

## CSP DINÁMICO: UN ALGORITMO DINÁMICO PARA LA RESECUENCIACIÓN EN UN ALMACÉN DE LÍNEAS EN PARALELO.

María Valero Herrero<sup>1</sup>, Pablo Molina Morte<sup>1</sup>

<sup>1</sup>ROGLE. Dpto. de Organización de Empresas. Universidad Politécnica de Valencia. Camino de Vera S/N 46021 Valencia.  
[mavahe@upv.es](mailto:mavahe@upv.es) ; [pabmomor@etsii.upv.es](mailto:pabmomor@etsii.upv.es)

**Resumen:** En el presente trabajo se muestra el modelo utilizado para la resecuenciación en una línea de ensamblaje con mezcla de modelos donde un almacén con configuración en líneas en paralelo se utiliza para resecuenciar y minimizar la violación de restricciones de la línea de montaje.

**Abstract:** *This paper shows the model used for resequencing where a selectivity bank consisting of parallel line segments is applied to reshuffle a given initial sequence and rule violations are to be minimized.*

**Palabras clave:** MMAL, CSP, Resecuenciación, Problema dinámico, Líneas en paralelo

**Keywords:** *Mixed Model Assembly Lines, Car Sequencing Problem, Resequencing, Dynamic problem, Selectivity Banks*

### 1. Introducción.

Las líneas de ensamblaje dentro de plantas de fabricación de automóviles, son un ejemplo de las líneas de montaje con mezcla de modelos (MMAL- Mixed Model Assembly Line), en las que se ensamblan en una misma línea gran cantidad de modelos diferentes. Se obtiene así un flujo eficiente de producción a pesar de tener una cartera de productos muy diversa.

La secuenciación en las líneas de montaje es un problema de decisión importante y que ha recibido especial atención en la literatura (Boysen et al., 2009). El problema de secuenciación tiene como objetivo establecer el orden de aparición de los productos en la línea, teniendo en cuenta diferentes objetivos como son la nivelación en el consumo de componentes, la nivelación en la aparición de modelos en la línea, la carga de trabajo en las estaciones y el tiempo total de parada de la línea (Xiaobo et al., 1999).

Existen dos enfoques principales para el problema de la secuenciación en líneas de montaje: la secuenciación en líneas con mezcla de modelos (MMS *Mixed Model Sequencing*) y la Secuenciación de Coches (CSP *Car Sequencing Problem*). El enfoque MMS tiene por objeto evitar o minimizar la sobrecarga de trabajo dependiente de la secuencia, basado en un programación detallada que toma los tiempos de operación, los movimientos de los trabajadores, las fronteras de la estación y otras características operacionales de la línea (Bolat, 1997; Celano et al., 2004; Scholl et al., 1998).

Para evitar el gran esfuerzo de recopilación de datos que acompaña a la secuenciación de modelos mixtos, el enfoque CSP intenta minimizar la sobrecarga de trabajo dependiente de



Cicirello y Smith, 2003; Cicirello y Smith, 2004; Ding y Sun, 2004; Han y Zhou, 2010; Kittithreerapronchai y Anderson, 2003; Moon et al., 2005a; Nouyan, 2002; Spieckermann et al., 2004) aparecen en la literatura.

Sin embargo, pocas son las aportaciones sobre la resecuenciación mediante líneas en paralelo previas a la línea de ensamblaje. En la mayoría de ellas el objetivo es la reordenación, (Ding y Sun, 2004; Fournier y Agard, 2007; Jayaraman et al., 1997; Meissner, 2010) reordenan la secuencia para obtener la secuencia inicial. Pocos autores (Choi y Shin, 1997; Moon et al., 2005b; Valero-Herrero et al., 2011a; Valero-Herrero et al., 2011b) marcan como objetivo en la resecuenciación la minimización en la violación de las restricciones establecidas por la línea de montaje. En algunos casos (Inman y Schmeling, 2003) se puede dar un peso (penalización) diferente para cada una de las restricciones, minimizando por tanto la penalización total.

