

# Secuenciación heurística de un proyecto con restricciones temporales

Enric Crespo i Escobar. Universitat de València(\*)

M<sup>a</sup> Fulgencia Villa Juliá. Florida Universitaria(\*\*)

## RESUMEN

El presente trabajo aborda el análisis de un problema real del mantenimiento que una compañía aeronáutica debe practicar sobre una flota de aviones del tipo Boeing-747. En primer lugar analizamos en qué consiste dicho mantenimiento, los elementos que lo definen y los objetivos que se persiguen. A continuación estudiamos sus similitudes y diferencias con el RCPS (*Resource Constrained Project Scheduling*), un modelo clásico aplicado a este tipo de problemas. Entre estas diferencias se encuentran la existencia de restricciones temporales para la secuenciación de determinadas tareas así como otros tipos especiales de relaciones entre las actividades además de las clásicas de precedencia. Concretamos, finalmente, la metodología a aplicar para abordar el problema planteado.

**Palabras Clave:** Dirección de Operaciones, Secuenciación, Heurísticos, Secuenciación con recursos limitados.

**Clasificación A.M.S.:** 90B35, 90B36, 90C27.

**Clasificación J.E.L.:** L93

---

(\*) Dep. de Matemàtica Economicoempresarial. Edificio Departamental Oriental. Avda. dels Tarongers,s/n 46022 València. Tel. 963.82.83.69 - Fax: 963.82.83.70 - Email: enric.crespo@uv.es

(\*\*) Centro adscrito a la Universitat de València. C/ Rei En Jaume I. N° 2.- 46470 Catarroja. València. Tel. 961.22.03.80 -Fax 961.26.99.33 - Email: fvilla@florida-uni.es.

## 1. INTRODUCCIÓN

Las aeronaves requieren un mantenimiento periódico para que puedan operar con la máxima seguridad, fiabilidad, y con los menores costes posibles. Los fabricantes de aviones así como los organismos responsables de cada país indican con qué periodicidad una aeronave debe ser sometida a dichas tareas de mantenimiento y de qué características han de ser éstas.

Nuestro trabajo es una aproximación para secuenciar un problema de mantenimiento de aeronaves ajustado a las condiciones reales de una compañía del sector. Esta empresa no sólo cubre las necesidades de mantenimiento de sus propias aeronaves, sino que presta este servicio a otras empresas. Trabaja sobre diversas flotas y elabora un *Plan de Mantenimiento* para cada una de ellas. Este Plan recoge todas las tareas a realizar en cada tipo de mantenimiento, se elabora utilizando las indicaciones que establece el fabricante y es aprobado por la Aviación Civil Española, o por el organismo correspondiente del país al que pertenece la compañía de la aeronave a revisar.

Existen tres tipos habituales de mantenimiento: *Mantenimiento en Línea*, *Mantenimiento Menor* y *Mantenimiento Mayor (Overhaul)* que incluyen distintas inspecciones. Nuestro trabajo lo hemos centrado en el Mantenimiento Menor, y dentro de éste, en el estudio y análisis de la problemática de la “Inspección C” de la flota B-747 o Jumbo.

La compañía se plantea el problema como demandante del servicio y también como oferente del mismo a otras sociedades que no pueden, o no están interesadas, en realizarlo por si mismas. Por tanto, el incremento de la eficiencia en su realización es importante por diversos motivos. Como demandante del servicio para su propia flota permite aumentar la productividad y rentabilidad de la misma ya que, al reducir el tiempo del proceso, el avión puede estar operativo antes reduciéndose el coste de oportunidad que soporta la empresa al tener la aeronave parada. Además, puede ampliar su oferta sin necesidad de incrementar su inversión en partidas de activo fijo. Por otro lado, se encuentran los beneficios comerciales al poder incrementar los servicios ofrecidos.

Como oferente del servicio de mantenimiento la mayor eficiencia en la revisión reduce sus costes estructurales y aumenta su competitividad. Lo primero es debido a las dimensiones de los hangares y aviones que no permiten trabajar simultáneamente con más de una aeronave. Lo segundo es consecuencia de la fuerte competencia que existe en este servicio donde se valora tanto la calidad del mismo como el periodo durante el cual el avión no está disponible.

Además de los factores económicos mencionados, hay otras razones por las cuales a la empresa también le podría interesar disponer de un método informático de planificación de las tareas de mantenimiento. En primer lugar, permite hacer un análisis exhaustivo de la revisión en el cual se pueden detectar deficiencias o aspectos que se habrían pasado por alto. Cuando sólo se controla manualmente se recogen actuaciones perpetuadas en el tiempo, fruto de la experiencia o de la transmisión oral entre los operarios, que no suelen aparecer formalizadas por escrito en ningún documento y que carecen de una efectividad comprobada. Un segundo motivo es que mejora la toma de decisiones, tanto actuales como futuras, ya que permite simular de forma rápida qué sucedería si cambiasen los datos del problema. Por último, puede servir como guía para resolver otras posibles revisiones aunque sean de flotas distintas.

En nuestro trabajo hemos propuesto un algoritmo para mejorar la planificación de las tareas, así como la duración total del proceso de mantenimiento "C" de los Boeing 747. Para la realización del trabajo hemos usado la información facilitada por la compañía. Es muy importante señalar que muy frecuentemente, y en concreto en el caso aquí tratado, los problemas reales de secuenciación presentan notables diferencias con los modelos teóricos por lo cual, es necesario generar procedimientos adaptados al modelo concreto a resolver, si se quiere encontrar un método realmente operativo. El elemento más significativo de nuestro caso, aparte de sus considerables dimensiones, es que presenta diferencias importantes respecto al problema clásico de secuenciación de proyectos con recursos limitados (RCPS). Los más importantes son: la existencia de otro tipo de relaciones entre las actividades además de las de precedencia y el hecho de que determinadas actividades tengan que respetar, además de las restricciones de

precedencia y la disponibilidad de recursos, restricciones temporales específicas de nuestro modelo.

TABLA 1.1.: FASES DE LA REVISIÓN.

