

## Heurísticas para el Visibility Windows Assembly Line Balancing Problem (VWALBP)

Gema Calleja<sup>1</sup>, Albert Corominas<sup>1</sup>, Alberto García-Villoria<sup>1</sup>, Rafael Pastor<sup>1</sup>

<sup>1</sup> Instituto de Organización y Control (IOC). Universitat Politècnica de Catalunya (UPC). Av. Diagonal 647, Planta 11, 08028 Barcelona [gema.calleja@upc.edu](mailto:gema.calleja@upc.edu), [albert.corominas@upc.edu](mailto:albert.corominas@upc.edu), [alberto.garcia-villoria@upc.edu](mailto:alberto.garcia-villoria@upc.edu), [rafael.pastor@upc.edu](mailto:rafael.pastor@upc.edu)

**Palabras clave:** ventanas de visibilidad, equilibrado de líneas

### 1. Introducción

En este trabajo se trata el problema conocido como problema de equilibrado de líneas de montaje con ventanas de visibilidad o Visibility Windows Assembly Line Balancing Problem (VWALBP), que ocurre en varios entornos de producción automatizados. En particular este problema surge, por ejemplo, en la producción de placas de circuito impreso (PCIs) en líneas *pick&place*. Este tipo de líneas consta de varias estaciones en paralelo que montan los componentes en posiciones predefinidas sobre la superficie de la placa. El montaje se realiza de modo cíclico (en cada ciclo se completa una pieza) y consiste en escoger (*pick*) un componente de un alimentador, trasladarlo hacia la placa, y colocarlo (*place*) en su posición correspondiente.

A diferencia de los problemas tradicionales de equilibrado de líneas, en los que se suele asumir que cada estación tiene acceso a toda una pieza entera, el VWALBP presenta la siguiente particularidad: la longitud de la pieza puede ser mayor que el ancho de la estación que la procesa y, en consecuencia, cada estación solamente puede acceder a la porción limitada de las piezas que está dentro de su *ventana de visibilidad*.

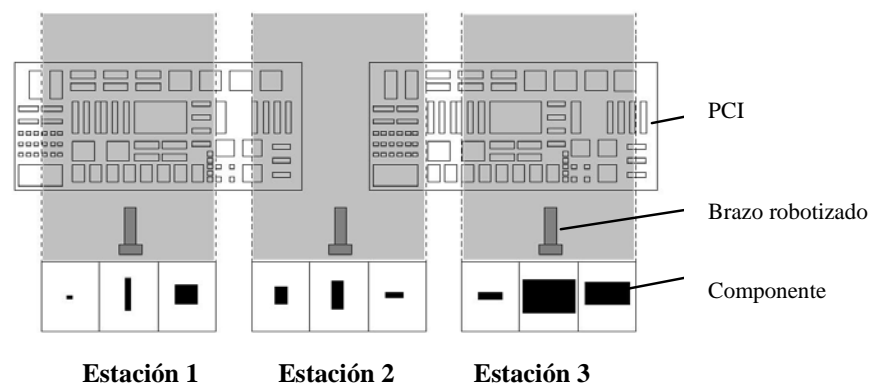
El VWALBP fue descrito por Müller-Hannemann y Weihe (2006) y formalizado con un modelo de programación lineal entera mixta (PLEM) por Corominas y Pastor (2010). Este modelo fue utilizado por Calleja et al. (2011) y se resolvieron ejemplares de un tamaño hasta cierto límite.

En la actualidad se están desarrollando nuevas líneas de investigación basadas en procedimientos heurísticos para resolver los ejemplares de mayor tamaño que quedan fuera del alcance del modelo de PLEM. En esta comunicación se presenta la estrategia heurística considerada en la investigación en curso, estructurada como sigue. En la sección 2 se describe el problema. En las secciones 3 y 4 se presentan la estrategia de resolución y los resultados computacionales, respectivamente. Por último la sección 5 contiene las conclusiones y las futuras líneas de investigación.