La resecuenciación mediante este tipo de almacenes, con varias líneas donde almacenar los productos, tiene a su vez dos problemas de decisión: la entrada y la salida del almacén.

En este sentido Moon (2005b) establece los siguientes criterios de entrada y salida. En primer lugar selecciona los coches que superan el tiempo máximo permitido en la primera posición de una línea, y que supere el tiempo máximo permitido en el almacén. De ellos, busca si hay algún coche que no viole restricciones. Entre aquellos que no violen, escoge el que tenga menor ratio de alisado. Se puede medir el alisado de los componentes o de los modelos, Moon (2005b) mide únicamente el alisado de los modelos, busca una secuencia en la que cada modelo sea fabricado de manera uniforme. Si no existe ningún coche que no viole ninguna restricción, escoge según el tiempo que llevan en la línea.

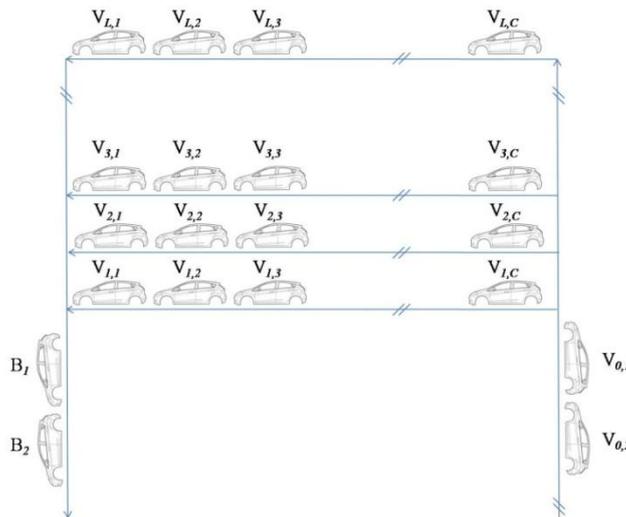
Choi y Shin (1997) asignan líneas a determinados productos. Si el producto no tiene línea asignada o si la línea que le corresponde está llena se introduce en la línea cuyo último producto sea menos parecido. El grado de similitud ( $\delta_i$ ) con cada línea  $i$  se establece según la cantidad de propiedades que coinciden entre el producto que entra ( $V_{o1k}$ ) con el último producto introducido en una línea ( $V_{iQik}$ ), según la siguiente expresión (Véase Tabla 1):

$$\delta_i = \frac{R}{k=1} V_{iQik} - V_{o1k}^2 \quad (1)$$

Para la salida distinguen entre restricciones de espacio absolutas, no absolutas y de nivelación. Para las restricciones de espacio no absolutas (aquellas que indican el espacio entre opciones debe ser mayor o igual que un número especificado), que son las que nos ocupan y las más utilizadas en la literatura y en la práctica, busca minimizar el número de violaciones. En caso de no violar, se sacaría el producto con menor regularidad dentro del almacén.

### 3. Modelo para la resecuenciación CSP en un almacén de líneas en paralelo.

El modelo planteado es de aplicación para un almacén de líneas en paralelo (Figura 2) utilizado para la resecuenciación, teniendo en cuenta restricciones de espacio, o también denominadas restricciones físicas (García-Sabater, 2001).



**Figura 2.** Almacén con L líneas en paralelo.

El modo usual de expresar estas restricciones, generalización del propuesto por Monden (1993), es:

*“No más de M coches de tipo k de cada N unidades consecutivas”.*

Esta misma restricción se podría aplicar como un control del mínimo, que aunque no estaría asociado a una restricción física, este tipo de restricciones se usan para garantizar la regularidad en la aparición de productos y componentes.

