Cátedra Nissan

-PROTHIUS-

Retorno al teorema de Johnson

Ramón Companys Pascual y Joaquín Bautista Valhondo

WP-17/2010
(Rec. DIT 99/03- CB- 1999)

Departamento de Organización de Empresas

Universidad Politécnica de Cataluña

Publica:

Universitat Politècnica de Catalunya
www.upc.edu

Edita:

Cátedra Nissan
www.nissanchair.com
director@nissanchair.com

TÍTOL:

RETORNO AL TEOREMA DE JOHNSON

Autores: R. Companys
J. Bautista

Document Intern de
Treball
(D.I.T.) 99/03

RETORNO AL TEOREMA DE JOHNSON

Ramon Companys Pascual y Joaquín Bautista Valhondo
DOE - ETSEIB
UNIVERSITAT POLITECNICA DE CATALUNYA

1. INTRODUCCION

En la resolución de un problema de programación con flujo regular deseabamos utilizar una cota basada en la obtención de F_{max} en el problema $n/2/F/F_{max}$ (específicamente se trataba de la cota de Nabeshima para el problema $n/m/P/F_{max}$ que se describe en el Anexo IV) y encontramos al aplicar nuestra primera formulación ciertas incongruencias en los resultados, alguna vez el valor de la función era mejor que el de la cota, lo que siempre resulta inquietante. A primera vista parecía que habíamos utilizado propiedades que en toda la literatura se enuncian como ciertas. El análisis de las causas de las incongruencias nos llevó a detectar algunos puntos delicados referentes a la transitividad de la condición de Johnson, extremo probablemente advertido por algunos autores por cuanto la relectura de sus textos nos mostró cierta incomodidad al tratar el tema. Más aún los algoritmos usualmente propuestos para la resolución del problema (incluso el propuesto inicialmente por Johnson) no derivan directamente de la condición de Johnson sino que se basan en la que denominamos condición de French. Incluso algunos autores soslayan la condición de Johnson (que es más general) y únicamente enuncian y demuestran dicha última condición. No habíamos detectado anteriormente las incongruencias en la transitividad de la condición de Johnson por nuestra inveterada costumbre de utilizar una regla de desempate (análoga a la que más adelante denominaremos [CA1]), que precisamente resuelve la cuestión. En esta ocasión, sin embargo, habíamos soslayado dicha regla por no creer, erróneamente, en su utilidad para el caso.

El presente artículo tiene por objeto exponer algunas de nuestras conclusiones sobre este tema.

2. CONDICION DE JOHNSON

El problema que consideramos es el $n/2/F/F_{max}$, con todas las piezas y máquinas disponibles en el instante 0, y denominamos, para simplificar, a_i la duración de la operación de la pieza i (i=1,2, ... ,n) en la primera máquina (en lugar de la notación habitual $p_{1,i}$) y b_i la de la operación en la segunda máquina (en lugar de $p_{2,i}$). Sabemos, a partir del primer lema de Johnson que siempre existe una solución óptima en la que las secuencias de las piezas en ambas máquinas son idénticas: la misma permutación de las n piezas (lo que no descarta la existencia en algunos casos de soluciones óptimas en que ello no sea así). Johnson afirma que son óptimas las permutaciones en las que para cada pareja de piezas h e i tales que:

$$\min \{ a_h , b_i \} < \min \{ b_h , a_i \} \qquad [CJ]$$

la pieza h precede a la pieza i,

Theorem 1. An optimal ordering is given by the following rule. Item (i) precedes item (i+1) if ......

$$\min \{ a_i , b_{i+1} \} < \min \{ b_i , a_{i+1} \}$$

This ordering is transitive (proof follows), thus to leading to sequence S^*, unique except for some indifferent elements.

S. M. Johnson, "Optimal two- and three-stage production schedules with set-up time included"; *Naval Research Logistic Quarterly* **1**, 1954. (Tomado de J. F. Muth and G. L. Thompson, *Industrial Scheduling*, Prentice-Hall, 1963, pp. 15-16).

Lo anterior no descarta la existencia de soluciones óptimas que no cumplan la condición. Por ejemplo en el siguiente problema $3/2/F/F_{max}$:

ejemplo 1

i	**1**	**2**	**3**
$\mathbf{a_i}$	1	2	4
$\mathbf{b_i}$	3	2	2

tanto la permutación 1 - 2 - 3 como las 1 - 3 - 2 y 2 - 1 - 3 son óptimas, con $F_{max} = 9$, y sin embargo la tercera no satisface la condición pues:

$$\min \{ a_2 , b_1 \} = \min \{ 2 , 3 \} = 2$$

$$\min \{ b_2 , a_1 \} = \min \{ 2 , 1 \} = 1$$

En las tablas que siguen hemos indicado los instantes de finalización de las operaciones de cada pieza en cada máquina en los programas activos asociados a las permutaciones.

ocupación

permutación	**1**	**2**	**3**
máquina 1	1	3	7
máquina 2	4	6	9

$F_{max} = 9$

permutación	**1**	**3**	**2**
máquina 1	1	5	7
máquina 2	4	7	9

$F_{max} = 9$

permutación	**2**	**1**	**3**
máquina 1	2	3	7
máquina 2	4	7	9

$F_{max} = 9$

La demostración de la afirmación de Johnson (algo desvirtuada en su texto al denominar las piezas i e i+1 que dan imagen de piezas contiguas) puede realizarse, a nuestro entender, de la siguiente forma en tres fases:

F1 - demostrando que siempre existe una permutación (por lo menos) que cumple la condición,

F2 - demostrando que cualquier permutación puede transformarse, en un número finito de pasos, en una de las que cumple la condición, sin empeorar el valor de F_{max},

F3 - demostrando que todas las permutaciones que cumplen la condición tienen el mismo valor F_{max}.

Alternativamente, podemos obviar F3, si demostramos en F2 que la transformación conduce en todos los casos a la misma permutación que cumple la condición. Estea es la vía elegida por nosotros más adelante. F1 parece a primera vista trivial, pero la posibilidad de empates (cumplimiento de [CJ] con el signo igual) y los problemas con la transitividad que describiremos más adelante, exigen un cierto cuidado.

