

Una revisión de las decisiones tomadas en los heurísticos para RCPSP

ZUPIRIA GOROSTIDI, L.M., CASTRO IÑIGO, B y MARTINEZ ARNAIZ, J.A.

Universidad del País Vasco/Euskal Herriko Unibertsitatea

Resumen

Muchos de los heurísticos empleados para resolver problemas de secuenciación de proyectos con restricción de recursos no tienen en consideración las futuras consecuencias de las decisiones tomadas en una etapa. Así, en ocasiones, debido a las condiciones específicas del momento, se viola el orden de prioridad establecido por la regla seleccionada. En este trabajo se incorporan operaciones que permiten resolver situaciones creadas por equivocaciones cometidas anteriormente: la desprogramación de actividades menos prioritarias para dar paso a las que más urge secuenciar y la operación de desplazamiento que permite aprovechar recursos ociosos tras la desprogramación.

1. Introducción

Los procedimientos empleados para la obtención de la duración mínima de un proyecto han sido numerosos desde que se realizaran los primeros trabajos CPM y PERT. Es más, la consideración de la disponibilidad de los recursos empleados ha provocado la aparición y desarrollo de nuevas técnicas que permiten aproximarse a la solución, dado que el tratamiento exacto de los mismos no es operativo para el caso de proyectos de tamaño real. Precisamente, la consideración NP-hard de los problemas de secuenciación de proyectos con recursos escasos (RCPSP), sugiere nuevos tratamientos exactos que proporcionen soluciones óptimas de la forma más eficiente posible. Junto a estos y debido al tamaño del problema en cuestión se han propuesto distintas aproximaciones que, aunque no aseguran la obtención de la solución óptima, ofrecen soluciones operativas con esfuerzo computacional mucho mas razonable. Respecto a las técnicas exactas señalar que junto a las aproximaciones que utilizan programación entera 0-1, se han construido algoritmos de enumeración implícita basados en técnicas de ramificación y acotación que agilizan el proceso de obtención de secuenciaciones de duración óptima. En ellas se establecen nuevos criterios para la construcción del árbol sobre el que trabajar hasta obtener secuencia completa óptima, así como proporcionar reglas de dominancia para la poda de sus ramas. Hoy en día entre las más extendidas y aprobadas se encuentra la de Demeulemeester y Herroelen (1992).

Sin embargo, incluso el más destacado entre ellos no es operativo en proyectos de gran tamaño, por lo que resulta de gran utilidad el desarrollo de heurísticos que faciliten soluciones factibles aceptables. Los basados en reglas de prioridad han ocupado y ocupan un lugar relevante. Estos se componen de un esquema de secuenciación, en Serie (Kelley,

1963) o en Paralelo (Kelley, 1963; Bedworth y Bailey,1982) y una regla de prioridad con la que se ordena el conjunto de actividades a secuenciar, conocidas con el nombre de actividades elegibles en cada etapa del procedimiento. Esta última se puede clasificar, entre otros criterios, según la característica que la define en regla de actividad, de red y de recursos y según su actualización en cada etapa en Estático (no cambia) y Dinámico (se revisa).

Con el esquema en paralelo se construye una secuencia factible añadiendo en cada etapa nuevas actividades a las ya secuenciadas en etapas anteriores, respetando las relaciones de precedencia definidas, así como la disponibilidad de recursos y atendiendo el orden establecido por la regla de prioridad seleccionada. Se han realizado diferentes estudios comparativos de las distintas reglas (Patterson(1973,1976); Davis y Patterson(1975); Alvarez-Valdés y Tamarit(1996); Boctor(1990); Kolisch(1996)), sin que se pueda concluir que alguna de ellas proporcione siempre los mejores resultados.

