \sum_{i' \in P_i} t_{i'}}{TC} \right\rceil$	R11. Máximo número de tareas sucesoras dividido por la holgura	$\frac{NS_i}{CS_i - CI_i}$
R6. Mínima cota superior	$CS_i = n + 1 - \left\lceil \frac{t_i + \sum_{i' \in S_i} t_{i'}}{TC} \right\rceil$	R12. Máximo peso posicional medio	$\frac{PP_i}{NS_i + 1}$

Reglas de prioridad utilizadas por cada heurística para resolver el SALBP-1