

# MAEB

VIII CONGRESO ESPAÑOL SOBRE METAHEURÍSTICA,  
ALGORÍTMOS EVOLUTIVOS Y BIOINSPIRADOS

ALBACETE, 8 al 10 FEBRERO, 2012

**EDITORES:**

José Antonio Gámez Martín  
José Miguel Puerta Callejón  
Francisco Parreño Torres  
Luis de la Ossa Jiménez

ISBN: 978-84-615-6931-1

Juan Gómez-Sanchis, Delia Lorente, Nuria Aleixos, José M. Martínez-Martínez, Pablo Escandell-Montero, Joan Vila-Francés y José Blasco. <i>Detección automática de podredumbres en cítricos mediante técnicas de aprendizaje automático y visión hiperespectral.</i> . . . . .	645
Francisco Romero-Porta, Pablo Mesejo, Oscar Ibáñez y Ana B. Porto-Pazos. <i>Optimización mediante Computación Evolutiva de la interacción Neurona-Astrocito en Redes Neurogliales Artificiales.</i> . . . . .	651
Álvaro Rodríguez, María Bermúdez, Daniel Rivero, Marcos Gestal y Jerónimo Puertas. <i>Seguimiento visual de peces en escalas de hendidura vertical.</i> . . . . .	659
Andrea Valsecchi, Sergio Damas y José Santamaría. <i>Una Aproximación al Registrado de Imágenes Médicas con Algoritmos Genéticos.</i> . . . . .	667
<b>SS10: Programación de Juegos/Videojuegos</b>	
Alberto Fuentes Sánchez, Carlos Cotta Porras y Antonio J. Fernández Leiva. <i>Optimizando la configuración de coches en juegos de carreras mediante computación evolutiva.</i> . . . . .	675
David Mariscal Fernández y Antonio J. Fernández-Leiva. <i>Una experiencia de diseño de controladores en juegos de carreras de coche mediante algoritmos evolutivos multiobjetivos y sistemas expertos.</i> . . . . .	683
Antonio Mora, Antonio Fernández-Ares, Pablo García Sánchez, J. J. Merelo y Carlos Fernandes. <i>Optimización evolutiva de bots para el juego Planet Wars.</i> . . . . .	691
Mariela Nogueira, Carlos Cotta y Antonio J. Fernández Leiva. <i>Apuntes del estudio sobre modelado, evaluación e incremento de la satisfacción del jugador.</i> . . . . .	699
<b>SS11: Problemas Dinámicos de Optimización y con Incertidumbre</b>	
Leticia Algarra y David A. Pelta. <i>DVRP-OCR: Un Método Cooperativo Basado en Heurísticas Simples para el DVRP.</i> . . . . .	705
Mario Cámara, Julio Ortega y Francisco De Toro. <i>Acelerando la optimización multiobjetivo para problemas dinámicos.</i> . . . . .	713
Jenny Fajardo Calderín, Juan R. González, David A. Pelta y Alejandro Rosete Suárez. <i>Comparación de estrategias de resolución de problemas de optimización dinámicos combinatorios.</i> . . . . .	721
Carlos M. Fernandes, Juan L. Laredo, Agostinho C. Rosa y Juan Julián Merelo. <i>Estudio y Optimización de un Operador de Mutación para Algoritmos Genéticos Basado en la Teoría de la Criticalidad Auto-Organizada.</i> . . . . .	729
Pavel Novoa Hernández, David A. Pelta y Carlos Cruz Corona. <i>Alcance de la evolución diferencial en ambientes dinámicos: un análisis empírico.</i> . . . . .	737
Briseida Sarasola, Karl F. Doerner y Enrique Alba. <i>Un algoritmo de búsqueda en vecindario variable para la asignación de rutas a vehículos con pedidos dinámicos.</i> . . . . .	745
<b>SS13: Metaheurísticas en Empresas y Producción</b>	
Joaquín Bautista, Alberto Cano y Rocío Alfaro. <i>Algoritmos GRASP para el MMSP-W con estaciones en serie y libre interrupción de operaciones.</i> . . . . .	751
Manuel Chica, Óscar Cordon, Sergio Damas y Joaquín Bautista. <i>Análisis del NSGA-II para el TSALBP-1/3 cuando existe variación de demanda en una producción mixta.</i> . . . . .	759
<b>SS14: Scheduling and Metaheurísticas</b>	
Jose Caceres Cruz, Alex Grasas, Helena R. Lourenço, Angel A. Juan, Mercè Roca y Rosa Colomé. <i>Aplicación de un algoritmo randomizado a un problema real de enrutamiento de vehículos heterogéneos.</i> . . . . .	767
Julio Mario Daza-Escorcía. <i>Programación de maquinas paralelas proporcionales con secuencia dependiente del tiempo de alistamiento usando la regla heurística de equilibrio.</i> . . . . .	775
Angel A. Juan, Helena R. Lourenço, Manuel Mateo, Alex Grasas y Alba Agustín. <i>Combinando Randomización Sesgada y Búsqueda Local Iterativa para Resolver Problemas de Flow-Shop.</i> . . . . .	779

