

TESIS DOCTORAL

Sistema de planificación
estocástico de proyectos:
Implicaciones en la gestión de
riesgos

Iñaki Agirre Pérez



UNIVERSIDAD DE LA RIOJA

TESIS DOCTORAL

Sistema de planificación
estocástico de proyectos:
Implicaciones en la gestión de
riesgos

Iñaki Agirre Pérez

Universidad de La Rioja
Servicio de Publicaciones
2007

Esta tesis doctoral, dirigida por el doctor D. Joaquín Ordieres Mere, fue leída el 19 de enero de 2007, y obtuvo la calificación de Sobresaliente Cum Laude Unanimidad.

© Iñaki Agirre Pérez

Edita: Universidad de La Rioja
Servicio de Publicaciones

ISBN 978-84-690-6192-3

Sistema de planificación estocástico de proyectos

Implicaciones en la gestión de riesgos

TESIS DOCTORAL EN EL DEPARTAMENTO DE INGENIERÍA MECÁNICA
UNIVERSIDAD DE LA RIOJA

AUTOR:

Iñaki Agirre Pérez

DIRECTOR:

Joaquín Ordieres Meré

A mis padres

Resumen

En este trabajo planteamos un modelo del riesgo en el proyecto en red estocástico, expresado mediante redes de Petri estocásticas más generales. El modelo contempla la incertidumbre sobre la duración de las actividades, e incluye los planes de contingencia y los ciclos de repetición. Caracterizamos probabilísticamente la incertidumbre sobre el resultado exitoso o fallido del proyecto como conjunto y de cada una de sus actividades.

En el modelo se incluyen distribuciones de probabilidad para las variables aleatorias ligadas a la duración de las actividades. A partir de una simulación de Monte Carlo, se ejecutan algoritmos de secuenciación y asignación de múltiples recursos cuyos resultados agregados muestran una información completa sobre las expectativas de comportamiento de la red estocástica del proyecto.

Hemos desarrollado un prototipo de software en Java para demostrar la aplicabilidad de nuestra propuesta. El prototipo toma como entrada un proyecto o un portafolio de proyectos, y ejecuta los algoritmos de planificación para ofrecer la información sobre criticidad, probabilidad de éxito, consumo de recursos, coste, duración, etc...

El modelo de riesgo y el prototipo se han probado en un conjunto de proyectos de pequeño tamaño encontrados en la literatura y en la práctica profesional. Los experimentos han demostrado que la información obtenible para la gestión del riesgo en el proyecto es más extensa y precisa de la que se obtiene con las técnicas en uso.

Este modelo de riesgo puede ser adecuado para nuevas áreas gestionadas por proyectos donde el riesgo es significativo, como la investigación, las tecnologías de la información o el desarrollo de nuevos productos.

Abstract

In this work titled 'Stochastic Project Scheduling System: Implications for Risk Management' we develop a stochastic risk model for the project's network depicted by more general Stochastic Petri Nets. The model expresses the uncertainty about activities' duration, and it includes contingency plans and repetition cycles. Uncertainty about the successful or failed outcome of the project as a whole or of its activities is characterized probabilistically.

The random variables bound to activity durations are modelled by probability distributions. After a Monte Carlo simulation, scheduling and multiresource allocation algorithms are executed in order to collect aggregated measures that depict a comprehensive information about the expected behaviour of the stochastic project network.

The feasibility of our proposal has been proven by a software prototype built in Java®. The program reads a project or a multiproject as input, executes the scheduling algorithms, and displays information such as criticality, success chance, resource consumption, cost, makespan, etc.

The risk model and the prototype have been tested in a set of small-sized projects chosen from the literature and from the professional practise. The experiments have shown that the information we get for project risk management is more extensive and precise than the one you could get with previous techniques.

This risk model could be very significant in some new areas managed by projects such as research, information technology or aggressive product development, where the risk factor is high.

Résumé

Dans ce travail on envisage le modèle en risque dans le projet de Réseau Stochastique exprimé en Réseaux de Petri Stochastiques plus généraux. Le modèle envisage l'incertitude de la durée des activités et inclut les plans de contingence et les cycles à répétition.

On caractérise probabilistiquement l'incertitude sur le résultat réussite ou dèçu du projet comme un ensemble et comme chacune de ses activités.

Le modèle comprend les distributions des probabilités pour les variables aléatoires ligüées par la durée des activités. Partant d'une simulation de Monte Carlo, s'exécutent les algorithmes de séquencage et d'assignation des ressources dont les résultats ajoutés montrent une information complète sur les expectatives du comportement de réseau stochastique du projet. On a développé un prototype de software en Java pour démontrer l'applicabilité de notre proposition.

Le prototype prend comme entrée un projet et exécute les algorithmes de planification pour offrir l'information sur criticité, probabilité de succès, consommation des ressources, du coût, de la durée, etc...

Le modèle de risque et le prototype ont été prouvés dans un ensemble de petits projets trouvés dans la littérature et la pratique professionnelle. Les expérimentations ont démontré que l'information obtenue par la gestion du risque dans le projet est plus extense et précise que celle qui s'obtient avec les techniques à usage.

Ce modèle de risque peut être adéquat pour les nouveaux secteurs gérés par des projets où le risque est significatif, comme l'investigation, les technologies de l'information ou le développement des nouveaux produits.

Índice de contenidos

Sistema de planificación estocástico de proyectos.....	1
Implicaciones en la gestión de riesgos.....	1
Resumen.....	4
Abstract.....	5
Résumé.....	6
1. Introducción.....	10
1.1 Motivación.....	10
1.2 Disposición de la memoria.....	11
1.3 Notación y conceptos.....	12
1.3.1 El método del Camino Crítico.....	15
1.3.1.1 Cálculo de tiempos más tempranos.....	16
1.3.1.2 Cálculo de tiempos más tardíos.....	16
1.3.1.2 Cálculo de holguras.....	17
1.3.1.5 Criticidad.....	18
1.3.2 El método de Monte Carlo.....	19
1.3.2.1 Técnicas de reducción de la varianza.....	21
1.3.2.1.1 La transformación antitética.....	21
1.3.2.1.2 Las variables de control.....	22
1.3.2.1.3 La esperanza condicional.....	23
1.3.3 La programación de la cadena crítica y la gestión de márgenes de protección.....	24
1.3.4 La asignación de recursos.....	24
2. Estado del arte.....	30
2.1 Una visión general.....	30
2.2 El análisis de coste.....	32
2.3 El análisis del riesgo temporal.....	33
2.3.1 Las reservas de contingencia.....	33
2.3.2 El valor presente neto.....	34
2.3.3 La puja del proyecto.....	35
2.3.4 La simulación de Monte Carlo.....	35
2.3.4.1 Las técnicas de reducción de la varianza.....	35
2.3.5 La red de actividades alternativas.....	36
2.3.6 La ingeniería concurrente.....	38
2.3.6.1 Los ciclos de repetición.....	39
2.4 La gestión del portafolio.....	40
2.5 La caracterización del riesgo en el proyecto.....	41
2.6 El método de la cadena crítica.....	42
2.7 El problema de la planificación de proyectos con recursos limitados.....	43
2.7.1 Los métodos heurísticos.....	45

2.7.2 El problema de la planificación multimodo de proyectos.....	47
2.7.2.1 El problema del compromiso entre el tiempo y el coste.....	49
2.7.2.2 El problema del compromiso entre el tiempo y los recursos.....	50
2.7.3 Los métodos metaheurísticos.....	50
2.7.3.1 Los algoritmos genéticos.....	50
2.7.3.2 La técnica de las hormigas.....	52
2.7.3.3 La búsqueda tabú.....	52
2.7.3.4 La optimización por enjambre de partículas.....	53
2.8 Otros problemas de asignación de recursos.....	53
2.8.1 La nivelación de recursos.....	54
2.9 Los conjuntos difusos.....	55
2.10 Las redes neuronales.....	56
2.11 Otros problemas de planificación.....	57
2.11.1 El problema de los puntos de control.....	57
2.11.2 La reparación del programa.....	58
2.11.3 El problema del tiempo medio.....	59
2.12 Nuestra visión: redes temporales estocásticas de actividades con alternativas.....	59
3. Modelo de datos.....	61
3.1 La red estocástica del proyecto.....	61
3.2 Modelo de actividad.....	63
3.2.1 Activación.....	67
3.2.1.1 Prescindibilidad.....	70
3.2.2 Desarrollo.....	73
3.2.2.1 La duración probabilística.....	74
3.2.2.2 Los recursos complejos.....	76
3.2.3 Terminación.....	77
3.3 La red generalizada del proyecto.....	78
3.3.1 Las redes de alternativas múltiples.....	78
3.3.2 Los planes de mitigación del riesgo.....	78
3.3.3 Los ciclos repetitivos.....	82
3.3.3.1 Algoritmo de despliegue de las iteraciones en los ciclos.....	84
3.3.3.2 Cálculo de tiempos más tempranos.....	85
3.3.3.3 Cálculo de tiempos más tardíos.....	85
3.4 Factores de riesgo comunes a ciertas actividades.....	86
3.5 Solución de la red mediante la simulación de Monte Carlo.....	88
3.6 Exégesis de los resultados.....	91
3.6.1 La planificación restringida por los recursos.....	94
3.6.1.1 Paso 1: Inicialización.....	96
3.6.1.2 Paso 2.....	96
3.6.1.3 Paso 3.....	96
3.6.1.4 Paso 4.....	97
3.6.1.5 Paso 5.....	97

3.6.1.6 Paso 6.....	97
3.6.1.7 Paso 7: Vuelta atrás.....	98
4. Prototipo de software.....	99
4.1 Introducción.....	99
4.2 Descripción del prototipo.....	99
4.2.1 Casos de uso.....	99
4.2.1.1 Analizar el plazo.....	100
4.2.2 Diagramas de clases.....	102
4.2.3 Interfaz gráfica.....	105
4.3 Formatos de los ficheros.....	112
4.3.1 Fichero de entrada para el portafolio.....	112
4.3.1.1 Ejemplo de fichero de entrada.....	115
4.3.2 Fichero de configuración.....	118
4.4 Rendimiento.....	119
5. Experimentación.....	121
5.1 Experimento número 11.....	121
5.2 Experimento número 12.....	125
5.3 Experimento número 13.....	131
6. Conclusiones y líneas de futuro.....	136
6.1 Conclusiones.....	136
6.2 Líneas de futuro.....	138
7. Referencias.....	140

1. Introducción

1.1 Motivación

Se puede considerar que el estudio sistemático de la organización industrial comenzó con la publicación de “Principios de la Administración Científica” por [Taylor 1911]. Frederick W. Taylor postulaba, entre otros principios, la necesidad de descomponer las actividades en sus elementos constituyentes. En 1917¹ Henry Gantt desarrolló su diagrama de barras, explicitando por vez primera la necesidad industrial de planificar y programar.

Las técnicas descritas en este documento son aplicables a cualquier proyecto realizado en cualquier campo de la actividad humana, pero son especialmente útiles en aquellos proyectos complejos que se encuentren rodeados de un alto grado de incertidumbre. Los proyectos de desarrollo informático, de investigación, desarrollo e innovación, o de inversión rural en países en vías de desarrollo son algunos de los tipos de proyectos que se verán beneficiados por el uso de las técnicas aquí plasmadas.

El análisis del riesgo temporal es especialmente importante en los proyectos de innovación y en el desarrollo y lanzamiento de nuevos productos. En estos casos, los retardos en la entrega del proyecto resultan mucho más caros que las desviaciones del presupuesto.

La incertidumbre en los proyectos puede deberse a causas epistémicas, esto es, puede deberse a la ignorancia de los participantes en el proyecto sobre los mecanismos subyacentes en su realización. Aunque esta incertidumbre se reduzca con la experiencia y la habilidad de establecer las adecuadas analogías entre el proyecto a abordar, único por naturaleza, y los proyectos realizados anteriormente, la otra fuente de incertidumbre es aleatoria, y con esto queremos decir que se debe a una naturaleza intrínsecamente probabilística. En este último caso, sólo podremos esperar de los conocimientos y experiencia de los participantes una adecuada caracterización de la incertidumbre, caracterización sobre la que aplicaremos las técnicas y procedimientos desarrollados para extraer las previsiones y planes de contingencia más adecuados.

Algunas actividades tienen un rango amplio de posibles tiempos de terminación. Por ejemplo, algunas de las actividades en la fase de lanzamiento, como la obtención de los fondos para un proyecto pueden llevar el tiempo de una llamada de teléfono o necesitar ímprobos esfuerzos durante semanas, meses o años. En estos casos, una descomposición ulterior de la estructura del trabajo

¹ Algunas de estas ideas estaban ya en [Gantt 1910].

puede mostrar que existen muchas subactividades cuya ejecución depende de los resultados de otras subactividades anteriores. Es lo que [Williams 1999a] define como incertidumbre estructural, que a su vez tiene dos dimensiones, según el número de elementos y las dependencias entre los elementos.

En un proyecto de implementación de una tecnología nueva, por ejemplo, puede no estar claro si se necesitarán realizar actividades de formación con parte de la plantilla. En ciertos proyectos farmacéuticos actividades con pruebas adicionales dependen de factores externos como la aprobación de una nueva legislación. Tras una encuesta de satisfacción con el cliente, un producto final puede no necesitar más trabajo, o necesitar nuevas actividades para mejorarlo. Las pruebas de rendimiento normalmente conducen a situaciones similares, donde la ejecución de las tareas de ajuste o refinado depende del resultado de las tareas de prueba.

Otra fuente de complejidad en la programación de proyectos viene de la repetición de tareas. Si el número de repeticiones es conocido, se suele copiar varias veces la actividad en la planificación. Por ejemplo, en las negociaciones para la consecución de un acuerdo de compra se pueden secuenciar la primera, segunda y tercera rondas. Sin embargo, si el número de repeticiones es impredecible, como cuando se tiene que repetir la preparación y envío de unos planos de un edificio hasta conseguir la aprobación del arquitecto municipal, esta aproximación no es realizable. Otro ejemplo estaría en las pruebas y depuración del código de los programas informáticos, que requieren varias repeticiones hasta que se alcanza el objetivo de calidad deseado.

Nuestro modelo para tratar la incertidumbre en la gestión de proyectos puede ser aplicado a cualquier nivel de la EDP (Estructura de Descomposición del Proyecto: *WBS, Work Breakdown Structure*), ya sea en las fases iniciales de preparación de la oferta, o en la fase de ejecución del proyecto, cuando ya parte de éste ha sido realizado y nuevas actividades de mayor nivel de detalle van siendo añadidas.

Esta investigación se ha plasmado en la construcción de un prototipo de software con el que se demuestra la aplicabilidad de las técnicas desarrolladas. En cierto sentido, la investigación trata de saltar la brecha que se ha producido entre la investigación en operaciones de los últimos años y la práctica de la gestión de proyectos.

1.2 Disposición de la memoria

Esta memoria se divide en 6 capítulos. En esta “Introducción” motivamos la necesidad de realizar la investigación, presentamos las obras introductorias al campo y describimos los conceptos fundamentales y la notación que emplearemos en el texto. A continuación emprendemos en el

capítulo 2, “Estado del Arte”, una revisión comentada de las aportaciones significativas que se han realizado últimamente en la planificación de proyectos. El campo es bastante extenso, así que esta será una reseña centrada en los aspectos o problemas para los que esta tesis aporta algo. En particular revisamos las redes alternativas en la planificación de proyectos, los ciclos de repetición en la ingeniería concurrente, la gestión del portafolio o multiproyecto, la caracterización del riesgo en el proyecto, el problema de la planificación del proyecto con recursos limitados y el problema de los puntos de control en el proyecto. En el capítulo 3, “Modelo de datos”, desgranaremos las características del modelo estocástico que hemos desarrollado para capturar la incertidumbre y la complejidad de la gestión del riesgo asociada. El modelo divide la actividad en tres zonas, activación, desarrollo y terminación, lo que nos permite abordar la incertidumbre sobre la duración de la actividad y sobre el empleo de sus recursos, y también la incertidumbre sobre la propia red de actividades del proyecto. Veremos la caracterización de la red de actividades alternativas AND/OR y los ciclos de repetición de secuencias de actividades. En el capítulo 4, “Prototipo de Software”, describiremos el producto que hemos programado con la intención de demostrar que la aproximación caracterizada en el “Modelo de datos” es realizable. La descripción explica los detalles de la entrada y salida del prototipo, desde el punto de vista de la adecuación al modelo. El capítulo 5, “Experimentación”, trata sobre la aplicación del modelo y el prototipo a algunos problemas que hemos considerado significativos. Cerramos esta memoria con un capítulo 6, “Conclusiones y líneas de futuro”, en el que recopilamos los puntos fundamentales y abordamos lo que podrían ser futuras líneas de investigación. Como es costumbre, la bibliografía aparece al final, como “Referencias”.

El anexo A, en un CD, contiene la documentación del prototipo realizado en Java en formato Javadoc, con vistas a su mejor comprensión y a su posible integración posterior en nuevos desarrollos de investigación. El anexo B, en el mismo CD, contiene el código fuente del prototipo.

1.3 Notación y conceptos

La estructura de descomposición del proyecto es un árbol de paquetes de trabajo que se crea dividiendo sucesivamente éstos en sus partes constituyentes hasta llegar al nivel de actividad, con el detalle necesario para la gestión del proyecto. Normalmente está relacionada con una estructura de descomposición organizativa. Sobre estos conceptos recomendamos [Burke 1992] en su capítulo 4º y [Kerzner 1998] en su sección 11.10.

La red de actividades de un proyecto se describe mediante el par (N, A) donde $N = \{1, \dots, m\}$ representa el conjunto de nodos o vértices de la red y $A = \{(a_i, b_i) : a_i, b_i \in N, i = 1, \dots, n\}$ representa el conjunto de arcos dirigidos, siendo $a_i \in N$ el nodo inicial y $b_i \in N$ el nodo final. En la

representación más usual, la del Método de los Potenciales, las actividades se colocan en los nodos, y los arcos dirigidos plasman las relaciones de prelación entre las actividades [Roy 1962], [Roy 1964]. Se definen al menos una actividad inicial $A_1 \in N$ y una actividad final $A_n \in N$.

Cada actividad $A_i \in N$ que no sea una actividad inicial tiene un conjunto de predecesores $P_i \in N$. La actividad A_i se extiende durante d_i periodos de tiempo. En el PERT² [Malcolm et al. 1959] clásico, esta duración indeterminada se estimaba en un número determinado, aunque desde el PERT probabilístico [Fulkerson 1962] [MacCrimmon; Ryavec 1964] [Hartley; Wortham 1966] se le suele caracterizar como una variable aleatoria δ_i . En este trabajo usaremos funciones de densidad de probabilidad continuas para esta variable, pero los resultados son extensibles a las funciones discontinuas. Para una explicación del método PERT véanse los capítulos 3º, 4º y 5º de [Ordieres 1999] o el capítulo 10º de [Anderson et al. 2005]. Para estudiar las variables aleatorias en PERT, véase [Nádas 1979].

En general, en la investigación sobre planificación de proyectos, se suele considerar que una actividad no puede suspender su ejecución una vez que ha comenzado. Esta es la simplificación que nosotros hemos seguido, aunque haremos notar en el capítulo 2, “Estado del arte”, que no todos los investigadores la asumen. Si se permite la interrupción y reanudación posterior de actividades se puede acortar significativamente la duración del proyecto en aquellas situaciones en las que los recursos están severamente limitados.

Además de actividades y prelacones, el proyecto viene caracterizado por un conjunto de tipos de recursos disponibles $R = \{R_j : j = 1, \dots, r\}$, cada uno de los cuales tiene una capacidad finita k_j definida en $\vec{K} = \{k_j : R_j \in R, k_j \geq 0\}$. A su vez, estos recursos se pueden categorizar en renovables, no renovables y doblemente restringidos [Błażewicz et al. 1985], [Błażewicz et al. 1986]. De los recursos renovables, una actividad A_i consume un cierto número de unidades de algunos de ellos durante el periodo de ejecución. Esto se puede caracterizar también en un vector $\vec{C}_i = \{c_{ij} : A_i \in N, R_j \in R, 0 \leq c_{ij} \leq k_j\}$. Ejemplos de recursos renovables son las máquinas, o el personal.

Los recursos no renovables están limitados para el total del proyecto, sin restricciones sobre cada periodo. Ejemplos de recursos no renovable son el presupuesto del proyecto y los materiales de construcción.

Los recursos doblemente restringidos son los que están limitados a la vez en su totalidad acumulada, como los recursos no renovables, y en su capacidad por tiempo, como los renovables.

2 *Program Evaluation and Review Technique*, técnica desarrollada en 1958 en la Marina de los Estados Unidos para el desarrollo del sistema de misiles balísticos Polaris de la flota de submarinos.

En ocasiones, la actividad puede disponer de diversos, incluso infinitos modos de realización. Un modo determina la duración de la actividad, o al menos el tipo de la variable aleatoria δ_i , si la actividad resulta interrumpida, y los requerimientos de recursos de varias categorías. También determina los posibles flujos monetarios que ocurren al activar, durante el desarrollo y al terminar la actividad.

Si las variables de decisión f_k representan los tiempos de finalización de las actividades, el vector $\vec{F} = \{f_k : A_k \in N, 0 \leq f_k\}$ representa una de las posibles planificaciones del proyecto. Definida la planificación, es interesante definir el conjunto de actividades que se estarán ejecutando en un determinado momento t $A_{[t]} = \{A_i \in A : f_j - d_j \leq t < f_j\}$ planificación que será realizable con respecto de las restricciones de prelación y recursos si

$$f_i + d_j \leq f_j \quad j=1, \dots, v, A_i \in P_j \quad [f1]$$

$$\sum_{(j \in A_{[t]})} c_{jh} \leq k_h \quad h=1, \dots, r \quad t=1, \dots, T \quad [f2]$$

Veremos en el capítulo 3, “Modelo de datos”, como no siempre se sostiene la primera ecuación.

Las relaciones de prelación entre dos actividades A_i, A_j puede ser más variadas. Las relaciones de precedencia generalizadas de desfase mínimo se agrupan en cuatro clases [Elmaghraby; Kamburoski 1992]:

1. De fin a comienzo

$$s_i + d_i + l_{ij} \leq s_j, \quad A_i \in P_j \quad [f3]$$

donde l_{ij} es el tiempo de desfase positivo o negativo entre las actividades A_i, A_j y s_i, s_j los tiempos de comienzo respectivos.

2. De comienzo a comienzo

$$s_i + l_{ij} \leq s_j, \quad A_i \in P_j \quad [f4]$$

3. De comienzo a fin

$$s_i + l_{ij} \leq s_j + d_j, \quad A_i \in P_j \quad [f5]$$

4. De fin a fin

$$s_i + l_{ij} + d_i \leq s_j + d_j, \quad A_i \in P_j \quad [f6]$$

En esta investigación nos centraremos en las relaciones de precedencia de fin a comienzo sin tiempo de desfase, pero no por ello perderemos generalidad. En [Bartusch et al. 1988] se encuentra una forma estándar de representar todos los tipos de relaciones de precedencia generalizados como relaciones de precedencia del tipo 2, de comienzo a comienzo. Con pequeños ajustes es posible aplicar las mismas técnicas que vamos a emplear al resto de relaciones de precedencia.

La función de evaluación de la planificación del proyecto cuantifica la calidad de la solución alcanzada, y permite elegir entre varias soluciones. También se conoce como función objetivo a optimizar. La función de evaluación más común es la duración del proyecto. Se define como $\delta = f_n - s_1$ si A_1, A_n son las actividades inicial y final respectivamente.

Otras funciones de evaluación que se estudian son el valor presente neto [Bey et al. 1981], el coste o la calidad. Nosotros estudiaremos el problema de optimización para la minimización de la duración del proyecto. Sobre la evaluación del proyecto recomendamos [Baca 1990].

Sobre una red de actividades es posible, en principio, definir un orden topológico $<$ de modo que $A_i < A_j \Leftrightarrow A_i \in P_j$. Un tipo de relación de precedencia especial que debemos considerar aparece cuando se produce un ciclo en la red, y $\exists A_i, A_j \mid A_j \in P_i \wedge A_i \in P_j^*$ ³.

Entre los elementos de la red contamos también, como veremos, con ciertas restricciones sobre el tiempo en que se pueden producir algunos eventos, o el número de veces que se puede recorrer un determinado ciclo.

Cuando la red de actividades del proyecto puede ejecutarse con distintos caminos, sin necesidad de realizarlos todos, se conoce como red alternativa. Consideraremos que el resultado exitoso o fallido de una actividad A_i viene representado por una operación Or Exclusiva estocástica. Esta característica, junto con las operaciones lógicas And, Or y Not que enlazan las actividades entre sí, nos permitirá modelar una completa red alternativa con el objetivo de incluir la gestión del riesgo dentro de la planificación del proyecto.

1.3.1 El método del Camino Crítico

El MCC (en inglés *CPM, Critical Path Method*) o método del camino crítico [Kelley; Walker 1959]

³ P_j^* se define como:
$$A_k \in P_j \rightarrow A_k \in P_j^* \\ A_l \in P_k \wedge A_k \in P_j^* \rightarrow A_l \in P_j^*$$

fue desarrollado por la empresas Remington Rand Univac y Du Pont en Estados Unidos. Para un estudio general del método, véase el capítulo 6° de [Ordieres 1999].

El método tiene 3 pasos:

- 1.Cálculo de tiempos más tempranos
- 2.Cálculo de tiempos más tardíos
- 3.Cálculo de holguras

1.3.1.1 Cálculo de tiempos más tempranos

El tiempo temprano de un suceso trata de medir el tiempo mínimo necesario para llegar a ese suceso. Para ello se comienza por el suceso inicio del proyecto, al que se le asigna un tiempo temprano 0. Este suceso coincide en el tiempo con el comienzo temprano de todas las actividades definidas como iniciales. Luego se procede recursivamente, calculando los tiempos tempranos de las actividades sucesoras de cada actividad cuyo tiempo temprano conozcamos.

El tiempo temprano de un cierto suceso j , que representaremos por t_j , será

$$t_j = \max [t_i + t_{ij}] \quad \forall i \quad [f7]$$

donde t_{ij} es d_k , la duración de la actividad A_k que comienza en el suceso i y finaliza en el suceso j .

El tiempo temprano del suceso fin de proyecto tiene una importancia especial, pues nos indica el tiempo mínimo necesario para poder finalizar el proyecto.

1.3.1.2 Cálculo de tiempos más tardíos

Los tiempos tardíos se calculan empezando por el suceso fin del proyecto, al que se le asigna un tiempo tardío igual al tiempo temprano previamente calculado. Luego se procede recursivamente, calculando los tiempos tardíos de las actividades predecesoras de una dada.

El tiempo tardío de cierto suceso i , que representaremos por t_i^* será:

$$t_i^* = \min[t_j^* - t_{ij}] \quad \forall j \quad [f8]$$

1.3.1.2 Cálculo de holguras

La holgura de un cierto suceso, que representaremos por H_i , se define como la diferencia entre los tiempos tardíos y tempranos del suceso:

$$H_i = t_i^* - t_i \quad [f9]$$

La holgura de un suceso nos indica el número de unidades de tiempo en que puede retrasarse la realización del mismo, de manera que la duración del proyecto (medida por el tiempo temprano del suceso fin del proyecto) no experimente ningún retraso.

La holgura total de una actividad A_k definida entre los sucesos i y j , que representaremos por H_{ij}^T , se define como el tiempo que resulta de restar al tiempo tardío del suceso final el tiempo temprano del suceso inicial y la duración de la actividad, es decir:

$$H_k^T = H_{ij}^T = t_j^* - t_i - t_{ij} = t_j^* - t_i - d_k \quad [f10]$$

La holgura total de una actividad nos indica el número de unidades de tiempo en que puede retrasarse la realización de la actividad con respecto al plan previsto (fin de actividad temprano), de manera que la duración del proyecto no experimente ningún retraso.

Aquellas actividades cuya holgura total sea cero se denominan actividades críticas. Uniendo todas las actividades críticas se forma un camino que va desde el vértice que representa el suceso inicio del proyecto al vértice que representa el suceso fin del proyecto. Este camino recibe el nombre de camino crítico y resulta esencial para efectuar el control del proyecto. El responsable del proyecto deberá extremar la vigilancia de estas actividades críticas, pues un retraso en la realización de cualquiera de ellas producirá un retraso en la finalización del proyecto.

Holgura o flotante libre de una actividad A_k definida entre los sucesos i y j , que representaremos por H_{ij}^L , es el tiempo que resulta de restar al tiempo temprano del suceso final el tiempo temprano del suceso inicial y la duración de la actividad.

$$H_k^L = H_{ij}^L = t_j - t_i - t_{ij} = t_j - t_i - d_k \quad [f11]$$

La holgura libre representa la parte de la holgura total que puede ser consumida sin perjudicar a las actividades siguientes.

La holgura o flotante independiente de una actividad A_k definida entre los sucesos i y j , que representaremos por H_{ij}^I , es el tiempo que resulta de restar al tiempo temprano del suceso final el tiempo tardío del suceso inicial y la duración de la actividad.

$$H_k^I = H_{ij}^I = t_j - t_i^* - t_{ij} = t_j - t_i^* - d_k \quad [f12]$$

Las holguras ayudan a a clasificar la importancia de las actividades, según sean de signo negativo, positivo o nulas. No obstante, esta clasificación sólo es útil en los casos en los que los recursos no son una limitación del sistema. Incluso en estos casos, las decisiones de secuenciación de las actividades no tienen ningún reflejo en las holguras, lo cual reduce su utilidad. En el capítulo 2, “Estado del arte”, veremos algunas nuevas definiciones y usos de las holguras.

Para el estudio de las bases matemáticas del método del camino crítico remitimos al lector a [Kelley 1961], donde se demuestra que el cálculo de los tiempos tempranos y tardíos produce un programa que es óptimo respecto de la duración total del proyecto.

1.3.1.5 Criticidad

En la gestión de proyectos probabilística, se llama criticidad de una actividad a la probabilidad de formar parte del camino crítico. Las actividades con alta criticidad pueden actuar como un cuello de botella del proyecto, porque su retraso o adelanto repercutirá en la duración total del proyecto.

La criticidad condicional de una actividad es la probabilidad de que la actividad forme parte del camino crítico, supuesto que la actividad suceda. Particularmente en nuestro modelo, esto significa

que la actividad resulte activada. Es un dato que sugiere la atención que se le debe dedicar a la actividad durante su desarrollo.

Dominio relativo⁴ de un camino es la probabilidad de que resulte ser el camino crítico. Veremos algunos ejemplos de uso de este indicador en el capítulo 5, “Experimentación”.

Crucialidad de una actividad es el valor absoluto de la correlación entre la duración de la actividad y la duración del proyecto [Williams 1992]. Aunque esta medida resuelve algunos problemas de la criticidad, principalmente la dificultad de aplicación en un contexto de programación restringida por los recursos, tiene el poco intuitivo efecto de declarar de nula crucialidad las actividades de duración determinada [Elmaghraby 2000].

1.3.2 El método de Monte Carlo

El método de Monte Carlo toma su nombre del famoso Casino de Monte Carlo, ya que emplea cualquier técnica de muestreo estadístico para encontrar soluciones aproximadas a problemas cuantitativos. También se conoce como método de simulación por muestreo aleatorio. Gracias a la potencia de las computadoras, estos muestreos se pueden automatizar.

El método de Monte Carlo fue desarrollado por el matemático polaco Stanislaw Ulam en 1946, al comienzo de la era de la computación, con el objeto de traducir procesos descritos mediante ciertas ecuaciones diferenciales en formas equivalentes interpretables mediante una sucesión de operaciones aleatorias, para poder resolverlos más fácilmente. Ulam y Nicholas Metropolis publicaron su trabajo en [Metropolis; Ulam 1949], pero hay antecedentes del método entre los archivos clasificados de la investigación atómica durante la segunda guerra mundial, en particular en Los Alamos National Laboratory, New Mexico, USA⁵. En [Eckhart 1987] se cuentan las circunstancias de las que surgió el método de Monte Carlo. Para un estudio del método recomendamos [Brandt 1999], la sección 8.6 de [Brassard; Bratley 1987], [Knuth 1969], [Rubinstein 1981], [Sobol' 1994] y [Fishman 1996]. Sobre la eficiencia del método, véase la sección 5.1 de [Hammersley; Handscomb 1964] y [Fox 1986].

En el caso general, el método de Monte Carlo trata de evaluar la integral definida multivaluada

4 Relative dominance

5 citado en [Brassard; Bratley 1987]

2

$$\Psi = \int_0^1 \int_0^1 \cdots \int_0^1 f(u_1, u_2, \dots, u_n) du_1 du_2 \cdots du_n \quad [f13]$$

la cual se expresa resumidamente como

$$\psi = \int_{(0,1)^n} f(u) du \quad [f14]$$

Esta fórmula puede representar un problema estocástico o no. En cualquier caso, el método de Monte Carlo consiste en buscar una solución aproximada al problema introduciendo un vector aleatorio U. El vector aleatorio se distribuye uniformemente en la región de integración $(0,1)^n$. En este caso $f(U)$ representa una variable estocástica cuya esperanza es

$$E[f(\vec{U})] = \int_{(0,1)^n} f(u) \Phi(u) du \quad [f15]$$

donde Φ representa la función de densidad de probabilidad de U. Ya que Φ se distribuye uniformemente en la región de integración,

$$E[f(\vec{U})] = \int_{(0,1)^n} f(u) du \quad [f16]$$

Así, obtenemos la expresión probabilística de la integral Ψ

$$\Psi = E[f(\vec{U})] \quad [f17]$$

Los momentos de la variable estocástica $f(U)$ son pues la media Ψ y la desviación estándar σ . Definimos un estimador insesgado

$$H = \frac{1}{m} \sum_{k=1}^m f(\vec{U}^{[k]}) \quad [f18]$$

donde $U^{[i]}$ representa una de las m variables aleatorias independientes, la que está asociada al turno i. Por lo tanto cada $f(U^{[i]})$ es una variable estocástica también independiente de otra $f(U^{[j]})$, pero idénticamente distribuida. El error estándar del estimador insesgado H es

$$\frac{\sigma}{\sqrt{m}} \quad [f19]$$

El primero en aplicar la técnica de Monte Carlo a la planificación de proyectos fue [McGowan 1964], por lo que nosotros sabemos. El mismo corpus de conocimiento sobre gestión de proyectos [PMBOK 1996] recomienda su utilización diciendo:

“Debería utilizarse la simulación del programa en cualquier proyecto grande o complejo, ya que las técnicas de análisis matemático tradicionales como el método del camino crítico (MCC) o la técnica de evaluación y revisión del programa (PERT) no toman en cuenta la convergencia de caminos, y subestiman, por lo tanto, la duración del proyecto”.

La técnica de Monte Carlo, al contrario que el cálculo operacional, nos permitirá enfrentarnos a problemas de tamaño real o medio, a la vez que mantenemos un modelo del proyecto que tiene a nuestro juicio gran potencia expresiva y aplicabilidad.

1.3.2.1 Técnicas de reducción de la varianza

Como se ve en la fórmula anterior, es sencillo reducir el error estándar del estimador en un análisis de Monte Carlo. Basta con aumentar el tamaño de la muestra. No obstante, se pueden emplear técnicas de reducción de la varianza, como la transformación antitética, las variables de control, el muestreo enfatizado (*importance sampling*), el muestreo estratificado o la esperanza condicional.

Para una recopilación de las técnicas de reducción de la varianza consúltense [Wilson 1984] y [Nelson 1987].

1.3.2.1.1 La transformación antitética

La transformación antitética es una subclase de la técnica conocida como inducción de correlación (*correlation induction*). En general, se trata de obtener una correlación negativa respecto de determinado resultado W , que puede ser o no el resultado objetivo Y . W es, por lo tanto, función de los números aleatorios exógenos

$$W = w(U_i \mid i \in I_w) \quad [f20]$$

donde I_w denota los índices ordenados de un subconjunto de los números aleatorios exógenos U . En la transformación antitética se agrupan los números aleatorios en dos series, tomando los de la segunda serie como complementarios de los de la primera. Si k fueran las diferentes rondas de muestreo previstas

$$U_i^{(k/2+j)} = 1 - U_i^{(j)} \quad [f21]$$

Con esta transformación se obtienen dos números aleatorios negativamente correlacionados. Los marginales de la distribución conjunta $[U_i^{(j)}, U_i^{(k/2+j)}]$ son uniformes en (0,1), y por lo tanto la varianza se ve reducida.

1.3.2.1.2 Las variables de control

La variable de control es una variable aleatoria obtenida mediante una función real ξ cuya media se define como:

$$\mu_\xi = E[\xi(\vec{U})] \quad [f22]$$

A partir de esta variable de control obtenemos una nueva variable aleatoria $f^*(U)$

$$f^*(\vec{U}) = f(\vec{U}) - c[\xi(\vec{U}) - \mu_\xi] \quad [f23]$$

Siendo c una constante,

$$E[f^*(\vec{U})] = E[f(\vec{U})] = \Psi \quad [f24]$$

Para estimar la integral Ψ utilizamos un estimador insesgado

$$H^* = \frac{1}{m} \sum_{k=1}^m f^*(\vec{U}^{[k]}) = \frac{1}{m} \sum_{k=1}^m (f(\vec{U}^{[k]}) - c[\xi(\vec{U}^{[k]}) - \mu_\xi]) \quad [f25]$$

Para que este nuevo estimador tenga un error estándar menor que el estimador clásico de Monte Carlo, es preciso que la desviación estándar σ^* de $f^*(U)$ sea menor que la desviación estándar σ de $f(U)$, lo cual se produce cuando hay una correlación alta entre las variables aleatorias $f(U)$ y $\xi(U)$. Precisamente

$$\sigma^* = std[f(\vec{U}) - c[\xi(\vec{U}) - \mu_\xi]] = std[f(\vec{U}) - c\xi(\vec{U})] = \sqrt{\sigma^2 + c^2\sigma_\xi^2 - 2c\sigma\sigma_\xi\rho} \quad [f26]$$

donde ρ es el coeficiente de correlación de $\xi(U)$ con $f(U)$ y σ_ξ es la desviación estándar de $\xi(U)$. Por lo tanto,

$$\sigma^* < \sigma \Leftrightarrow \rho > c \frac{\sigma_\xi}{(2\sigma)} \quad [f27]$$

El valor mínimo de σ^* se encuentra en

$$c = \sigma \frac{\rho}{\sigma_{\xi}} \Leftrightarrow \sigma^* = \sigma \sqrt{1 - \rho^2} \quad [\text{f28}]$$

El método de la variable de control consiste en encontrar una función ξ que se aproxime mucho a f , y para la cual $E[\xi(\vec{U})]$ sea fácil de calcular. A menudo, tanto ρ como σ_{ξ} se desconocen. Se puede usar un análisis de Monte Carlo a parte para estimarlas, o, si ξ se aproxima mucho a f , definir $c = 1$.

1.3.2.1.3 La esperanza condicional

La esperanza condicional (*conditional expectation*) es otra técnica de reducción de la varianza cuyos resultados son muy prometedores. En esta, se utiliza un vector aleatorio \vec{E} tal que su esperanza condicional sea

$$h(x) \equiv E[\vec{Y} | \vec{E} = x] \quad [\text{f29}]$$

donde x representa cualquier posible valor del vector aleatorio \vec{E} . A partir de esta fórmula definimos una nueva variable aleatoria Z

$$Z \equiv h(\vec{E}) \quad [\text{f30}]$$

Esta variable aleatoria forma un nuevo estimador de θ basado en un sólo paso de simulación. Extrapolando sobre una serie de n resultados $\{X_i : i = 1, \dots, n\}$, obtenemos una muestra $\{Z_i = h(X_i) : i = 1, \dots, n\}$. El estimador de esperanza condicional es

$$\hat{\theta}_{EC}(n) \equiv \frac{1}{n} \sum_{i=1}^n Z_i \quad [\text{f31}]$$

La varianza del nuevo estimador es siempre menor o igual que $\text{Var}[Y(n)]$ ya que

$$\sigma_Z^2 \equiv \text{Var}(Z) = \sigma_Y^2 - E[\text{Var}(\vec{Y} | \vec{X})] \quad [\text{f32}]$$

En nuestro prototipo no hemos empleado ninguna técnica de reducción de la varianza, aunque no lo descartamos como línea de actuación futura. La razón está en que hemos dispuesto de suficiente potencia de cálculo para realizar los experimentos planteados con la exactitud que nos interesaba a base de fijar un tamaño de muestra grande.

1.3.3 La programación de la cadena crítica y la gestión de márgenes de protección⁶

La aplicación a la gestión de proyectos de la teoría de las limitaciones se conoce como la programación de la cadena crítica y la gestión de márgenes de protección. Desde la publicación de [Goldratt 1997] esta aplicación ha recibido una atención creciente por parte de la comunidad de especialistas. Otros evangelistas muy influyentes han sido [Newbold 1998] y [Leach 2000].

La cadena crítica se define como el conjunto de actividades que determina la duración del proyecto. Antes del libro de [Goldratt 1997] se definió también como la secuencia crítica [Woodworth; Shanahan 1988]. Para su cálculo se tienen en cuenta tanto las prelacións como las restricciones sobre los recursos. Esto quiere decir que si un recurso está limitado para varias actividades que se planificaron en paralelo, las actividades que no estaban en el camino crítico se añaden a la cadena crítica debido a la limitación en la capacidad del recurso. La duración de las actividades se planifica con un intervalo de confianza del 50%, con el objeto de minimizar la influencia de la Ley de Parkinson, que establece que el trabajo se expande hasta ocupar todo el tiempo disponible. Por la misma razón, no se utilizan ni hitos ni fechas de terminación de las actividades. Tampoco se permite la multitarea. Las actividades se planifican para comenzar lo más tarde posible, y los conflictos sobre los recursos se solucionan adelantando actividades.

La incertidumbre sobre la duración de las actividades se controla mediante la gestión de márgenes de protección. La cadena crítica tiene asociado un margen de protección de proyecto que se coloca al final, y cuya duración viene a ocupar el 50% de la duración de la cadena crítica. Los márgenes de protección de alimentación se colocan donde una línea de actividades se incorpora a la cadena crítica, para proteger a ésta de la variabilidad de las otras cadenas. Existen también unos márgenes de protección de recurso que se utilizan donde un recurso se emplea en una actividad de la cadena crítica, y la actividad anterior en la cadena crítica emplea un recurso distinto, aunque su utilidad consiste en este caso en avisar con anticipación de que el recurso debe estar disponible. Para la programación multiproyecto se utilizan también unos márgenes de protección de cuello de botella para los recursos que actúan como limitadores en todos los proyectos.

1.3.4 La asignación de recursos

La asignación de recursos o carga de recursos es el proceso de asignar recursos, ya sea personal,

⁶ Critical Chain Scheduling and Buffer Management. Otras traducciones de “buffer” son tampón, protección, amortiguador y simplemente buffer

infraestructuras o equipamiento, a los proyectos. El consumo de los recursos determina los costes del proyecto. Sobre ingeniería de costes en el proyecto véase [Ahuja; Walsh 1983]. Generalmente se entiende que los recursos se asignan hasta en el nivel de actividad, lo que lleva implícita la toma de decisiones sobre que actividades se ejecutarán en qué periodo para evitar los conflictos por los recursos.

La planificación de proyectos restringida por los recursos es materia de investigación desde hace más de 40 años. Los trabajos seminales están en [Kelley 1963] y [Wiest 1964]. Para un estudio general del problema, véanse [Arranz 1993], el capítulo 9º de [Ordieres 1999], [Tormos; Lova 1998] o [Klein 1999]. Para un estudio más detallado, véase [Demeulemeester; Herroelen 2002], especialmente los capítulos 6º y 7º, y el capítulo 9º sobre el problema multimodo que se produce cuando las actividades se pueden ejecutar de diversas maneras, cada una con unos requisitos sobre los recursos y una duración distintos.

La planificación restringida por los recursos se ha vuelto más importante desde hace décadas, por los cambios económicos, principalmente la pérdida de beneficio del capital en la actividad industrial que ha llevado a valorar la reducción de costes por encima de la producción y el beneficio total. Paralelamente se ha producido una especialización de la mano de obra, en un proceso de milenios, y un agotamiento de los recursos naturales. Esto obliga a las empresas, y a la postre a la administración también, a trabajar con escasez de recursos, donde son éstos los que limitan en gran medida los proyectos. Además de todo esto, desde [Wickwire et al. 2001] los juzgados estadounidenses exigen programaciones con cargas de recursos explícitas en los análisis de tiempos de los juicios por responsabilidad civil, lo que desde luego lleva a un renovado interés por aplicar estas técnicas en los proyectos de ingeniería.

De los distintos algoritmos ligados a la planificación de proyectos restringida por los recursos, nos hemos centrado en el de asignación de recursos. El problema consiste en encontrar un programa del proyecto realizable dentro de las restricciones de gestión. Estas restricciones consisten en una o varias fechas límites para finalizar el proyecto o alguna de sus actividades, y uno o varios límites en la cantidad de recursos disponibles para realizar las actividades.

El problema de la planificación de proyectos restringida por los recursos, más conocido por sus siglas en inglés *RCPSP*, se clasifica como **m, 1 | cpm | C_{max}** según [Herroelen et al. 1999]. El primer campo en [Herroelen et al. 1999] indica los recursos utilizados, m tipos de recurso, todos renovables (1). El segundo describe las prelacións entre las actividades, fin-inicio sin retraso. El tercero campo señala el criterio de optimalidad, según el cuál se intenta minimizar el tiempo máximo de finalización de las actividades.

Según la clasificación de [Brucker et al. 1999] es un problema **PS | prec | C_{max}**. El primer campo en [Brucker et al. 1999] indica el tipo de problema, programación de proyectos. El segundo describe también las prelacións entre las actividades, señalando en este caso relaciones de precedencia de tipo general. El tercer campo, el criterio de optimalidad, coincide⁷ con [Herroelen et al. 1999].

Los límites en la cantidad de recursos disponibles se suele expresar como

$$\sum_{i \in S(t)} r_{ij} \leq k_j, \quad t=1,2,\dots, f_n \quad R_j \in R \quad [f33]$$

donde S(t) representa el conjunto de actividades en ejecución durante el intervalo (t-1, t]

$$S(t) = i \mid f_i - d_i < t \leq f_i \quad [f34]$$

Si los recursos no son renovables, la limitación se expresa como

$$\sum_t \sum_{i \in S(t)} r_{ij} \leq K_j \quad [f35]$$

En cualquier caso, [Błażewicz et al. 1983] demostraron que el problema es de la clase NP-duro en el sentido fuerte. Es además un problema de difícil aproximación. Los problemas NP-duros son un tipo de problemas de decisión para los cuales existe una reducción en tiempo polinomial de cualquier problema de decisión en NP. Para un estudio de la teoría de la complejidad computacional⁸ veáse el capítulo 10º del mencionado [Brassard; Bratley 1987], [Cook 1971], [Karp 1972], [Aho et al. 1974], [Sahni; Horowitz 1978], [Horowitz; Sahni 1978] y [Hopcroft; Ullman 1979] – [Hopcroft; Motwani; Ullman 2006]. Una autoridad en completitud no polinomial es la monografía [Garey; Johnson 1979].

7 Ambos grupos de investigación se basan en la clasificación $\alpha|\beta|\gamma$ de [Graham et al. 1979] para los problemas de planificación, según la cual α especifica el tipo de maquinaria, β especifica el tipo de trabajo y γ el criterio de optimalidad.

8 El estudio de la complejidad computacional empezó con la demostración de la indecibilidad del problema de parada en [Turing 1936]

Si la disponibilidad de un recurso no fuera constante, la fórmula se puede complicar como

$$\sum_{i \in S(t)} r_{ij} \leq k_j(t) \quad [f36]$$

Normalmente en la práctica profesional de la gestión de proyectos hay que lidiar con ambos tipos de restricciones, restricciones en el tiempo y restricciones en los recursos disponibles. Ante la imposibilidad de respetarlas todas surgen costes de penalización o bonificación respecto de los plazos, y también los costes de sobrecarga o horas extras en las que se incurre cuando no se pueden respetar las restricciones sobre la cantidad de recursos disponibles.

Los algoritmos de asignación de recursos se dividen básicamente en seriales y paralelos, según. Su esquema de generación de secuencias o esquema de secuenciación. Ambos tienen conceptos comunes, como la secuencia parcial, que es el conjunto de actividades que ha logrado programar el algoritmo hasta esta etapa. También comparten el concepto de actividad elegible, que es aquella actividad no programada cuyas predecesoras sí que están en la secuencia parcial.

$$E = \{A_i \notin PS \mid P_i \subseteq PS\} \quad [f37]$$

La asignación en serie es la más rápida y simple. Trata de aumentar en una el número de actividades de la secuencia parcial en cada etapa. Los algoritmos tienen dos fases. Primero ordenan todas las actividades en una lista por algún criterio, y luego pasan a programar las actividades de la lista de una en una. Los criterios de secuenciación pueden ser el orden descendente por fin tardío, orden ascendente por comienzo tardío, orden ascendente por holgura total u otros. La programación trata de planificar las actividades dentro de su holgura respetando la disponibilidad de los recursos y las relaciones de prelación. A veces se intenta minimizar el producto en proceso o se usa otro criterio.

Un algoritmo de programación en serie tiene la siguiente forma:

1. Inicialización: $s_1 = f_1 = 0$; $PS_1 = \{A_1\}$
2. Para $g = 2$ hasta $n-1$ hacer:
 1. Calcular $E_g, F_g, \widetilde{R}_k(t) (k \in K; t \in F_g)$

2. Seleccionar $A_i \in E_g$
3. $ES_i = \max_{h \in P(A_i)} \{f_h\}$
4. $s_i = \min \{ t : t \geq ES_i \wedge t \in F_g \wedge r_i, k \leq \tilde{R}_k(\tau) \forall k \in K, \tau \in [t, t+d_j[\cap F_g \}$
5. $f_j = s_j + d_j$
6. $PS_g = PS_{g-1} \cup A_i$
3. $f_n = \max_{h \in P(A_n)} \{f_h\}$

Donde D_g es el conjunto de actividades elegibles, F_g el conjunto de tiempos de finalización, y $\tilde{R}_k(t)$ la disponibilidad restante del recurso de tipo k en el tiempo t .

La asignación en paralelo utiliza potencialmente todas las actividades en cada ciclo de programación. Los algoritmos tienen tres fases. Primero se utiliza el marco de tiempo que se va a programar. Luego se seleccionan aquellas actividades que aún no se han programado totalmente y que podrían ejecutarse en el marco de tiempo dado. Y finalmente se van eligiendo aquellas actividades cuyos recursos estén disponibles y se van programando hasta que no queden más recursos o más actividades. Los criterios para elegir actividades en la tercera fase pueden ser si estaba programada en el marco anterior, o parcialmente programada, o si ha sido retrasada, o la holgura restante, o la utilización de un determinado recurso crítico, o una propia prioridad de la actividad.