En cambio no resulta complicado establecer que pueden existir varias permutaciones que cumplan la condición [CJ] si está garantizado que existe una que la cumpla. En efecto, si en una permutación que cumple [CJ] existen dos piezas consecutivas h e i tales que:

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

Las piezas h e i pueden permutarse sin que se viole [CJ] ni entre estas dos piezas ni respecto a todas las demás. Por tanto, en la medida que existan empates, es necesario identificar una permutación concreta que cumpla la condición, para ajustarnos a la vía elegida con el enunciado alternativo de F2. Por ello añadiremos unas condiciones suplementarias, en caso de que entre dos piezas h e i:

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

y por tanto [CJ] no concluya respecto al orden entre h e i,

si $a_h - b_h < a_i - b_i$	h precede a i	[CA1]
si $a_h - b_h = a_i - b_i$ y $h < i$	h precede a i	[CA2]

El conjunto de las condiciones [CJ], [CA1] y [CA2] (condición de Johnson ampliada [CJA]) definen una relación de orden total entre las piezas, dadas dos piezas h e i quedará establecido que h precede a i o bien que i precede a h, y además esta precedencia es transitiva (ver Anexo 1 para la demostración). Por consiguiente esta relación de orden define una permutación única que cumple la condición de Johnson. Queda por tanto demostrada la existencia de una permutación, por lo menos, que cumple la condición de Johnson, y la existencia y unicidad de una permutación que cumple la condición de Johnson ampliada, que denominaremos PJA.

La demostración clásica de Johnson de que una permutación que cumple [CJ] minimiza F_{max} se basa en probar que si en una permutación cualquiera existen dos piezas contiguas h e i tales que:

$$\min \{ a_h , b_i \} > \min \{ b_h , a_i \}$$

el cambio del orden entre h e i conduce a otra permutación que cumple [CJ] en dichas piezas sin empeorar el valor F_{max} (el valor F_{max} de la nueva permutación es menor o igual al valor F_{max} de la anterior). La demostración se detalla en el Anexo II, y adicionalmente, también prueba que si:

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

el cambio de orden entre h e i no altera el valor de F_{max}.

Por consiguiente la aplicación sistemática de la condición de Johnson ampliada a una permutación cualquiera, invirtiendo el orden de las parejas de piezas contiguas que no la cumplen, la transforma en un número finito de pasos en la permutación PJA sin empeorar en ninguno de ellos el valor F_{max}. Por tanto la permutación PJA es óptima. También serán óptimas todas aquellas permutaciones que puedan deducirse de PJA invirtiendo el orden de las piezas contiguas tales que:

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

Todas las permutaciones que cumplen la condición de Johnson, dada la transitividad de PJA, pueden obtenerse de esta forma, por tanto todas ellas son óptimas.

Notemos que la aplicación sistemática de la condición de Johnson (no ampliada) a piezas contiguas no garantiza que la permutación resultante cumpla la condición de Johnson entre cualquier par de piezas. Sea el problema $3/2/F/F_{max}$ siguiente:

ejemplo 2

i	**1**	**2**	**3**
a_i	26	20	31
b_i	20	20	30

La permutación 1 - 2 - 3 cumple la condición de Johnson entre las parejas contiguas:

$$\min \{ 26 , 20 \} = 20 = \min \{ 20 , 20 \}$$
$$\min \{ 20 , 30 \} = 20 = \min \{ 20 , 31 \}$$

pero no entre la primera y tercera piezas:

$$\min \{ 26 , 30 \} = 26 > \min \{ 20 , 31 \} = 20$$

en consecuencia 107 (su valor de F_{max}) no es mínimo; la permutación 3 - 2 - 1 que satisface la condición entre las tres parejas tiene un valor de F_{max} inferior, 101.

ocupación

permutación	**1**	**2**	**3**
máquina 1	26	46	77
máquina 2	46	66	107

$F_{max} = 107$

permutación	3	2	1
máquina 1	31	51	77
máquina 2	61	81	101

$F_{max} = 101$

La falta de transitividad es causada por la existencia de la pieza 2, en que $a_2 = b_2 = 20$.

La forma de la condición de Johnson implica que si para un ejemplar del problema $n/2/F/F_{max}$ disponemos de una permutación que la satisface y por tanto es óptima, si eliminamos una pieza cualquiera, en el ejemplar del problema $(n-1)/2/F/F_{max}$ correspondiente se obtiene una solución óptima eliminando en aquella permutación la pieza y manteniendo el orden de las demás (aunque aquella pieza no sea la situada en uno de los extremos).

3. CONDICION DE FRENCH

El enfoque de French (que éste atribuye a Rinooy Kan) es distinto, afirma que una permutación, por ejemplo 1 - 2 - .... - n es óptima si se cumple:

$$\min \{ a_h, a_{h+1}, \ldots ,a_i, b_h, b_{h+1}, \ldots , b_i \} = \min \{ a_h, b_i \} \qquad \forall\ h,i \text{ con } h < i \qquad [CF]$$

La demostración autónoma de este resultado puede verse en el Anexo III pero una vía directa puede basarse en la observación de que la formulación anterior implica que si se cumple [CF] se cumple:

$$\min \{ a_h , b_i \} \leq \min \{ b_h , a_i \}$$

para toda pieza h que precede a toda pieza i, lo que coincide con la condición [CJ]. Obviamente siempre existe una permutación que satisface [CF].

Hemos demostrado, por tanto, que la condición de French implica la condición de Johnson, y que la permutación que satisface [CF] es óptima. El recíproco no es cierto, una permutación puede cumplir la condición de Johnson (y ser óptima) sin cumplir la condición de French. Sea el siguiente problema $3/2/F/F_{max}$:

ejemplo 3

i	1	2	3
a_i	31	20	27
b_i	30	20	25

La permutación 1 - 2 - 3 cumple la condición de Johnson:

$$\min \{ 31 , 20 \} = 20 = \min \{ 30 , 20 \}$$
$$\min \{ 31 , 25 \} = 25 < \min \{ 30 , 27 \} = 27$$
$$\min \{ 20 , 25 \} = 20 = \min \{ 20 , 27 \}$$

sin embargo no cumple la de French:

$$\min \{ 31, 20, 27, 30, 20, 25 \} = 20 < \min \{ 31 , 25 \} = 25$$

Existen dos permutaciones que cumplen la condición de French son 2 - 1 - 3 y 1 - 3 - 2; estas más la anterior son óptimas con $F_{max} = 106$

ocupación

permutación	**1**	**2**	**3**
máquina 1	31	51	78
máquina 2	61	81	106

$F_{max} = 106$

permutación	**2**	**1**	**3**
máquina 1	20	51	78
máquina 2	40	81	106

$F_{max} = 106$

permutación	**1**	**3**	**2**
máquina 1	31	58	78
máquina 2	61	86	106

$F_{max} = 106$

Por consiguiente la condición de Johnson es más general que la de French.