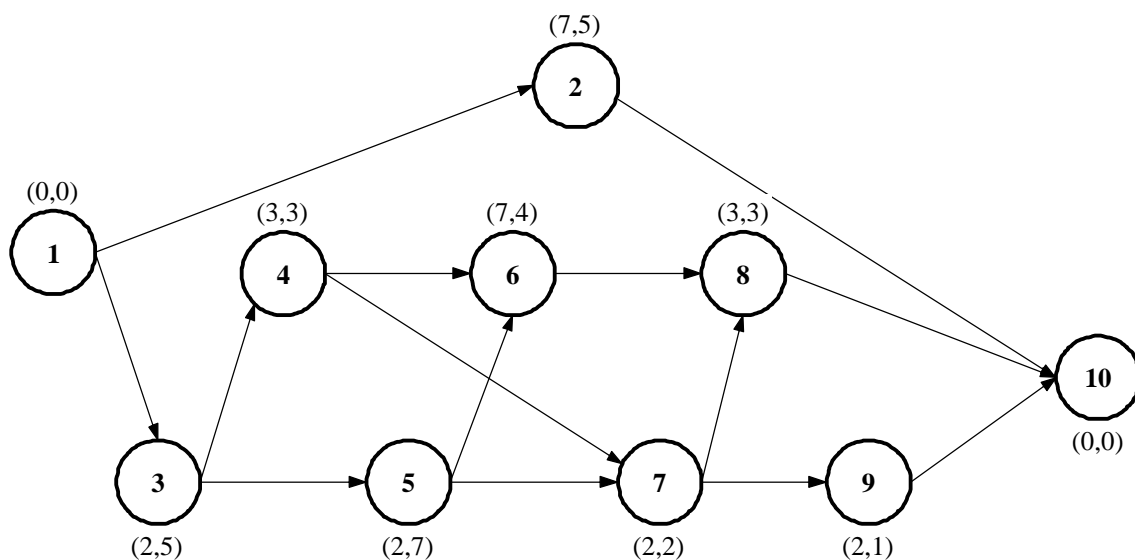
El heurístico aquí presentado se enmarca dentro de aquellos multirecurso que con un esquema en paralelo y una regla de prioridad dinámica, sin permitir la interrupción de las actividades del proyecto, trata de obtener la duración mínima del proyecto, considerando constante la duración de las actividades. Además, trata de corregir en cada etapa posibles "equivocaciones" cometidas en etapas anteriores en las que, debido a las condiciones específicas del momento, normalmente referidas a la disponibilidad de los distintos recursos, se ha violado el orden de prioridad establecido por la regla seleccionada.

En este trabajo se incorporan las operaciones de desprogramación y desplazamiento, con las que se pretende revisar las decisiones tomadas en etapas anteriores. Además, se trata de evaluar y obtener las condiciones bajo las que dichas consideraciones nos proporcionen secuenciaciones que dominen a las resultantes de obviarlas, mediante la comparación de sus

secuencias parciales. Por último se ha desarrollado un estudio computacional que permite evaluar la bonanza del método propuesto.

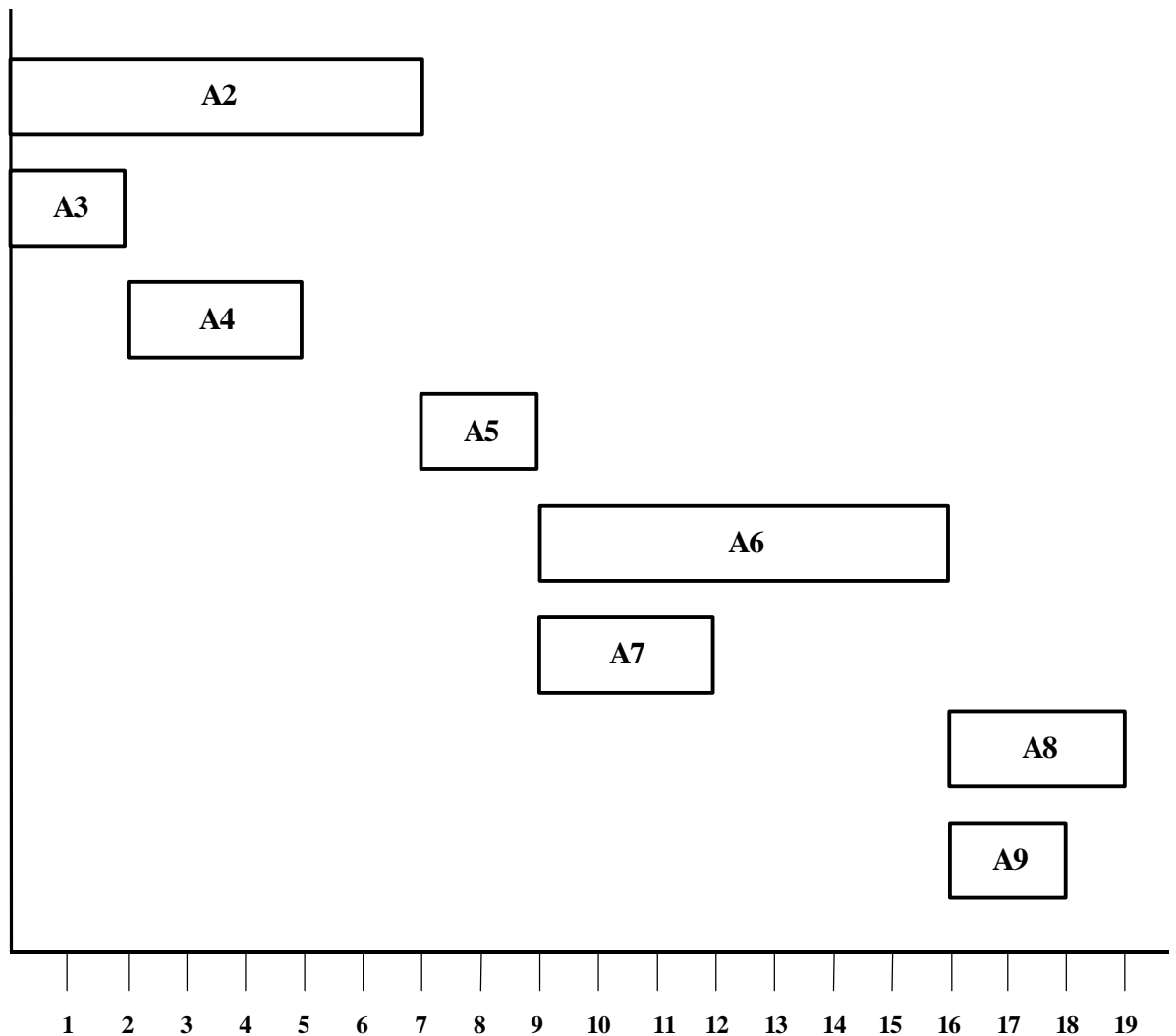
2. La operación de desprogramación.

