

Incorporando preferencias basadas en EMO a un algoritmo ACO multiobjetivo para el equilibrado de líneas de montaje considerando tiempo y espacio

Manuel Chica, Óscar Cordon, Sergio Damas

European Centre for Soft Computing
33600 Mieres, Asturias

Joaquín Bautista

UPC de Catalunya - Nissan Chair
Barcelona

{manuel.chica, oscar.cordon, sergio.damas}@softcomputing.es joaquin.bautista@nissanchair.com

Resumen

En este trabajo introducimos el uso de preferencias por parte del usuario en un proceso de búsqueda multiobjetivo para el equilibrado de líneas de montaje. Nos enfocaremos en la variante más realista del equilibrado de líneas de montaje considerando tiempo y espacio. Considera la minimización conjunta del número de estaciones y su área en una línea de montaje. El algoritmo de optimización multiobjetivo está basado en optimización por colonias de hormigas. El enfoque propuesto se basa en un esquema de preferencias heredado de la comunidad de optimización evolutiva multiobjetivo. Las preferencias utilizadas aquí están basadas en el conocimiento de los expertos de la planta industrial de Nissan y en variables económicas reales. La experimentación se realizó con datos reales del motor Nissan Pathfinder, obteniendo las soluciones más apropiadas para los decisores según cada uno de los seis escenarios de Nissan establecidos.

1. Introduction

Una línea de montaje está compuesta de un conjunto de estaciones de trabajo, en serie o paralelo, a lo largo de las cuales el producto se fabrica. Los productos de un sólo tipo (modelo único) o de distintos tipos (modelo mixto) van pasando por las estaciones de forma sucesiva, realizándose las distintas tareas de producción.

La configuración de la línea de montaje se ve afectada por la asignación óptima de subcon-

juntos de tareas a cada una de las estaciones, cumpliendo siempre restricciones de tiempo y precedencias. Brevemente, el problema es conseguir agrupar tareas minimizando la ineficiencia de la línea y respetando las restricciones de las tareas y estaciones. Este problema se llama equilibrado de líneas de montaje (ALB, en inglés) [1] y está ampliamente extendido tanto en la primera instalación de la línea como en sus reconfiguraciones. Es un problema de optimización combinatoria muy complejo (NP completo) de gran interés para los managers, jefes de planta y profesionales.

Bautista y Pereira propusieron recientemente un entorno de trabajo más realista, el problema del equilibrado de líneas de montaje considerando tiempo y área (TSALBP, en inglés) [2]. Esta familia de problemas considera una restricción adicional de espacio para asemejarse más a los problemas que existen en la industria. En este artículo tratamos de resolver la variante 1/3 del TSALBP, cuyo objetivo es el de minimizar el número de estaciones y su área para un determinado tiempo de ciclo de producto.

En anteriores trabajos [3] resolvimos satisfactoriamente la variante TSALBP-1/3 por medio de un algoritmo multiobjetivo basado en colonias de hormigas (MOACO, en inglés) [4], específicamente el algoritmo *Multiple Ant Colony System* (MACS) [5]. En [6], el trabajo anterior fue extendido incorporando información específica del problema suministrada por los expertos de planta en forma de preferencias *a priori* para discriminar entre configuracio-

nes de línea que eran prometedoras y que tenían los mismos valores de objetivos, es decir, el mismo número de estaciones y área. Basamos nuestro estudio en la idea de que bajo las mismas condiciones, un decisor de Nissan preferiría una solución con una configuración de estaciones más equilibrada, ya que provocaría menos conflictos de recursos humanos. De esta manera reducíamos el número de soluciones eficientes que se les pasaban al jefe de planta con sólo una configuración por cada conjunto de valores de los objetivos.

En este trabajo intentamos ampliar el diseño anterior incorporando preferencias en el espacio objetivo. Ahora reduciremos el tamaño del frente de Pareto, enfocándonos sólo en la parte del Pareto más interesante para el decisor. Estas preferencias cambiarán dependiendo de la ubicación final de la planta industrial (escenario). Definiremos y usaremos seis escenarios reales de todo el mundo para incorporar preferencias en el espacio objetivo dentro del algoritmo MACS. Las preferencias las definiremos usando metas o *goals*. El modelo de uso de tales preferencias será el propuesto por Deb en [7, 8] para optimización evolutiva multiobjetivo (EMO, en inglés).