4. DIFICULTADES EN LAS DEMOSTRACIONES CLASICAS

En varios textos clásicos se utiliza un argumento falaz para demostrar la optimalidad de la condición de Johnson. Directa (Baker, 1974) o indirectamente (Conway, Maxwell and Miller, 1967) se asume que en una permutación que no cumple la condición de Johnson existen dos piezas contiguas h e i tales que:

$$\min \{ a_h , b_i \} > \min \{ b_h , a_i \}$$

lo cual ha quedado desmentido por el ejemplo 2.

> Now consider a schedule S, which does not satisfy the ordering prescribed by Theorem 6.1. This means that there exist a pair of adjacents jobs h and i, with i following h, such that
>
> $$\min \{ a_h , b_i \} > \min \{ b_h , a_i \}$$
>
> K. R. Baker, *Introduction to sequencing and scheduling*, Wiley, 1974, pp. 145

En todos los casos la argumentación necesita apoyarse en la transitividad de la relación que garantizaría la existencia de este par de piezas contiguas que no satisfacen [CJ], pero la transitividad no está asegurada en el caso:

$$a_i = b_i = \min \{ a_h , b_i \} = \min \{ a_i , b_j \}$$

pues entonces de

$$\min \{ a_h , b_i \} \leq \min \{ b_h , a_i \}$$
$$\min \{ a_i , b_j \} \leq \min \{ b_i , a_j \}$$

no puede deducirse

$$\min \{ a_h , b_j \} \leq \min \{ b_h , a_j \}$$

Aquí suelen argüir algunos autores que i no induce orden respecto a h o j y que puede situarse en cualquier posición, lo que es cierto respecto a h o j pero no respecto a otras piezas. En el ejemplo 4 (una ampliación del ejemplo 3), la pieza 4 no induce orden respecto la 3 ni la 5 (sus contiguas) ni tampoco respecto a la 2, sin embargo, en virtud de la condición de Johnson, 1 debe preceder a 4 y 4 debe preceder a 6.

ejemplo 4

i	**1**	**2**	**3**	**4**	**5**	**6**
$\mathbf{a_i}$	15	22	31	20	27	25
$\mathbf{b_i}$	18	25	30	20	25	18

Sin embargo los autores afirman:

> The case 3 in the above proof occurs only when the assumed inequalities are booth equalities and $a_i = b_i$ holds. And in this case the orders of two jobs h, i and two jobs i, j are arbitrary respectively. We may regard that job h precedes job j. ....
>
> Bellman, Esogbue and Nabeshima, *Mathematical aspects of scheduling and applications*, Pergamon Press, 1982, pp. 148-149.

o bien:

> Theorem 5-3 (Johnson 1954): In an $n/2/F/F_{max}$ problem with all jobs simultaneous avalaible, job j should precede job j+1 if
>
> $$\min \{ a_j , b_{j+1} \} < \min \{ b_j , a_{j+1} \}$$
>
> The proof is in two parts. The first shows that sequencing according this inequality minimizes the maximum of the Y_k's; the second illustrates that the relation is transitive so that a complete schedule is in fact specified by this pair wise ordering. ......
>
> Transivity of Relationship. We must show that for three jobs, h, i and j, if
>
> $$\min\{a_h , b_i\} \leq \min\{b_h , a_i\} \text{ and } \min\{a_i , b_j\} \leq \min\{b_i , a_j\}$$
> $$\text{then } \min\{a_h , b_j\} \leq \min\{b_h , a_j\}$$
>
> except possibly for cases in which the relationship is an equality between booth h and i and i and j
>
> Conway, Maxwell, Miller; *Theory of Scheduling*, Addison Wesley, 1967, pp. 86-87 (la parte omitida corresponde a una demostración semejante a la del Anexo II).

Johnson ya era consciente de este hecho, pues afirma, respecto al caso considerado:

> Then $a_i = b_i$ and we have item i indifferent to item h and item j. In this case, item h may or may not precede item j, but there is no contradiction to transitivity as long as we order item h and item j first and put item i anywhere.
>
> S. M. Johnson, *loc. cit.*, pp. 16

que "estrictu senso" podría interpretarse como una demostración de la no transitividad.

Los hay más sutiles:

> An optimal permutation can be found in O(n·log n) time by applying Johnson's Rule; a permutation is optimal if job h precedes job i whenever Johnson's Condition:
>
> $$\min \{ a_h , b_i \} < \min \{ b_h , a_i \}$$
>
> is satisfied. A simple proof is provided below. An optimal permutation always exists since Johnson's Condition can be seen to be transitive.