Tal y como indicábamos anteriormente, y bajo la consideración del criterio MINSLK como regla de prioridad implementada en un heurístico en paralelo, es posible que en una etapa determinada se inicie una actividad, impidiendo la realización, en etapas posteriores, de otra actividad crítica. Wiest (1967) desarrolló un heurístico en el que se revisaban las decisiones adoptadas en etapas anteriores. Si en una etapa en la que no se podía secuenciar una actividad crítica, estaba en progreso otra actividad no crítica, una vez actualiza su holgura, que consumía mayor o igual cantidad de recursos, se desprogramaba la actividad en progreso para dar paso a la crítica. Para explicar dicha operación consideramos el proyecto



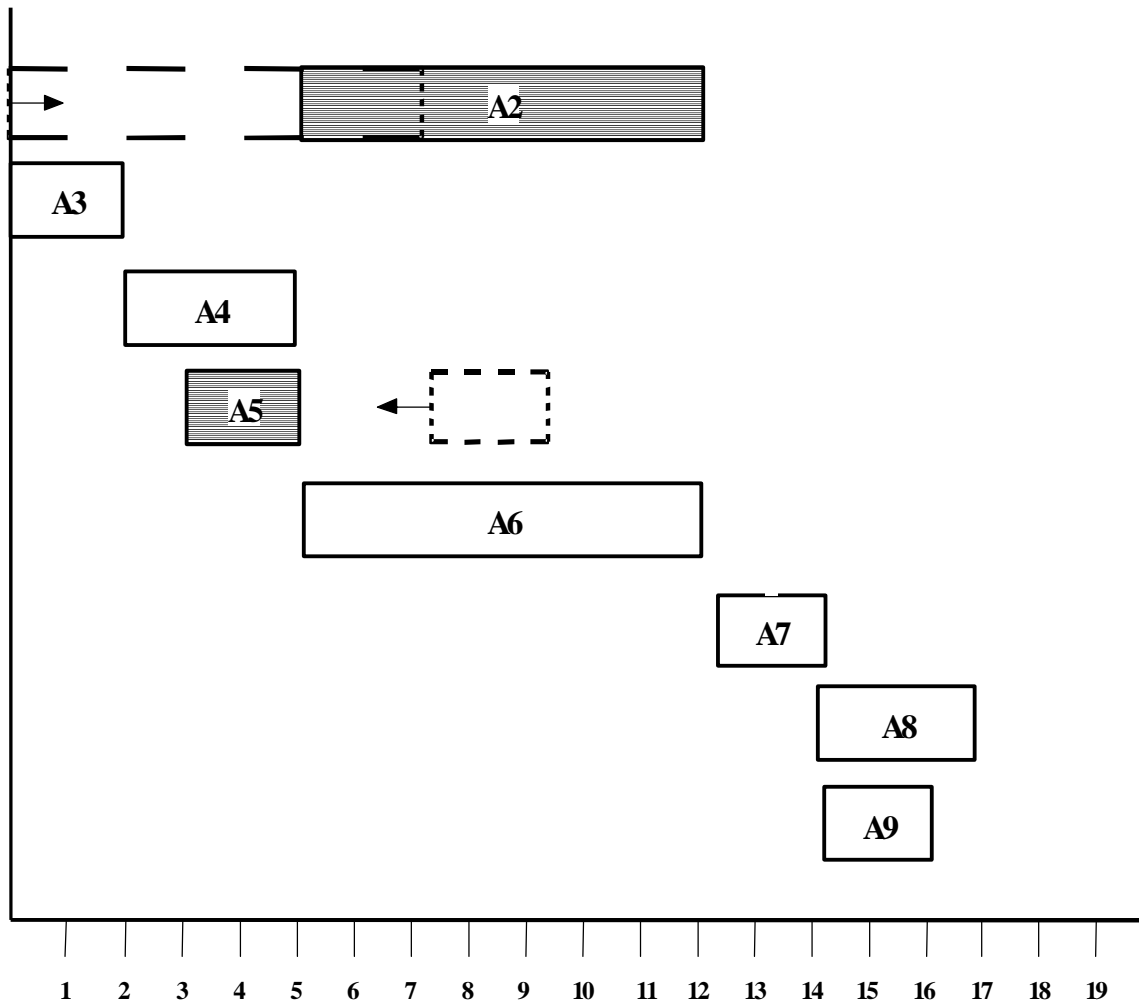
representado mediante la siguiente red, donde las actividades se recogen en los nodos, los números que los acompañan indican la duración en días el primero y el consumo del único recurso considerado, siendo la disponibilidad del mismo de 10 unidades al día.