Nuestro algoritmo MACS y el mecanismo de preferencias es probado con instancias TSALBP-1/3 y con una instancia real de la planta de Nissan. Esta instancia corresponde al proceso de fabricación del motor Nissan Pathfinder, ubicado en la planta de Nissan en Barcelona (España). Aparte, distintos escenarios y datos de costes reales han sido usados para probar los algoritmos.

En la Sección 2 explicamos el problema y algoritmo MOACO. Presentamos un breve estudio sobre cómo se pueden incorporar preferencias en algoritmos de optimización multiobjetivo en general y metaheurísticas en particular (Sección 3). En la Sección 4, los costes y variables industriales reales para utilizar preferencias son presentados. Explicamos el enfoque de Deb, diseño de experimentos y rendimiento de los algoritmos en la Sección 5. Finalmente, las conclusiones aparecen en la Sección 6.

2. Preliminaries

2.1. El problema del equilibrado de líneas de montaje considerando tiempo y área

La fabricación de un producto se divide en un conjunto V de n tareas. Cada tarea j requiere un tiempo operativo $t_j > 0$ que se determina en función de las tecnologías de fabricación y los recursos empleados. A cada estación k se le asigna a un subconjunto de tareas S_k ($S_k \subseteq V$), llamada carga de trabajo. Cada tarea j es asignada a una única estación k .

Cada tarea j tiene un conjunto directo de predecesores, P_j , las cuales tienen que estar terminadas antes de que la tarea comience. Estas restricciones se representan normalmente mediante un grafo de precedencias acíclico cuyos vértices son las tareas. Cada arco directo (i, j) indica que la tarea i debe haber finalizado antes de que la tarea j empiece. De esta forma, si $i \in S_h$ y $j \in S_k$, entonces debe cumplirse que $h \leq k$. Cada estación k tiene un tiempo de carga de trabajo $t(S_k)$ que es igual a la suma de las duraciones de las tareas asignadas a la estación k . El problema SALBP [1] se centra en agrupar tareas en estaciones de trabajo de una forma eficiente y coherente. Hay una gran variedad de procedimientos exactos y heurísticos para resolver este problema [9].

Existe una necesidad de introducir restricciones de área en el modelo. Dicha restricción puede ser definida como: la asociación del área requerida a_j a cada tarea j , y el área disponible A_k a cada estación k . Por simplicidad, esta área será idéntica para todas las estaciones: $A : A = \max_{\forall k \in \{1..n\}} \{A_k\}$. De esta manera, cada estación k requerirá una área $a(S_k)$ que es igual a la suma de las áreas de todas las tareas asignadas a la estación k . Se crea así una familia de problemas llamada TSALBP [2].

El problema genérico se define como: dado un conjunto de n tareas con sus atributo temporal t_j y espacial a_j ($1 \leq j \leq n$), y un grafo de precedencias, cada tarea puede ser asignada a una única estación tal que: (i) se satisfacen las restricciones, (ii) el tiempo de trabajo de la estación $t(S_k)$ no es mayor que el tiempo de ciclo c , y (iii) no se requiere un área de estación

$a(S_k)$ mayor que A .

El TSALBP tiene ocho variantes, dependiendo del criterio a optimizar: m (el número de estaciones), c (el tiempo de ciclo) y A (el área de las estaciones). Dentro de estas variantes encontramos cuatro problemas multiobjetivo. Nosotros nos centraremos en uno de ellos, el TSALBP-1/3. Consiste en la minimización del número de estaciones m y el área de las estaciones A , dado un valor fijo de tiempo de ciclo c . Para más información, acceder a la referencia [3].

2.2. El algoritmo MACS para el problema TSALBP 1/3

En esta sección mostramos brevemente nuestra anterior propuesta multiobjetivo basada en el algoritmo MACS. La descripción completa del algoritmo MACS se puede encontrar en [5], y nuestra propuesta en detalle en [3].