Un algoritmo de programación en paralelo tiene la siguiente forma:

1. Inicialización: $g = 0, f_1 = 0, S_0 = C_0 = \{A_1\}, \tilde{R}_k(0) = R_k, \forall k \in K$
2. Mientras $|S_g \cup C_g| \leq n - 2$ hacer:
 1. $g = g + 1$
 2. $t_g = \min_{A_j \in S_{g-1}} \{f_j\}$
 3. Calcular $C_g, S_g, \tilde{R}_k(t_g), D_g$
 4. Mientras $D_g \neq \emptyset$ hacer:
 1. Seleccionar $A_j \in D_g$
 2. $s_j = t_g$
 3. $f_j = s_j + d_j$

4. Actualizar $\tilde{R}_k(t_g), S_g, D_g$
3. $f_n = \max_{A_h \in P(A_n)} \{f_h\}$

Donde t_g representa el tiempo de secuenciación, C_g el conjunto de las actividades programadas y terminadas en t_g , y S_g el conjunto de las actividades programadas, comenzadas y no terminadas en t_g . D_g representa el conjunto de actividades elegibles en t_g , es decir, todas las actividades que pueden ser programadas respetando las relaciones de precedencia y las restricciones de los recursos.

2. Estado del arte

2.1 Una visión general

Esta investigación se ha plasmado en la construcción de un prototipo de software con el que se demuestra la aplicabilidad de las técnicas desarrolladas, principalmente para muestreo aleatorio y asignación de recursos. En cierto sentido, la investigación trata de saltar la brecha que se ha producido entre la investigación en operaciones de los últimos años y la práctica de la gestión de proyectos. La razón, según [Williams 2003], está en que el mismo cuerpo de conocimiento de la gestión de proyectos se basa en premisas que hoy en día son insuficientes para representar los proyectos modernos. Para una revisión histórica de las técnicas de programación de proyectos en relación con los sistemas de software a partir del sistema RAMPS (*Resource Analysis and MultiProject Scheduling*, 1959/1960), véase [Gordon; Tulip 1997]. Al poco, en [Pollack-Johnson; Liberatore 1998], encontramos una revisión de las capacidades del software comercial de gestión de proyectos. Nuevamente, [Maroto et al. 1999] estudian este tipo de software, especialmente respecto de los algoritmos de planificación restringida por los recursos y planificación del portafolio con recursos compartidos. Destacan en la primera funcionalidad CA-Superproject y Time Line. En la planificación del portafolio, sobresale Primavera Project Planner. Sobre la implementación de algoritmos de planificación restringida por los recursos se puede consultar el estudio de [Kolisch 1999], como veremos en el apartado 2.7, “El problema de la planificación de proyectos con recursos limitados”. La quinta parte de [Turner 1993] está dedicada a los procedimientos de gestión de proyectos y su mecanización en los sistemas de información. Esta obra puede servir de introducción a la gestión de proyectos.

Una revisión histórica de la investigación en la gestión de proyectos entre los años 1960 y 1999 fue realizada por [Kloppenborg; Opfer 2000]. En este vasto trabajo, donde un gran equipo anotó más de 3.500 referencias en inglés para identificar las líneas de investigación más populares, se eligieron también los artículos considerados los mejores, los más útiles. Citaremos aquí los artículos destacados en las áreas de conocimiento sobre las que tiene impacto la presente investigación. Sobre el coste: [Fisher et al. 1995], [Mahler; Mazina 1982], [Benjamin et al. 1990], [Luby et al. 1995] y [Tavakoli; Riachi 1990]. Sobre el riesgo: [Baker et al. 1998], [Mallak; Kurstedt 1997], [Williams 1997], [Couillard 1995] y [Williams 1993]. Sobre el tiempo: [Rasdorf; Abydayyeh 1991], [Hughes; Wilcox 1996], [Starr 1990], [Hughes 1986], [Yates 1992], [Benjamin et al. 1990] nuevamente y [Evans et al. 1993]. Además, en el área de la calidad está [Lawyer 1992].

Existen varios buenos libros de introducción a la programación de proyectos en relación con la gestión del riesgo y la incertidumbre. A partir del trabajo seminal de [Elmaghraby 1977], podemos

recomendar a [Moder et al. 1983], [Baca 1990], [López 1993], [Arranz 1993], [Chapman; Ward 1997], [Tavares 1998], [Brucker 1998] y a [Ordieres 1999]. En el campo de la construcción destaca [Hegazy 2002], y sobre la aplicación del método del camino crítico en la construcción, [Antill; Woodhead 1970].

Específicamente sobre el análisis y gestión del riesgo en el proyecto, el estado del arte en la industria está representado en [Norris et al. 1992], [Simon et al. 1997] y [RAMP 1998].

La corriente ortodoxa dentro de la gestión de proyectos está representada por el [PMBOK 1996]. En este trabajo se define el conjunto de temas, áreas de conocimiento⁹ y procesos que se deben usar conjuntamente con los principios sólidos de la gestión de proyectos para realizar un proyecto. El PMBoK del Project Management Institute ha emprendido exitosamente el camino para convertirse en un estándar [ANSI 2001] [IEEE 2004]. A pesar de ello, esta corriente ha sido criticada duramente, y se han creado nuevas líneas de trabajo. Entre ellas destaca principalmente la aplicación de la teoría de las limitaciones que se conoce como la programación de la cadena crítica [Goldratt 1997]. La describimos en la sección 2.6, “El método de la cadena crítica”, de este capítulo, junto con algunas reacciones.

[Goldratt 1997] no es el único que ha intentado remover los cimientos de la gestión de proyectos. Otras aproximaciones vienen de la teoría general de sistemas, como la de [Kerzner 1998]. [Koskela; Howell 2002] estudian las tres teorías subyacentes a la gestión de proyectos del [PMBOK 1996], que identifican como gestión por planificación, modelo de ejecución y modelo del termostato, y afirman no sólo que no son las mejores teorías de gestión disponibles, sino incluso que sus defectos explican el fracaso de la gestión de proyectos como disciplina. Incluso trabajos más ortodoxos como [Devaux 1999] y [Williams 1992] proponen nuevos indicadores y medidas para una mejor gestión del riesgo. En este ambiente es recomendable para un área de investigación contar con un prototipo propio en el que ir probando las nuevas propuestas que van surgiendo, amén de desarrollar las ideas propias y demostrar su aplicabilidad.

Para una visión general de la investigación en modelos matemáticos para la gestión de proyectos, es muy recomendable la recopilación de [Williams 2003], que abarca tanto modelos deterministas como no deterministas, y en su 5º apartado los modelos estocásticos de redes temporales. El artículo termina clamando por una nueva teoría de gestión de proyectos, reclamación que sostendrán también otros como [Williams 1999a]. Una reseña invitada está disponible en [Tavares 2002], donde se revisan los distintos modelos de proyectos planteados, las aproximaciones a la programación y monitorización de proyectos y el modelado estocástico del riesgo en el proyecto. [Herroelen; Leus 2005b] publican una reseña sobre la planificación de proyectos en condiciones de incertidumbre. Incluyen también referencias a trabajos en planificación de máquinas cuando la

⁹ Ámbito, calidad, tiempo, coste, riesgo, recursos humanos, contratación/aprovisionamiento, comunicaciones e integración del proyecto.

literatura de gestión de proyectos es insuficiente. Distinguen las áreas de planificación estocástica, planificación difusa, planificación proactiva o robusta y análisis de sensibilidad. La primera parte de [Józefowska; Węrglarz 2006] también está dedicada a los nuevos modelos de investigación operativa que ofrecen alternativa a la PERT.

2.2 El análisis de coste

El método de Monte Carlo se viene aplicando hace ya tiempo a la gestión del riesgo en los proyectos. [Raz; Michael 2001] realizan un estudio sobre la contribución de las herramientas de gestión de riesgo a la buena gestión de proyectos. Entre las herramientas evaluadas en el artículo destaca por su buena valoración la simulación. Muchos investigadores, sin embargo, se centran en estudio de una serie de parámetros clave que pueden definir el coste o el beneficio del proyecto. Este análisis, en cierto modo, se realiza antes de la estructura de descomposición del trabajo, o no la necesita. Para este tipo de análisis del riesgo, las prelacións entre las actividades del proyecto son irrelevantes. [Chau 1995], en un estudio sobre los costes en construcción, critica el uso de la distribución triangular y la independencia entre variables y propone una log-triangular y un modelado de la dependencia por coeficientes de correlación. [Wall 1997] recalca que las correlaciones deben ir incluidas en la simulación de Monte Carlo si no se quiere infravalorar el riesgo en los análisis de coste. Demuestra mediante pruebas de χ^2 que la distribución lognormal es mejor que la beta en su industria. También en el campo de los análisis de Coste/Beneficio [Balcombe; Smith 1999] escriben un repaso de las aproximaciones al análisis del riesgo por Monte Carlo. Hacen hincapié en las correlaciones entre variables aleatorias y los ciclos o correlaciones seriales, y explican una aproximación práctica para definir y tratar las matrices de correlación y presentar el análisis de Monte Carlo cuando aparecen estas perturbaciones. [Lorance; Wendling 1999a] y [Lorance; Wendling 1999b] defienden la simulación de Monte Carlo para analizar los costes, y aconsejan sobre el correcto establecimiento de las variables aleatorias y sobre la presentación de los resultados. [Chapman; Ward 2000] presentan un enfoque minimalista para trabajar con la incertidumbre en las rejillas de probabilidad por impacto en las fases iniciales del proyecto, durante el planteamiento o la preparación de ofertas. Se centran en la correcta estimación, que corrigen evitando la influencia¹⁰ optimista, y en la definición del nivel de detalle necesario. Defienden el uso de herramientas de software para el cálculo y la presentación gráfica del riesgo. En la primera vuelta de su procedimiento aconsejan la distribución de probabilidad uniforme, que nosotros definimos en el capítulo 3, “Modelo de datos” sección 3.2, “Modelo de actividad”. También estudian la dependencia perfectamente positiva entre variables aleatorias. Nuestra aproximación al problema, sin embargo, consiste en identificar una a una las variables aleatorias ligadas las duraciones y a los estados de terminación de las actividades del proyecto. Estas variables son consideradas variables independientes, en general, salvo ciertos casos en los que postulamos una independencia condicional. En el capítulo arriba mencionado seguimos la discusión de la

10 *bias*

dependencia.

Otro análisis de Coste/Beneficio se encuentra en [Elkjaer 2000], donde se utilizan las facetas básicas del principio sucesivo para identificar y aislar los factores de riesgo generales. Estos factores llevan asignados distribuciones de probabilidad a partir de una estimación de tres puntos, y posteriormente se incorporan a una simulación de Monte Carlo. De este modo se aseguran la independencia entre las variables aleatorias ligadas a los costes.

2.3 El análisis del riesgo temporal

Nuestra aproximación está más en la línea de integrar el análisis del coste con el análisis del plazo. Partimos de la red de actividades del proyecto, red en la que están dispuestas las relaciones de precedencia. Especialmente, desaprobamos el uso de estimaciones de un solo punto como sustitutas de las distribuciones de probabilidad. [Wendling; Lorange 2000] reclaman una mayor calidad de los modelos durante las estimaciones en cinco clases, y una especial atención al análisis de sensibilidad y a la presentación de resultados de manera congruente para conseguir esta integración. [Elmaghraby 2005] aboga por el correcto tratamiento de las variables aleatorias que definen la incertidumbre en la gestión del proyecto respecto de la duración y el coste. Su propuesta, basada en la programación dinámica¹¹, tiene notables dificultades computacionales, pero merece la pena recoger el guante lanzado a propósito de la simulación de Monte Carlo, donde sugiere basarse en la política óptima. Trataremos esta idea en el apartado 2, “Modelo de actividad”. Ya que planteamos un prototipo de software para la gestión de proyectos, empleamos simulación en vez de cálculo analítico para obtener el mismo dato: la probabilidad de éxito del proyecto. Pensamos que la resolución analítica no es abordable para proyectos de tamaño real.

2.3.1 Las reservas de contingencia

Las reservas de contingencia¹² se constituyen mediante apartados del presupuesto o del tiempo del proyecto que se destinan a hacer frente a las incicencias que no pueden ser previstas con exactitud. Su propósito es tratar con la incertidumbre de modo que la fecha y el coste esperados del proyecto sean más realistas.

En esta otra línea trabajan [Dillon et al. 2005], quienes se plantean el uso óptimo de las reservas de contingencia. Basándose en el análisis de riesgo probabilístico estimado por el valor de opción, despliegan un marco dinámico de elección para la asignación de los recursos en diversos momentos del tiempo. Básicamente, se trata de elegir entre emplear las reservas ahora o en un momento

11 Para un estudio de la programación dinámica, véase el capítulo 18º de [Anderson et al. 2005]

12 *Contingency reserve, management reserve, known unknowns*

posterior. En particular, se puede elegir entre emplearlas en incrementar la robustez técnica o añadir valor.

[Houghton et al. 2001] presentan un armazón de trabajo para calcular, con tres niveles de precisión, las reservas de contingencia en tiempo que se deducen de un análisis de riesgo cualitativo seguido de una estimación cuantitativa especializada ligada a la duración de las actividades. En los niveles dos y tres, realizan una simulación de Monte Carlo con distribuciones de incertidumbre no-normales, y reparten la holgura entre las actividades en forma de reservas de contingencia.

2.3.2 El valor presente neto

Maximizar el valor presente neto es un criterio de gran interés para la financiación del proyecto [Battersby 1964]. El primer algoritmo de resolución de un problema de este tipo se debe a [Russell 1970]. [Herroelen et al. 1997] hicieron una reseña completa de los distintos tipos de problemas de flujo de efectivo descontado a valor presente¹³. [Grinold 1972] demostró que el problema de la planificación de pagos restringida por las relaciones de precedencia, $PS_{\infty} / temp, \bar{d} / \sum c_j^F \beta^{C_j}$ ¹⁴ siguiendo la notación de [Brucker et al. 1999], se podía transformar en un problema de programación lineal¹⁵, que resolvía por el método simplex¹⁶. En [Neumann; Zimmermann 2000a] se presenta un heurístico para el valor presente neto restringido por los recursos, y un procedimiento exacto eficiente para el valor presente neto sin la restricción que es una adaptación del algoritmo de [Grinold 1972]. En otro informe técnico del mismo año, [Neumann; Zimmerman 2000b] discuten una búsqueda de máximo gradiente para el problema con retardos máximos y mínimos. Introducen un esquema de generación que va añadiendo relaciones de precedencia para relajar conflictos, de manera similar a [Demeulemeester; Herroelen 1992]. La búsqueda discutida se aplica en el problema restringido y sin restringir por los recursos, y se define también un método eficaz para truncar la búsqueda. [Schwindt; Zimmermann 2001] presentan un algoritmo ascendente de máxima pendiente para el problema restringido sólo por las relaciones de precedencia. [Vanhoucke et al. 2001] maximizan el valor presente neto para unos flujos de caja que se producen al término de cada actividad, dependiendo de su fecha de terminación. Utilizan una búsqueda recursiva, y tampoco contemplan restricciones en los recursos. Una búsqueda recursiva híbrida muy eficiente para este problema con relaciones de precedencia generalizadas se encuentra en [Vanhoucke 2006].

13 *Discounted Cash Flow*: también llamado flujo de caja descontado, trata del valor del dinero invertido teniendo en cuenta el tiempo que transcurre hasta que se producen los ingresos [Williams 1937].

14 c_j^F representa el flujo de efectivo de la actividad A_j , C_j el tiempo de terminación de la misma actividad, y β^{C_j} el factor de descuento correspondiente

15 Para el estudio de la programación lineal, véanse los capítulos 2º, 3º, 4º, 5º y 8º de [Anderson et al. 2005]

16 Para el estudio del método simplex, véanse los capítulos 6º y 7º de [Anderson et al. 2005]

2.3.3 La puja del proyecto

El análisis del riesgo general del proyecto tiene una importante aplicación directa en la fase de preparación de ofertas. Esta aplicación se conoce como la puja del proyecto en un entorno probabilístico. Desde [Elmaghraby 1990] hasta [Paul; Gutierrez 2005] ha recibido muchas aportaciones. Por ejemplo, [Chapman et al. 2000] presentan un modelo sencillo del problema de manera transparente, ofreciendo una aproximación iterativa para el desarrollo de la puja, y evitando precisiones inapropiadas. [Alquier et al. 2002] describen un sistema de ayuda a la decisión para la gestión y análisis del riesgo durante la primera fase del proyecto. El diseño del sistema prevee una versión probabilística basada en la simulación de Monte Carlo. Incluyen un módulo para la recogida y reutilización de información sobre proyectos y pujas anteriores. Nosotros hemos preferido no hacer un seguimiento estricto del campo y presentar como resultado la probabilidad de éxito del proyecto, en la seguridad de que estos investigadores podrán aprovechar el trabajo realizado en esta investigación.

2.3.4 La simulación de Monte Carlo

Sobre el problema de la simulación de la variable aleatoria ligada a la duración de la actividad, ha trabajado [Dawood 1998]. Define esta variable en base a otras variables aleatorias de los factores de riesgo conocidos en la industria de la construcción, y desarrolla una simulación de Monte Carlo modificada. Para los factores de riesgo utiliza distribuciones uniformes, triangulares y trapezoidales. [Dorp; Duffey 1999] estudian también la dependencia de la duración de la tarea de los factores de riesgo comunes aplicando el modelo de las variables latentes. Desarrollan también método de muestreo especial basado en los métodos de correlación de rangos. [Tavares et al. 1998] desarrollan un modelo de decisión a partir del denominado factor de holgura que emplean en la temporalización del proyecto. Posteriormente simulan las variables aleatorias de la duración de la actividad y de su coste para estimar el riesgo del proyecto. Ambas variables son interdependientes y usan regresión lineal para ligarlas. Siguen pues la estela de [Thomasen; Butterfield 1993], quienes consideraron las incertidumbres ligadas al rendimiento de los recursos.

Alternativamente, [Pontrandolfo 2000] presenta una solución analítica para calcular la duración de un proyecto que consisten en transformaciones de los grafos PERT que se aproxima a la simulación de Monte Carlo. Se basa en las técnicas PERT-estado y PERT-camino. Desafortunadamente, las transformaciones son muy costosas computacionalmente, pero su resolución en proyectos de complejidad media permite un mejor entendimiento de la evolución de los proyectos.

2.3.4.1 Las técnicas de reducción de la varianza

El análisis de Monte Carlo aplicado a las actividades del proyecto, en proyectos de gran tamaño, puede consumir un gran esfuerzo de computación. [Avramidis; Wilson 1993a] han desarrollado una técnica de reducción de la varianza para reducir este esfuerzo que consiste en dividir la muestra en tres grupos para calcular las variables de control. A partir de éstas, estiman la media y varianza de la duración total del proyecto, independientemente de la distribución. Desafortunadamente, su procedimiento no corrige la pérdida de cobertura en casos de anormalidad extrema. En otro trabajo [Avramidis; Wilson 1993b] experimentan con las combinaciones de distintas técnicas de reducción de la varianza para estimar la media de la duración total del proyecto. Las técnicas que intentan integrar son la esperanza condicional, las variables de control y la correlación inducida en sus variantes la transformación antitética y el muestreo de hipercubo latino. La combinación que recomiendan consiste en mezclar la esperanza condicional con el muestreo de hipercubo latino, con la cual consiguen una notable reducción de la varianza. [Avramidis; Wilson 1996] reafirma los mismos resultados aplicados sobre un nuevo ejemplo.

[Shih 2005] aplica las técnicas de muestreo por importancia y muestreo adaptativo para estimar la probabilidad de cumplir el plazo del proyecto en redes de actividades estocásticas. Se basa en conocimiento del dominio, en un subconjunto de actividades cuyo comportamiento define el de la variable de finalización del proyecto.

2.3.5 La red de actividades alternativas

Nuestra aportación se inscribe en el campo de las redes estocásticas aplicadas a la gestión de proyectos, y en particular sigue la estela de la GERT [Pritsker; Happ 1966] [Pritsker; Whitehouse 1966] [Pritsker 1966] [Whitehouse; Pritsker 1969]. En estas redes, los nodos denotan operaciones lógicas, y cada arco dirigido lleva asociada la probabilidad de que sea realizado. Otros parámetros del arco dirigido, como el tiempo requerido para recorrerlo, pueden ser descritos también por una variable aleatoria. En la misma línea está la VERT¹⁷ [Moeller 1972], [Moeller; Digman 1981], un modelo de red de actividades generalizado con la actividad en el arco que también soporta ejecución probabilística. Esta línea de investigación no tuvo mucho éxito debido a su complejidad, pero últimamente ha recibido la aportación de algunos trabajos teóricos como [Golenko-Ginzburg; Blokh 1997]. En el mencionado artículo, se propone un modelo de red de actividades alternativas generalizado (*Generalized Alternative Activity Network, GAAN*), que incluye a la GERT como caso particular. Además de las actividades tipo PERT, se admiten actividades alternativas deterministas (puntos de decisión) y actividades alternativas probabilistas. También se propuso en [Voropajev et al. 1999b] un nuevo tipo de red estocástica alternativa y temporal (*stochastic alternative time-oriented model, SATM*) que permitía un amplio abanico de prelacións entre actividades y nodos

¹⁷ *Venture Evaluation and Review Technique*, utiliza los elementos lógicos de entrada INITIAL, AND, PARTIAL AND (OR con espera) y OR y los elementos lógicos de salida TERMINAL, ALL, MONTE CARLO (activación probabilística de una actividad) y FILTER (restricciones complejas). Pueden consultar un apretado resumen en [Moeller; Digman 1978].

(eventos), e incluía múltiples caminos alternativos deterministas o probabilistas. Establecían una secuencia de puntos de control en los nodos deterministas, y planteaban un estudio formal para encontrar una solución cuasi óptima que consiste en reducir la red a una red generalizada, más clásica, multialternativa. Contamos con una revisión de las publicaciones en torno a las redes de actividades alternativas en los problemas de secuenciación con máquinas en [Neumann 1999]. [Laslo et al. 2005] desarrollan un procedimiento heurístico para monitorizar las redes de actividades alternativas generalizadas de [Golenko-Ginzburg; Blokh 1997] cuando están restringidas por los recursos. El procedimiento incluye simulación para determinar la probabilidad de que las actividades sean críticas.

Con un enfoque más práctico abordaron el problema [Dawson; Dawson 1998], donde definieron una notación específica para mostrar las redes de actividades generalizadas sobre una red PERT. Su trabajo incluía la posibilidad de interrumpir y abandonar una actividad si se completa otra que se esté ejecutando paralelamente. Éste es uno de los pocos trabajos sobre planificación expansiva desde [Demeulemeester; Herroelen 1996].

Sobre una clasificación de los distintos modelos de red de actividades, referimos al lector a [Voropajev et al. 2000], donde explican las redes según los tipos de elementos de la red, los parámetros que acepten y el grado de alternancia.

[Dinic 1990] estudiaron en el caso de relaciones AND/OR con duraciones determinadas positivas la aplicación del algoritmo de Dijkstra para el cálculo de los tiempos tempranos. [Möhring et al. 2000] desarrollan un algoritmo para deducir las relaciones AND/OR implícitas en una red de alternativas. Su aplicación a la programación de proyectos restringida por los recursos consiste en representar los conflictos en los recursos como relaciones de precedencia OR. [Stork 2001] utiliza las políticas preselektivas como procesos de decisión dinámicos en el contexto de la planificación estocástica restringida. Para ello, estudia las relaciones de precedencia AND/OR y desarrolla los algoritmos básicos para el cálculo de los tiempos tempranos y para detectar relaciones de precedencia transitivas. A partir de estos, desarrolla 5 algoritmos de ramificación y poda. Podemos encontrar avances de las distintas políticas en [Stork 2000] y [Möhring; Stork 2000], que citaremos en relación con la programación de proyectos restringida por los recursos. [Möhring et al. 2004] describen nuevamente las relaciones de precedencia AND/OR y desarrollan algoritmos eficientes para varios cálculos básicos sobre variantes del modelo, entre ellos el cálculo de los tiempos tempranos.

[Adelson-Velsky; Levner 2002] describen un algoritmo polinomial para el cálculo de los tiempos tempranos en grafos AND/OR con ciclos de longitud 0. [Levner et al. 2002] se enfrentan a una generalización de las relaciones de precedencia AND/OR que modelan mediante actividades umbral. Una actividad umbral que hayan terminado cierto número de sus predecesoras para comenzar. Presentan un algoritmo tipo Dijkstra para encontrar los tiempos tempranos, incluso en redes con ciclos.

Citaremos en el apartado 2.11, “Otros problemas de planificación” a [Herroelen; Leus 2004], cuyo modelo permite definir actividades con duración normal 0 y duración perturbada definida por una variable aleatoria, lo que a la postre significa actividades con cierta probabilidad de ocurrencia.

2.3.6 La ingeniería concurrente

La línea de trabajo que sí está recibiendo atención es la de la ingeniería concurrente [Syam; Menon 1994]. En ella, la estructura de la información que fluye entre las actividades se independiza del orden de ejecución de éstas. [Yassine et al. 1999] ya presentaban un esquema con árbol de decisión para elegir entre ejecución secuencial, solapada y concurrente donde había riesgo con las repeticiones de actividades. Su modelo incluía también incertidumbre sobre la duración de las actividades.

Anteriormente, [Smith; Eppinger 1997a] analizaron el caso de la iteración secuencial con una matriz de estructura de diseño¹⁸ [Steward 1981] cuantitativa con repetición probabilística que resolvían mediante cadenas de Markov. No obstante, las restricciones del modelo lo hacían de limitada aplicación en la ingeniería de diseño, aunque contiene una extensión para permitir duración variable de las actividades que veremos en el capítulo 3, “Modelo de datos”.

En [Luh et al. 1999] se planteaba un modelo más complejo que describe la repetición de secuencias de actividades dentro de subproyectos. Utilizan la relajación de Lagrange, la programación dinámica estocástica, la optimización ordinal y algunos heurísticos para analizar la planificación. El modelo busca un equilibrio entre fidelidad y complejidad computacional.

[Noelle et al. 2002] identifican los requerimientos de un sistema informático para el soporte de la ingeniería concurrente. [Kao et al. 2002] diseñan un sistema basado en multiagentes que comparten unas pizarras en una intranet con el mismo objetivo. El diseño utiliza redes de Petri¹⁹ [Petri 1962] de alto nivel para representar las operaciones de los agentes.

[Yassine; Braha 2003] defienden el uso de la matriz de estructura de diseño para reducir la complejidad del desarrollo de productos, revelar claramente los flujos de información entre las actividades de sistemas complejos y replanificar los ciclos iterativos. Entre otras ventajas, esta

18 *Design Structure Matrix (DSM)*: Matriz de precedencia que describe los flujos de información entre actividades de ingeniería de diseño

19 Representación matemática de un sistema distribuido discreto. Las redes de Petri son una generalización de la teoría de autómatas que permiten expresar eventos concurrentes. Una red de Petri está formada por lugares, transiciones y arcos dirigidos. Los arcos conectan un lugar a una transición o una transición a un lugar. Los lugares contienen un número cualquiera de fichas. Las transiciones se disparan, es decir, consumen fichas de una posición de inicio y producen fichas en una posición de llegada. Una transición está habilitada si tiene fichas en todas sus posiciones de entrada. Para el estudio de las redes de Petri clásicas véase [Peterson 1977].

técnica redundante en una mejor estimación de los tiempos y costes del diseño. Además, su modelo incluye la dinámica del número de problemas de diseño remanentes. [Ford; Sterman 2003a] presentan un modelo dinámico formal que explora las estructuras de los proyectos de desarrollo en los que se utiliza la ingeniería concurrente para describir, cuantificar y simular el intercambio de información entre los procesos y las decisiones de gestión, fundamentalmente en cuanto a los ciclos de iteración. El modelo no incluye restricciones en los recursos, pero es aplicado exitosamente para describir un caso real.

Más cercano a nuestro modelo está el enfoque de [Hegazy; Kassab 2003], quienes combinan la simulación de diagramas de flujo con un algoritmo genético para encontrar la combinación óptima de recursos en un espacio de actividades multimodo respecto de la ratio coste / producción. Los diagramas de flujo modelan las actividades repetitivas, los ciclos y las alternativas condicionales. [Shen et al. 2005] plantean un complejo modelo para la industria informática con el que establecen repositorios de actividades, de productos y de recursos. Sobre el modelo aplican redes de Petri temporales estocásticas, y estudian con actividades multimodo el plazo de terminación y la calidad del producto final.

2.3.6.1 Los ciclos de repetición

La repetición de actividades puede ser capturada por distintos modelos. [Valls et al. 1998] partieron de actividades de duración determinada que son interrumpibles durante un tiempo aleatorio, tras el cual la actividad prosigue un tiempo extra aleatorio también. Sobre este modelo, plantearon el problema de la planificación con recursos renovables limitados, con el objetivo de minimizar la esperanza del retraso ponderado de las actividades. Para encontrar la solución, utilizaron una búsqueda dispersa.

En la industria del software, [Kusumoto et al. 1997] desarrollan un simulador de proyectos donde las actividades entran en ciclos de repetición hasta alcanzar el equilibrio deseado de calidad y coste. La duración de las actividades es función del contenido de trabajo, de los recursos asignados y del factor de eficiencia en la asignación de recursos. Cada actividad se divide en tres fases que se modelan mediante redes de Petri estocásticas generalizadas, con funciones de ejecución complejas propias del dominio. En la misma industria, [Chang; Christensen 1999] proponen un modelo que captura los ciclos de repetición mediante puntos de decisión al término de algunas actividades, donde se determina si se ha producido un éxito o un fallo. Basado en redes de Petri, el modelo contempla también artefactos, restricciones y actividades abstractas que agrupan otras actividades.

También se han propuesto algunos modelos orientados hacia la industria de la construcción. [Yi et al. 2002] presentan un modelo en red para la secuenciación de este tipo de actividades y la asignación de equipos de trabajo. Utilizan una búsqueda en profundidad para encontrar el mínimo local de la duración de este tipo de proyectos.

2.4 La gestión del portafolio

La gestión del portafolio de proyectos es un proceso de gestión diseñado para ayudar a una organización a adquirir y revisar la información sobre todos o un conjunto orientado de sus proyectos, de modo que pueda priorizar cada proyecto de acuerdo con un criterio común como valor estratégico, impacto o coste.

[Ghomi; Ashjari 2002] apuntan las líneas maestras de una simulación para multiproyectos con duración estocástica de las actividades y un sólo recurso por actividad. Definen el problema como un sistema de colas multicanal. [Fenbert; Fleener 2002] aplican la teoría de las limitaciones y la gestión de la cadena crítica a los multiproyectos, intentando obtener mejores predicciones de la fecha de finalización de los proyectos. Para ello, cuentan con un algoritmo de programación modificado y balancean los recursos críticos entre múltiples proyectos, aunque no dan muchos detalles. Estas aproximaciones, como la nuestra, intentan predecir los resultados de las interacciones entre varios proyectos en su competencia por los recursos. [Artigues et al. 1999] proponen un procedimiento para determinar conjuntos de secuenciaciones de varios proyectos con restricciones en los recursos renovables, sobre el que volveremos más adelante. [Golenko-Ginzburg et al. 2001a], como veremos en el apartado 2.7.1, “Los métodos heurísticos”, plantean la asignación de recursos comunes y especiales entre los proyectos de un portafolio. En [Engwall; Jerbrant 2003] disponemos de un breve estudio sobre los problemas operacionales asociados a la gestión multiproyecto. Define un síndrome de la asignación de recursos en las empresas que se ve aumentado por los fallos de planificación y el compromiso por encima de las expectativas reales. Todo esto lleva a un comportamiento gerencial oportunista. Una aproximación más general a la incertidumbre interactiva externa al proyecto puede verse en [Jensen et al. 2006], en un estudio que incluye la interacción multiproyecto pero también otros tipos de incertidumbres ambientales que sufre la organización del proyecto.

Otro problema relacionado es el de la selección de proyectos en un portafolio. [Sefair; Medaglia 2005] mezclan la selección de los proyectos y su secuenciación en un modelo que incluye variables estocásticas para el flujo de caja, restricciones sobre un recurso (el presupuesto) y relaciones de precedencia entre proyectos. Para el doble objetivo de maximizar el valor neto actual del portafolio, y minimizar el riesgo, entendido como la variabilidad, utilizan la programación mixta de enteros²⁰. Además, añaden la simulación de Monte Carlo para la predicción del flujo de caja futuro.

La programación mixta de enteros es también una de las técnicas que aporta [Hans 2001] para resolver la planificación *grosso modo* de la capacidad en un nivel táctico por encima de los

²⁰ Modelo para problemas de optimización donde las restricciones se pueden expresar mediante relaciones lineales. En la programación mixta de enteros algunas o todas las variables son números enteros. Para una reseña del estado del arte en programación de enteros véase [Wolsey 1998].

algoritmos de planificación de proyectos que veremos a continuación. En su tesis, contempla una capacidad flexible para la carga de los recursos y relaciones de precedencia entre los paquetes de trabajo. Propone varios algoritmos exactos y heurísticos para la planificación de la capacidad en un entorno de portafolio. Sus algoritmos buscan el equilibrio entre los costes de la capacidad extra y los costes de la penalización por retraso.

2.5 La caracterización del riesgo en el proyecto

Otro enfoque planteado para mejorar las capacidades del software de gestión de proyectos, en cuanto a gestión del riesgo, parte de la necesidad de bases de datos de riesgos. En este sentido abogan [Chadbourne 2000], [Schuyler 2000] y [Hillson 2000]. [Mulholland; Christian 1999] han desarrollado un sistema de información propio para la industria de la construcción. [Elkjaer; Felding 1999] definen el bucle de control de la gestión de riesgo en el proyecto, y con él los valores de predictabilidad e influencia para cada riesgo identificado. Plantean un ciclo continuo de identificación, evaluación, respuesta y monitorización del riesgo. [Pritchard 2000] propugna la creación de taxonomías propias mediante el uso de los diagramas de afinidad. Señala que la calidad de la identificación de riesgos depende de la cantidad de identificadores estudiados. En la cuantificación del riesgo, el autor distingue la cuantificación cualitativa como el clásico análisis de los modos de fallo y sus efectos, la cuantificación estadística como el método de Monte Carlo y la cuantitativo-analítica que propone en la línea de desarrollar protocolos o estrategias normalizadas de respuesta al riesgo. Nuestra aproximación no aborda directamente esta problemática, aunque hemos cuidado la exportación de nuestros resultados para poder contribuir a la creación de dichas bases de datos.

El riesgo en el proyecto se puede definir mediante la teoría de la probabilidad. [Schafer 1976] distinguía entre probabilidad aleatoria, la que acarrea situaciones intrínsecamente inciertas, y probabilidad epistémica, la relativa a la creencia o a la falta de conocimiento sobre los procesos. Otra caracterización del riesgo proviene de la teoría de la posibilidad, la cual pretende modelar el conocimiento incompleto mediante conjuntos difusos. A ellos dedicamos el apartado 2.9, “Los conjuntos difusos”.

[Jørgensen 1999] identifica 10 influencias en los modelos de planificación de proyectos que les dificultan tratar con la incertidumbre de los proyectos, y defiende una aproximación desde la programación estocástica y las políticas que permitan una mayor flexibilidad al gestor. [Williams 2000] defiende un modelo sistémico de gestión del riesgo en el proyecto, construido con la ayuda de mapas conceptuales y que incluya tanto la dinámica del sistema como la naturaleza probabilística del riesgo. Clama también por una integración del modelo con la gestión táctica del proyecto, en la línea de los sistemas de ayuda a la decisión. [de Klerk 2001] opta por revitalizar la

teoría de la utilidad esperada²¹ para incorporar la actitud hacia el riesgo en el armazón de gestión del riesgo en el proyecto de la empresa.

El citado [Noelle et al. 2002] describen los indicadores de incertidumbre en las actividades e incertidumbre en los procesos como tipos diferenciados de incertidumbre estructural. Ambos conceptos son abordables con nuestro modelo, como veremos en el capítulo siguiente.

[Fan; Yu 2004] proponen un esquema para incorporar las redes de creencia bayesianas²² a la gestión del riesgo en el proyecto. Su procedimiento intenta decidir la mejor asignación de recursos a cada actividad para minimizar la esperanza de la pérdida económica de proyecto, y tiene en cuenta las pérdidas de eficiencia de los recursos en las actividades.

[Chapman 2006] abre el debate sobre la práctica de la gestión del riesgo en base a probabilidades objetivas, critica el uso de índices de probabilidad por impacto, y recomienda trabajar con probabilidades subjetivas y tener en cuenta las suposiciones y condiciones de estas probabilidades.

2.6 El método de la cadena crítica

Sobre el método de la cadena crítica, [Herroelen; Leus 2001], aun reconociendo los méritos de la programación de la cadena crítica, aseveran que la regla de 50% para el margen de protección de proyecto, conduce a una seria sobrestimación de la duración total del proyecto. El estudio de los principios de la programación de la cadena crítica y la gestión de márgenes de protección se basa en simulaciones por ordenador, para los cuales se ha considerado una distribución lognormal redondeada para la duración de las actividades. Otros aspectos de esta técnica son criticados, como el mantenimiento en serie de la cadena crítica. De todos modos, la recomendación fundamental de estos investigadores es recalcular frecuentemente la planificación básica, y utilizar algoritmos de planificación como el de ramificación y poda. Nosotros empleamos este algoritmo para resolver el problema de la planificación del proyecto restringida por los recursos.

[Raz; Barnes; Dvir 2003], [Raz; Barnes; Dvir 2004] también encuentran que falta evidencia de que se sobrestime la duración de las actividades y de que el trabajo se expanda hasta el tiempo estimado. Tampoco encuentran fundamento para los recortes típicos del 33%, ya que es difícil estimar la propia sobrestimación. Creen que el acortamiento de los plazos debilitará el compromiso de los implicados, y causará mayores sobrestimaciones posteriormente. Al igual que [Herroelen; Leus 2001] demuestran que el método de la cadena crítica no soluciona los problemas causados por

21 La utilidad de una proposición incierta se compara con una apuesta estándar de lotería en el punto de indiferencia, donde el agente, la empresa en este caso, no muestra preferencia por una u otra [von Neumann; Morgenstern 1944].

22 Una red de creencia bayesiana [Heckerman; Wellman 1995] es un grafo acíclico con probabilidades condicionadas asociadas. Los nodos representan variables aleatorias, y los arcos relaciones causales. Añadiendo nodos de decisión y utilidad se puede obtener un diagrama de influencia.

las cadenas no críticas que se vuelven críticas. El uso de los márgenes de protección también presenta contraindicaciones, ya que no se determina qué algoritmos de planificación restringida por los recursos se deberían emplear como base, y no creen que, en caso de competencia, la tasa de penetración en el margen de protección deba ser el criterio fundamental. De todos modos, no menosprecian los beneficios de la técnica, y la recomiendan para organizaciones sin procesos de control y planificación de proyectos que ejecuten un gran número de proyectos similares en un entorno de matriz preocupado por las fechas de entrega. En la misma línea, se expresan en [Herroelen; Leus 2005a], un artículo con ejemplos aclaratorios que citaremos de nuevo. En [Van de Vonder et al. 2006] se compara el método de la cadena crítica con un heurístico desarrollado ex profeso con respecto de la estabilidad de la programación de actividades, y la probabilidad de completar el proyecto a tiempo, en un conjunto amplio de proyectos con distintas características. El heurístico considera las penalizaciones de cada hito, e introduce precedencias extraídas de la red de flujo de recursos para solventar las restricciones en los recursos. Se concluye que las ventajas de CC/BM son difíciles de defender, si el plazo de entrega es realmente importante.

2.7 El problema de la planificación de proyectos con recursos limitados

Sobre el problema de la planificación con recursos limitados, hay un elevado número de contribuciones en la literatura. [Baar et al. 1999] hicieron un exhaustivo repaso de la literatura hasta la fecha, citando más de doscientos artículos. Definieron un esquema de clasificación y notación que luego seguiremos. [Neumann et al. 2000] propone un esquema de clasificación de las secuencias con desfases máximos y mínimos entre actividades, con funciones objetivo regulares e irregulares. Extiende los conceptos de programa activo y semi-activo para estos tipos de desfase. En la monografía de [Neumann et al. 2002] se estudia el problema desde el punto de vista de la función objetivo, definiendo hasta 7 clases de funciones. La segunda parte de [Józefowska; Węrglarz 2006] se dedica a perfilar y evaluar los últimos algoritmos aportados a este problema.

A pesar de tanta investigación realizada, como muestra [Kolisch 1999], los algoritmos empleados en el software comercial de gestión de proyectos dan unos resultados subóptimos. Destacamos de este estudio sobre 7 paquetes de software, que los proyectos se clasifican según varios parámetros, y unos paquetes funcionan mejor que otros frente a ellos. En general, los mejores resultados se conseguían con Primavera Project Planner y CA-SuperProject. Nosotros hemos optado por implementar en nuestro prototipo el algoritmo de resolución óptimo de [Demeulemeester; Herroelen 1992]. En [Demeulemeester; Herroelen 1997] se encuentra un estudio comparativo del rendimiento del algoritmo, y se plantean también algunas mejoras en cuanto a uso de la memoria y un nuevo límite inferior.

Se puede encontrar una descripción formal de las diferencias entre los esquemas de generación del

programa seriales y paralelos en [Kolisch 1995]. En otro trabajo, [Kolisch 1996] demostró que los algoritmos de asignación de recursos en paralelo obtienen el óptimo de las secuencias sin retraso, esto es, aquellas en las que ninguna actividad puede empezar antes sin retrasar alguna otra actividad, aunque sea parcialmente, interrumpiéndola. Las secuencias sin retraso son un subconjunto de las secuencias activas, que son el espacio de búsqueda de los algoritmos de asignación en serie. Una secuencia es activa si no permite que ninguna de sus actividades sean adelantadas (desplazadas hacia la izquierda) sin violar una restricción de precedencia o prelación, ni retrasar ninguna otra actividad. [Sprecher et al. 1995] demostraron que siempre hay una solución óptima que es activa. Sin embargo, no hay garantía de que la secuencia óptima sea sin retraso, por lo que los algoritmos paralelos, aunque eficientes, no aseguran que vayan a encontrar esa secuencia óptima. En [Nübel; Schwindt 1997] se definen las clases de secuencias y su relación con los esquemas de generación para proyectos con relaciones de precedencia generalizadas y funciones objeto regulares e irregulares. A parte de las clases de secuencias mencionadas, se definen las pseudo-semiactivas, semiactivas, pseudo-semiestables, semiestables, estables, voraces, y balanceadas. No obstante, no todos los investigadores siguen esta clasificación cuando hablan de programas estables.

[Schwindt 1998] propone una solución para el problema con lapsos mínimos y máximos entre los comienzos de las actividades, una variante de aplicación en el área de planificación de la producción en plazos cortos. Reduce el espacio de búsqueda del algoritmo de ramificación y poda introduciendo relaciones de precedencia disjuntivas²³ adicionales entre conjuntos factibles máximos (según los recursos) y alternativas de demora mínimas.

[Kolisch; Sprecher 1996] publicaron una biblioteca de redes de actividades de proyectos generados con distintos parámetros para uso de la comunidad de investigadores en los problemas de planificación de proyectos con recursos limitados. La librería, conocida como PSPLIB (*Project Scheduling Problem LIBrary*), incluye problemas multimodo, y además permite a los investigadores aportar sus soluciones a los problemas, óptimas o cuasi óptimas. En [Kolisch et al. 1999] se documentan nuevas instancias de los problemas. El generador que utilizaron para la librería es actualizado y mejorado en [Drexel et al. 2000].

[Herroelen; Leus 2005a] insisten en que el procedimiento empleado para resolver los conflictos de los recursos tiene un importante impacto en la duración planificada para el proyecto. Además, advierten que la cadena crítica no siempre determina la duración del proyecto, y que las secuencias activas no son las más estables. Otro procedimiento de resolución exacto se debe a [Dorndorf et al. 1999], [Dorndorf et al. 2000], quienes usan un algoritmo de ramificación y poda sobre un esquema

23 Las relaciones de precedencia disjuntivas se definen entre conjuntos disjuntos de actividades A y B, y obligan a retrasar el comienzo $\forall j \in B$ hasta que finalice la primera $i \in A$

de generación en serie basado en la propagación de restricciones²⁴ con una prueba de desplazamiento a la izquierda para ceñirse a las secuenciaciones activas. Su algoritmo no usa límites inferiores, y escala bien hasta los problemas de tamaño medio del conjunto J120 de la mencionada [Kolisch et al. 1999].

[Klein; Scholl 2000] presentan un procedimiento de resolución exacto para el problema generalizado de la planificación de proyectos con recursos limitados. El problema generalizado admite fechas de lanzamiento y entrega por actividad, de modo que la actividad sólo se pueda ejecutar en ese intervalo de tiempo, y disponibilidad de los recursos variable. Además, se permite definir relaciones de precedencia de comienzo a comienzo positivas.

[Bowers 2000] define tres nuevas medidas de la holgura para los casos en que distintas asignaciones de los recursos produzcan planificaciones alternativas, ya que incorpora a la red del proyecto los cambios de asignación de recursos en la misma planificación como restricciones comienzo-fin. [Kim; de la Garza 2003] estudian también la holgura y el camino crítico en condiciones de planificación restringida por los recursos, e identifican las holguras reales y los programas alternativos añadiendo relaciones de precedencia para solucionar conflictos por los recursos.

[Brucker; Knust 2000] desarrollan una de las mejores cotas inferiores del problema mediante propagación de restricciones y programación lineal, las cuales aplican de manera destructiva para descartar cotas irrealizables. Su método es aplicable con relaciones de precedencia generalizadas, esto es, con retrasos mínimos y máximos. [Möhring et al. 2003] presentan una aproximación clásica, la relajación de Lagrange, para calcular eficientemente cotas inferiores del problema, y producir secuencias realizables. Utilizan el teorema del corte mínimo sobre un grafo con los pesos del tiempo de comienzo de las actividades. [Demasse 2003] continúa la aproximación de [Brucker; Knust 2000] y presenta una relajación de Lagrange sobre un modelo lineal con propagación de restricciones. Calcula también cotas inferiores del problema y describe un innovador árbol de búsqueda. [Kovács; Váncza 2004] emplean las soluciones parciales completables para evitar el crecimiento del árbol de búsqueda, y poder enfrentarse a problemas de tamaño grande.

[Damay 2005] presenta una formulación lineal mixta del problema de planificación expulsiva que resuelve con un algoritmo de ramificación y poda.

2.7.1 Los métodos heurísticos

[Kumar; Ganesh 1998] desarrollaron un modelo de red de actividades del proyecto basado en redes

²⁴ *Constraint Propagation*, una aproximación basada en el razonamiento lógico formal. Las restricciones se utilizan en un proceso deductivo que permite detectar rápidamente inconsistencias en el dominio de definición de las variables. Es una técnica usada en Inteligencia Artificial [Prosser 1993] y en Investigación Operativa [Esquirol et al. 1995].

de Petri coloreadas en el cual las actividades se representaban mediante transiciones. La asignación de recursos se guía por el criterio de criticidad.

[Lu; Li 2003] plantean un método heurístico (*RACPM, Resource-Activity Critical-Path Method*, método del camino crítico de actividades y recursos) para la asignación de recursos basado en el esfuerzo, con el que además identifican los recursos críticos. Para ello, utilizan un esquema de secuenciación serial. [Kim; de la Garza 2005] discuten un procedimiento para la asignación de recursos en un entorno determinista, el método del camino crítico restringido por los recursos (*RCPM, Resource-constrained Critical Path Method*), según el cual se introducen en el grafo del proyecto, mediante un esquema de generación serial, ciertas prelaciónes entre actividades originadas por las dependencias de los recursos. El procedimiento es subóptimo, aunque es eficiente y respeta el significado de las holguras.

Pese a contar con una solución óptima, el problema de la planificación con recursos limitados continúa atrayendo la atención de muchos investigadores. El problema de la planificación con recursos limitados es NP-duro en el sentido fuerte [Błażewicz et al. 1983], así que las soluciones óptimas sólo son aplicables en instancias de tamaño pequeño. Hace ya un tiempo, gran parte del esfuerzo investigador se concentró en desarrollar algoritmos heurísticos para resolver este conocido problema. Uno de los primeros artículos publicados en esta línea era ya [Kelley 1963]. Por ejemplo, [Bell; Han 1991] presentaron un algoritmo heurístico en dos fases. En la primera fase, resolvían los conjuntos de conflicto mínimos añadiendo arcos de prelación nuevos, y en la segunda fase examinaban estos arcos para estudiar la posibilidad de quitarlos y volver a emplear el algoritmo en la red así producida. Hay una buena reseña de los métodos heurísticos publicados en [Hartmann; Kolisch 2000]. [Abeyasinghe et al. 2001] presentan un heurístico muy eficiente, LINRES, que mejora casi todos los propuestos hasta la fecha. [DePuy; Whitehouse 2000] y nuevamente [DePuy; Whitehouse 2001] aplican el heurístico COMSOAL (Computer Method of Sequencing Operations for Assembly Lines, método computacional para la secuenciación de operaciones en cadenas de montaje), originalmente desarrollado para el problema del equilibrado de cadenas de montaje, a la planificación con recursos limitados. Es un método mixto, heurístico y aleatorio. Utilizan varios esquemas de prioridades para seleccionar la actividad a secuenciar. Su método tiene la ventaja de que en cada iteración produce una programación del proyecto realizable. [Ballestín 2001] desarrolla dos algoritmos heurísticos nuevos para el problema, pero se decanta por la técnica metaheurística de los algoritmos genéticos, como veremos luego.

[Stork 2000] hace un estudio exhaustivo de los algoritmos de ramificación y poda que utilizan políticas preselectivas en general, donde existe alguna regla que decide qué actividad se atrasará en caso de conflicto de recursos. En su experimento con proyectos de tamaño pequeño muestran que las políticas preselectivas lineales tienen un rendimiento mucho mejor que las políticas preselectivas en general. [Möhring; Stork 2000] se centran en el estudio de las políticas preselectivas lineales para resolver los problemas de programación estocástica de proyectos con

recursos limitados, representándolas mediante relaciones de precedencia AND/OR. [Valls et al. 2005] muestran que la doble justificación de las actividades a derecha e izquierda es una técnica sencilla que, sin embargo, ofrece unas significativas mejoras en la calidad de los resultados cuando se usa tanto con algoritmos heurísticos sencillos como con algoritmos genéticos e incluso enfriamiento estocástico²⁵ [Kirkpatrick et al. 1983], [Eglese 1990]. Los detalles sobre la técnica de justificación se pueden obtener en el citado [Ballestín 2001]. Similarmente defendían [Raz; Marshall 1996] la definición de holgura total programada y holgura libre programada, aunque faltara una caracterización clara de la aplicación inversa de los algoritmos heurísticos de planificación con recursos limitados. Esta línea de trabajo, la de la doble justificación, se ha tornado bastante productiva. Veremos alguna aplicación en el campo de los algoritmos genéticos.

[Tormos; Lova 2001] presentan un heurístico muy eficiente, un heurístico aleatorio de método híbrido. Las secuencias se construyen con un método de muestreo aleatorio sesgado basado en la peor elección, y cada secuencia se justifica a derecha e izquierda. La regla de prioridad para el muestreo es LFT, *minimum Latest Finish Time*. Utilizan tanto algoritmo de secuenciación paralelo como serie. Sus resultados son computacionalmente excelentes en problemas de tamaño pequeño.

[Golenko-Ginzburg et al. 2000], [Golenko-Ginzburg et al. 2001a] discuten la asignación de recursos entre los proyectos de un portafolio cuyas actividades tienen duraciones aleatorias. Caracterizan algunos recursos especiales, recursos raros y externos cuya asignación a un proyecto es puntual, en fecha determinada, y tratan de minimizar los costes no operacionales, incluyendo costes de demora y penalizaciones por tener los recursos especiales parados. Utilizan una técnica heurística combinada con una simulación.