La secuenciación del proyecto obtenida de aplicar el heurístico en paralelo con la regla de prioridad MINSLK y sin considerar ninguna revisión es la siguiente:



Incorporemos ahora la posibilidad de deprogramar una actividad para dar paso a una crítica. Resulta que el día 3 la actividad A5 es crítica y sin embargo, no puede arrancar por no disponer de recursos suficientes. Investigando la situación de las actividades en progreso, concluimos que la actividad A2, que se estaba realizando desde el inicio, no es crítica, si actualizamos su holgura al día 3 y además, libera los recursos suficientes para dar paso a la crítica A5. Con la desprogramación propuesta obtenemos una secuenciación que con una

duración del proyecto de 17 días frente a la anterior propuesta de 19 días, tal y como se recoge a continuación.

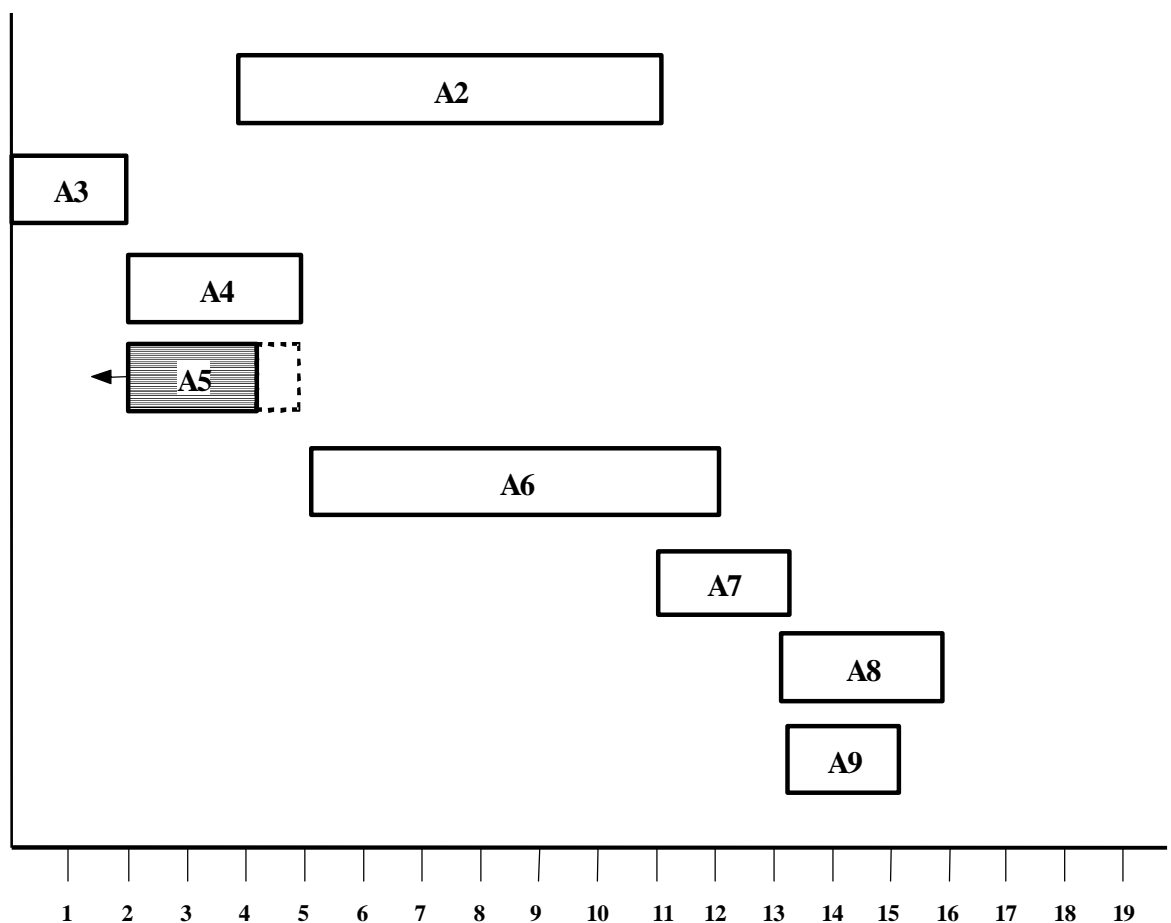


Como se puede observar se ha secuenciado la actividad A5 el día 3, desprogramando para ello la A2. De esta forma se ha conseguido una secuenciación de menor duración.

