Visión del Hoist Scheduling Problem como un problema clásico de Programación Dinámica.*

Manuel Mateo, Ramón Companys, Joaquín Bautista
ETSEIB, Dep. Organización Empresas, Universitat Politècnica Catalunya, mmateo@oe.upc.es
ETSEIB, Dep. Organización Empresas, Universitat Politècnica Catalunya, companys@oe.upc.es
ETSEIB, Dep. Organización Empresas, Universitat Politècnica Catalunya, bautista@oe.upc.es

RESUMEN

En la programación de la fabricación de objetos homogéneos para el problema considerado, Hoist Scheduling Problem (HSP), debe determinarse un ciclo de movimientos. A cada posible secuencia cíclica puede asociarse un grafo, cuyas propiedades permiten que su resolución sea equivalente a la utilización de la programación dinámica. ***Palabras Clave****: Hoist Scheduling Problem, grafos, programación dinámica.*

1 Introducción.

En un sistema productivo de las características del problema HSP, si todos los objetos tratados son idénticos, el objetivo es determinar la programación cíclica de un puente-grúa tal que minimice el tiempo de ciclo. Phillips y Unger [1] fueron los primeros investigadores en plantear un modelo para el caso básico con un transportador o puente-grúa, con tanques químicos mono-baño y mono-función, basado en programación lineal entera.

Autores posteriores, como Shapiro y Nuttle [2], rehicieron el modelo mediante otras técnicas, como la exploración arborescente. En este caso, se requiere evaluar todas aquellas subsecuencias y secuencias de movimientos posibles derivadas de un vértice, mientras no se demuestre la infactibilidad de éste. Chen, Chu y Proth [3] proponen explotar la teoría de grafos para resolver cada una de las secuencias de movimientos.

La programación de la grúa, mediante resolución de un grafo **G** con Tiempo de Ciclo Acotado [4], corresponde al problema de determinar un camino máximo en dicho grafo **G**, cuyo vértice final sea un vértice *i*. La transformación aplicada en el cálculo se corresponde con la forma progresiva de la ecuación de recurrencia utilizada en la programación dinámica determinista y discreta, en el espacio de los estados.

2 El problema HSP (*Hoist Scheduling Problem*).

Un conjunto de *n* piezas (j=1,...,n) deben recibir un tratamiento (operación) en *m* estaciones (i=1,...,m), en un sistema productivo formado por un conjunto de tanques llenos de productos químicos. La permanencia de una pieza en cada estación debe durar un tiempo comprendido entre dos valores, uno mínimo y otro máximo.

Uno o más transportadores (o grúas) realizan los movimientos de las piezas entre estaciones, en un tiempo no despreciable. Las piezas se recogen en la estación de carga, E(0), punto de entrada de las piezas al sistema. El punto de salida de las piezas del sistema, donde se depositan una vez han pasado por todas las operaciones, es la estación de descarga, E(m+1).

* El presente estudio se ha realizado en el marco del proyecto de investigación TAP98-0494 financiado por la CICYT.

Las condiciones generales del problema pueden enunciarse como:

- Las duraciones de las operaciones no son fijas, sino acotadas.
- Los tiempos de transportes de grúa no son negligibles.
- No existen colas ni stocks intermedios entre tanques.

El número de variantes en el problema HSP depende básicamente de tres factores:

- Los tipos de tanques: según el número de objetos en cada tanque (mono-baño o multi-baño) y según el número de tratamientos distintos en cada baño (mono-función o multi-función).
- Los tipos de productos: según sean piezas u objetos idénticos o distintos (homogéneos o heterogéneos).
- Los elementos de transporte: según el número de grúas (mono-grúa o multi-grúa).

3 Determinación de la secuencia cíclica, para objetos homogéneo s.

Para el caso de tanques mono-baño, mono-función, productos homogéneos y mono-grúa, el objetivo es determinar una secuencia cíclica, tal que minimice el tiempo de ciclo, del conjunto de movimientos de una única grúa sobre objetos homogéneos, que pasan por *m* etapas, cada una realizada en un tanque con su respectiva restricción de ventana.

Las hipótesis en que se basa el modelo a plantear son las siguientes:

- El tiempo de proceso en cada estación debe estar entre el intervalo dado.
- El transporte de los objetos lo realiza una única grúa.
- La operación de transporte no se puede interrumpir.
- La etapa E(0) tiene los objetos preparados cuando llega la grúa.
- El tanque de la etapa E(i) debe estar vacío cuando llega un objeto.
- Todos los tanques están ordenados según las etapas del proceso.
- Todos los objetos son idénticos y siguen la misma ruta entre E(0) y E(m+1).