Título:	VIII CONGRESO ESPAÑOL SOBRE METAHEURÍSTICAS, ALGORITMOS EVOLUTIVOS Y BIOINSPIRADOS.
ISBN:	978-84-615-6931-1.
Patrocina:	Ministerio de Ciencia e Innovación (TIN2011-15220-E).
Edita:	José A. Gámez, José M. Puerta, Francisco Parreño y Luis de la Ossa. Universidad de Castilla-La Mancha.

# Análisis del NSGA-II para el TSALBP-1/3 cuando existe variación de demanda en una producción mixta

Manuel Chica, Óscar Cordón, Sergio Damas y Joaquín Bautista

*Resumen*— En este trabajo introducimos la aplicación de un algoritmo genético multi-objetivo basado en el NSGA-II para resolver la variante más realista del equilibrado de líneas de montaje considerando tiempo y espacio, el TSALBP-1/3. El enfoque propuesto posee una representación y diseño de componentes avanzado que le permiten generar soluciones más óptimas al problema multi-objetivo. Además, este algoritmo se aplica al estudio de distintos programas de producción de la línea de montaje de motores de Nissan en Barcelona. De esta manera, se pretende realizar un análisis de los resultados del algoritmo cuando tenemos escenarios de producción-demanda variables. En particular, cómo afectarían las soluciones generadas a la configuración de la línea final, respecto al número de puestos y estaciones de trabajo y respecto al área de estas estaciones.

*Palabras clave*— Optimización Multi-objetivo, Algoritmos Genéticos, NSGA-II, Líneas de Montaje, Producción Mixta Variable.

## I. INTRODUCCIÓN

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 solo 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 subconjuntos 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.

Se han aplicado distintas metaheurísticas multi-objetivo de búsqueda global para resolver el TSALBP-1/3. Entre ellas, un algoritmo multi-objetivo basado en colonias de hormigas, MACS, [3] o un GRASP multi-objetivo [4]. En [5] se propuso un nuevo diseño avanzado de algoritmo genético multi-objetivo basado en el NSGA-II que ha obtenido los mejores resultados para el TSALBP-1/3 hasta el momento gracias a su adecuada representación y sus componentes avanzados. Este último algoritmo será, por su mejor comportamiento, el que utilizaremos en este trabajo.

Generalmente, todos los algoritmos anteriores devuelven un conjunto de soluciones pseudo-optimales para el diseño de una línea de montaje con productos o modelos fijos. Estas soluciones consisten en establecer el número de puestos o estaciones de trabajo y las operaciones que se deben realizar en cada puesto y, por otra, ofrecer una distribución en planta en la que queden definidas las dimensiones de las estaciones y los espacios adicionales que éstas requieren para el buen desempeño de los trabajos. Cuando la línea es de productos mixtos (productos distintos que se fabrican en secuencia), los tiempos de proceso de las operaciones son el resultado de un promedio entre los tiempos de proceso específicos de cada producto considerando también la demanda de éstos.

Sin embargo, con la anterior situación de líneas de productos mixtos surge el problema que proponemos en este trabajo. Esta problemática consiste en analizar la robustez de las configuraciones de línea de montaje recomendadas por el algoritmo basado en la metaheurística evolutiva multi-objetivo ante la variación del mix de producción. Concretamente, se propone: a) estudiar la variación del número de estaciones de trabajo y el área de éstas en función de distintos programas de producción-demanda y b), estudiar como afecta la variación del mix de producción a la composición de los puestos de trabajo: operaciones asignadas a cada puesto o estación.

M. Chica, O. Cordón y S. Damas pertenecen al European Centre for Soft Computing, Mieres (Asturias). O. Cordón también pertenece al Dpto. de Ciencias de la Computación e Inteligencia Artificial, E.T.S. Informática y Telecomunicación, Granada. J. Bautista pertenece a la UPC de Catalunya - Nissan Chair. E-mails: {manuel.chica, oscar.cordon, sergio.damas}@softcomputing.es y joaquin.bautista@nissanchair.com.