2.7.2 El problema de la planificación multimodo de proyectos

La planificación restringida por los recursos es más compleja si la duración de las actividades depende de la cantidad de recursos asignados. Esto se conoce como el problema de la planificación multimodo de proyectos o problema de coste/duración, que se representa como **m,1T | grp,mu | C_{max}** en notación de [Herroelen et al. 1999] o **MPS | temp | C_{max}** en notación de [Brucker et al. 1999]. Si no hay tiempos de desfase, el problema se denota **m,1T | cpm, mu | C_{max}** o **MPS | prec | C_{max}** respectivamente. [Sprecher et al. 1997] extendieron el esquema de enumeración de [Demeulemeester; Herroelen 1992] al problema de planificación multimodo limitada por los recursos. En [Sprecher; Drexl 1998] añadían un criterio de dominio más potente al procedimiento de ramificación y poda. [De Reyck; Herroelen 1999] presentan un heurístico para la asignación de recursos cuando cada actividad tiene varios modos de ejecución con distintas duraciones y necesidades de recursos. Además, su modelo contempla las relaciones de precedencia

²⁵ *Simulated Annealing*, también conocido como temple simulado, es una técnica de optimización según la cual siguen soluciones locales peores con una cierta probabilidad que depende de la temperatura, que va descendiendo.

generalizadas.

[Özdamar 1999] describe un algoritmo genético basado en la codificación de prioridades. [Artigues et al. 1999] proponen un procedimiento basado en prioridades para determinar las secuencias grupales²⁶ y su factibilidad. En [Artigues; Roubellat 2000] presentan una técnica de inserción de actividades que resuelve en tiempo polinomial, entre otros, el problema de la planificación multimodo con recursos renovables. Definen una red de flujo de recursos para identificar cómo es transferido cada recurso de actividad en actividad. [Alcaraz 2001] adapta a este problema un algoritmo genético con un nuevo operador de cruce de dos puntos que desarrolla para la versión monomodo.

[Józefowska et al. 2000] resuelven el problema de planificación de proyectos discreto-continuo, en el que determinado recurso renovable puede ser asignado a una actividad en cualquier cantidad dentro de un intervalo²⁷, cantidad de la que es función continua la duración de la actividad. Para resolverlo, lo transforman en un problema de planificación multimodo mediante discretización, y le aplican un procedimiento a medida de enfriamiento estocástico con el objetivo de minimizar la duración total del proyecto. En [Józefowska et al. 2001] aplican la misma técnica, basándose en la lista de actividades factible en cuanto a relaciones de precedencia, e introduciendo una función de penalización. [Józefowska et al. 2003] presentan un heurístico para el mismo problema, junto con otro heurístico de reparación del programa en los casos en los que cambia la disponibilidad o la demanda de los recursos.

Para el problema general, [Heilmann 2003] presenta un procedimiento de búsqueda exacto, con ramificación y poda. El procedimiento evalúa las decisiones respecto de los modos o los recursos conjuntamente, de modo que elige las decisiones más difíciles cuanto antes en el árbol de decisión.

[Turnquist; Nozick 2003] [Turnquist; Nozick 2004] plantean también la ejecución multimodo de las actividades del proyecto. En su modelo, relacionan la probabilidad de éxito de la actividad con el plazo impuesto y el modo, que depende del número de recursos asignados. Utilizan distribuciones de Weibull para modelar los modos, y calculan analíticamente la probabilidad de éxito del proyecto. Ejecutan también una optimización de la asignación de recursos y plazos.

Un problema similar es abordado por [Tereso et al. 2004], [Tereso et al. 2003], donde la variable aleatoria de la duración de una actividad depende del modo elegido. Mediante una computacionalmente costosa programación dinámica, analizan el modelo para calcular el coste

26 Conjunto de actividades que pueden programarse en cualquier orden sin afectar a la fecha de terminación del proyecto.

27 El recurso continuo es un recurso renovable y disponible en el intervalo [0,1]. De este último depende la tasa de ejecución de la actividad, y su duración a la postre. Los recursos discretos son también renovables y están disponibles en cantidades discretas. La tasa de ejecución de la actividad no depende de la asignación de estos últimos recursos.

asociado a los modos y los costes de retraso. Utilizan distribuciones exponenciales en la duración de las actividades. [Jaśkowski; Sobotka 2004] plantean un problema de planificación multimodo con disponibilidad de los recursos variable. Lo veremos en el apartado 2.7.3.1, “Los algoritmos genéticos”. Del mismo apartado obtenemos una aplicación al problema multimodo con el objetivo de minimizar el coste de [Hegazy; Petzold 2003].

[Ramachandra 2006] plantea una variable aleatoria de distribución exponencial para la cantidad de trabajo de la actividad, y después modela la asignación de recursos como una cadena de Markov de tiempo continuo. Para la resolución del problema, en el que la duración de la actividad depende del modo y este puede reconsiderarse en el transcurso de la misma, perfila una optimización basada en simulación de Monte Carlo con técnicas de reducción de la varianza por muestreo de hipercubo latino y variables antitéticas.

Nosotros no planteamos la ejecución multimodo, pero sí las relaciones entre el resultado de una actividad y su duración.

2.7.2.1 El problema del compromiso entre el tiempo y el coste

Un subproblema de problema multimodo se denomina el problema de coste / duración o problema del compromiso entre el tiempo y el coste²⁸. Para el modelo estocástico, donde las actividades tienen una duración variable antes de tomar ninguna decisión de asignación, publicaron un algoritmo heurístico y una búsqueda exhaustiva [Golenko-Ginzburg; Gonik 1998a]. El problema que ellos resolvían tiene varias simplificaciones, como la necesidad de asignar en cada punto de control todos los recursos disponibles. Para la estimación de la criticidad utilizan el método de la simulación. [Hegazy 1999b] desarrolla un algoritmo genético en el que los cromosomas representan el modo de cada actividad. [Zheng et al. 2002] estudian la idoneidad de las técnicas heurísticas, la programación lineal, la programación de enteros²⁹ y el algoritmo genético de [Hegazy 1999b] para el problema discreto con duraciones determinadas, y aconsejan el empleo de este último.

[Gutjahr et al. 2000] vuelven al modelo estocástico, frente al cual se pueden tomar diversas medidas por actividad, cada una con un coste asociado, para evitar incurrir en costes de penalización. Proponen un algoritmo heurístico para determinar el subconjunto óptimo de medidas (asignaciones discretas de recursos). Las duraciones de las actividades siguen la distribución beta.

[Laslo 2003] define un modelo de la duración y el coste estocásticos de una actividad bajo diversos supuestos, con restricciones de tiempo y coste. Para la estimación de las variables estocásticas

28 *Time-cost trade-off*. El planteamiento para el modelo determinista y sin restricciones en los recursos, corresponde a [Fulkerson 1961]

29 Para una introducción a la programación lineal de enteros, véase el capítulo 8º de [Anderson et al. 2005]

emplea el método de los fractiles [Selvidge 1980], que veremos con detalle en el apartado 3.2.2.1 “La duración probabilística”.

2.7.2.2 El problema del compromiso entre el tiempo y los recursos

Otro subproblema del problema multimodo, intrínsecamente relacionado con el anterior, se denomina el problema del compromiso entre el tiempo y los recursos. Si las elecciones de compromiso son discretas, el problema se representa como $1,1 \mid \text{cpm, disc, mu} \mid C_{\max}$ en notación de [Herroelen et al. 1999]. Para la planificación no expulsiva la primera solución se dio en [Talbot 1982]. [Williams 1999b] describe un modelo de dinámica de sistemas, apoyándose en un caso real de la industria naval, para encontrar el equilibrio óptimo entre adaptar la definición de un proyecto sin aportar más recursos o añadir recursos para cumplir con el programa inicial. [Demeulemeester et al. 2000] presentan un procedimiento de búsqueda en profundidad con ramificación y poda basado en su solución de [Demeulemeester; Herroelen 1992], con unas reglas de dominio nuevas sobre modos redundantes y desplazamiento a la izquierda modificado.

2.7.3 Los métodos metaheurísticos

En general, hay heurísticos mejores y peores, pero no se ha encontrado ninguno que funcione mejor que los demás para todo tipo de problemas. [Davis; Patterson 1975] presentaron una ratio hace tiempo. [Ulusoy; Özdamar 1989], al introducir su heurístico WRUP (*Weighted Resource Utilization ratio and Precedence*, ratio de utilización de recursos y precedencia ponderadas) compararon el rendimiento de distintos heurísticos según el tipo de red de proyecto y las características de los recursos. Desde entonces el posicionamiento respectivo de los heurísticos no ha cambiado mucho.

2.7.3.1 Los algoritmos genéticos

Debido a las dificultades computacionales de las soluciones exactas al problema de la planificación de proyectos con recursos limitados, y a que no se ha encontrado un algoritmo heurístico general, que funcione bien para todo tipo de proyectos, sigue habiendo interés en desarrollar metaheurísticos como los algoritmos genéticos³⁰ [Holland 1975], que buscan la mejor solución en cada caso. Los algoritmos genéticos se llevan aplicando a la planificación restringida por los recursos desde [Davis 1985]. Uno de los artículos más influyentes está en [Hartmann 1998], donde se estableció la representación en permutación de precedencia realizable para el genotipo, lo que luego se conoció

³⁰ *Genetic Algorithms* son algoritmos de búsqueda que codifican la información de cada solución en una cadena que luego alteran y cruzan con otras soluciones para desarrollar una nueva generación de soluciones. Véase una introducción a esta técnica en [Mitchell 1996].

como lista de actividades. La producción del fenotipo se realizaba por esquema serial. [Hegazy 1999a] proponía algunas mejoras en los heurísticos de la planificación restringida por los recursos y la nivelación, de la que hablaremos luego. Usaba algoritmos genéticos de búsqueda cuasi óptima para resolverlos simultáneamente. Para la asignación de recursos asignaba prioridades a las tareas, mientras que usaba dobles momentos para la nivelación.

[Wang; Lu 2002] utilizan un algoritmo genético para establecer las prioridades de cada actividad ante el algoritmo de planificación restringida por los recursos. Definen también los parámetros adecuados para proyectos de tamaño práctico. En un trabajo similar, [Leu; Hung 2002] combinan la simulación de una red de proyecto estocástica con un algoritmo genético para buscar soluciones cuasi óptimas al problema de la planificación restringida por los recursos.

[Hindi et al. 2005] diseñan un algoritmo evolutivo para el problema monomodo que ofrece buenos resultados sobre dos juegos de problemas, tanto en alcanzar el óptimo conocido, como en minimizar el error medio. [Hartmann 2002] presenta otro algoritmo genético donde añade un gen más a la lista de actividades que indica si la producción del fenotipo se realiza por esquema serial o paralelo.

[Valls et al. 2002] presentan un algoritmo genético híbrido con un operador de cruce diseñado especialmente para el problema, el cruce de pico, y utilizan un operador de doble justificación como optimizador local. El rendimiento de algoritmo genético híbrido es sobresaliente para los problemas medianos (PSPLIB J120). Los detalles del algoritmo HGA (*Hybrid Genetic Algorithm*) están en el citado [Ballestín 2001].

[Alcaraz; Maroto 2001] presentan un algoritmo genético con codificación de la lista de actividades, y con un gen más que el algoritmo de [Hartmann 1998]. Distinguen si la lista de actividades se secuencia hacia adelante o hacia atrás, e incluyen un amplio estudio sobre diversas variantes de población, operador de cruce y otros parámetros. Los detalles están en el citado [Alcaraz 2001]. [Kochetov; Stolyar 2003] presentan también un algoritmo genético híbrido con búsqueda local GRASP³¹ [Feo; Resende 1989] y lista tabú³² [Glover 1989], [Glover 1990]. [Senouci; Eldin 2004] utilizan también un algoritmo genético híbrido sobre un modelo con distintas relaciones de precedencia, distintos modos en la actividad, y compromisos entre el tiempo y el coste total del proyecto para encontrar soluciones cuasi óptimas a los problemas de nivelado, planificación

31 *Greedy Randomized Adaptive Search Procedure*, procedimiento de búsqueda adaptativa aleatorizada voraz. Consta de dos fases. En la fase de construcción, se evalúan los elementos y se elige el mejor con cierta probabilidad. La probabilidad de elección depende de los valores evaluados. Una vez escogido el mejor, los valores de los otros elementos se adaptan, y se itera por todos los elementos. En la fase de mejora, se utiliza una búsqueda local. Ambas fases se repiten varias veces.

32 *Tabu Search*, técnica de optimización que permite empeorar en contra de la mejor solución del entorno local, pero que utiliza una lista de movimientos prohibidos para evitar ciclos. A esta técnica dedicamos el apartado 2.7.3.3 “La búsqueda tabú”. Para un estudio de la técnica, puede consultarse la monografía [Glover; Laguna 1997].

restringida por los recursos y minimización del coste total del proyecto.

[Debels; Vanhoucke 2005] utilizan un algoritmo genético especial inspirado en la doble justificación con el que consiguen los mejores resultados sobre conjuntos de actividades medianos (PSPLIB J60 y J120). [Jaśkowski; Sobotka 2004] se enfrentan, por el contrario, a un problema de asignación de recursos modificado, donde las actividades tienen distintos modos y la disponibilidad de los recursos es variable. Para resolverlo, plantean un algoritmo genético donde los modos y prioridades de las actividades son codificados en los cromosomas. [Hegazy; Petzold 2003] emplean un algoritmo genético para la optimización del coste también con actividades multimodo en un entorno dinámico de seguimiento del proyecto.

[Hegazy et al. 2004] presentan un algoritmo genético para una variante del problema de asignación de recursos. Su modelo de programación distribuida describe el trabajo concurrente en varias localizaciones, y tiene en cuenta la interrupción de actividades, los costes económicos y temporales de la relocalización de los recursos y productividad ligada al emplazamiento.

[Zhuang; Yassine 2004] emplean un algoritmo genético para el conocido problema de la planificación restringida por los recursos minimizando la duración, pero en un entorno multiproyecto. Parten de la representación en matriz de estructura de diseño de los proyectos, y apuntan el tratamiento para la realimentación estocástica, con la que modelan la probabilidad de que determinada actividad deba repetirse.

2.7.3.2 La técnica de las hormigas

La técnica de las hormigas³³ [Dorigo; Di Caro 1999] ha sido aplicada con éxito por [Bautista; Pereira 2002]. La mejor hormiga de cada subcolonia deja una pista (feromona) entre pares de actividades correlativas o entre una actividad y su posición en la lista. [Merkle et al. 2000] también consiguen unos buenos resultados con un método que combina la evaluación de dos feromonas. Posteriormente [Merkle et al. 2001], [Merkle et al. 2002] ajustan el algoritmo cambiando la influencia del heurístico en las decisiones de las hormigas, y añaden una vista global mediante otra feromona que produce un efecto similar a la técnica del apartado 2.7.3.4 “La optimización por enjambre de partículas”. Los resultados de su técnica en el conjunto de problemas de PSPLIB son muy buenos.

2.7.3.3 La búsqueda tabú

³³ *Ant Colony Optimization*, un metaheurístico en el que generaciones de hormigas artificiales buscan soluciones a través de decisiones probabilísticas. Las que encuentran buenas soluciones marcan el camino recorrido para atraer a la siguiente generación de exploradoras.

A parte de los algoritmos genéticos y la técnica de las hormigas, se han utilizado otros metaheurísticos para resolver el dispar rendimiento de los heurísticos disponibles. La búsqueda tabú de [Glover 1989] y [Glover 1990] es una técnica de optimización que utiliza ciertas estructuras de memoria para modificar la búsqueda en la vecindad de una solución. La estructura más sencilla es una lista de las soluciones visitadas recientemente.

[Tsai; Gemmill 1998] aplicaron la búsqueda tabú al problema de la planificación restringida por los recursos, en sus variantes determinada y estocástica. En la variante estocástica modelaban la duración de las actividades mediante distribuciones beta calculadas a partir de tres puntos. Utilizaban simulación para calcular la duración del proyecto. En ambas variantes utilizan el heurístico de mínima holgura.

[Viana; de Sousa 2000] estudian los metaheurísticos multiobjetivo con el objeto de encontrar buenas aproximaciones para el conjunto de soluciones no dominadas. En particular aplican el enfriamiento estocástico de Pareto³⁴ y la búsqueda tabú para tres objetivos distintos en algunos problemas de tamaño medio. En la misma línea, [Al-Fawzan; Haouari 2005] presentan un algoritmo de búsqueda con tabú que se aproxima al conjunto de soluciones no dominadas, con el doble objetivo de minimizar la duración total y maximizar la robustez del programa. Más adelante, en el apartado 3.6, “Exégesis de los resultados”, volveremos sobre la robustez de los programas. Estos investigadores miden la robustez mediante la suma de las holguras libres. Aunque las duraciones de las actividades sean determinadas, la robustez permite elegir aquellas planificaciones menos sensibles a los retrasos o a la irrupción de trabajos no previstos.

2.7.3.4 La optimización por enjambre de partículas

[Zhang et al. 2006] introducen el uso de la optimización por enjambre de partículas³⁵ [Kennedy; Eberhart 1995] en el problema de la planificación restringida por los recursos mediante un espacio de búsqueda N-dimensional, donde cada dimensión representa la prioridad de cada actividad.

2.8 Otros problemas de asignación de recursos

La asignación de recursos es el problema que se plantearon [Voropajev et al. 1999a] con su modelo *RGNM* (*Random Generalized Network Model*) en el que, además de duración aleatoria de las actividades, se permitían nuevas relaciones de precedencia. Sobre este modelo explicaban el

34 *Pareto Simulated Annealing (PSA)*, técnica heurística de optimización multi-objetivo que se basa en el método de enfriamiento estocástico para generar en un tiempo corto un conjunto de soluciones que forme una buena aproximación al conjunto de soluciones eficientes (según la eficiencia de Pareto [Pareto 1896], [Pareto 1897])

35 *Particle Swarm Optimization (PSO)*, técnica de optimización que se inspira en el comportamiento de las bandadas de aves buscando alimento. Combina los óptimos local y general en el cálculo de la siguiente posición.

problema: la determinación de un calendario o programa de disponibilidad de algunos recursos escasos o muy caros, con el objetivo de minimizar el coste medio. Aunque no dieran muchos detalles, en la solución utilizaron tanto la simulación como la programación no-lineal.

Un problema derivado de la planificación restringida por los recursos con actividades multimodo se produce cuando la intensidad de la actividad es variable durante su ejecución. Es un problema complejo que ya plantearon [Leachman et al. 1990]. En intervalos discretos, se puede, por lo tanto, cambiar el consumo de recursos afectando a la duración de la propia actividad. [Kis 2005] desarrolla un algoritmo eficiente de ramificación y corte para este problema.

2.8.1 La nivelación de recursos

Otro problema de asignación de recursos, en la línea de integrar coste y tiempo, es la nivelación de recursos. El objetivo que hay que minimizar representa de algún modo la variación en la utilización de los recursos en el tiempo. Se cuenta con una solución óptima desde [Bandelloni et al. 1994]. [Savin et al. 1996] [Savin et al. 1998] desarrollaron una red neuronal³⁶ [MacCulloch; Pitts 1943] [Rosenblatt 1958] de tipo retroalimentada [Hebb 1949] [Hopfield 1982] que cargaban inicialmente con los tiempos tempranos de las actividades. [Kartam; Tongthong 1998] crearon una red neuronal específica, mezclando los tipos Hopfield y competitiva [Kohonen 1982], y consiguieron mejores resultados que con los heurísticos disponibles en la fecha.

A este mismo problema aplicaban [Son; Skibniewski 1999] un optimizador local multiheurístico cuyos resultados mejoraban con enfriamiento estocástico. Ya hemos citado el trabajo con algoritmos genéticos y conjuntos difusos de [Leu et al. 1999]. En el citado [Neumann; Zimmermann 2000], se especifican procedimientos exactos y heurísticos polinomiales en el problema de nivelación de recursos con y sin restricciones en los recursos, con distintos objetivos.

[Jeetendra et al. 2000] utilizan redes de Petri estocásticas coloreadas extendidas con una dinámica propia basada en la matriz de precedencia para modelar la red del proyecto, y describen un algoritmo heurístico de nivelación de recursos sobre su modelo.

[Hiyassat 2000] modifica el método heurístico de nivelación de los momentos [Harris 1978] cambiando el criterio de selección de la actividad desplazable por aquella con mayor producto tasa de recurso por holgura. En [Hiyassat 2001] aplica el nuevo heurístico a la nivelación de múltiples recursos, obteniendo unos resultados que en algunos casos mejoran aquellos de [Harris 1978]. [Song et al. 2002] aplican una red neuronal para elegir las actividades que se deben desplazar para

³⁶ *Artificial Neural Network*, un paradigma de aprendizaje y de procesamiento automático distribuido inspirado en el sistema nervioso animal al que dedicamos el apartado 2.10 “Las redes neuronales”. Para una introducción a las redes neuronales, véase [Gurney 1997].

nivelar el programa. [Zheng et al. 2003] utilizan un algoritmo genético multiobjetivo para la nivelación de dos recursos, mediante pesos adaptativos.

2.9 Los conjuntos difusos

Algunos investigadores han encontrado pertinente la teoría de conjuntos difusos³⁷ [Zadeh 1965] para modelar la incertidumbre ligada a la planificación de proyectos. Es una aproximación apropiada cuando falta la información histórica o es irrelevante para el cálculo probabilístico. Permite manejar la información ambigua suministrada por los expertos mediante funciones de pertenencia. Las funciones de pertenencia se basan en la teoría de la posibilidad, por contraste con las distribuciones que hemos visto hasta ahora, y que se basan en la teoría de la probabilidad.

La primera aplicación de la teoría en el campo de la planificación de proyectos se debe a [Prade 1979]. No obstante, aún queda camino por recorrer. Parte del creciente interés en programación difusa se encuentra recogido en [Słowiński; Hapke 2000]. Un defensor de la teoría, [Pender 2001] critica la base probabilística de la gestión de riesgo al uso, y defiende un nuevo armazón de conocimiento incompleto donde tengan cabida los conjuntos difusos o las opciones reales. [Dubois et al. 2003] presentan una reseña de la aplicación de los conjuntos difusos y la teoría de la posibilidad en planificación, y distinguen dos áreas principales: la relajación de restricciones y la utilización del conocimiento incompleto.

[Hapke et al. 1994] aplicaron los números difusos L-R a los algoritmos de planificación restringida por los recursos basados en reglas de prioridad, modelando la incertidumbre sobre la duración, la fecha preferible y la fecha tope de finalización de las actividades. Sobre un modelo, [Wang 1999] utilizó un algoritmo paralelo de secuenciación para maximizar la satisfacción de restricciones temporales difusas, en este caso, las fechas preferibles y tope de entrega del proyecto.

[Leu et al. 1999] utilizaron conjuntos difusos y algoritmos genéticos para resolver el problema de nivelación cuando hay incertidumbre en la duración de las actividades. Posteriormente, [Leu et al. 2001] aplican la misma combinación de técnicas para buscar soluciones al problema del compromiso entre el tiempo y el coste ante distintos perfiles de aceptación del riesgo. Los conjuntos difusos definen la incertidumbre sobre la duración de las actividades, de la cual se deriva el coste.

[Özdamar; Alanya 2001] plantean un armazón matemático en el cual modelan la incertidumbre de las macroactividades tanto en cuanto a su duración como a sus relaciones de precedencia de comienzo a comienzo. Utilizan números difusos de seis puntos y cuatro heurísticos basados en

³⁷ *Fuzzy sets*, extensión de la teoría de conjuntos que se utiliza en lógica difusa. Utiliza funciones de pertenencia de los elementos a los conjuntos de la forma $\mu \rightarrow [0, 1]$

prioridades que contemplan varios modos de ejecución de las actividades.

[Pan et al. 2001] emplean también los números difusos para modelar la duración de las actividades, y aplican un algoritmo paralelo de planificación restringida por los recursos, para el que definen los equivalentes de 10 reglas de prioridad famosas³⁸. [Liberatore 2002] aplica también la función de pertenencia difusa a la duración de las actividades. Intenta una aproximación computacional directa al análisis del camino crítico aplicando el principio de extensión de Zadeh³⁹. Define la aritmética del cálculo de los tiempos tempranos, pero la solución de los tiempos tardíos se muestra inconsistente. Es un problema abierto desde [Buckley 1989], quien advirtió la dificultad de computar las funciones de los tiempos tardíos y las holguras. [Liberatore 2002] obtiene otros datos como la criticidad mediante un procedimiento de enumeración. [Slyeptsov; Tyshchuk 2003] aplican también el principio de extensión al método del camino crítico y consiguen calcular no sólo los tiempos tempranos, sino también las holguras mediante un grafo de descomposición.

[Pan; Yeh 2003] muestran en un artículo bien trabado el uso del enfriamiento estocástico con un algoritmo paralelo difuso de planificación restringida por los recursos. También ellos utilizan una función de pertenencia difusa para la duración de las actividades. Un caso particular de uso de los conjuntos difusos se encuentra en [Kim et al. 2003] donde se presenta un algoritmo genético híbrido que utiliza un controlador de lógica difusa para regular las tasas de cruce y mutación.

[Wang 2002] utiliza una búsqueda en abanico con el objetivo de minimizar el riesgo del proyecto, definido como la combinación de los criterios optimista y pesimista de la teoría de la posibilidad respecto de la duración difusa del proyecto. En [Wang 2004] incluye un algoritmo genético sobre el criterio pesimista con la intención de producir un programa robusto que ofrezca el mejor comportamiento en el peor caso.

2.10 Las redes neuronales

Las redes neuronales forman un paradigma de computación paralela basado en la neurona artificial. Esta neuronal está compuesta de múltiples entradas, una salida y unas funciones de propagación, activación y transferencia que definen su comportamiento. El paradigma ha encontrado muchas y variadas aplicaciones en la ingeniería [Haykin 1994].

No obstante, pocos trabajos se centran en la aplicación de las redes neuronales al problema de la planificación. Hemos citado en el apartado 2.8.1 “La nivelación de recursos” las redes neuronales

38 Mínimo comienzo temprano, mínimo final temprano, mínimo comienzo tardío, mínima holgura (MINSLK), máxima demanda de recursos, mínima duración, máxima duración, mayor número de sucesoras directas y menor número de sucesoras directas

39 Sea * una operación aritmética básica, la función de pertenencia $\mu_{\tilde{A} * \tilde{B}}(z) = \max_{x * y = z} \{ \min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)) \}$

propuestas por [Kartam; Tongthong 1998], [Savin et al. 1996] y [Savin et al. 1998]. Podemos citar también a [Senouci; Adeli 2001], que utilizan una dinámica neuronal basada en una patente de invención de [Adeli; Park 1998] para planificar los recursos y optimizar el coste sobre un modelo matemático con distintas relaciones de precedencia, distintos modos en la actividad y compromisos entre el tiempo y el coste total del proyecto.

2.11 Otros problemas de planificación

2.11.1 El problema de los puntos de control

El problema del control también ha atraído la atención de varios investigadores. [Eisner 1962] introdujo el concepto de caja de decisión, un punto en el programa que incluye arcos de salida alternativos deterministas o aleatorios. [Golenko-Ginzburg; Gonik 1997] desarrollaron un modelo para una red temporal estocástica restringida por los recursos, y plantearon un sistema de decisión dinámico de los tiempos de comienzo de las actividades utilizando programación 0-1. Las duraciones de las actividades vienen definidas por variables aleatorias (betas, uniformes o normales) para las que existen cotas superiores e inferiores. Utilizan simulación para determinar la probabilidad de que la actividad sea crítica. Estos mismos investigadores continúan en [Golenko-Ginzburg; Gonik 1998b], y estudian un algoritmo heurístico eficiente para determinar los puntos de control en un proyecto con duración aleatoria de las actividades. En estos puntos, se puede alterar la velocidad de ejecución de las actividades, y el algoritmo intenta determinar el siguiente punto de control sin que las acciones correctivas lleguen demasiado tarde.

[de Falco; Macchiaroli 1998] propusieron guiar la programación de las actividades de control por el criterio de producto en proceso⁴⁰, pero la utilidad de este criterio es puesta en duda por [Herroelen; Leus 2001]. [Bushuyev; Sochnev 1999] propusieron aplicar una medida de la entropía, a partir de la teoría de la información de [Shannon 1948], para controlar la ejecución del proyecto. En su aproximación cualitativa la entropía se utiliza como un recurso más, y sirve como medida única sobre el presupuesto, el programa, la reserva de gestión y la calidad.

[Zafra-Cabeza et al. 2005] proponen una aplicación del control predictivo basado en modelos sobre un sistema estocástico de gestión del riesgo que citaremos, por su interés, en el apartado 2.12, “Nuestra visión”.

Recordemos ahora que [Hegazy; Petzold 2003], citados en el apartado 2.7.3.1, “Los algoritmos genéticos”, planteaban un entorno de seguimiento dinámico del proyecto para la optimización del coste con actividades multimodo.

40 *Work in progress*

2.11.2 La reparación del programa

La reparación de un programa es un problema de replanificación, donde ante incidencias como nuevas actividades para secuenciar o cambios en la disponibilidad de los recursos por averías u otras causas, se intenta producir un nuevo programa que sea computable en poco tiempo, y lo que es más importante, contenga pocos cambios respecto del programa original.

[Briand et al. 2002] definen un heurístico para proponer programas flexibles en un entorno de producción bajo pedido multimodo y con tiempos de retardo mínimos y máximos entre tareas. Se basan en el concepto de secuencia de grupos, donde un grupo es un conjunto de tareas que pueden ser secuenciadas en cualquier orden sin afectar a los tiempos de finalización de los trabajos.

[Elkhyari et al. 2002] desarrollan un entorno dinámico basado en explicaciones para añadir o eliminar restricciones del problema de planificación restringida por los recursos. La propagación de estas restricciones produce un nuevo programa similar al anterior. Los resultados computacionales no son concluyentes [Elkhyari et al. 2003].

Los citados [Herroelen; Leus 2004] trabajan con programación lineal para producir una secuenciación del proyecto que sea estable ante una perturbación que se producirá en alguna actividad, y que alterará la duración de ésta. Su objetivo es minimizar la esperanza de la desviación ponderada de los tiempos de comienzo de las actividades.

[Wang 2005] compara dos soluciones metaheurísticas, mediante un algoritmo genético y mediante enfriamiento estocástico, para el problema de la reparación de un programa de actividades. Los eventos que ocasionan la violación de relaciones de precedencia o de las limitaciones de capacidad de los recursos pueden ser adelantos de la fecha de entrega, nuevas relaciones de precedencia descubiertas a posteriori, cambios en la capacidad de los recursos en ciertos intervalos de tiempo o errores en la estimación de la duración de las actividades. Estos eventos disruptivos son introducidos como nuevas restricciones dinámicas, y los metaheurísticos realizan una búsqueda intentando minimizar determinada función de coste ligada al consumo de los recursos. Su recomendación es emplear principalmente el algoritmo genético, y utilizar la búsqueda por enfriamiento estocástico cuando se precise un solución mejor. Hemos citado ya la técnica de reparación heurística de [Józefowska et al. 2003], para los casos en los que cambia la disponibilidad o la demanda de los recursos.

[Bidot 2005] propone una clasificación de los métodos de resolución para secuenciación de actividades en condiciones de incertidumbre basada en cómo y cuándo se toman las decisiones. El modelo teórico representa las relaciones entre la generación del programa y su ejecución y control, e introduce conceptos como las soluciones parciales y los programas flexibles.

2.11.3 El problema del tiempo medio

Cuando los proyectos incluyen entregas parciales que generan ingresos, económicamente es interesante minimizar no la duración total del proyecto, sino el tiempo medio de finalización de las actividades. [Hall et al. 1996] muestra un algoritmo de aproximación en tiempo polinomial para el problema de una sólo máquina con relaciones de precedencia.

2.12 Nuestra visión: redes temporales estocásticas de actividades con alternativas

Nuestra aproximación, en lo que atañe a la multialternativa, es muy similar a la de [Liberatore; Pollack-Johnson 2003] y [Pollack-Johnson; Liberatore 2005]. Ellos extienden las capacidades de GERT para incluir distintas duraciones de las actividades repetidas en un ciclo, y los casos de dependencia entre las duraciones de distintas actividades. No obstante, prefieren una aproximación de nivel global al problema, aproximación que llaman de escenarios, estudiando las ramas probabilistas y los ciclos. Nosotros nos atenemos a la aproximación clásica a nivel de actividad, pero obtenemos los mismos resultados globales por agregación. Lo demostraremos en el capítulo 3, “Modelo de datos” sección 3.6, “Exégesis de los resultados”. Pollack-Johnson y Liberatore utilizan cálculos detallados con software de ofimática para resolver su modelo. Las duraciones de las actividades en su modelo son discretas.

Hay aproximaciones similares a la nuestra para al problema de la gestión del riesgo en el proyecto que utilizan redes de Petri temporales⁴¹. Por ejemplo, [Sawhney 1997] proponía un completo modelo mediante redes de Petri t-temporales⁴² donde las actividades se modelaban mediante transiciones, y se tenía en cuenta el resultado positivo o negativo de la actividad mediante arcos probabilísticos. También contemplaba los ciclos en la red. Proponía asignar distribuciones estadísticas para el tiempo o el coste de actividades o grupos de actividades. Los grupos de actividades se descomponían en un jerarquía de redes de Petri. Utilizaba fichas distintas en una red de Petri coloreada para la unidad de control y los distintos tipos de recursos. Para la asignación de recursos creaba lugares de fusión⁴³.

[Zafra-Cabeza et al. 2004a], [Zafra-Cabeza et al. 2004b], [Zafra-Cabeza et al. 2005] proponen un

41 Las redes de Petri temporales se definieron en [Chretienne 1983]. Consisten en una red de Petri a la que se añade una función de duración del conjunto de transiciones en \mathbb{N}^* .

42 Las redes de Petri t-temporales fueron introducidas por [Ramachandani 1974]. En ellas cada transición lleva asociado un tiempo de disparo.

43 Los lugares de fusión, propuestos por [MSC 1993] y [Jensen 1994] son lugares igualados a otros, de modo que contengan el mismo número y tipo de fichas. En este artículo sirven para controlar la disponibilidad de los recursos.

modelo del programa del proyecto con el objeto de gestionar el riesgo, como nosotros, aunque con métodos distintos. Ellos plantean las actividades de mitigación del riesgo asociadas a uno o varios factores de riesgo, los cuales a su vez influyen en una o varias actividades de la red del proyecto. La red del proyecto se modela mediante una red de Petri p-temporal⁴⁴ sin alternativas ni ciclos, y el problema de optimización estocástico es resuelto mediante programación lineal mixta de enteros.

En el campo de la ingeniería del software, contamos con una aportación similar a la nuestra por parte de [Coronado; Jaén 2002]. En su modelo, contemplan tanto las redes probabilísticas del proyecto como los ciclos para rehacer el trabajo. Lo que nosotros estudiamos como probabilidad de éxito ellos lo abordan desde el punto de vista de la probabilidad de fallo, que va ligada a un esfuerzo y duración suplementarias. Aunque no dan detalles de los procedimientos de resolución, sí que indican que utilizan la simulación de Monte Carlo para la estimación.

44 Al contrario que en las t-temporales, en las redes de Petri p-temporales son los lugares los que llevan asociado un tiempo de disparo.

3. Modelo de datos

3.1 La red estocástica del proyecto

Una red estocástica de actividades se describe mediante el par (N, A) , donde $N = \{1, \dots, m\}$ representa el conjunto de nodos o vértices de la red y $A = \{(a_i, b_i) : a_i, b_i \in N, i = 1, \dots, n\}$ representa el conjunto de arcos dirigidos. Sobre esta red se utilizan dos representaciones. En la representación de actividad en el arco o grafo PERT las actividades se corresponden con los elementos de A , siendo $a_i \in N$ el nodo inicial y $b_i \in N$ el nodo final de la actividad A_i . No obstante, para representar prelacones lineales y de convergencia o divergencia simultáneas es necesario introducir actividades ficticias, por lo cual hemos optado por utilizar la representación de actividad en el nodo o grafo Roy. Se definen también un nodo inicial $r = a_1 \in N$ y un nodo final $s = b_n \in N$.

Cada actividad A_i tiene una duración definida por la variable aleatoria no negativa $\vec{\delta}_i$. Asimismo, cada actividad A_i tiene asociado un estado de terminación bivaluado en $E = \{\text{éxito, fracaso}\}$ definido por la variable aleatoria discreta $\vec{\tau}_i$.

La estructura probabilista de la red queda descrita por la distribución conjunta de los vectores aleatorios

$$\vec{\Delta} = \{\vec{\delta}_1, \dots, \vec{\delta}_n\} \quad [\text{f38}]$$

$$\mathcal{F} = \{\vec{\tau}_1, \dots, \vec{\tau}_n\} \quad [\text{f39}]$$

donde cada variable aleatoria $\vec{\delta}_i, \vec{\tau}_i$ es, a su vez, un vector aleatorio

$$\vec{\delta}_i = \{\delta_{i1}, \dots, \delta_{iz_i}\} \quad [\text{f40}]$$

$$\vec{\tau}_i = \{\tau_{i1}, \dots, \tau_{iz_i}\} \quad [\text{f41}]$$

y la variable aleatoria δ_{ij} define la duración de la actividad A_i en su j -ésima repetición, a la que llamaremos A_{ij} . Del mismo modo, la variable aleatoria discreta τ_{ij} describe el estado de terminación de la actividad A_i en su j -ésima repetición, suponiendo que esta repetición se materialice.

Si x es el cardinal del conjunto de caminos $\Pi_{A_{gh}, A_{lo}}$ entre dos nodos cualesquiera A_{gh}, A_{lo} , y $A(j)$ el

conjunto de actividades en el camino j-ésimo, $j = 1, \dots, x$, la duración del camino j-ésimo viene definida por la variable estocástica

$$\pi_j = \sum_{(A_{ik} \in A(j))} \delta_{ik} * \alpha_{ik} \quad [f42]$$

donde α_{ik} es una variable aleatoria endógena y booleana, que indica si la repetición k-ésima de la actividad A_i se produce. Es importante remarcar que el propio proyecto también tiene asociado un estado de terminación τ en E. Normalmente, este estado de terminación τ coincide con el estado de terminación de una actividad final, posiblemente vacía, del proyecto, que denotamos A_n . Pero es además posible definir un conjunto U de actividades cuya terminación exitosa conduzca a un éxito en el proyecto. Cualquiera de estas actividades finales que termine exitosamente supone el triunfo del proyecto. En este caso, la duración total del proyecto, una de las tres variables principales que definen el proyecto en el mundo real, se define como la variable estocástica

$$\delta/(\tau = verdadero) = \min_{\pi_i \in \Pi_{A_{ik}, A_{in}} \wedge \tau_{io} = \acute{e}xito \wedge A_{io} \in U} \{ \pi_i \} \quad [f43]$$

$$\delta/(\tau = falso) = \max_{\pi_i \in \Pi_{A_{ik}, A_{in}} \wedge \tau_{io} = fracaso} \{ \pi_i \} \quad [f44]$$



Figura 1: Estado de terminación del proyecto

Las fórmulas [f43] y [f44] cobran sentido definiendo el estado de terminación del proyecto como:

$$p(\tau = \acute{e}xito) = p\left(\bigcup_{A_j \in U \forall A_{ij}} \tau_{ij} = \acute{e}xito \right) \quad [f45]$$

Ambas variables τ, δ son estocásticas, en el sentido de que vienen definidas por la suma (o el máximo y mínimo de x sumas) de un número indeterminado de variables aleatorias.

3.2 Modelo de actividad

Definimos dos niveles de abstracción. En un primer nivel, más elevado o conceptual, la actividad A_i tiene una duración definida por la variable aleatoria $\vec{\delta}_i$. Así mismo, cada actividad A_i tiene asociada un estado de terminación bivaluado en $E = \{\text{éxito}, \text{fracaso}\}$ definido por la variable aleatoria $\vec{\tau}_i$. En este nivel, la actividad existe como concepto, por ejemplo, “Construir una planta de hormigón armado”, y puede llevar asociados unos recursos que se consideren necesarios. La actividad se encuentra engarzada en la red de relaciones de precedencia que definen los arcos de la red estocástica de actividades, red que en principio se puede caracterizar como grafo dirigido cíclico.

En un segundo nivel de abstracción, consideramos las repeticiones de cada actividad A_{ij} , de modo que desplegamos el grafo anterior en una representación que forma un grafo dirigido acíclico. En este nivel la actividad existe como la materialización del j -ésimo intento de llevar la actividad de primer nivel a buen término. En este nivel nos conciernen los componentes δ_{ij} y τ_{ij} que describen las variables aleatorias de la duración y terminación de la actividad de segundo nivel.

La actividad de nuestro modelo se puede dividir conceptualmente en tres zonas:

1. Activación: Es el punto en el que se decide si la actividad se pone en marcha.
2. Desarrollo: Es la zona, de duración variable, donde la actividad va empleando sus recursos para la consecución de sus fines, consecución que no está garantizada.
3. Terminación: Es el momento donde se decide si la actividad ha cumplido sus objetivos y ha sido un éxito, o si, por el contrario, hay que considerarla un fracaso



Figura 2: Zonas de una actividad

Para expresar el comportamiento de una actividad de estas características en la red estocástica de actividades del proyecto, vamos a emplear el formalismo de las redes de Petri temporales. En el capítulo 2, “Estado del arte”, hemos visto algunos trabajos en el área con redes de Petri [Sawhney 1997], [Kusumoto et al. 1997], [Chang; Christensen 1999], [Jeetendra et al. 2000], [Kao et al. 2002], [Zafra-Cabeza et al. 2004a], [Zafra-Cabeza et al. 2004b], [Zafra-Cabeza et al. 2005]. Nosotros utilizaremos redes de Petri t-temporales, un tipo de red que ha encontrado gran aceptación en la modelización de variados sistemas de tiempo real [Montano et al. 2000], [Staines 2006] y flujo de trabajo [van der Aalst 1998]. En nuestro modelo, incluiremos tanto transiciones estocásticas como transiciones deterministas, lo que nos inscribe en el formalismo de las redes de Petri estocástico-deterministas⁴⁵. Debido a que las transiciones temporales no siguen distribuciones exponenciales ni geométricas, no podemos seguir los formalismos de las redes de Petri estocásticas generalizadas [Marsan et al. 1983], [Marsan et al. 1995], ni de las redes de Petri estocásticas markovianas. Podemos considerar a este formalismo una red de Petri estocástica más general.

Para definir la red de Petri empleamos la tupla $(S, T, F, M_0, W, K, H, B, B')$, donde:

- S es el conjunto finito de lugares.
- T es el conjunto finito de transiciones.
- F es el conjunto de arcos que conectan S y T , de modo que $F \subseteq (S \times T) \cup (T \times S)$, es decir, que ningún arco conecte entre sí dos lugares o dos transiciones.
- $M_0: S \rightarrow \mathbb{N}$ es el marcado inicial, según el cual se coloca una ficha en el lugar de activación de la actividad inicial. También, se colocan tantas fichas como unidades de recursos se dispongan en los lugares correspondientes a los recursos.
- $W: F \rightarrow \mathbb{N}^+$ es el conjunto de pesos de los arcos, indicando alternativamente cuántas fichas se consumen de un lugar, si el arco $f \in S \times T$, o se producen en una transición, si el arco $f \in T \times S$

⁴⁵ Para el estudio de las redes de Petri estocástico-deterministas véanse los capítulos 2º y 3º de [Lindemann 1998].

- $K: S \rightarrow \mathbb{N}^+$ es el conjunto de restricciones de capacidad de los lugares.
- $H \subset F$ es el conjunto de arcos de inhibición, que impiden el disparo de una transición cuando en su lugar de origen hay fichas.
- $B = (Dist, escalada)^b$ Es el primer vector de atributos de las b transiciones temporales, cuyo primer componente indica la distribución aleatoria y cuyo segundo componente indica que la anterior distribución puede depender linealmente del número de fichas. No es necesario que las distribuciones sean exponenciales.
- $B' = (Bernoulli, escalada)^b$ Es el segundo vector de atributos de las b transiciones temporales, cuyo primer componente indica la ley de Bernoulli de la probabilidad de conducir a los lugares de salida asociados al éxito, y cuyo segundo componente indica que dicha ley depende linealmente del número de fichas.

Si para una transición $t \in T$ llamamos $I(t) \subseteq S$ al conjunto de lugares normales de entrada, es decir, lugares que preceden a la transición de modo que exista un arco en F que los conecte directamente. Llamamos a $L(t) \subseteq S$ los lugares inhibidores de entrada, de modo que exista un arco en H que los conecte directamente. Suponemos que $L(t) \cap I(t) = \emptyset \forall t$.

En las transiciones, distinguimos $T' \subset T$ un subconjunto de transiciones inmediatas, siendo el resto T'' temporales $|T'| = |T| - |T''| = |T| - b$.

Sea $J(t) \subseteq S$ el conjunto de lugares de salida de una transición. Si $t \in T''$, distinguimos los subconjuntos $E(t) \subseteq J(t)$ y $F(t) \subseteq J(t)$ correspondientes a los estados de la actividad en $E = \{\text{éxito, fracaso}\}$ respectivamente.

$$E(t) \cap F(t) = \emptyset \quad [f46]$$

$$E(t) \cup F(t) = J(t) \quad [f47]$$

Sea $M_p = \{m_1, m_2, \dots, m_L\}$ un marcado de la red de Petri donde $L = |S|$ y m_j es el número de fichas en el lugar $s_j \in S$. Para este marcado definimos:

$$E(M_p) = \{t \in T : m_j \geq 1 \forall s_j \in I(t) \wedge m_i = 0 \forall m_i \in L(t)\} \quad [f48]$$

Esto representa el conjunto de transiciones que se hallan habilitadas. Sobre este conjunto se define un orden de prioridad de disparo. Las transiciones inmediatas se disparan primero. Sea la primera transición $t \in E(M_p) \wedge t \in T''$ una transición temporal habilitada, definimos la probabilidad:

$$b'(t, M_p) = p(\forall j \in E(t) : m_j' = m_j + W(t, j) / M_p) \quad [f49]$$

donde $m_j' \in M_{p+1}$ y $W(t, j)$ representa el peso del arco que lleva de la transición al lugar de salida.

Nótese que la probabilidad depende del marcado M_p . En particular, depende del marcado de un único lugar que representa el número de repeticiones de la actividad.

En la figura 3 podemos ver una actividad típica de yunción, que tiene varias actividades predecesoras:

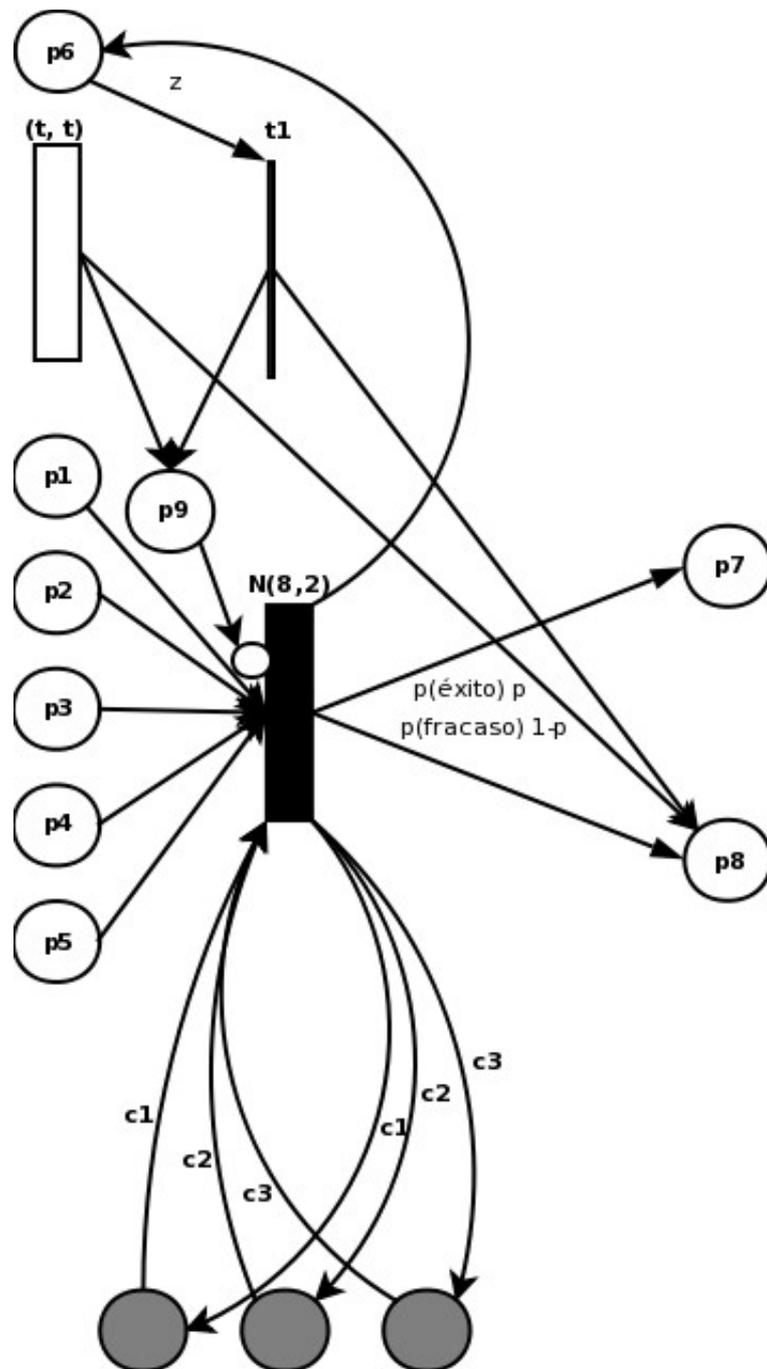


Figura 3: Modelo simple de actividad en red de Petri

Las relaciones de precedencia están representadas por los arcos que salen de los lugares p1, p2, p3, p4 y p5. La duración de la actividad, $\vec{\delta}_i$ está modelada en la transición temporal marcada por el trazo grueso N(8,2), con lo que expresamos que le corresponden variables aleatorias normales. La terminación de la actividad, $\vec{\tau}_i$ está representada por los arcos etiquetados que llevan a los lugares p7 y p8. Siguiendo el modelo de las redes de Petri estocásticas más generales [Haas; Schedler 1987], los arcos indican la probabilidad de seguir uno y otro camino. Las transiciones representadas por una línea son inmediatas, las transiciones representadas por un rectángulo son temporales.

A continuación, veremos otros detalles del modelo.

3.2.1 Activación

Hemos definido globalmente los arcos entre los lugares p1,...,p5 de la figura 3 y las transiciones como relaciones de precedencia. Siendo más precisos, son los eventos de activación Ev_{ij} los que relacionan a la actividad con sus predecesoras, de modo que $A_j \in P_i$ donde P_i es el conjunto de actividades que preceden inmediatamente a A_i . Cada evento de activación puede ser de signo positivo, si el evento colabora en la activación cuando la actividad predecesora tiene una terminación exitosa, o de signo negativo, si el evento colabora cuando la actividad predecesora falla. No existen otros eventos de activación, aunque ahora pasaremos a considerar los eventos de desactivación ligados al tiempo y al número de repeticiones de la actividad.

