



**4. ESCENARIO II:
PROGRAMACIÓN DE
TRABAJOS FIJOS CON
MÁQUINAS UNIFORMES
EN PARALELO**

4. ESCENARIO II: PROGRAMACIÓN DE TRABAJOS FIJOS CON MÁQUINAS UNIFORMES EN PARALELO

4.1 INTRODUCCIÓN

Se aborda a continuación un problema FSP con máquinas uniformes en paralelo. Se trata de un problema *con una clase de máquina*¹, en el cual la palabra uniforme indica que éstas presentan diferentes velocidades de procesado. Dos máquinas distintas, de ser asignadas a un mismo trabajo, tardarían distinto tiempo en ejecutarlo.

Se plantea a continuación un problema conformado por 5 trabajos (J1,...,J5) y dos máquinas uniformes de la misma clase (M1 y M2), con factores de velocidad 1 y 2 respectivamente. Como dato de partida se tiene el instante de inicio de los trabajos (s_i), los pesos de éstos (w_i), y dos posibles intervalos de duración de los trabajos: uno para el procesado con la primera máquina (p_{i1}), otro para el procesado con la segunda (p_{i2}). En línea discontinua se representa cada trabajo en caso de ser realizado por M1 y en línea continua en caso de ser realizado por la M2 (Figura 15). Nótese que los trabajos, de ser realizados por la M1, tendrían el doble de duración que si fueran realizados por M2.

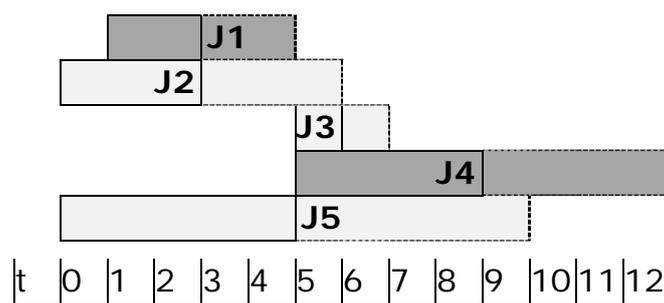


Figura 15

Sean los pesos (Tabla 5):

Nº de Trabajo	J1	J2	J3	J4	J5
Peso (w_i)	3	1	4	2	7

¹ Para más información sobre este tipo de problema, ver Capítulo 2.

Tabla 5

Ante un objetivo operacional que pretenda maximizar la suma de los pesos del total de trabajos realizados, una posible solución sería (Figura 16):

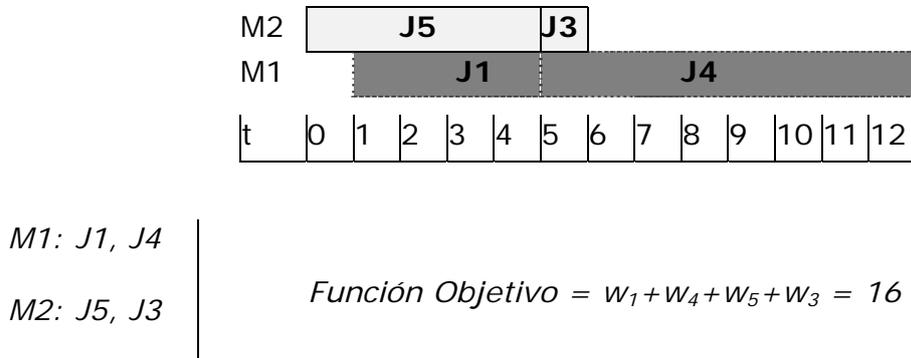


Figura 16

Este escenario es típico del entorno industrial, en el cual un mismo trabajo puede ser realizado por máquinas de distinta generación, y por consiguiente con distintas velocidades de procesado. Otro ejemplo de aplicación sería el caso en que haya operarios expertos y operarios noveles a la hora de realizar un cierto trabajo, por lo que lo ejecutarían en distintos tiempos.