Una secuencia cíclica (**H**,**T**) se define a partir de dos vectores:

- Secuencia de movimientos de grúa: **H** = $<h_{[0]}, h_{[1]}, ..., h_{[k]}, ..., h_{[m]}>$, donde $h_{[k]}=h_i$ es el movimiento de un objeto de E(i) a E(i+1), $0 \leq i \leq m$.
- Tiempos de inicio de los movimientos: **T** = $<t_{[0]}, t_{[1]}, ..., t_{[k]}, ..., t_{[m]}>$, donde $t_{[k]}=t_i$ es el tiempo de inicio del movimiento h_i, $0 \leq i \leq m$.

Tomando como referencia el movimiento sobre la estación de carga ($h_{[0]}=0 \quad {}_{[0]}=0$), se plantea el problema de minimizar el tiempo de ciclo TC(**H**) asociado a una secuencia **H** propuesta.

Los datos de partida son:

- las restricciones de ventana de cada tanque: a_i, b_i ($i = 1,2, ...,m$);

- los tiempos de movimiento de la grúa: bien sea con carga, f_i (i = 0,1, ..., m); bien sea sin carga, $e_{i,j}$ (i,j = 0,1, ..., m+1);
- la secuencia de movimientos propuesta: $\mathbf{H} = <h_{[0]}=0, h_{[1]}, ..., h_{[m]}>$.

Para resolver el modelo, se requieren *m+1* variables: el tiempo de inicio de los movimientos, en el vector $\mathbf{T} = <0, t_{[1]}, ..., t_{[m]}>$, y el tiempo de ciclo TC(**H**). El modelo contiene *3m+1* restricciones: *2m* restricciones se deben a los límites de tiempo de proceso del objeto en cada uno de los *m* **tanques**, y *m+1* restricciones responden al tiempo de viaje de la **grúa** en cada movimiento.

Todas las restricciones pueden expresarse como diferencia de dos variables del vector **T**. Dicho valor debe ser mayor que el valor del término independiente, el cual puede depender de la variable del tiempo de ciclo (que llamaremos para simplificar TC).

Las restricciones de operaciones para un objeto depositado en un tanque *i* en un ciclo anterior al que se recoge tienen un término independiente dependiente de TC ($a_i+f_{i-1}-TC$; $-b_i-f_{i-1}+TC$); mientras que en los demás casos esto no es así (a_i+f_{i-1}; $-b_i-f_{i-1}$). De las restricciones debidas al tiempo de transporte de grúa, sólo aquélla referida al último movimiento, movimiento m-ésimo, de un ciclo depende de TC ($f_{[m]}+e_{[m]+1,0}-TC$); en las demás, es una suma de tiempos de viaje ($f_{[k]}+e_{[k]+1,[k+1]}$) para k=0,...,m-1.

4 Programación dinámica aplicada en la resolución de un grafo.

4.1 Grafo G y subgrafo G_l asociado al problema de minimización de TC.

Sea el problema de optimización, definido por la función objetivo (1) sometida a *3m+1* restricciones del tipo de la ecuación (2), con $t_i \geq 0$ (i=0, ..., m) y b_j que depende del valor asignado a la variable de tiempo de ciclo TC.

$$[\text{MIN}]\ TC \tag{1}$$

$$t_{Gto(j)} - t_{Gfrom(j)} \geq b_j(TC) \quad j=1, ..., 3m+1 \tag{2}$$

Las *3m+1* restricciones del problema sugieren la búsqueda del camino máximo en un grafo **G** con las siguientes características:

- A cada variable del problema t_i (i=0, ..., m), excepto la variable TC, se asocia un vértice v(i).
- Entre los vértices $v(t_{Gfrom})$ y $v(t_{Gto})$, se asocia un arco con valor $b_j(TC)$. En la búsqueda de caminos máximos, éste conduce a aplicar la restricción original.

En función del valor asignado a TC, el grafo **G** puede tener caminos extremos, o sea máximos, acotados o no tenerlos.

Definición 1: Un grafo **G** es coherente con TC=C, Coh[TC=C], si para toda pareja de vértices [v(i), v(j)] existe un camino máximo. Para un grafo orientado, como **G**, la coherencia supone que todos los circuitos de dicho grafo tengan valor no positivo.