3. La operación de desplazamiento.

En el heurístico desarrollado, además de aplicar la desprogramación descrita incorporamos la operación de desplazamiento que resulta de la intención de aprovechar al

máximo la operación de desprogramación, así como la situación generada tras su aplicación. Una vez desprogramada la actividad y secuenciada la crítica estudiamos si es posible ocupar el hueco dejado por la actividad desprogramada. En nuestro ejemplo, una vez realizada la operación de desprogramación, estudiamos si alguna actividad puede iniciarse antes del día 3 y así aprovechar los recursos liberados desde el día 0. El resultado de este análisis indica que, en efecto, la actividad A5 puede iniciarse el día 2 adelantando su inicio un día sobre el proporcionado por el esquema resultante de llevar a cabo la desprogramación. La secuenciación obtenida para el proyecto considerando las operaciones de desprogramación y de desplazamiento es de 17 días, tal y como se representa a continuación:



4. Un estudio sobre la conveniencia de dichas operaciones.

Sin embargo, aunque a priori pueda parecer lo contrario, tanto la estrategia de desprogramación como la de desplazamiento un instante de tiempo pueden proporcionar secuenciaciones de mayor duración que aquella que obtendríamos de no considerarlas, por lo que resulta de gran interés el estudio y la obtención de condiciones bajo las cuales se garantice el éxito de las mismas.

Antes de continuar establezcamos la notación necesaria para nuestra exposición. Consideremos un proyecto multirecurso donde A representa el conjunto de actividades del proyecto ($A=\{1,\dots,n\}$).

Sea K el número de recursos que se utilizan en el proyecto. Se supone que la disponibilidad del recurso k es conocida, se mantiene constante a lo largo de la duración del proyecto y se denota por b_k .

Para cada actividad i del proyecto se denota por:

d_i = duración de la actividad i

f_i = tiempo de finalización de la actividad i

r_{ik} = cantidad de recurso k requerida por la actividad i

h_{it} = Holgura total de la actividad i en t .

Así mismo, se definen en t los siguientes conjuntos de actividades:

PS_t = secuencia parcial en t . Actividades acabadas y en progreso en t .

S_t = actividades en progreso en t .

E_t = actividades elegibles en t .

Se denota por $AI(i) = \{j \in A \mid \exists k \text{ con } r_{ik} + r_{jk} > b_k\}$

Definición: Se denomina *corte en t* de un esquema parcial PS_t al conjunto de las actividades sin secuenciar que tienen todos sus predecesores en PS_t

Se considera un instante t en la aplicación del heurístico a un proyecto en el que se realiza una desprogramación para incluir en el esquema la actividad crítica k . Supongamos que d es la actividad que libera los recursos suficientes para dar paso a la actividad k . De acuerdo con el heurístico la estrategia a seguir sería:

- Desequenciar del esquema parcial en t la actividad d , incluir en dicho esquema la actividad k y completar el esquema parcial para ese instante.

Sin embargo, imaginemos que no llevamos a cabo la desprogramación. En tal caso la decisión sería.

- Completar el esquema parcial para dicho instante sin considerar la desprogramación, es decir, aplicando el heurístico inicial que no considera tal operación.

Se denotará por PS_t^D el esquema resultante de aplicar la primera estrategia y por PS_t^{SD} el resultante de la segunda. A su vez por s_i^D y f_i^D los tiempos de inicio y final de la actividad i en el esquema con desprogramación y por s_i^{SD} y f_i^{SD} los mismos tiempos en el esquema sin desprogramación.

Si se ordenan las actividades en progreso del esquema PS_t^D en orden creciente de sus tiempos de finalización: $\{i_1, i_2, \dots, i_n\}$, se denotará por $i^* = i_s$, la primera actividad que

cumple: $\sum_{m=s+1}^n r_{i_mv} + r_{dv} \leq b_k$, $v = \{1, \dots, K\}$ y $\sum_{m=s}^n r_{i_mv} + r_{dv} > b_k$ para algún v

Teorema 1. Dados los esquemas PS_t^D y PS_t^{SD} si se cumple que el corte del esquema PS_t^D contiene únicamente la actividad d y el corte de PS_t^{SD} contiene únicamente la actividad k , la duración del proyecto que se obtiene siguiendo de forma óptima la secuencia parcial PS_t^D es al menos la que resulta de seguir de forma óptima PS_t^{SD} , si se cumple alguna de las condiciones:

1. $k \in AI(d)$
2. $k \in A \setminus AI(d)$ y $s_d^{SD} + d_k \leq f_{i^*}^D$

Centrémonos ahora en la situación en la que en un instante t se realiza la desprogramación de la actividad d para incluir en el esquema la actividad crítica k . Sea $L_t = \{l \in E_{t-1} / l \text{ no está aún secuenciada}\}$. En el caso en el que exista una actividad $l \in L_t$ para la que al desprogramar d en t resten recursos suficientes para secuenciarla, siguiendo el orden de prioridad en t , se puede optar entre las dos siguientes alternativas:

- Fijar l en su comienzo más temprano respetando las relaciones de precedencia y restricciones de recursos. En el caso en el que $f_l > t$ se completa el esquema parcial para el instante t mientras que en otro caso, se completará el esquema parcial a partir de f_l , incluyendo los cambios que esta decisión conlleve.