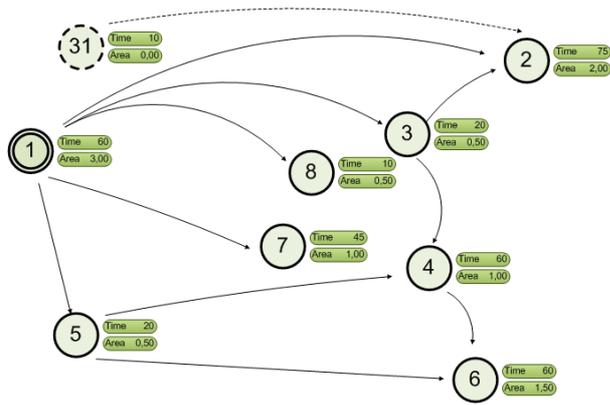


Fig. 1. Grafo de precedencias de 8 tareas de Nissan.

Para validar las configuraciones ofrecidas por el algoritmo de diseño avanzado basado en el NSGA-II en un entorno industrial, se ha tomado como caso de estudio una línea de producción y montaje de motores de la planta de *Nissan Spanish Industrial Operations* (NSIO), localizada en Barcelona.

En la Sección II se explica el problema TSALBP-1/3. Las características del diseño avanzado del NSGA-II se comentan en la Sección III. Presentamos el novedoso estudio de la variación de la demanda en mix de producción en la Sección IV para el caso de Nissan. En la Sección V se analizan los resultados obtenidos en la experimentación. Y por último, las conclusiones se reflejan en la Sección VI.

## II. 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 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 [6].

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  $k$ .

En la Figura 1 podemos observar el grafo de precedencias de las primeras 8 tareas de la instancia real de Nissan junto con sus datos de tiempo y área.

Por simplicidad, este área será idéntica para todas las estaciones:  $A : A = \max_{k \in \{1..m\}} \{A_k\}$ . 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 atributos temporales  $t_j$  y espaciales  $a_j$  ( $1 \leq j \leq n$ ), 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].

## III. UN DISEÑO AVANZADO DEL NSGA-II PARA EL TSALBP 1/3

A partir del algoritmo NSGA-II original [7] se ha diseñado una representación del individuo más adecuada así como unos operadores más efectivos para resolver el TSALBP-1/3. Las características principales del diseño avanzado del NSGA-II se describen en las siguientes secciones.

### A. Representación del individuo

La representación de la solución era el mayor problema de los métodos anteriores que se basaban en el NSGA-II para el TSALBP-1/3. Estos métodos estaban basados en propuestas clásicas de algoritmos genéticos para el SALBP. Como el SALBP era un problema mono-objetivo, no era estrictamente necesario representar una solución como una asignación de tareas a estaciones. Por contra, se solía utilizar una representación de orden específica para las tareas de la línea, realizándose una asignación posterior de tareas a estaciones [8].

Sin embargo, la anterior representación no es una buena opción para resolver el TSALBP-1/3 ya que sesga la búsqueda a sólo una estrecha parte del frente de Pareto, como se vió en el estudio experimental de [3]. En esta propuesta de diseño avanzado de NSGA-II sin embargo, se introduce una nueva codificación que explícitamente representa asignaciones de tareas a estaciones según el tiempo de ciclo disponible para la línea. Esta representación asegurará una correcta exploración del espacio de búsqueda para la optimización conjunta del número de estaciones y del área

en el TSALBP-1/3. Por último, también se facilitará la generación de soluciones factibles respecto a las relaciones de precedencia existentes en la línea de montaje.

La asignación de tareas a estaciones se realiza por medio de separadores, que ya han sido utilizados en las representaciones de individuos en otros problemas [9]. Los separadores son genes que no representan ninguna tarea específica y que se insertan entre la secuencia de tareas de la línea de montaje. De esta forma, los separadores podrán definir subgrupos de tareas que se asignarán a cada una de las estaciones. El número máximo de separadores para cada línea será de  $n - 1$ , siendo  $n$  el número de tareas, que corresponderá a una configuración de línea con  $n$  estaciones, cada una formada por una única tarea. Las tareas son codificadas por número enteros en el intervalo  $\{1, \dots, n\}$ , mientras que los separadores podrán tomar valores en el intervalo  $\{n + 1, \dots, 2 \cdot n - 1\}$ . De esta manera, el genotipo sigue teniendo una representación de orden.