## 2. Descripción del problema

En esta sección se describen las características del VWALBP. El output esperado del problema y el objetivo a optimizar se especifican en los apartados 2.1 y 2.2, respectivamente.

La característica más importante del VWALBP es la existencia de ventanas de visibilidad en la línea, de manera que una tarea únicamente puede ser procesada si está dentro de la ventana de visibilidad de la estación en la que debe ser realizada. La longitud de las piezas es mayor que el ancho de las estaciones, lo que significa que una misma estación puede procesar partes de dos piezas consecutivas y una misma pieza puede ser procesada por varias estaciones. La Figura 1 muestra un ejemplo de una línea de montaje con tres estaciones. Nótese que las estaciones 1 y 3 solamente pueden procesar una parte de una pieza, mientras que la estación 2 puede procesar simultáneamente partes de dos piezas consecutivas.



**Figura 1.** Ejemplo de una línea con tres estaciones. El área gris corresponde a las ventanas de visibilidad.

A continuación se describe el proceso de montaje. La línea debe procesar un número de piezas iguales. Las piezas se colocan sobre la línea con una separación fija entre ellas y son transportadas a través de las estaciones mediante una cinta transportadora. El número de estaciones es conocido. Como particularidad de este problema respecto de otros problemas de equilibrado se asume que cada estación debe procesar un conjunto preasignado de tareas.

El proceso cíclico de montaje consta de una serie de *etapas estacionarias*, separadas entre sí por un *desplazamiento de avance*. En una etapa estacionaria la línea, con las piezas sobre ella, está inmóvil. Cada estación realiza sucesivas tareas de *pick&place*. Una vez se han completado todas las tareas de una etapa estacionaria específica, la línea comienza el desplazamiento de avance. La cinta hace avanzar la línea (y al mismo tiempo las piezas) en un desplazamiento a determinar. El mínimo desplazamiento en que se podría mover la línea es un valor  $\Delta$  llamado paso elemental que depende de la tecnología de la línea. Los desplazamientos de avance entre etapas (el número de pasos elementales que se desplaza la línea) no son necesariamente iguales. Mientras la línea está en movimiento no se permite realizar ninguna tarea sobre las piezas. Después, comienza la siguiente etapa estacionaria. Las etapas estacionarias y los desplazamientos de avance se repiten cíclicamente.

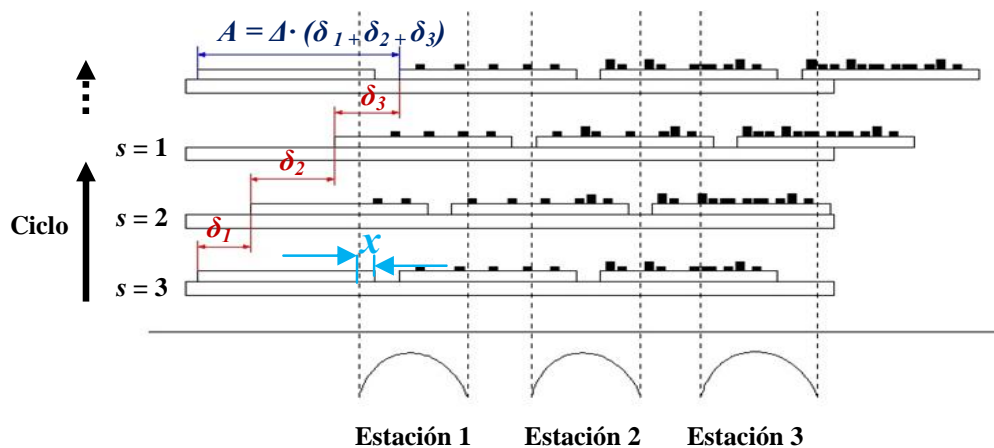
Entre dos etapas estacionarias consecutivas, existe un tiempo  $T$  necesario para acelerar/desacelerar la cinta.