- No situar l y completar el esquema parcial en el instante t .

Se denotará por PS_t^L el esquema resultante de aplicar la primera estrategia y por PS_t^{SL} el resultante de la segunda. A su vez por s_i^L y f_i^L los tiempos de inicio y final de la actividad i en el esquema con desplazamiento y por s_i^{SL} y f_i^{SL} los mismos tiempos en el esquema sin desplazamiento.

Teorema 2. Si $f_l^L = t' \leq t$, entonces la duración del proyecto que se obtiene siguiendo de forma óptima la secuencia parcial PS_t^{SL} es al menos la que resulta de seguir de forma óptima PS_t^L .

Teorema 3. Si $f_l^L > t$ y los esquemas PS_t^{SL} y PS_t^L difieren únicamente en la actividad l entonces si se cumple alguna de las condiciones que siguen, la duración del proyecto que se obtiene siguiendo de forma óptima la secuencia parcial PS_t^{SL} es al menos la que resulta de seguir de forma óptima PS_t^L :

1. $f_l^{SL} - d_l = t$
2. $f_l^L \leq \min\{f_i^{SL} - d_i \leq t < f_i^{SL}\}$

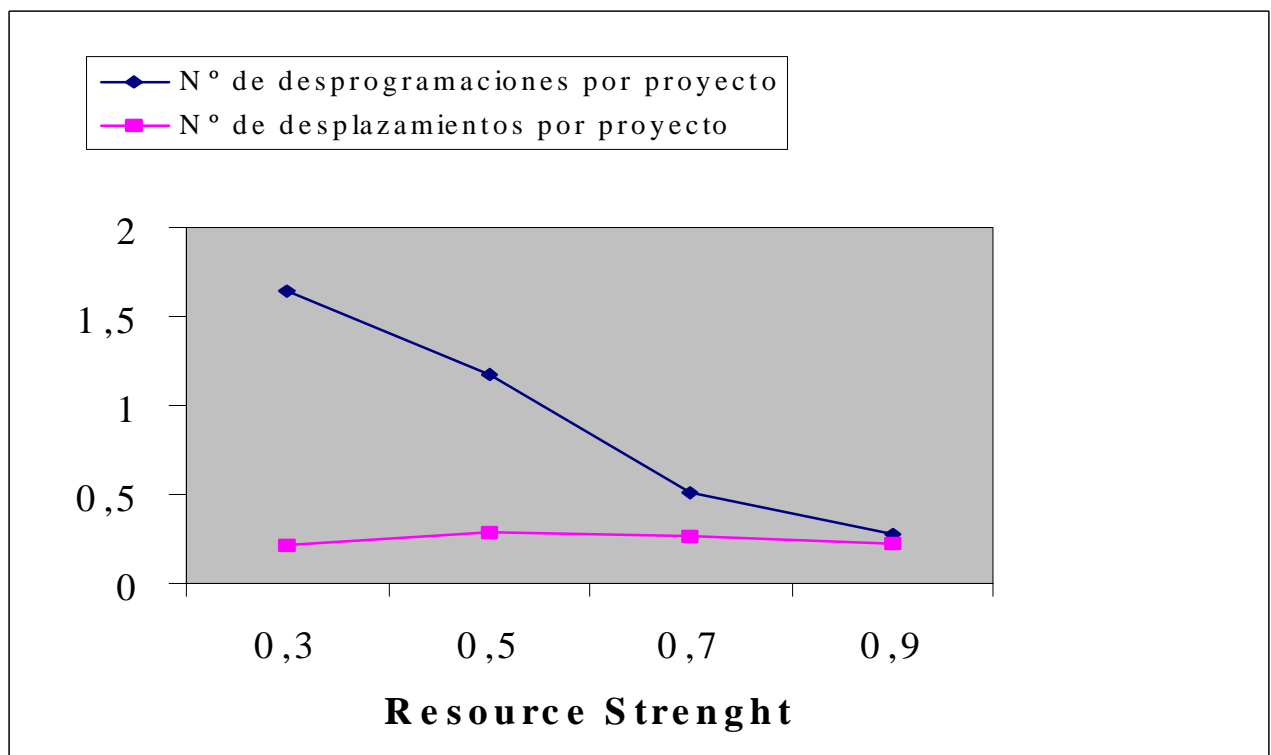
Corolario. Si $k=l$ y $f_l^L > t$ entonces la duración del proyecto que se obtiene siguiendo de forma óptima la secuencia parcial PS_t^{SL} es al menos la que resulta de seguir de forma óptima PS_t^L .

5. Resultados computacionales.

Se ha implementado un algoritmo en paralelo multirecurso single mode sin interrupción de las actividades que incluye los resultados teóricos anteriormente presentados. Posteriormente y para verificar la eficiencia del mismo, se han simulado con el programa PROGEN 100 ejemplos de cada una de las combinaciones de parámetros que

caracterizan los proyectos generados con dicho simulador: resource strenght: 0.9, 0.7, 0.5, 0.3; resource factor: 0.5, 0.8, 1; complejidad: 1.5, 2.5.