El algoritmo MACS se propuso como una extensión del algoritmo ACS [10] para hacer frente a problemas multiobjetivo. El MACS usa una única matriz de feromona, τ , y distintas funciones de información heurística, η_k (en nuestro caso, η^0 para el tiempo de cada tarea t_j , y η^1 para su área a_j). La regla de transición es modificada para atender a ambas funciones de información heurística. Debido a que el MACS es un algoritmo basado en el Pareto, el rastro de feromona se actualiza utilizando las soluciones actuales del conjunto de soluciones no dominadas (archivo de Pareto).

Ya que el número de estaciones no es fijo, usamos un enfoque constructivo y orientado a la estación (como se hace normalmente para el SALBP [9]) para cumplir las precedencias. De esta forma, nuestro algoritmo abrirá una estación e irá seleccionando tareas hasta cumplirse el criterio de parada. Después se abrirá una nueva estación.

La experimentación realizada mostró que el rendimiento es mejor si el algoritmo MACS es guiado sólo por información de feromona, sin hacer uso de la heurística. El rastro de feromona debe memorizar qué tareas son las más apropiadas para ser asignadas a cada estación. Por consiguiente, la feromona se asociará al par $(station_k, task_j)$, $k = 1..n$, $j = 1..n$,

siendo la matriz de feromona bidimensional. Usamos dos algoritmos greedy para obtener el valor inicial de feromona τ_0 .

También introducimos un mecanismo para cerrar estaciones de acuerdo a una distribución de probabilidad, dada por el ratio de llenado de la estación: $p(closing) = \left(\sum_{v_i \in S_k} t_i\right) / c$. Este mecanismo ayuda al algoritmo a obtener soluciones más diversas al cerrar las estaciones por medio de un proceso no determinístico. La probabilidad es calculada a cada paso constructivo, incrementándose progresivamente. Después, un número aleatorio es generado para decidir si se cierra la estación o no.

El algoritmo posee unos umbrales de llenado que son diferentes entre sí y están asociados a las hormigas. Cuanto mayor sea el umbral de la hormiga, más llena estará la estación (habrá menos posibilidades de cerrar la estación durante el proceso de construcción de la misma). De esta forma, la población de hormigas tendrá un comportamiento de búsqueda muy diverso, permitiendo al algoritmo explorar correctamente las distintas partes del Pareto.

3. Preferencias en optimización multiobjetivo y EMO

Existen muchos trabajos para incorporar preferencias del experto en la búsqueda en investigación operativa (OR, en inglés). Numerosas técnicas se han aplicado para resolver la toma de decisiones multicriterio (MCDM, en inglés): relaciones de superación (*outranking*), funciones de utilidad, o metas deseables [11, 12]. Una de las preguntas más importantes es el momento en el que se requiere al experto para dar su información de preferencia. Básicamente existen tres formas [12]:

- **Antes de la búsqueda** (enfoques *a priori*): la mayor dificultad y desventaja de esta familia es encontrar una información de preferencias preliminar y global.
- **Durante la búsqueda** (enfoques interactivos): estos enfoques han sido normalmente preferidos por los investigadores ya que el decisor puede tener mejores percepciones cuando conoce el conjunto de

elementos o soluciones que existen en una determinada situación.

- **Tras las búsqueda** (enfoques *a posteriori*): la mayor ventaja de incorporar preferencias después de la búsqueda es que no se requiere una función de utilidad para el análisis. Sin embargo, algunos problemas reales son demasiado grandes y complejos como para ser resueltos de esta forma.

Centrándonos en la EMO y otras metaheurísticas, la mayoría del trabajo existente está basado en enfoques *a posteriori* en donde la intervención del decisor se realiza cuando el algoritmo ha llegado ya a un conjunto de soluciones eficientes. Sin embargo, ésto a veces es problemático ya que el proceso de elegir entre un conjunto de soluciones posibles no suele ser una tarea trivial [13].

En los últimos años podemos encontrar enfoques EMO basados en el uso de objetivos antes del inicio del proceso de búsqueda (enfoques *a priori*) [8]. Así como el manejo de preferencias durante la búsqueda (enfoques interactivos, como por ejemplo los propuestos en [14] y en [15]), los cuáles están siendo cada vez más importantes. Un amplio estudio sobre la incorporación de preferencias en EMO se da en [16].