*“No menos de M coches de tipo k de cada N unidades consecutivas”.*

### 3.1. Notación

La notación utilizada en el desarrollo de los algoritmos se presenta en la Tabla 1.

**Tabla 1.** Notación

Índices	
$i$ (L):	Índice referido a las líneas del almacén.
$j$ (C):	Índice referido a la posición dentro de cada línea, siendo la primera posición la más cercana a la salida.
$k$ (NR):	Índice referido a las propiedades del producto. Cada propiedad va asociada a una restricción.
Parámetros	
$V_{0,j,k}$ :	Valor de la propiedad $k$ del producto situado en la posición $j$ de la línea previa al almacén. Toma valor 1 si tiene dicha propiedad. Toma valor 0 en caso contrario.
$V_{i,j,k}$ :	Valor de la propiedad $k$ en la posición $j$ de la línea $i$ del almacén. Toma valor 1 si tiene dicha propiedad. Toma valor 0 en caso contrario.
$B_{j,k}$ :	Valor de la propiedad $k$ en la posición $j$ del <i>buy-off</i> , línea previa a la línea de montaje final. Toma valor 1 si tiene dicha propiedad. Toma valor 0 en caso contrario.
L:	Número de líneas en paralelo del almacén.
C:	Capacidad de cada línea del almacén.
R:	Número de restricciones.
P:	Profundidad. Número de unidades que se utilizan para comparar.

M: Valor alto.

$\omega_k$ : Peso asociado a la violación de la restricción  $k$ , violando cuando coinciden.

$\omega'_k$ : Peso asociado a la violación de la restricción  $k$ , violando cuando no coinciden.

$M_k:N_k$ : Regla de Secuenciación. No más de  $M_k$  productos de  $N_k$  consecutivos con la opción  $k$ .

$m_k:n_k$ : Regla de Secuenciación. No más de  $m_k$  productos de  $n_k$  consecutivos con la opción  $k$ .

$mix_k$ : Proporción de productos cuya propiedad  $k$  es igual a 1. Se calcula respecto la totalidad de productos. Datos definidos en la entrada de productos.

M: Número muy grande

#### Variables

$Q_i$ : Cantidad de productos existentes en una línea. Se considera que existen desde que son asignados aunque no hayan llegado.

$W_i$ : Valor que cuantifica el peso de cada línea para asignar un producto a una línea.

$S_{jk}$ : Valor de la propiedad  $k$  en la posición  $j$  de la secuencia a evaluar.

$Z_i$ : Penalización total de una secuencia.

A: Línea Asignada

### 3.2. Algoritmo de entrada

El algoritmo de entrada se ejecuta cuando un producto se sitúa en la posición  $V_{o1}$  (Figura 1), decidiendo la línea en la cual se almacenará el producto.

Para situar los productos en el almacén se tiene en cuenta el espacio disponible en cada línea y la similitud entre productos.

Se obtiene un peso para cada línea según la regla de decisión que se considere. Finalmente se asignará al producto la línea con mayor peso.

$$A = i \quad \text{tal que } W_i = \max W_1, \dots, W_L \quad (2)$$

$$W_i = M \quad \text{si } Q_i = 0$$

Se considera que cuando una línea esté totalmente vacía tiene más prioridad para que se le asigne un producto. Es decir que si hay una línea vacía, el producto entrante se asignará a dicha línea sin tener en cuenta las reglas de decisión pertinentes.

A continuación se detallan cada una de las reglas de manera independiente, aunque podrían ser combinadas entre ellas.

#### Regla 1: Línea más vacía

El peso que se da a cada línea es inversamente proporcional a la cantidad de productos existentes en la misma.