El número de separadores incluidos en el genotipo es variable y dependerá del número de estaciones de la configuración de la línea. Entonces, el algoritmo NSGA-II tendrá que trabajar con una codificación de tamaño variable, aunque su representación de orden nos evitará tener que utilizar un mecanismo especial a este respecto. El tamaño máximo del individuo será de  $2 \cdot n - 1$  para permitir el mayor número posible de estaciones, según las tareas existentes. Por otro lado, y aunque la representación asegura que se cumplen las precedencias entre tareas, el tiempo de ciclo permitido para cada estación podría ser sobrepasado y es por ello por lo que se hace necesario el uso de un operador genético que asegure la factibilidad de la solución respecto al tiempo de ciclo.

En resumen, la representación de este diseño avanzado de NSGA-II posee dos claras ventajas. Por un lado, es clara y natural representando la solución. Por otro, el genotipo sigue siendo una permutación, lo que facilita la aplicación de los operadores genéticos.

### B. Operador de cruce

El diseño de un buen operador de cruce que combine las características relevantes de los padres y que genere descendientes factibles es la mayor dificultad que aparece cuando se diseña una representación que no es completamente estándar. Sin embargo, ya que nuestra representación está basada en una de orden, el operador de cruce puede estar basado en los operadores de cruce estándar para orden. Los operadores de cruce para orden más importantes de la literatura son el *partially mapped crossover* (PMX), *order crossover*, *order crossover # 2*, *position based crossover*, y el *cycle crossover* [10]. En nuestro caso hemos seleccionado el más extendido de todos ellos, el PMX, que ya ha sido utilizado en un algoritmo genético

mono-objetivo para el SALBP [8].

El operador de cruce PMX genera dos descendientes a partir de dos padres mediante el siguiente proceso: a) se seleccionan dos puntos de corte aleatorios, b) para el primer descendiente, los genes fuera de los dos puntos de corte son copiados directamente del primer padre, y c) los genes dentro de los dos puntos de corte son copiados al primer descendiente pero con el orden del segundo padre. El mismo mecanismo se sigue para el segundo descendiente pero intercambiando el orden de los padres.

Gracias a la codificación avanzada en la representación de la solución y el uso del operador de cruce de orden, los descendientes son factibles respecto a las precedencias. Sin embargo, es necesario asegurar que: a) no hay ninguna estación que exceda el tiempo de ciclo y b), no exista ninguna estación vacía en la configuración de la línea.

Por consiguiente, un operador reparador es aplicado a cada descendiente tras haber sido generado. El uso de este tipo de operadores está muy extendido en el área de computación evolutiva cuando se trabaja con problemas de optimización combinatoria con restricciones fuertes [11]. El operador reparador debe ser diseñado con cuidado para no sesgar la convergencia del algoritmo genético y para no perder información importante de los padres.

Además, y para incrementar la diversidad en la búsqueda y obtener un buen frente de Pareto, se ha utilizado un mecanismo reciente de inducción de diversidad para el NSGA-II. Este mecanismo se conoce como el cruce basado en similitud de Ishibuchi [12]. Esta componente del algoritmo le ayuda a tener una mejor convergencia. Para ver la descripción completa se remite al lector a la referencia [5].

### C. Operadores de mutación

Dos operadores de mutación se han diseñado específicamente para este diseño avanzado de NSGA-II para el TSALBP-1/3. Se han aplicado uniformemente a la población de individuos del algoritmo.

El primero de ellos, el operador de mutación de mezclado, se basa en reordenar una parte de la secuencia y de las asignaciones de tareas a estaciones que tiene cada individuo. Este operador funciona como sigue: tras elegir dos puntos aleatorios de un individuo, las tareas entre esos puntos se reordenan formando una nueva secuencia de tareas de tal forma que esta nueva secuencia sigue siendo factible respecto a las relaciones de precedencia. Los separadores existentes entre los dos puntos aleatorios se ignoran y se produce una nueva recolocación de tareas a estaciones por medio de una generación aleatoria de separadores.

Para realizar esta generación aleatoria de separadores entre los dos puntos aleatorios, se utiliza un mecanismo similar al utilizado en los umbrales del algoritmo MACS para el TSALBP-1/3 [3]).

Además, se ha utilizado otro operador de muta-

ción por división para obtener frentes de Pareto con más diversidad. Este operador de mutación por división intenta generar individuos descendientes que minimizan más el área de las estaciones. El operador funciona de la siguiente manera: a) se elige aleatoriamente una estación del fenotipo del individuo que tenga más de una tarea, b) se introduce un gen separador en dicha estación de forma aleatoria para separar la estación en dos nuevas estaciones.