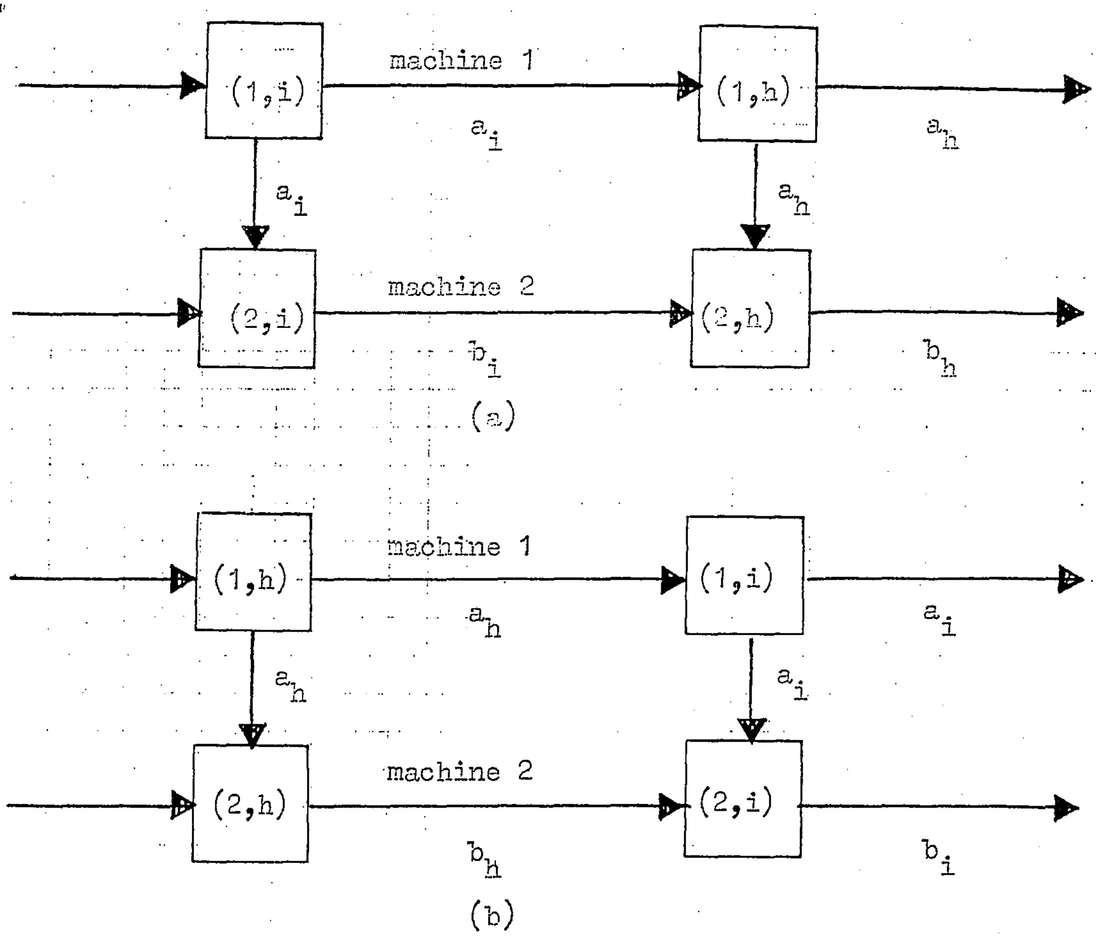

Figure 2. - Critical Paths for Two Machines.

We note that an optimal permutation ordered by Johnson's Rule has the property that an optimal permutation for any subset of the jobs is given by the order of these jobs in the original permutation. Also, the worst possible permutation is obtained by reversing the order obtained by Johnson's Rule.

The critical-path approach leads to a simple proof of Johnson's result. Consider the graph representing a permutation with job i immediately preceding job h as shown in figure 2(a). Interchanging these jobs, as shown in figure 2(b), does not increase the completion time of the schedule if the weight of no critical path is increased, i. e., if:

$$\max \{ a_h + a_i + b_i , a_h + b_h + b_i \} \leq \max \{ a_i + a_h + b_h , a_i + b_i + b_h \}$$

This is easily seen to imply Johnson's Condition.

C. L. Monma, A. H. G. Rinnooy Kan, "A concise survey of efficiently solvable special cases of the permutation flow-shop problem", *RAIRO*, Vol. 17, no. 2, mai 1983, pp.107.

o bien

Considerons maintenant un sequencement S associé à un ordre total non compatible avec [CJ], il existe deux travaux h et i tels que:

a) h et i sont deux tâches consécutives sur la machine 1;

b) $\min \{ a_h , b_i \} \geq \min \{ b_h , a_i \}$

Soit alors T le sequencement obtenu par échange des tâches h et i, i.e.: si $j \notin \{i,j\}$ $S^{-1}(j)=T^{-1}(j)$, $T^{-1}(h)=k+1$, $T^{-1}(i)=k$ avec $k=S^{-1}(h)$. .....

Nous terminons la démonstration en remarquant:

a) que l'on peut passer d'un sequencement quelconque non compatible avec [CJ] a un sequencement compatible avec [CJ] par une suite d'au plus n(n-1)/2 ,changes améliorants.

b) que tous les séquencements compatibles avec [CJ] ont le même délai; ils ne diffèrent, en effet, entre eux que par des échanges de travaux h et i pour lesquels

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

laissant invariant le critère. QED.

J. Carlier, Ph. Chrétienne; *Problèmes d'ordonnancement*; Masson, 1988, pp. 215-217 (la parte omitida corresponde a una demostración semejante a la del Anexo II)

Aunque el más espectacular es:

To complete de proof, the second fase is to demonstrate the transivity of the job ordering. In other words, supose $\min \{ a_h , b_i \} \leq \min \{ b_h , a_i \}$ and $\min \{ a_i , b_j \} \leq \min \{ b_i , a_j \}$. Then it will always be true that $\min \{ a_h , b_j \} \leq \min \{ b_h , a_j \}$. (The details of this part of the proof are given as Exercise 3.1).

K. R. Baker, *loc. cit.*, pp. 146

EXERCISE 3.1. Show that Johnson's rule is transitive.

K. R. Baker, *loc. cit.*, pp. 148

No cabe buscar una demostración estudiando el intercambio de piezas no contiguas. En el siguiente problema $3/2/F/F_{max}$:

ejemplo 5

i	1	2	3
a_i	5	3	8
b_i	4	5	7

comparando las piezas 1 y 3 obtenemos:

$$\min \{ 5 , 7 \} = 5 > \min \{ 4 , 8 \} = 4$$

pero permutandolas el valor F_{max} empeora:

ocupación

permutación	1	2	3
máquina 1	5	8	16
máquina 2	9	14	23

$F_{max} = 23$

permutación	3	2	1
máquina 1	8	11	16
máquina 2	15	20	24

$F_{max} = 24$

La permutación óptima es 2 - 3 - 1:

ocupación

permutación	2	3	1
máquina 1	3	11	16
máquina 2	8	18	22

$F_{max} = 22$

Bellman, Esogbue y Nabeshima en la obra citada (pp. 149-152, y anteriormente D. J. White, *Dynamic Programming*, Oliver and Boyd, 1969, pp. 71-74) adjuntan una demostración basada en la programación dinámica. Sea f(E,t) el tiempo mínimo de proceso de las piezas del conjunto E ($E \subset \{ 1,2, ...,n \}$) cuando la máquina 1 está disponible en el instante 0 y la máquina 2 en el instante t. Si la primera pieza de E que elaboramos es la h, entonces:

$$f(E,t) = ah + f(E - \{h\},t_h)$$

donde:

$$t_h = b_h + \max \{ t-a_h , 0 \}$$

Si detrás de h se procesa la pieza i:

$$f(E,t) = a_h + a_i + f(E-\{h,i\},t_{h,i})$$

donde:

$$
\begin{aligned}
t_{h,i} &= b_i + \max \{ t_h - a_i , 0 \} = \\
&= b_i + \max \{ \max \{ t + b_h - a_h - a_i , b_h - a_i \} , 0 \} = \\
&= b_h + b_i - a_i + \max \{ t - a_h , 0 , a_i - b_h \} = \\
&= b_h + b_i - a_h - a_i + \max \{ t , a_h , a_h + a_i - b_h \}
\end{aligned}
$$