4. Escenarios de Nissan basados en costes de fabricación

Cuando un decisor tiene un conjunto de posibles soluciones (el conjunto de soluciones no dominadas del Pareto), uno de los criterios más usados es elegir una o un subconjunto de ellas teniendo en cuenta sus costes de desarrollo. Para definir los costes variables dentro del TSALBP, consideraremos dos tipos de costes operacionales:

- **Costes laborales:** asociados a los trabajadores (y consecuentemente al número de estaciones m). Se define como el coste laboral medio por trabajador en la fabricación de vehículos de motor. Hemos usado datos reales para este trabajo (tomados de la *International Labour Organi-*

*sation*¹) y dólares americanos como moneda. Otros indicadores relacionados con los costes laborales que se pueden usar también son: productividad, horas trabajadas, etc...

- **Coste del suelo industrial:** directamente asociado al objetivo del área A . En nuestro caso, los datos fueron recogidos del informe *2007 Industrial Space Across the World*². Las unidades usadas fueron dólares americanos por pies cuadrados en un año.

Obviamente, los costes operacionales no son fijos. Sus diferencias están sujetas a la localización en la que el manager quiera establecer la fábrica. De esta manera, una solución eficiente (configuración de línea de montaje) no estará suficientemente bien definida si no consideramos su posible localización. Ya que nuestro problema real pertenece a la planta industrial de Nissan, las localizaciones candidatas para la solución industrial puede perfectamente ser uno de los países en donde existen fábricas de Nissan (escenarios) (coloreados en la Figura 1). Hemos seleccionado seis de estos países para nuestro estudio, los cuáles, conjuntamente con sus costes reales³ se muestran en el Cuadro 1.

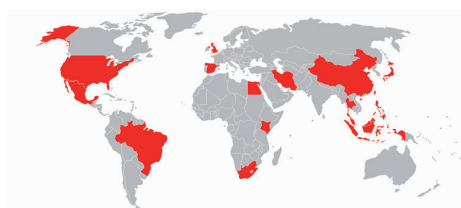


Figura 1: Localizaciones mundiales de fábricas de Nissan Motors.

De estos datos, los expertos industriales son capaces de definir la importancia de cada uno

¹<http://laborsta.ilo.org>

²Realizado por *Cushman & Wakefield Research*, <http://www.cushwake.com>

³La productividad se mide como producto interior bruto (PIB). Fuente *Groningen Growth and Development Centre* (Universidad de Groningen).

Cuadro 1: Costes laborales, productividad y costes de suelo industrial

País	Labour cost por hora (\$)	Productividad	Coste laboral compensado con productividad	Suelo industrial (\$/ft. ² año)
España	28.36	21.67	1.31	15.59
Japón	30.60	25.61	1.19	19.51
Brasil	8.79	7.99	1.10	10.05
Reino Unido	31.61	30.13	1.05	28.91
EE.UU.	30.39	35.29	0.86	11.52
México	6.57	9.24	0.71	5.02

de los dos objetivos para poder definir preferencias, o incluso fijar un conjunto de metas dependiendo del país en el que se quiera establecer la planta industrial. Por ejemplo, en aquellos países donde el coste del suelo industrial (respectivamente, el coste laboral) es muy caro, el objetivo m (respectivamente, el objetivo A) será más importante de minimizar y por tanto su peso será mayor.

5. Definición de preferencias mediante metas para los objetivos m y A

En esta sección introducimos primero el esquema de uso de preferencias basado en EMO, que será incluido en nuestro algoritmo MOACO. Más tarde, presentamos el diseño experimental y análisis de resultados.

5.1. Enfoque de Deb para incluir preferencias basadas en EMO

El objetivo de la programación por metas es encontrar una solución que minimice la desviación d entre el logro de la meta y el objetivo deseado t [17]. Estas metas pueden usarse como un conjunto de preferencias definidas por el experto. Puede haber distintos tipos de metas, entre las que hemos elegido cuatro de las más importantes, que son: *menor-igual-que* ($f(x) \leq t$), *mayor-igual-que* ($f(x) \geq t$), *igual-que* ($f(x) = t$) y *dentro de un rango* ($f(x) \in [t^l, t^u]$). Por ejemplo, podemos fijar el área total de una planta industrial como un área menor que unos metros cuadrados dados, o que nuestro número de estaciones tiene que estar, si es posible, dentro del intervalo

[100, 200]. En nuestros escenarios, algunas relaciones de preferencia pueden ser establecidas por un experto de Nissan, como está hecho en el Cuadro 2. No hemos considerado la relación *mayor-igual-que* ya que no tiene sentido en un problema de minimización como el nuestro.