#### IV. VARIACIÓN DE LA DEMANDA EN EL MODELO TSALBP-1/3 PARA PRODUCCIÓN MIXTA

Tal y como se comentó anteriormente, se ha considerado la línea de producción y montaje de motores de la planta de Nissan Spanish Industrial Operations (NSIO) como eje para la validación de los resultados del algoritmo genético multi-objetivo.

En dicha línea se fabrican hasta 9 tipos de motor distintos ( $p_1, \dots, p_9$ ) con diferentes destinos y características de ensablado: los 3 primeros tipos de motor son para vehículos 4x4; los tipos  $p_4$  y  $p_5$  son para furgonetas (*vans*); y los 4 últimos tipos tienen por destino los camiones comerciales (*trucks*) de tonelaje medio. En condiciones de equilibrio de demanda (idéntica demanda para los 9 tipos de motor) y un tiempo de ciclo de unos 3 minutos, la línea consta de 21 estaciones en serie con una longitud media de 4 metros cada una.

El número de tareas elementales para la fabricación de un motor asciende a 380, no obstante, para simplificar el problema, dichas tareas se han agrupado en 140 operaciones. Para cada tipo de motor, los tiempos de proceso de dichas operaciones ( $j = 1, \dots, 140$ ) varía. A modo de ejemplo, se han recogido los tiempos de proceso de 5 tareas en la Tabla I. También se recoge en dicha tabla el tiempo medio que dura cada operación teniendo en cuenta una demanda idéntica para cada tipo de motor (columna *t-medio*).

Lógicamente, la demanda de motores no es homogénea en el tiempo ni idéntica para los 9 tipos de motor dada una demanda global. Este hecho supone que la línea, aunque mantenga una capacidad de producción diaria en 270 unidades por ejemplo, deberá ser capaz de adaptarse a diferentes programas de producción en función de la demandas parciales de cada tipo de motor. Dicho de otro modo, el mix de producción para los 270 motores diarios no tiene porqué ser constante, lo que suscita el estudio del equilibrado de la línea ante distintos escenarios.

Para este trabajo se ha empleado un tiempo de ciclo efectivo  $c = 180s$ . Dicho ciclo equivale a la fabricación de 270 motores al día con una jornada efectiva de 13.5 horas (teniendo en cuenta los descansos) distribuidas equitativamente en dos turnos de trabajo.

Existen 23 programas de producción para los 9 motores y un día de trabajo. Cada programa, representativo de un escenario, se corresponde con un

conjunto de tiempos medios de operación, ponderados por la demanda de los 9 tipos de motor. Resumimos aquí las características de los 23 programas objeto de estudio (escenarios) agrupándolos en 7 categorías según la demanda para cada tipo de motor y siendo la producción total diaria de motores igual a 270 unidades. Así, para cada unas de las categorías elegiremos un programa representativo:

*Categoría-1* (programa representativo: #1): el ejemplar #1 presenta idéntica demanda para los 9 tipos de motor (30 unidades por tipo).

*Categoría-2* (programa representativo: #2): el ejemplar #2 presenta idéntica demanda para las tres familias de tipo de motor: 4x4, van y trucks (90 unidades por familia).

*Categoría-3* (programa representativo: #3): los ejemplares #3, #4 y #5 representan situaciones en que la demanda de una de las familias de tipo de motor es baja y la de las otras dos es alta e idéntica.

*Categoría-4* (programa representativo: #6): los ejemplares #6, #7 y #8 representan situaciones en que la demanda de una de las familias de tipo de motor es alta y la de las otras dos es mediana e idéntica.

*Categoría-5* (programa representativo: #9): los ejemplares #9, #10 y #11 representan situaciones en que la demanda de una de las familias de tipo de motor es alta y la de las otras dos es baja e idéntica.

*Categoría-6* (programa representativo: #12): los ejemplares #12 a #17 representan situaciones en que las demandas por familias siguen una progresión aritmética.

*Categoría-7* (programa representativo: #18): los ejemplares #18 a #23 representan situaciones en que las demandas por familias siguen una progresión geométrica.

Las producciones de los 7 programas elegidos, uno por categoría, se muestran en la Tabla II. En definitiva, la variación del mix de producción afecta a la duración media ponderada de cada una de las 140 operaciones que intervienen en el proceso. Por consiguiente, puede ser preciso el reequilibrado de la línea. Por ejemplo, la operación  $j = 1$  presenta unos tiempos de operación de 64.8, 61.2, 60, 54, 58.8, 55.2, 63, 66 y 57 segundos, para los tipos de motor  $p_1$  a  $p_9$ , respectivamente. Por otra parte, el mix de producción correspondiente al programa #12 para los 9 tipos de motor es 24, 23, 23, 45, 45, 28, 28, 27 y 27 unidades. En consecuencia, el tiempo medio ponderado de proceso de la operación 1 para el programa #12 es 59.44 ( $= (64,8 \times 24 + 61,2 \times 23 + \dots + 66 \times 27 + 57 \times 27) / 270$ ) segundos frente a los 60 segundos asociados al programa #1.