	DENOMINACIÓN DE LA FASE
<b>Fase 0</b>	PREPARACIÓN.
<b>Fase 1</b>	TRABAJOS DE ENTRADA.
<b>Fase 2</b>	LIMPIEZA DEL AVIÓN.
<b>Fase 3</b>	APERTURA DE REGISTROS Y DESMONTAJES.
<b>Fase 4</b>	INSPECCIÓN.
<b>Fase 5</b>	ENGRASES Y APLICACIÓN DE DINITROL.
<b>Fase 6</b>	PUESTA CONFIGURACIÓN MANDOS DE VUELO.
<b>Fase 7</b>	PRUEBAS OPERATIVAS.
<b>Fase 8</b>	CIERRE DE REGISTROS Y MONTAJES.
<b>Fase 9</b>	TRABAJOS FINALES EN EL HANGAR.
<b>Fase 10</b>	TRABAJOS DE SALIDA.

*Fuente:* ELABORACIÓN PROPIA

En el apartado 2 describimos el problema planteado. En el 3 analizamos el RCPS, un modelo clásico de la Dirección de Operaciones, y su posible adecuación al caso tratado. El apartado 4 contiene nuestra propuesta concreta para abordar el problema, tanto en la decisión referente al esquema general del procedimiento como en la elección de los componentes concretos del mismo. En el apartado 5 explicamos los elementos concretos que diferencian nuestro algoritmo del clásico RCPS que utilizamos como guía. El último contiene los resultados y conclusiones de nuestro trabajo. Para acabar exponemos las referencias utilizadas.

## 2. DESCRIPCIÓN DEL PROBLEMA

La revisión “C” presenta un conjunto de actividades, vinculadas entre sí por un conjunto de relaciones de diversas clases, que se han de realizar utilizando unos recursos disponibles que son limitados. Los elementos del problema son los siguientes:

A) **Actividades.** Consta de 644 actividades realizadas por distintos equipos que tienen asignadas las tareas según su especialidad. El proceso tiene una serie de fases claramente diferenciadas (Ver tabla 1.1.).

Una fase estará integrada por un conjunto de actividades, todas ellas con un objetivo común, independientemente del lugar del avión donde se lleven a cabo o del equipo que las realice.

Además, según la información facilitada, las fases 2, 6, 9 y 10 tienen una única actividad final. Esto permite desagregar el proyecto en cuatro subproyectos con las características siguientes:

- El primero, formado por las fases 0, 1 y 2, con un total de 8 actividades reales.
- El segundo, formado por las fases 3, 4, 5 y 6, con un total de 373 actividades reales.
- El tercero, formado por las fases 7, 8 y 9, con un total de 190 actividades reales.
- El cuarto, formado por la fase 10, con un total de 7 actividades reales.

La duración de las actividades, así como los tipos y la cantidad de recursos que necesita es facilitada por la empresa.

Analizando la información definimos la tipología de relaciones que podían establecerse entre las actividades:

- 1) *Relaciones de precedencia:* Debido a razones tecnológicas y a la sucesión de fases de la revisión, algunas actividades deben finalizarse antes de que otras puedan comenzar. En nuestro problema encontramos 1.556 relaciones de este tipo.
- 2) *Relaciones de complementariedad o simultaneidad:* Existen actividades que deben ejecutarse conjuntamente porque los trabajos que implica la realización de una ayudan a la realización de la otra. Se realizan sobre la misma zona del avión. Durante la fase de inspección se establecen 13 relaciones de este tipo.

- 3) *Relaciones de incompatibilidad*: Se establecen entre actividades que no se pueden ejecutar a la vez por motivos de seguridad. En nuestro problema se producen 43 relaciones de esta clase.

Además, por motivos técnicos, determinadas actividades están sujetas a restricciones temporales en su instante de secuenciación. Son de dos tipos:

- 4) *Actividades ininterrumpibles*: en el sentido de que son actividades que se han de terminar en la jornada de trabajo en la que se empezaron.
- 5) *Actividades encajables*: hemos denominado así a determinadas actividades que, por su toxicidad, deben acabarse justo cuando acaba el turno de trabajo de mañana. De esta forma la zona del avión en la que se realizan dispondrá de 16 horas libres para que se evaporen los gases. Esta restricción hace que ese tipo de actividades sólo pueda secuenciarse cuando el tiempo restante hasta el final del turno de mañana sea exactamente el que la actividad necesita para ser realizada.

TABLA 1.2.: EQUIPOS DE TRABAJO Y ACTIVIDADES ASIGNADAS.

EQUIPO	TIPO DE ACTIVIDADES ASIGNADAS	Cantidad
075	TRABAJOS DE CONTROL DE CALIDAD EN TRENES DE ATERRIZAJE	2
076	TRABAJOS DE CONTROL DE CALIDAD EN CABINA DE TRIPULACIÓN	1
079	TRABAJOS DE CONTROL DE CALIDAD EN BODEGA, A/A, COMP.E/E Y KARMANS	2
100	TRABAJOS DE MECÁNICA EN PUERTAS DE PASAJEROS	44
101	TRABAJOS DE MECÁNICA EN FUSELAJE POSTERIOR Y ESTABILIZADORES	31
102	TRABAJOS DE MECÁNICA EN BORDE DE SALIDA Y MANDOS DE VUELOS DE LAS ALAS	68
103	TRABAJOS DE MECÁNICA EN CABINA DE PASAJEROS	50
104	TRABAJOS DE MECÁNICA EN BORDE DE ATAQUE DE LAS ALAS, MOTORES Y PYLONS	189
105	TRABAJOS DE MECÁNICA EN TRENES DE ATERRIZAJE	39
106	TRABAJOS DE MECÁNICA EN CABINA DE TRIPULACIÓN Y SALÓN SUPERIOR	30
107	TRABAJOS DE MECÁNICA EN TANQUES DE COMBUSTIBLE	19
109	TRABAJOS DE MECÁNICA EN BODEGAS INFERIORES, AIRE ACONDICIONADO Y KARMANS	33
153	TRABAJOS DE MECÁNICA EN EL EQUIPO DE SALVAMENTO DE LA CABINA DE PASAJEROS	3
156	TRABAJOS DE MECÁNICA EN EQUIPO SALVAMENTO DE LA CABINA DE TRIPU. Y SALÓN	1
203	TRABAJOS DE AVIÓNICA EN CABINA DE PASAJEROS	15
206	TRABAJOS DE AVIÓNICA CABINA DE TRIPULACIÓN, SALÓN, COMP. E/E Y BODEGAS	26
208	TRABAJOS DE AVIÓNICA EN EL RESTO DEL AVIÓN	41
308	TRABAJOS DE LIMPIEZA EN TODO EL AVIÓN	1
803	TRABAJOS DE PINTURA EN LA CABINA DE PASAJEROS	4
806	TRABAJOS DE PINTURA EN CABINA DE TRIPULACIÓN, SALÓN Y COMP. E/E	3
808	TRABAJOS DE PINTURA EN EL RESTO DEL AVIÓN	24
903	TRABAJOS DE GUARNECIDO O INTERIORES EN CABINA DE PASAJE	12
906	TRABAJOS DE GUARNECIDO O INTERIORES EN CABINA DE TRIPULACIÓN Y SALÓN	4
908	TRABAJOS DE GUARNECIDO O INTERIORES EN EL RESTO DEL AVIÓN.	2
		<b>644</b>