Deb propuso una técnica para transformar la programación basada en metas en un problema de optimización multiobjetivo, que luego se puede resolver utilizando algoritmos EMO [7], [8]. La función objetivo del algoritmo EMO intentara minimizar la desviación absoluta a los objetivos alcanzados. Este enfoque sólo fue utilizado en [7] para transformar metas en objetivos. Sin embargo, se puede usar también para incorporar preferencias en cualquier algoritmo de optimización multiobjetivo, como nuestro MACS para el TSALBP-1/3.

La programación por metas puede ser modificada para incorporar preferencias a la función objetivo cambiando las funciones objetivo originales como sigue:

Meta	Función objetivo
$f_i(x) \leq t_j$	minimiza $\langle f_j(x) - t_j \rangle$
$f_i(x) \geq t_j$	minimiza $\langle t_j - f_j(x) \rangle$
$f_i(x) = t_j$	minimiza $ f_j(x) - t_j $
$f_i(x) \in [t_j^l, t_j^u]$	minimiza $\max(\langle t_j^l - f_j(x) \rangle, \langle f_j(x) - t_j^u \rangle)$

Aquí, el operador $\langle \rangle$ devuelve el valor del operando si es positivo, en cualquier otro caso devuelve 0. Hemos trasladado nuestras preferencias de metas para cada país del Cuadro 2 para modificar nuestras funciones objetivo siguiendo la conversión propuesta por Deb. Ya que nuestras metas son genéricas, nuestros seis escenarios iniciales han sido agrupados en tres: España, Japón y Reino Unido. Debido a sus

Cuadro 2: Metas para nuestros objetivos: número de estaciones m , y área A (hemos usado diferentes operadores relacionales para cada instancia)

Instancia del problema	España	Japón	Reino Unido
barthol2 (=, \leq)	$m = 51$ $A \leq 120$	$m = 60$ $A \leq 100$	$m = 68$ $A \leq 90$
barthold (\in , \leq)	$m \leq 8$ $A \leq 650$	$m \leq 14$ $A \leq 500$	$m \leq 16$ $A \leq 400$
weemag (\leq , \in)	$m \leq 30$ $A \in [56, 61]$	$m \leq 35$ $A \in [46, 51]$	$m \leq 45$ $A \in [40, 45]$
Nissan (=, =)	$m = 16$ $A = 5,7$	$m = 23$ $A = 3,8$	$m = 27$ $A = 3$

características económicas, España se enfoca en configuraciones de línea que otorgan más importancia a los costes laborales (objetivo m , el número de estaciones), Reino Unido necesita soluciones con menos coste de suelo industrial (objetivo A), y Japón está más interesado en un balance entre los dos objetivos.

5.2. Diseño de la experimentación

Tres instancias de problemas han sido usadas para la experimentación: **barthol2**, **barthold**, y **weemag**⁴. Además, hemos considerado un problema real correspondiente al proceso de ensamblado del motor Nissan Pathfinder de la planta de Nissan en Barcelona (España). Para más detalles sobre la instancia real de Nissan, el lector puede acceder a la página Web de Nissan Chair o a la referencia [2]. Los valores de los parámetros se muestran en el Cuadro 3.