En el nivel de actividad, a parte de las variables aleatorias $\vec{\delta}_i$ y $\vec{\tau}_i$, tenemos una variable aleatoria booleana endógena definida por una función de activación $\vec{\alpha}_i = f_i(\vec{\xi}_i, \vec{t}_i, z_i)$ que indica si la actividad comienza a ejecutarse. Los parámetros de la función son el vector $\vec{\xi}_i = \{\xi_1, \dots, \xi_j, \dots, \xi_{|P_i|} : \xi_j \leftarrow \vec{\tau}_j, A_j \in P_i\}$, determinado vector de \vec{t}_i tiempos máximos de realización de los eventos de activación y un contador z_i que marca el número máximo de repeticiones que la actividad admite.

El contador de repeticiones z_i se encuentra reflejado en la figura 3 mediante el lugar p6, de modo que si se supera el número de repeticiones z_i , la actividad atraviesa la transición no temporal t1 y termina en un estado de fracaso. A su vez, desde el lugar p9 se inhibe la transición principal N(8,2). El vector \vec{t}_i indica los tiempos máximos de realización de los eventos de activación. En la figura 3 se muestra una versión simplificada, según la cual todos los tiempos máximos son idénticos a t, y la transición temporal (t, t) lleva al estado de fracaso p8 una vez transcurrido el tiempo t. Igualmente, la transición principal queda deshabilitada.

Gráficamente, despiezando la figura 2 veríamos:

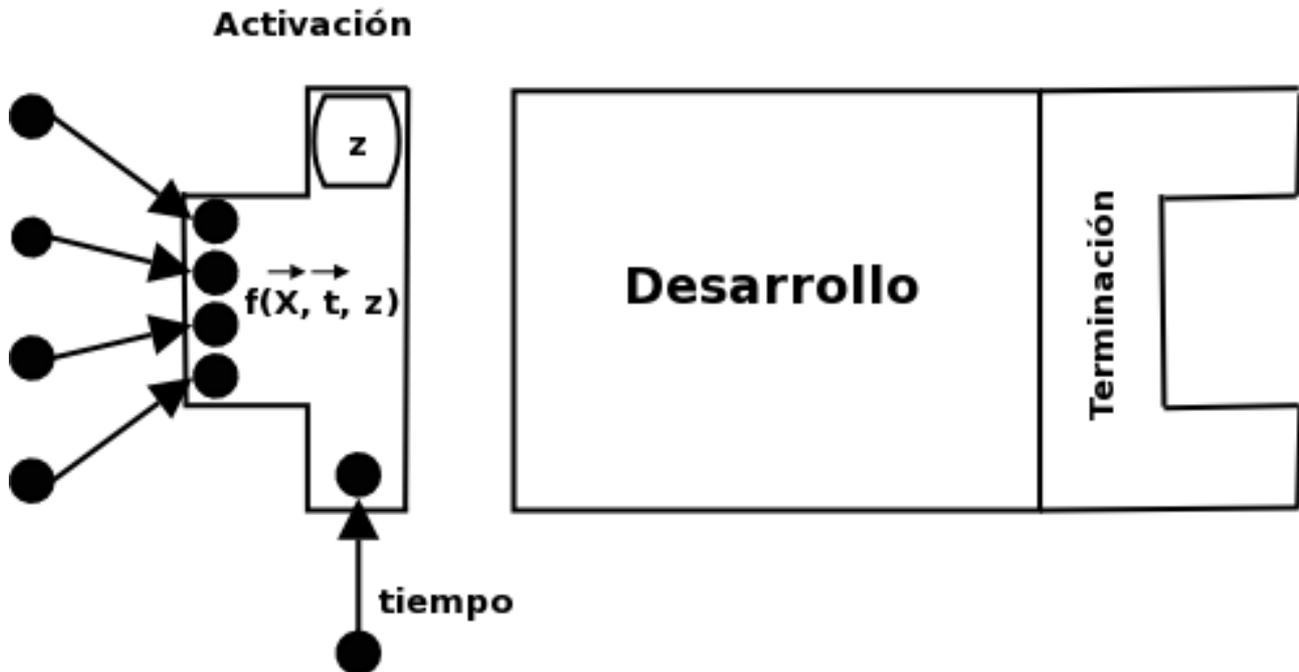


Figura 4: Despiece de la activación

Una característica del evento son los tiempos de activación. Independientemente del signo del evento Ev_{ij} ,

$$t_j < f_j \Rightarrow \alpha_{ij} = false \quad [f50]$$

donde llamamos α_{ij} el valor de la función de activación en lo que respecta al evento Ev_{ij} , y f_j al tiempo de finalización de la actividad predecesora A_j en el programa P. Para ello es preciso determinar el programa P mediante un algoritmo de secuenciación, como veremos en el apartado 3.6, “Exégesis de resultados”. Es lo que en la figura 3 expresa el arco que va de la transición temporal (t, t) al lugar p8.

La función de activación determina el curso que tomará la actividad, ya que si la actividad no llega a activarse, no puede llegar a buen término:

$$\alpha_{ij} = false \Rightarrow \tau_{ij} = false \quad [f51]$$

Además, la duración de la actividad también se ve afectada, de modo que la actividad que no llegue a activarse no consume tiempo, ni por lo tanto recursos:

$$\alpha_{ij} = false \Rightarrow \delta_{ij} = 0 \quad [f52]$$

La función de activación trata de describir ciertas relaciones entre las actividades que, aunque

presentes en la práctica real de la gestión por proyectos, no han encontrado hasta ahora una manera sencilla de entrar en las aplicaciones informáticas. Por ejemplo, es posible que una actividad comience su ejecución cuando alguna de sus predecesoras concluya exitosamente, sin necesidad de que lo hagan las demás predecesoras. Para ello, hemos incluido la función booleana OR como una de las posibles funciones de activación.

Por ejemplo, y siguiendo el formalismo de las redes de Petri, esta actividad contiene una función de activación OR:

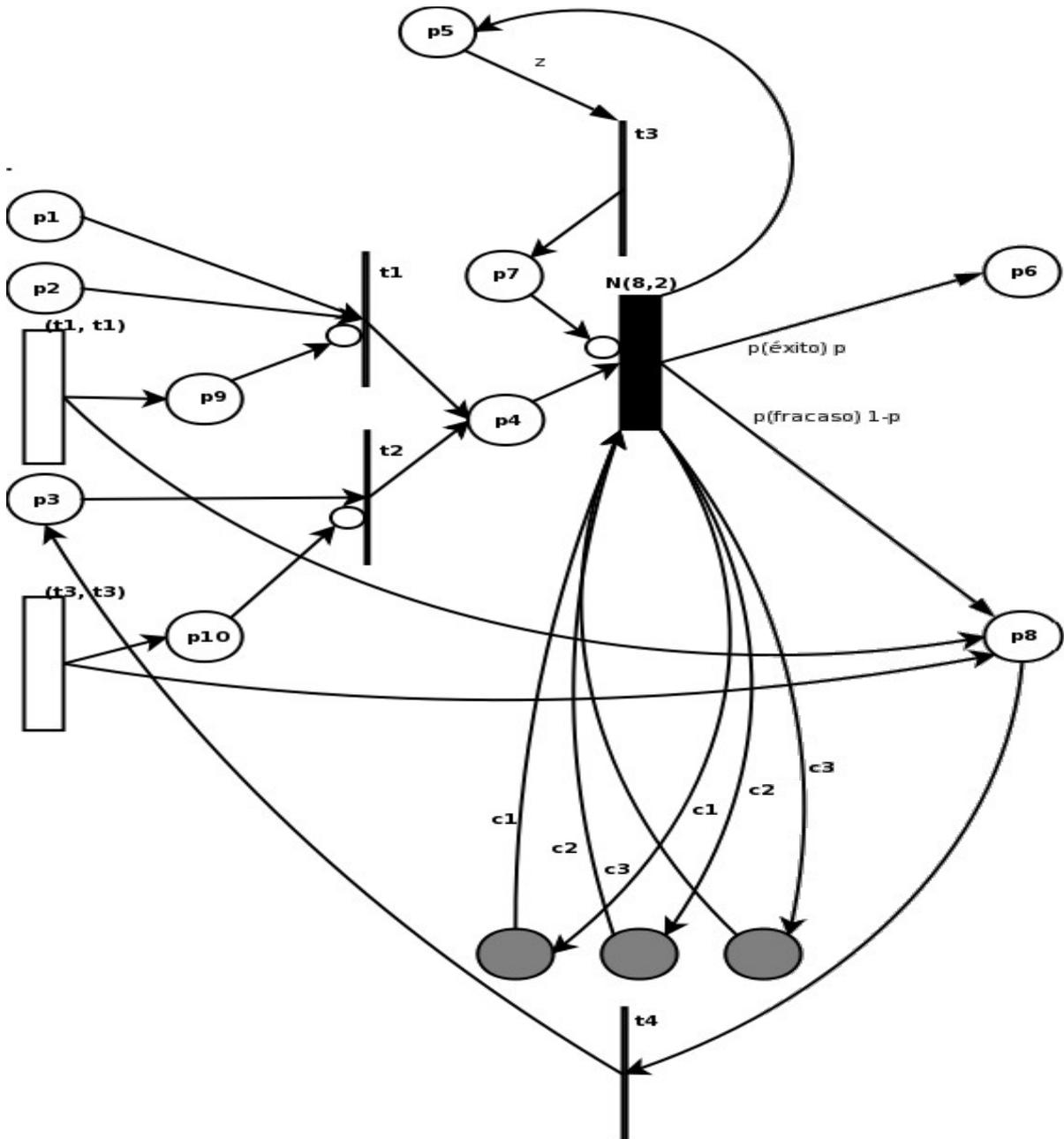


Figura 5: Modelo de actividad con activación OR y retroalimentación en red de Petri

En el ejemplo de la figura 5 hemos introducido un lugar p_4 , que representa la activación de OR una u otra cláusula. La primera cláusula, con los lugares p_1 y p_2 y la transición inmediata t_1 , es el punto de yunción de dos actividades precedentes. La segunda cláusula indica un bucle de retroalimentación. Está representada mediante el lugar p_3 y la transición inmediata t_2^{46} , en este caso, y mediante el lugar p_8 , que representa el fracaso de la actividad modelada y la transición inmediata t_4 . Es decir, la actividad se dispara cuando dos predecesoras hayan finalizado en el estado adecuado o cuando la propia actividad termine en estado de fracaso.

Nuestro sistema incluye las funciones de activación AND y OR. La función AND implica que todos los eventos de activación tienen que suceder para que la actividad comience. Entre ambas funciones lógicas y los signos positivo y negativo de los eventos de activación podemos modelar funciones lógicas de complejidad arbitraria.

3.2.1.1 Prescindibilidad

Cuando en un sistema, en un mismo proyecto, se mezclan ambas funciones de activación, surge la cuestión de saber cuáles de las actividades son realmente necesarias para la completitud del proyecto.

Desde el punto de vista estocástico, definimos la necesidad de una actividad para terminar el proyecto como imprescindibilidad. Suponemos que, si la actividad no es necesaria para la terminación exitosa del proyecto, podemos eliminarla de manera segura y ahorrar el esfuerzo de los recursos asociados. Esto es lo que [Gillies; Liu 1995] llamaron un modelo AND/OR con omisión. Todo esto se realiza en el contexto de la simulación, de modo que si una actividad falla con probabilidad p , y este resultado no es válido para la activación de una actividad posterior definida por la función lógica OR, entonces, con una probabilidad p , los recursos de la actividad en cuestión no serán necesarios.

Existen dos causas de prescindibilidad. La primera está en el resultado de la actividad, es decir, que el resultado no sea válido para un determinado evento de activación posterior. La segunda está en el tiempo, el caso que se da cuando la actividad obtiene el resultado que lanza el evento de activación posterior, pero lo obtiene después de que haya sucedido otro evento de activación del que depende la función lógica OR. Está claro que esta segunda causa de prescindibilidad depende de la planificación concreta que se siga en el proyecto.

46 Obsérvese que la presencia de la transición t_4 impide que la actividad se dispare una vez más de lo necesario. Otra manera de evitar la condición de carrera entre t_2 y t_3 consiste en definir un orden prioridad para disparo en el conjunto de transiciones inmediatas. También es posible eliminar t_3 , fusionar p_5 y p_7 y pasar el peso del arco (p_5 , t_3) al arco inhibitor.

Cuando la causa de la prescindibilidad reside en el resultado de la actividad, la actividad sólo es prescindible, y, por lo tanto, sólo podremos ahorrar sus recursos, si la no activación de la actividad conduce al resultado que no lanza el evento de activación de la actividad posterior. Es decir, actuamos como si la imprescindibilidad de la tarea fuese una condición necesaria para su activación.

La prescindibilidad de una actividad es un atributo que se propaga por sus predecesoras. Veamos un ejemplo:

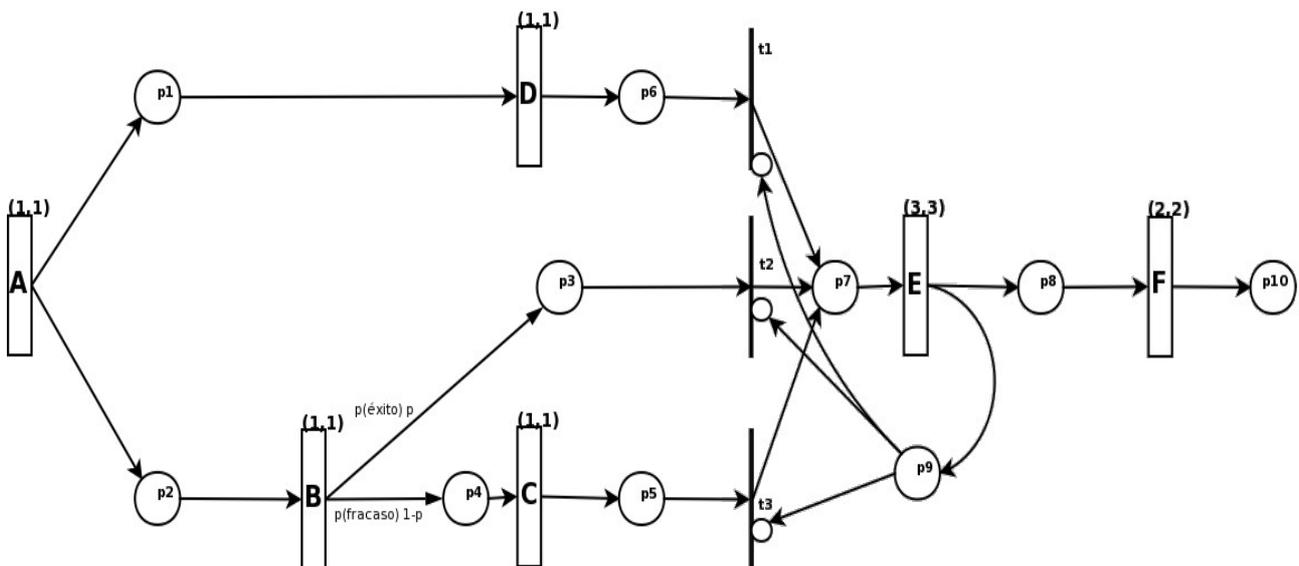


Figura 6: Ejemplo de proyecto con actividad prescindible

Los puntos de color representan el final exitoso o fallido de la actividad. La actividad E es una actividad OR, y puede ser activada por D o por C o por B. En este caso determinado, si la planificación que se ha seguido es voraz y programa las actividades para empezar cuanto antes, la actividad E es activada por la terminación exitosa de D en el tiempo 2. La actividad C termina en el tiempo 3, y es prescindible por causa del tiempo. Su ejecución puede ser omitida, y sus recursos, por lo tanto, pueden ahorrarse.

Ahora, ¿qué sucede con los recursos de la actividad B? Sus recursos tampoco son necesarios, ya que no es necesario producir las condiciones de activación de la actividad C; así que la actividad B es indirectamente prescindible. También es directamente prescindible, ya que su resultado fallido no dispara ningún evento de activación en D.

La prescindibilidad que juzgamos tras la particular simulación del proyecto es un dato obtenido a posteriori. En este sentido, el dato que ofrecemos al director de proyecto indica la futilidad del esfuerzo dedicado a determinado plan de trabajo concurrente.

En determinados casos, que pueden ser previstos a partir de la simulación, esta prescindibilidad puede ser determinada por el director de proyecto en tiempo real. Obsérvese en el proyecto simulado en la figura 6 que en el tiempo 2 han finalizado las actividades A, B y D. En ese momento, el director de proyecto tiene información suficiente para saber con certeza que la actividad C es prescindible. El director de proyecto puede llegar a la conclusión de que una actividad es prescindible en medio de su ejecución. En ese caso, la decisión de suspender la ejecución de la actividad parece consecuente. Por lo tanto, es oportuno averiguar en qué momento se produce esta perspicuidad de la prescindibilidad para afinar la previsión de recursos empleados en los planes alternativos.

Para estudiar el proyecto, obtenemos una simulación de todas las variables que contiene Q rondas. Una vez realizada la temporalización del proyecto, en una ronda cualquiera de simulación q definimos $B_i^{[q]}$ como el conjunto de las actividades que precediendo a la actividad A_i cumplen

$$B_i^{[q]} = \{A_j \mid A_j \in P_i \wedge \text{sign}(\tau_j^{[q]}) = \text{sign}(Ev_{ij})\} \quad [f53]$$

$$\forall q, i, B_i^{[q]} \subset P_i$$

Definimos también como $D_i^{[q]}$ las actividades que primero terminan de entre las de $B_i^{[q]}$.

$$D_i^{[q]} = \{A_j \mid A_j \in B_i^{[q]} \wedge f_j = \min(f_k) \forall k : A_k \in B_i^{[q]}\} \quad [f54]$$

Llamaremos $t_i^{[q]}$ al tiempo f_j en que terminan las actividades de $D_i^{[q]}$. Dos subconjuntos de los conjuntos recién definidos nos serán útiles. Primero, declaramos las actividades activadoras positivas.

$$E_i^{[q]} = \{A_j \mid A_j \in B_i^{[q]} \wedge \tau_j^{[q]} = \text{éxito}\} \quad [f55]$$

y, a continuación, aplicamos esta declaración sobre $D_i^{[q]}$, que contiene las primeras actividades activadoras

$$F_i^{[q]} = E_i^{[q]} \cap D_i^{[q]} \quad [f56]$$

En los casos en los que la función de activación contenga respecto del componente \vec{E}_i una función lógica atómica, simple, podemos definir A_{OR} para representar el conjunto de actividades del proyecto que tienen una función de activación OR, y A_{AND} el conjunto de actividades del proyecto que tienen una función de activación simple AND o tienen una única predecesora, o ninguna. Entonces, podemos definir recursivamente la imprescindibilidad de una tarea mediante

$$\begin{aligned}
 A_i \in I^{[q]} \wedge A_i \in A_{AND} &\Rightarrow E_i^{[q]} \subset I^{[q]} \\
 A_i \in I^{[q]} \wedge A_i \in A_{OR} &\Rightarrow F_i^{[q]} \subset I^{[q]}
 \end{aligned}
 \tag{f57}$$

partiendo del caso básico

$$A_n \in I^{[q]}
 \tag{f58}$$

Por lo tanto, la imprescindibilidad de una actividad A_i se estima como

$$i_i = \frac{\sum_{q=1}^M \left(\begin{array}{l} A_i^{[q]} \in I^{[q]} \Rightarrow 1 \\ A_i^{[q]} \notin I^{[q]} \Rightarrow 0 \end{array} \right)}{Q}
 \tag{f59}$$

En los casos en los que la función de activación sea compleja, como en el ejemplo de la figura 5166, procederemos de análoga manera con los lugares que corresponden a las funciones lógicas simples, y aplicaremos las fórmulas [f57] hasta colegir la imprescindibilidad de la propia actividad.

3.2.2 Desarrollo

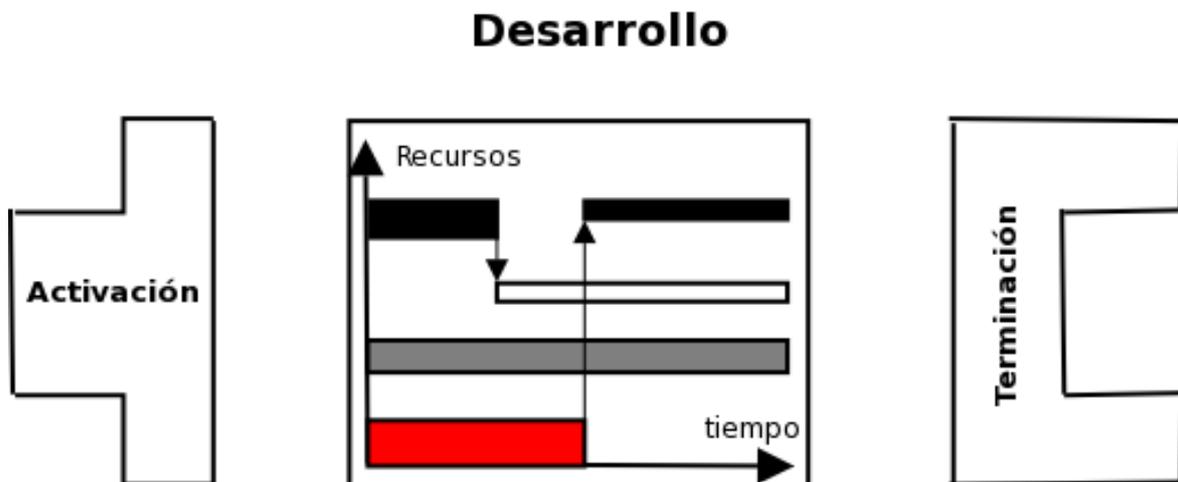


Figura 7: Despiece del desarrollo

En la figura 7 podemos observar un despiece del modelo de una actividad, destacando la fase de desarrollo. Aunque la repetición de actividades está contemplada en el modelo, no se permite que la misma actividad se esté ejecutando dos veces simultáneamente. Es lo que, en el formalismo de

redes de Petri, se conoce como red de Petri temporal no reentrante [Carlier; Chretienne 1988].

3.2.2.1 La duración probabilística

La duración de la actividad, supuesto que realmente llegue a activarse, viene definida por una función de distribución de probabilidad, o por un simple número cuando se considera que la estimación de un sólo punto es suficiente.

Las distribuciones de probabilidad pueden tomar varias formas, paramétricas o no paramétricas. Para la aplicación concreta de la duración de la actividad utilizaremos distribuciones paramétricas continuas⁴⁷. A saber:

1. Uniforme: Distribución equiprobable entre dos puntos⁴⁸.
2. Normal: Distribución simétrica con forma de campana $N(\mu, \sigma)$ ⁴⁹.
3. Lognormal: Distribución curva con una cola final más alargada⁵⁰.

Una de las dificultades del uso de las distribuciones paramétricas reside en la estimación de los parámetros, que no suelen ser intuitivos. Para la estimación subjetiva de los parámetros de una distribución normal, nos basamos en que el valor más probable coincide con la media μ y en que el intervalo $[\mu - 2\sigma, \mu + 2\sigma]$ contiene el 95% de los valores. Por lo tanto, si se estima el valor máximo con un 95% de confianza, como se suele pedir para usar una BetaPERT⁵¹,

$$\sigma = \frac{(\max - \text{moda})}{2} \quad [\text{f60}]$$

La distribución normal se extiende de -infinito a +infinito, aunque nosotros la truncamos en 0 ya que $\delta_i \in \mathbb{R}^+$.

Otro método de evaluación es el método de los fractilos [Selvidge 1980], según el cuál d_α es el fractil α , y el fractil $d_{1-\alpha}$ se define como el punto donde

$$P(\delta > d_{1-\alpha}) = 1 - \alpha \quad [\text{f61}]$$

47 Para el estudio de las distribuciones de probabilidad continuas véase [Johnson et al. 1970]

48 La referencia de la distribución uniforme se puede encontrar en [Johnson et al. 1995a]

49 La referencia de la distribución normal se puede encontrar en [Johnson et al. 1994a]

50 La distribución lognormal describe una variable aleatoria cuyo logaritmo tiene una distribución normal. La referencia de la distribución lognormal se puede encontrar en [Johnson et al. 1994b]

51 La distribución beta es una curva de forma variada en el intervalo [0,1]. Incluye a la distribución uniforme entre estos dos puntos. La PERT define un método específico para estimar sus parámetros. La referencia de la distribución beta se puede encontrar en [Johnson et al. 1995b]

Específicamente, para la estimación subjetiva de los parámetros de una uniforme, podemos seguir a [Chapman; Ward 2000] y pedir los valores mínimos y máximos en los percentiles 10 y 90%. Posteriormente, calculamos los mínimos y máximos reales extendiendo un 10% este rango.

En los casos en los que la ejecución de la actividad debe repetirse varias veces, a veces sucede que los parámetros de distribución de probabilidad dependen número de repetición de la actividad. En el formalismo de las redes de Petri se define como una dependencia escalar respecto del marcado de un lugar.

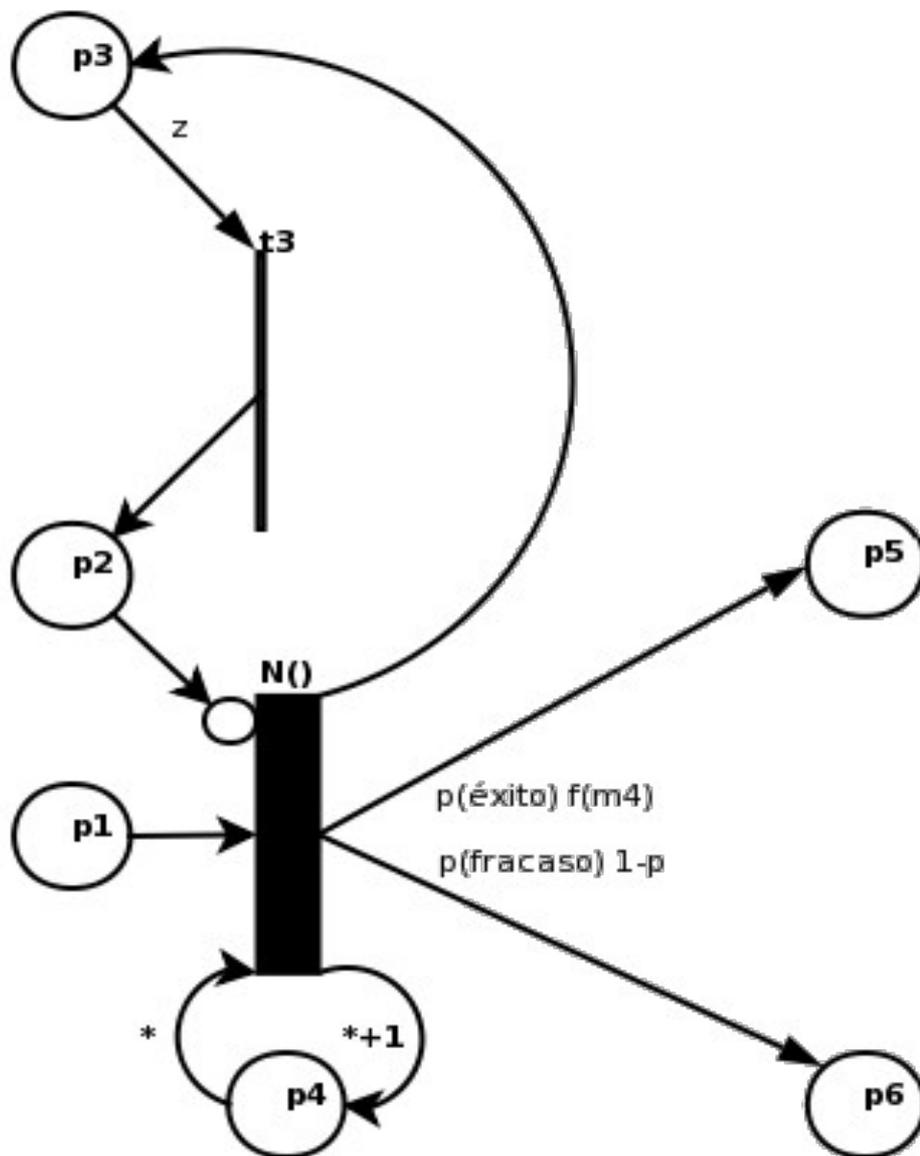


Figura 8: Dependencia paramétrica del número de repeticiones

Veremos la aplicación de este modelo en el apartado 3.3.3 “Los ciclos repetitivos”.

3.2.2.2 Los recursos complejos

El desarrollo de la actividad incorpora el concepto de recursos complejos, según el cuál las necesidades de los recursos de la actividad no son únicas durante toda la duración de la misma, sino que unos recursos pueden entrar a trabajar en un determinado punto del desarrollo de la actividad, o dejar de ser utilizados antes de su finalización.

El coste de una actividad está ligado a los recursos complejos. Entendemos que el coste se va produciendo durante el desarrollo de la actividad, una vez que ésta haya sido activada. El coste X_{ij} de la actividad repetida A_{ij} depende de la variable aleatoria ligada a su duración, en el sentido de que

$$X_{ij} = \sum_{R_k \in \vec{R}_i} c_{ijk} l_{ik} \delta_{ij} = \delta_{ij} \sum_{R_k \in \vec{R}_i} c_{ijk} l_{ik} \quad [f62]$$

donde c_{ijk} representa el consumo previsto del recurso R_k en la actividad A_i durante su j -ésima repetición y l_{ik} representa la fracción de X_{ij} que espera ocupar el recurso complejo R_k . es una $\vec{R}_i = \{R_1, \dots, R_k, \dots\}$ agrupación de conveniencia que representa los recursos necesarios para llevar a cabo la actividad A_i . Estos recursos pertenecen necesariamente a los recursos del proyecto \vec{R} , y a su vez, en un nivel superior de agregación, a los recursos del portafolio \vec{RG} . Esto nos permite predecir y gestionar conflictos por los recursos en el nivel táctico-estratégico de la organización (véase la figura 12).

$$\vec{R}_i = \{R_1, \dots, R_k, \dots\}, R_k \in \vec{R} \subseteq \vec{RG} \quad [f63]$$

3.2.3 Terminación



Figura 9: Despiece de la terminación

La terminación de la actividad refleja el éxito de ésta. La terminación puede ser exitosa o fallida. Este resultado se establece en base a una ley estadística de Bernoulli⁵². El fin exitoso o fallido de la actividad es probabilístico por naturaleza, aunque se previene que, por conveniencia, en muchos casos se establezca una probabilidad de éxito igual a 1. El parámetro de la ley de Bernoulli lo define el planificador en función de su percepción del riesgo intrínseco de la actividad. Esta percepción será subjetiva la mayor parte de las veces, aunque la información histórica sobre el resultado de actividades similares puede ayudar mucho a la correcta estimación del parámetro.

La ley de Bernoulli puede parametrizarse, en los casos en los que se produzca la repetición de la actividad, para que dependa del número de repeticiones, como se muestra en la figura 8. Veremos la aplicación de este modelo en el apartado 3.3.3 “Los ciclos repetitivos”.

No deben incluirse en la ley estadística dudas sobre la terminación correcta de actividades precedentes cuyo producto necesite la actividad en cuestión, ni dudas sobre la disponibilidad de los recursos necesarios para el desarrollo de la misma. Estos aspectos de la incertidumbre estructural son tratados específicamente por los algoritmos de simulación y asignación de recursos.

Tampoco recomendamos, de acuerdo con [Demeulemeester; Herroelen 2002], evitar conflictos en los recursos mediante relaciones de precedencia de comienzo a comienzo, relaciones que aprovechen la bajada de la demanda de recursos hacia el final de la actividad para permitir cierto grado de solapamiento. Las relaciones de precedencia deben marcar únicamente restricciones técnicas, y los posibles conflictos en los recursos se dejan para que los resuelva el algoritmo de

⁵² Distribución de probabilidad discreta que toma valor 1 con probabilidad p y valor 0 con probabilidad $q = 1 - p$

asignación de recursos. Como bien indican [Clough et al. 1972]⁵³:

[...]tales restricciones se introducían en la red como relaciones lógicas. Este método de “cablear” las restricciones, sin embargo, dificulta el proceso de actualizar y cambiar el programa, ya que la reordenación de las prioridades de los recursos en el proyecto debe ir siempre acompañada del recableado de tales relaciones restrictivas.

Otro asunto es que este algoritmo aproveche dicha información, modelada bien mediante recursos complejos o bien mediante un nivel más de detalle en la estructura de descomposición del problema, para permitir o no la concurrencia de actividades.

3.3 La red generalizada del proyecto

3.3.1 Las redes de alternativas múltiples

Este tipo de red es conocido en la literatura como los caminos alternativos⁵⁴, el problema del PERT con alternativas o la red de actividades con relaciones de precedencia AND/OR.

A menudo se tiende a obviar o eliminar las alternativas durante un proyecto. El personal del proyecto suele considerar una sola alternativa, y un único curso de acción. En general, se suele elegir la alternativa más segura, la más conservadora [McCray et al. 2002]. Las razones de este comportamiento están en la dificultad de contemplar todas las opciones a la vez. Como consecuencia, las aproximaciones más creativas a los problemas son eliminadas, y muchas oportunidades de innovación y posicionamiento competitivo se esfuman. Por ello, proponemos en este prototipo un modelo donde las alternativas de ejecución se contemplan a la vez.

3.3.2 Los planes de mitigación del riesgo

Una de las utilidades del nuestro modelo, que combina la terminación biestable de la actividad con la activación mediante disparadores AND/OR, está en el estudio de los planes de mitigación del riesgo. En su variante más habitual, éstos toman la forma de un plan de contingencia que se debe activar ante determinado evento desfavorable. En nuestro modelo, el evento desfavorable es la terminación fallida de una actividad.

⁵³ La cita se refiere a la 4ª edición, de 2000.

⁵⁴ *Alternate paths*

El plan de contingencia puede ser tan complejo como se necesite, e incluir, además, sus propios planes de contingencia. Puede consumir recursos de reserva, o utilizar los propios recursos asignados al proyecto.

Por ejemplo, presentamos este caso de la industria del software:

Para la integración de tres aplicativos se proponen las siguientes actividades:

Actividad	Descripción	Probabilidad de éxito	Duración
A ₁	Instalación de plataforma de pruebas	0.7	1
A ₂	Configuración del entorno de pruebas	1	7
A ₃	Programación en base de datos	0.9	7
A ₄	Programación mediante la interfaz 1	1	1.5
A ₅	Programación mediante la interfaz 2	1	1.5
A ₆	Sincronización con 3 ^r sistema	1	1.5
A ₇	Cuaderno de pruebas	1	1
A ₈	Pruebas del módulo	1	10
A ₉	Pruebas de integración	1	10

Tabla 1: Proyecto sin plan de contingencia

La probabilidad de éxito indica la confianza del director de proyecto en que el objetivo de la actividad se alcance en un plazo razonable. Es decir, se considera que la actividad fracasa cuando alguien evalúa que el producto de la actividad es defectuoso, y hay que tomar otra alternativa o recomenzar la actividad, o cuando la actividad se ha alargado hasta tal punto que el responsable decide abandonar la actividad. No se considera fracaso, *per se*, respecto de esta probabilidad, que la actividad se alargue más allá de su tiempo de terminación o que la evaluación del producto indique la necesidad de continuar la actividad. Obviamente, la baja probabilidad de éxito de A₁, una actividad inicial, cuestiona de manera inaceptable la viabilidad del proyecto entero. El problema radica en la necesidad de conseguir un hardware especial de un proveedor con un historial de fallos y retrasos.

Se prepara un plan de contingencia para el caso en que este hardware no llegue o no sea conforme a la especificación. En su caso, se trabajaría con otro tipo de plataforma. Desgraciadamente, la

plataforma de contingencia plantea una incompatibilidad con la base de datos de A_3 , y es necesario añadir más actividades para asegurar la entrega final.

Estas serían las actividades que se deberían realizar en el caso de que fuese necesario ejecutar el plan de contingencia.

Actividad	Descripción	Probabilidad de éxito	Duración
A_1	Instalación de plataforma de pruebas A	0.7	1
A_2	Configuración del entorno de pruebas en A	1	7
A_3	Instalación de plataforma de pruebas B	1	1
A_4	Configuración del entorno de pruebas en B	1	6
A_5	Programación en base de datos	0.9	7
A_6	Programación en sistema de ficheros	1	10
A_7	Programación mediante la interfaz 1	1	1.5
A_8	Programación mediante la interfaz 2	1	1.5
A_9	Sincronización con 3 ^r sistema	1	1.5
A_{10}	Cuaderno de pruebas	1	1
A_{11}	Cuaderno de migración	1	5
A_{12}	Pruebas del módulo	1	10
A_{13}	Pruebas de integración	1	10

Tabla 2: Proyecto con plan de contingencia

La aproximación disponible en los sistemas de gestión de proyectos actuales consiste en plantear un escenario para el caso en que A_1 termine exitosamente, y otro escenario para el caso en que fracase. Ambos escenarios se programan por separado, como proyectos distintos, y luego se realiza algún tipo de integración ad hoc. Es una aproximación perfectamente realizable si los escenarios son pocos, pero escala muy mal. Podemos observar que la probabilidad de éxito en la actividad A_3 es también preocupante. Podría necesitarse otro plan de contingencia, lo que normalmente tendrá un efecto multiplicativo sobre el número de escenarios a considerar.

En nuestro modelo, todos los planes de contingencia y acciones mitigadoras pueden ser integrados en la planificación del proyecto. Por lo tanto, facilitamos que el profesional siga las

recomendaciones de incluir las actividades de gestión de los riesgos en el marco del ciclo de vida del proyecto [Chapman; Ward 1997], [Jaafari 2001]. Los indicadores de transición de los riesgos también pueden ser incluidos [DeMarco; Lister 2003] a modo de actividades de control que se activan en función de los resultados de otras actividades de chequeo. Para el caso que nos ocupa, esto supone considerar la siguiente red de alternativas:

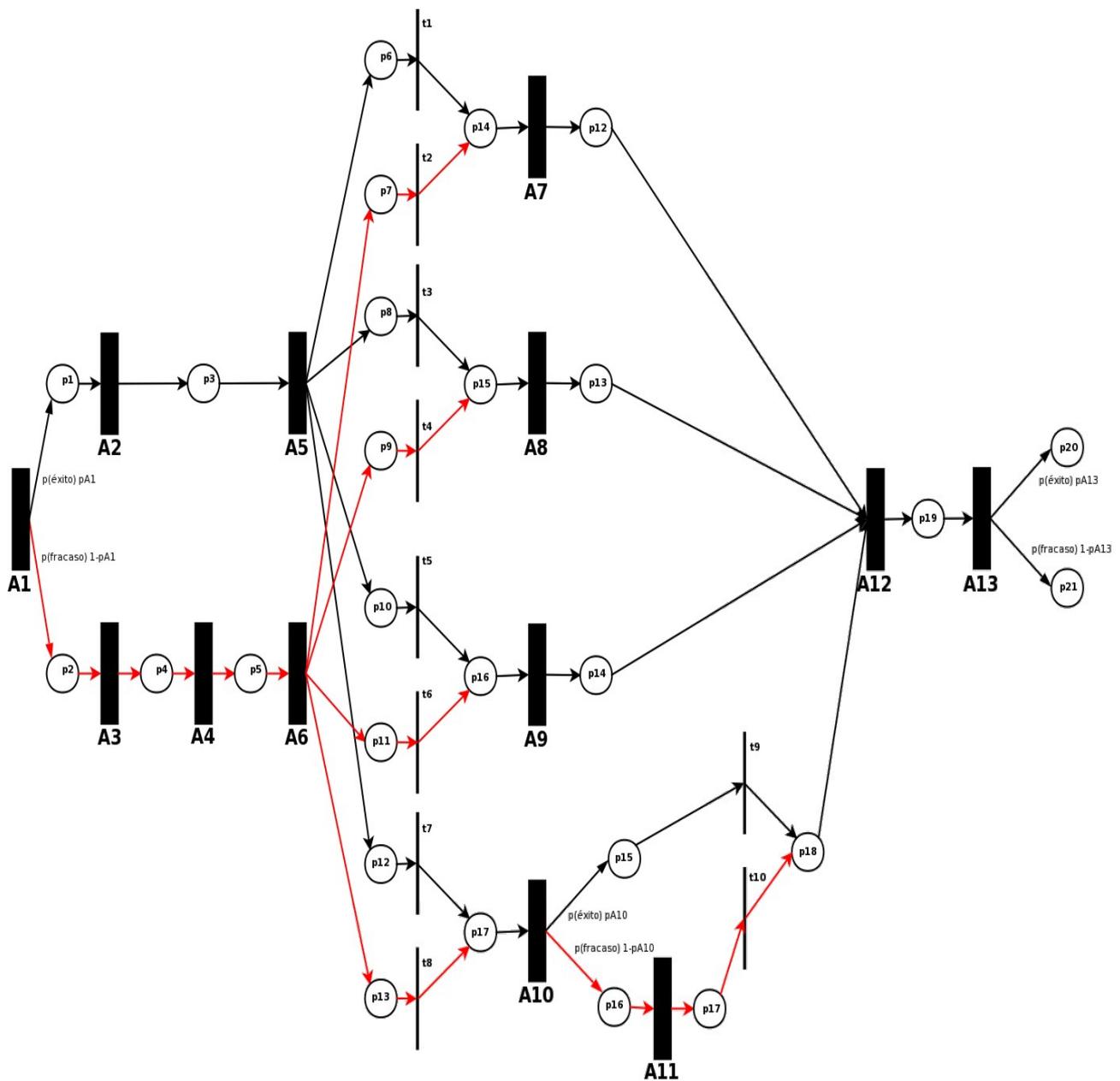


Figura 10: Red del proyecto con planes de contingencia

Las líneas rojas discontinuas indican el plan de contingencia. El plan de contingencia se lanza en el caso de fallo de la actividad A₁ (punto rojo). Las cuatro actividades que se ejecutan en paralelo son

necesarias en cualquier caso, así que su activación refleja con un símbolo de puerta OR los dos eventos que pueden activarlas. El hecho de que el plan de contingencia incluya otra actividad correctora posterior, A_3 , hace necesaria una función de activación compuesta en la actividad A_{12} .

Este modelo permite no sólo comprender gráficamente, de un vistazo, el desarrollo del plan de contingencia, sino calcular con exactitud tiempos y datos como la esperanza o el rango del tiempo de terminación del proyecto o del tiempo de comienzo de alguna de sus actividades. Incluso aspectos más recónditos, como la probabilidad de conflicto por los recursos entre la actividad de contingencia A_{11} y la actividad normal A_7 pueden ser calculados para tomar las medidas oportunas con antelación.

3.3.3 Los ciclos repetitivos

Los estudios clásicos sobre los grafos de actividades en la gestión de proyectos, ya sea grafos de actividad en los nodos [Roy 1962] o grafos de actividad en los arcos [Malcolm et al. 1959], exigen que el grafo sea dirigido acíclico. Esto se suele expresar como un grafo dirigido en el cual ningún camino empieza y termina en el mismo nodo.

Para modelar situaciones como las respuestas a defectos de calidad que implican la repetición de ciertas secuencias de actividades, nosotros hemos considerado interesante relajar el requisito que impide la formación de ciclos. Permitimos indicar entre las actividades que disparan la función de activación de una dada, alguna que sea a su vez sucesora suya de orden n . Es decir, puede darse el caso de que $\exists A_i, A_j \mid A_j \in S_i^* \wedge A_j \in P_i^*$ donde S_i^* y P_i^* indican respectivamente los conjuntos de los sucesores y predecesores de cualquier orden de A_i . Esta característica permite la existencia de ciclos o bucles de repetición dentro del grafo. Por ejemplo, en la figura 11:

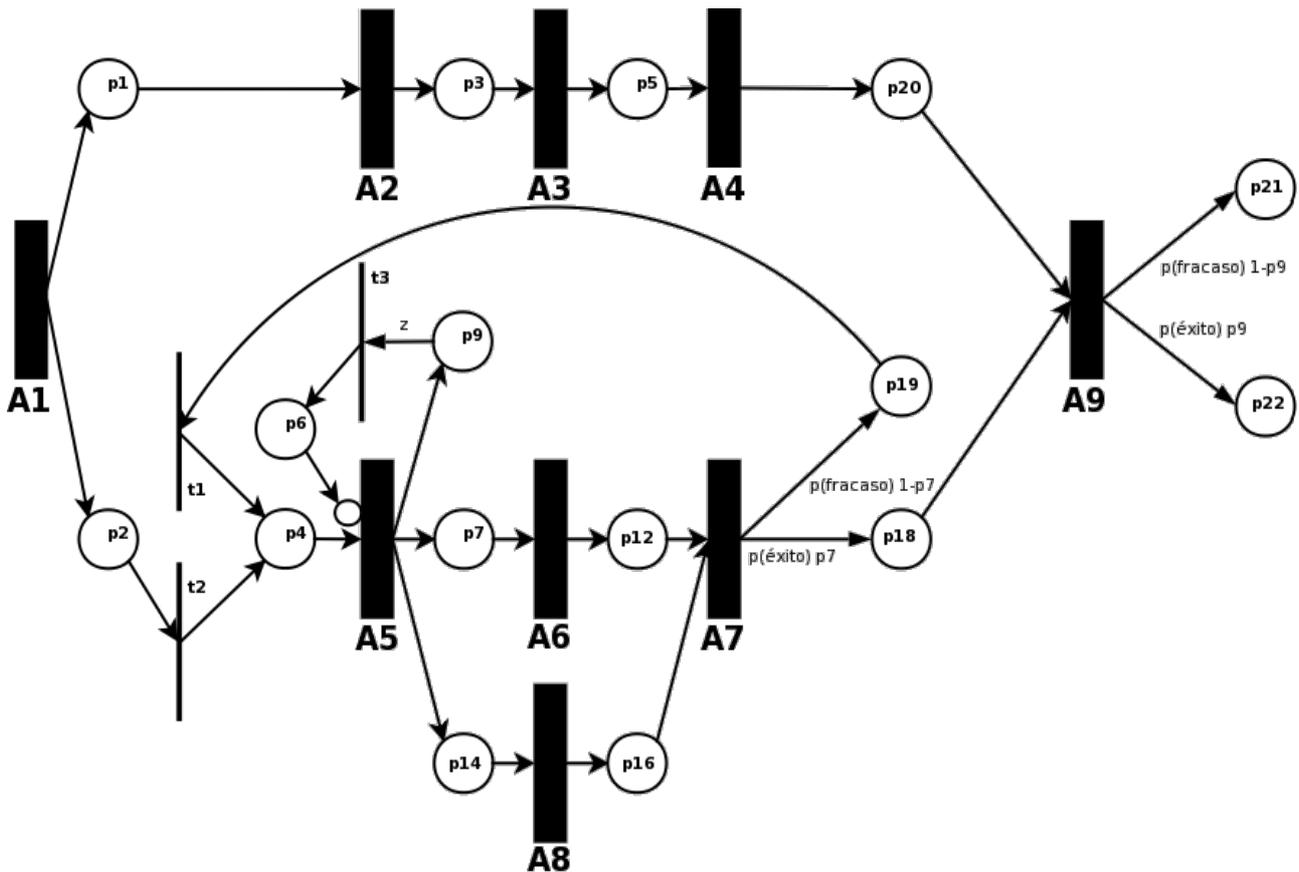


Figura 11: Proyecto con ciclos

La prelación $\langle A_7, A_5 \rangle$ establece un ciclo en el grafo, que afecta a las actividades A_7 y A_5 , pero también a las actividades A_6 y A_8 . La utilidad de este tipo de ciclos estriba en que permiten expresar de manera natural aquellas situaciones en las que se torna preciso repetir parte del trabajo realizado. En ingeniería, el director de proyecto puede prever una vuelta atrás para repetir desde la fase de diseño. En construcción, a veces es preciso acometer una actividad de derribo y volver a emprender el levantamiento de un muro. Asociado a estos ciclos, aparece el concepto de control de calidad, que nosotros representamos mediante la variable aleatoria fijada a la terminación exitosa o fracasada de la actividad. Es este control el que dispara en ciertos casos el ciclo de repetición.

Por conveniencia, definimos P'_i como el conjunto de predecesores inmediatos de A_i excluidos los arcos de retroalimentación, y P''_i el conjunto de los predecesores inmediatos a través de dichos arcos. $P_i = P'_i \cup P''_i$. Análogamente, tenemos S'_i y S''_i . $S_i = S'_i \cup S''_i$. Definimos también A_{RD} como el conjunto de actividades a las que llega un arco de retorno, y A_{RO} el conjunto de actividades de las que sale un arco de retorno. En el ejemplo de la , $A_{RD} = \{A_7\}$ y $A_{RO} = \{A_5\}$. Es necesario que $A_{RD} \subset A_{OR}$ para que el proyecto sea realizable.

En los casos en los que los ciclos están presentes, la función de activación de determinada actividad

A_i , que se define como $\vec{\alpha}_i = f_i(\vec{E}_i, \vec{t}_i, z_i)$, debe evaluarse antes de que algún ξ_j , el evento de activación que corresponde $\xi_j \leftarrow \vec{t}_j : A_j \in P_i$ al vector de variables aleatorias \vec{t}_j de terminación de la actividad A_j , haya podido ser definido. Aunque puede definirse z_i distinto por cada actividad, por simplificar la explicación asumiremos que todas las funciones de activación utilizan la cota máxima $z = \max_{\forall A_i} [z_i]$. La referencia a eventos de activación incompletos nos obliga a implementar una función voraz de evaluación de f_i , del mismo tipo que la evaluación de actividades OR. Por la misma razón, la actividad debe recordar cuál fue la configuración de evento que produjeron sus anteriores activaciones, para evitar volver a dispararse sin que haya un cambio que indique una vuelta atrás por el ciclo. Así mismo, para volver a disparar actividades como A_6 , A_7 u A_8 del ejemplo, actividades que pertenezcan al ciclo sin estar activadas por el arco de vuelta atrás, es necesario que la actividad anterior propague una señal que indique el nuevo ciclo.

La solución que proponemos consiste en contraponer la actividad teórica A_i con la actividad simulada en el turno q . Elementos muestrales de la actividad teórica en el turno q forman el conjunto

$$F_i^{[q]} = \{A_{i1}^{[q]}, \dots, A_{iz}^{[q]}\} \quad z \in \mathbb{N}, z \geq 0 \quad [\text{f64}]$$

3.3.3.1 Algoritmo de despliegue de las iteraciones en los ciclos

El proceso de creación del conjunto $F_i^{[q]}$ sigue el algoritmo:

1. Definir la ordenación topológica T del conjunto de actividades A , excluyendo los arcos de retroalimentación.
2. Tomar la primera actividad de T , $A_i \leftarrow A_1$. Iniciar el número de ciclo $k \leftarrow 1$.
3. Si $F_i^{[q]} = \emptyset$, evaluar la función de activación excluyendo los arcos de retroalimentación. Crear elemento muestral y añadirlo. $F_i^{[q]} \leftarrow \{A_{i1}^{[q]}\}$
4. Si $k < z$, evaluar la función de activación teniendo en cuenta todas las relaciones de precedencia.
5. Si $k = z$, evaluar la función de activación teniendo en cuenta todas las relaciones de precedencia, pero deshabilitar los ciclos de retorno.
6. Evaluar el desarrollo y terminación de la actividad simulada. $F_i^{[q]} \leftarrow F_i^{[q]} \cup \{A_{ik}^{[q]}\}$
7. Si los ciclos de retorno están habilitados, $\forall A_j \in S_i$ volver al paso 3 en el orden de T , incrementando k en cada arco de retroalimentación.
8. Si no, $\forall A_j \in S'_i$ volver al paso 3 en el orden de T .