Fuente: ELABORACIÓN PROPIA

Obviamente, en el procedimiento realmente implementado, establecemos unos parámetros de holgura respecto al momento de secuenciación de ambos tipos de

actividades. Lo hacemos así porque consideramos que este proceder responde a lo que se hace en la práctica.

**B) Recursos.** Los recursos necesarios para poder llevar a cabo la revisión del Boeing-747 son renovables, es decir, que su cantidad disponible se renueva periodo a periodo y sólo la cantidad total utilizada en cada instante está limitada. Hay un total de 33 recursos diferentes que se dividen en dos grandes tipos: *recursos humanos* y *recursos materiales*.

*b.1.) Recursos Humanos:* La empresa ha formado diversos equipos de trabajo para la realización de las actividades (Ver tabla 1.2.). Las personas necesarias son cedidas, por el tiempo que dure la revisión, por las diferentes secciones, talleres o unidades de trabajo existentes en la empresa. Cada equipo tiene asignado un número determinado de actividades (Ver tabla 1.2., 3ª columna). La asignación de tareas a los diferentes equipos de trabajo viene determinada tanto por la especialidad requerida para llevar a cabo la actividad como por la zona de trabajo donde se lleve a cabo. Por ejemplo, para realizar una determinada tarea se necesitara no un mecánico en general sino un mecánico en motores. Esto hace más restrictivo el problema desde el punto de vista de los recursos ya que no se pueden transferir recursos humanos de unos equipos a otros, a pesar de compartir la misma especialidad profesional general.

*b.2.) Recursos Materiales:* El análisis de los mismos nos lleva a concluir que, en la práctica, son ilimitados para el tipo de tareas a realizar. Es decir, que toda la plantilla podría estar trabajando a la vez sin necesidad de incrementar la cantidad actual disponible de este tipo de recursos.

**C) Objetivo.** Encontrar aquella secuenciación de actividades que permita llevar a cabo la revisión en el menor tiempo posible respetando, tanto la limitación de recursos, como los diferentes tipos de relaciones que existen entre las diferentes actividades y las restricciones temporales por las que se vean afectadas. Obviamente el trabajo se ha de hacer manteniendo los estándares de calidad y seguridad exigidos.

El problema real es de mayor complejidad que la aproximación aquí realizada y contiene otros elementos que se podrían considerar en un modelo más detallado (aparición de imprevistos, variación temporal de los recursos disponibles, diversos turnos de trabajo...) que puede ser abordado más adelante en función del interés de la empresa.

Toda esta información nos permite delimitar que nos encontramos, si miramos las líneas generales, con un problema muy parecido al que la literatura recoge como de secuenciación de proyectos con limitación de recursos (Resource Constrained Project Scheduling - RCPS -). No obstante, como hemos visto en la descripción del mismo, el problema real presenta un conjunto de elementos diferenciadores de considerable importancia que impide que su resolución sea una mera aplicación de los métodos propuestos para el RCPS.

Estas diferencias, así como la cantidad de tipos de recursos empleados, hacen imposible la utilización de los paquetes comerciales de secuenciación, debiéndose buscar una solución ajustada al caso concreto.

### **3. EL R.C.P.S. : UN MODELO PRÓXIMO A NUESTRO CASO**

#### **3.1. El modelo clásico del RCPS**

El problema clásico de RCPS puede ser expresado como un proyecto que consta de un conjunto  $J = \{0, 1, \dots, n, n+1\}$  de actividades o tareas que deben ser ejecutadas, sin posibilidad de interrumpir su proceso una vez éste ha comenzado. Las tareas 0 y  $n+1$  son ficticias y representan el inicio y el final del proyecto, respectivamente. Además, se asumen los siguientes supuestos:

- a) Entre las actividades se establecen relaciones del tipo “Fin-Inicio”, lo cual implica que ninguna actividad  $j$  puede comenzar hasta que todas sus inmediatamente predecesoras, recogidas en el conjunto  $P_j$ , hayan finalizado.
- b) No existen fechas obligadas sobre las actividades. Es decir, a priori no se establece una fecha tope de finalización a las actividades.



- c) Cada actividad se realizará mediante un único modo de operación (“single mode case”). Es decir que el tiempo necesario para ejecutar una actividad no puede reducirse asignando recursos adicionales.
- d) Cada actividad  $j$  tiene una duración,  $d_j$ , constante (el tiempo de puesta en marcha se considerará despreciable o incluido en la duración establecida).
- e) La realización de las actividades requiere la utilización de recursos renovables que presentan una capacidad limitada. Existen  $K$  tipos de recursos renovables recogidos en el conjunto  $K = \{1, 2, \dots, K\}$ .
- f) La disponibilidad de los distintos recursos,  $R_k$ , es una cantidad conocida y constante a lo largo del proyecto.
- g) Cada actividad  $j$  requiere para su proceso  $r_{j,k}$  unidades constantes de cada recurso  $k \in K$ .
- h) Los parámetros,  $r_{j,k}$  y  $R_k$  son cantidades enteras y no negativas.
- i) Para las actividades que marcan el comienzo y el final del proyecto tanto su  $p_j$  como su  $r_{j,k}$  para todo  $k \in K$  son iguales a cero.
- j)  $F_j$  indica el instante final de la actividad  $j$ .
- k) A través de la resolución del problema, se intenta determina un vector de tiempos finales  $(F_1, F_2, \dots, F_n)$  denominado secuencia, que respeta las relaciones de precedencia, la limitación de recursos y que hace que el tiempo de ejecución del proyecto sea mínimo. Al conjunto de actividades que están en proceso (activas) en un momento del tiempo  $t$ , se le llama  $A(t) = \{j \in J \mid F_j - d_j \leq t < F_j\}$ .