#### V. EXPERIMENTACIÓN Y ANÁLISIS

##### A. Parámetros de la experimentación

Se ha utilizado la instancia real de la planta de Nissan Spanish Industrial Operations (NSIO) localizada en Barcelona. Para más detalles sobre la instancia real de Nissan, el lector puede acceder a la página

TABLA I  
TIEMPOS DE PROCESO Y TIEMPO MEDIO PARA 5 TAREAS DE LA LÍNEA DE MOTORES NSIO.

no. tarea	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$p_6$	$p_7$	$p_8$	$p_9$	t-medio
1	64.8	61.2	60	54	58.8	55.2	63	66	57	60
5	19	19.6	18.4	20	21	20.4	18	21.6	22	20
9	20	19.6	19	18	20.4	18.4	21.6	21	22	20
3	18.4	18	20	19.6	19	21.6	21	20.4	22	20
8	9.8	9	10.5	10.8	9.5	11	9.2	10	10.2	10

TABLA II  
UNIDADES DE PRODUCCIÓN POR PROGRAMA PARA CADA TIPO DE MOTOR.

Familia	Motor	Programas de producción						
		MIX1	MIX2	MIX3	MIX6	MIX9	MIX12	MIX18
4 x 4	$p_1$	30	30	10	50	70	24	60
	$p_2$	30	30	10	50	70	23	60
	$p_3$	30	30	10	50	70	23	60
van	$p_4$	30	45	60	30	15	45	30
	$p_5$	30	45	60	30	15	45	30
trucks	$p_6$	30	23	30	15	8	28	8
	$p_7$	30	23	30	15	8	28	8
	$p_8$	30	22	30	15	7	27	7
	$p_9$	30	22	30	15	7	27	7

Web de Nissan Chair <sup>1</sup> o a la referencia [2]. Los valores de los parámetros del algoritmo multi-objetivo genético basado en el NSGA-II se muestran en la Tabla III.

TABLA III  
VALORES DE LOS PARÁMETROS

Parámetro	Valor
Ejecuciones	10
Tiempo ejecución	900 segundos
PC Espec.	Intel Pentium <sup>TM</sup> D 2 CPUs a 2.80GHz
Sistema Operativo	CentOS Linux 4.0 GCC 3.4.6
Parámetros específicos NSGA-II para el TSALBP-1/3	
Tamaño población	100
Valores $\gamma$ y $\delta$ para Ishibuchi	10
Probabilidad de cruce	0,8
Probabilidad de mutación	0,1
Valores $\alpha$ para la mutación de mezclado	{0,0,8}

### B. Análisis de resultados

En la Tabla IV se recogen las soluciones no dominadas resultantes de la fusión de los frentes de Pareto de las 10 ejecuciones del algoritmo genético

multi-objetivo para cada ejemplar representante de las 7 categorías de mix de producción.

En dicha tabla podemos observar los siguientes hechos:

- Para poner en funcionamiento la línea con 18 estaciones o puestos de trabajo se requiere una longitud por estación comprendida entre 5 y 6 metros; por tanto, con 6 metros se pueden satisfacer todos los mix analizados sin necesidad de rediseñar la línea. Con 5.5 metros de longitud por estación se satisfacen todos los mix a excepción de MIX12. Con la longitud 5 metros sólo es posible satisfacer el MIX1.
- Con 19 estaciones es suficiente una longitud de estación igual a 4.75 metros para satisfacer los 7 mix representativos.
- Con 20 estaciones bastan 4.5 metros para satisfacer todos los mix.
- Con 21 estaciones y 4 metros se satisfacen todos los mix excepto el MIX9 y el MIX12 que requieren, respectivamente, 22 y 23 puestos de trabajos si se desean mantener los 4 metros como longitud de estación.
- Si deseamos 3.75 metros como longitud de estación serán necesarios 24 puestos de trabajo para no tener que re-equilibrar la línea.
- Finalmente, si se desea que la longitud de cada estación sea de 3.5 metros harán falta 26 puestos de trabajo.