Debido al comportamiento cíclico de la línea, la posición de las piezas en la línea en cada una de las etapas estacionarias queda determinada según un patrón denominado *esquema de avance*, constituido por:

- La posición de referencia  $x$ : es la distancia del borde derecho de la pieza respecto al límite izquierdo de la primera ventana de visibilidad en la primera fase estacionaria.
- El número  $S$  de etapas estacionarias (que es igual al número de desplazamientos del esquema de avance).
- Los desplazamientos de avance  $\delta_s$  ( $s = 1, \dots, S$ ).

La Figura 2 muestra un esquema de avance con tres etapas estacionarias ( $S = 3$ ). Después del último desplazamiento de avance, las piezas han sido desplazadas exactamente en la distancia  $A$ , que corresponde a la distancia entre los bordes derechos de dos piezas consecutivas y coincide con la suma de los desplazamientos de avance:

$$A = \Delta \cdot \sum_{s=1}^S \delta_s \quad (1)$$



**Figura 2.** Esquema de avance cíclico de una línea con tres etapas estacionarias.

## 2.1. Output

El output deseado consiste en:

- i) un esquema de avance y
- ii) la asignación de cada tarea a una de las etapas estacionarias.

Recuérdese que las tareas están a priori asignadas a las estaciones (la asignación de tareas a estaciones es parte del input).

## 2.2. Objetivo

El objetivo del problema consiste en minimizar el tiempo de ciclo expresado en la ecuación (2), el cual consiste en la suma de:

- i) el número  $S$  de etapas estacionarias multiplicado por el tiempo  $T$
- ii) las duraciones de cada etapa estacionaria  $s$ ,  $C_s$ , que constituyen un ciclo.

$$[MIN]z = S \cdot T + \sum_{s=1}^S C_s \quad (2)$$

## 3. Estrategia de resolución

La estrategia de resolución se basa en descomponer el VWALBP en dos subproblemas: cálculo de un esquema de avance factible y asignación de tareas a etapas estacionarias, los cuales se describen en los apartados 3.1 y 3.2, respectivamente. La solución del problema global se obtiene a partir de la unión de las soluciones obtenidas en los dos subproblemas.

### 3.1. Cálculo de un esquema de avance factible

El esquema de avance se calcula mediante un algoritmo diseñado a medida. Se ha desarrollado un heurístico que, dada una posición de referencia  $x$  inicial, genera un esquema de avance factible con el menor número posible de etapas estacionarias  $S$ . La motivación de obtener un esquema con el número mínimo de etapas de avance es reducir al máximo los tiempos de aceleración/desaceleración. De esta manera consideramos que se obtendrán soluciones buenas.

A continuación se describe el algoritmo desarrollado. El procedimiento debe determinar el valor de los desplazamientos del esquema de avance de la línea de forma que cada tarea sea visible en su estación correspondiente al menos en una de las etapas estacionarias. Para cada estación, y para cada tarea asignada a dicha estación, se calcula la máxima distancia que se podría desplazar la tarea para poder ser realizada, que corresponde a la distancia entre la posición de la tarea en la línea hasta la posición del límite derecho de su estación. De esta manera se evita que la tarea pueda desplazarse más allá de la ventana de visibilidad de su estación correspondiente. El mínimo valor de entre los máximos desplazamientos posibles en para todas las estaciones de la línea determina el valor del primer desplazamiento del esquema de avance ( $\delta_1$ ). Los desplazamientos siguientes ( $\delta_2, \dots, \delta_S$ ) se calculan siguiendo el mismo razonamiento, y el cálculo finaliza cuando el desplazamiento acumulado de los desplazamientos es igual a  $A/\Delta$ .