Proposición 1: En el grafo **G** existen $3^{m-1}\cdot2^2$ subgrafos $\mathbf{G_l}$, formados por un conjunto de arcos que incidan en cada uno de los *m+1* vértices del grafo.

Demostración: Si se detallan las 3m+1 restricciones, los vértices v(0) y v(m) de un grafo **G** tienen dos arcos incidentes (uno de restricciones de tanque y otro de grúa), mientras que los demás vértices, [v(1), ..., v(m-1)], del grafo **G** tienen tres arcos incidentes (dos arcos de restricciones de tanque y uno de grúa).

Sea un vector **X** formado por componentes asociados a los vértices del grafo **G**.

Proposición 2: A todo grafo **G** puede asociarse una transformación Φ_I de un vector **X** en otro vector **Y** (de idénticas características a **X**) tales que $\mathbf{X,Y}\in\Re^{m+1}$.

Demostración: La transformación Φ_I consiste en aplicar la expresión (3):

$$y_i = \text{MAX}_{k(i)} \{ b_{j,i}^{k(i)} + x_j^{k(i)} \} \qquad i = 0, ..., m \tag{3}$$

con $b_{j,i}^{k(i)}$(TC) entero que puede depender de TC; $i,j\in\{0,1, ..., m\}$; $k(i)\in\{1,2\}$ si i=0,m; $k(i)\in\{1,2,3\}$ si i=1,2,...,m-1, que coincide con la determinación de caminos máximos en un grafo.

En nuestro caso, el grafo **G** se llama Grafo con Tiempo de Ciclo Acotado, al poder hallarse cotas sobre el valor de TC [4], que facilitan la convergencia de dicho procedimiento en él.

Proposición 3: La transformación Φ_I divide el espacio $\Re^{m+1}$ en zonas según las restricciones activas en cada vértice *i*. El número máximo de zonas es $3^{m-1}\cdot2^2$, en este caso. A cada zona *l* corresponde una transformación reducida $\Phi_{I,l}$, a la cual puede asociarse también un subgrafo $\mathbf{G_l}$. A cada vértice de dicho subgrafo incide un sólo arco.

Demostración: Según la Proposición 2, existe un grafo **G** asociado a la transformación Φ_I. Para cada vértice *i* del grafo, existirá un valor *k(i)* tal que $y_i = b_{j,i}^{k(i)} + x_j^{k(i)}$. Dicha expresión refleja la restricción activa en el vértice *i*.

Además, según la demostración de la Proposición 1, existirá siempre una restricción activa para cada vértice, por lo menos. En la frontera entre dos zonas, existen dos restricciones activas en un cierto vértice *i*.

La transformación reducida $\Phi_{I,l}$ incluye una selección de *m+1* valores k(0), k(1), ..., k(m). De manera equivalente, en el subgrafo $\mathbf{G_l}$, existen *m+1* arcos que inciden en los vértices 0,1,...,m, respectivamente.

Proposición 4: En un subgrafo $\mathbf{G_l}$ existe como mínimo un circuito, aunque alguno de los arcos del subgrafo puede no formar parte de un circuito.

Demostración: Sea un vértice *j* cualquiera del subgrafo $\mathbf{G_l}$. Según la Proposición 3, sólo incide un arco **e[v(i),v(j)]** en dicho vértice, siendo el origen de dicho arco el vértice *i*. Tómese ahora el vértice *i* como vértice destino.

Dado que el número de vértices del subgrafo $\mathbf{G_l}$ es finito, después de realizar la anterior operación durante *k* iteraciones, el vértice origen en la iteración k+1-ésima coincidirá con alguno de los *k* vértices origen ya visitados.

Sea el conjunto de vértices origen, en orden inverso al visitado, finalizando cuando se visitó por primera vez al vértice repetido. Entonces, se dispone de un circuito recorrido en sentido contrario a la orientación de sus arcos.

Si el vértice repetido es el primero, el circuito pasa por todos los vértices. Si no lo es, entonces queda una parte de arcos del grafo fuera del circuito. Si quedaran vértices por visitar, esto implica que o bien se forma otro circuito o bien se visitará nuevamente algún vértice del circuito ya formado.