4.2 PROBLEMA

El caso tratado consta de n trabajos a ser procesados por m máquinas. Cada trabajo tiene una fecha de comienzo (s_i) y un peso asociado a su realización (w_i). Cada máquina posee una velocidad de procesado particular, a la cual denotaremos como v_j , siendo j el índice que indica de qué máquina se trata. Una vez que un trabajo i es asignado a una máquina j , éste es procesado sin interrupción entre s_i y f_i ($f_i = s_i + p_{ij}$). p_{ij} se define como el tiempo de procesado del trabajo i en la máquina j , y es el producto del tiempo básico de procesado del trabajo (p_i) y el factor de velocidad de la máquina (v_j). Se asume que cada trabajo puede ser asignado como máximo a una máquina, y que éstas están disponibles en todo el horizonte de planificación. El objetivo es seleccionar una serie de trabajos y asignarlos a máquinas de forma tal de maximizar el peso total de las tareas realizadas. Los modelos encontrados habitualmente en la literatura asumen que las máquinas son idénticas y con factor de velocidad común $v_j = 1$.

Excepcionalmente, en Nakajima et al² se estudió el objetivo táctico del problema con dos tipos de máquinas: baratas y lentas, rápidas y caras. Bekki et al³, con la intención de completar estos estudios en el campo operacional, presentaron el modelo que se detalla más adelante, desarrollando un algoritmo Branch & Bound para su resolución.

El modelo planteado por Bekki et al usa restricciones de solapamiento dos a dos, es decir una para cada par de trabajos solapados, y usa un algoritmo Branch & Bound específico para el problema en cuestión. A la vista de esto, surge como interrogante si este mismo problema podría resolverse más rápidamente usando restricciones que restrinjan los trabajos solapados no dos a dos sino todos a la vez, pudiendo recurrir ahora al uso de librerías XA⁴, sin necesidad de crear un algoritmo específico.

4.3 EL MODELO

4.3.1 OBSERVACIONES

A continuación se presentan dos modelos de resolución del caso introducido más arriba.

El primero, al que también denominaremos modelo *de partida*, es el presentado por Bekki et al. El segundo, o modelo *adaptado*, es la alternativa aquí estudiada al modelo de partida, con el objetivo de estudiar los tiempos de resolución devueltos en ambos casos y las técnicas de resolución empleadas. En este último las restricciones de solapamiento de trabajos, que indican que una máquina puede realizar un solo trabajo a la vez, se realizan para cada máquina para todos los trabajos simultáneos, y no con trabajos dos a dos como en el modelo de partida.

En cuanto a la nomenclatura, por mantener coherencia con el resto del trabajo, se tratará con trabajos (tareas) y máquinas (recursos).

² En Nakajima et al, 1982.

³ En Bekki, Ö.B. et al, 2007.

⁴ Para información sobre librerías XA, ver Anexo 1.

4.3.2 MODELO DE PARTIDA

Variables:

- x_{ij} Variable binaria $\{0,1\}$, que toma el valor de la unidad si el trabajo i lo realiza la máquina j .
- x_{kj} Variable binaria $\{0,1\}$, que toma el valor de la unidad si el trabajo k lo realiza la máquina j .

Datos:

- s_i Instante de comienzo del trabajo i .
- f_i Instante de finalización del trabajo i .
- w_i Peso asociado al trabajo i .
- v_j Factor de velocidad asociado a la máquina j .
- p_k Tiempo básico de procesado del trabajo k .

Restricciones:

$$\sum_j x_{ij} \leq 1 \quad \forall i \quad (4.1)$$

Indica que cada trabajo es asignado como máximo a una máquina.

$$x_{kj} + x_{ij} \leq 1 \quad \forall i, j, k \{r_k \leq r_i < r_k + v_j \cdot p_k\} \quad (4.2)$$

Indica que los trabajos solapados no sean realizados por una misma máquina. Los restringe dos a dos.