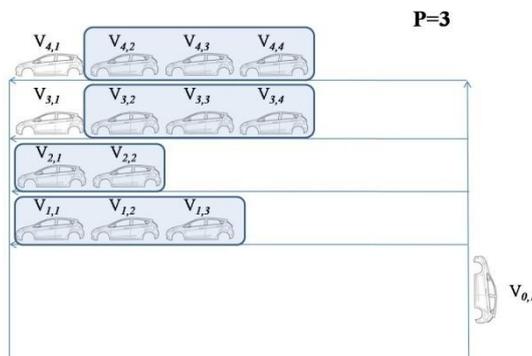
$$W_i = \frac{1}{Q_i} \quad (3)$$

#### Regla 2: Línea más similar

Para cada propiedad se compara el producto que va a entrar con el último de cada línea.

$$W_i = \frac{\sum_{k=1}^R V_{i Q_i k} \cdot V_{o1 k} + 1 - V_{i Q_i k} \cdot 1 - V_{o1 k}}{R} \tag{4}$$

Esta regla se podría aplicar en profundidad, es decir comparando no solamente con el último producto existente en la línea. Por ejemplo (Figura 3), si se considera que profundidad es 3, se compara con los tres últimos productos de cada línea. Si en alguna línea el número de productos es menor que el valor de profundidad tomado, se compara con todos los productos existentes en la línea.



**Figura 3.** Comparando el producto de entrada con cada una de las líneas tomando P=3.

Por tanto, si se compara en profundidad, el valor de  $W_i$  se obtendría de la siguiente manera:

Si  $Q_i \geq P$ :

$$W_i = \frac{\sum_{k=1}^R V_{o1 k} \cdot \sum_{j=Q_i-P}^{Q_i} \frac{V_{i j k}}{Q_i} + 1 - V_{o1 k} \cdot \sum_{j=Q_i-P}^{Q_i} \frac{1 - V_{i j k}}{Q_i}}{R \cdot P} \tag{5}$$

Si  $Q_i < P$ :

$$W_i = \frac{\sum_{k=1}^R V_{o1 k} \cdot \sum_{j=1}^{Q_i} \frac{V_{i j k}}{Q_i} + 1 - V_{o1 k} \cdot \sum_{j=1}^{Q_i} \frac{1 - V_{i j k}}{Q_i}}{R \cdot Q_i} \tag{6}$$

Regla 3: Línea más similar ponderando según la probabilidad de violar y/o la penalización

$$W_i = \frac{\sum_{k=1}^R P_k \cdot V_{i Q_i k} \cdot V_{o1 k} + P_k' \cdot 1 - V_{i Q_i k} \cdot 1 - V_{o1 k}}{R} \tag{7}$$

La ponderación se puede hacer teniendo en cuenta las siguientes variables y combinación de las mismas:

- Penalización asignada a la violación de una restricción. Cuanto más grande sea la penalización por la violación de una propiedad, más peso se asignará a la línea cuyo último producto tenga dicha propiedad igual.
- Mix de productos con dicha propiedad. Cuanta más cantidad de productos tengan una propiedad, más peso se asignará a la línea cuyo último producto tenga dicha propiedad igual. Se considera el mix total, de todos los productos que se van a procesar. Se podría considerar el mix de los productos existentes en el almacén tal y como hace Choi en (1997), o considerar el mix de los productos existentes en el sistema, tanto almacén como líneas previas.

- Espacio entre Productos según Restricciones. Siendo las Restricciones del tipo “No más de  $M_k$  productos de  $N_k$  consecutivos con la opción  $k$  y “No más de  $m_k$  productos de  $n_k$  consecutivos con la opción  $k$ .” se dará más peso cuanto mayor sea el cociente  $\frac{M}{N}$ .

De esta manera  $P_k$  y  $P'_k$  tomarían los valores indicados en la Tabla 2.