Si el orden hubiese sido el contrario, i antes de h:

$$t_{i,h} = b_h + b_i - a_h - a_i + \max \{ t , a_i , a_h + a_i - b_i \}$$

Dado que si $t < t'$ se cumple $f(E,t) \leq f(E,t')$, será conveniente situar h antes de i si $t_{i,h} < t_{i,h}$ (en caso de igualdad el orden entre h e i será indiferente). Por tanto si:

$$\max \{ t , a_h , a_h + a_i - b_h \} < \max \{ t , a_i , a_h + a_i - b_i \}$$

convendrá situar h antes que i, lo cual se cumple si:

$$\max \{ a_h , a_h + a_i - b_h \} < \max \{ a_i , a_h + a_i - b_i \}$$

es decir si

$$\max \{ - a_i , - b_h \} < \max \{ -a_h , - b_i \}$$

o bien:

$$\min \{ a_h , b_i \} < \min \{ b_h , a_i \}$$

De esta última expresión puede deducirse que un criterio razonable para que una permutación sea óptima es que dadas dos piezas contiguas h e i si:

$$\min \{ a_h , b_i \} < \min \{ b_h , a_i \}$$

h preceda a i. Lamentablemente esta condición no es necesaria ni suficiente, ni queda demostrado que todas las permutaciones que la cumplan son óptimas, ni que no haya permutaciones óptimas que no la cumplan, a pesar de la afirmación:

> This condition for job h to precede job i is independent of E and t and hence is a sufficient condition for h to precede i anywhere in the sequence if they are adjacent.
>
> D. J. White, *loc. cit.*, pp. 73

5. ALGORITMOS

Los algoritmos propuestos, para obtener una permutación óptima, habitualmente son los siguientes:

algoritmo 1: búsqueda de mínimos (ordenación de French)

Inicialmente se establece una permutación cualquiera y se toma $k_1 = 1$ y $k_2 = n$. Se busca el mínimo de los valores a_i, b_i del segmento de la permutación comprendido entre k_1 y k_2, resolviendo los empates como se desee. Si el mínimo corresponde a un valor a_i la pieza i se sitúa en la posición k_1 y se hace $k_1 = k_1+1$; si el mínimo corresponde a un valor b_i la pieza i se coloca en la posición k_2 y se hace $k_2 = k_2-1$. Cuando $k_2 = k_1$ se ha obtenido una permutación óptima.

La complejidad de este algoritmo es O(n2). Este algoritmo es el propuesto por Johnson en su artículo de 1954 (*working rule*).

algoritmo 2: dos listas

Se establecen dos listas con las piezas:

lista U, con las piezas tales que $a_i \leq b_i$
lista V, con el resto de piezas.

Se ordena la lista U en orden creciente de a_i y la lista V en orden decreciente de b_i. Se concatenan las dos listas ordenadas (la lista U seguida de la lista V).

La complejidad de este algoritmo es $O(n \cdot \log n)$.

algoritmo 3: regla de Gupta

Se asocia a cada pieza un índice de la forma:

$$S_{4,i} = \frac{x}{min\{a_i, b_i\}}$$

donde x es igual a -1 si ai $\leq b_i$ y a +1 en caso contrario,

Las piezas se ordenan en orden creciente de $S_{4,i}$. (Este procedimiento coincide con la primera heurística de Gupta propuesta para el problema $n/3/P/F_{max}$).

La complejidad de este algoritmo es $O(n \cdot \log n)$.

algoritmo 4: ordenación de Johnson

Se toma una permutación cualquiera. Sistemáticamente se analizan parejas de piezas contiguas, h e i. Si:

$$\min \{ a_h , b_i \} > \min \{ b_h , a_i \}$$

o bien

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

y

$$a_h - b_h > a_i - b_i$$

se invierte el orden de h e i.

Cuando no se produce ninguna inversión de orden al recorrer toda la permutación el algoritmo finaliza y la permutación es óptima.

La complejidad de este algoritmo es O(n·log n).

Observese que en los algoritmos 2 y 3 se procura agrupar en una parte de la permutación (mediante la lista o mediante el signo) las piezas en las que $a_i = b_i$ a fin de evitar problemas con la transitividad.

6. CONCLUSION

Las incongruencias que nos preocupaban cesaron al utilizar el Algoritmo 4 descrito en el apartado anterior, pero nuestra confianza en los textos de los autores de renombre menguó considerablemente.

NOTA: todas las citas son textuales salvo retoques de la notación para homogeneizarla con la utilizada en este texto.

ANEXO I: TRANSITIVIDAD DE LA CONDICION DE JOHNSON AMPLIADA

Demostrar la transitividad exige demostrar que si h precede a i e i precede a j también h precede a j. La precedencia de una pieza, por ejemplo h, respecto a otra, por ejemplo i, puede resultar por tres vías diferentes correspondientes a grupos de condiciones distintos:

grupo [CJ]

$$\min \{ a_h , b_i \} < \min \{ b_h , a_i \}$$

grupo [CA1]

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

$$a_h - b_h < a_i - b_i$$

grupo [CA2]

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

$$a_h - b_h = a_i - b_i$$

$$h < i$$

Ello obliga a considerar los nueve casos posibles, resultantes de combinar las tres posibilidades de cada una de las precedencias:

	h precede i	i precede j
Caso 1_	[CJ]	[CJ]
Caso 2_	[CJ]	[CA1]
Caso 3_	[CJ]	[CA2]
Caso 4_	[CA1]	[CJ]
Caso 5_	[CA1]	[CA1]
Caso 6_	[CA1]	[CA2]
Caso 7_	[CA2]	[CJ]
Caso 8_	[CA2]	[CA1]
Caso 9_	[CA2]	[CA2]

Además, cada caso se subdivide en cuatro subcasos según qué pareja de los cuatro valores a_h, b_i, a_i, b_j corresponde a $\min \{ a_h , b_i \}$ y $\min \{ a_i , b_j \}$

	min { a_h , b_i }	min { a_i , b_j }
subcaso _1	a_h	a_i
subcaso_2	a_h	b_j
subcaso_3	b_i	a_i
subcaso _4	b_i	b_j

Consideraremos en detalle dos de los casos:

Caso 1 ([CJ], [CJ]):

$$\min \{ a_h , b_i \} < \min \{ b_h , a_i \}$$

$$\min \{ a_i , b_j \} < \min \{ b_i , a_j \}$$

subcaso 11

$a_h \leq b_i$ $\quad$ $a_h < b_h , a_i$

$a_i \leq b_j$ $\quad$ $a_i < b_i , a_j$

por tanto $a_h < b_j , a_j$;

$$\min \{ a_h , b_j \} = a_h < \min \{ b_h , a_j \}$$

subcaso 12

$a_h \leq b_i$ $\quad$ $a_h < b_h , a_i$

$b_j \leq a_i$ $\quad$ $b_j < b_i , a_j$

por tanto, dado que $a_h < b_h$ y $b_j < a_j$

$$\min \{ a_h , b_j \} < \min \{ b_h , a_j \}$$

subcaso 13

$b_i \leq a_h$ $\quad$ $b_i < b_h , a_i$

$a_i \leq b_j$ $\quad$ $a_i < b_i , a_j$

imposible pues b_i no puede ser a la vez mayor y menor que a_i

subcaso 14

$b_i \leq a_h$ $\quad$ $b_i < b_h , a_i$

$b_j \leq a_i$ $\quad$ $b_j < b_i , a_j$

por tanto $b_j < a_h$, b_h

$$\min \{ a_h , b_j \} = b_j < \min \{ b_h , a_j \}$$

<u>Caso 5 ([CA1], [CA1]):</u>

$$\min \{ a_h , b_i \} = \min \{ b_h , a_i \}$$

$$a_h - b_h < a_i - b_i$$

$$\min \{ a_i , b_j \} = \min \{ b_i , a_j \}$$

$$a_i - b_i < a_j - b_j$$

Los cuatro subcasos 51, 52 y 54 son idénticos a los 11, 12 y 14, respectivamente cambiando el signo < por el ≤, por lo que el resultado obtenido es:

$$\min \{ a_h , b_j \} \leq \min \{ b_h , a_j \}$$

complementado, en caso necesario, mediante la relación:

$$a_h - b_h < a_j - b_j$$

y en el subcaso 53:

<u>*subcaso 53*</u>

$b_i \leq a_h$ $\qquad$ $b_i \leq b_h , a_i$

$a_i \leq b_j$ $\qquad$ $a_i \leq b_i , a_j$

de donde $a_i = b_i$; por tanto:

$$a_h - b_h < 0 < a_j - b_j$$

$a_h < b_h$ $\qquad$ $b_j < a_j$

lo que conduce a:

$$\min \{ a_h , b_j \} < \min \{ b_h , a_j \}$$

Para el resto de casos y subcasos la demostración de la transitividad se realiza análogamente.

ANEXO II : DEMOSTRACION DEL RESULTADO DE JOHNSON PARA PIEZAS CONTIGUAS

Sea una permutación cualquiera P y denominemos [k] a la pieza que ocupa la posición k^a de la permutación y analicemos los tiempos muertos en la segunda máquina si se realizan las operaciones en el orden definido por P (en la primera máquina no existirán tiempos muertos, salvo, en todo caso, después de la última operación en la misma). Llamaremos X_k al tiempo muerto en la segunda máquina inmediatamente anterior a la operación en la pieza [k]. Podemos escribir:

$$X_1 = a_{[1]}$$

$$X_2 = \max \{ a_{[1]} + a_{[2]} - X_1 - b_{[1]} , 0 \}$$

y en general:

$$X_k = \max \{ \sum_{s=1}^{k} a_{[s]} - \sum_{s=1}^{k-1} X_s - \sum_{s=1}^{k-1} b_{[s]} , 0 \}$$

De otra forma:

$$X_1 = a_{[1]}$$

$$X_1 + X_2 = \max \{ a_{[1]} + a_{[2]} - b_{[1]} , a_{[1]} \}$$

$$X_1 + X_2 + X_3 = \max \{ a_{[1]} + a_{[2]} + a_{[3]} - b_{[1]} - b_{[2]} , X_1 + X_2 \} =$$

$$= \max \{ a_{[1]} + a_{[2]} + a_{[3]} - b_{[1]} - b_{[2]} , a_{[1]} + a_{[2]} - b_{[1]} , a_{[1]} \}$$

Sea en general:

$$Y_k = \sum_{s=1}^{k} a_{[s]} - \sum_{s=1}^{k-1} b_{[s]}$$

y entonces:

$$\sum_{s=1}^{k} X_s = \max \{ Y_1 , Y_2 , \dots , Y_k \}$$

El valor $F_{max}(P)$, correspondiente a la permutación P, será:

$$F_{max}(P) = \sum_{k=1}^{n} b_{[k]} + \sum_{k=1}^{n} X_k = \sum_{i=1}^{n} b_i + \max \{ Y_1 , Y_2, \dots , Y_n \}$$

y puesto que el primer término es constante, el mínimo de F_{max} se obtendrá minimizando el máximo de Y_k.

Si existen dos piezas contiguas, h e i, en la secuencia, la k^a y la $(k+1)^a$, tales que no satisfacen la condición de Johnson, entonces:

$$\min \{ a_h , b_i \} > \min \{ a_i , b_h \}$$

La relación anterior puede escribirse también:

$$\max \{ -a_h , -b_i \} < \max \{ -a_i , -b_h \}$$

Sumando a los cuatro términos la cantidad:

$$\sum_{s=1}^{k+1} a_{[s]} - \sum_{s=1}^{k-1} b_{[s]}$$

que no depende del orden relativo de las piezas h e i (de cuál ocupa la posición k y cuál ocupa la posición k+1) tenemos:

$$\max \{ Y'_k , Y'_{k+1} \} < \max \{ Y_k , Y_{k+1} \}$$

en donde las Y suponen el orden indicado de las piezas h antes de i, y las Y' el contrario. Puesto que las n-2 Y_k restantes no varían tenemos ventaja utilizando el orden que corresponde a Y' en lugar del que corresponde a Y: la permutación de estas dos piezas no aumenta el valor F_{max} y en algunos casos puede disminuirlo.

Este mismo argumento garantiza que si las dos piezas contiguas h e i son tales que:

$$\min \{ a_h , b_i \} = \min \{ a_i , b_h \}$$

la permutación de h e i no altera el valor de F_{max}

ANEXO III : DEMOSTRACION DE LA CONDICION DE FRENCH

La demostración tiene dos fases:

primera fase

a) Si

$$\min \{ a_1, a_2 ,..., a_n , b_1 ,..., b_n \} = a_h$$

existe (por lo menos) una permutación óptima cuya primera posición la ocupa la pieza h,

b) si

$$\min \{ a_1, a_2 ,..., a_n , b_1 ,..., b_n \} = b_h$$

existe(por lo menos) una permutación óptima cuya última posición la ocupa la pieza h.