El problema así definido puede plantearse (Christofides et al 1987) como sigue:

$$\text{Min } F_{n+1} \tag{1}$$

$$F_h \leq F_j - d_j \quad j = 1, \dots, n+1; h \in P_j \tag{2}$$

$$\sum_{j \in A(t)} r_{j,k} \leq R_k \quad k \in K; t \geq 0 \tag{3}$$

$$F_j \geq 0 \tag{4}$$

La función objetivo (1) consiste en minimizar el tiempo de finalización de la última actividad del proyecto y por lo tanto, del tiempo total del proyecto (*makespan*). La restricción (2) impone las relaciones de precedencia entre actividades que hay que

respetar. La restricción (3) limita, para cada tipo de recurso renovable  $k$  y cada instante  $t$ , que la cantidad de recursos utilizados por las actividades en curso no supere el máximo disponible. Finalmente, (4) define las variables de decisión.

### 3.2. Diferencias entre el modelo del RCPS y el problema abordado

Nuestro problema se ajusta a muchas de las características básicas del RCPS. Sin embargo, además de las similitudes presenta las siguientes notables diferencias:

1. Además de las relaciones de precedencia del esquema clásico, se presentan *relaciones de complementariedad (o simultaneidad) y de incompatibilidad*.
2. Existen las llamadas actividades *ininterrumpibles*. Veamos unas consideraciones que nos permitan entender sus consecuencias sobre el problema:

Supongamos que estamos en un instante del turno  $x$ ,  $Tm_x$ , y que en ese momento existen recursos suficientes para realizar una actividad  $j$  de duración  $d_j$ . Si  $Tn_x$  es el momento en el que finaliza el turno de trabajo  $x$  y  $t_r$  es el tiempo que restante del turno de trabajo  $x$  ( $t_r = Tn_x - Tm_x$ ), podemos encontrar las siguientes situaciones:

- a)  $d_j \leq t_r$ . En este caso la actividad  $j$  se realizará en ese turno de trabajo.
- b)  $d_j > t_r$ . En este caso debemos diferenciar:

b.1.) *Actividades normales*. Estas tareas comenzarían en el turno  $x$  y acabarían en el siguiente.

b.2.) *Actividades ininterrumpibles*. Estas tareas no pueden comenzar en el turno  $x$  y acabar en el turno siguiente. Por lo tanto, la ejecución de estas actividades habrá que retrasarla hasta el siguiente turno.

3. Existen las llamadas actividades *encajables*.

Aunque la función objetivo y las restricciones de precedencia y de recursos de la formulación del RCPS son válidas para nuestro caso, el modelo clásico no incluye las diferencias descritas. Habrá, pues, que buscar maneras de ajustar al modelo general

aquellas que sean posibles y, cuando no sea posible, retocar dicho modelo para incluir los aspectos diferentes que hayan quedado pendientes.

#### 4. NUESTRA PROPUESTA GENERAL PARA RESOLVER EL PROBLEMA

El RCPS, generalización del clásico problema de secuenciación job shop, pertenece a los problemas de optimización de tipo NP-hard (Blazewicz et al. 1983). Si no hubiese recursos limitados, el RCPS se reduce a un caso de CPM (Critical Path Method – Método del Camino Crítico) que puede resolverse en tiempo polinomial (Elmaghraby 1977). Teniendo en cuenta la inherente intratabilidad del RCPS, una multitud de métodos exactos y heurísticos se han ido proponiendo a lo largo del tiempo.

Entre las técnicas exactas podemos destacar la *Programación lineal y la Entera*, la *Programación dinámica* y la *Enumeración implícita con Branch and Bound*. La mayoría de trabajos publicados que plantean la resolución óptima del problema lo hacen a través de aplicar la técnica del Branch and Bound diferenciándose, normalmente, en el esquema de enumeración que utilizan. Hasta ahora (Kolisch et al 1998), los algoritmos exactos que han demostrado ser más competitivos han sido los de Brucker et al (1998); Demeulemeester et al. (1998); Mingozzi et al. (1997) y Sprecher (1996).

Sin embargo, aunque existen métodos exactos competitivos, cuando nos enfrentamos a problemas de gran tamaño, es indispensable la utilización de métodos aproximados, comúnmente llamados heurísticos, para resolverlos ya que la naturaleza combinatoria del problema alargaría el tiempo necesario para su resolución hasta extremos no aceptables. Por eso, el uso de procedimientos exactos de resolución se limita a proyectos de tamaño pequeño.

Entre las técnicas heurísticas las más frecuentemente usadas son la secuenciación basada en reglas de prioridad, el *Branch-and-Bound* truncado, *Disjunctive arc concepts* y las técnicas metaheurísticas.

Dado que el problema es de gran tamaño y con importantes variaciones respecto al esquema clásico del RCPS decidimos utilizar un método heurístico para resolverlo.

Además del tamaño otra razón de peso para esta opción es que el conjunto de particularidades del problema real son más fáciles de introducir en una metodología heurística que en una técnica de resolución exacta.

Proponemos utilizar la metodología de la *secuenciación basada en reglas de prioridad*. A pesar de ser el procedimiento más antiguo para tratar el RCPS, es todavía una de las más importantes técnicas heurísticas para solucionar el problema (Kolisch et al. 1996b) por diversas razones:

1. Es un metodología intuitiva y fácil de utilizar. Por ello, se emplea muy a menudo en paquetes informáticos que se comercializan a las empresas para resolver este tipo de problemas.
2. Es rápido desde el punto de vista computacional. Por ello, es útil en los métodos de búsqueda local y los metaheurísticos para conseguir una solución de partida a partir de la cuál generar las nuevas.
3. Si además se implementan técnicas de multi-pasada el algoritmo mejora los resultados.
4. Se recomienda (Valls et al. 1992) cuando los problemas de gran tamaño deben ser resueltos rápidamente siempre que se encuentren buenas reglas de prioridad.

Además de lo expuesto anteriormente, hemos decidido utilizar esta técnica porque:

5. Si utilizamos un método multi-pasada podemos presentar a la empresa una gama más amplia de secuencias factibles, lo que le permite ampliar su capacidad de decisión.
6. El ser un método intuitivo facilita su interpretación y comprensión. Esto es muy útil para exponer a los responsables de la empresa el método de resolución empleado ya que para entender esta metodología no es imprescindible poseer conocimientos demasiado técnicos.