3.3.3.2 Cálculo de tiempos más tempranos

Debido a las iteraciones, el cálculo de los tiempos tempranos debe ser modificado. Debemos calcular varios tiempos de comienzo tempranos para cada actividad A_i , que denominaremos sucesivamente $t_{i,1}, \dots, t_{i,z}$. Para ello utilizamos el siguiente algoritmo:

1. Definir la ordenación topológica T del conjunto de actividades A, excluyendo los arcos de retroalimentación.
2. Tomar la primera actividad de T, $A_i \leftarrow A_1$. Iniciar el número de ciclo $k \leftarrow 1$, y $t_{i,1} = 0$.
3. Si $A_i \in A_{AND}$, $t_{i,k} = \max_{A_j \in P_i} [t_{j,k} + t_{ji,k}]$
4. Si $A_i \in A_{RD}$,
5. Si $k = 1$, $t_{i,1} = \min_{A_j \in P_i} [t_{j,1} + t_{ji,1}]$
6. En otro caso, $t_{i,k} = \min_{A_j \in P''_i} [t_{j,k-1} + t_{ji,k-1}]$
7. Si $A_i \in A_{OR} \wedge A_i \notin A_{RD}$, $t_{i,k} = \min_{A_j \in P_i} [t_{j,k} + t_{ji,k}]$
8. $\forall A_j \in S_i$ volver al paso 3 en orden de T, incrementando k en cada arco de retroalimentación.

Si la función lógica de activación es compleja, aplicaremos el algoritmo de forma análoga sobre los lugares que corresponden a las funciones lógicas más simples.

3.3.3.3 Cálculo de tiempos más tardíos

Para el cálculo de los tiempos más tardíos, procedemos a la inversa del algoritmo 3.3.3.2, recorriendo la ordenación T desde la actividad de fin de proyecto hacia atrás, y recorriendo los ciclos desde z hasta 1. Igualmente, partimos de la ordenación topológica T del conjunto de actividades A, excluyendo los arcos de retroalimentación. Posiblemente, nos encontraremos con instancias de proyectos en las que $\exists F_i : |F_i| < Z$. Esto puede ser debido a que estas actividades antecedan al ciclo ($A_{RD} \subset S_i^* \wedge A_{RO} \subset S_i^*$), o por el contrario, a que corran en paralelo a él. En este último caso, interpretamos los elementos desde el final, de modo que las actividades que para el cálculo de los tiempos tempranos no se han visto afectadas por un bucle, se alineen con el último ciclo.

1. Tomar la última actividad de T, $A_i \leftarrow A_n$. Iniciar el número de ciclo $k \leftarrow z$, y $t_{i,z}^* = 0$.
2. Si $A_i \notin A_{RO}$
3. Si $\exists A_{ik} \in F_i$, $t_{i,k}^* = \min_{A_j \in S_i} [t_{j,k}^* - t_{ij,k}]$
4. En otro caso, si corre en paralelo, $t_{i,1}^* = \min_{A_j \in S_i} [t_{j,k}^* - t_{ji,k}]$. Este cálculo no se repite para otro valor de k. Si antecede al ciclo, ignorar y continuar.

5. Si $A_i \in A_{RO}$, $t_{i,k}^* = \min_{A_j \in S'_i} [t_{j,k+1}^* + t_{ji,k+1}]$
6. $\forall A_j \in P_i$ volver al paso 2 en orden de T, menguando k en cada arco de retroalimentación.

3.4 Factores de riesgo comunes a ciertas actividades

Un aspecto que hay que considerar en los ciclos es que existen factores de riesgo que son comunes a las actividades que se repiten. Por ejemplo, los problemas de diseño pueden afectar a las actividades de programación. Si es necesario repetir una actividad de programación, y los problemas de diseño no se han corregido, este factor de riesgo afectará también a las repeticiones de la actividad. Si estos tipos de dependencias en los riesgos no se incluyen en el modelo, la simulación de Monte Carlo subestimaré la incertidumbre asociada a la duración de las actividades incluidas en los ciclos.

Si $\delta_{ij}, j < k$, es la variable aleatoria que indica la duración de la actividad A_i en el ciclo j y en el supuesto caso de que la actividad realmente se active, es preciso calcular la variable δ_{ij+1} . La interpretación más sencilla es suponer que $\delta_{ij+1} = c(j)\delta_{ij}$, donde $c(j)$ es una función discreta $c: \mathbb{N} \rightarrow \mathbb{R}^+$. Esto permite al planificador especificar una primera variable aleatoria para la duración de la tarea, pongamos $N(12, 1,2)$, e indicar que se emplearían en las siguientes repeticiones unos tiempos que serían 0,75, y 0,5 de la ejecución anterior, respectivamente. Es decir, se pueden tratar todas las $\delta_{ij}, j > 1$ como variables endógenas o variables perfectamente correlacionadas con δ_{i1} .

Particularmente [Smith; Eppinger 1997a] proponen $c(1) = \alpha(j)$, $c(j) = 1 \forall j > 1$ para un modelo con δ_{i1} determinado, basándose en que la mayor parte de los efectos del aprendizaje se aplican en la primera iteración, y en que no hay grandes avances después de la primera. También argumentan que es raro que se produzcan varias ($j \geq 2$) iteraciones.

Más realista es considerar que las variables δ_{ij} están altamente correlacionadas entre sí. El problema surge en la caracterización del coeficiente de correlación ρ_{ij} . En general, es muy difícil describir este coeficiente, incluso en aquellos entornos de trabajo donde exista un registro adecuado de pasados proyectos que pudieran contener ciclos homólogos de actividades. Si la dependencia entre variables es lineal, se puede usar la correlación del producto-momento.

$$\rho(i, j) = E[(X_i - \mu_i)(X_j - \mu_j)] / (\sigma_i \sigma_j) \quad [f65]$$

Si existen datos históricos, los datos se obtienen mediante la correlación de Pearson⁵⁵.

Algunos investigadores han definido un juego de correlaciones subjetivas aplicadas a los costes en la gestión de proyectos como altas, moderadas o bajas. Estas correlaciones se trasladan a {0,8, 0,45, 0,15} [Touran 1993] o {0,85, 0,55, 0,25} [Chau 1995]. Aun así, para caracterizar la dependencia entre variables aleatorias en el caso de repetición de actividades, habría que especificar las correlaciones de todas las repeticiones de la actividad con cada una, esto es, k^2 coeficientes de correlación.

El coeficiente de correlación ρ_{ij} puede ser positivo o negativo. Observando la serie de δ_{ij} en su conjunto podemos afirmar que si P_i^+ son los coeficientes positivos y P_i^- son los coeficientes negativos, cuando $|P_i^+| \sim |P_i^-|$ menor será $\text{Var}\left(\sum_{j=0}^{k-1} \delta_{ij}\right)$.

En los casos en los que las variables δ_{ij} no sean normales, o su dependencia no sea lineal, nos planteamos emplear la correlación fractil [Phoon et al. 2004], o, partiendo de datos históricos, el coeficiente de correlación de rangos no-paramétrico ρ de Spearman. No obstante, es difícil especificar el grado de dependencia mediante la correlación de rangos. Igualmente nos enfrentamos a una matriz de correlación con k^2 elementos. Además, la matriz de correlación de rangos debería ser definida positiva, y esto no siempre se consigue.

Otra opción sencilla y aceptable [Chapman; Ward 2000] consistiría en asumir la independencia en un turno, asumir una correlación perfecta en la segunda vuelta e interpolar entre ambos valores durante el resto de turnos, de modo que los valores esperados muestren en su conjunto un nivel de dependencia conservador, pero apropiado. No hemos encontrado justificación teórica suficiente para esta aproximación, y los propios autores recomiendan encarecidamente alguna forma más compleja en esta línea.

Por ello todo, en los casos de repetición de actividades donde detectamos una dependencia entre las variables aleatorias ligadas a la duración de la actividad en cada repetición, postulamos la existencia de un factor de riesgo común ϕ_i que tratamos como una variable latente. Asumimos, por lo tanto, que existe una independencia condicional entre las otras variables, es decir, que conocido el valor del factor de riesgo, se comportan como independientes entre sí.

Suponemos también que las distribuciones conjuntas ϕ_i, δ_{ij} son distribuciones bivariantes cuyas marginales son uniformes en [0,1]. Específicamente, usamos la distribución de la banda diagonal.

55 El coeficiente de correlación de Pearson, descrito por primera vez por Francis Galton, se aplica especialmente bien

a variables normales.
$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n-1) s_x s_y}$$
 donde s_x y s_y son las desviaciones estándares.

Las ventajas de esta distribución se discuten en [Dorp; Duffey 1999]:

1. El método de muestreo bivariado es muy eficiente
2. El parámetro de la distribución puede ser estimado fácilmente

El parámetro de la distribución de la Banda Diagonal se obtiene a partir del porcentaje ϵ de explicación del factor de riesgo común. El porcentaje de explicación responde a la pregunta de qué porcentaje de nuestra incertidumbre respecto de la duración de la actividad desaparece conociendo el valor exacto del factor de riesgo latente. Si el porcentaje es 0%, significa que el factor de riesgo no afecta a dicha duración. El parámetro θ se estima entonces como:

$$\theta = \sqrt{\epsilon/100} \quad [f66]$$

También podrían definirse las variables δ_{ij} como variables independientes entre sí, pero si así fuera, sería más lógico pensar que, en realidad, no estamos tratando con la repetición de una actividad, sino con la ejecución de otra actividad similar. En estos casos, preferimos modelar las actividades en cuestión como actividades independientes. En cualquier caso, seguimos a [Chapman; Ward 2000] al afirmar que la independencia por sí sola no es una opción por defecto razonable.

La varianza de variables estocásticas, como la duración total del proyecto, la duración total de la actividad A_i en todas sus repeticiones o el consumo de algún recurso, se ve incrementada cuando se definen estas correlaciones entre las variables estadísticas de las duraciones de las actividades en los sucesivos ciclos.

3.5 Solución de la red mediante la simulación de Monte Carlo

Como hemos introducido en el apartado 1.3, “Notación y conceptos”, el método de Monte Carlo es un método de muestreo aleatorio simple. Esto quiere decir que si el número de iteraciones no es lo suficientemente alto, es posible que muestreen en exceso algunos segmentos de la distribución que se quiere replicar, y que se muestreen en defecto otros segmentos. Para evitarlo, hemos utilizado un número elevado de iteraciones en todos los experimentos.

Nuestro modelo de simulación es un modelo estocástico y continuo. En general, hemos preferido modelar la duración de las actividades mediante funciones de distribución continuas, ya que entendemos que responden mejor al comportamiento del mundo real. Ya hemos tratado en el apartado 3.2.2.1, “La duración probabilística”, las dificultades inherentes a la estimación de estas variables. Ahora nos centraremos en su tratamiento de cara a la solución del problema por simulación.

Para utilizar el método de Monte Carlo es preciso que las variables aleatorias tengan una estructura probabilística conocida. Esto significa que tanto para las variables exógenas ligadas a la duración de las tareas $\vec{\Delta}=\{\vec{\delta}_1, \dots, \vec{\delta}_n\}$, como para las variables ligadas a su estado de terminación $\vec{\mathcal{F}}=\{\vec{\tau}_1, \dots, \vec{\tau}_n\}$, tenemos que buscar la manera de generar unas muestras con la distribución correcta. Para ello, contamos con unas funciones de transformación g y h que nos dan $\vec{\Delta}=g(\mathcal{P})$ y $\vec{\mathcal{F}}=h(\mathcal{P})$ a partir de 2n números aleatorios básicos e independientes $\mathcal{P}=\{\rho_1, \dots, \rho_{(2n)}\}$. Estos números aleatorios básicos son los provistos por la clase Java java.util.random:

1. .nextFloat() Coma flotante, un número aleatorio uniformemente distribuido entre [0.0,1.0[
2. .nextGaussian() Campana de Gauss, un número aleatorio normalmente distribuido según la distribución normal estándar N(0,1)

La clase java.util.random usa una serie de números pseudo-aleatorios a partir de una semilla de 48 bits. A partir de la primera semilla, un generador lineal congruente va produciendo números pseudo-aleatorios, según la fórmula de [Knuth 1969] volumen 2 sección 3 apartado 2.1. Este tipo de generadores se distinguen porque el número pseudo-aleatorio a generar es función de todos los números precedentes:

$$x_{j+1}=f(x_j, x_{j-1}, \dots, x_1) \quad [f67]$$

En realidad, el generador utilizado en java.util.random es del tipo:

$$x_{j+1}=(ax_j+c) \bmod m \quad [f68]$$

Por lo tanto, cada número pseudo-aleatorio obtenido es función de los parámetros multiplicador, incremento y módulo, y del número pseudo-aleatorio anterior. La secuencia de números pseudo-aleatorios es periódica. En la implementación de java.util.random el período m depende de los bits de la máscara, esto es $m=2^{48}$. La simulación de Monte Carlo con miles de turnos (típicamente Q = 10.000), con decenas de actividades y unas pocas variables aleatorias por actividad está claramente por debajo de este límite, con lo cual la secuencia pseudo-aleatoria es válida para el uso que le vamos a dar.

A partir de ellos obtenemos una muestra de una variable δ_i determinada mediante las siguientes funciones de transformación:

1. Normal: Si δ_i tiene una distribución normal $N(\mu, \sigma^2)$, $g(\rho_i)=\rho_i\sigma+\mu$ donde ρ_i es un número aleatorio de campana de Gauss
2. Uniforme: Si δ_i tiene una distribución uniforme U(a,b), $g(\rho_i)=\rho_i(b-a)+a$ donde

ρ_i es un número aleatorio de coma flotante

3. Lognormal: Si δ_i tiene una distribución lognormal $L(\mu, \sigma^2)$, $g(\rho_i) = e^{\rho_i \sigma + \mu}$ donde ρ_i es un número aleatorio de campana de Gauss

En la literatura y la práctica profesional se han utilizado otras funciones de distribución para caracterizar la duración aleatoria de la actividad. Se puede encontrar una discusión sobre el método para elegir la función de distribución y su estimación en [Williams 1992b]. Nuestro procedimiento es independiente de la función de distribución particular que se elija para modelar estas variables, así que con nuevas funciones de transformación podríamos obtener muestras para distribuciones beta PERT [Malcolm et al. 1959], beta PERT de dos puntos [Golenko-Ginzburg 1988], triangular [Elmaghraby 1977], [Johnson 1997], gamma [Lootsma 1966], [Lootsma 1967] o de Berny [Berny 1989].

A partir de ellos obtenemos una muestra de una variable τ_i determinada mediante la siguiente función de transformación:

1. Bernoulli: Si τ_i tiene una distribución de Bernoulli $B(\mu)$
- $$h(\rho_j) = \begin{cases} true \Leftrightarrow \rho_j < \mu \\ false \Leftrightarrow \rho_j \geq \mu \end{cases} \text{ donde } \rho_j \text{ es un número aleatorio de coma flotante}$$

Si la variable δ_i no es independiente, sino que depende de un factor de riesgo común ϕ_k , el método de muestreo es algo más complejo.

1. Muestreamos u desde ϕ_k utilizando un número aleatorio de coma flotante ρ_k
2. Obtenemos los límites de la Banda Diagonal $a = u - 1 + \theta$ $b = u + 1 - \theta$ donde θ es el parámetro de la distribución conjunta
3. Muestreamos ρ'_i según una distribución uniforme $U(a,b)$ $\rho'_i = g'(\rho_k, \rho_l) = \rho_l(b-a) + a$ donde ρ_l es un número aleatorio de coma flotante
4. Ajustamos las zonas terminales de la Banda Diagonal para obtener ρ_i

$$\rho_i = g(\rho_k, \rho_l) = \begin{cases} \rho_i = -\rho'_i \Leftrightarrow \rho'_i < 0 \\ \rho_i = \rho'_i \Leftrightarrow \rho'_i \in [0,1] \\ \rho_i = 1 - \rho'_i \Leftrightarrow \rho'_i > 1 \end{cases}$$

5. Muestreamos la variable δ_i a partir de ρ_i mediante la función de transformación apropiada para su distribución de probabilidad

A partir de las funciones de transformación, e instanciando los números aleatorios, obtenemos las muestras $\vec{D} = g(\vec{R})$ y $\vec{T} = h(\vec{R})$. Este plan de muestreo lo repetimos Q veces, las veces que el usuario considere necesarias. En el capítulo 5, "Experimentación", utilizaremos $Q = 10.000$. Las muestras obtenidas en cada paso de simulación son independientes e idénticamente distribuidas.

3.6 Exégesis de los resultados

En este apartado, planteamos algunas consideraciones sobre la salida de la simulación del modelo, es decir, sobre los resultados que podemos extraer una vez se han tomado las muestras mediante los métodos explicados en el apartado anterior.

Los datos que proporciona el modelo pueden dividirse, grosso modo, en datos obtenidos en la fase de secuenciación⁵⁶, y datos obtenidos en la fase de planificación restringida por los recursos⁵⁷, siguiendo la terminología de [Meredith; Mantel 1995].

Entre los datos obtenidos en la fase de secuenciación, se encuentran los tiempos de comienzo s_{ij} y final f_{ij} de cada actividad en cada ronda de simulación. Estos tiempos cumplen las restricciones de prelación; por lo tanto, se encuentran entre los tiempos tempranos y tardíos de la actividad en cada ronda.

$$es_{ij} \leq s_{ij} \leq ls_{ij} \quad [f69]$$

$$ef_{ij} \leq f_{ij} \leq lf_{ij} \quad [f70]$$

En la fase de secuenciación, la estrategia seguida por cada director de proyecto o cada compañía decide dónde se colocan exactamente las actividades, dentro de este intervalo. Generalmente, se prefiere utilizar una estrategia remolona⁵⁸ y colocar las actividades lo más tarde posible, de modo que se arriesgue menos esfuerzo contra el evento de interrupción y abandono del proyecto. A veces, existe también una razón financiera, al retrasar el gasto lo más posible. Otros, sin embargo, prefieren una estrategia más voraz⁵⁹, y comienzan las actividades lo antes posible, a medida que sus actividades predecesoras vayan acabando. Con esta estrategia, se disminuye la probabilidad de incurrir en retrasos debido a los caminos de ejecución acríticos. [Tavares et al. 1998] definieron para este problema el factor de holgura α , de manera que el tiempo efectivo de comienzo de cada actividad se exprese en términos de α multiplicada por la holgura de A_i , Δ_i .

$$s_i(\alpha) = es_i + \alpha \Delta_i \quad [f71]$$

La utilización del factor de holgura permite realizar un análisis de escenarios como parte de un sistema de ayuda a la decisión. Conviene recalcar, no obstante, que en el modelo de [Tavares et al.

56 *time phasing the project*

57 *scheduling subject to finite capacities*

58 *backward loading*

59 *front loading*

1998] el primer uso del factor de holgura se realiza antes de la simulación, por lo que

$$s_i(\alpha)^{[q]} = \begin{cases} es_i^{[q]} & \Leftrightarrow es_i^{[q]} > s_i(\alpha) \\ s_i(\alpha) & \Leftrightarrow es_i^{[q]} \leq s_i(\alpha) \end{cases} \quad [f72]$$

$$es_i^{[q]} = \max_{A_j \in P_i} \{s_j(\alpha) + d_j^{[q]}\} \quad [f73]$$

Como última opción de configuración, consideramos que el director de proyecto no tiene ninguna estrategia, o en realidad, tiene una estrategia que no podemos modelar. En este caso de incertidumbre, utilizamos un número aleatorio auxiliar t_i que nos colocará los tiempos de comienzo y final dentro de sus intervalos de manera equiprobable.

Especial relevancia adquieren los datos correspondientes a las actividades finales del proyecto. Sea $F = \{A_x, \dots, A_y\} \neq \emptyset$ el conjunto de actividades cuyo fin exitoso conduce al éxito del proyecto. La duración total del proyecto δ se estima como

$$\bar{\delta} / (\tau = \text{éxito}) = \frac{\sum_{q=0}^Q \min_{A_n \in F, z \leq z_n} \{f_{nz}^{[q]}\}}{Q} \quad [f74]$$

$$\bar{\delta} / (\tau = \text{fracaso}) = \frac{\sum_{q=0}^Q \max_{A_i \notin F, z \leq z_i} \{f_{iz}^{[q]}\}}{Q} \quad [f75]$$

donde $f_{nz}^{[q]}$ representa el tiempo de finalización de A_{nz} , una de las actividades últimas del proyecto, durante el ciclo de simulación q .

Estos son los datos de la fase de temporalización, y sólo son útiles si se puede considerar que los recursos son ilimitados o realmente son suficientes para ejecutar las actividades del proyecto concurrentemente.

La criticidad c_{ij} de determinada actividad A_{ij} se define a partir de la holgura h_{ij} . La criticidad es la probabilidad de que la actividad A_i esté en el camino crítico, y se estima como

$$c_{ij} = \frac{\sum_{q=0}^Q b_{ij}^{[q]}}{Q} \quad [f76]$$

donde b_{ij} se define como

$$h_{ij}=0 \Rightarrow b_{ij}=1 \quad [f77]$$

$$h_{ij} \neq 0 \Rightarrow b_{ij}=0 \quad [f78]$$

La criticidad de determinada actividad nos indica la importancia relativa de la actividad frente a otras actividades del programa. La distribución de la criticidad entre las actividades, sin embargo, nos muestra la rigidez del programa. Si hay un porcentaje alto de actividades con la criticidad por encima de un umbral, esto significa que va a ser difícil reducir el tiempo total del proyecto, ya que los ahorros de tiempo en un camino crítico producen como resultado que otro camino llegue a ser crítico.

De manera similar, extraemos la criticidad $c_{ij}/\alpha_{ij}=true$ de determinada actividad A_i , dado que A_i realmente ocurra. Dado que la criticidad es una medida de la atención que debe dedicarle el equipo del proyecto a una actividad, esta última medida resulta más indicada para la gestión del proyecto.

La robustez de un programa es la cualidad de comportarse bien durante la fase operacional. A pesar de lo detallado que se sea durante la planificación, la ejecución de un proyecto está sujeta a innumerables perturbaciones. Ya hemos estudiado la variabilidad de la duración de las actividades en el apartado 3.2.2.1 “La duración probabilística”, y en el apartado 3.3 “La red generalizada del proyecto”, la variabilidad respecto de la propia red del proyecto, incluyendo tanto la ordenación de las actividades como su instanciación. Además, es posible que se descubra, durante la ejecución, la necesidad de introducir nuevas actividades, o que determinados recursos no estén disponibles o no lo estén en la cantidad presupuesta. Puede suceder que la previsión de la demanda de los recursos en la actividad fuera errónea, o varíe durante el transcurso de ésta. En general, todas las perturbaciones llevan a la necesidad de replanificar el proyecto. Esta replanificación puede ser completa o parcial. En este último caso, hablamos de técnicas de reparación de programas como, por ejemplo, el citado [Józefowska et al. 2003]. Los motivos para reparar un programa son diversos: en proyectos muy grandes, planificados con detalle, se alegan costes computacionales. Además, desde la gestión de los recursos humanos, se habla de la desmotivación que produce la sensación de descontrol cuando el programa anterior ha sido ya comunicado al equipo del proyecto. También hay costes materiales en las contraórdenes, y actividades de renegociación de las condiciones con los proveedores. Nosotros no planteamos ningún algoritmo de reparación del proyecto; nuestra recomendación es replanificar continuamente el proyecto según esté disponible nueva información.

De la comparación entre en programa inicial y el programa realmente ejecutado se extraen diversas medidas de la robustez [Sanlaville 2002]. Nuestro prototipo no dispone de funcionalidad de seguimiento del proyecto, así que este tipo de medidas no son parte de la investigación. Otros, como [Al-Fawzan; Haouari 2005] definen la robustez a priori, en base a la programación inicial, como la suma de holguras libres.

Más alejado de nuestro algoritmo de planificación se halla el trabajo de [Artigues et al. 1999], con el que se busca caracterizar un conjunto de programas compatibles con las restricciones principales, dentro de los cuales una secuencia de actividades es permutable libremente.

3.6.1 La planificación restringida por los recursos

La planificación restringida por los recursos es sólo uno de los problemas a los que tiene que hacer frente la gestión de proyectos. En realidad, es una parte pequeña del problema de los recursos en la gestión por proyectos, como se puede ver en la figura 12:



Figura 12: Niveles en la gestión de recursos

Una vez que se ha obtenido una muestra mediante la simulación de Monte Carlo, aplicamos el algoritmo de ramificación y poda de [Demeulemeester; Herroelen 1992] para conseguir una planificación óptima. El procedimiento DH, como a ellos les gusta llamarlo, sirve para trabajar con recursos renovables, que estén disponibles en cantidad constante durante el proyecto. Además, la demanda de un recurso en una actividad debe también ser constante durante la duración de la actividad.

Antes de aplicar el procedimiento DH, realizamos un par de transformaciones del grafo del

proyecto. La primera transformación es habitual en este tipo de procedimientos, y consiste en añadir una actividad inicial y otra final para cerrar el grafo. La segunda, sin embargo, es un cambio de representación que realizamos para desplegar los ciclos de la red de actividades. Para representar las actividades utilizamos dos representaciones distintas, ambas de actividad en el nodo. En la representación del alto nivel R, la actividad A_i abarca todas las posibles repeticiones de la actividad, y el grafo dirigido puede contener ciclos. La representación de alto nivel R se despliega en la representación de bajo nivel S, donde cada actividad A_i es sustituida por las actividades $\{A_{i1}, \dots, A_{iy}\}$, y $y \leq z_i$. Estas actividades encarnan las sucesivas repeticiones de la actividad A_i , y se encuentran acotadas por el contador z_i , máximo número de repeticiones que la actividad A_i admite. La representación S forma un grafo dirigido acíclico.

A continuación, mostramos la estructura general del algoritmo:

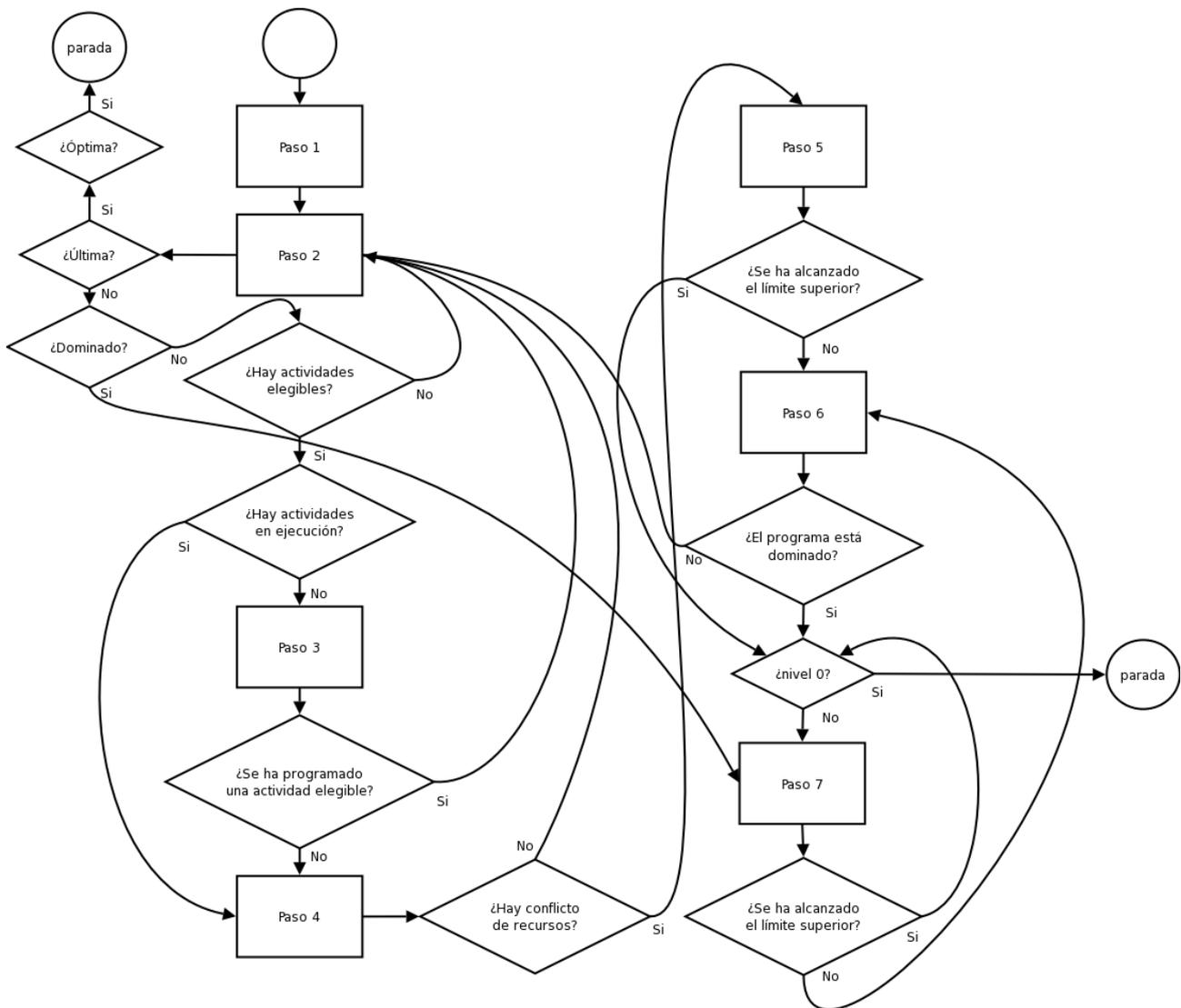


Figura 13: Diagrama de flujo del algoritmo de asignación de recursos

3.6.1.1 Paso 1: Inicialización

Sea $T = 9999$ el límite superior de la duración del proyecto. Sea $p = 0$ el nivel de profundidad en el árbol de ramificación y poda. Sea $m = 0$ el punto de decisión. Para toda actividad A_i se calcula la longitud del restante camino crítico $RCPL_i$. Se inicializan los tiempos de finalización de las actividades $f_i = T$. Se programa la falsa actividad inicial en $f_1 = 0$. El límite inferior $LB(0) = RCPL_1$. Se inicializa el programa parcial $PS = \{A_1\}$, el conjunto de las actividades en ejecución $S = \{A_1\}$ y el conjunto de corte $C = \{A_i \mid pred(A_i) = \{A_1\}\}$. Inicializamos los tiempos de comienzo tempranos de las actividades en C , $s_x = 0$.

3.6.1.2 Paso 2

El siguiente punto de decisión es $m = \min\{f_i \mid A_i \in S\}$. Actualizamos el conjunto de actividades en ejecución $S = S - \{A_j \mid A_j \in S \wedge f_j = m\}$. Si la última actividad programada es la falsa actividad final n , hemos terminado un programa. La duración del programa es $T = f_n$. Si $T = LB(0)$, hemos encontrado la solución óptima, si no, hay que ir al paso 7.

Ahora se comprueba si el conjunto de corte C es dominado por otro conjunto de corte guardado previamente. En ese caso, se va al paso 7, si no, se guarda el conjunto de corte C , el conjunto de las actividades en ejecución S junto con sus tiempos de finalización f_j , y el punto de decisión m .

Finalmente, construimos el conjunto de las actividades elegibles $E = \{A_i \mid A_i \in C \wedge s_i = m\}$. Si $E = \emptyset$, repetimos el paso 2. Si el conjunto de actividades en ejecución $S \neq \emptyset$, saltamos al paso 4, si no, seguimos al paso 3.

3.6.1.3 Paso 3

Para cada actividad elegible $A_i \in E$ contamos cuántas actividades no programadas (no necesariamente actividades elegibles) se pueden poner en ejecución simultáneamente con A_i sin violar ninguna restricción de precedencia o recursos. Si no hay otra, se pone en ejecución A_i : $PS = PS \cup \{A_i\}$, $S = \{A_i\}$, $f_i = m + d_i$ y se actualiza el conjunto de corte $C = C - \{A_i\} \cup \{A_j \mid A_j \in suc(A_i) \wedge pred(A_j) \subset PS\}$. También se actualizan los tiempos de comienzo temprano de las actividades en el conjunto de corte $s_x = f_i$.

Si la actividad elegible $A_i \in E$ puede ser puesta en ejecución simultáneamente con una sola otra

actividad $A_j \in E \mid d_i \geq d_j$ se lanzan ambas: $PS = PS \cup \{A_i, A_j\}$, $S = \{A_i, A_j\}$, $f_i = m + d_i$, $f_j = m + d_j$, y se actualiza el conjunto de corte $C = C - \{A_i, A_j\} \cup \{A_k \mid A_k \in \text{suc}(A_i) \cup \text{suc}(A_j) \wedge \text{pred}(A_k) \subset PS\}$. También se actualizan los tiempos de comienzo temprano de las actividades en el conjunto de corte $s_x = f_i$.

Si alguna actividad ha sido programada en este paso, volvemos al paso 2, si no, continuamos con el paso 4.

3.6.1.4 Paso 4

Ponemos temporalmente todas las actividades elegibles en ejecución: $PS = PS \cup E$, $S = S \cup E$, $f_i = m + d_i \forall i, A_i \in E$. Actualizamos el conjunto de corte $C = C - E + \{A_j \mid \exists A_i \in E : A_j \in \text{suc}(A_i) \wedge \text{pred}(A_j) \subset PS\}$. También actualizamos los tiempos de comienzo temprano de las actividades en el conjunto de corte $s_x = \max\{f_a \mid (a, x) \in H\}$.

Para cada tipo de recurso k se comprueba si $\sum_{i \in S} r_{ik} \leq b_k$. Si en algún tipo de recurso k existe conflicto de recursos, se sigue al paso 5. En otro caso, se vuelve al paso 2.

3.6.1.5 Paso 5

Actualizamos el nivel de profundidad $p = p + 1$. Calculamos para cada tipo de recurso k cuántas unidades se deben liberar para solucionar el conflicto de recursos $c_k = \sum_{i \in S} r_{ik} - b_k$. El conjunto que se debe retrasar lo definimos como $D(p) = \{D_q \subset S \mid \sum_{i \in D_q} r_{ik} \geq c_k \forall k \wedge \text{not} \exists D_v \subset D_q : D_v \in D(p)\}$.

Para cada subconjunto $D_q \in D(p)$ buscamos la actividad de fin más temprano que esté en ejecución y no vaya a ser retrasada $j \in (S - D_q)$. A continuación, definimos el conjunto de pares ordenados $G_q = \{(j, i)\} \forall i \in D_q$ y calculamos el límite de la secuencia crítica L_q .

Entre todos los subconjuntos se elige el del límite L_q^* menor, D_q^* . Se actualiza $D(p) = D(p) - D_q^*$ y se fija $LB(p) = \max\{LB(p-1), L_q^*\}$. Si $LB(p) \geq T$, se salta al paso 7, en otro caso, se almacenan los datos necesarios para una posible vuelta atrás.

3.6.1.6 Paso 6

Creamos una rama nueva. Definimos DS como el conjunto de todas las actividades que comenzaron

antes de m pero deben ser retrasadas $DS = \{A_i \in D_q^* \mid f_i < m + d_i\}$. Introducimos relaciones de precedencia adicionales $H = H \cup G_q^*$. Retrasamos las actividades programadas $PS = PS - D_q^*$, $S = S - D_q^*$, $\forall A_i \in D_q^* f_i = 9999$. Actualizamos el conjunto de corte $C = C + D_q^* - \{A_r \mid A_x \in D_q^* \wedge A_x \in P_r\}$.

Ajustamos los tiempos de comienzo de las actividades que deben ser retrasadas $\forall A_i \in D_q^* s_i = f_j \mid (i, j) \in G_q^*$

Si $DS \neq \emptyset$, utilizamos la regla del dominio sobre el desplazamiento a izquierda. Ésta consiste en considerar que, si las relaciones de precedencia establecidas en niveles superiores del árbol de búsqueda llevan a que A_i sea elegible en el tiempo m , y además la decisión actual ha sido lanzar A_i en m , y además tras retrasar DS se podría desplazar A_i a la izquierda sin causar un conflicto de recursos, esta programación se encuentra dominada. En este caso, se sigue con el paso 7, en otro caso, se vuelve al paso 2.

3.6.1.7 Paso 7: Vuelta atrás

Si hemos llegado al nivel $p = 0$, paramos. Si no, borramos las relaciones de precedencia que hemos introducido en este nivel $H = H - G_q^*$. Si $D(p) = \emptyset$ entonces volvemos con $p = p - 1$ al paso 7.

Elegimos el conjunto $D_q^* \in D(p)$ con menor L_q^* . Actualizamos el conjunto a retrasar $D(p) = D(p) - D_q^*$. Calculamos el límite inferior $LB(p) = \max\{LB(p-1), L_q^*\}$. Si $LB(p) \geq T$ volvemos con $p = p - 1$ al paso 7.

Se recuperan los tiempos de finalización de las actividades, el programa parcial, el conjunto de actividades en ejecución, las actividades del conjunto de corte y sus tiempos de inicio temprano, más el punto de decisión m , y se vuelve al paso 6.

4. Prototipo de software

4.1 Introducción

El modelo de datos del proyecto descrito en el capítulo anterior ha sido plasmado en un prototipo de software para justificar su viabilidad. El software cumple las funciones de planificación de proyectos, actuando de manera interactiva con el ingeniero de planificación.

Este prototipo ha sido implementado en Java 1.4, y en su desarrollo se han utilizado los entornos interactivos JBuilder de Borland, versiones 3, 6, y 7, y Netbeans de Sun Microsystems, versiones 4 y 5.

El prototipo, llamado RIES, es una aplicación java con interfaz gráfica que recibe órdenes del usuario, toma como entrada algunos ficheros con la descripción de la red estocástica de los proyectos a planificar, y produce una salida gráfica y también en varios ficheros, correspondientes a las opciones de planificación que haya elegido el usuario. Además, su comportamiento es modificable en ciertos aspectos mediante un fichero de configuración.

4.2 Descripción del prototipo

Para describir el prototipo RIES utilizaremos la notación UML en el caso de uso básico y los diagramas de clases. En el anexo A, en formato CDROM, utilizaremos la notación de javadoc para describir la interfaz del programador de aplicaciones del prototipo. En el anexo B, en el mismo CDROM, está disponible el código fuente del prototipo y el código intermedio.

En cuanto a las reglas de estilo en UML, hemos preferido colocar las clases externas, las que interactúan con los papeles arriba, de modo que las asociaciones estereotipadas como <<usa>> se puedan leer <arriba> usa <abajo>.

4.2.1 Casos de uso

Los casos de uso son una técnica habitual en el desarrollo de software que se utiliza para capturar y documentar los requerimientos funcionales de los sistemas. Cada caso plantea uno o varios escenarios en los cuales el sistema interactúa con los usuarios descritos como actores. En nuestro

caso, pretendemos mostrar cuál sería el uso esperado del prototipo.

4.2.1.1 Analizar el plazo

Caso de uso: Analizar el riesgo y plazo

Actores: Planificador, RIES

Tipo: Primario

Descripción: El planificador revisa la definición de las actividades para comprobar que tanto la asignación de recursos como la definición probabilística de la duración se encuentran definidas para todas las actividades. Revisa también los límites de los recursos disponibles para el o los proyectos. A continuación, solicita una simulación de n pasos.

Curso normal de los eventos

Acción de los actores

Respuesta del sistema

1. Este caso de uso comienza cuando el Jefe de proyecto solicita un plan ya definido

2. Presenta el estado actual del plan

3. El planificador revisa la función de duración

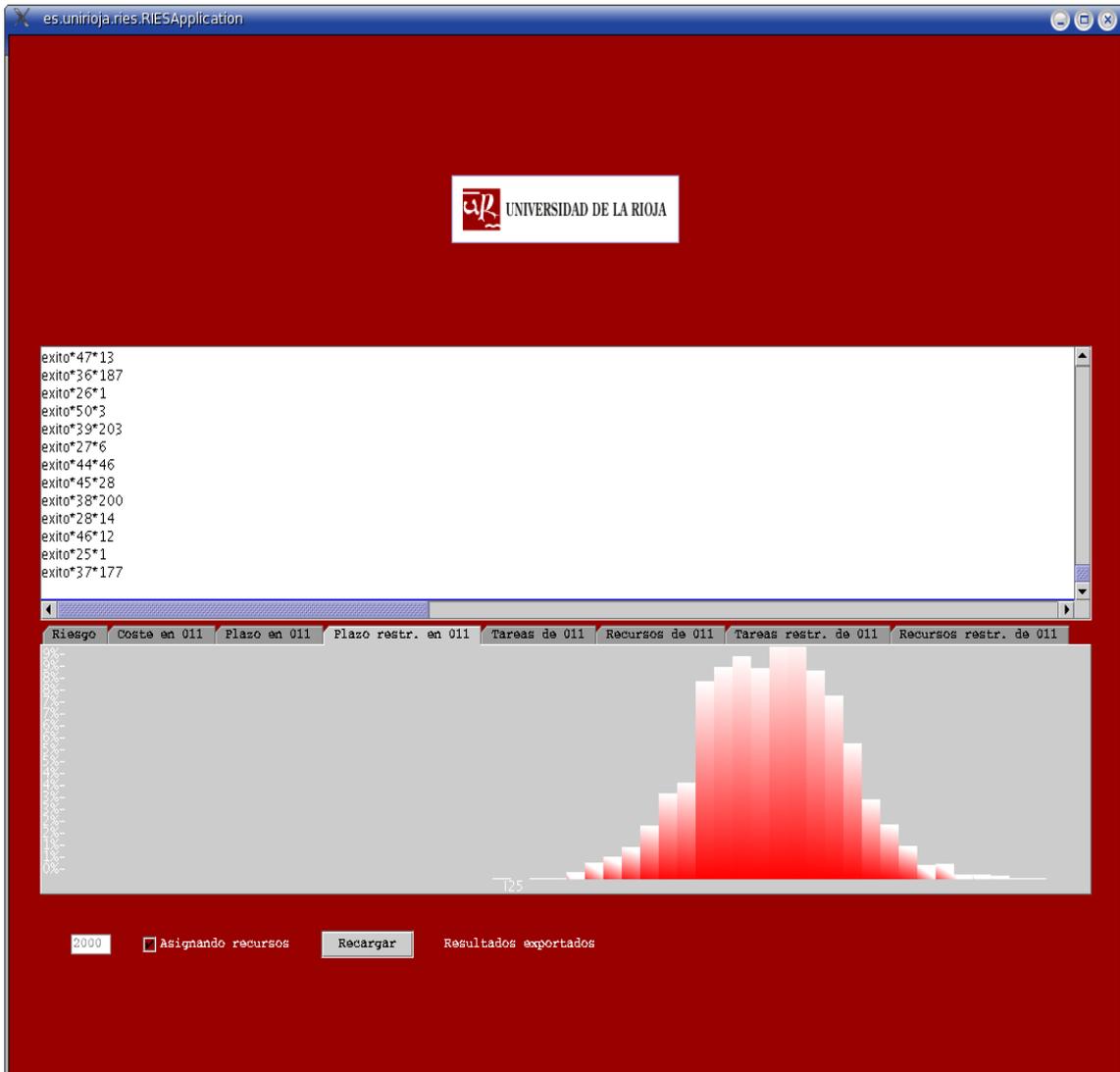
4. El planificador revisa las restricciones de los recursos

5. Al terminar de revisar duraciones, el planificador solicita una simulación de n pasos

6. Realiza la simulación

7. Al llegar al criterio de convergencia, presenta los resultados de la simulación

A continuación se muestra una captura de pantalla del prototipo tras la simulación del experimento 11 con la opción de asignar recursos habilitada. La pestaña muestra la distribución de la fecha de terminación del proyecto, teniendo en cuenta las restricciones de los recursos.



Captura 1: Plazo restringido por los recursos en proyecto 11

4.2.2 Diagramas de clases

A continuación detallamos el diagrama de clases del paquete **es.unirioja.ries**.

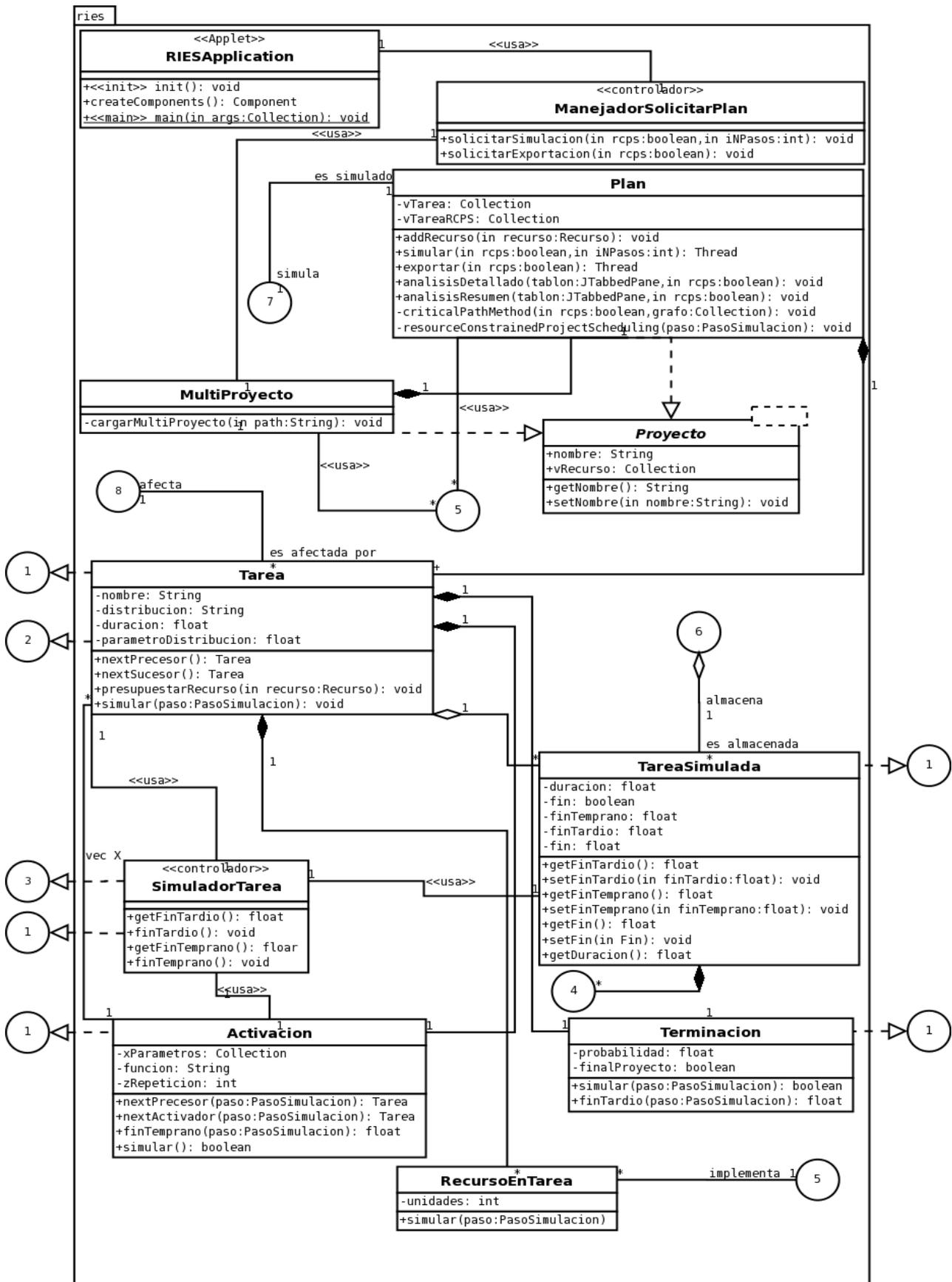


Figura 14: Diagrama de clases de ries.unirioja.es, 1

El paquete **es.unirioja.ries** contiene 24 clases, de las cuales hemos representado en el diagrama de dos páginas anterior las 17 más significativas. El paquete implementa las siguientes interfaces de **java.lang**

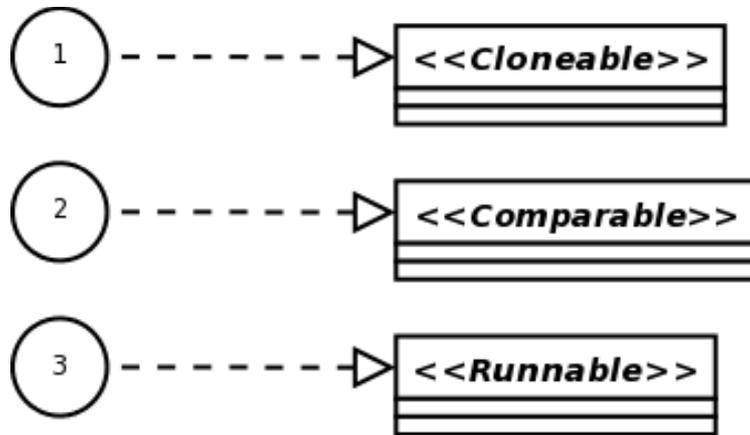


Figura 16: Diagrama de clases del paquete *ries.unirioja.es*, 3

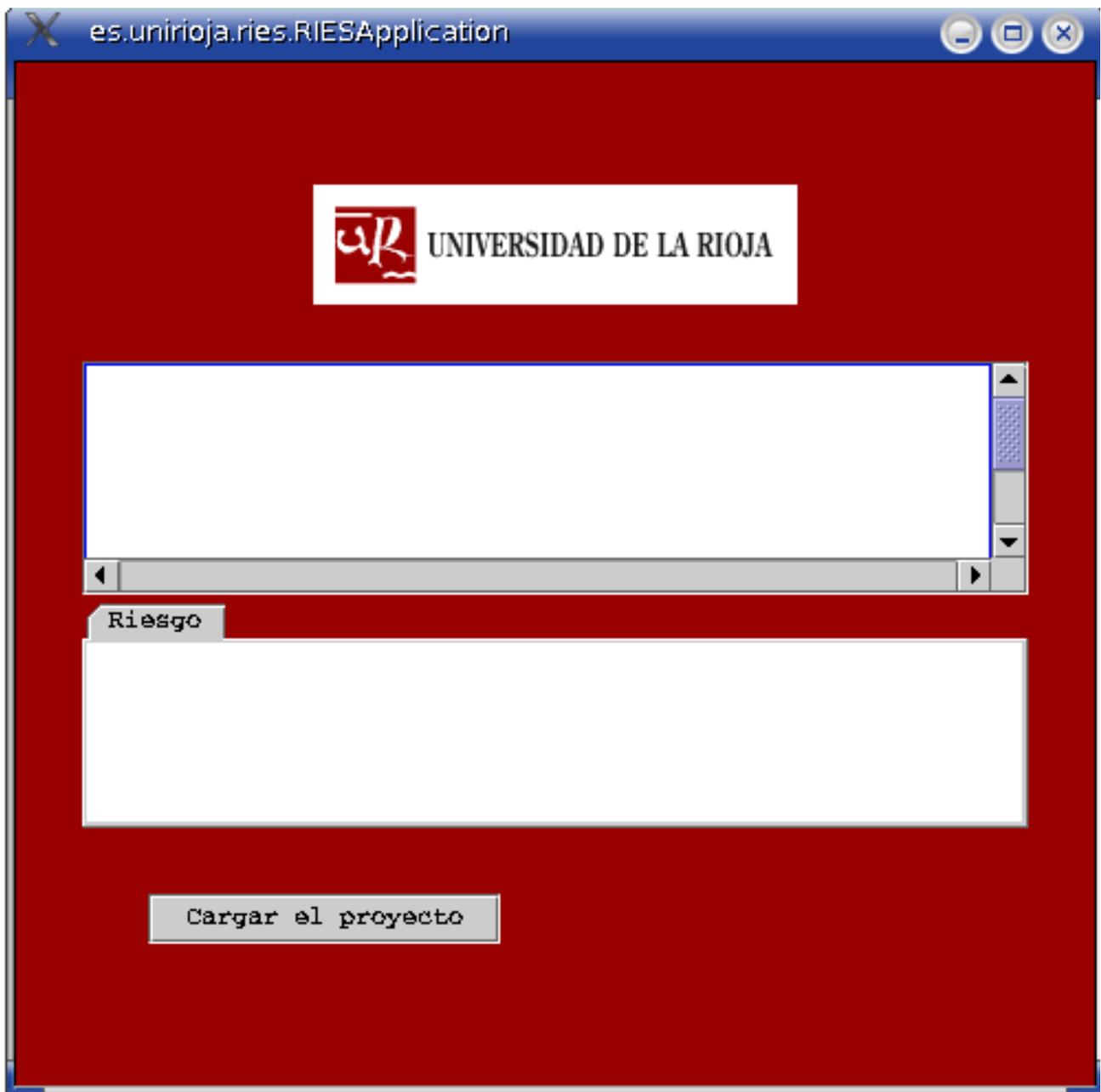
El prototipo incluye también el paquete **es.unirioja.ries.entrada**, responsable de adquirir los datos de los proyectos a partir de ficheros XML. El paquete contiene 22 clases, pero no hemos considerado interesante incluirlas aquí, ya que tienen poca relación con la investigación. Los detalles sobre el formato de los ficheros XML se encuentran en esta sección, en el apartado 4.3, “Formatos de los ficheros”.