**Tabla 2.** Valores utilizados para la ponderación de la similitud entre productos.

$P_k$		$P'_k$	
Penalización asignada a la violación de una restricción	$\omega_k$	Penalización asignada a la violación de una restricción	$\omega'_k$
Mix de productos con dicha propiedad	$mix_k$	Mix de productos sin dicha propiedad	$1 - mix_k$
Espacio entre Productos según Restricciones	$\frac{M_k}{N_k}$	Espacio entre Productos según Restricciones	$\frac{m_k}{n_k}$

Al igual que en la regla anterior, se podría comparar en profundidad:

Si  $Q_i \geq P$ :

$$W_i = \frac{\prod_{k=1}^R P_k \cdot V_{o1k} \cdot \prod_{j=Q_i}^{Q_i-P} \frac{V_{ijk}}{P} + P_k' \cdot 1 - V_{o1k} \cdot \prod_{j=Q_i}^{Q_i-1} \frac{V_{ijk}}{P}}{R \cdot P} \quad (8)$$

Si  $Q_i < P$ :

$$W_i = \frac{\prod_{k=1}^R P_k \cdot V_{o1k} \cdot \prod_{j=Q_i}^{V_{ijk}} \frac{V_{ijk}}{Q_i} + P_k' \cdot 1 - V_{o1k} \cdot \prod_{j=Q_i}^1 \frac{1 - V_{ijk}}{Q_i}}{R \cdot Q_i} \quad (9)$$

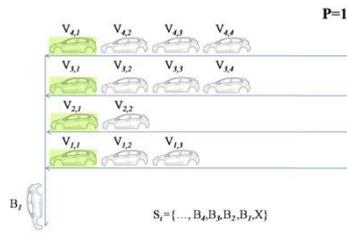
Las reglas anteriores pueden considerarse también en sentido contrario, es decir en lugar de considerar el más parecido o más vacío, considerar el menos parecido o más lleno.

Para cada una de ellas se consideraría que

$$W_i' = 1 - W_i \quad (10)$$

### 3.3. Algoritmo de salida

El algoritmo de salida se ejecuta cada *takt time*, y devuelve la línea de la cual se extraerá el siguiente producto. Para extraer los productos del almacén se tiene en cuenta la calidad de la secuencia obtenida (Z), analizándola en profundidad, es decir se analizan también las secuencias disponibles después de la extracción de un producto (S). Cuando la profundidad es igual a 1, se analizan solamente las secuencias obtenidas de la extracción de un producto. En el caso de un almacén con 4 líneas (Figura 4), el valor de Z para cada línea se obtiene evaluando la secuencia obtenida al añadir a la secuencia a evaluar (S) el primer producto de dicha línea.



**Figura 4.** Elementos disponibles para introducir en la secuencia a evaluar cuando P=1.

Para cada línea se evalúa la secuencia obtenida al extraer el primer producto de dicha línea. Para cada una de las restricciones, se aplica las ecuaciones (11) y (12).

$$\sum_{j=1}^{N_k-1} S_{j k} + 1 > M_k \wedge V_{i1k} = 1 \rightarrow Z_i = Z_i + \omega_k \tag{11}$$

$$\sum_{j=1}^{n_k-1} S_{j k} < m_k \wedge V_{i1k} = 0 \rightarrow Z_i = Z_i + \omega'_k \tag{12}$$

Si se considera  $P > 1$  la penalización de una secuencia consideraría también las penalizaciones de todas las secuencias previas. Por ejemplo, la penalización de la secuencia obtenida de extraer el segundo producto de la línea 1 ( $Z_{21}$ ), se sumaría a la penalización de la secuencia obtenida de extraer solamente el primer producto de dicha línea ( $Z_{11}$ ). Asimismo, se escoge la secuencia con penalización mínima. En la figura 5 se representan los elementos disponibles después de la extracción del primer producto de cada línea.