### **Elección de los componentes de la técnica heurística**

A partir de la elección hecha sobre metodología de la secuenciación basada en reglas de prioridad para abordar el trabajo es necesario determinar que componentes concretas de esta técnica vamos a utilizar en cada uno de los diversos aspectos que tiene. Lo exponemos a continuación.

### 1. El esquema para generar secuencias

Los algoritmos creados a partir de la secuenciación basada en reglas de prioridad vienen definidos por dos componentes: un esquema capaz de generar secuencias posibles y una o varias reglas de prioridad. Se distinguen dos esquemas diferentes: *secuenciación en serie y en paralelo*. En ambos se generan secuencias factibles a partir de secuencias parciales en sucesivas etapas. Una secuencia parcial se define como un subconjunto de actividades a las que se ha asignado un tiempo de finalización. En cada etapa se construye el conjunto de actividades secuenciables al que se aplica la regla de prioridad que establece el orden en el que serán realizadas.

*TABLA 4.1.: CARACTERIZACIÓN DE NUESTROS SUBPROYECTOS*

	NACT	D	RF	RS
SUBPROYECTO 1	8	0,614815	0,07222	1
SUBPROYECTO 2	374	0,190126	0,058806	0,12886
SUBPROYECTO 3	191	0,117079	0,047668	0,14281
SUBPROYECTO 4	6	0,892857	0,178571	1

*Fuente:* ELABORACIÓN PROPIA

La mayor parte de los algoritmos propuestos en la literatura y usados en la práctica son algoritmos en paralelo ya que ofrecen, en general, mejores resultados que los algoritmos en serie (Álvarez et al 1989b). Kolisch et al (1996a) proponen estudiar la complejidad del grafo (*Network Complexity -NC-*) y lo restrictivo que es el proyecto con respecto a los recursos (*Resource Factor -RF-* y *Resource Strenght -RS-*) para escoger el esquema. En base a ello obtienen las conclusiones siguientes:

- a) Los parámetros significativos para utilizar un esquema u otro son el RF y el RS.
- b) Cuando el problema es “hard” (difícil), es decir, con un alto RF y/o un bajo RS es mejor utilizar un esquema en paralelo. Sin embargo, ante problemas “easy” (fáciles) es preferible aplicar un esquema en serie.

En nuestro problema las fases más importantes son la segunda y la tercera debido al número de actividades y porque en ellas se concentran las restricciones temporales.

Ambas presentan un RS próximo a cero por lo que lo clasificaríamos como “difícil”. Esto nos decidió por utilizar un esquema en paralelo.

## 2. Una técnica multi-pasada

Parece obvio pensar que cuantas más secuencias seamos capaces de generar, siempre que sea en un tiempo computacional razonable, mejor será la solución encontrada. Además, se observa (Kolisch 1996, Boctor 1990) que los resultados mejoran al implementar técnicas de multi-pasada.

Pensamos que la mejor alternativa era emplear tanto un *método multiprioridad determinista*, pasando varias veces el esquema con reglas diferentes, como un *método aleatorio (Biased Random Sampling)*, pasando varias veces el algoritmo y en cada pasada aplicaremos las mismas reglas que en el método determinista pero aleatorizadas.

## 3. Reglas a aplicar

Hemos analizado una serie de estudios computacionales que comprueban la bondad de diferentes reglas de prioridad.

TABLA 4. 2.: REGLAS CLÁSICAS DE PRIORIDAD UTILIZADAS POR DIVERSOS AUTORES

AUTORES	MEJORES REGLAS
Davis y Patterson, 1975	MINSLK > LFT > RSM
Cooper, 1976	CUMRED > GRPW > MTS > LFS
Lawrence, 1984	LST > GRPW > LFT > LF/S
Álvarez-Valdés et al, 1989	CUMRED > GRPW > LFT > LST > MTS > RSM
Boctor, 1990	MINSLK > LFT > RSM
Valls et al, 1992	MINSLK > GRPW > MTS
Kolisch, 1996	LFT > MINSLK > RSM > MTS > GRPW

Fuente : ELABORACIÓN PROPIA

Aunque todos trabajan con problemas más pequeños que el nuestro, Álvarez et al [1989a] concluyen que los algoritmos que trabajan bien en pequeños problemas también lo hacen en los grandes. Analizando el ranking de las reglas testadas expuesto en la tabla 4.2 y las conclusiones de estos estudios decidimos utilizar las siguientes reglas:

- LFT, RSM y la LST porque en la mayoría de los trabajos las reglas basadas en el camino crítico proporcionan muy buenos resultados. Además para problemas con un RS bajo, que son “difíciles”, las reglas que proporcionan mejores resultados son aquellas basadas en el camino crítico (Kolisch 1996a).
- CUMRED, GRPW y MTS porque también aparecen como las mejor valoradas en al menos dos trabajos de los analizados.
- Además, utilizamos también la regla SPT por su facilidad de implementación.

#### **4. Algunas adaptaciones previas necesarias**

Para aplicar el procedimiento anteriormente expuesto decidimos eliminar todas aquellas diferencias con el modelo general que fuera posible mediante retoques sencillos en la modelización. Este es el caso de las relaciones incluídas en el punto 1 del apartado 3.2. Lo hicimos de la manera siguiente:

- *Relaciones de simultaneidad*: dos actividades simultáneas serán sustituidas por una sola actividad con la mayor duración de las dos y cuyo consumo de recursos sea la suma de las sustituidas. Así se garantiza la simultaneidad de ambas. Esta sustitución no supone alterar de manera importante la situación porque las actividades complementarias tienen duraciones bastante parecidas ya que se realizan simultáneamente.
- *Relaciones de incompatibilidad*: para cada pareja de actividades incompatibles se creará un recurso *ficticio* renovable específico con una sola unidad disponible. El requerimiento de cada una de estas actividades del recurso mencionado será también uno. Así se evita la simultaneidad de ambas.

A pesar de estas modificaciones quedan los puntos 2 y 3 del apartado 3.2 que no ha sido posible solucionar en la modelización. Para ello adaptamos el algoritmo general a las particularidades citadas.