Por otra parte, estas mejores soluciones halladas por el algoritmo genético multi-objetivo basado en el NSGA-II se recogen en los frentes de Pareto de la Figura 2, correspondiendo nuevamente las soluciones representadas al conjunto combinado de frentes de

<sup>1</sup><http://www.nissanchair.com/TSALBP>

TABLA IV  
VALORES DE LOS OBJETIVOS DE NÚMERO DE ESTACIONES Y ÁREA PARA LAS SOLUCIONES NO DOMINADAS DE LOS FRENTES DE PARETO EN LOS 7 PROGRAMAS DE PRODUCCIÓN.

MIX1	MIX2	MIX3	MIX6	MIX9	MIX12	MIX18
(18, 5)	(18, 5.5)	(18, 5.5)	(18, 5.5)	(18, 5.25)	(18, 6)	(18, 5.5)
(19, 4.5)	(19, 4.5)	(19, 4.5)	(19, 4.75)	(19, 4.5)	(19, 4.5)	(19, 4.5)
(21, 4)	(21, 4)	(21, 4)	(20, 4.5)	(22, 4)	(23, 4)	(21, 4)
		(26, 3.5)	(21, 4)	(24, 3.75)		
				(25, 3.5)		

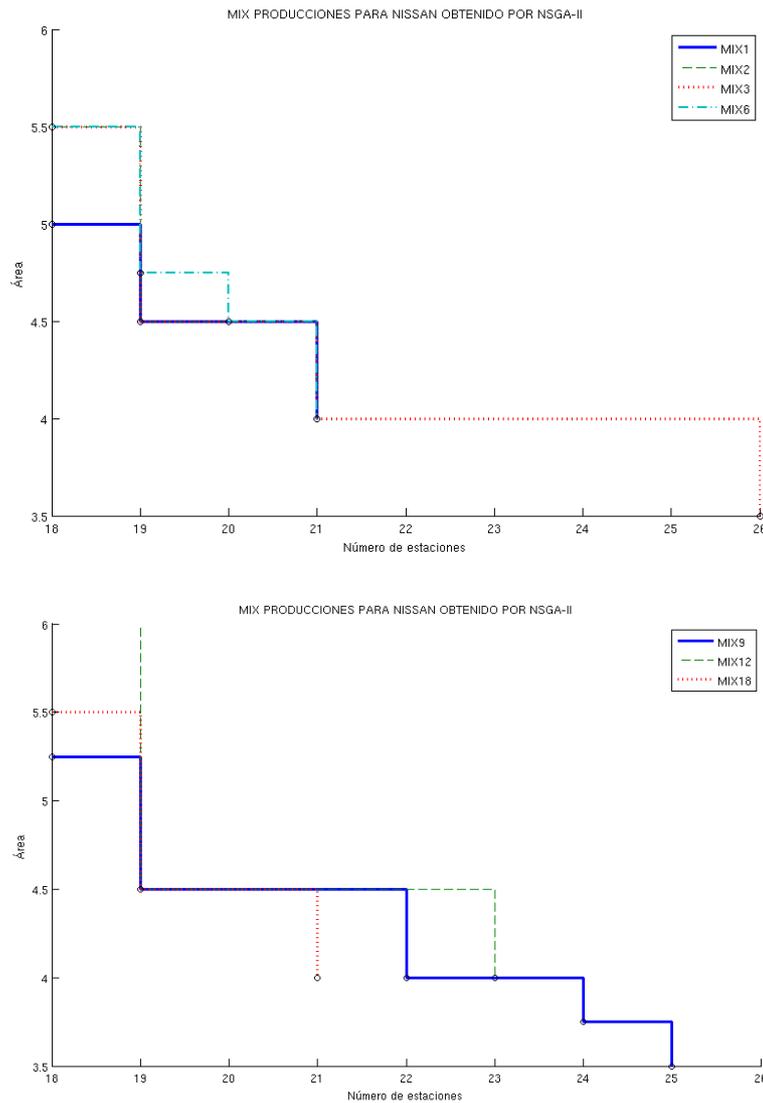


Fig. 2. Frentes de Pareto para las 7 categorías de mix de producción.

Pareto de las 10 ejecuciones por mix de producción.

A la vista de dicho conjunto combinado podemos afirmar que la envolvente superior de frentes de Pareto contiene las mejores soluciones que permiten afrontar todas las variaciones del mix de producción sin necesidad de proceder a un re-equilibrado de la línea. Es decir, las soluciones de la envolvente

podrían asimilar sin cambios de plantilla ni de distribución en planta todas las variaciones de la demanda vinculadas a las instancias analizadas.

Las soluciones de la envolvente superior de frentes de Pareto a las que nos referimos se muestran en la Tabla V.