Función Objetivo:

$$Max \sum_i \sum_j w_i \cdot x_{ij} \quad (4.3)$$

Modelo:

$$Max \sum_i \sum_j w_i \cdot x_{ij}$$

$$\sum_j x_{ij} \leq 1 \quad \forall i$$

$$x_{kj} + x_{ij} \leq 1 \quad \forall i, j, k \{r_k \leq r_i < r_k + v_j \cdot p_k\}$$

4.3.3 MODELO ADAPTADO

Variables:

x_{ij} Variable binaria $\{0,1\}$, que toma el valor de la unidad si el trabajo i lo realiza la máquina j .

Datos:

s_i Instante de comienzo del trabajo i .
 f_i Instante de finalización del trabajo i .
 w_i Peso asociado al trabajo i .
 v_j Factor de velocidad asociado al recurso j .
 p_i Tiempo básico de procesado del trabajo i .

Restricciones:

$$\sum_j x_{ij} \leq 1 \quad \forall i \quad (4.4)$$

Indica que cada trabajo es asignado como máximo a una máquina.

$$\sum_i x_{ij} \leq 1 \quad \forall i, j, \{r_i \leq r_k < r_i + v_j \cdot p_i\} \quad (4.5)$$

Indica que los trabajos solapados no sean realizados por una misma máquina. Una misma restricción para todos aquellos que sean simultáneos.

Función Objetivo:

$$\text{Max} \sum_i \sum_j w_i \cdot x_{ij} \quad (4.6)$$

Maximiza el peso total de los trabajos realizados.

Modelo:

$$\text{Max} \sum_i \sum_j w_i \cdot x_{ij}$$

$$\sum_j x_{ij} \leq 1 \quad \forall i$$

$$\sum_i x_{ij} \leq 1 \quad \forall i, j, \{r_i \leq r_k < r_i + v_j \cdot p_i\}$$

4.4 RESULTADOS

4.4.1 GENERACIÓN DE PROBLEMAS⁵

Se generan problemas aleatorios con un número de trabajos entre 20 y 70, incrementados de a 10 unidades. Se usan 2, 3 y 5 máquinas. Para el caso en que se dispone de 2 máquinas, se estudia la posibilidad de que sus velocidades sean respectivamente (1,2) o (1,3). Al observarse que para la combinación de velocidades (1,2) los problemas son más difíciles de resolver, se usan los factores (1,2,3) cuando se cuenta con 3 máquinas. y (1,2,3,4,5) cuando se cuenta con 5 máquinas, es decir las instancias más complicadas.

En cuanto a los pesos (w_i) y la duración de los trabajos (p_i), se estudian las cuatro combinaciones siguientes:

Caso 1: $w_i \in [1, 10]$; $p_i \in [5, 10]$

Caso 2: $w_i \in [1, 10]$; $p_i \in [5, 20]$

Caso 3: $w_i \in [1, 50]$; $p_i \in [5, 10]$

Caso 4: $w_i \in [1, 50]$; $p_i \in [5, 20]$

Para cada combinación posible, según lo especificado hasta ahora, se genera aleatoriamente una serie de 10 problemas. Al analizar las soluciones, se realiza el promedio de los resultados de cada uno de los diez problemas equivalentes, para obtener soluciones más fiables que de estudiar un solo problema aleatorio de cada clase.

La batería de problemas aquí generada es idéntica a la utilizada por Bekki et al, para poder comparar adecuadamente los resultados obtenidos.

4.4.2 COMPARACIÓN DE RESULTADOS

En primer lugar se resuelven en XA el modelo de partida y el modelo adaptado. Si bien las soluciones al modelo de partida presentadas por

⁵ Para más información sobre las baterías de problemas, incluyendo nomenclatura, ver Anexo 3.