4.2 Los puntos dobles en una transformación $\Phi_{I,l}$.

Definición 2: La transformación $\Phi_{I,l}$, asociada al subgrafo $\mathbf{G_l}$, tiene un punto doble $\mathbf{X^o}$ por lo menos, si se cumple (4):

$$\mathbf{X^o}=\Phi_{I,1}(\mathbf{X^o}) \tag{4}$$

En nuestro caso, los puntos dobles forman parte de la solución buscada, ya que indican que tras recorrer los *m+1* vértices del subgrafo $\mathbf{G_l}$ los valores se repetirán cíclicamente tras un período de tiempo TC. No obstante, la existencia de puntos dobles depende del valor asignado a la variable TC.

Definición 3: Si $\mathbf{X}\in\Re^{m+1}$ se encuentra en el interior de una zona *l*, esta transformación reducida $\Phi_{I,1}$ es equivalente a una transformación lineal del tipo (5):

$$\mathbf{Y}=\mathbf{B}+\mathbf{A}\cdot\mathbf{X} \tag{5}$$

donde cada fila corresponde a la restricción activa incidente para cada vértice. La matriz **A** es una matriz estocástica de dimensión *m+1*.

Teorema 1: Si existe un punto doble $\mathbf{X^o}$ de $\Phi_{I,l}$, todos los circuitos en el grafo $\mathbf{G_l}$ tienen valor nulo.

Demostración: Sea el un circuito en el grafo $\mathbf{G_l}$, formado por *a* vértices. Aplicando la transformación $\Phi_{I,l}$, sobre el primer vértice:

$$x_1 = b_1 + x_a \tag{6}$$

Repitiéndolo para cada vértice del circuito:

$$x_i = b_i + x_{i-1} \qquad i=2,...,a \tag{7}$$

Sumando la ecuación (6) a las *a-1* ecuaciones de la forma (7), se obtiene:

$$0=\sum_{i=1}^{a} b_i \tag{8}$$

Corolario: Si todos los circuitos del grafo $\mathbf{G_I}$ tienen valor nulo, existen puntos dobles de $\Phi_{I,I}$.

4.3 La programación dinámica en la resolución de un grafo y sus consecuencias.

Sea $\mathbf{X_0}$, vector formado por los componentes iniciales asignados al vector $\mathbf{X}$, así como que $\mathbf{Y_N}=\mathbf{B}+\mathbf{A}\cdot\mathbf{X_N}$ y $\mathbf{X_{N+1}}=\mathbf{Y_N}$ (N=0,1,2, ...), con una o diversas componentes convexas. Si $\Phi_{I,I}$ tiene punto doble $\mathbf{X^o}$, según la Definición 2, $\mathbf{X^o}=\Phi_{I,I}(\mathbf{X^o})$.

Entonces, conviene estudiar lo que sucede si $\mathbf{X^1}\neq\mathbf{X^0}+k\cdot\mathbf{1}$ $\forall$k, suponiendo el caso de un solo circuito, $\mathbf{Y^1}=\Phi_{I,I}(\mathbf{X^1})$ con $\mathbf{Y^1}\neq\mathbf{X^1}$.

Proposición 5: Si $\mathbf{X^1}$ no es punto doble de $\Phi_{I,I}$, entonces para *p* suficientemente grande:

$$\Phi_{I,I}{}^{p+q}(\mathbf{X^1}) = \Phi_{I,I}{}^{p}(\mathbf{X^1}) \tag{9}$$

donde *q* es múltiplo del número de vértices del circuito, y en caso de varios circuitos, mínimo común múltiplo de los respectivos números de vértices.

Proposición 6: En las condiciones anteriores, con $\mathbf{X^1}$ definido como en la Proposición 5:

$$\mathbf{X^2} = \frac{1}{q}\sum_{j=1}^{q}\phi_{I,I}{}^{p+j}(\mathbf{X^1}) \tag{10}$$

es tal que $\mathbf{X^2}=\Phi_{I,I}(\mathbf{X^2})$.

Demostración: Dada una matriz $\mathbf{A}$. El vector $\mathbf{X_N}$, después de aplicar la transformación *N* iteraciones, se obtiene como:

$$\mathbf{X_N} = \mathbf{B} + \mathbf{A}\cdot\mathbf{B} + \mathbf{A^2}\cdot\mathbf{B} + ... + \mathbf{A^{N-1}}\cdot\mathbf{B} + \mathbf{A^N}\,\mathbf{X_0} \tag{11}$$

$$\mathbf{X_N} = (\mathbf{I} + \mathbf{A} + ... + \mathbf{A^{N-1}})\,\mathbf{B} + \mathbf{A^N}\,\mathbf{X_0} \tag{12}$$

Este procedimiento iterativo permite, gracias a las propiedades de la matriz estocástica $\mathbf{A}$ y del Teorema 1, que se demuestre [5] que la proposición se cumple en caso que la secuencia $\mathbf{H}$ sea factible.