Para la selección de una de estas soluciones en-

TABLA V

VALORES DE LOS OBJETIVOS PARA LAS 6 SOLUCIONES QUE FORMAN LA ENVOLVENTE SUPERIOR DE TODOS LOS PARETOS DE LOS 7 MIX DE PRODUCCIÓN.

No. solución	No. estaciones	Área disponible
$S_1$	18	5.5
$S_2$	19	4.75
$S_3$	20	4.5
$S_4$	23	4
$S_5$	24	3.75
$S_6$	26	3.5

tre todo el conjunto de ellas, el decisor deberá disponer de datos sobre costes de la mano de obra y costes vinculados a las instalaciones entre los que se incluyen los correspondientes al espacio requerido. Además, en caso que se quiera adoptar más de una solución, se deberá tener en cuenta el coste vinculado al proyecto y a la ejecución de la transformación o re-equilibrado de la línea para pasar de una alternativa a otra con más o menos puestos de trabajo y/o menos o más longitud disponible en cada estación.

Por otra parte, la envolvente inferior de frentes de Pareto de todos los mix de producción representa el conjunto global de soluciones no-dominadas, que son aquellas que permiten satisfacer al menos un mix de producción.

## VI. CONCLUSIONES

En este trabajo se ha aplicado un algoritmo genético multi-objetivo basado en el NSGA-II con un diseño y unas componentes avanzadas para el problema TSALBP-1/3. En concreto, se ha aplicado a los datos de la fábrica *Nissan Spanish Industrial Operations* (NSIO) localizada en Barcelona para un conjunto de 7 mix de producción representativos de una demanda variable en la planta. Estos mix de producción representan 7 escenarios tipo que varían su demanda en el montaje de motores para 4x4, *trucks* y *vans*.

Se han obtenido frentes de Pareto similares para los 7 mix de producción y se han evaluado las soluciones no dominadas que mejor se ajustan a todos los mix de producción con el objetivo final de no re-equilibrar la línea cuando la demanda fluctúa. En este sentido, los resultados del algoritmo son prometedores ya que se ha conseguido homogeneizar todos los frente de Pareto en 6 soluciones que tienen un número de estaciones y área optimizado pero a su vez, válido para todos los mix de producción de la planta.

Posibles líneas de investigación futuras son: (i) la introducción de costes y/o preferencias para seleccionar un grupo más reducido de soluciones finales, y (ii) aplicar el algoritmo a otra batería mayor de datos de mix de producción.

## AGRADECIMIENTOS

Los autores agradecen la colaboración prestada por NSIO (*Nissan Spanish Industrial Operations*) y la financiación parcial de este trabajo por la Cátedra Nissan UPC y el Gobierno Español a través de los proyectos DPI2010-16759 (PROTHIUS-III) 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, no. 3, pp. 2016–2032, 2007.
- [3] M. Chica, O. Cerdón, S. Damas, and J. Bautista, "Multiobjective, constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search," *Information Sciences*, vol. 180, pp. 3465–3487, 2010.
- [4] M. Chica, O. Cerdón, S. Damas, and J. Bautista, "A multiobjective GRASP for the 1/3 variant of the time and space assembly line balancing problem," in *Trends in Applied Intelligent Systems*, vol. 6098 of *Lecture Notes in Artificial Intelligence*, pp. 656–665. June 2010.
- [5] M. Chica, O. Cerdón, and S. Damas, "An advanced multi-objective genetic algorithm design for the time and space assembly line balancing problem," *Computers and Industrial Engineering*, vol. 61, no. 1, pp. 103–117, 2011.
- [6] 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, no. 3, pp. 666–693, 2006.
- [7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [8] I. Sabuncuoglu, E. Erel, and M. Tayner, "Assembly line balancing using genetic algorithms," *Journal of Intelligent Manufacturing*, vol. 11, no. 3, pp. 295–310, 2000.
- [9] A. M. Robertson and P. Willett, "Generation of equiprobable groups of words using a genetic algorithm," *Journal of Documentation*, vol. 50, no. 3, pp. 213–232, 1994.
- [10] P. W. Poon and J. N. Carter, "Genetic algorithm crossover operators," *Computers and Operations Research*, vol. 22, no. 1, pp. 135–147, 1995.
- [11] P. Chootinan and A. Chen, "Constraint handling in genetic algorithms using a gradient-based repair method," *Computers and Operations Research*, vol. 33, no. 8, pp. 2263–2281, 2006.
- [12] H. Ishibuchi, K. Narukawa, N. Tsukamoto, and Y. Nojima, "An empirical study on similarity-based mating for evolutionary multiobjective combinatorial optimization," *European Journal of Operational Research*, vol. 188, no. 1, pp. 57–75, 2008.