En la Figura 3 se muestra el pseudocódigo utilizado para la generación del esquema de avance. El funcionamiento básico del algoritmo es el siguiente. En cada iteración se calcula el valor del desplazamiento  $\delta_s$  (correspondiente al número de pasos elementales que se desplaza la línea) hasta que el desplazamiento acumulado sea igual a  $A/\Delta$ . Para calcular  $\delta_s$  se determina el mínimo de los máximos desplazamientos posibles para cada estación mediante el método *NumMaxPasosElem*, que devuelve el número máximo de pasos elementales que puede recorrer la tarea  $j$  para que sea visible en la máquina  $i$ . Para ello se parte de una posición de referencia inicial para la primera pieza en la línea ( $x$ ). Se elige como valor de  $x$  la máxima

distancia posible del borde derecho de la pieza respecto al límite izquierdo de la primera estación,  $x = R_1 + a_1^{min}$ , donde  $R_1$  es la posición del límite derecho de la primera estación y  $a_1^{min}$  es la posición de la tarea más cercana al borde derecho de la pieza. Dado el número total de pasos elementales desplazados en el ciclo actual  $S^T$ , la estación  $i$ , el conjunto  $J_i$  de tareas asignadas a dicha estación, las coordenadas de su límite izquierdo  $L_i$  y derecho  $R_i$ , una tarea  $j$  asignada a la estación  $i$  y su posición  $a_j$  respecto al borde derecho de la pieza, se calcula el valor  $aux$  correspondiente a la distancia desde la posición de la tarea en la línea hasta el lado izquierdo de su estación correspondiente. El valor de  $aux$  corresponde a la tarea más cercana a la izquierda de su estación y por tanto determina el máximo desplazamiento que puede moverse la línea de forma que dicha tarea quede dentro de la ventana de visibilidad.

Una vez calculado el valor de  $\delta_s$  para todas las tareas y para todas las estaciones, se actualiza el valor del desplazamiento acumulado  $S^T$ , y se repite iterativamente el cálculo para obtener el valor de los siguientes desplazamientos del esquema de avance. Finalmente, el procedimiento finaliza cuando se cumple la condición  $S^T = A/\Delta$  y como resultado se obtiene un esquema de avance factible con  $S$  desplazamientos  $(\delta_1, \delta_2, \dots, \delta_S)$ .

```

s := 1; ST := 0

fin := (ST = A/Δ)
mientras (!fin) hacer
    δs := min ( A/Δ - ST, min∀i min∀j ∈ Ji ( NumMaxPasosElem(ST, i, j) ) )
    ST := ST + δs
    fin := (ST = A/Δ)
    si (!fin) entonces s := s + 1 fsi
fmientras

NumMaxPasosElem(ST, i, j) {
    aux = [ Li - (x + ST · Δ - aj) ] % A
    si aux = 0 entonces aux := A fsi
    d := aux + (Ri - Li)
    devolver δs := ⌊ d/Δ ⌋
}

```

**Figura 3.** Pseudocódigo del cálculo del esquema de avance factible

### 3.2. Asignación de tareas a etapas estacionarias

Una vez generado un esquema de avance factible, el siguiente paso consiste en asignar las tareas a una de las etapas estacionarias en que es visible. Para ello se ha propuesto un modelo de PLEM que calcula, utilizando como input el esquema de avance generado en el punto anterior, una asignación óptima de cada tarea a una de las etapas estacionarias. A continuación se muestra el modelo utilizado:

#### Datos:

- $m$  número de estaciones
- $N$  número de tareas
- $J_i$  conjunto de tareas de la estación  $i$   $J_i \cap J_k \neq \emptyset \quad \forall i, k \ (i=1, \dots, m, k=1, \dots, m, i \neq k)$
- $p_j$  tiempo de proceso de la tarea  $j \ (j = 1, \dots, N)$
- $S$  número de desplazamientos en un ciclo
- $\Pi_j$  conjunto de etapas estacionarias en las que la tarea  $j$  es visible dentro de la ventana de visibilidad de la estación en la que debe realizarse  $(j = 1, \dots, N)$