#### **5. EL ALGORITMO ADAPTADO**

En primer lugar hay que indicar que, además del conjunto tradicional de actividades elegibles en cada instante de secuenciación (ELG), se han creado otros dos llamados

ELGp (actividades elegibles posibles) y ELGnp (actividades elegibles no posibles). El primero de ellos albergará las actividades que, en un instante dado, pueden secuenciarse porque se dan las condiciones de precedencia y temporalidad. El segundo contendrá aquellas actividades *encajables* que, aunque cumplan las condiciones de precedencia, no pueden secuenciarse por motivos temporales

En cada instante de secuenciación se crea un conjunto llamado PMS (Posibles Momentos de Secuenciación) al que pertenecen el primer momento en que una actividad libera recursos como en el algoritmo clásico, y también los momentos donde se puede secuenciar alguna de las actividades que estaba vedada por razones temporales.

Comenzaremos por señalar el tratamiento de estas particularidades y, posteriormente, expondremos las líneas generales del algoritmo adaptado.

### **5.1. Tratamiento de las particularidades.**

#### *a) Actividades ininterrumpibles.*

No entrarán en la lista ELGp si no queda tiempo suficiente para acabarlas en el turno de trabajo en el que se esté secuenciando. Simulando la realidad, se utiliza aquí un parámetro de tolerancia que permite que la actividad acabe un poco antes o después de la finalización real del turno, aunque su tiempo de finalización a efectos de sus sucesoras es exactamente el instante final del turno.

Para ordenarlas en la lista de actividades disponibles (AVL) se les da prioridad sobre las actividades sin restricciones temporales. Entre ellas se ordenan por duración ya que la más larga será más difícil de secuenciar en instantes posteriores.

#### *b) Actividades encajables.*

Salvando la tolerancia que se les conceda igual que a las actividades ininterrumpibles, sólo las incluiremos en la ELGp si su duración encaja con el tiempo restante del turno. En el caso de que *no encajen* las guardamos en el conjunto ELGnp.



Como son más restrictivas que las anteriores en la AVL se ordenan antes que éstas. Entre ellas se ordenan por orden de llegada ya que son todas igualmente *urgentes*, es decir, difíciles de encajar.

## 5.2. Modificación y adaptación de las etapas.

Hemos realizado una serie de cambios estructurales que consisten tanto en la modificación de algunas etapas del algoritmo paralelo clásico como en la creación de otras nuevas que nos permitan abordar las particularidades temporales con las que hemos de trabajar. La visualización general del algoritmo adaptado se puede observar en el Grafo 1. Los elementos que aparecen en el mismo son los siguientes:

- ELG: conjunto de actividades elegibles según el algoritmo paralelo clásico
- ELGp y ELGnp: conjuntos creados en el trabajo y ya explicados anteriormente.
- AVL: conjunto de actividades disponibles para secuencias según el algoritmo paralelo clásico.
- ACT: conjunto de actividades en ejecución según el algoritmo paralelo clásico.
- JOBS: conjunto de actividades pendientes de secuenciar.
- PMS: conjunto de posibles momentos de secuenciación cuya construcción se explica más adelante.

Las etapas creadas o modificadas vienen indicadas en el grafo por un fondo oscuro.

Los aspectos modificados son los siguientes:

- Etapas de construcción de ELGp y ELGnp: han quedado suficientemente explicadas en el apartado 5.1.
- Etapa de construcción de las actividades disponibles (AVL): difiere del algoritmo clásico en que solamente consideramos las actividades que están en ELGp, descartando así aquellas que no pueden secuenciarse por motivos temporales.
- Etapa de ordenación de las AVL: sólo se ordenarán con el criterio de las reglas de prioridad las actividades no sometidas a restricciones temporales. Las otras se ordenarán delante de ellas siguiendo los criterios expuestos en el apartado 5.1.

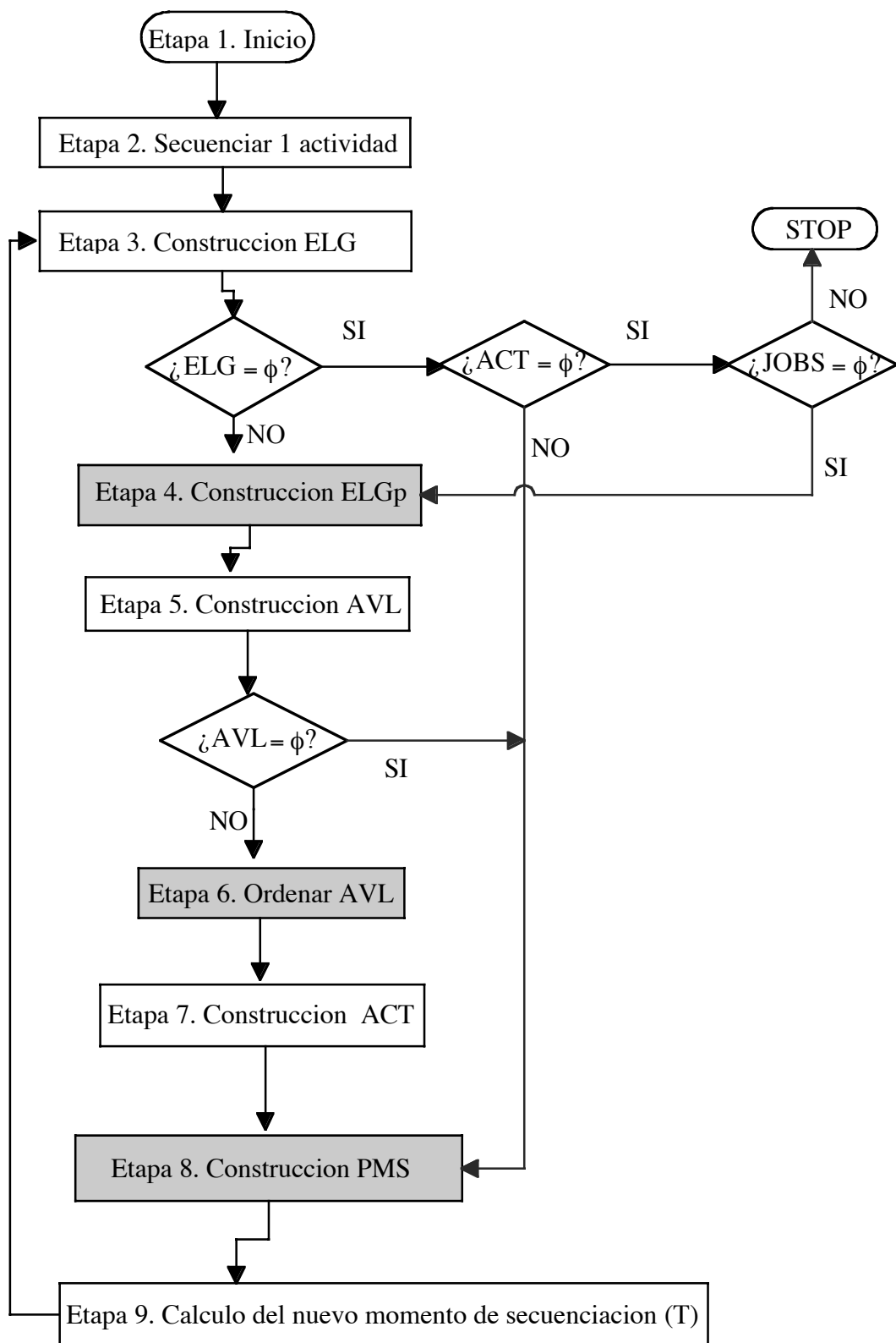
- Etapa de construcción del conjunto de momentos posibles de secuenciación (PMS): para construirlo diferenciamos entre las actividades pertenecientes a ACT y las de ELGnp.