Bekki, resueltas con su algoritmo Branch & Bound específico, son mejores que las devueltas por XA para su modelo, es interesante analizar en primer lugar la resolución en XA de ambos modelos (de partida y adaptado) para hacer una primera comparación de los dos casos.

Con la resolución en XA del modelo de partida y el modelo adaptado, se obtienen los mismos resultados en cuanto al número de trabajos

Realizados, por lo que éstos no son incluidos en la tabla. Interesa comparar los *Tiempos de Resolución* obtenidos (Tabla 6):

Número de Máquinas:			2		2		3		5	
Velocidades:			(1,2)		(1,3)		(1,2,3)		(1,2,3,4,5)	
			Tiempo de Resolución (s)							
N	Pi	wi	P-XA	A-XA ⁶	P-XA	A-XA	P-XA	A-XA	P-XA	A-XA
20	[5,10]	[1,10]	3	1	2	1	5	2	161	1
		[1,50]	1	1	1	1	3	1	90	1
	[5,20]	[1,10]	1	2	1	1	13	1	923	1
		[1,50]	1	1	2	1	10	1	633	1
30	[5,10]	[1,10]	13	1	7	1	1187	1	20185	1
		[1,50]	12	1	15	1	24	1	---	1
	[5,20]	[1,10]	31	1	29	1	336	1	---	2
		[1,50]	28	1	3	1	67	1	---	1
40	5,10]	[1,10]	145	2	140	1	4893	1	---	1
		[1,50]	78	2	166	1	21362	1	---	1
	[5,20]	[1,10]	85	1	86	1	---	1	---	1
		[1,50]	62	1	68	1	---	1	---	1
50	[5,10]	[1,10]	100	1	472	1	---	1	---	2
		[1,50]	5620	1	399	1	---	1	---	1
	[5,20]	[1,10]	695	1	464	1	---	1	---	1
		[1,50]	411	0	304	1	---	2	---	1
60	[5,10]	[1,10]	2305	1	---	1	---	1	---	1
		[1,50]	---	2	---	1	---	1	---	1
	[5,20]	[1,10]	---	2	---	1	---	1	---	1
		[1,50]	---	1	---	1	---	1	---	1
70	[5,10]	[1,10]	---	1	---	1	---	1	---	1
		[1,50]	---	1	---	1	---	2	---	1
	[5,20]	[1,10]	---	1	---	1	---	2	---	1
		[1,50]	---	1	---	2	---	1	---	1

Tabla 6

Los tiempos de resolución usando el modelo adaptado se encuentran para todos los casos entre 1 y 2 segundos. Los tiempos de resolución del modelo de partida son notablemente superiores. En el modelo de partida, el tiempo

⁶ De los diez problemas aleatorios del mismo tipo resueltos para el modelo A-XA (p.e: File1_20_1_0, File 1_20_1_1, File1_20_1_2..., File1_20_1_9, se incluye en esta tabla un resultado orientativo de entre los 10 obtenidos. Respetando la nomenclatura usada, el último número se indica con "s" de forma genérica.

⁷ --- significa que no se ha hallado solución en menos de 12 hs.

de resolución al aumentar el número de trabajos realizados y el número de máquinas uniformes disponibles. El crecimiento asociado al número de trabajos se debe a que este fenómeno, a iguales p_i , produce un aumento del solapamiento entre trabajos, lo cual se traduce en un aumento del número de restricciones. El aumento del tiempo de resolución asociado a un mayor número de máquinas uniformes es mucho más pronunciado que el anterior, lo cual es razonable considerando que el número de restricciones con una máquina se ve multiplicado por el número de máquinas con que contemos. Al ser el número de máquinas uniformes el factor más influyente en el aumento del tiempo de resolución, incluso el aumento asociado al número de trabajos es más pronunciado cuantas más máquinas se tengan. En cuanto a los 4 casos utilizados, no se observa tendencia asociada a éstos. Cabe destacar que modelo adaptado arroja los mismos resultados en un tiempo notablemente menor.