En este caso, el valor *p* es suficientemente grande cuando iguala al grado de multiplicidad de los valores propios 0 de la matriz $\mathbf{A}$.

En definitiva, los principios de la programación dinámica [6] demuestran la expresión (13):

$$\mathbf{X_N} \to N\,\mathbf{g} + \mathbf{O_N} + \mathbf{W} \tag{13}$$

donde $\mathbf{g} = \mathbf{A^*}\,\mathbf{B}$, y $\mathbf{O_N}$ es un posible término oscilante en *q* etapas tal que $\mathbf{O_{N+q}}=\mathbf{O_N}$, cuya media puede hacerse 0 para centrarlo en el punto doble $\mathbf{W}$.

4.4 Estados y políticas al utilizar programación dinámica.

En este problema, siguiendo el esquema de la programación determinista y discreta en el espacio de los estados [6]:

$$\mathbf{B}+\mathbf{A}\cdot\mathbf{W} = \mathbf{g}+\mathbf{W} \tag{14}$$

Así pues, interesa **g**=**0** dado que se busca el punto doble **W**.

En particular, también se puede iterar en el espacio de las políticas. Entonces, el uso de problemas lineales permite interpretar soluciones posibles del problema primal, que en el problema dual corresponden a políticas.

Se consideran las políticas con un solo circuito en el subgrafo $\mathbf{G}_\mathbf{l}$, dado que la iteración en el espacio de las políticas tiende a ello En caso de varios circuitos, la corrección debe basarse en el mayor valor p·g>0, siendo *p* la longitud del circuito y *g* la variación del valor de referencia t_0.

Para determinar un circuito, a partir de una política, puede hacerse, a partir de un vértice cualquiera, retrocediendo en el sentido de la relación. Además, se determinarán las componentes conexas asociadas.

Si considerada la transformación como un proceso polietápico de decisión con ecuación de recurrencia progresiva Φ_l, una política conduce a un valor g>0 el circuito que determina tiene valor p·g = s > 0. Esto indica el interés en modificar en general, y en aumentar TC en este caso, si éste puede corregirse.

Si al intentar corregir el valor de un circuito s>0, la longitud de éste es independiente de TC, **G** no es coherente para ningún valor de TC: la secuencia de movimientos es infactible. También puede ser detectado al acotar la variable TC [4].

En caso que para la política óptima g<0, también habrá que corregir el valor de TC en general, y en particular disminuirlo. No obstante, esto puede modificar el circuito óptimo u otro, que pase a ser óptimo al aumentar su valor *g*. Para ser prudente, es mejor empezar por la cota inferior de TC e ir aumentando su valor.

5 Conclusión.

En resumen, la transformación $\Phi_{I,l}$ es la forma progresiva de la ecuación de recurrencia (15) utilizada en la programación dinámica determinista y discreta, en el espacio de los estados:

$$\mathbf{g} + \mathbf{W} = \Phi_{I,l}(\mathbf{W}) \tag{15}$$

cumpliendo **g** ciertas condiciones, y que en caso de un circuito único en el subgrafo $\mathbf{G}_\mathbf{l}$:

$$\mathbf{g} = g\cdot\mathbf{1} \tag{16}$$

Referencias

[1] Phillips, L.W., Unger, P.S., (1976), "Mathematical programming solution of a hoist scheduling program", *AIIE Transactions*, vol. 8, nº 2, pp. 219-225.

[2] Shapiro, G.W., Nuttle, H.W., (1988), "Hoist Scheduling for a PCB electroplating facility", *IEE Transactions*, vol. 20, nº 2, pp. 157-167.

[3] Chen, H., Chu, C., Proth, J.-M., (1995), "Cyclic hoist scheduling based on graph theory", *IEEE 0-7803-2535-4/95*, pp. 451-459.

[4] Mateo, M., Companys, R., Bautista, J., (2000), Bounded Cycle Time for the Cyclic Hoist Scheduling Problem, *I World Conference on Production and Operations Management* (Sevilla).

[5] Mateo, M., (2001), *Procedimientos de secuenciación y programación en un sistema productivo de estaciones en serie con transportadores asíncronos de material*, Tesis Doctoral, Universitat Politècnica de Catalunya (Barcelona).

[6] Companys, R. 2000, *Programación dinámica*, (Barcelona, CPDA).