Respecto a las primeras incluiremos en PMS su tiempo de finalización con los criterios siguientes:

- aquellas que no tengan restricciones temporales su momento real de finalización.
- para las ininterrumpibles su tiempo de finalización con la salvedad ya señalada de que acaban exactamente al final del turno en el caso que realmente lo sobrepasen dentro de la tolerancia permitida.
- para las encajables, el final del turno en curso.

Si las actividades pertenecen a ELGnp guardaremos en PMS el instante más próximo en que esas actividades puedan convertirse en ELGp es decir que vuelven a *encajar*.

**GRAFO 1**



Fuente: ELABORACION PROPIA

## 6. RESULTADOS COMPUTACIONALES Y CONCLUSIONES

### 6.1. Resultados

El algoritmo adaptado ha sido implementado en lenguaje C++ y ha corrido en un PC con procesador Pentium a 200 Mhz.

Los parámetros de tolerancia han sido el 20% en todos los casos.

Los resultados obtenidos con cada uno de los métodos multipasada vienen expuestos en las tablas 6.1. y 6.2. El significado de las columnas comunes entre ellas es el siguiente:

- El número de vértices (NV) o actividades y el número de aristas (NA) o relaciones de precedencia (*columnas 2 y 3, respectivamente*).
- Lo más pronto que pueden comenzar (*columna 4*). En este trabajo hay que tener en cuenta la existencia de restricciones temporales que condicionan la secuenciación de algunas actividades. Por ello no se puede comenzar cada subproyecto desde un tiempo cero sino que está condicionado por el subproyecto anterior. Para cada uno de ellos el tiempo de inicio viene dado por el mejor conseguido en el anterior.
- Su duración en unidades temporales con cada una de las reglas de prioridad.
- La solución que daría el CPM, es decir su duración si no existiese limitación de recursos ni restricciones temporales.

En ambos casos el instante final de proyecto viene indicado en la última línea.

Los resultados que muestran la tabla 6.2 corresponden a 500 iteraciones. El tiempo de computación consumido por la secuenciación de cada subproyecto se encuentra entre paréntesis pero, como término medio, se aproxima a un minuto para el conjunto de los cuatro subproyectos, es decir para el mantenimiento completo. Con fondo resaltado se indica la iteración en la que se consiguió la mejor solución.

TABLA 6.1.: SOLUCIONES CON EL MÉTODO MULTIPRIORIDAD

SUBPROYECTO	N.V.	N.A	INICIO	REGLAS DE PRIORIDAD							
				SPT	Cumred	GRPW	LFT	LST	MTS	RSM	CPM
1	10	14	0	225	225	225	225	225	225	225	225
2	376	974	225	2.486	2.426	2.426	2.426	<b>2.366</b>	2.426	2.486	1.406
3	193	416	2.591	2.152	2.148	2.148	2.148	2.160	2.148	2.148	1.655
4	8	9	4.739	160	160	160	160	160	160	160	160
<b>FINAL PROYECTO</b>			<b>4.899</b>								

Fuente: ELABORACIÓN PROPIA

TABLA 6.2.: SOLUCIONES CON EL MÉTODO MULTI-PASS SAMPLIG

SUBPROYECTO	N.V	N.A.	INICIO	REGLAS DE PRIORIDAD							
				SPT	Cumred	GRPW	LFT	LST	MTS	RSM	CPM
1 <i>(tiempo)</i>	10	14	0	225 <i>(1,98'')</i>	225 <i>(1,98'')</i>	225 <i>(1,98'')</i>	225 <i>(1,92'')</i>	225 <i>(1,92'')</i>	225 <i>(1,86'')</i>	225 <i>(1,81'')</i>	225
ITERACIÓN				[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	
2 <i>(tiempo)</i>	376	974	225	2.436 <i>(26,36'')</i>	2.366 <i>(29,70'')</i>	2.366 <i>(29,72'')</i>	2.366 <i>(30,38'')</i>	2.366 <i>(30,86'')</i>	2.366 <i>(30,32'')</i>	2.366 <i>(33,01'')</i>	1.406
ITERACIÓN				[36]	[5]	[8]	[7]	[0]	[2]	[52]	
3 <i>(tiempo)</i>	193	416	2.591	2.148 <i>(12,47'')</i>	2.148 <i>(16,36'')</i>	2.148 <i>(13,89'')</i>	2.148 <i>(13,84'')</i>	2.148 <i>(13,78'')</i>	2.148 <i>(13,29'')</i>	2.148 <i>(13,89'')</i>	1.655
ITERACIÓN				[0]	[1]	[1]	[1]	[3]	[0]	[1]	
4 <i>(tiempo)</i>	8	9	4.739	160 <i>(1,98'')</i>	160 <i>(1,92'')</i>	160 <i>(1,87'')</i>	160 <i>(2,03'')</i>	160 <i>(1,98'')</i>	160 <i>(1,92'')</i>	160 <i>(1,98'')</i>	160
ITERACIÓN				[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	[ 0 ]	
<b>FINAL PROYECTO</b>			<b>4.899</b>								

Fuente: ELABORACIÓN PROPIA

## 6.2. Conclusiones

Las conclusiones que podemos obtener del trabajo realizado se pueden resumir de la manera siguiente:

1. La metodología heurística se muestra como absolutamente conveniente para tratar problemas de RCPS que presenten particularidades tan especiales como el aquí abordado. Además el algoritmo desarrollado consigue sus soluciones en un tiempo despreciable para el tipo de problema. De esta forma, es posible optar entre diversas soluciones alternativas.