Consideremos el caso (a) y una permutación en la que h no ocupa la primera posición, y sea i la pieza que precede inmediatamente a la pieza h. Sean f_1 y f_2 los instantes en que quedan libres las máquinas 1 y 2, respectivamente, antes de procesar la pieza i (si la pieza i fuese la primera $f_1 = f_2 =0$). Llamando c_i y c_h los instantes en que termina la operación en la segunda máquina de las piezas i y h, podemos escribir:

$$c_i = \max \{ f_2 , f_1 + a_i \} + b_i$$

$$c_h = \max \{ c_i , f_1 + a_i + a_h \} + b_h$$

Si las piezas i y h se invirtieran de orden dichos valores serían

$$c_h' = \max \{ f_2 , f_1 + a_h \} + b_h$$

$$c_i' = \max \{ c_h , f_1 + a_h + a_i \} + b_i$$

Tenemos $c_i < c_h$ y $c_h' < c_i'$ (salvo que $b_i = 0$ en que podríamos tener $c_h' = c_i'$, lo que no influye en lo que sigue). Vamos a demostrar que $c_h \geq c_i'$

$$c_h = \max \{ f_2 + b_i , f_1 + a_i + b_i , f_1 + a_i + a_h \} + b_h$$

pero por hipótesis $a_h \leq b_h , a_i , b_i$, por tanto $a_i + a_h \leq a_i + b_i$

$$c_h = \max \{ f_2 + b_i , f_1 + a_i + b_i \} + b_h = \max \{ f_2, f_1 + a_i \} + b_h + b_i$$

Análogamente:

$$c_i' = \max \{ f_2 , f_1 + a_h , f_1 + a_h + a_i - b_h \} + b_h + b_i$$

pero

$$f_l + a_h \leq f_l + a_i$$

$$f_l + a_h + a_i - b_h \leq f_l + a_i$$

por tanto $c_i' \leq c_h$.

Como $F_{max} = \max\{ c_j \}$, al permutar las piezas i y h no empeora el valor de F_{max} y en algunos casos puede mejorar (si la permutación de partida no fuese óptima). Por tanto mediante inversiones sucesivas de piezas contiguas a partir de una permutación óptima podemos situar la pieza h en primera posición sin empeorar el valor F_{max}, lo que demuestra que existe una permutación óptima (por lo menos) en la que la pieza h está en primera posición.

Análogamente demostraríamos el resultado (b).

segunda fase

Consideremos un procedimiento de construcción de una permutación óptima basado en la condición de French (algoritmo 1 del apartado 5). Después de k pasos hemos colocado las k_1 piezas iniciales de la permutación y las k_2 finales ($k_1 + k_2 = k$). Sea E el conjunto de las n-k piezas no situadas. Debemos demostrar que:

a) si $\min_{i \in E} \{ a_i , b_i \} = a_h$ existe una permutación óptima cuya posición $k_1 + 1$ la ocupa la pieza h,

b) si $\min_{i \in E} \{ a_i , b_i \} = b_h$ existe una permutación óptima cuya posición $k_2 - 1$ la ocupa la pieza h.

La demostración puede basarse en el mismo procedimiento de intercambio utilizado en la fase 1, teniendo en cuenta que en el mismo solo quedan afectadas las piezas del subconjunto E y no las restantes ya situadas.

La aplicación sistemática del procedimiento conduce a una permutación que cumple la condición de French, sin empeorar el valor F_{max}.

Queda demostrado que dada una permutación óptima se puede transformar en una permutación que cumple [CF] también óptima. Para completar la demostración faltaría mostrar que todas las permutaciones que cumplen [CF] tienen el mismo valor de F_{max} y son por tanto óptimas, lo cual es sencillo ya que si existen varias pueden transformarse unas en otras utilizando el procedimiento de intercambio descrito sin empeorar el valor F_{max}.

ANEXO IV : COTA DE NABESHIMA

Dado un problema $n/m/P/F_{max}$ podemos establecer una cota de F_{max} mediante la siguiente expresión:

$$K_{j,k} = MIN_{h \neq i} \{ P_{1,j}(h) + P_{k,m}(i) \} + S_{j,k}$$

donde $1 \leq j \leq k \leq m$

Denominando $p_{j,i}$ la duración de la operación en la máquina j de la pieza i, los valores anteriores tienen el siguiente desarrollo:

$$P_{1,j}(h) = \sum_{l=1}^{j-1} p_{l,h} \qquad \text{si } 1 < j \qquad P_{1,1}(h) = 0$$

$$P_{k,m}(i) = \sum_{l=k+1}^{m} p_{l,h} \qquad \text{si } k < m \qquad P_{m,m}(i) = 0$$

y $S_{j,k}$ idealmente es el mínimo del valor F_{max} del ejemplar del subproblema $n/k-j+1/P/F_{max}$ deducido del original considerando unicamente las k-j+1 máquinas comprendidas entre la j y la k. En muchos casos $S_{j,k}$ será una una cota inferior de F_{max}.

Si k = j el subproblema consta de una única máquina y:

$$S_{j,j} = \sum_{i=1}^{n} p_{j,i}$$

coincidiendo la tradicional cota longitudinal.

Si k = j + 1 el subproblema es un problema $n/2/P/F_{max}$ y el valor $S_{j,j+1}$ puede obtenerse ordenando las piezas siguiendo el esquema de Johnson.

Si k > j + 1 el subproblema es un problema del mismo tipo que el original, de menor dimensión pero de similar complejidad. Por consiguiendo el cálculo del mínimo de F_{max} puede exigir demasiados recursos. Relajando la condición de que en las máquinas comprendidas entre la j y la k las operaciones no deben solaparse (pero manteniendo que en ambas máquinas el orden debe ser el mismo) obtenemos un problema $n/2/P/F_{max}$ con tiempos de latencia entre las dos operaciones de una misma pieza q_i:

$$q_i = \sum_{l=j+1}^{k-1} p_{l,i}$$

cuyo mínimo de F_{max} puede obtenerse ordenando las piezas aplicando el esquema de Johnson a $p_{j,i} + q_i$ y a $p_{k,i} + q_i$ (ver ANEXO V). Por tanto es calculable un valor eficiente de $S_{j,k}$. Puede observarse que las cotas transversales tradicionales son una simplificación de este caso.

La idea de estas cotas se debe a Nabeshima y han sido extendidas por Potts.