El último paquete del desarrollo es **es.unirioja.ries.grafo**. Contiene sólo dos clases, y es el responsable de plasmar los resultados de manera gráfica, mediante diagramas de barras.

A parte del software de producción propia, utilizamos dos clases del paquete `com.byrden.util` desarrollado por David Byrden para su uso gratuito (<http://byrden.com/java/>).

4.2.3 Interfaz gráfica

La interfaz gráfica del prototipo presenta al usuario un área de texto informativo, bajo ella un contenedor de fichas para la salida gráfica del programa, y en la parte inferior una botonera. A continuación podemos ver el aspecto inicial que presenta el prototipo:

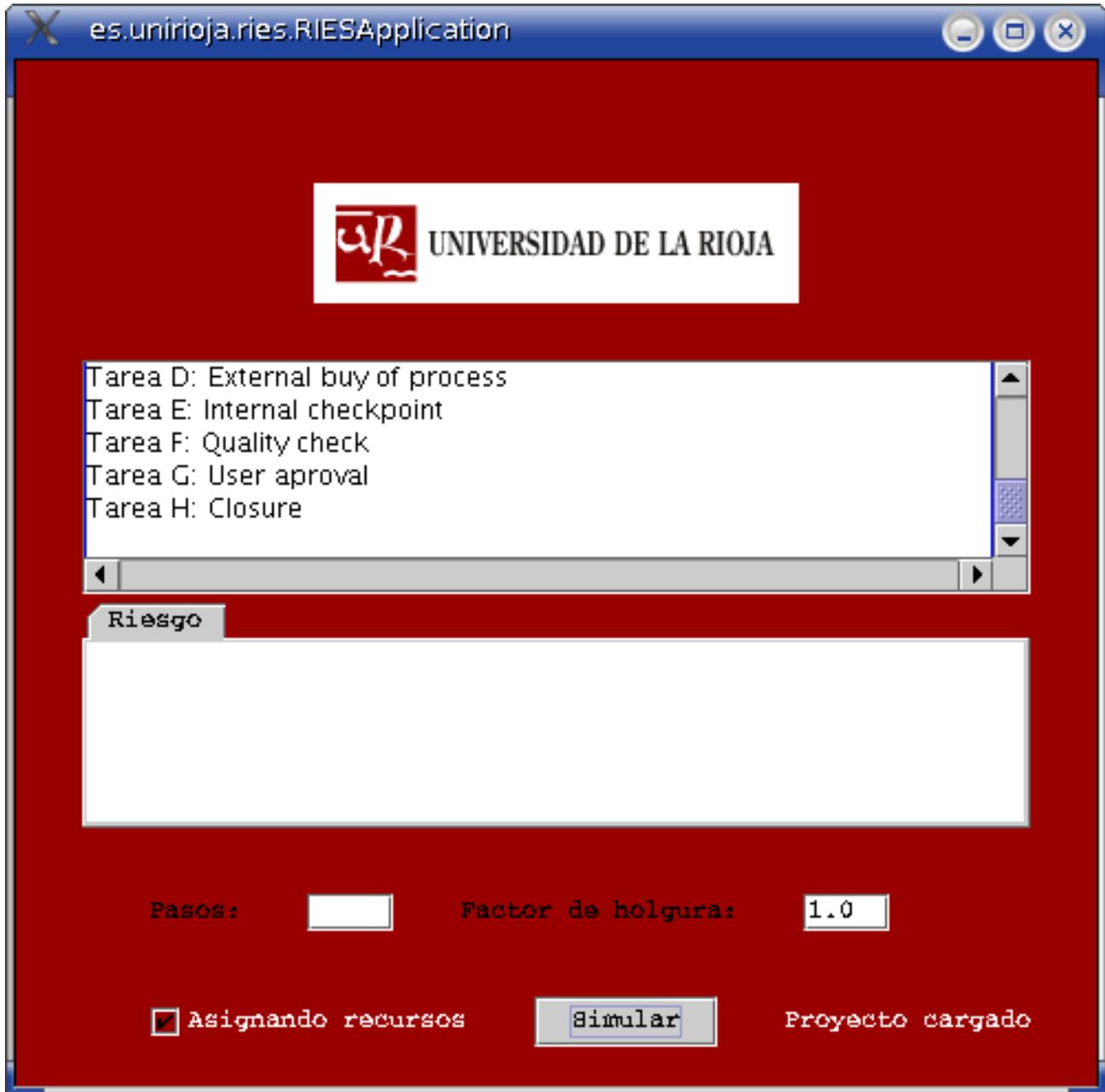


Captura 2: Pantalla inicial de RIES

Pulsando el botón inferior, el prototipo permite introducir la red estocástica del portafolio por medio de una aplicación XML. Véase el formato en el apartado 4.3, “Formatos de los ficheros”. A continuación, la interfaz gráfica muestra al usuario el o los proyectos leídos en el área de texto, bajo ella el contenedor de fichas, aún vacío, y bajo este unos campos de entrada para controlar el comportamiento del planificador:

- Pasos: indica el número Q de rondas de la simulación. Normalmente trabajamos con $Q=10.000$.
- Factor de holgura: factor propuesto por [Tavares et al. 1998] para temporalizar las actividades entre sus tiempos tempranos y tardíos.
- Asignando recursos: si este control está chequeado, se llamará al algoritmo de planificación restringida por los recursos.

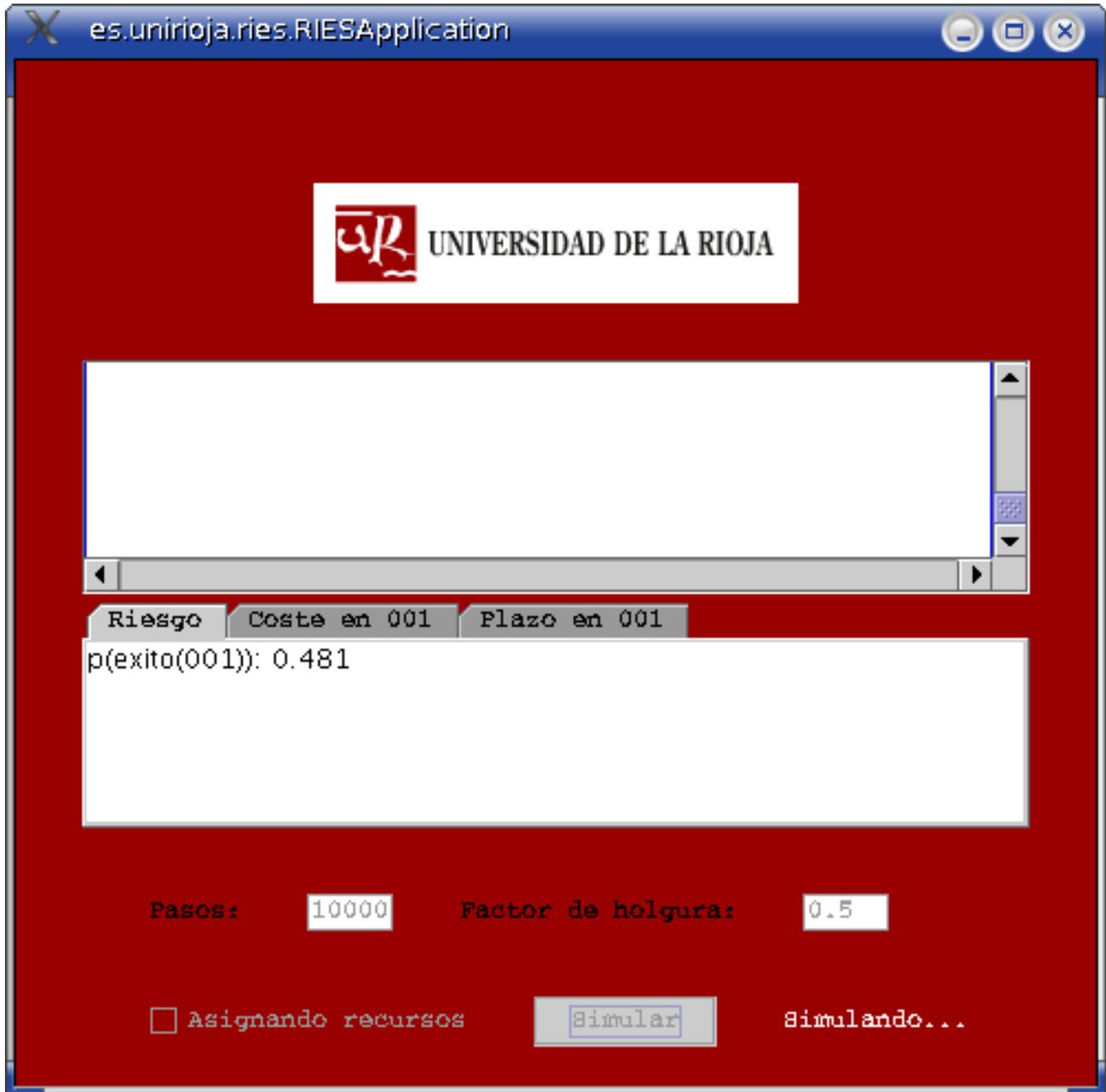
A continuación, un botón ofrece la acción de lanzar la simulación de los proyectos:



Captura 3: Aspecto del prototipo tras cargar un portafolio

Tras la simulación, el contenedor de fichas se llena de un número variable de éstas, dependiendo del número de proyectos en el portafolio y de si se ha elegido una simulación con asignación de recursos o no. Las fichas principales son:

- Riesgo: Indica la probabilidad de terminar cada proyecto con éxito:



Captura 4: Ficha Riesgo en RIES

- Coste: Muestra gráficamente las distribuciones de probabilidad para los costes asociados a los recursos de un proyecto:



Captura 5: Ficha Coste en RIES

- Plazo: Muestra las distribución de probabilidad de la variable τ de la duración del proyecto, distinguiendo las terminaciones en éxito y en fracaso:



Captura 6: Ficha Plazo en RIES

4.3 Formatos de los ficheros

4.3.1 Fichero de entrada para el portafolio

Cada portafolio se define en un fichero XML, en aras de la portabilidad. La aplicación XML que utilizamos es de creación propia, ya que la propuesta que más adeptos tenía en la gestión de proyectos, PMXML, resultaba insuficiente para los objetivos de esta investigación. Posteriormente, Pacific Edge Software ha dejado de soportar PMXML, y se han propuesto Mr. Project 0.6 DTD y Microsoft Project XML [MS Project XML 2003], pero además de tener una continuidad discutible, tampoco cubren nuestras necesidades.

A continuación incluimos el formato del fichero de entrada en DTD. En este fichero se definen las redes estocásticas de los proyectos incluido en el portafolio.

Las primeras 6 aserciones han sido heredadas de PMXML:

```
<!ELEMENT From (#PCDATA)>
<!ATTLIST From locationID CDATA #IMPLIED>
<!ATTLIST From locationType CDATA #IMPLIED>
<!ATTLIST From process CDATA #IMPLIED>
<!ATTLIST From path CDATA #IMPLIED>
<!ATTLIST From handle CDATA #IMPLIED>
```

A continuación se define la actividad:

```
<!ELEMENT Task (#PCDATA | TaskLabor | TaskActivation | SharedRisk)*>
```

La actividad puede incluir una relación de precedencia simple con evento de activación positivo:

```
<!ATTLIST Task ActivatedAfterSuccessful CDATA #IMPLIED>
```

o negativo:

```
<!ATTLIST Task ActivatedAfterFailing CDATA #IMPLIED>
```

Tras dos atributos normalizados se define la variable aleatoria de la duración de la actividad:

```
<!ATTLIST Task TaskID CDATA #IMPLIED>
```

```
<!ATTLIST Task Name CDATA #IMPLIED>  
<!ATTLIST Task DurationDistribution CDATA #IMPLIED>  
<!ATTLIST Task Duration CDATA #IMPLIED>  
<!ATTLIST Task DistributionParameter CDATA #IMPLIED>
```

Mediante esta aserción se añade un límite al número de repeticiones de la actividad:

```
<!ATTLIST Task MaxOccurs CDATA #IMPLIED>
```

Aquí definimos la influencia de las repeticiones sobre la duración:

```
<!ATTLIST Task RepeatingDurationFactor CDATA #IMPLIED>
```

Y seguimos con la actividad definiendo la probabilidad de éxito:

```
<!ATTLIST Task SuccessLaw CDATA #IMPLIED>  
<!ATTLIST Task SuccessProbability CDATA #IMPLIED>
```

Este atributo determina las actividades del conjunto F, aquellas cuya terminación exitosa conlleva el éxito del proyecto:

```
<!ATTLIST Task DeterminingSuccessfulProject CDATA #IMPLIED>
```

Este par de aserciones han sido heredadas de PMXML:

```
<!ELEMENT Owner (#PCDATA)>  
<!ATTLIST Owner ResourceID CDATA #IMPLIED>
```

A continuación definimos relaciones de precedencia:

```
<!ELEMENT ActivatingTask (#PCDATA)>  
<!ATTLIST ActivatingTask TaskID CDATA #IMPLIED>
```

con el signo del evento de activación:

```
<!ATTLIST ActivatingTask ActivatingEvent CDATA #IMPLIED>
```

y un atributo que avisa de los ciclos en la red del proyecto:

```
<!ATTLIST ActivatingTask CyclingPrelation CDATA #IMPLIED>
```

Estas seis aserciones ha sido heredadas de PMXML:

```
<!ELEMENT To (#PCDATA)>  
<!ATTLIST To locationID CDATA #IMPLIED>  
<!ATTLIST To locationType CDATA #IMPLIED>  
<!ATTLIST To process CDATA #IMPLIED>  
<!ATTLIST To path CDATA #IMPLIED>
```

```
<!ATTLIST To handle CDATA #IMPLIED>
```

Aquí definimos el conjunto RG de recursos genéricos del portafolio:

```
<!ELEMENT ResourcePool (#PCDATA | Resource)*>
```

En este elemento comienza a definirse el portafolio:

```
<!ELEMENT Body (#PCDATA | MultiProject)*>
```

La definición del proyecto:

```
<!ELEMENT Project (#PCDATA | Owners | Resources | RiskFactors | Tasks)*>  
<!ATTLIST Project ProjectID CDATA #IMPLIED>  
<!ATTLIST Project Name CDATA #IMPLIED>  
<!ATTLIST Project Start NMTOKEN #IMPLIED>  
<!ATTLIST Project Finish CDATA #IMPLIED>  
<!ATTLIST Project CreatedDate CDATA #IMPLIED>  
<!ATTLIST Project CalendarName CDATA #IMPLIED>  
<!ATTLIST Project SuccessProbability CDATA #IMPLIED>  
<!ATTLIST Project Title CDATA #IMPLIED>
```

y un recurso del proyecto o del portafolio:

```
<!ELEMENT Resource (#PCDATA)>  
<!ATTLIST Resource ParentResourceID CDATA #IMPLIED>  
<!ATTLIST Resource ResourceID CDATA #IMPLIED>  
<!ATTLIST Resource Name CDATA #IMPLIED>
```

cuya disponibilidad máxima puede estar acotada:

```
<!ATTLIST Resource MaxAvail CDATA #IMPLIED>
```

Los factores de riesgo:

```
<!ELEMENT RiskFactor (#PCDATA)>  
<!ATTLIST RiskFactor ParentRisk CDATA #IMPLIED>  
<!ATTLIST RiskFactor RiskID CDATA #IMPLIED>  
<!ATTLIST RiskFactor Name CDATA #IMPLIED>  
<!ATTLIST RiskFactor MaxAvail CDATA #IMPLIED>
```

El conjunto de actividades del proyecto:

```
<!ELEMENT Tasks (#PCDATA | Task)*>
```

La definición del portafolio:

```
<!ELEMENT MultiProject (#PCDATA | Project | ResourcePool |  
MultiProjectRiskFactors)*>  
<!ATTLIST MultiProject MultiProjectID CDATA #IMPLIED>
```

```
<!ATTLIST MultiProject Name CDATA #IMPLIED>
<!ATTLIST MultiProject Start CDATA #IMPLIED>
<!ATTLIST MultiProject Finish CDATA #IMPLIED>
<!ATTLIST MultiProject CreatedDate CDATA #IMPLIED>
<!ATTLIST MultiProject CalendarName CDATA #IMPLIED>
<!ATTLIST MultiProject Title CDATA #IMPLIED>
```

En estas aserciones se definen los recursos que necesita la actividad:

```
<!ELEMENT TaskLabor (#PCDATA)>
<!ATTLIST TaskLabor TaskLaborID CDATA #IMPLIED>
<!ATTLIST TaskLabor ResourceID CDATA #IMPLIED>
<!ATTLIST TaskLabor ScheduledWork CDATA #IMPLIED>
```

Nuevamente aserciones de PMXML:

```
<!ELEMENT Owners (#PCDATA | Owner)*>
<!ELEMENT BizTalk (#PCDATA | Route | Body)*>
<!ATTLIST BizTalk xmlns CDATA #IMPLIED>
<!ATTLIST BizTalk xmlns:xsi CDATA #IMPLIED>
<!ATTLIST BizTalk xsi:schemaLocation CDATA #IMPLIED>
<!ELEMENT Route (#PCDATA | From | To)*>
```

Este elemento engloba los eventos de activación de una actividad, y define su función de activación:

```
<!ELEMENT TaskActivation (#PCDATA | ActivatingTask)*>
<!ATTLIST TaskActivation ActivationCondition CDATA #IMPLIED>
<!ATTLIST TaskActivation ActivationTime CDATA #IMPLIED>
```

Conjunto R de recursos del proyecto:

```
<!ELEMENT Resources (#PCDATA | Resource)*>
```

Y por último, los factores de riesgo:

```
<!ELEMENT RiskFactors (#PCDATA | RiskFactor)*>
<!ELEMENT MultiProjectRiskFactors (#PCDATA | RiskFactor)*>
<!ELEMENT SharedRiskFactor (#PCDATA)>
<!ATTLIST SharedRiskFactor PctExplained CDATA #IMPLIED>
```

4.3.1.1 Ejemplo de fichero de entrada

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE BizTalk SYSTEM "project.dtd">

<BizTalk>
<Route>
  <From locationID="11111111" locationType="" process="" path="" handle="45" />
  <To locationID="22222222" locationType="" process="" path="" handle="45" />
```

```

</Route>
<Body>
<MultiProject
  MultiProjectID="000"
  Name="Multiproject Portfolio 000"
  Start="1999-11-19T08:00:00:00"
  Finish="1999-11-19T08:00:00:00"
  CreatedDate="1999-11-19T08:00:00:00"
  CalendarName="Standard"
  Title="MultiProject000">
<Project
  ProjectID="000"
  Name="000"
  Start="1999-11-19T08:00:00:00"
  Finish="1999-11-19T08:00:00:00"
  CreatedDate="1999-11-19T08:00:00:00"
  CalendarName="Standard"
  Title="Project000">
<Owners><Owner ResourceID="0" /></Owners>
<Resources>
<!-- Recurso limitado a 20 unidades -->
  <Resource
    ParentResourceID="001"
    ResourceID="000"
    Name="Ingeniero"
    MaxAvail="20"/>
<!-- Recurso limitado a 4 unidades -->
  <Resource
    ParentResourceID="002"
    ResourceID="001"
    Name="Contrata"
    MaxAvail="4"/>
</Resources>
<Tasks>
<!-- Actividad inicial -->
  <Task
    TaskID="0"
    Name="A: Problem Analysis"
    DurationDistribution="Normal"
    Duration="3"
    DistributionParameter="0.5f"
    SuccessLaw="Bernoulli"
    SuccessProbability="0.9f">
    <TaskLabor
      TaskLaborID="0"
      ResourceID="000"
      ScheduledWork="1"></TaskLabor>
  </Task>
<!-- Actividad B, sucesora de la actividad inicial -->
  <Task
    TaskID="1"
    Name="B: Process Design"
    ActivatedAfterSuccessful="0"
    DurationDistribution="Normal"
    Duration="18"
    DistributionParameter="0.8f"

```

```

        SuccessLaw="Bernoulli"
        SuccessProbability="0.75f">
    <TaskLabor
        TaskLaborID="0"
        ResourceID="000"
        ScheduledWork="1"></TaskLabor>
    </Task>
<!-- Actividad C, la cual se activa si la actividad B falla -->
<Task
    TaskID="2"
    Name="C: External buy of process"
    ActivatedAfterFailing="1"
    DurationDistribution="Uniform"
    Duration="3.5f"
    DistributionParameter="0.5f"
    SuccessLaw="Bernoulli"
    SuccessProbability="1">
<!-- Recurso para la actividad C, dos unidades de R1 -->
    <TaskLabor
        TaskLaborID="0"
        ResourceID="001"
        ScheduledWork="2"></TaskLabor>
</Task>
<Task
    TaskID="3"
    Name="D: Basic engineering"
    DurationDistribution="Normal"
    Duration="20"
    DistributionParameter="0.6f"
    SuccessLaw="Bernoulli"
    SuccessProbability="0.65f">
<!-- Activación de la actividad por juntura OR -->
    <TaskActivation
        ActivationCondition="0r"
        ActivationTime="After">
        <ActivatingTask
            TaskID="1"
            ActivatingEvent="Success"/>
        <ActivatingTask
            TaskID="2"
            ActivatingEvent="Sucess"/>
    </TaskActivation>
    <TaskLabor
        TaskLaborID="0"
        ResourceID="000"
        ScheduledWork="1"></TaskLabor>
</Task>
<Task
    TaskID="4"
    Name="E: Detail engineering"
    ActivatedAfterSuccessful="3"
    DurationDistribution="Uniform"
    Duration="9"
    DistributionParameter="1"
    SuccessLaw="Bernoulli"
    SuccessProbability="1"
    DeterminingSuccessfulProject="Yes">

```

```

<!-- Actividad que define el final exitoso del proyecto -->
  <TaskLabor
    TaskLaborID="0"
    ResourceID="001"
    ScheduledWork="2"></TaskLabor>
  </Task>
</Tasks>

</Project>

<ResourcePool>
  <Resource
    ResourceID="001"
    Name="Ingeniero"
    MaxAvail="100"
  />
  <Resource
    ResourceID="002"
    Name="Contrata"
    MaxAvail="100"
  />
  <Resource
    ResourceID="003"
    Name="Tecnico"
    MaxAvail="100"
  />
</ResourcePool>
</MultiProject>
</Body>
</BizTalk>

```

4.3.2 Fichero de configuración

Tras esta aplicación XML, describimos el fichero XML de configuración del prototipo RIES, mediante el cual se manipulan algunos aspectos de la simulación. El fichero se llama RIES.xml.

Prefacio XML:

```

<?xml version="1.0" standalone="yes"?>
<!DOCTYPE Revision>

```

Etiqueta raíz:

```

<Configuration Name="RIES">

```

Solicita que se valide la aplicación XML del portafolio contra su DTD

```

  <XMLValidation/>

```

Indica en qué fichero están los datos de entrada:

```
<MultiProject FileName="project20.xml"/>
```

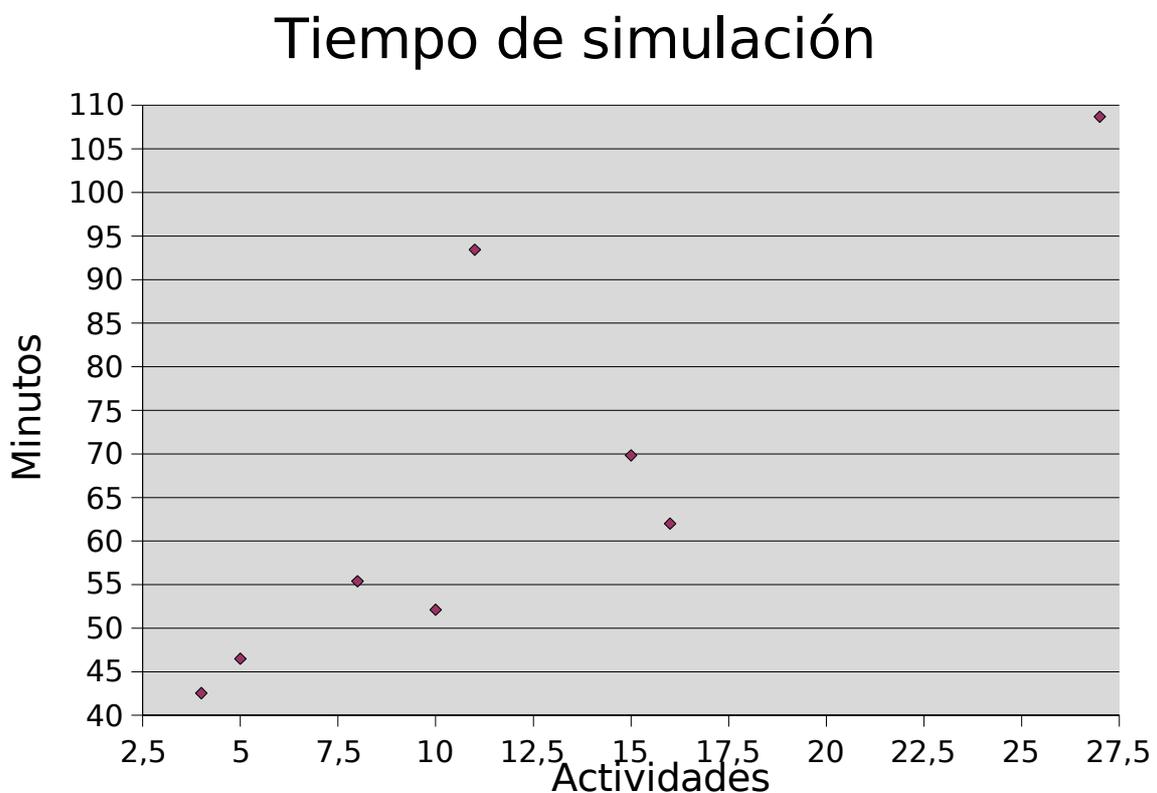
Indica el nivel de locuacidad de la información que se envía a la salida estándar y al área de texto.