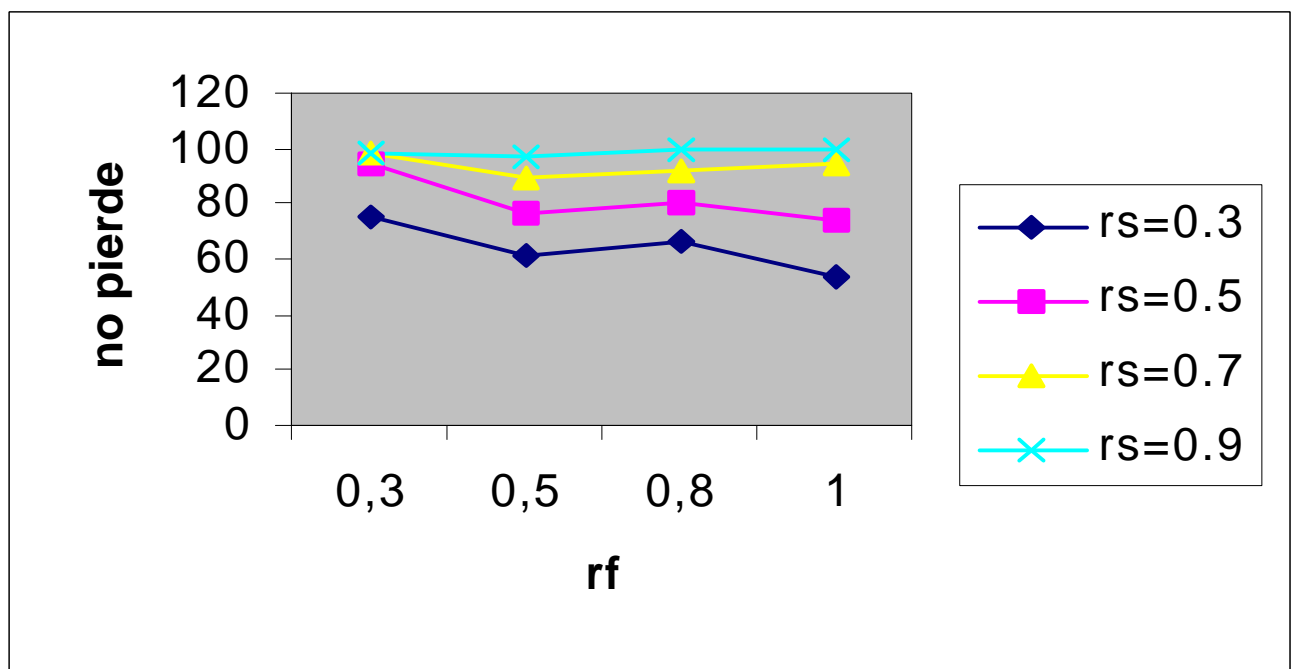
Se ha observado que el único parámetro que tiene relación con las operaciones de desprogramación y desplazamiento es el resource strenght, existiendo una relación inversa entre el número desprogramaciones y, por tanto de desplazamientos, que se producen en la ejecución del algoritmo. Se han realizado ajustes Chi cuadrado que no rechazan la hipótesis de independencia respecto al resto de parámetros. Las submuestras con resource factor 0.3 muestran un comportamiento menos homogéneo que el resto y, por ello, se han agregado todas las submuestras con el mismo resource strenght y resource factor diferente de 0.3. En el siguiente gráfico se muestra el número medio de desprogramaciones por proyecto, de las simulaciones con complejidad 1.5. Así mismo, se incluye otra serie que recoge el número medio de desplazamientos referido a la situación en que se pueden dar, es decir, por desprogramación para los mismos proyectos.