Este esquema de cotas es generalizable a la situación que se presenta al aplicar el algoritmo de Lomnicki, cuando la parte inicial de la permutación ha sido ya definida. Para un vértice determinado sea **I** el subconjunto de piezas que componen la parte inicial de la permutación, σ dicha permutación, $\mathbf{E} = \{1,2,..,n\} - \mathbf{I}$ y $f_j(\sigma)$ el instante en que queda liberada la máquina j realizando las piezas de I en el orden definido por σ sin tiempos muertos innecesarios (partiendo de un instante inicial $f_j(0)=0$, $j=1,2,.., m$). La adaptación de la cota conduce a:

$$K_{j,k}(\sigma) = f_j(\sigma) + MIN_{i\in \mathbf{E}} \{ P_{k,m}(i) \} + S_{j,k}(\mathbf{E})$$

Donde $S_{j,k}(\mathbf{E})$, analogamente al caso anterior, es una cota del valor F_{max} del subproblema $|\mathbf{E}|/k\text{-}j+1/P/F_{max}$ deducido del original considerando unicamente las k-j+1 máquinas comprendidas entre la j y la k y las piezas del subconjunto **E** (puede afinarse más teniendo en cuenta $f_k(\sigma)$).

En el algoritmo Lomnicki Pendular, en un instante dado, las primeras piezas posibles del problema inverso forman un subconjunto **F'**, y estas piezas son las únicas susceptibles a ser las últimas en el problema directo; denominemos $\mathbf{L} = \mathbf{F'}\cap\mathbf{E}$. Si **L** es vacío, puede asignarse al vértice una cota superior al mejor valor hallado (y por tanto eliminarlo), en caso contrario la cota puede escribirse:

$$K_{j,k}(\sigma) = f_j(\sigma) + MIN_{i\in \mathbf{L}} \{ P_{k,m}(i) \} + S_{j,k}(\mathbf{E})$$

Por otra parte de las relaciones inverso-directo puede deducirse que ciertas piezas no pueden ocupar todas las posiciones de la permutación sino sólo una franja, más o menos reducida, de la misma, lo cual puede tenerse en cuenta en el cálculo de $S_{j,k}(\mathbf{E})$, obligando a que dichas piezas ocupen dicha franja como criterio dominante, al que se subordinará el esquema de Johnson.

ANEXO V : DEMOSTRACION DE LA CONDICION DE JOHNSON PARA EL CASO DOS MAQUINAS CON TIEMPO DE LATENCIA INTERMEDIO

Sean a_i y b_i los tiempos de proceso de la pieza i en dos máquinas y q_i el tiempo mínimo que debe transcurrir entre la terminación de la primera operación y el comienzo de la segunda (puede interpretarse como el tiempo de transporte). Las operaciones en las máquinas no pueden solaparse pero sí los tiempos de latencia q_i. Consideramos n piezas (i = 1,2,..,n) disponibles lo mismo que las máquinas en el instante 0 y deseamos encontrar unas secuencias de las piezas en la primera y segunda máquina que minimicen F_{max}. En este tipo de problema la estricta minimización de F_{max} puede conducir a permutaciones de las n piezas distintas en ambas máquinas, como puede apreciarse en el siguiente ejemplo:

$n = 2;\ a_1 = 4;\ b_1 = 2;\ q_1 = 12;\ a_2 = 3;\ b_2 = 5;\ q_2 = 2$

secuencia M1	1	2	2	1	1	2	2	1
$f_{1,i}$	4	7	3	7	4	7	3	7
secuencia M2	1	2	2	1	2	1	1	2
$r_{2,i}$	16	9	9	19	9	16	19	9
$f_{2,i}$	18	23	14	21	14	18	24	26

donde $r_{2,i}$ indica el instante en que la pieza i está disponible para la segunda máquina y $f_{j,i}$ el instante en que la máquina j termina su operación sobre la pieza i. El valor mínimo de F_{max} es 18 y corresponde a permutaciones distintas en la primera y segunda máquinas.

Limitaremos nuestra consideración a los casos en que las secuencias en ambas máquinas son idénticas. Llamando [k] a la pieza que ocupa la posición k de la pemutación y X_k al tiempo muerto en la segunda máquina (el de la primera puede hacerse nulo) antes de la operación en la pieza [k] tenemos:

$$X_1 = a_{[1]} + q_{[1]}$$

$$X_2 = MAX\ \{\ 0\ ,\ a_{[1]} + a_{[2]} + q_{[2]} - (X_1 + b_{[1]})\ \}$$

$$X_3 = MAX\ \{\ 0\ ,\ a_{[1]} + a_{[2]} + a_{[3]} + q_{[3]} - (X_1 + b_{[1]} + X_2 + b_{[2]})\ \}$$

etc.

de donde:

$$Y_1 = X_1 = a_{[1]} + q_{[1]}$$

$$Y_2 = X_1 + X_2 = MAX\ \{\ a_{[1]} + q_{[1]}\ ,\ a_{[1]} + a_{[2]} + q_{[2]} - b_{[1]}\ \}$$

$$Y_3 = X_1 + X_2 + X_3 = MAX\ \{\ a_{[1]} + q_{[1]}\ ,\ a_{[1]} + a_{[2]} + q_{[2]} - b_{[1]}\ ,\ a_{[1]} + a_{[2]} + a_{[3]} + q_{[3]} - b_{[1]} - b_{[2]}\ \}$$

es decir las mismas expresiones que aparecen en la clásica demostración del teorema de Johnson (Anexo II) para el caso $n/2/F/F_{max}$ sin tiempos de latencia salvo la substitución de:

a_i por $a_i + q_i$

b_i por $b_i + q_i$

Por consiguiente, reiterando el razonamiento, podemos inferiri que se obtendrá una permutación óptima si a todo par de piezas h e i tales que:

$$\min \{ a_h + q_h , b_i + q_i \} < \min \{ b_h + q_h , a_i + q_i \}$$

la pieza h está situada antes que la pieza i.

Un razonamiento análogo puede utilizarse para justificar el caso especial del problema $n/3/F/F_{max}$ en el que:

$$\min_i p_{1,i} \geq \max_i p_{2,i} \quad \text{o/y} \quad \min_i p_{3,i} \geq \max_i p_{2,i}$$

obteniendose una permutación óptima si a todo par de piezas h e i tales que:

$$\min \{ p_{1,h} + p_{2,h} , p_{2,i} + p_{3,i} \} < \min \{ p_{2,h} + p_{3,h} , p_{1,i} + p_{2,i} \}$$

la pieza h está situada antes que la pieza i. Este resultado se encuentra ya en el artículo seminal de Johnson.