**Figura 5.** Elementos disponibles para introducir en la secuencia a evaluar cuando P=2.

#### 4. Conclusiones

En este trabajo se expone un modelo para la resecuenciación CSP mediante un almacén de líneas en paralelo, a falta de ser implementado. Este modelo aborda un problema poco estudiado y presenta un algoritmo de decisión a la salida del almacén, en el que se evalúa la secuencia en profundidad, es decir, analizando las secuencias disponibles. Esta evaluación de la secuencia en profundidad abordaría una de las deficiencias observables en cualquiera de los algoritmos presentados en la literatura, su carácter miope en la construcción de la secuencia. Esta miopía se fundamenta en el hecho de que, al considerar la elección de un determinado producto no se consideran las posibles repercusiones en posiciones posteriores.

Como líneas de investigación futuras, además de la implementación y experimentación con diferentes instancias, se podrían añadir otras reglas de decisión tanto a la entrada como a la salida, por ejemplo de manera que se tenga en cuenta la regularidad de una propiedad, o analizar la secuencia también para la decisión a la entrada.

## Agradecimientos

El presente trabajo se ha desarrollado gracias a la ayuda DPI2010-18243 del Ministerio de Ciencia e Innovación del Gobierno de España dentro del programa de Proyectos de Investigación Fundamental no orientada, con el título "Coordinación de operaciones en redes de suministro/demanda ajustadas, resilientes a la incertidumbre: modelos y algoritmos para la gestión de la incertidumbre y la complejidad".

Este trabajo también ha sido financiado parcialmente a partir del proyecto DPI2011-27633 y título "Programación de producción en cadenas de suministro sincronizadas multietapa con ensamblajes/desensamblajes con renovación constante de productos en un contexto de innovación"

## Referencias

- Bolat, A. (1997). Stochastic procedures for scheduling minimum job sets on mixed model assembly lines. *Journal of Operational Research Society*, Vol. 48, pp. 490-501.
- Boysen, N.; Fliedner, M. (2007). Comments on "Solving real car sequencing problems with ant colony optimization". *European Journal of Operational Research*, Vol. 182, n°. 1, pp. 466-468.
- Boysen, N.; Fliedner, M.; Scholl, A. (2009). Sequencing mixed-model assembly lines: Survey, classification and model critique. *European Journal of Operational Research*, Vol. 192, n°. 2, pp. 349-373.
- Boysen, N.; Scholl, A.; Wopperer, N. (2012). Resequencing of mixed-model assembly lines: Survey and research agenda. *European Journal of Operational Research*, Vol. 216, n°. 3, pp. 594-604.
- Campos, M.; Bonabeau, E.; Theraulaz, G.; Deneubourg, J. L. (2000). Dynamic scheduling and division of labor in social insects. *Adaptive Behavior*, Vol. 8, n°. 2, pp. 83-95.
- Celano, G.; Costa, A.; Fichera, S.; Perrone, G. (2004). Human factor policy testing in the sequencing of manual mixed model assembly lines. *Computers & Operations Research*, Vol. 31, n°. 1, pp. 39-59.
- Choi, W.; Shin, H. (1997). A real-time sequence control system for the level production of the automobile assembly line. *Computers & Industrial Engineering*, Vol. 33, n°. 3-4, pp. 769-772.
- Cicirello, V.; Smith, S. (2003). Distributed Coordination of Resources via Wasp-Like Agents Innovative Concepts for Agent-Based Systems, en W. Truszkowski, M. Hinchey, y C. Rouff (dir), pp. 71-80. Springer Berlin / Heidelberg.
- Cicirello, V. A.; Smith, S. F. (2004). Wasp-like Agents for Distributed Factory Coordination. *Autonomous Agents and Multi-Agent Systems*, Vol. 8, n°. 3, pp. 237-266.
- Ding, F. Y.; Sun, H. (2004). Sequence alteration and restoration related to sequenced parts delivery on an automobile mixed-model assembly line with multiple departments. *International Journal of Production Research*, Vol. 42, n°. 8, pp. 1525-1543.
- Fournier, X.; Agard, B. (2007). Improvement of earliness and lateness by postponement on an automotive production line. *International Journal of Flexible Manufacturing Systems*, Vol. 19, n°. 2, pp. 107-121.
- Gagne, C.; Gravel, M.; Price, W. L. (2006). Solving real car sequencing problems with ant colony optimization. *European Journal of Operational Research*, Vol. 174, n°. 3, pp. 1427-1448.
- Garcia-Sabater, J. P. (2001). The problem of JIT dynamic sequencing. A model and a parametric procedure, in 1st Operations Research Post-graduate Peripatetic Programme ORP3 2001, Paris, September 26-29.