Este análisis es interesante para ver que el modelo adaptado que se propone en este proyecto es interesante y buen candidato a ser comparado con los resultados arrojados por el algoritmo Branch & Bound usado por Bekki et al.

Se comparan en la Tabla 7 los resultados obtenidos por Bekki con el algoritmo Branch & Bound específico con el que resuelve su modelo, o modelo de partida (P-B&B), y los obtenidos con XA según el modelo propuesto en este proyecto, o modelo adaptado (A-XA):

Número de Máquinas:		2	2	3	5					
Velocidades:		(1,2)	(1,3)	(1,2,3)	(1,2,3,4,5)					
		Tiempo de Resolución (s)								
N	pi	wi	P-B&B	A-XA	P-B&B	A-XA	P-B&B	A-XA	P-B&B	A-XA
20	[5,10]	[1,10]	0	1 ⁸	0	1	1	2	1	1
		[1,50]	0	1	0	1	0	1	0	1
	[5,20]	[1,10]	0	2	0	1	0	1	0	1
		[1,50]	0	1	0	1	0	1	0	1
30	[5,10]	[1,10]	0	1	0	1	21	1	50	1
		[1,50]	0	1	0	1	2	1	64	1
	[5,20]	[1,10]	0	1	1	1	4	1	340	2
		[1,50]	0	1	1	1	5	1	309	1
40	5,10]	[1,10]	0	2	11	1	594	1	---9	1
		[1,50]	0	2	22	1	1013	1	---	1
	[5,20]	[1,10]	0	1	13	1	1070	1	0	1
		[1,50]	0	1	13	1	1203	1	---	1
50	[5,10]	[1,10]	11	1	10	1	---	1	---	2
		[1,50]	13	1	11	1	---	1	---	1
	[5,20]	[1,10]	10	1	4	1	---	1	---	1
		[1,50]	12	0	5	1	---	2	---	10
60	[5,10]	[1,10]	386	1	380	1	---	1	---	1
		[1,50]	390	2	456	1	---	1	---	1
	[5,20]	[1,10]	114	2	56	1	---	1	---	1
		[1,50]	122	1	53	1	---	1	---	1
70	[5,10]	[1,10]	---	1	210	1	---	1	---	1
		[1,50]	----	1	----	1	----	2	----	1
	[5,20]	[1,10]	435	1	30	1	---	2	---	1
		[1,50]	483	1	32	2	---	1	---	1

Tabla 7

Puede verse que para los problemas de 20 trabajos, independientemente de la clase a la que pertenezcan y del número de máquinas uniformes con que cuenten, el modelo de partida funciona perfectamente, incluso mejor que el adaptado. Para problemas de 30 trabajos y 3 o más máquinas, los tiempos de ejecución del modelo de partida ya empiezan a presentar mayores tiempos de resolución que los del modelo adaptado. El crecimiento de los tiempos de resolución, al igual que lo observado en la resolución con XA de modelo de partida, se debe tanto al aumento del número de trabajos como al del número de máquinas uniformes. Una vez más, el incremento asociado al número de máquinas es más pronunciado, y la explicación vuelve a ser la entonces mencionada. Nótese que para todos los números de trabajos aquí

⁸ De los diez problemas aleatorios del mismo tipo resueltos para el modelo A-XA (p.e: File1_20_1_0, File 1_20_1_1, File1_20_1_2..., File1_20_1_9, se incluye en esta tabla un resultado orientativo de entre los 10 obtenidos. Respetando la nomenclatura usada, el último número se indica con "s" de forma genérica.

⁹ --- significa que no se ha hallado solución en menos de 12 hs.

propuestos se ha hallado solución numérica, omitiendo algún caso excepcional, mientras que para un número de máquinas mayor que 2, son numerosas las ocasiones en que no se obtiene solución en menos del tiempo máximo estipulado.