2. El análisis de los parámetros del proyecto nos indicaban claramente características de extrema rigidez inherentes al mismo. Como se observa en la tabla 4.1. el problema es muy restrictivo respecto a los recursos en sus fases centrales que se incluyen en los subproyectos 2 y 3. Estos subproyectos ocupan cerca del 97% del tiempo del proyecto por lo que podemos considerar el proyecto como globalmente muy restrictivo.

Además las restricciones temporales de ciertas actividades (un 15% aproximadamente) imprimen otro aspecto de rigidez al problema.

Los resultados computacionales obtenidos apuntan claramente en este mismo sentido como se comprueba en el hecho de que, a pesar de la gran cantidad de actividades, los tiempos y las secuencias son muy similares. Esto nos indica que las posibilidades combinatorias son muy escasas.

Esta rigidez también se muestra en el hecho de que las soluciones aleatorias no mejoren las obtenidas por el método determinista. Aunque, en general, mejoran los resultados conseguidos por las reglas deterministas, siempre hay alguna de ellas que consigue llegar al nivel de las aleatoria. Es más, hemos probado a incrementar las iteraciones hasta 1000 sin haber conseguido mejorar las soluciones.

3. Aparte de la obtención de las secuencias globales el algoritmo nos permite hacer un análisis de la utilización de los recursos en cada instante. De este modo podemos saber cuales de ellos son críticos y en qué momentos lo son. Esto permitiría decidir sobre la conveniencia de su ampliación con la finalidad de mejorar el tiempo de secuenciación.

4. Los cuellos de botella que se forman al final de cada subproyecto alargando el coste total del proyecto hacen dudar de que los cuatro subproyectos disjuntos en los

que se supone que se puede dividir el proyecto según la información recibida lo sean totalmente. Es necesario revisar, junto con la empresa, los resultados obtenidos así como la información facilitada para comprobar su corrección. Esto nos muestra, una vez más, el interés de disponer de métodos informatizados que permitan analizar la información y sus consecuencias sobre los resultados como manera de medir su coherencia.

5. Con la información revisada y ajustada a los procesos reales es posible obtener rápidamente nuevas soluciones. De esta manera, en función de las necesidades de la empresa, es posible decidir si el algoritmo implementado es suficiente o, por el contrario, sería interesante desarrollar alguno más sofisticado de tipo metaheurístico.

## **BIBLIOGRAFÍA**

- ALVAREZ-VALDES R. y TAMARIT J.M. (1989a): "Heuristic algorithms for resource-constrained project scheduling: A review and an empirical analysis". En *Advances in Project Scheduling*, editado por R. Sowinski, J. Weglarz Elsevier Science Publishers B.V.,Amsterdam, 1989, págs. 114-134.
- ALVAREZ-VALDES R. y TAMARIT J.M. (1989b): "Algoritmos heurísticos deterministas y aleatorios en secuenciación de proyectos con limitación de recursos". *Qüestiiio* nº13, págs. 173-191.
- BLAZEWICZ J, LENSTRA J.K. y RINNOOY KAN A.H.G (1983):" Scheduling subject to resource constraints: classification and complexity". *Discrete Applied Mathematics* págs. 11-24
- BOCTOR F.F. (1990): "Some efficient multi-heuristic procedures for resource-constrained project scheduling". *European Journal of Operational Research* 49 págs. 3-13.
- BRUCKER P., KNUST S., SCHOO A. y THIELE O (1998)."A branch and bound algorithm for the resource-constrained project scheduling problem". *European Journal of Operational Research* 107 págs 272-288.
- COOPER D.F. (1976). "Heuristics for scheduling resource-constrained projects : an experimental investigation". *Management Science* 22 (11) págs 1186-1194.

- CHRISTOFIDES N., ALVAREZ-VALDÉS R. y TAMARIT J.M. (1987) ."Project scheduling with resource constraints: A branch and bound approach". European Journal of Operational Research 29 págs 262-273.
- DAVIS E.W. y PATTERSON J.H. (1975): "A comparison of heuristic and optimal solutions in resource-constrained project scheduling". Management Science 21 (8) págs 944 - 955.
- DEMEULEMEESTER E. y HERROELEN W. (1998): "New benchmark results for the resource-constrained project scheduling problem". Management Science.
- ELMAGHRABY S.E. (1977): Activity Networks : Project Planning and Control by Network Models. Wiley, New York
- KOLISCH R. y DREXL A. (1996): "Adaptative search for solving hard project scheduling problems". Naval Research Logistics 43 págs 987-999.
- KOLISCH R (1996a): "Efficient priority rules for the resource- constrained project scheduling problem". Journal of Operations Management, 14 págs 179-192.
- KOLISCH R (1996b): "Serial and parallel resource-constrained project scheduling". Journal of Operations Management 14 (3) págs 179-192.
- KOLISCH R. y HARTMANN S. (1998): "Heuristic algorithms for solving the resource-constrained project scheduling problem: Classification and computational analysis". En Handbook on Recent Advances in Project Scheduling, editado por J. Weglarz , Kluwer Academic Publishers, Dordrecht.
- LAWRENCE S.R. (1985): "An experimental investigation of heuristic scheduling techniques. GSIA". Carnegie-Mellon, Pittsburgh
- MINGOZZI A., MANIEZZO V., RICCIARDELLI S. y BIANCO L. (1997): "An exact algorithm for project scheduling with resource constrained based on a new mathematical formulation". Management Science.
- SPRECHER A. (1996): "Solving tje RCPSP efficiently at modest memory requirements". Technical Report 425, Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel.
- VALLS V., PÉREZ M.A. y QUINTANILLA M.S. (1992): "Heuristic performance in large resource-constrained projects". Working paper .Departament D'Estadística i Investigació Operativa. Universitat de València, Spain.