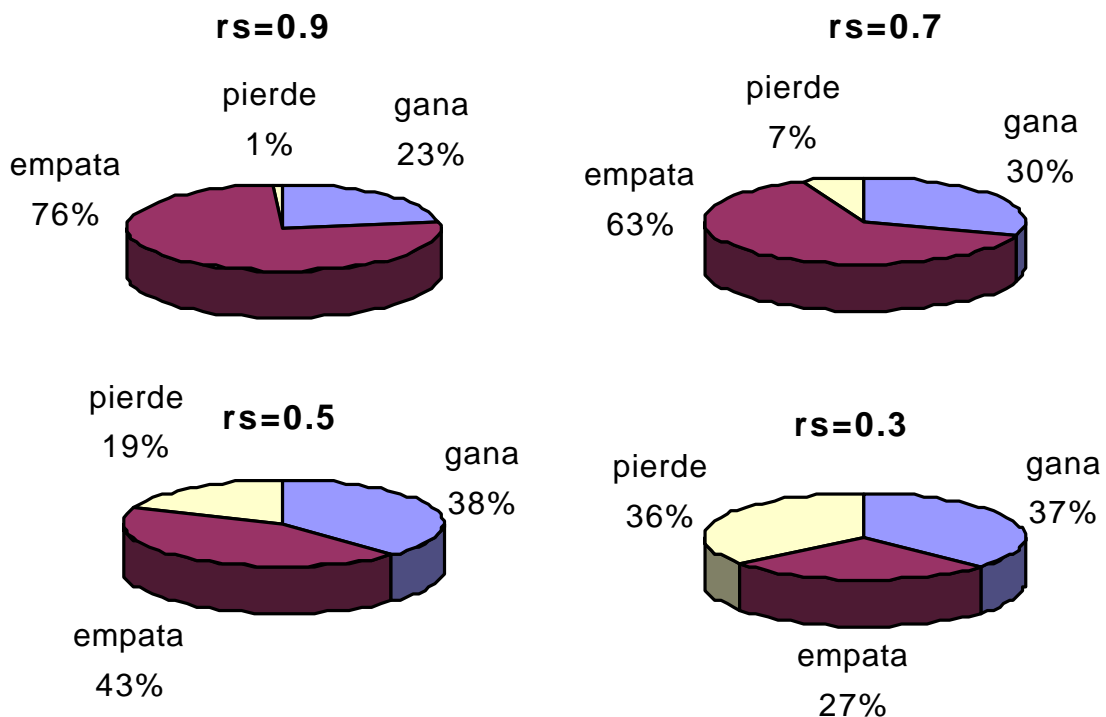
Como se puede observar, el número medio de desprogramaciones por proyecto aumenta a medida que se intensifica la disputa por los recursos, es decir, a medida que disminuye resource strenght. Sin embargo, el número de desplazamientos por desprogramación no varía con este parámetro, así que las condiciones de aplicación de la operación no se ven afectadas con el nivel de resource strenght del proyecto.

Centrándonos en los resultados obtenidos con el algoritmo programado y atendiendo a la duración obtenida del proyecto señalar que se han comparado con los obtenidos con el criterio MINSLK para los 1600 proyectos de complejidad 1.5.

En el siguiente gráfico se muestra el porcentaje de proyectos en los que el heurístico programado proporciona una duración del proyecto inferior o igual a la del otro considerado, para un resource strenght fijo.



Los siguientes diagramas de sectores presentan para las mismas submuestras, las frecuencias de pérdida, empate y ganancia del heurístico respecto al MINSLK.



Por último, se ha comparado el porcentaje de desviación al óptimo del heurístico para algunas submuestras, resultando el más destacado el caso de resource strenght 0.9. Para éste mismo valor de resource strenght se ha obtenido la frecuencia con que se consigue la duración óptima del proyecto. En la siguiente tabla se recoge dicha información, incluyendo además el número de veces que el heurístico MINSLK obtiene el óptimo.

	rf=0.3	rf=0.5	rf=0.8	rf=1
Desviación respecto al óptimo	0.003995	0.005927	0.001261	0.0004167
% óptimo MINSLK	71	65	78	82
% óptimo heurístico	92	92	96	99

Bibliografia

- Demeulemeester, E. and Herroelen, W. (1992). A branch-and-bound procedure for the multiple resource-constrained project scheduling problem. *Management Science*, 38, pp. 1803-1818.
- Patterson, J. H. (1973). Alternate methods of project scheduling with limited resources. *Naval Research Logistics Quarterly*, 20, pp. 767-784.
- Patterson, J. H. (1976). Project scheduling: the effects of problem structure on heuristic performance. *Naval Research Logistics Quarterly*, 23, pp. 95-123
- Kelley, J. E. (1963) The Critical-path Method: Resources Planning and Scheduling. Chapter 21 in J. F. Muth and G. L. (Eds). Thompson, Industrial Scheduling, Prentice-Hall, inc., Englewood Cliffs, New Jersey.
- Bedworth, D. D. and Bailey, J. E. (1982) Integrated Production Control Systems-Management, analysis, design, J. Wiley and Sons, New York.
- Davis, E. W. And Patterson, J. H. (1975) A comparison of heuristic and optimum solutions in resource-constrained project scheduling, *Management Science*, 21, pp. 944-955.
- Alvarez-Valdés, R. and Tamarit, J. M. (1989) Heuristic algorithms for resource-constrained project scheduling: A review and an empirical analysis, in Slowinski, R. and Weglarz, J (Eds), *Advances in project scheduling*, Elsevier, Amsterdam, pp. 113-134.
- Boctor, F. F. (1990) Some efficient multi-heuristic procedures for resource-constrained project scheduling, *European Journal of Operational Research*, 49, pp. 3-13
- Kolisch, R. (1996) Efficient priority rules for the resource-constrained project scheduling problem, *Journal of Operational Management*, 14, pp. 179-192.
- Wiest, J. D. (1967) A heuristic model for scheduling large projects with limited resources. *Management Science*, 13, pp. B359-B377