Cuadro 3: Valores de los parámetros

Parámetro	Valor
Ejecuciones	10
Tiempo ejecución	900 segundos
PC Espec.	Intel Pentium TM D 2 CPUs a 2.80GHz
Sistema Operativo	CentOS Linux 4.0 GCC 3.4.6
Número de hormigas	10
β	2
ρ	0,2
q_0	0,2
Umbral de las hormigas	{0,2, 0,4, 0,6, 0,7, 0,9} (2 hormigas por valor de umbral)

⁴Instancias en <http://www.nissanchair.com/TSALBP>

5.3. Análisis de resultados

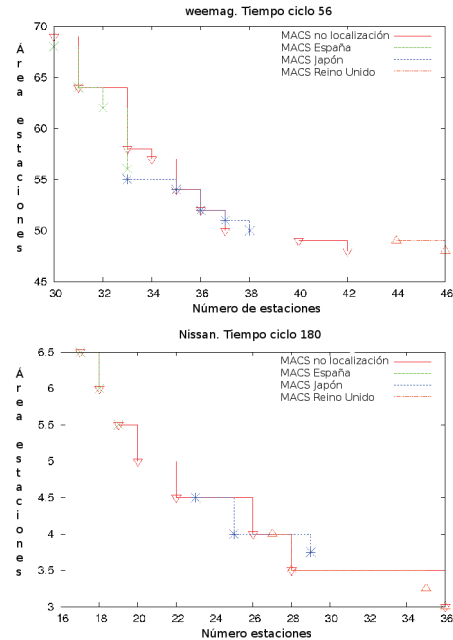


Figura 2: Frentes de Pareto para las instancias **weemag** y **Nissan** usando el enfoque de Deb para los distintos escenarios.

Los frentes de Pareto generados por el algoritmo MACS con metas para los diferentes escenarios se muestran en la Figura 2. Podemos ver cómo los algoritmos MACS para una determinada localización se comportan respecto

a los de otras localizaciones. Estas aproximaciones a los frentes muestran el uso de metas para los diferentes escenarios, obteniendo soluciones que pertenecen a zonas diferentes del Pareto. Así por ejemplo, las soluciones para el manager de una planta española tendrán el menor número de estaciones así como aquellas plantas para un experto británico tendrá el mínimo área de estación. En el caso del escenario japonés, las configuraciones con un buen balance entre número de estaciones y área son las que se obtienen. Cada escenario tendrá su propia parte del Pareto, distinta a las demás. Por consiguiente, podemos concluir que el esquema de preferencias funciona correctamente cuando son incorporadas al algoritmo MACS.

Ya que el MACS específico para cada localización se centra en una determinada parte del Pareto, sus soluciones no son dominadas por las demás y dominarán al resto de las soluciones de los otros algoritmos. Generalmente, la convergencia del algoritmo que incorpora preferencias basadas en metas es la misma que el MACS genérico, aunque a veces el conjunto de soluciones perteneciente al MACS genérico consigue una mejor convergencia.

Hemos considerado la métrica binaria C [16] para comparar las soluciones de Pareto y ver el rendimiento de las variantes de los algoritmos. Los gráficos de la Figura 3 son “boxplots” basados en la métrica C que comparan algoritmos dos a dos, calculando la dominancia de sus respectivos conjuntos de soluciones del frente de Pareto. Cada rectángulo contiene cuatro boxplots representando la distribución de valores de C para un cierto par de algoritmos en las cuatro instancias de problemas. Cada caja que se refiera al algoritmo A en las filas y al algoritmo B en las columnas nos da una idea de las soluciones de B cubiertas o dominadas por A ($C(A, B)$).

En la Figura 3 vemos cómo el algoritmo MACS para Japón obtiene un número bajo de soluciones no dominadas en comparación con los otros algoritmos. La razón es que este algoritmo expande la búsqueda en toda la región del frente de Pareto, cosa que no ocurre con las otras variantes. En general, parece que existe una ligera mejor convergencia del MACS sin

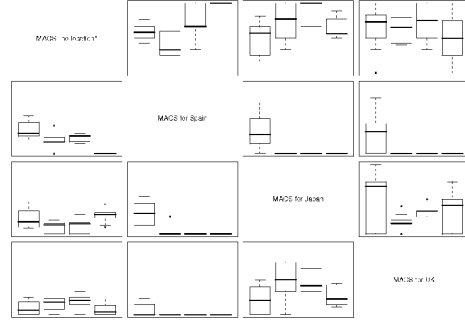


Figura 3: Valores de la métrica C representados con boxplots. Se comparan las variantes del algoritmo MACS para los diferentes escenarios usando el enfoque de Deb.

preferencias respecto al MACS que sí las usa.