#### Variables:

- $y_{js} \in \{0,1\}$ ,  $y_{js} = 1$  sii la tarea  $j$  se realiza en la etapa estacionaria siguiente al desplazamiento  $s - 1 \ (j = 1, \dots, N; s \in \Pi_j)$
- $C_s$  duración, para toda la línea, correspondiente a la etapa estacionaria siguiente al desplazamiento  $s - 1 \ (s = 1, \dots, S)$

#### Modelo:

$$[MIN] \ z = \sum_{s=1}^S C_s \quad (3)$$

$$\sum_{s=1}^S y_{js} = 1 \quad j = 1, \dots, N \quad (4)$$

$$\sum_{j \in J_i} p_j \cdot y_{js} \leq C_s \quad i = 1, \dots, m; \ s = 1, \dots, S \quad (5)$$

A partir del esquema de avance, se conoce el conjunto de etapas estacionarias en las que cada tarea es visible dentro de su estación correspondiente,  $\Pi_j$ . El objetivo (3) es minimizar la duración de las etapas estacionarias. Las restricciones (4) imponen que cada tarea es asignada a una, y solamente una, de las etapas estacionarias del ciclo, y (5) aseguran que la duración correspondiente a las etapas estacionarias no es menor que el tiempo de proceso de cualquiera de las estaciones.

## 4. Experiencia computacional

Se generó un juego de datos a partir de diferentes rangos de valores para la separación entre piezas ( $A = \{11-16, 17-21, 22-26, 27-31, 32-36, 37-41\}$ ), el número máquinas ( $m = \{5-10, 11-20, 21-30, 31-40\}$ ), y el número de tareas ( $N = \{50-200, 201-400, 401-600, 601-800, 801-1000\}$ ). La longitud de la pieza se puede deducir del parámetro  $A$ , ya que se corresponde con

el valor de  $A$  menos el valor del espacio entre dos piezas consecutivas en la línea (en nuestro caso dicho espacio es de 1 unidad de longitud  $-ul-$ , luego la longitud de la pieza es  $A - 1$ ). Para cada combinación de estos rangos se generaron 10 ejemplares, obteniendo así un juego total de  $6 \times 4 \times 5 \times 10 = 1.200$  ejemplares. El ancho de las estaciones se fijó en 10  $ul$ , y el tiempo  $T$  de aceleración/desaceleración se fijó en 200 unidades de tiempo ( $ut$ ). La posición de las tareas sobre la pieza se obtuvo de forma aleatoria a lo largo de la longitud de la misma. El tiempo de proceso de las tareas se obtuvo aleatoriamente entre 100 y 150  $ut$ . Por último, las tareas se asignaron a las estaciones de forma equiprobable, teniendo en cuenta que cada estación debe tener al menos una tarea asignada.

Se intentó resolver este juego de datos con el modelo de PLEM mediante el solver IBM ILOG CPLEX12.2, limitando el tiempo computacional a una hora. Se observa que la longitud de la pieza y el número de tareas son los parámetros que más influyen en la complejidad de resolución del problema: cuanto mayor es la longitud de la pieza respecto al ancho de las estaciones y mayor es el número de tareas, más difícil resulta hallar una solución óptima. El número de estaciones no parece tener una influencia significativa. Así, para piezas de hasta 15  $ul$  el modelo es capaz de resolver ejemplares de hasta 1.000 tareas, mientras que para piezas de hasta 20 y 25 el límite es de hasta 800 y 600 tareas, respectivamente. Finalmente, para piezas de dimensiones mayores, el modelo alcanzó a resolver hasta 200 tareas.