```
<Verbose Level="0"/>
</Configuration>
```

4.4 Rendimiento

Durante el diseño e implementación del prototipo RIES no fueron tomados en cuenta criterios de rendimiento o escalabilidad. El propósito del prototipo es demostrar que el modelo descrito en el capítulo 3, “Modelo de datos”, puede ser implementado. No obstante, describimos en la figura 17 los tiempos de computación de las simulaciones en una máquina PC con procesador AMD Athlon de 64 bits a 2.01GHz y con 1GB de RAM. El sistema operativo es Windows XP Profesional SP2.

Figura 17: Tiempos de simulación para $Q=10.000$



En el capítulo 6, “Conclusiones y líneas de futuro”, describiremos alguna vía para mejorar el rendimiento del prototipo.

5. Experimentación

5.1 Experimento número 11

Este es un problema propuesto en [Elmaghraby 1977], página 275, que hemos elegido porque lo han resuelto otros investigadores [Avramidis; Wilson 1993b]. A continuación, se expresa el grafo del proyecto según la representación original, actividades en los arcos.

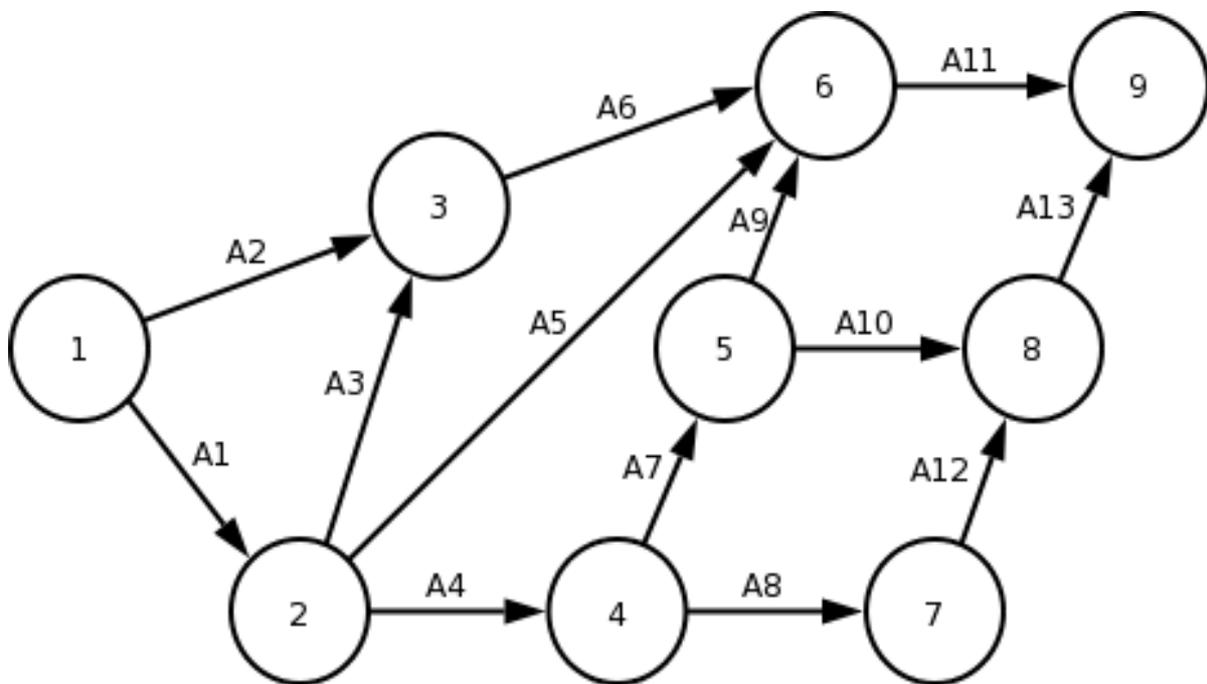


Figura 18: Proyecto del experimento 11 formato AeA

El proyecto contiene dos tareas iniciales, A_1 y A_2 , y dos tareas finales, A_{11} y A_{13} . Este es el grafo resultante en la representación de las actividades mediante redes de Petri estocásticas más generales:

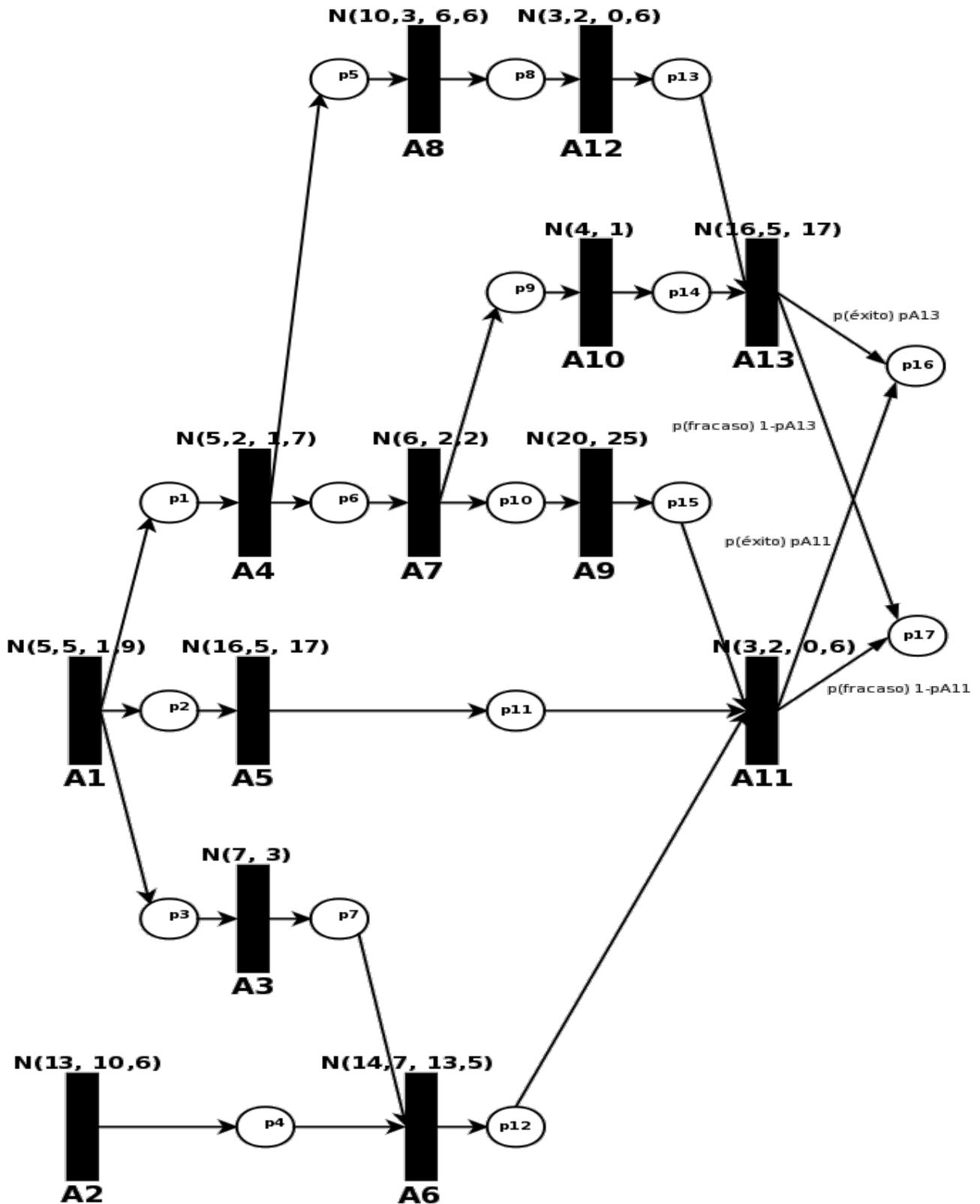


Figura 19: Proyecto 11 en red de Petri

A continuación, se detallan las distribuciones de probabilidad que utilizamos para la experimentación. Primero, partimos de las duraciones medias dadas por el original, y asignamos una desviación estándar de la cuarta parte de aquélla.

Actividad	Duración media	Desviación estándar	Distribución
A ₁	5,5	1,375	N(5,5, 1,891)
A ₂	13	3,25	N(13, 10,563)
A ₃	7	1,75	N(7, 3,063)
A ₄	5,2	1,3	N(5,2, 1,69)
A ₅	16,5	4,125	N(16,5, 17,016)
A ₆	14,7	3,675	N(14,7, 13,506)
A ₇	6	1,5	N(6, 2,25)
A ₈	10,3	2,575	N(10,3, 6,631)
A ₉	20	5	N(20, 25)
A ₁₀	4	1	N(4, 1)
A ₁₁	3,2	0,8	N(3,2, 0,64)
A ₁₂	3,2	0,8	N(3,2, 0,64)
A ₁₃	16,5	4,125	N(16,5, 17,016)

Tabla 3: Actividades del proyecto del experimento 11

Los resultados obtenidos con una simulación de 10.000 pasos se pueden resumir así en cuanto a la duración total del proyecto δ :

$$\bar{\delta}=43,34$$

$$\text{Var}(\bar{\delta})=19,87$$

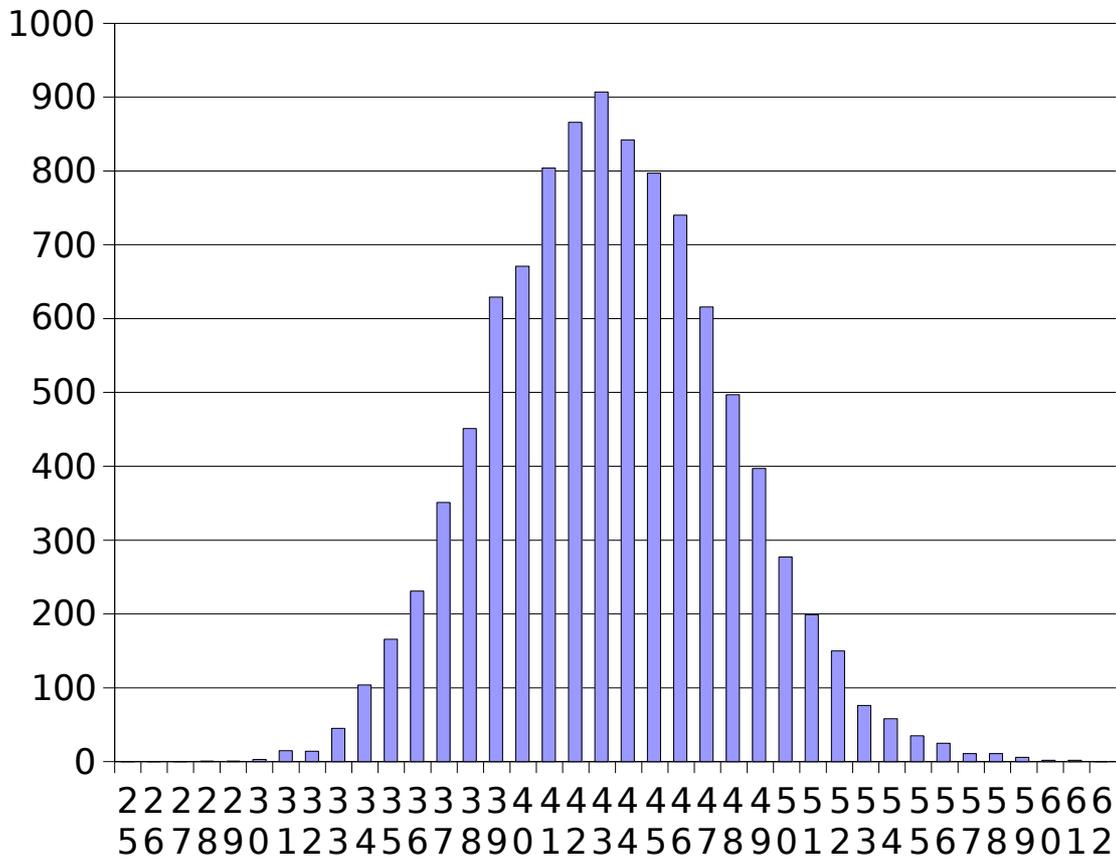
$$\max(\bar{\delta})=61,47$$

$$\min(\bar{\delta})=27,86$$

A continuación, se incluye el diagrama de barras de la frecuencia muestral de la variable δ :

Figura 20: Distribución simulada de la duración del proyecto 11

Duración del proyecto



[Avramidis; Wilson 1993b] utilizan un estimador integrado de esperanza condicional y muestreo de hipercubo latino para aproximarse al valor $\bar{\delta}$ con un número bajo de iteraciones. Nuestro punto de vista es distinto; no nos interesa sólo $\bar{\delta}$ sino que pretendemos ofrecer la distribución completa para una mejor apreciación de la incertidumbre sobre la duración del proyecto.

Otro estudio interesante, según defienden [Wendling; Lorance 1999], es el de los posibles caminos críticos. Especialmente nos fijaremos en el dominio relativo de unos posibles caminos críticos sobre otros. En este caso, vemos que el camino crítico dominante es $A_0-A_1-A_4-A_8-A_{12}-A_{13}-A_{14}$, pero bastante cerca se sitúa el camino $A_0-A_1-A_4-A_7-A_9-A_{11}-A_{14}$.

Camino crítico	Probabilidad
$A_0-A_1-A_4-A_8-A_{12}-A_{13}-A_{14}$	0,48
$A_0-A_1-A_4-A_7-A_9-A_{11}-A_{14}$	0,43
$A_0-A_1-A_4-A_7-A_{10}-A_{13}-A_{14}$	0,05
$A_0-A_2-A_6-A_{11}-A_{14}$	0,03
$A_0-A_1-A_3-A_6-A_{11}-A_{14}$	0,01

Tabla 4: Dominio relativo de los caminos críticos del proyecto 11

Aparte de los cinco reflejados en la tabla superior, también aparecen algunos otros caminos críticos, pero sus probabilidades son despreciables.

5.2 Experimento número 12

Este es un problema propuesto en [Bell; Han 1991], página 320, que a su vez es una versión simplificada de un problema publicado por [Stinson et al. 1978], página 258. Este problema es interesante por incluir restricciones simultáneas en varios recursos. A continuación, se expresa el grafo del proyecto según la representación de red de Petri. Los lugares de fusión correspondientes a los recursos del proyecto sólo se muestran, por aligerar el gráfico, para la actividad A_4 :

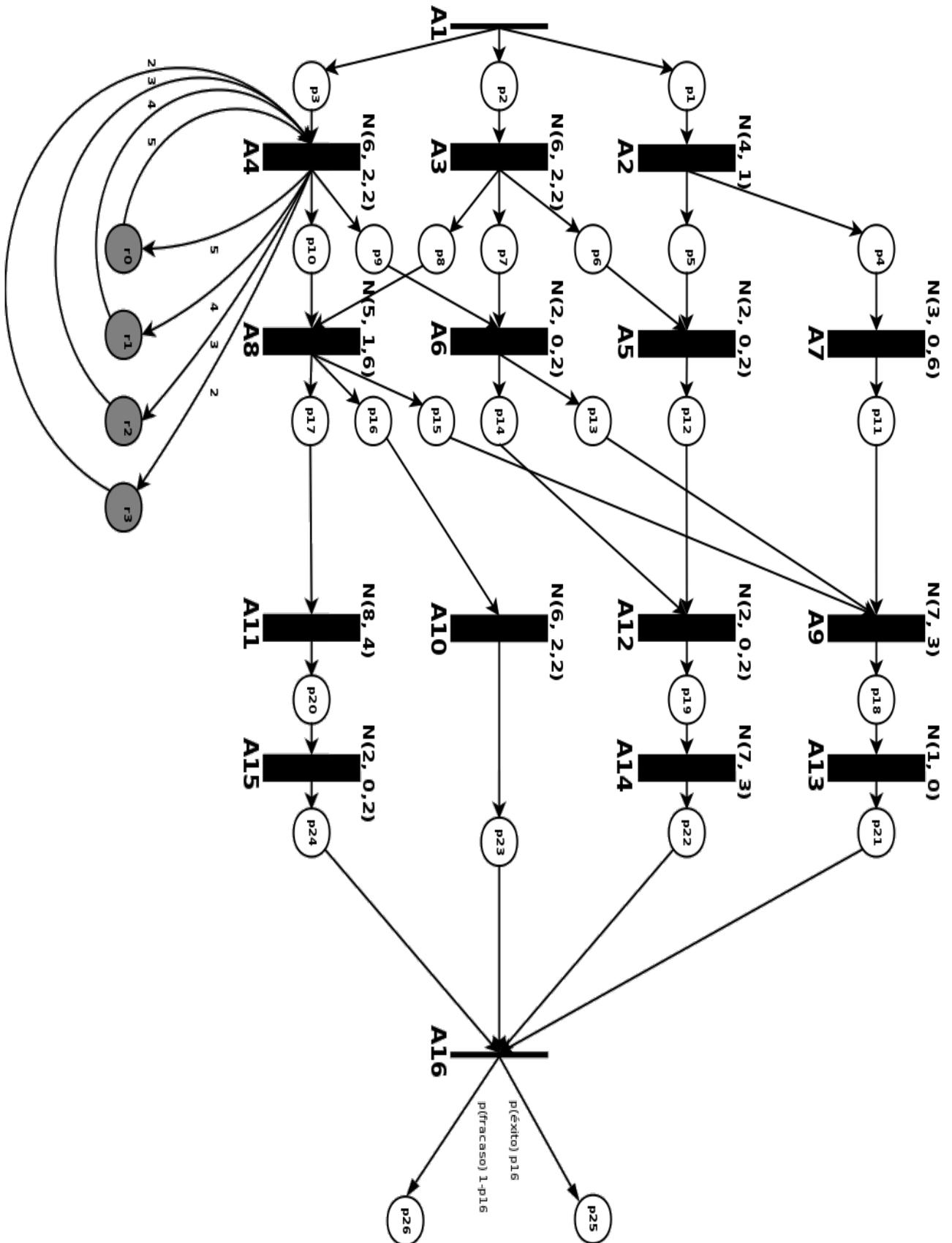


Figura 21: Proyecto 12 en red de Petri

Sistema de planificación estocástico de proyectos

En la tabla siguiente, se detallan las distribuciones de probabilidad que utilizamos para la experimentación. Primero, partimos de las duraciones medias dadas por el original, y asignamos una desviación estándar de la cuarta parte de aquélla. La últimas cuatro columnas detallan los requerimientos de cada actividad sobre los cuatro recursos de los que dispone el proyecto. La dificultad de este experimento reside en que los cuatro recursos se encuentran limitados a 10 unidades.

Actividad	Duración media	Desviación estándar	Distribución	R0	R1	R2	R3
A ₁	0	0	-	0	0	0	0
A ₂	4	1	N(4, 1)	3	2	4	2
A ₃	6	1,5	N(6, 2,25)	6	2	3	1
A ₄	6	1,5	N(6, 2,25)	5	4	3	2
A ₅	2	0,5	N(2, 0,25)	1	3	3	1
A ₆	2	0,5	N(2, 0,25)	4	1	6	2
A ₇	3	0,75	N(3, 0,563)	4	1	4	5
A ₈	5	1,25	N(5, 1,563)	1	2	2	2
A ₉	7	1,75	N(7, 3,063)	1	5	2	3
A ₁₀	6	1,5	N(6, 2,25)	6	1	5	3
A ₁₁	8	2	N(8, 4)	6	1	6	3
A ₁₂	2	0,5	N(2, 0,25)	5	1	6	6
A ₁₃	1	0,25	N(1, 0,063)	3	4	4	3
A ₁₄	7	1,75	N(7, 3,063)	1	1	1	1
A ₁₅	2	0,5	N(2, 0,25)	1	1	4	1
A ₁₆	0	0	-	0	0	0	0

Tabla 5: Actividades del proyecto del experimento 12

Los resultados obtenidos con una simulación de 10.000 pasos se pueden compendiar así en cuanto a la duración total del proyecto:

$$\bar{\delta}=22,31$$

$$\text{Var}(\bar{\delta})=5,87$$

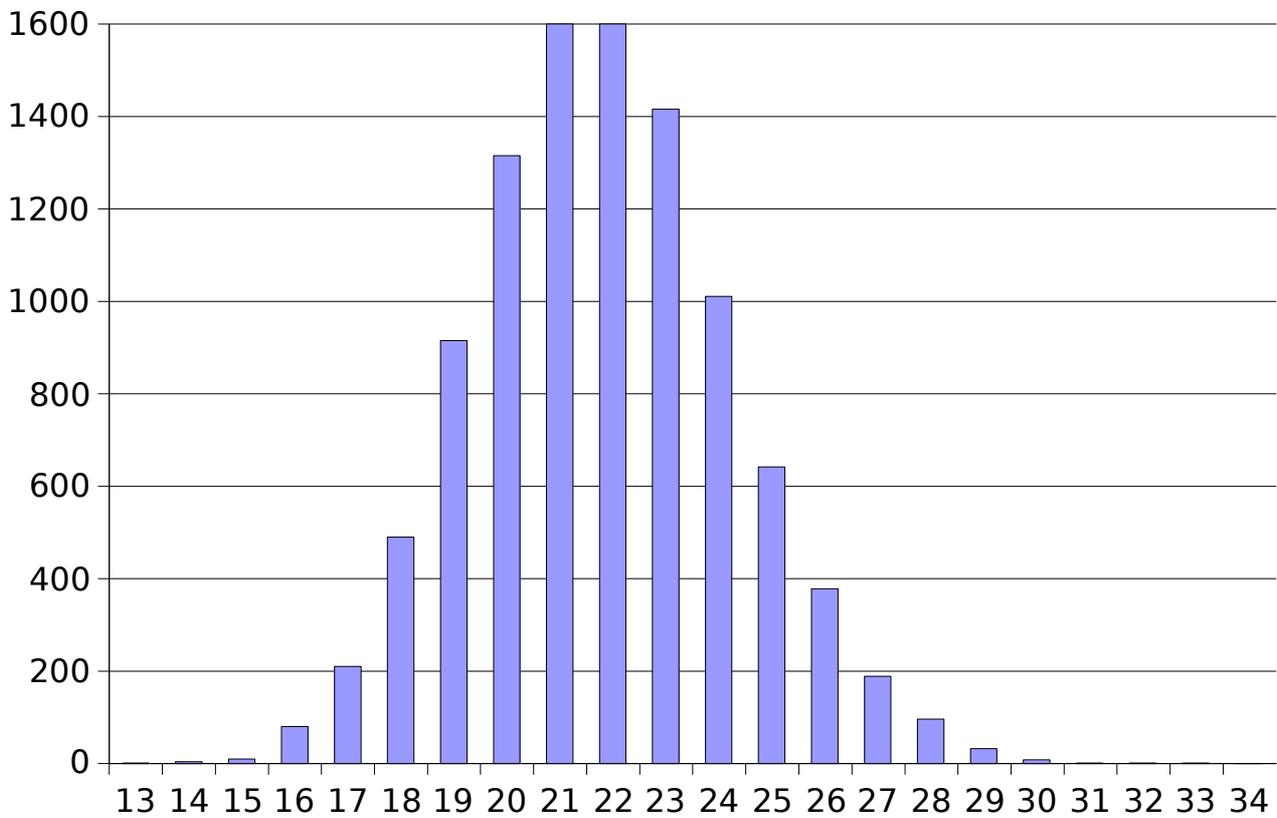
$$\max(\bar{\delta})=33,25$$

$$\min(\bar{\delta})=13,65$$

A continuación, se incluye el diagrama de barras de la frecuencia muestral de la variable duración del proyecto sin tener en cuenta las restricciones en los recursos:

Figura 22: Distribución simulada de la duración del proyecto 12

Distribución de la duración



No obstante, teniendo en cuenta las limitaciones de los recursos y aplicando el algoritmo de ramificación y poda para buscar la solución óptima en cada paso de la simulación, la duración del proyecto se puede describir mediante:

$$\bar{\delta}'=38,47$$

$$\text{Var}(\bar{\delta}')=26,58$$

$$\max(\bar{\delta}')=69,59$$

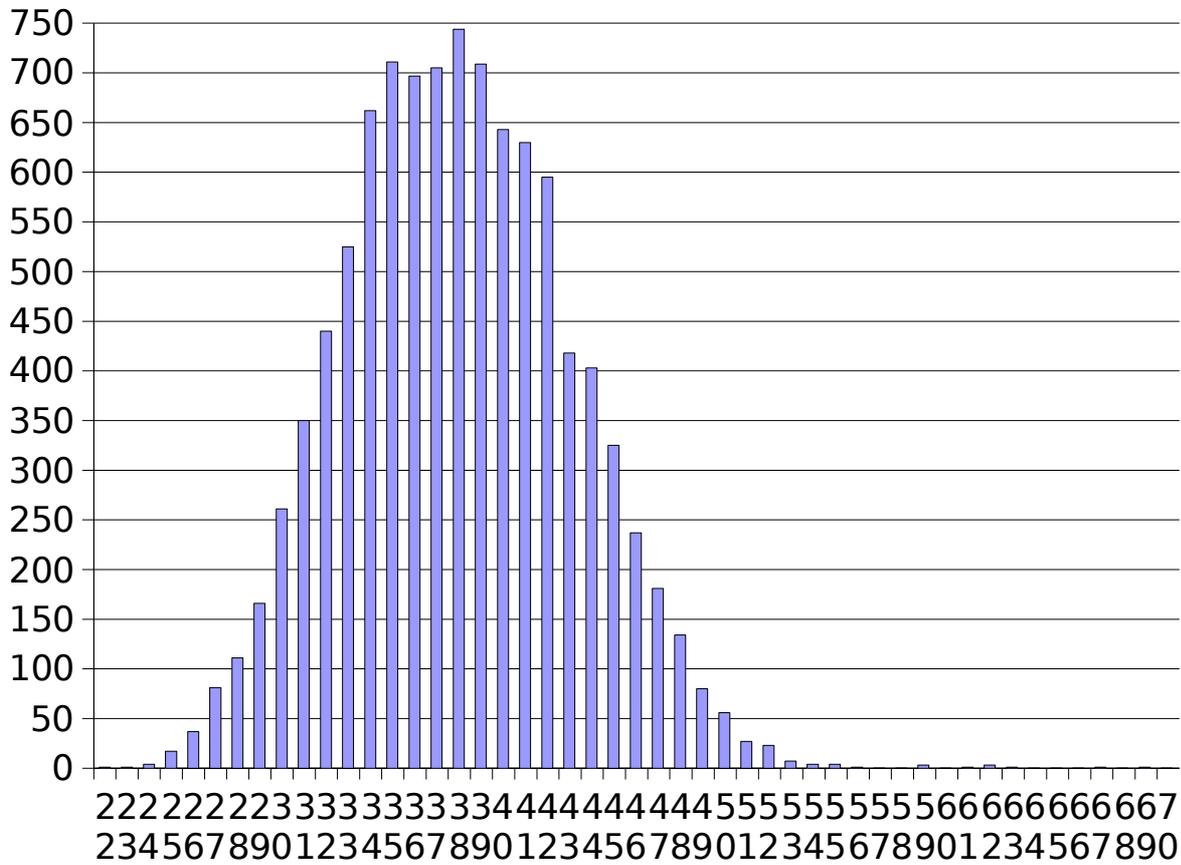
Sistema de planificación estocástico de proyectos
 $\min(\bar{\delta}')=22,56$

[Bell; Han 1991] propusieron un heurístico para este problema de planificación multirecurso, con duraciones discretas de las actividades. Su heurístico encontraba solución óptima $\delta'=31$ en un tiempo de computación “centenares de veces” menor que el procedimiento exacto, aunque hay que tener en cuenta que entonces no se había publicado el procedimiento DH de [Herroelen; Demeulemeester 1992] que usamos aquí. En cualquier caso, nuestro interés no está en reducir los tiempos de computación, sino en ofrecer una visión completa del riesgo inherente al proyecto. En los datos presentados por nuestro algoritmo podemos observar el efecto de la variabilidad en las duraciones de las actividades sobre el cumplimiento de los programas óptimos. Este efecto, presente cuando aparecen varios caminos cuasi críticos, es bien conocido en la comunidad de dirección de proyectos, pero rara vez es medido y presentado con exactitud. Más aún, podemos presentarlo gráficamente, mediante un diagrama de barras de la frecuencia muestral.

Obsérvese también que la probabilidad de cumplir el plazo del proyecto, si se utilizara la estimación de la media $\bar{\delta}$, es tan pequeña que ni se distingue en una simulación de Monte Carlo de 10.000 pasos.

Figura 23: Distribución simulada y restringida de la duración del proyecto 12

Duración restringida



5.3 Experimento número 13

Este es un problema que contiene dos ciclos de diferente longitud, propuesto por [Pollack-Johnson; Liberatore 2005], página 18. Por facilitar la programación del paquete de entrada es.unirioja.ries.entrada, hemos utilizado durante la carga de datos la representación en grafo dirigido acíclico que se obtiene con el algoritmo 3.3.3.1 “Algoritmo de despliegue de las iteraciones en los ciclos”. Aquí presentamos el grafo del proyecto mediante redes de Petri:

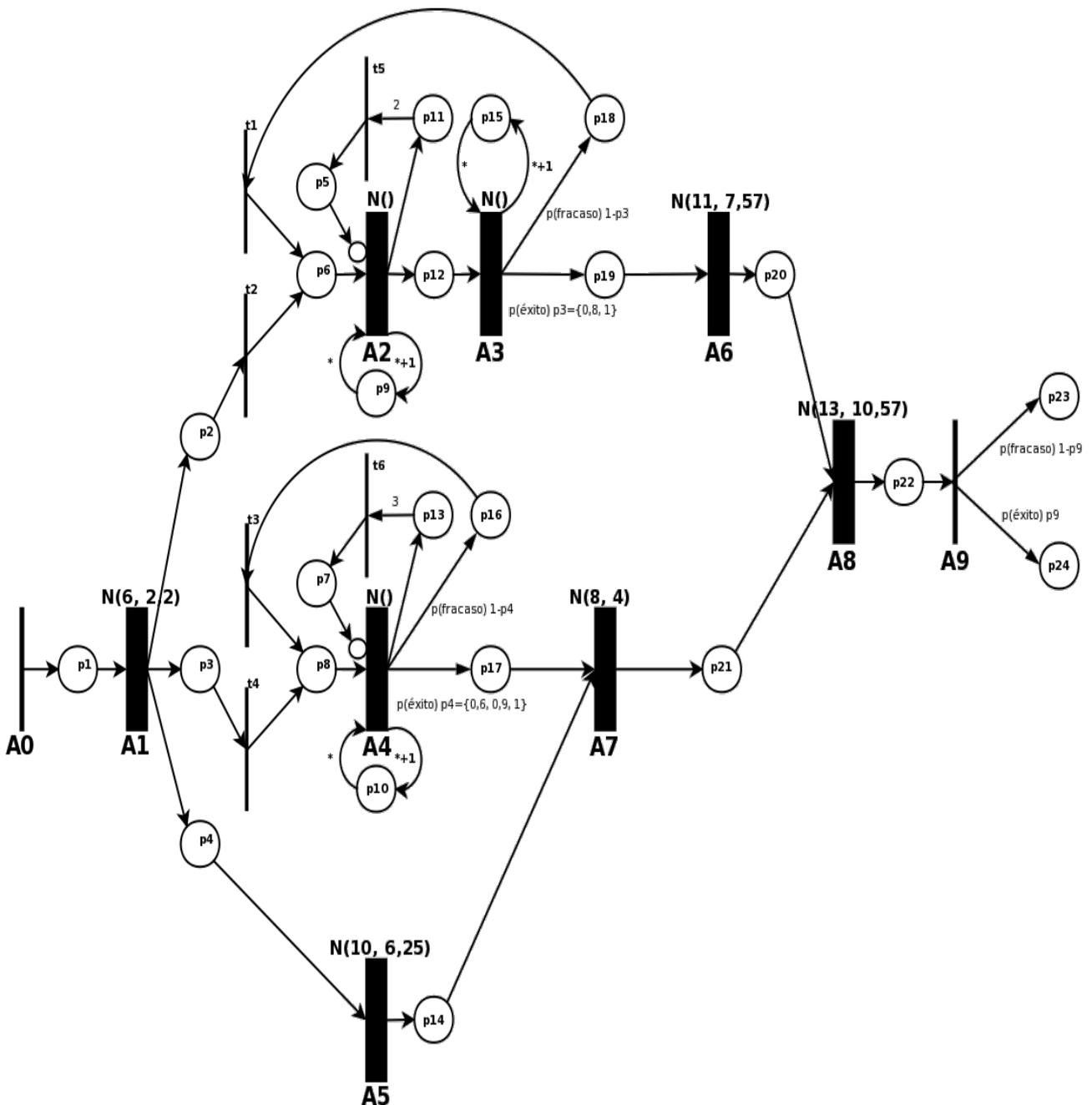


Figura 24: Proyecto 13 en red de Petri

En la tabla siguiente, se detallan las distribuciones de probabilidad que utilizamos para la experimentación, detalladas para cada instancia de cada actividad. Primero, partimos de las duraciones medias dadas por el original, y asignamos una desviación estándar de la cuarta parte de aquella. La última columna muestra las probabilidades de éxito de cada repetición de cada actividad.

Actividad	Duración media	Desviación estándar	Distribución	Probabilidad de éxito
A ₀₁	0	0	-	1
A ₁₁	6	1,5	N(6, 2,25)	1
A ₂₁	3	0,75	N(3, 0,563)	1
A ₂₂	1	0,25	N(1, 0,063)	1
A ₃₁	4	1	N(4, 1)	0,8
A ₃₂	2	0,5	N(2, 0,25)	1
A ₄₁	5	1,25	N(5, 1,563)	0,6
A ₄₂	7	1,75	N(7, 3,063)	0,9
A ₄₃	9	2,25	N(9, 5,063)	1
A ₅₁	10	2,5	N(10, 6,25)	1
A ₆₁	11	2,75	N(11, 7,563)	1
A ₇₁	8	2	N(8, 4)	1
A ₈₁	13	3,25	N(13, 10,563)	1
A ₉₁	0	0	-	1

Tabla 6: Instancias de actividades del proyecto 13

Los resultados obtenidos con una simulación de 10.000 pasos se pueden resumir así en cuanto a la duración total del proyecto $\bar{\delta}$:

$$\bar{\delta} = 39,29$$

$$\text{Var}(\bar{\delta}) = 10,8$$

$$\max(\bar{\delta}) = 55,9$$

$$\min(\bar{\delta}) = 29,84$$

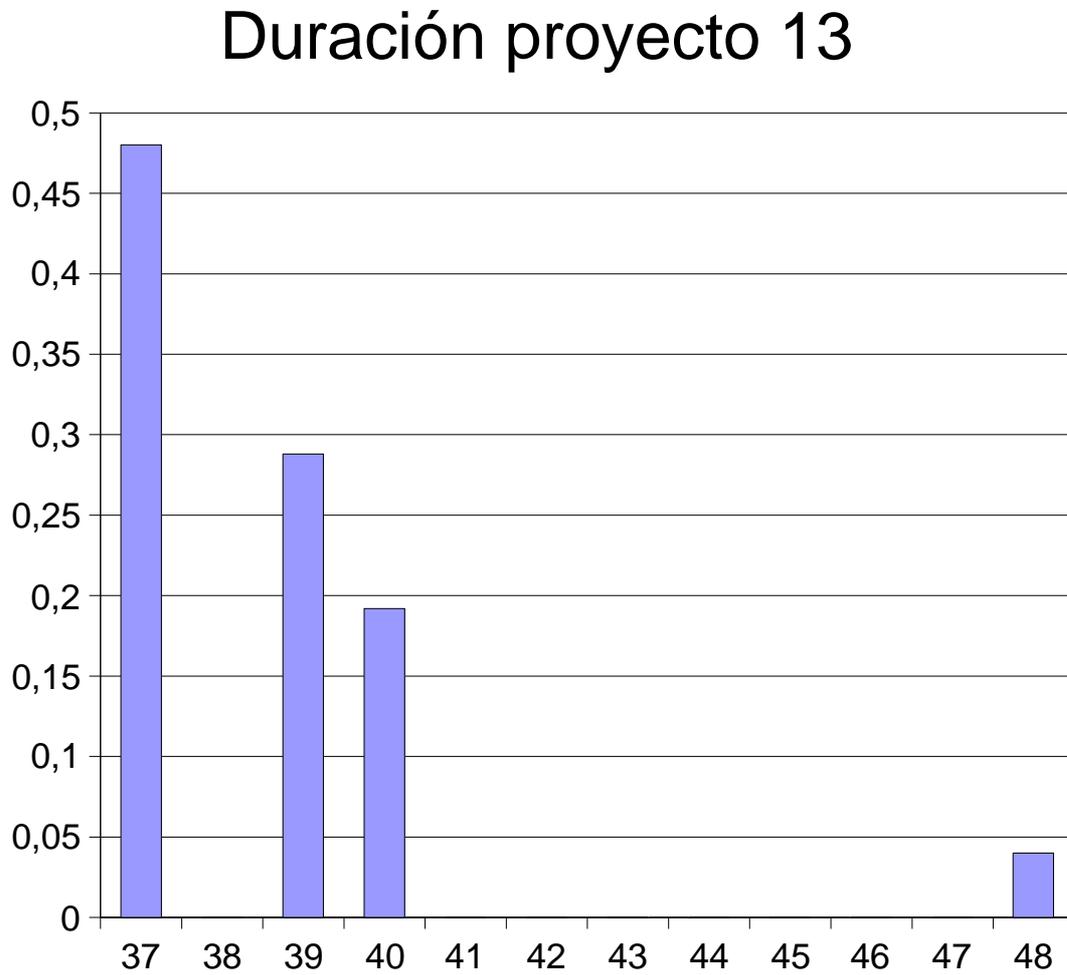
Respecto del modelo de escenarios propuesto por [Pollack-Johnson; Liberatore 2005], la diferencia empieza a despuntar en la esperanza de la duración del proyecto, que según su cálculo de escenarios es:

$$E[\delta] = 38,59$$

La diferencia se explica por la presencia de una mayor incertidumbre en nuestro modelo, porque contemplamos simultáneamente los ciclos y la duración probabilística de las actividades. En [Pollack-Johnson; Liberatore 2005] se presenta también un ejemplo de duración probabilística con distribución discreta. Ambos aspectos se pueden integrar, aunque manejarlos simultáneamente es trabajoso y poco escalable. En su modelo, a diferencia del nuestro, no se contemplan distribuciones continuas para la duración de las actividades.

Por expresar gráficamente la diferencia, esto sería lo que hubieran presentado Bruce Pollack-Johnson y Matthew Liberatore como distribución de probabilidad de la duración del proyecto, tras su análisis de 6 escenarios sobre los ciclos del proyecto. Compárese con nuestro resultado en la figura 25 :

Figura 26: Distribución de la duración del proyecto 13 según el análisis de [Pollack-Johnson; Liberatore 2005]



6. Conclusiones y líneas de futuro

6.1 Conclusiones

La principal contribución de esta tesis es el planteamiento de un modelo del riesgo en el proyecto que permite capturar 1) la incertidumbre sobre la duración de las actividades individuales, 2) la incertidumbre que se plantea entre las diversas alternativas de ejecución del proyecto, incluidos los planes de contingencia y los ciclos de repetición, y 3) la incertidumbre sobre el propio resultado del proyecto como conjunto y de cada una de sus actividades.

El modelo que proponemos es un modelo en red estocástico, expresado mediante las redes de Petri estocásticas más generales. La actividad se descompone en tres zonas:

Activación: Es el punto en el que se decide si la actividad se pone en marcha. Esta decisión depende de los resultados de las actividades precedentes, del tiempo y del número de repetición.

Desarrollo: Es la zona, de duración variable, donde la actividad va empleando sus recursos para la consecución de sus fines.

Terminación: Es el momento donde se decide si la actividad ha cumplido sus objetivos y ha sido un éxito, o si, por el contrario, hay que considerarla un fracaso.

Las actividades incluidas en la red del proyecto contemplan dos estados de terminación posibles, el estado de éxito y el estado de fracaso. Ésta es una primera y fundamental modelización del riesgo en el proyecto, aunque no sea un aspecto que se suela considerar en la literatura. Para el estado de terminación definimos una variable aleatoria booleana regida por una ley de Bernoulli. En los casos de repetición de actividades, además permitimos modelar escalarmente esta probabilidad de éxito respecto del número de instancia de actividad.

El riesgo de fracaso en el proyecto se define mediante el conjunto de variables aleatorias ligadas al estado de ciertas actividades consideradas finales. Este riesgo es independiente de consideraciones temporales, cuya incertidumbre tratamos aparte. Se refiere a la probabilidad de obtener el producto correcto.

Ante la posibilidad de no obtener el producto correcto en una actividad, se abren vías alternativas de ejecución. Las vías alternativas de ejecución contemplan tanto los planes de contingencia, entendidos como secuencias de actividades o subproyectos conducentes a conseguir una alternativa satisfactoria, como los ciclos de repetición, donde son las mismas actividades las que se ejecutan una o varias veces más para conseguir obtener el mismo producto correctamente.

Aparte de las mencionadas vías alternativas, la incertidumbre sobre la temporalidad del proyecto y sus actividades se caracteriza en el modelo mediante varias aportaciones. Por un lado, se permite definir bucles de repetición sobre cadenas de actividades, lo que expresa el caso real que se da cuando los resultados no son aceptables en términos de calidad, o cuando las especificaciones del producto cambian. Por otro lado, permitimos expresar la duración de la actividad mediante una variable aleatoria de distribución paramétrica. Finalmente, la correlación entre las duraciones de las actividades afectadas por un mismo factor de riesgo se puede expresar mediante una variable latente.

La búsqueda de soluciones óptimas en el modelo se basa en una simulación de Monte Carlo, a partir de la cuál se implementan algoritmos de secuenciación y asignación de recursos cuyos resultados agregados muestran una información completa sobre las expectativas de comportamiento de la red del proyecto.

La segunda contribución de esta tesis consiste en la demostración de la viabilidad del modelo anteriormente expuesto mediante la implementación de un prototipo de software de planificación de proyectos. El prototipo toma como entrada un proyecto definido según el modelo de riesgo propuesto, y ejecuta diversos algoritmos de planificación para ofrecer al profesional de la dirección de proyectos la información que necesita sobre criticidad, probabilidad de éxito, consumo de recursos, coste, duración, etc... En lugar de la estimación de un solo punto, el profesional se encuentra con una descripción completa de la variabilidad asociada al proyecto, descripción que le permite una mejor comprensión del riesgo presente en cada aspecto.

El prototipo cuenta con algunas funcionalidades añadidas que aportan facilidad de uso y una mejor visión de los proyectos y recursos gestionados. El prototipo permite definir un multiproyecto como un portafolio de proyectos que se gestionan de forma simultánea. Estos proyectos pueden compartir ciertos recursos mediante una relación jerárquica entre los recursos genéricos del portafolio y los recursos de cada proyecto.

Los recursos asignados a una actividad, a su vez, pueden ser de distinto tipo, definiéndose para cada tipo un punto de entrada y de salida, o relaciones de precedencia entre los recursos, de modo que su consumo no sea constante durante el desarrollo de la actividad.

El modelo de riesgo y el prototipo se han probado en un conjunto de proyectos de pequeño tamaño extraídos de la literatura y de la práctica profesional. Los experimentos han demostrado que la información obtenible para la gestión del riesgo en el proyecto es más extensa y precisa que la que se obtiene con las técnicas usuales de la industria.

Con esta tesis esperamos abrir la vía a la adopción por parte de la comunidad profesional de la dirección de proyectos de técnicas avanzadas de gestión del riesgo en el proyecto. Pensamos que la disponibilidad de un software similar al prototipo, de software de planificación avanzado que contemple el tipo de técnicas y el modelo de gestión de riesgo que hemos desarrollado, contribuirá

a una mejor gestión de los proyectos en general. Especialmente creemos contribuir a una mayor penetración de la disciplina de ingeniería de proyectos en los campos en los que la presencia de riesgo alto ha supuesto un obstáculo hasta ahora, como la investigación y desarrollo o la ingeniería del software.

6.2 Líneas de futuro

Los resultados de esta investigación abren un gran número de líneas de investigación para mejorar el modelo o los algoritmos descritos, para aplicar el modelo en otros problemas de planificación de proyectos y para facilitar la aplicación de las técnicas desarrolladas en la práctica profesional.

Con respecto de la mejora del modelo de riesgo presentado, la dependencia de factores de riesgo ofrece un abanico amplio de posibilidades. Otras caracterizaciones, quizá no basadas en la variable latente, son posibles. Es interesante estudiar su aplicación a la probabilidad de éxito de las actividades, o, fuera de la repetición de actividades, a actividades ligadas por características comunes, a los recursos y sobre todo a los proyectos de un mismo portafolio. Para esta aplicación se pueden estudiar las lecciones obtenidas en la gestión de portafolios de inversión.

En la mejora del modelo, se pueden explorar otro tipo de redes de Petri como expresión gráfica y dinámica. Hemos visto que otros investigadores han optado por otras redes de Petri. Del estudio de las redes de Petri pueden surgir nuevos formalismos más completos que se adapten mejor al problema que tratamos de describir. Los estudios de alcanzabilidad cronológica del grafo son prometedores en las redes de Petri estocásticas.

En la mejora de algoritmos, un aspecto para desarrollar es la asignación de recursos con recursos complejos. La definición de tiempos de comienzo y fin del uso de los recursos en la actividad requiere una modificación del procedimiento exacto de asignación de recursos, que quizá pueda dar pie a alguna optimización por reducción del espacio de búsqueda.

En la línea de la optimización, otro aspecto para investigar es la mejora del rendimiento del prototipo. En el apartado 1.3.2.1 “Técnicas de reducción de la varianza” hemos presentado varias técnicas que permiten mejorar el rendimiento de la simulación de Monte Carlo, al precio de ocultar parte de la variabilidad, y por tanto, del riesgo del proyecto. Aunque su aplicación directa en la simulación no es recomendable para los objetivos de esta tesis, la búsqueda de un compromiso entre fidelidad y rendimiento puede ser interesante para una aplicación profesional.

Por otro lado, pero en cierta manera relacionado con el rendimiento, está la cuestión de la anticipación. El algoritmo de secuenciación de actividades y el algoritmo de asignación de recursos trabajan suponiendo que la incertidumbre sobre el resultado y la duración de cada actividad ha sido resuelta, y toman decisiones en consecuencia. Hay dos cautelas que guardar: 1) no anticiparse,

evitar suponer que se tomarán decisiones no implementables, decisiones basadas en información que el equipo del proyecto no tendrá en el momento de tomarlas y 2) no simplificar en exceso, no ignorar la información de que se dispone pese a su carácter incierto. Nosotros hemos optado por planificar una sola vez, y sin modelar la actitud hacia el riesgo de director de proyecto. Suponemos que el proyecto se replanifica periódicamente desde el momento actual en adelante. Una línea de investigación abierta está en definir los puntos del apartado 2.11.1 “Los puntos de control” y anticiparse a las decisiones que se puedan tomar en el futuro. Aquí pueden aplicarse las técnicas de reducción de la varianza.

En la práctica profesional, es necesario estudiar la interpretación del riesgo en un entorno empresarial. Cómo se percibe la incertidumbre, qué influencias pueden alterar esta percepción o qué procesos son recomendables en cada fase del proyecto para gestionar el riesgo son cuestiones abiertas que afectan a la aplicación del tipo de técnicas presentadas en la tesis. Los aspectos visuales en la presentación de las previsiones sobre el riesgo o aspectos psicológicos también deberían ser tenidos en cuenta.

Una línea extraordinariamente interesante se articula en torno al análisis de sensibilidad. Es un aspecto cuya importancia se ha resaltado reiteradamente desde el mundo profesional, pero que rara vez se ha abordado en la comunidad académica. Preguntas como cuáles son los límites de cambio de un parámetro (duración de una actividad, probabilidad de éxito, etc) de modo que una solución siga siendo la óptima, o dado un cambio en un parámetro, cuál sería el objetivo óptimo alcanzable y la nueva solución óptima. Algunas de estas preguntas son abordables desde el área de investigación que hemos visto en el apartado 2.11.2 “La reparación del programa”: ¿En qué condiciones permanece óptimo un programa del proyecto? ¿Cómo se evalúa la robustez de una solución óptima o no? La respuesta a otras parece más lejana: ¿En qué condiciones permanece óptimos los valores de la función objetivo? ¿Qué análisis de sensibilidad se puede realizar a partir de una solución parcialmente especificada?

7. Referencias

- Abeyasinghe et al. 2001. An efficient method for scheduling construction projects with resource constraints / M. Chelaka I. Abeyasinghe; David J. Greenwood; D. Eric Johansen. *International Journal of Project Management* 19, no. 1 (2001): 29-45
- Adeli; Park 1998. Method and apparatus for efficient design automation and optimization, and structure produced thereby / Hojjat Adeli; Hyo Seon Park. United States Patent, 5,815,394, 1998
- Adelson-Velsky; Levner 2002. Project Scheduling in AND-OR Graphs: A generalization of Dijkstra's algorithm / George M. Adelson-Velsky; Eugene Levner. *Mathematics of Operations Research* 27, no. 3 (2002): 504-517
- Aho et al. 1974. *The Design and Analysis of Computer Algorithms* / Alfred V. Aho; John E. Hopcroft; Jeffrey D. Ullman. Addison-Wesley, Reading, Massachusetts, 1974
- Ahuja; Walsh 1983. *Successfull Methods in Cost Engineering* / Hira N. Ahuja; Michael A. Walsh. Wiley, New York, New York, 1983. Traducción al castellano en [Ahuja; Walsh 1989]
- Ahuja; Walsh 1989. *Ingeniería de costos y admistración de proyectos* / Hira N. Ahuja; Michael A. Walsh. Alfaomega, México D. F., México, 1989
- Alcaraz 2001. Algoritmos Genéticos para Programación de Proyectos con Recursos Limitados / Javier Alcaraz. Universidad Politécnica de Valencia, Tesis Doctoral, 2001
- Alcaraz; Maroto 2001. A Robust Genetic Algorithm for Resource Allocation in Project Scheduling / J. Alcaraz; C. Maroto. *Annals of Operations Research* 102 (2001): 83-109
- Al-Fawzan; Haouari 2005. A bi-objective model for robust resource-constrained project scheduling / M. A. Al-Fawzan; Mohamed Haouari. *International Journal of Production Economics* 96, no. 2 (2005): 175-187
- Alquier et al. 2002. Analysis of External and Internal Risks in Project Early Phase / Anne Marie Alquier; Enrico Cagno; Franco Caron; V. Leopoulos; Miguel A. Ridao. in *The Frontiers of Project Management Research* / Dennis P. Slevin; David I. Cleland; Jeffrey K. Pinto (ed.), Project Management Institute, Newton Square, Pennsylvania, 2002. 349-363 (capítulo 21)
- Anderson et al. 2005. *An Introduction to Management Science: Quantitative Approaches to Decision Making* / David R. Anderson; Dennis J. Sweeney; Thomas A. Williams. Thomson, Mason, Ohio, 2005. 11ª edición
- Antill; Woodhead 1970. *Critical Path Methods in Construction Practice* / James M. Antill; Ronald W. Woodhead. Wiley, New York, New York, 1970. 2ª edición. 4ª edición en 1990. Traducción al castellano en [Antill; Woodhead 1995]
- Antill; Woodhead 1995. *Método de la ruta crítica y su aplicación a la construcción* / James M. Antill; Ronald W. Woodhead. Limusa, México D. F., México, 1995
- ANSI 2001. ANSI/PMI 99-001-2000. American National Standards Institute, 2000. Aprobado en marzo de 2001.

- Arranz 1993. *Planificación y control de proyectos* / Antonio Arranz Ramonet. Megabyte, México D. F., México, 1993
- Artigues et al. 1999. Characterization of a set of schedules in a resource constrained multi-project scheduling problem with multiple modes / Christian Artigues; Françoise Roubellat; Jean-Charles Billaut. *International Journal of Industrial Engineering* 6, no. 2 (1999): 112-122
- Artigues; Roubellat 2000. A polynomial activity insertion algorithm in a multi-resource schedule with cumulative constraints and multiple modes / Christian Artigues; Françoise Roubellat. *European Journal of Operational Research* 127, no. 2 (2000): 297-316
- Avramidis; Wilson 1993a. A splitting scheme for control variates / Athanassios N. Avramidis; James R. Wilson. *Operations Research Letters* 14 (1993): 187-198
- Avramidis; Wilson 1993b. Integrated variance reduction strategies / Athanassios N. Avramidis; James R. Wilson. in *Proceedings of the 1993 Winter Simulation Conference* / G. W. Evans; M. Mollaghasemi; E. C. Russell; W. E. Biles (eds.), IEEE, Piscataway, 1993. 445-454
- Avramidis; Wilson 1996. Integrated variance reduction strategies for simulation / Athanassios N. Avramidis; James R. Wilson. *Operations Research* 44, no. 2 (1996): 327-346
- Baar et al. 1999. Tabu search algorithms and lower bounds for the resource-constrained project scheduling problem / Tonius Baar; Peter Brucker; Sigrid Knust. in *Metaheuristics* / S. Voss; S. Martello; I. H. Osman; C. Roucairol (eds.), Kluwer, 1999. 1-18
- Baca 1990. *Evaluación de proyectos: Análisis y administración del riesgo* / Gabriel Baca Urbina. McGraw-Hill, México D. F., México, 1990. 4ª edición en 2001
- Baker et al. 1998. Techniques for the analysis of risk in major projects / S. Baker; D. Ponniah; S. Smith. *Journal of the Operational Research Society* 49, no. 6 (1998): 567-572
- Balcombe; Smith 1999. Refining the Use of Monte Carlo Techniques for Risk Analysis in Project Planning / K. G. Balcombe; L. E. D. Smith. *The Journal of Development Studies* 36, no. 2 (1999): 113-135
- Ballestín 2001. Nuevos métodos de resolución del problema de secuenciación de proyectos con recursos limitados / Francisco Ballestín. Departamento de Estadística e Investigación Operativa, Universidad de Valencia, Spain, Ph. D. Thesis, 2001
- Bandelloni et al. 1994. Optimal resource levelling using non-serial dynamic programming / M. Bandelloni; M. Tucci; R. Rinaldi. *European Journal of Operational Research* 78, no. 2 (1994): 162-177
- Bartusch et al. 1988. Scheduling project networks with resource constraints and time windows / M. Bartusch; Rolf H. Möhring; Franz J. Radermacher. *Annals of Operations Research* 16, no. 1-4 (1988): 201-240
- Battersby 1964. *Network analysis for planning and scheduling* / Albert Battersby. MacMillan, London, United Kingdom, 1964. 3ª edición en 1970
- Bell; Han 1991. A New Heuristic Solution Method in Resource-Constrained Project Scheduling / Colin E. Bell; Jaemin Han. *Naval Research Logistics* 38, no. 3 (1991): 315-331

- Benjamin et al. 1990. Knowledge-Based Prototype for Improving Scheduling Productivity / C. O. Benjamin; D. L. Babcock; N. B. Yunus; J. Kincaid. *Journal of Computing in Civil Engineering* 4, no. 2 (1990): 124-134
- Berny 1989. A New Distribution Function for Risk Analysis / J. Berny. *Journal of the Operational Research Society* 40, no. 12 (1989): 1121-1127
- Bey et al. 1981. The Net Present Value Criterion: Its Impact on Project Scheduling / Roger B. Bey; Robert H. Doersch; James H. Patterson. *Project Management Quarterly* 12, no. 2 (1981): 35-45
- Bidot 2005. A General Framework Integrating Techniques for Scheduling under Uncertainty / Julien Bidot. Ecole Nationale d'Ingénieurs de Tarbes – Institut National Polytechnique de Toulouse, France, Ph. D. Thesis, Serial Number 2297, 2005
- Błażewicz et al. 1983. Scheduling subject to resource constraints / J. Błażewicz; J. K. Lenstra; A. H. G. Rinnooy Kan. *Discrete Applied Mathematics* 5 (1983): 11-24
- Błażewicz et al. 1985. *Scheduling in Computer and Manufacturing Systems* / Jacek Błażewicz; Klaus H. Ecker; Erwin Pesch; Günter Schmidt; Jan Węglarz. Springer, Berlin, Deutschland, 1985. 2ª edición en 2002.
- Błażewicz et al. 1986. *Scheduling under Resource Constraints: Deterministic Models* / Jacek Błażewicz; Wojciech Cellary; Roman Słowiński; Jan Węglarz. Baltzer, Basel, Switzerland, 1986
- Bowers 2000. Interpreting float in resource constrained projects / J. A. Bowers. *International Journal of Project Management* 18 (2000): 386-392
- Brandt 1999. Computer Generated Random Numbers: The Monte Carlo Method / Siegmund Brandt. in *Data Analysis: Statistical and Computational Methods for Scientists and Engineers* / Siegmund Brandt, Springer, Berlin, Deutschland, 1999. 49-86 (capítulo 4º)
- Brassard; Bratley 1987. *Algorithmique: Conception et analyse* / Gilles Brassard; Paul Bratley. Masson, Paris, 1987. Traducción al castellano en [Brassard; Bratley 1990].
- Brassard; Bratley 1990. *Algorítmica* / Gilles Brassard; Paul Bratley. Masson, Barcelona, 1990
- Briand et al. 2002. Scheduling with time lags and preferences / Cyril Briand; Emmanuelle Despontin; Francis Roubellat. in *Eighth International Workshop on Project Management and Scheduling*, València, 2002.
- Brucker 1998. *Scheduling Algorithms* / Peter Brucker. Springer, Berlin, Deutschland, 1998. 4ª edición en 2004.
- Brucker et al. 1999. Resource-constrained project scheduling: Notation, classification, models, and methods / Peter Brucker; Andreas Drexl; Rolf H. Möhring; Klaus Neumann; Erwin Pesch. *European Journal of Operational Research* 112, no. 1 (1999): 3-41
- Brucker; Knust 2000. A linear programming and constraint propagation-based lower bound for the RCPSP / Peter Brucker; Sigrid Knust. *European Journal of Operational Research* 127 (2000): 355-362
- Buckley 1989. Fuzzy PERT / J. J. Buckley. in *Applications of the Fuzzy Set Methodologies in Industrial Engineering* / G. Evans; W. Karwowski; M. Wilhelm (eds.), Elsevier, 1989. 103-114

- Bushuyev; Sochnev 1999. Entropy measurement as a project control tool / S. D. Bushuyev; S. V. Sochnev. *International Journal of Project Management* 17, no. 6 (1999): 343-350
- Burke 1992. *Project Management: Planning and Control* / Rory Burke. Wiley, Chichester, UK, 1992. 2ª edición en 1994.
- Carlier; Chretienne 1988. Timed Petri-net schedules / J. Carlier; P. Chretienne. in *Advances in Petri Nets* / G. Rozenberg (ed.), Springer, Berlin, Deutschland, 1988. 62-84
- Chadbourne 2000. Put Risk Management Training Wheels on Your Project Support Office / Bruce C. Chadbourne. in *Proceedings of the Project Management Institute Annual Seminars & Symposium 2000*
- Chang; Christensen 1999. A Net Practice for Software Project Management / Carl K. Chang; Mark Christensen. *IEEE Software* 16, no. 6 (1999): 80-89
- Chapman 2006. Key points of contention in framing assumptions for risk and uncertainty management / Chris Chapman. *International Journal of Project Management* 24, no. 4 (2006): 303-313
- Chapman et al. 2000. Incorporating uncertainty in competitive bidding / C. B. Chapman; S. C. Ward; J. A. Bennell. *International Journal of Project Management* 18, no. 5 (2000): 337-347
- Chapman; Ward 1997. *Project Risk Management: Processes, Techniques and Insights* / Chris B. Chapman; Stephen C. Ward. John Wiley & Sons, Chichester, UK, 1997. 2ª edición 2003
- Chapman; Ward 2000. Estimation and evaluation of uncertainty: a minimalist first pass approach / Chris Chapman; Stephen Ward. *International Journal of Project Management* 18 (2000): 369-383
- Chau 1995. Monte-Carlo simulation of construction costs using subjective data / K. W. Chau. *Construction Management and Economics* 13, no. 5 (1995): 369-383
- Chretienne 1983. Les réseaux de Petri temporisés / Philippe Chretienne. Université Paris VI, Paris, France, Thèse d'état, 1983
- Clough et al. 1972. *Construction Project Management* / Richard H. Clough; Glenn A. Sears; S. Keoki Sears. Wiley, New York, New York, 1972. 4ª edición en 2000
- Coronado; Jaén 2002. A Software Project Management Method: A3 / Sergio Coronado; José Alberto Jaén. in *Frontiers of Project Management Research and Application* / Dennis P. Slevin; David I. Cleland; Jeffrey K. Pinto (ed.), Project Management Institute, Pennsylvania, 2002. 171-179
- Cook 1971. The complexity of theorem-proving procedures / Stephen A. Cook. in *Proceedings of the 3rd annual ACM symposium on Theory of computing*, ACM Press, New York, New York, 1971. 151-158
- Couillard 1995. The Role of Project Risk in Determining Project Management Approach / Jean Couillard. *Project Management Journal* 26, no. 4 (1995): 3-15
- Damay 2005. Techniques de résolution basées sur la Programmation Linéaire pour l'Ordonnancement de Projet / Jean Damay. Université Blaise Pascal - Clermont II, Thèse de Doctorat, 2005

- Davis 1985. Job-shop scheduling with genetic algorithms / L. Davis. in Proceedings of the 1st International Conference on Genetic Algorithms / J. Grefenstette (ed.), Lawrence Erlbaum, Hillsdale, New Jersey, USA, 1985. 136-140
- Davis; Patterson 1975. A Comparison of Heuristic and Optimum Solutions in Resource-constrained Project Scheduling / Edward W. Davis; James H. Patterson. *Management Science* 21, no. 8 (1975): 944-955
- Dawood 1998. Estimating project and activity duration: a risk management approach using network analysis / Nashwan Dawood. *Construction Management and Economics* 16, no. 1 (1998): 41-48
- Dawson; Dawson 1998. Practical proposals for managing uncertainty and risk in project planning / Ray J. Dawson; Christian W. Dawson. *International Journal of Project Management* 16, no. 5 (1998): 299-310
- Debels; Vanhoucke 2005. A bi-population based genetic algorithm for the resource-constrained project scheduling problem / Dieter Debels; Mario Vanhoucke. Vlerick Leuwe Gent Management School, Ghent University and Katholieke Universiteit Leuven, Working Paper Series 2005/8, 2005
- de Falco; Macchiaroli 1998. Timing of control activities in project planning / Massimo de Falco; Roberto Macchiaroli. *International Journal of Project Management* 16, no. 1 (1998): 51-58
- de Klerk 2001. The Value of Project Risk Management / Antoine M. de Klerk. in Management of Engineering and Technology: PICMET'01, vol. 2, 2001. 570-576
- DeMarco; Lister 2003. *Waltzing with Bears: Managing Risk on Software Projects* / Tom DeMarco; Timothy Lister. Dorset House, New York, New York, 2003
- Demassez 2003. Méthodes Hybrides de Programmation par Contraintes et de Programmation linéaire pour le problème d'Ordonnancement de Projet à Contraintes de Ressources / Sophie Demassez. Université d'Avignon, Thèse de Doctorat, 2003
- Demeulemeester; Herroelen 1992. A branch-and-bound procedure for the multiple resource-constrained project scheduling problem / Erik Demeulemeester; Willy Herroelen. *Management Science* 38, no. 12 (1992): 1803-1818
- Demeulemeester; Herroelen 1996. An efficient optimal solution procedure for the preemptive resource constrained project scheduling problem / Erik Demeulemeester; Willy Herroelen. *European Journal of Operational Research* 90, no. 2 (1996): 334-348
- Demeulemeester; Herroelen 1997. New Benchmark Results for the Resource-Constrained Project Scheduling Problem / Erik Demeulemeester; Willy Herroelen. *Management Science* 43, no. 11 (1997): 1485-1492
- Demeulemeester; Herroelen 2002. *Project Scheduling: A Research Handbook* / Erik Demeulemeester; Willy Herroelen. Kluwer, Boston, Massachusetts, 2002
- Demeulemeester et al. 2000. The discrete time/resource trade-off problem in project networks: A branch and bound approach / Erik Demeulemeester, Bert De Reyck, Willy Herroelen. *IIE Transactions* 32, no. 11 (2000): 1059-1069
- DePuy; Whitehouse 2000. Applying the COMSOAL computer heuristic to the constrained

- resource allocation problem / Gail W. DePuy; Gary E. Whitehouse. *Computers & Industrial Engineering* 38, no. 3 (2000): 413-422
- DePuy; Whitehouse 2001. A simple and effective heuristic for the resource constrained project scheduling problem / Gail W. DePuy; Gary E. Whitehouse. *International Journal of Production Research* 39, no. 14 (2001): 3275-3287
- De Reyck; Herroelen 1999. The multi-mode resource-constrained project scheduling problem with generalized precedence relations / Bert De Reyck; Willy Herroelen. *European Journal of Operational Research* 119, no. 2 (1999): 538-556
- Devaux 1999. *Total Control Project: A manager's guide to integrated project planning, measuring, and tracking* / Stephen A. Devaux. Wiley, New York, 1999
- Dillon et al. 2005. Optimal Use of Budget Reserves to Minimize Technical and Management Failure Risks During Complex Project Development / R. L. Dillon; M. E. Paté-Cornell; S. D. Guikema. *IEEE Transactions on Engineering Management* 52, no. 3 (2005): 382-395
- Dinic 1990. The fastest algorithm for the PERT problems with AND- and OR-nodes / E. Dinic. in Proceedings of the Workshop on Combinatorial Optimization, University of Waterloo Press, Waterloo, Ontario, Canada, 1990. 185-187
- Dorigo; Di Caro 1999. The Ant Colony Optimization Meta-Heuristic / Marco Dorigo; Gianni Di Caro. in *New Ideas in Optimization* / David Corne; Marco Dorigo; Fred Glover (eds.), McGraw-Hill, London, 1999. 11-32
- Dorndorf et al. 1999. A branch-and-bound algorithm for the resource-constrained project scheduling problem / Ulrich Dorndorf; Erwin Pesch; Toàn Phan-Huy. Institute für Gesellschafts- und Wirtschaftswissenschaften, Universität Bonn, Germany, 1999
- Dorndorf et al. 2000. A branch-and-bound algorithm for the resource-constrained project scheduling problem / Ulrich Dorndorf; Erwin Pesch; Toàn Phan-Huy. *Mathematical Methods of Operations Research* 52, no. 3 (2000): 413-439
- Dorp; Duffey 1999. Statistical dependence in risk analysis for project networks using Monte Carlo methods / J. R. van Dorp; M. R. Duffey. *International Journal of Production Economics* 58, no. 1 (1999): 17-29
- Drexl et al. 2000. ProGen/ $\pi\chi$ / Andreas Drexl; Rüdiger Nissen; James H. Patterson; Frank Salewski. *European Journal of Operational Research* 125, no. 1 (2000): 59-72
- Dubois et al. 2003. Fuzzy scheduling: Modelling flexible constraints vs. coping with incomplete knowledge / Didier Dubois; H el ene Fargier; Philippe Fortemps. *European Journal of Operational Research* 147, no. 2 (2003): 231-252
- Eckhart 1987. Stan Ulam, John von Neumann, and the Monte Carlo method / Roger Eckhart. *Los Alamos Science Special Issue*, no. 15 (1987): 131-137
- Eglese 1990. Simulated Annealing: A tool for Operational Research / R. W. Eglese. *European Journal of Operational Research* 46, no. 3 (1990): 271-281
- Eisner 1962. A Generalized Network Approach to the Planning and Scheduling of a Research Project / Howard Eisner. *Operations Research* 10, no. 1 (1962): 115-125
- Elkhyari et al. 2002. Explanation-based repair techniques for solving dynamic scheduling problems / Abdallah Elkhyari; Christelle Gu eret; Narendra Jussien. in Working Notes of

- the AIPS'02 Workshop on On-line Planning and Scheduling, Toulouse, France, 2002
- Elkhyari et al. 2003. Solving dynamic RCPSP using explanations-based constraint programming / Abdallah Elkhyari; Christelle Guéret; Narendra Jussien. in MAPSP'03: Sixth workshop on Models and Algorithms for Planning and Scheduling Problems, Aussois, France, 2003. 119-120
- Elkjaer; Felding 1999. Applied Project Risk Management: Introducing the Project Risk Management Loop of Control / Martin Elkjaer; Finn Felding. *Project Management* 5, no. 1 (1999): 16-25
- Elkjaer 2000. Stochastic Budget Simulation / Martin Elkjaer. *International Journal of Project Management* 18, no. 2 (2000): 139-147
- Elmaghraby 1977. *Activity Networks: Project planning and control by network models* / Salah Eldin Elmaghraby. Wiley-Interscience, New York, 1977
- Elmaghraby 1990. Project bidding under deterministic and probabilistic activity durations / Salah Eldin Elmaghraby. *European Journal of Operational Research* 49 (1990): 14-34
- Elmaghraby 2000. On criticality and sensitivity in activity networks / Salah E. Elmaghraby. *European Journal of Operational Research* 127, no. 2 (2000): 220-238
- Elmaghraby 2005. On the fallacy of averages in project risk management / Salah E. Elmaghraby. *European Journal of Operational Research* 165, no. 2 (2005): 307-313
- Elmaghraby; Kamburowski 1992. The Analysis of Activity Networks under Generalized Precedence Relations (GPR) / Salah E. Elmaghraby; Jerzy Kamburowski. *Management Science* 38, no. 9 (1992): 1245-1263
- Engwall; Jerbrant 2003. The resource allocation syndrome: The prime challenge of multi-project management? / Mats Engwall; Anna Jerbrant. *International Journal of Project Management* 21, no. 6 (2003): 403-409
- Esquirol et al. 1995. Constraint Programming / P. Esquirol; P. Lopez; H. Fargier; T. Schiex. *Belgian Journal of Operational Research* 35, no. 2 (1995): 5-36
- Evans et al. 1993. The Denver airport: Managing a megaproject / Ginger S. Evans; Richard F. Haury; Guy M. Stricklin. *Civil Engineering* 63, no. 5 (1993)
- Fan; Yu 2004. BBN-based software project risk management / Chin-Feng Fan; Yuan-Chang Yu. *Journal of Systems and Software* 73, no. 2 (2004): 193-203
- Fenbert; Fleener 2002. Implementing TOC Multiproject Management in a Research Organization / Jeffrey A. Fenbert; Nancy K. Fleener. in *Frontiers of Project Management Research and Application* / Dennis P. Slevin; David I. Cleland; Jeffrey K. Pinto (ed.), Project Management Institute, Pennsylvania, 2002. 225-229
- Feo; Resende 1989. A probabilistic heuristic for a computationally difficult set covering problem / T. A. Feo; M. G. C. Resende. *Operations Research Letters* 8 (1989): 67-71
- Fisher et al. 1995. Benchmarking in Construction Industry / Deborah Fisher; Susan Miertschin; David R. Pollock Jr. *Journal of Management in Engineering* 11, no. 1 (1995): 50-57
- Fishman 1996. *Monte Carlo Concepts, Algorithms, and Applications* / George S. Fishman. Springer, New York, 1996

- Ford; Sterman 2003a. Overcoming the 90% syndrome: Iteration Management in Concurrent Development Projects / David N. Ford; John D. Sterman. *Concurrent Engineering* 11, no. 3 (2003): 177-186
- Fox 1986. Implementation and Relative Efficiency of Quasirandom Sequence Generators / Bennett L. Fox. *ACM Transactions on Mathematical Software* 12, no. 4 (1986): 362-376
- Fulkerson 1961. A Network Flow Computation for Project Cost Curves / D. R. Fulkerson. *Management Science* 7 (1961): 167-178
- Fulkerson 1962. Expected Critical Path Lengths in PERT Networks / D. R. Fulkerson. *Operations Research* 10 (1962): 808-817
- Gantt 1910. Work, Wages and Profit / Henry Laurence Gantt. *The Engineering Magazine* (1910)
- Garey; Johnson 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness* / M. R. Garey; P. S. Johnson. Freeman, San Francisco, California, 1979
- Ghomi; Ashjari 2002. A simulation model for multi-project resource allocation / S. M. T. Fatemi Ghomi; B. Ashjari. *International Journal of Project Management* 20, no. 2 (2002): 127-130
- Gillies; Liu 1995. Scheduling tasks with AND/OR precedence constraints / Donald W. Gillies; Jane W.-S. Liu. *SIAM Journal on Computing* 24, no. 4 (1995): 797-810
- Glover 1989. Tabu Search: Part I / Fred Glover. *ORSA Journal on Computing* 1, no. 3 (1989): 190-206
- Glover 1990. Tabu Search: Part II / Fred Glover. *ORSA Journal on Computing* 2, no. 1 (1990): 4-32
- Glover; Laguna 1997. *Tabu Search* / Fred W. Glover; Manuel Laguna. Kluwer, Norwell, Massachusetts, 1997. Nueva edición en Springer, Berlin, Deutschland, 2004
- Goldratt 1997. *Critical Chain* / Eliyahu M. Goldratt. The North River Press Publishing Corporation, Great Barrington, 1997. Traducción al castellano en [Goldratt 2001]
- Goldratt 2001. *Cadena Crítica: Una novela empresarial sobre la Gestión de Proyectos* / Eliyahu M. Goldratt. Díaz de Santos, Madrid, 2001
- Golenko-Ginzburg 1988. On the distribution of activity time in PERT / Dimitri Golenko-Ginzburg. *Journal of the Operational Research Society* 39, no. 8 (1988): 767-771
- Golenko-Ginzburg; Blokh 1997. A generalized activity network model / D. Golenko-Ginzburg; D. Blokh. *Journal of the Operational Research Society* 48, no. 4 (1997): 391-400
- Golenko-Ginzburg et al. 2000. Resource supportability model for stochastic network projects under a chance constraint / Dimitri Golenko-Ginzburg; Aharon Gonik; Shimon Sitniakovski. *Communications in Dependability and Quality Management* 3, no. 1 (2000): 89-102
- Golenko-Ginzburg et al. 2001a. Algorithms of Optimal Supply of Resources to a Group of Projects: (Stochastic Networks) / Dimitri I. Golenko-Ginzburg; Sergey M. Lyubkin; Vladimir S. Rezer; Shimon L. Sitnyakovskii. *Avtomatika i Telemekhanika*, no. 8 (2001): 157-167. Original en ruso, traducido en [Golenko-Ginzburg et al. 2001b]

- Golenko-Ginzburg et al. 2001b. Algorithms of Optimal Supply of Resources to a Group of Projects: (Stochastic Networks) / Dimitri I. Golenko-Ginzburg; Sergey M. Lyubkin; Vladimir S. Rezer; Shimon L. Sitnyakovskii. *Automation and Remote Control* 62, no. 8 (2001): 1366-1375
- Golenko-Ginzburg; Gonik 1997. Stochastic network project scheduling with non-consumable limited resources / Dimitri Golenko-Ginzburg; Aharon Gonik. *International Journal of Production Economics* 48, no. 1 (1997): 29-37
- Golenko-Ginzburg; Gonik 1998a. A heuristic for network project scheduling with random activity durations depending on the resource allocation / Dimitri Golenko-Ginzburg; Aharon Gonik. *International Journal of Production Economics* 55 (1998): 149-162
- Golenko-Ginzburg; Gonik 1998b. High performance heuristic algorithm for controlling stochastic network projects / Dimitri Golenko-Ginzburg; Aharon Gonik. *International Journal of Production Economics* 54, no. 3 (1998): 235-245
- Gordon; Tulip 1997. Resource scheduling / J. Gordon; A. Tulip. *International Journal of Project Management* 15, no. 6 (1997): 359-370
- Graham et al. 1979. Optimisations and approximation in deterministic sequencing and scheduling: A survey / R. L. Graham; E. L. Lawler; J. K. Lenstra; A. H. G. Rinnooy Kan. *Annals of Discrete Mathematics* 5 (1979): 287-326
- Grinold 1972. The payment scheduling problem / Richard C. Grinold. *Naval Research Logistics Quarterly* 19 (1972): 123-136
- Gurney 1997. An Introduction to Neural Networks / Kevin Gurney. UCL Press, London, England, United Kingdom, 1997