6. Conclusiones

En este artículo hemos estudiado cómo incorporar preferencias del usuario en el espacio de búsqueda objetivo para resolver el problema TSALBP-1/3, basándonos en el conocimiento de la planta industrial de Nissan. Una propuesta MOACO anterior basada en el algoritmo MACS ha sido extendida y mejorada usando un esquema de preferencias heredado de la comunidad EMO. Este esquema consiste en la definición de unos objetivos para alcanzar sólo la región del frente de Pareto que tiene el equilibrio deseado entre número de estaciones m y área A . Hemos usado tres instancias biobjetivo del problema ALB y una instancia real procedente de la planta industrial de Nissan en España para seis escenarios diferentes de Nissan. La aplicación de este esquema avanzado de preferencias fue satisfactorio ya que ayudó al algoritmo a encontrar conjuntos de soluciones eficientes sólo en la zona de búsqueda en la que los jefes de planta estaban interesados.

Posibles líneas de investigación futuras son: (i) formas más avanzadas de incorporar conocimiento *a priori* en el algoritmo, y (ii) uso de procedimientos interactivos de incorporación de preferencias [15].

Agradecimientos

Trabajo financiado parcialmente por la Cátedra Nissan y el Gobierno Español a través de los proyectos PROTHIUS-II/III (DPI2007-63026 y DPI2010-16759) y TIN2009-07727, incluyendo fondos FEDER.

Referencias

- [1] A. Scholl, *Balancing and Sequencing of Assembly Lines (2nd. Edition)*. Physica-Verlag, Heidelberg, 1999.
- [2] J. Bautista and J. Pereira, "Ant algorithms for a time and space constrained assembly line balancing problem," *European Journal of Operational Research*, vol. 177, pp. 2016–2032, 2007.
- [3] M. Chica, O. Cordon, S. Damas, J. Bautista "Multi-objective, constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search," *Information Sciences* (in press, 2010).
- [4] C. García Martínez, O. Cordon, and F. Herrera, "A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP," *European Journal of Operational Research*, vol. 180, pp. 116–148, 2007.
- [5] B. Barán and M. Schaerer, "A multiobjective ant colony system for vehicle routing problem with time windows," in *21st IASTED International Conference*, Innsbruck (Germany), February 2003, pp. 97–102.
- [6] M. Chica, O. Cordon, S. Damas, J. Bautista, and J. Pereira, "Incorporating preferences to a multi-objective ant algorithm for time and space assembly line balancing," in *Ant Colony Optimization and Swarm Intelligence. LNCS 5217*. Springer, September 2008, pp. 331–338.
- [7] K. Deb, "Solving goal programming problems using multi-objective genetic algorithms," in *IEEE Congress on Evolutionary Computation (CEC 99)*, Washington (USA), 1999, pp. 77–84.
- [8] K. Deb and J. Branke, "Integrating user preferences into evolutionary multi-objective optimization," in *Knowledge Incorporation in Evolutionary Computation*, Y. Jin, Ed. Springer, 2005, pp. 461–476.
- [9] A. Scholl and C. Becker, "State-of-the-art exact and heuristic solution procedures for simple assembly line balancing," *European Journal of Operational Research*, vol. 168, pp. 666–693, 2006.
- [10] M. Dorigo and L. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.
- [11] V. Chankong and Y. Y. Haimes, *Multiobjective Decision Making Theory and Methodology*. North-Holland, 1983.
- [12] M. Ehrgott, "Multicriteria optimization." Springer, 2000, vol. 491.
- [13] K. Miettinen, *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.
- [14] S. Phelps and M. Koksalan, "An interactive evolutionary metaheuristic for multiobjective combinatorial optimization," *Management Science*, vol. 49, no. 12, pp. 1726–1738, 2003.
- [15] J. Molina, L. V. Santana, A. G. Hernández-Díaz, C. A. Coello, and R. Caballero, "g-dominance: Reference point based dominance for multiobjective metaheuristics," *European Journal of Operational Research*, 2008.
- [16] C. A. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-objective Problems (2nd edition)*. Springer, 2007.
- [17] C. Romero, *Handbook Of Critical Issues In Goal Programming*. Pergamon Press, 1991.