- Han, Y. H.; Zhou, C. (2010). Dynamic sequencing of jobs on conveyor systems for minimizing changeovers. *The International Journal of Advanced Manufacturing Technology*, Vol. 49, n° 9, pp. 1251-1259.
- Inman, R. R.; Schmeling, D. M. (2003). Algorithm for agile assembling-to-order in the automotive industry. *International Journal of Production Research*, Vol. 41, n° 16, pp. 3831-3848.
- Jayaraman, A., Narayanaswamy, R., & Gunal, A. K. (1997). A Sortation System Model, in *Simulation Conference, 1997., Proceedings of the 1997 Winter*, pp. 866-871.
- Kittithreerapronchai, O. & Anderson, C. (2003). Do ants paint trucks better than chickens? Markets versus response thresholds for distributed dynamic scheduling, in *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*, pp. 1431-1439.
- Meissner, S. (2010). Controlling just-in-sequence flow-production. *Logistics Research*, Vol. 2, n° 1, pp. 45-53.
- Monden, Y. (1993). *Toyota Production System: An integrated approach*. Second Edition. Industrial Engineering and Management Press
- Moon, D. H.; Kim, H. S.; Song, C. (2005a). A Simulation Study for Implementing Color Rescheduling Storage in an Automotive Factory. *SIMULATION*, Vol. 81, n° 9, pp. 625-635.
- Moon, D. H.; Song, C.; Ha, J. H. (2005b). A Dynamic Algorithm for the Control of Automotive Painted Body Storage. *SIMULATION*, Vol. 81, n° 11, pp. 773-787.
- Nouyan, S. (2002). *Agent-Based Approach to Dynamic Task Allocation*
- Ant Algorithms*, en M. Dorigo, G. Di Caro, y M. Sampels (dir), pp. 43-99. Springer Berlin / Heidelberg.
- Parrello, B. D.; Kabat, W. C.; Wos, L. (1986). Job-shop scheduling using automated reasoning: A case study of the car-sequencing problem. *Journal of Automated Reasoning*, Vol. 2, n° 1, pp. 1-42.
- Scholl, A.; Klein, R.; Domschke, W. (1998). Pattern Based Vocabulary Building for Effectively Sequencing Mixed-Model Assembly Lines. *Journal of Heuristics*, Vol. 4, n° 4, pp. 359-381.
- Smith, B. M. (1996). Succeed-first or fail-first: A case study in variable and value ordering 96.26.
- Spieckermann, S.; Gutenschwager, K.; Voss, S. (2004). A sequential ordering problem in automotive paint shops. *International Journal of Production Research*, Vol. 42, n° 9, pp. 1865-1878.
- Valero-Herrero, M., Garcia-Sabater, J. P., Coronado-Hernandez, J. R., & Maheut, J. (2011a). Planteamiento dinámico del problema de secuenciación en líneas de montaje con mezcla de modelos, in *XV Congreso de Ingeniería de Organización // 5th International Conference on Industrial Engineering and Industrial Management*, pp. 288-296.
- Valero-Herrero, M., Garcia-Sabater, J. P., & Maheut, J. (2011b). An approach to the real circumstances of the car sequencing problem, in *41st International Conference on Computers and Industrial Engineering*.
- Xiaobo, Z.; Zhou, Z.; Asres, A. (1999). A note on Toyota's goal of sequencing mixed models on an assembly line. *Computers & Industrial Engineering*, Vol. 36, n° 1, pp. 57-65.