Para obtener una estimación de la calidad de las soluciones producidas por este heurístico, comparamos las soluciones obtenidas con aquellos ejemplares de los que conocemos la solución óptima. Para el 69,9% de estos ejemplares, el heurístico devolvió una solución óptima. Para el resto, la diferencia respecto al valor óptimo de la función objetivo es en promedio de 7,4%. Además, este heurístico es extremadamente rápido: el tiempo de ejecución para la generación del esquema de avance es despreciable, y para la asignación de tareas a etapas estacionarias es en promedio del orden de centésimas de segundo. Se observó también que el número de etapas de avance de las soluciones devueltas por el heurístico coincide con el número de etapas de las soluciones óptimas excepto en un caso. En dicho caso la solución óptima tiene una etapa menos que en el heurístico, ya que la posición de referencia  $x$  en el modelo es parte del output a determinar. Así, parece razonable presuponer que los esquemas de avance con un número pequeño de etapas de avance forman parte de las buenas soluciones.

## **5. Conclusiones y futuras líneas de investigación**

En este trabajo se presenta el problema de equilibrado de líneas de montaje de visibilidad (Visibility Windows Assembly Line Balancing Problem, VWALBP), en el que, a diferencia de los problemas tradicionales de equilibrado de líneas, cada estación solamente tiene acceso a una porción limitada de las piezas dentro del área correspondiente a su ventana de visibilidad. El objetivo consiste en obtener el esquema de avance de la línea y la asignación de tareas a etapas estacionarias de forma que el tiempo de ciclo sea mínimo.

La resolución del VWALBP se aborda mediante la utilización de un método heurístico para ejemplares de grandes dimensiones que quedan fuera del alcance del modelo de PLEM. El heurístico descompone el problema en dos partes. En primer lugar se genera un esquema de avance factible mediante un algoritmo. Después, en la segunda parte se utiliza dicho esquema de avance como input de un modelo exacto que calcula una asignación óptima de las tareas a una de las etapas estacionarias del ciclo.

La eficiencia con que se resuelve el modelo de asignación permite pensar en desarrollar otras heurísticas basadas en reducir el espacio de búsqueda al esquema de avance. Por ejemplo, generar un vecindario a partir de un esquema de avance inicial y a continuación utilizar el modelo de asignación para obtener la asignación óptima de tareas correspondiente a uno de los esquemas de avance generados. De este modo se obtiene un vecindario de soluciones al que se puede aplicar búsqueda local para tratar de mejorar la solución inicial.

Otra propuesta es considerar como punto de partida un esquema de avance inicial con el mayor número de etapas estacionarias (es decir, el caso en el que la línea se desplaza entre etapas el valor del paso elemental  $\Delta$ ). La heurística consistiría en generar iterativamente nuevos esquemas de avance agrupando, si es posible, las dos etapas estacionarias consecutivas que mayor ahorro proporcionen en el tiempo de ciclo al ser agrupadas. A partir de este heurístico se podría desarrollar un algoritmo tipo GRASP (Greedy Randomized Adaptive Search Procedure) donde la selección de la pareja de etapas a agruparse se aleatoriza.

Otra posible línea de investigación es el desarrollo de mateheurísticas que, a partir de una solución heurística inicial, utilicen el modelo matemático para generar mejores soluciones, dentro de un vecindario definido por restricciones adicionales en el modelo. Un ejemplo de posible aplicación es el Corridor Method (CM), que permitiría reducir el espacio de búsqueda a porciones restringidas del espacio de soluciones del problema.

## Referencias

Calleja, G.; Corominas, A.; García-Villoria, A.; Pastor, R.; (2011). Exact and Heuristic Approaches for the Visibility Windows Assembly Line Balancing Problem (VWALBP). Actas del 12º Congrès Annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF), Volumen II, página 583.

Corominas, A.; Pastor, R. (2010). A MILP model for the Visibility Windows Assembly Line Balancing Problem (VWALBP): the case of the Müller-Hannemann & Weihe problem. Technical report.

Müller-Hannemann, M.; Weihe, K.; (2006). Moving policies in cyclic assembly line scheduling. Theoretical Computer Science, No. 351, pp. 425-436.