- Gutjahr et al. 2000. A stochastic branch-and-bound approach to activity crashing in project management / W. J. Gutjahr; C. Strauss; E. Wagner. *INFORMS Journal on Computing* 12, no. 2 (2000): 125-135
- Hall et al. 1996. Scheduling To Minimize Average Completion Time: Off-line and On-line Algorithms / Leslie A. Hall; David B. Shmoys; Joel Wein. in Proceedings of the 7th annual ACM-SIAM Symposium on Discrete Algorithms, 1996. 142-151
- Hammersley; Handscomb 1964. *Monte Carlo Methods* / J. M. Hammersley; D. C. Handscomb. Methuen, London, United Kingdom, 1964. 3ª edición en 1967
- Hans 2001. Resource Loading by Branch-and-Price Techniques / Erwin Hans. Universiteit Twente, Nederlanden, Ph. D. Thesis, 2001
- Harris 1978. *Precedence and arrow networking techniques for construction* / Robert B. Harris. Wiley, New York, 1978
- Hartley; Wortham 1966. A statistical theory for pert critical path analysis / H. O. Hartley; A. W. Worthman. *Management Science* 12, no. 10 (1966): b469-b487
- Hartmann 1998. A Competitive Genetic Algorithm for Resource-Constrained Project Scheduling / Sönke Hartmann. *Naval Research Logistics* 45, no. 7 (1998): 733-750
- Hartmann 2002. A self-adapting genetic algorithm for project scheduling under resource constraints / Sönke Hartmann. *Naval Research Logistics* 49, no. 5 (2002): 433-448
- Hartmann; Kolisch 2000. Experimental evaluation of state-of-the-art heuristics for the

- resource-constrained project scheduling problem / Sönke Hartmann; Rainer Kolisch. *European Journal of Operational Research* 127, no. 2 (2000): 394-407
- Haas; Schedler 1987. Stochastic Petri Net Representation of Discrete Event Simulations / Peter J. Haas; Gerald S. Shedler. in International Workshop on Petri Nets and Performance Models, PNPM'87, 1987. Publicado en [Haas; Schedler 1989]
- Haas; Schedler 1989. Stochastic Petri Net Representation of Discrete Event Simulations / Peter J. Haas; Gerald S. Shedler. *IEEE Transactions on Software Engineering* 15, no. 4 (1989): 381-393
- Hapke et al. 1994. Fuzzy project scheduling system for software development / Maciej Hapke; Andrzej Jaskiewicz; Roman Słowiński. *Fuzzy Sets and Systems* 67, no. 1 (1994): 101-117
- Haykin 1994. Neural Networks: A Comprehensive Foundation / Simon Haykin. Prentice Plenum Press, New York, New York, 1994. 2ª edición en 1999
- Hebb 1949. The Organization of Behaviour / Donald O. Hebb. Wiley, New York, New York, 1949
- Heckerman; Wellman 1995. Bayesian networks / David Heckerman; Michael P. Wellman. *Communications of the ACM* 38, no. 3 (1995): 23-30
- Hegazy 1999a. Optimization of resource allocation and leveling using genetic algorithms / Tarek Hegazy. *Journal of Construction Engineering and Management* 125, no. 3 (1999): 167-175
- Hegazy 1999b. Optimization of construction time-cost trade-off analysis using genetic algorithms / Tarek Hegazy. *Canadian Journal of Civil Engineering* 26, no. 6 (1999): 685-697
- Hegazy 2002. *Computer-Based Construction Project Management* / Tarek Hegazy. Prentice Hall, Upper Saddle River, New Jersey, USA, 2002
- Hegazy et al. 2004. Distributed scheduling model for infrastructure networks / Tarek Hegazy; Ahmed Elhakeem; Emad Elbeltagi. *Journal of Construction Engineering and Management* 130, no. 2 (2004): 160-167
- Hegazy; Kassab 2003. Resource optimization using combined simulation and genetic algorithms / Tarek Hegazy; Moustafa Kassab. *Journal of Construction Engineering and Management* 129, no. 6 (2003): 698-705
- Hegazy; Petzold 2003. Genetic optimization for dynamic project control / Tarek Hegazy; Kevin Petzold. *Journal of Construction Engineering and Management* 129, no. 4 (2003): 395-404
- Heilmann 2003. A branch-and-bound procedure for the multi-mode resource-constrained project scheduling problem with minimum and maximum time lags / Roland Heilmann. *European Journal of Operational Research* 144, no. 2 (2003): 348-365
- Herroelen et al. 1997. Project network models with discounted cash flows: a guided tour through recent developments / Willy Herroelen; Patrick Van Dommelen; Erik Demeulemeester. *European Journal of Operational Research* 100, no. 1 (1997): 97-121
- Herroelen et al. 1999. A Classification Scheme for Project Scheduling / Willy Herroelen; Erik

- Demeulemeester; Bert De Reyck. in *Project Scheduling* / J. Węglarz (ed.), Kluwer Academic, Boston, Massachusetts, 1999. 1-26
- Herroelen; Leus 2001. On the merits and pitfalls of critical chain scheduling / W. Herroelen; R. Leus. *Journal of Operations Management* 19, no. 5 (2001): 559-577
- Herroelen; Leus 2004. The construction of stable project baseline schedules / Willy Herroelen; Roel Leus. *European Journal of Operational Research* 156, no. 3 (2004): 550-565
- Herroelen; Leus 2005a. Identification and illumination of popular misconceptions about project scheduling and time buffering in a resource-constrained environment / Willy Herroelen; Roel Leus. *Journal of the Operational Research Society* 56, no. 1 (2005): 102-109
- Herroelen; Leus 2005b. Project scheduling under uncertainty: survey and research potentials / Willy Herroelen; Roel Leus. *European Journal of Operational Research* 165, no. 2 (2005): 289-306
- Hillson 2000. Project Risks: identifying causes, risks and effects / David Hillson. *PM Network* 14, no. 9 (2000): 48-51
- Hindi et al. 2005. An evolutionary algorithm for resource-constrained project scheduling / Khalil S. Hindi; Hongbo Yang; Krzysztof Fleszar. *IEEE Transactions on Evolutionary Computation* 6, no. 5 (2002): 512-518
- Hiyassat 2000. Modification of minimum moment approach in resource leveling / Mohammed A. Salem Hiyassat. *Journal of Construction Engineering and Management* 126, no. 4 (2000): 278-284
- Hiyassat 2001. Applying modified minimum moment method to multiple resource leveling / Mohammed A. Salem Hiyassat. *Journal of Construction Engineering and Management* 127, no. 3 (2001): 192-198
- Holland 1975. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence* / John H. Holland. University of Michigan Press, Ann Arbor, Michigan, 1975
- Hopcroft; Motwani; Ullman 2006. *Introduction to Automata Theory, Languages, and Computation* / John E. Hopcroft; Rajeev Motwani; Jeffrey D. Ullman. Addison-Wesley, Reading, Massachusetts, 2006
- Hopcroft; Ullman 1979. *Introduction to Automata Theory, Languages, and Computation* / John E. Hopcroft; Jeffrey D. Ullman. Addison-Wesley, Reading, Massachusetts, 1979. 3ª edición con Rajeev Motwani en [Hopcroft; Motwani; Ullman 2006]
- Hopfield 1982. Neural networks and physical systems with emergent collective computational abilities / J. J. Hopfield. in *Proceedings of the National Academy of Sciences*, vol. 79, 1982. 2554-2558
- Horowitz; Sahni 1978. *Fundamentals of Computer Algorithms* / Ellis Horowitz; Sartaj Sahni. Computer Science, Potomac-Rockville, Maryland, 1978
- Houghton et al. 2001. LANL's development of schedule contingency based on probabilistic risk results / F. Kay Houghton; John P. Kindingerm; David Gemeinhart; George White. in

- Proceedings of the PMI Annual Symposium, (Nashville, Tennessee), 2001.-5
- Hughes 1986. Why Projects Fail: The Effects of Ignoring the Obvious / Michael William Hughes. *IIE Solutions* 18, no. 4 (1986): 14-18
- Hughes; Wilcox 1996. An Innovative Distributed Project Control Approach: A case study of the Cassini Management Information System (CMIS) / Michael W. Hughes; Reed E. Wilcox. *Engineering Management Journal* 8, no. 1 (1996): 9-14
- IEEE 2004. IEEE Guide Adoption of PMI Standard A Guide to the Project Management Body of Knowledge. IEEE, Std 1490-2003, 2003. Aprobado el 24 de mayo de 2004.
- Jaafari 2001. Management of risk, uncertainties and opportunities on projects: time for a fundamental shift / Ali Jaafari. *International Journal of Project Management* 19, no. 2 (2001): 89-101
- Jaśkowski; Sobotka 2004. Scheduling construction projects with resources accessibility limited and changeable in time / Piotr Jaśkowski; Anna Sobotka. *Journal of Civil Engineering and Management* 10, no. 4 (2004): 267-276
- Jeetendra et al. 2000. Petri Nets for Project Management and Resource Levelling / V. A. Jeetendra; O. V. Krishnaiah Chetty; J. Prashanth Reddy. *International Journal of Advanced Manufacturing Technology* 16, no. 7 (2000): 516-520
- Jensen et al. 2006. Project relationships: A model for analyzing interactional uncertainty / Christian Jensen; Staffan Johansson; Mikael Löfström. *International Journal of Project Management* 24, no. 1 (2006): 4-12
- Johnson et al. 1970. Continuous univariate distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan. Wiley, New York, New York, 1970. 2ª edición en 1994 (vol. 1) y 1995 (vol. 2)
- Johnson et al. 1994a. Normal Distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan. in Continuous univariate distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan, Wiley, 1994. 80-206 (capítulo 13°)
- Johnson et al. 1994b. Lognormal Distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan. in Continuous univariate distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan, Wiley, 1994. 207-297 (capítulo 14°)
- Johnson et al. 1995a. Uniform (Rectangular) Distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan. in Continuous univariate distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan, Wiley, 1995. 276-321 (capítulo 26°)
- Johnson et al. 1995b. Beta Distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan. in Continuous univariate distributions / Norman L. Johnson; Samuel Kotz; N. Balakrishnan, Wiley, 1995. 210-275 (capítulo 25°)
- Johnson 1997. The triangular distribution as a proxy for the beta distribution in risks analysis / David Johnson. *Journal of the Royal Statistical Society* 46, no. 3 (1997): 387-398
- Jørgensen 1999. Project Scheduling as a Stochastic Dynamic Decision Problem / Trond Jørgensen. Norges teknisk-naturvitenskapelige universitet (Norwegian University of Science and Technology), Trondheim, Noreg, Doktoravhandling, 1999
- Józefowska et al. 2000. Solving discrete-continuous project scheduling problems via its

- discretization / Joanna Józefowska; Jan Węglarz; Marek Mika; Rafał Różycki; Grzegorz Waligóra. *Mathematical Methods of Operations Research* 52 (2000): 489-499
- Józefowska et al. 2001. Simulated Annealing for Multi-Mode Resource-Constrained Project Scheduling / Joanna Józefowska; Marek Mika; Rafał Różycki; Grzegorz Waligóra; Jan Węglarz. *Annals of Operations Research* 102 (2001): 137-155
- Józefowska et al. 2003. Uncertainty in discrete-continuous project scheduling / Joanna Józefowska; Rafał Różycki; Jan Węglarz. in *Project Driven Manufacturing* / Z. Banaszak; Joanna Józefowska (eds.), WNT Warszawa, Warszawa, Polska, 2003. 23-33
- Józefowska; Węglarz 2006. *Perspectives in Modern Project Scheduling* / Joanna Józefowska; Jan Węglarz (eds.). Springer, Berlin, Deutschland, 2006
- Kao et al. 2002. I²QFD: a blackboard-based multiagent system for supporting concurrent engineering projects / Hsing-Pei Kao; Eric Su; Brian Wang. *International Journal of Production Research* 40, no. 5 (2002): 1235-1262
- Karp 1972. Reducibility among combinatorial problems / Richard M. Karp. in *Complexity of Computer Computations* / R. E. Miller; J. W. Thatcher (eds.), Plenum, New York, New York, 1972. 85-103
- Kartam; Tongthong 1998. An artificial neural network for resource leveling problems / Nabil Kartam; Tanit Tongthong. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 12, no. 3 (1998): 273-287
- Kelley 1961. Critical-path planning and scheduling: Mathematical Basis / J. E. Kelley. *Operations Research* 9 (1961): 296-320
- Kelley 1963. The Critical Path Method: Resources, Planning and Scheduling / J. E. Kelley. in *Industrial Scheduling* / J. F. Muth; G. L. Thompson (eds.), Prentice-Hall, 1963
- Kelley; Walker 1959. Critical path planning and scheduling / James E. Kelley Jr.; Morgan R. Walker. in *Proceedings of the Eastern Joint Computer Conference*, Boston, Massachusetts, 1959. 160-173
- Kennedy; Eberhart 1995. Particle swarm optimization / James Kennedy; Russell Eberhart. in *Proceedings of the IEEE Conference on neural networks*, vol. 4, Piscataway, New Jersey, 1995. 1942-1948
- Kerzner 1998. *Project Management: A Systems Approach to Planning, Scheduling and Controlling* / Harold Kerzner. Van Nostrand Reinhold, New York, New York, 1998. 8^a edición Wiley, Hoboken, New Jersey, 2003.
- Kim; de la Garza 2003. Phantom Float / Kyunghwan Kim; Jesús M. De la Garza. *Journal of Construction Engineering and Management* 129, no. 5 (2003): 507-517
- Kim; de la Garza 2005. Evaluation of the resource-constrained critical path method algorithms / Kyunghwan Kim; Jesús M. De la Garza. *Journal of Construction Engineering and Management* 131, no. 5 (2005): 522-532
- Kim et al. 2003. Hybrid genetic algorithm with fuzzy logic for resource-constrained project scheduling / Kwan Woo Kim; Mitsuo Gen; Genji Yamazaki. *Applied Soft Computing* 2, no. 3 (2003): 174-188
- Kirkpatrick et al. 1983. Optimization by Simulated Annealing / S. Kirkpatrick; C. D. Gelatt;

- M. P. Vecchi. *Science* 220, no. 4598 (1983): 671-680
- Kis 2005. A branch-and-cut algorithm for scheduling projects with variable-intensity activities / Tamás Kis. *Mathematical Programming* 103, no. 3 (2005): 515-539
- Klein 1999. *Scheduling of Resource-constrained Projects* / Robert Klein. Kluwer, Dordrecht, Nederlanden, 1999. Reimpresión en 2000.
- Klein; Scholl 2000. PROGRESS: Optimally solving the generalized resource-constrained project scheduling problem / Robert Klein; Armin Scholl. *Mathematical Methods of Operations Research* 52, no. 3 (2000): 467-488
- Kloppenborg; Opfer 2000. Forty years of project management research / Timothy J. Kloppenborg; Warren A. Opfer. in *Proceedings of PMI Research Conference 2000*, Project Management Institute, Pennsylvania, 2000. 41-59
- Knuth 1969. *The Art of Computer Programming: Volume 2: Seminumerical Algorithms* / Donald Knuth. Addison-Wesley, Reading, Massachusetts, 1969. 3ª edición en 1997.
- Kochetov; Stolyar 2003. Evolutionary local search with variable neighbourhood for the resource constrained project scheduling problem / Y. A. Kochetov; A. A. Stolyar. in *Proceedings of the 3rd International Workshop of Computer Science and Information Technologies*, 2003. 96-99
- Kohonen 1982. Self-organizing formation of topologically correct feature maps / Teuvo Kohonen. *Biological Cybernetics* 43, no. 1 (1982): 59-69
- Kolisch 1995. *Project Scheduling Under Resource Constraints: Efficient Heuristics for Several Problem Classes* / Rainer Kolisch. Phisica-Verlag, Heidelberg, 1995
- Kolisch 1996. Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation / Rainer Kolisch. *European Journal of Operational Research* 90, no. 2 (1996): 320-333
- Kolisch et al. 1999. Benchmark instances for project scheduling problems / Rainer Kolisch; C. Schwindt; Arno Sprecher. in *Handbook of Recent Advances in Project Scheduling* / Jan Węglarz (ed.), Kluwer, Amsterdam, Nederlanden, 1999. 197-212
- Kolisch; Sprecher 1996. PSPLIB: A project scheduling library / Rainer Kolisch; Arno Sprecher. *European Journal of Operational Research* 96, no. 1 (1996): 205-216. <http://129.187.106.231/psplib/main.html>
- Koskela; Howell 2002. The Underlying Theory of Project Management is Obsolete / Lauri Koskela; Greg Howell. in *Frontiers of Project Management Research and Application* / Dennis P. Slevin; David I. Cleland; Jeffrey K. Pinto (ed.), Project Management Institute, Pennsylvania, 2002. 293-301
- Kovács; Váncza 2004. Completable partial solutions in constraint programming and constraint-based scheduling / András Kovács; József Váncza. in *Principles and Practice of Constraint Programming* / M. Wallace (ed.), Springer, Berlin, 2004. 332-346
- Kumar; Ganesh 1998. Use of Petri Nets for Resource Allocation in Projects / Ashok Kumar V. K.; L. S. Ganesh. *IEEE Transactions on Engineering Management* 45, no. 1 (1998): 49-56
- Kusumoto et al. 1997. A New Project Simulator Based on Generalized Stochastic Petri-net /

- Shinji Kusumoto; Osamu Mizuno; Tohru Kikuno; Yuji Hirayama; Yasunari Takagi; Keishi Sakamoto. in Proceedings of the 19th International Conference on Software Engineering, ACM Press, New York, New York, 1997. 293-302
- Laslo 2003. Activity time-cost tradeoffs under time and cost chance constraints / Zohar Laslo. *Computers & Industrial Engineering* 44, no. 3 (2003): 365-384
- Laslo et al. 2005. Alternative Stochastic Network Projects with Renewable Resources / Zohar Laslo; Dimitri Golenko-Ginzburg; Aharon Gonik. *Computer Modelling and New Technologies* 9, no. 1 (2005): 40-46
- Lawley 1992. Importance of the planning and specification phases of a CIM project / Ralph Lawley. in Proceedings of the 32nd Symposium on Instrumentation in the Pulp and Paper Industry, 1992. 93-99
- Leach 2000. *Critical Chain Project Management* / Laurence P. Leach. Artech House, Norwood, Massachusetts, USA, 2000
- Leachman et al. 1990. Resource-constrained scheduling of projects with variable-intensity activities / Robert C. Leachman; Abdurrezak Dinçerler; Sooyoung Kim. *IIE Transactions* 22, no. 1 (1990): 31-40
- Leu et al. 1999. A fuzzy optimal model for construction resource leveling scheduling / Sou-Sen Leu; An-Ting Chen; Chung-Huei Yang. *Canadian Journal of Civil Engineering* 26 (1999): 673-684
- Leu et al. 2001. A GA-Based Fuzzy Optimal Model for Construction Time-Cost Trade-Off / Sou-Sen Leu; An-Ting Chen; Chung-Huei Yang. *International Journal of Project Management* 19, no. 1 (2001): 47-58
- Leu; Hung 2002. A genetic algorithm-based optimal resource-constrained scheduling simulation model / Sou-Sen Leu; Tzung-Heng Hung. *Construction Management and Economics* 20 (2002): 131-141
- Levner et al. 2002. Makespan minimization in projects with threshold activities / Eugene Levner; Shao Chin Sung; Milan Vlach. *Asia-Pacific Journal of Operational Research* 19 (2002): 195-204
- Liberatore 2002. A Fuzzy Approach to Critical Path Analysis / Matthew J. Liberatore. in Frontiers of Project Management Research and Application / Dennis P. Slevin; David I. Cleland; Jeffrey K. Pinto (ed.), Project Management Institute, Pennsylvania, 2002. 353-360
- Liberatore; Pollack-Johnson 2003. A Macro Approach to Modeling Projects with Uncertain Network Structures / Matthew J. Liberatore; Bruce Pollack-Johnson. in Proceedings of the Portland International Conference on Management of Engineering and Technology, 2003. 246-254
- Lindemann 1998. *Performance modeling with deterministic and stochastic Petri nets* / Christoph Lindemann. Wiley, Chichester, England, United Kingdom, 1998
- Lootsma 1966. Network Planning with Stochastic Activity Durations: An Evaluation of PERT / F. A. Lootsma. *Statistica Neerlandica* 20 (1966): 43-69
- Lootsma 1967. A Gamma Distribution as a Model of an Activity Duration / F. A. Lootsma. in

- Méthodes à Chemin Critique: Congrès International 1967, Wien, Österreich. Publicado dos años después en [Lootsma 1969]
- Lootsma 1969. A Gamma Distribution as a Model of an Activity Duration / F. A. Lootsma. in *Méthodes à Chemin Critique: Actes du Congrès International 1967*, Dunod, Paris, France, 1969. 219-225
- Lorance; Wendling 1999a. Basic Techniques for Analyzing and Presenting of Cost Risk Analysis / Randal B. Lorance; Robert V. Wendling. in 1999 AACE International Transactions, AACE, Morgantown, West Virginia, USA, 1999. risk.01.1-risk.01.7
- Lorance; Wendling 1999b. Techniques for Developing Cost Risk Analysis Models / Randal B. Lorance; Robert V. Wendling. *AACE Transactions* (1999)
- López 1993. *Técnicas de programación y control de proyectos* / Carlos López Romero. Pirámide, Madrid, 1993
- Lu; Li 2003. Resource-Activity Critical-Path Method for Construction Planning / Ming Lu; Heng Li. *Journal of Construction Engineering and Management* 129, no. 4 (2003): 412-420
- Luby et al. 1995. Component-based Work Breakdown Structure (CBWBS) / Robert E. Luby; Douglas Peel; William Swahl. *Project Management Journal* 26, no. 4 (1995): 38-43
- Luh et al. 1999. Scheduling of design projects with uncertain number of iterations / Peter B. Luh; Feng Liu; Brian Moser. *European Journal of Operational Research* 113, no. 3 (1999): 575-592
- Mac Crimmon; Ryavec 1964. An Analytical Study of the PERT Assumptions / K. R. Mac Crimmon; C. A. Ryavec. *Operations Research* 12 (1964): 16-37
- Mallak; Kurstedt 1997. Planning for crises in project management / Larry A. Mallak; Harold A. Kurstedt Jr. *Project Management Journal* 28, no. 2 (1997): 14-20
- Malcolm et al. 1959. Application of a Technique for Research and Development Program Evaluation / D. G. Malcolm; J. H. Roseboom; C. E. Clark; W. Fazar. *Operations Research* 7 (1959): 646-669
- Mahler; Mazina 1982. Earned Value Reporting Earns Its Keep / F. Mahler; M. Mazina. *Cost Engineering* 24 (1982): 13-17
- Maroto et al. 1999. The evolution of software quality in project scheduling / Concepción Maroto; Pilar Tormos; Antonio Lova. in *Project Scheduling* / Jan Węglarz (ed.), Kluwer Academic, Amsterdam, Nederlanden, 1999. 239-259 (capítulo 11)
- Marsan et al. 1983. A class of generalised stochastic petri nets for the performance evaluation of multiprocessor systems / Marco Ajmone Marsan; G. Balbo; G. Conte. in *Joint International Conference on Measurement and Modeling of Computer Systems*, ACM Press, New York, New York, 1983. 198-199
- Marsan et al. 1995. *Modelling with Generalized Stochastic Petri Nets* / Marco Ajmone Marsan; G. Balbo; G. Conte; S. Donatelli; G. Franceschini. Wiley, Chichester, England, United Kingdom, 1995
- McCray et al. 2002. Project Management Under Uncertainty: The Impact of Heuristics and Biases / Gordon E. McCray; Russell L. Purvis; Colleen G. McCray. *Project Management*

- Journal* 33, no. 1 (2002): 49-57
- McCulloch; Pitts 1943. A logical calculus of the ideas immanent in nervous activity / W. McCulloch; W. Pitts. *Bulletin of Mathematical Biophysics* 7 (1943): 115-133
- McGowan 1964. MONTE CARLO Techniques Applied to PERT Networks / L. L. McGowan. Texas A & M University, M. Sc. Thesis in Statistics, 1964
- Meredith; Mantel 1995. *Project Management: A Managerial Approach* / J. R. Meredith; S. J. Mantel. John Wiley & Sons, New York, 1995
- Merkle et al. 2000. Ant Colony Optimization for Resource-Constrained Project Scheduling / Daniel Merkle; Martin Middendorf; Hartmut Schmeck. in Proceedings of the Genetic and Evolutionary Computation Conference, (Las Vegas, Nevada) 2000
- Merkle et al. 2001. Ant Colony Optimization Techniques for the Resource-Constrained Project Scheduling Problem / Daniel Merkle; Martin Middendorf; Hartmut Schmeck. Institut für Angewandte Informatik und Formale Beschreibungsverfahren, Universität Karlsruhe, Deutschland, Bericht 409, 2001
- Merkle et al. 2002. Ant colony optimization for resource-constrained project scheduling / Daniel Merkle; Martin Middendorf; Hartmut Schmeck. *IEEE Transactions on Evolutionary Computation* 6, no. 4 (2002): 333-346
- Metropolis; Ulam 1949. The Monte Carlo method / Nicholas Metropolis; Stanislaw Ulam. *Journal of the American Statistical Association* 44, no. 247 (1949): 335-341
- Mitchell 1996. *An introduction to genetic algorithms* / Melanie Mitchell. MIT, Cambridge, Massachusetts, 1996. 2ª reimpresión en 1998
- MS Project XML 2003. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/pjsdk/html/pjxml_structure_HV01051032.asp
- Moder et al. 1983. *Project Management with CPM, PERT and Precedence Diagramming* / Joseph J. Moder; Cecil R. Philips; Edward W. Davis. Van Nostrand Reinhold, New York, 1983
- Moeller 1972. VERT: A Tool to Assess Risk / Gerald L. Moeller. in 23rd Conference of the American Institute of Industrial Engineering (ahora Institute of Industrial Engineering), Anaheim, CA, 1972. 211-221
- Moeller; Digman 1978. VERT: A technique to assess risks / Gerald L. Moeller; Lester A. Digman. in 10th Annual Conference of the American Institute for Decision Sciences, 1978. 292-
- Moeller; Digman 1981. Operations planning with VERT / Gerald L. Moeller; Lester A. Digman. *Operations Research* 29, no. 4 (1981): 676-697
- Montano et al. 2000. Using the Time Petri Net Formalismo for Specificatioin, Validation, and Code Generation in Robot-Control Applications / Luis Montano; Francisco José García; José Luis Villarroel. *International Journal of Robotics Research* 20, no. 1 (2000): 59-76
- Möhring et al. 2000. Forcing relations for AND/OR precedence constraints / Rolf H. Möhring; Martin Skutella; Frederik Stork. in Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, ACM; SIAM, New York, New York; Philadelphia, Pennsylvania, 2000. 235-236

- Möhring et al. 2003. Solving Project Scheduling Problems by Minimum Cut Computations / Rolf H. Möhring; Andreas S. Schulz; Frederik Stork; Marc Uetz. *Management Science* 49, no. 3 (2003): 330-350
- Möhring et al. 2004. Scheduling with AND/OR precedence constraints / Rolf H. Möhring; Martin Skutella; Frederik Stork. *SIAM Journal on Computing* 33, no. 2 (2004): 393-415
- Möhring; Stork 2000. Linear preselective policies for stochastic project scheduling / Rolf H. Möhring; Frederik Stork. *Mathematical Methods of Operations Research* 52, no. 3 (2000): 501-515
- Mulholland; Christian 1999. Risk Assessment in Construction Schedules / B. Mulholland; J. Christian. *Journal of Construction Engineering and Management* 125, no. 1 (1999): 8-15
- Nádas 1979. Probabilistic PERT / Arthur Nádas. *IBM Journal of Research and Development* 23 (1979): 339-347
- Nelson 1987. A Perspective on Variance Reduction in Dynamic Simulation Experiments / Barry L. Nelson. *Communications in Statistics - Simulation and Computation* 16, no. 2 (1987): 385-426
- Neumann 1999. Scheduling of projects with stochastic evolution structure / Klaus Neumann. in *Project Scheduling* / Jan Węglarz (ed.), Kluwer, Boston, Massachusetts, 1999. 309-332
- Neumann et al. 2000. Active and stable project scheduling / K. Neumann; H. Nübel; C. Schwindt. *Mathematical Methods of Operations Research* 52, no. 3 (2000): 441-465
- Neumann et al. 2002. *Project Scheduling with Time Windows and Scarce Resources: Temporal and Resource-Constrained Project Scheduling with Regular and Nonregular Objective Functions* / Klaus Neumann; Christoph Schwindt; Jürgen Zimmermann. Springer, Berlin, Deutschland, 2002. 2ª edición en 2003
- Neumann; Zimmermann 2000a. Procedures for resource leveling and net present value problems in project scheduling with general temporal and resource constraints / Klaus Neumann; Jürgen Zimmermann. *European Journal of Operational Research* 127 (2000): 425-443
- Neumann; Zimmermann 2000b. Branch-and-Bound and Truncated Branch-and-Bound Procedures for Resource-Constrained Project Scheduling with Discounted Cash Flows and General Temporal Constraints / Klaus Neumann; Jürgen Zimmermann. Universität Karlsruhe, Institut Für Wirtschaftstheorie Und Operations Research, Deutschland, Technical Report WIOR-581, 2000
- Newbold 1998. *Project Management in the Fast Lane: Applying the Theory of Constraints* / Robert C. Newbold. St. Lucie (pie de imprenta de CRC), New York, New York (o Boca Raton, Florida), 1998. 2ª edición Artech House, Norwood, Massachusetts, USA, 2004
- Noelle et al. 2002. Requirements for software-support in concurrent engineering teams / Tanja Noelle; Dirk Kabel; Holger Luczak. *Behaviour & Information Technology* 21, no. 5 (2002): 345-350
- Norris et al. 1992. *Project Risk Analysis and Management: Mini PRAM* / Catriona Norris; John Perry; Peter Simon. APM (Association for Project Management), High Wycombe, England, United Kingdom, 1992

- Nübel; Schwindt 1997. A Classification of Shifts, Schedules, and Objective Functions in Project Scheduling / Hartwig Nübel; Christoph Schwindt. Universität Karlsruhe, Deutschland, Technical Report WIOR-509, 1997
- Ordieres 1999. *Programación de proyectos* / Joaquín Ordieres. Universidad de la Rioja, Logroño, 1999
- Özdamar 1999. A genetic algorithm approach to a general category project scheduling problem / Linet Özdamar. *IEEE Transactions on Systems, Man, and Cybernetics* 29, no. 1 (1999): 44-59
- Özdamar; Alanya 2001. Uncertainty Modelling in Software Development Projects (With Case Study) / Linet Özdamar; Ebru Alanya. *Annals of Operations Research* 102 (2001): 157-178
- Pan et al. 2001. Resource-constrained Project Scheduling with Fuzziness / Hongqi Pan; Robert J. Willis; Chung-Hsing Yeh. in *Advances in Fuzzy Systems and Evolutionary Computation* / Mastorakis (ed.), WSES Press, Danvers, 2001. 173-179
- Pan; Yeh 2003. A Metaheuristic Approach to Fuzzy Scheduling / Hongqi Pan; Chung-Hsing Yeh. in *Knowledge-Based Intelligent Information and Engineering Systems: KES 2003 Conference*, University of Oxford, 2003. 2773: 1081-1087
- Pareto 1896. *Cours d'Économie politique* / Vilfredo Pareto. Rouge, Lausanne, Suisse, 1896. vol. 1
- Pareto 1897. *Cours d'Économie politique* / Vilfredo Pareto. Rouge, Lausanne, Suisse, 1897. vol. 2
- Paul; Gutierrez 2005. Simple probability models for project contracting / Anand Paul; Genaro Gutierrez. *European Journal of Operational Research* 165, no. 2 (2005): 329-338
- Pender 2001. Managing incomplete knowledge: Why risk management is not sufficient / Steven Pender. *International Journal of Project Management* 19, no. 2 (2001): 79-87
- Peterson 1977. Petri Nets / James L. Peterson. *ACM Computing Surveys* 9, no. 3 (1977): 223-252
- Petri 1962. *Kommunikation mit Automaten* / Carl Adam Petri. Institut für Instrumentelle Mathematik, Bonn, Deutschland, Ph. D. Thesis, 1962. Traducción al inglés en [Petri 1966]
- Petri 1966. *Communication with Automata* / Carl Adam Petri. Griffiss Air Force Base, Rome, New York, RADC-TR-65-377, 1966
- Phoon et al. 2004. Simulation of non-Gaussian processes using fractile correlation / Kok-Kwang Phoon; Ser-Tong Quek; Hongwei Huang. *Probabilistic Engineering Mechanics* 19, no. 4 (2004): 287-292
- PMBOK 1996. *A Guide to the Project Management Body of Knowledge* / William R. Duncan (ed.). Project Management Institute, Newtown Square, Pennsylvania, USA, 1996. 3ª edición en 2004.
- Pollack-Johnson; Liberatore 1998. Project Management Software Usage Patterns and Suggested Research Directions for Future Developments / Bruce Pollack-Johnson; Matthew J. Liberatore. *Project Management Journal* 28, no. 2 (1998): 19-28
- Pollack-Johnson; Liberatore 2005. Project Planning Under Uncertainty Using Scenario

- Analysis / Bruce Pollack-Johnson; Matthew J. Liberatore. *Project Management Journal* 36, no. 1 (2005): 15-26
- Pontrandolfo 2000. Project duration in stochastic networks by the PERT-path technique / Pierpaolo Pontrandolfo. *International Journal of Project Management* 18, no. 3 (2000): 215-222
- Prade 1979. Using Fuzzy Set Theory in a Scheduling Problem: A Case Study / Henri Prade. *Fuzzy Sets and Systems* 2, no. 2 (1979): 153-165
- Pritchard 2000. Advanced risk: How big is your crystal ball? / Carl Leonard Pritchard. in Proceedings of the Project Management Institute Annual Seminars & Symposium 2000, 2000
- Pritsker 1966. GERT: Graphical Evaluation and Review Rechnique / A. A. B. Pritsker. Rand Corporation, Santa Monica, California & National Aeronautics and Space Administration, RM-4973-NASA, 1966
- Pritsker; Happ 1966. GERT: Graphical Evaluation and Review Technique Part I. Fundamentals / A. Alan B. Pritsker; W. William Happ. *Journal of Industrial Engineering* 17, no. 5 (1966): 267-274
- Pritsker; Whitehouse 1966. GERT: Graphical Evaluation and Review Technique Part II. Probabilistic and Industrial Engineering Applications / A. Alan B. Pritsker; Gary E. Whitehouse. *Journal of Industrial Engineering* 17, no. 6 (1966): 293-301
- Prosser 1993. Hybrid algorithms for the constraint satisfaction problem / P. Prosser. *Computational Intelligence* 9, no. 3 (1993): 268-299
- Ramachandani 1974. Analysis of asynchronous concurrence systems by timed Petri Nets / C. Ramachandani. Massachusetts Institute of Technology, Cambridge, Massachusetts, Ph. D. Dissertation, 1974
- Ramachandra 2006. Optimal Dynamic Resource Allocation in Activity Networks / Girish A. Ramachandra. North Carolina State University, North Carolina, Ph. D. Dissertation, 2006
- RAMP 1998. *Risk Analysis and Management for Projects: RAMP* / Institution of Civil Engineers. Thomas Telford, London, England, United Kingdom, 1998. 2ª edición en 2005
- Rasdorf; Abydayyeh 1991. Cost and schedule control integration: Issues and needs / William J. Rasdorf; Osama Y. Abydayyeh. *Journal of Construction Engineering and Management* 117, no. 3 (1991): 486-502
- Raz; Barnes; Dvir 2003. A Critical Look at Critical Chain Project Management / Tzvi Raz; Robert Barnes; Dov Dvir. *Project Management Journal* 34, no. 4 (2003): 24-32
- Raz; Barnes; Dvir 2004. A Critical Look at Critical Chain Project Management / Tzvi Raz; Robert Barnes; Dov Dvir. *IEEE Engineering Management Review* 32, no. 2 (2004): 35-44
- Raz; Marshall 1996. Effect of resource constraints on float calculations in project networks / Tavi Raz; Bob Marshall. *International Journal of Project Management* 14, no. 4 (1996): 241-248
- Raz; Michael 2001. Use and benefits of tools for project risk management / T. Raz; E. Michael. *International Journal of Project Management* 19, no. 1 (2001): 9-17
- Rosenblatt 1958. The Perceptron: A Probabilistic Model for Information Storage and

- Organization in the Brain / Frank Rosenblatt. *Psychological Review* 65 (1958): 386-408
- Roy 1962. Graphes et ordonnancements / Bernard Roy. *Revue Française de Recherche Opérationnelle* 25 (1962): 323-326
- Roy 1964. *Les problèmes d'ordonnement* / Bernard Roy. Dunod, Paris, France, 1964
- Rubinstein 1981. *Simulation and the Monte Carlo Method* / Reuven Y. Rubinstein. John Wiley & Sons, New York, 1981
- Russell 1970. Cash Flows in Networks / A. H. Russell. *Management Science* 16 (1970): 357-373
- Sahni; Horowitz 1978. Combinatorial Problems: Reducibility and Approximation / Sartaj Sahni; Ellis Horowitz. *Operations Research* 26, no. 5 (1978): 718-759
- Sanlaville 2002. Flexibilité et Robustesse en Ordonnement / Eric Sanlaville. *Bulletin de la Société Française de Recherche Opérationnelle et d'Aide à la Décision* 8 (2002): 10-12
- Savin et al. 1996. Construction resource leveling using neural networks / Daniela Savin; Sabah Alkass; Paul Fazio. *Canadian Journal of Civil Engineering* 23, no. 4 (1996): 917-925
- Savin et al. 1998. Calculating weight matrix of neural network for resource leveling / Daniela Savin; Sabah Alkass; Paul Fazio. *Journal of Computing in Civil Engineering* 12, no. 4 (1998): 241-248
- Sawhney 1997. Petri net based simulation of construction schedules / Anil Sawhney. in *Proceedings of the 1997 Winter Simulation Conference* / S. Andradóttir; K. J. Healy; D. H. Withers; B. L. Nelson, The Society for Computer Simulation International, San Diego, California, 1997. 1111-1118
- Shafer 1976. *Mathematical Theory of Evidence* / Glenn A. Shafer. Princeton University Press, Princeton, New Jersey, 1976
- Schuyler 2000. Project Risk Management by the Numbers / John Schuyler. *PM Network* 14, no. 9 (2000)
- Schwindt 1998. A Branch-and-Bound Algorithm for the Resource-Constrained Project Duration Problem Subject to Temporal Constraints / Christoph Schwindt. Universität Karlsruhe, Deutschland, Technical Report WIOR-544, 1998
- Schwindt; Zimmermann 2001. A steepest ascent approach to maximizing the net present value of projects / Christoph Schwindt; Jürgen Zimmermann. *Mathematical Methods of Operations Research* 53, no. 3 (2001): 435-450
- Sefair; Medaglia 2005. Towards a model for selection and scheduling of risky project / Jorge A. Sefair; Andrés L. Medaglia. in *2005 IEEE Systems and Information Engineering Design Symposium*, 2005. 158-164
- Selvidge 1980. Assessing the extremes of probability distributions by the fractile method / J. E. Selvidge. *Decision Sciences* 11, no. 3 (1980): 493-502
- Senouci; Adeli 2001. Resource scheduling using neural dynamics model of Adeli and Park / Ahmed B. Senouci; Hojjat Adeli. *Journal of Construction Engineering and Management* 127, no. 1 (2001): 28-34
- Senouci; Eldin 2004. Use of genetic algorithms in resource scheduling of construction

- projects / Ahmed B. Senouci; Neil. N. Eldin. *Journal of Construction Engineering and Management* 130, no. 6 (2004): 869-877
- Shannon 1948. A mathematical theory of communication / Claude E. Shannon. *Bell System Technical Journal* 27, no. 3 (1948): 379-423 y 623-656
- Shen et al. 2005. A Petri-Net Based Modeling Approach to Concurrent Software Engineering Tasks / Jau-Ji Shen; S. Wesley Changchien; Tao-Yuan Lin. *Journal of Information Science and Engineering* 21, no. 4 (2005): 767-795
- Shih 2005. Estimating completion-time distribution in stochastic activity networks / N.-H. Shih. *Journal of the Operational Research Society* 56, no. 5 (2005): 744-749
- Simon et al. 1997. *Project Risk Analysis and Management Guide: PRAM Guide* / P. Simon; D. Hillson; K. Newland (eds.). APM Publishing, High Wycombe, England, United Kingdom, 1997. 2ª edición en 2004
- Słowiński; Hapke 2000. *Scheduling Under Fuzziness* / Roman Słowiński; Maciej Hapke (eds.). Springer, Berlin, Deutschland, 1999
- Slyeptsov; Tyshchuk 2003. Fuzzy temporal characteristics of operations for project management on the network models basis / Anatoliy I. Slyeptsov; Tetyana A. Tyshchuk. *European Journal of Operational Research* 147, no. 2 (2003): 253-265
- Smith; Eppinger 1997a. A Predictive Model of Sequential Iteration in Engineering Design / Robert P. Smith; Steven D. Eppinger. *Management Science* 43, no. 8 (1997): 1104-1120
- Sobol' 1994. *A Primer for the Monte Carlo Method* / Ilya Meerovich Sobol'. CRC-Press, Boca Raton, Florida, 1994
- Son; Skibniewski 1999. Multiheuristic Approach for Resource Leveling Problem in Construction Engineering: Hybrid Approach / Jaeho Son; Miroslaw J. Skibniewski. *Journal of Construction Engineering and Management* 125, no. 1 (1999): 23-31
- Song et al. 2002. Applying neural network to move operations in exchange heuristic in solving resource constrained scheduling / Inkap R. Song; Taeyong Yang; Jaihyun Kim. *International Journal of Industrial Engineering* 9, no. 2 (2002): 141-150
- Sprecher; Drexl 1998. Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm / Arno Sprecher; Andreas Drexl. *European Journal of Operational Research* 107, no. 2 (1998): 431-450
- Sprecher et al. 1995. Semi-active, active and non-delay schedules for the resource-constrained project scheduling problem / Arno Sprecher; Rainer Kolisch; Andreas Drexl. *European Journal of Operational Research* 80, no. 1 (1995): 94-102
- Sprecher et al. 1997. An exact algorithm for project scheduling with multiple modes / Arno Sprecher; Sönke Hartmann; Andreas Drexl. *OR Spectrum* 19, no. 3 (1997): 195-203
- Staines 2006. Using a Timed Petri Net (TPN) to Model a Bank ATM / Tony Spiteri Staines. in Proceedings of the 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems, 2006. 151-158
- Starr 1990. The role of project management in a fast response organization / Martin K. Starr. *Journal of Engineering and Technology Management* 7, no. 2 (1990): 89-110
- Steward 1981. The Design Structure System: A Method for Managing the Design of Complex

- Systems / Donald V. Steward. *IEEE Transactions on Engineering Management* 28 (1981): 71-74
- Stinson et al. 1978. Multiple Resource-Constrained Scheduling using Branch-and-Bound / Joel P. Stinson; Edward W. Davis; Basheer Khumawala. *AIIE Transactions* 10, no. 3 (1978): 252-259
- Stockmeyer; Chandra 1979. Intrinsically difficult problems / Larry J. Stockmeyer; Ashok K. Chandra. *Scientific American* 240, no. 5 (1979): 124-133. Vuelto a publicar en [Stockmeyer; Chandra 1988]
- Stockmeyer; Chandra 1988. Intrinsically Difficult Problems / Larry J. Stockmeyer; Ashok K. Chandra. in *Trends in Computing*, Scientific American, New York, New York, 1988. 88-97
- Stork 2000. Branch-and-bound algorithms for stochastic resource-constrained project scheduling / Frederik Stork. Technische Universität Berlin, Germany, 702/2000, 2000
- Stork 2001. Stochastic Resource-Constrained Project Scheduling / Frederik Stork. Technische Universität Berlin, Fakultät II - Mathematik und Naturwissenschaften, Germany, Ph. D. Thesis, 2001
- Syan; Menon 1994. *Concurrent Engineering* / Chanan S. Syan; Unny Menon. Springer, Berlin, Deutschland, 1994
- Talbot 1982. Resource-constrained project scheduling with time-resource tradeoffs: The nonpreemptive case / F. B. Talbot. *Management Science* 28 (1982): 1197-1210
- Tavakoli; Riachi 1990. CPM Use in ENR Top 400 Contractors / A. Tavakoli; R. Riachi. *Journal of Management in Engineering* 6, no. 3 (1990): 282-295
- Tavares 1998. *Advanced models for project management* / L. Valadares Tavares. Springer, Springer, 1998
- Tavares 2002. A review of the contribution of operational research to project management / L. Valadares Tavares. *European Journal of Operational Research* 136, no. 1 (2002): 1-18
- Tavares et al. 1998. On the Optimal Management of Project Risk / L. Valadares Tavares; J. A. Antunes Ferreira; J. Silva Coelho. *European Journal of Operational Research* 107, no. 2 (1998): 451-469
- Taylor 1911. *Principles of Scientific Management* / Frederick Winslow Taylor. Norton, New York, 1967 (republ.)
- Tereso et al. 2003. Experimental results of an adaptative resource allocations technique to stochastic multimodal projects / Anabela Pereira Tereso; M. Madalena T. Araújo; Salah E. Elmaghraby. in *Proceedings of the International Conference on Industrial Engineering and Production Management*, 2003.
- Tereso et al. 2004. Adaptative resource allocation in multimodal activity networks / Anabela P. Tereso; M. Madalena T. Araújo; Salah E. Elmaghraby. *International Journal of Production Economics* 92, no. 1 (2004): 1-10
- Thomasen; Butterfield 1993. Combining Risk Management and Resource Optimization in Project Management Software / O. B. Thomasen; L. Butterfield. *Cost Engineering* 35, no. 8 (1993): 19-24

- Tormos; Lova 1998. Gestión de proyectos informáticos: Riesgo e incertidumbre / Pilar Tormos; Antonio Lova. Universidad Politécnica de Valencia, Servicio de Publicaciones, SPUPV – 98.484, 1998
- Tormos; Lova 2001. A Competitive Heuristic Solution Technique for Resource-Constrained Project Scheduling / Pilar Tormos; Antonio Lova. *Annals of Operations Research* 102 (2001): 65-81
- Touran 1993. Probabilistic Cost Estimating with Subjective Correlations / Ali Touran. *Journal of Construction Engineering and Management* 119, no. 1 (1993): 58-71
- Tsai; Gemmill 1998. Using tabu search to schedule activities of stochastic resource-constrained projects / Ying-Wei Tsai; Douglas D. Gemmill. *European Journal of Operational Research* 111, no. 1 (1998): 129-141
- Turing 1936. On computable numbers, with an application to the Entscheidungs / Alan M. Turing. *Proceedings of the London Mathematical Society* 42, no. 2 (1936): 230-265
- Turner 1993. *The Handbook of Project-based Management: Improving the Process for Achieving Strategic Objectives* / J. Rodney Turner. McGraw-Hill, Maidenhead, England, United Kingdom, 1993. 2ª edición en 1998
- Turnquist; Nozick 2003. Allocating time and resources in project management under uncertainty / Mark A. Turnquist; Linda K. Nozick. in Proceedings of the 36th Hawaii International Conference on Systems Sciences, 2003
- Turnquist; Nozick 2004. A Nonlinear Optimization Model of Time and Resource Allocation Decisions in Projects with Uncertainty / Mark A. Turnquist; Linda K. Nozick. *Engineering Management Journal* 16, no. 1 (2004): 40-49
- Ulusoy; Özdamar 1989. Heuristic Performance and Network/Resource Characteristics in Resource-constrained Project Scheduling / Gündüz Ulusoy; Linet Özdamar. *Journal of the Operational Research Society* 40, no. 12 (1989): 1145-1152
- Valls et al. 1998. Project scheduling with stochastic activity interruptions / Vicente Valls; Manuel Laguna; Pilar Lino; Angeles Pérez; M. Sacramento Quintanilla. in Handbook of Recent Advances in Project Scheduling / Jan Węglarz (ed.), Kluwer, Amsterdam, Nederlanden, 1998. 333-353
- Valls et al. 2002. A hybrid genetic algorithm for the resource-constrained project scheduling problem with the peak crossover operator / Vicente Valls; Francisco Ballestín; Sacramento Quintanilla. in Eighth International Workshop on Project Management and Scheduling, València, 2002. 368-371
- Valls et al. 2005. Justification and RCPSP: A technique that pays / Vicente Valls; Francisco Ballestín; Sacramento Quintanilla. *European Journal of Operational Research* 165, no. 2 (2005): 375-359
- Van de Vonder et al. 2006. The trade-off between stability and makespan in resource-constrained project scheduling / Stijn Van de Vonder; Erik Demeulemeester; Willy Herroelen; Roel Leus. *International Journal of Production Research* 44, no. 2 (2006): 215-236
- van der Aalst 1998. The Application of Petri Nets to Workflow Management / Wil M. P. van

- der Aalst. *The Journal of Circuits, Systems and Computers* 8, no. 1 (1998): 21-66
- Vanhoucke 2006. An efficient hybrid search algorithm for various optimization problems / Mario Vanhoucke. Universiteit Gent, Faculteit Economie en Bedrijfskunde, België-Belgique, Working Paper 2006/365, 2006
- Vanhoucke et al. 2001. Maximizing the net present value of a project with linear time-dependent cash-flows / Mario Vanhoucke; Erik Demeulemeester; Willy Herroelen. *International Journal of Production Research* 39, no. 14 (2001): 3159-3181
- Viana; de Sousa 2000. Using metaheuristics in multiobjective resource constrained project scheduling / Ana Viana; Jorge Pinho de Sousa. *European Journal of Operational Research* 120, no. 2 (2000): 359-374
- von Neumann; Morgenstern 1944. *Theory of Games and Economic Behavior* / John von Neumann; Oskar Morgenstern. Princeton University Press, Princeton, New Jersey, 1944. 2ª edición en 1947
- Voropajev et al. 1999a. A Model for Supplying with Constrained Resources in Project Management under Random Disturbances / V. I. Voropajev; S. M. Ljubkin; D. Golenko-Ginzburg; A. Gonik. *Project Management* 5, no. 1 (1999): 68-73
- Voropajev et al. 1999b. Decision-making models for generalized alternative stochastic networks / V. I. Voropajev; S. M. Ljubkin; D. Golenko-Ginzburg. *Automatic and Remote Control* 60, no. 10 (1999): 1484-1490
- Voropajev et al. 2000. Structural classification of network models / V. I. Voropajev; S. M. Ljubkin; B. P. Titarenko; D. Golenko-Ginzburg. *International Journal of Project Management* 18, no. 5 (2000): 361-368
- Wall 1997. Distribution and correlations in Monte Carlo simulation / David M. Wall. *Construction Management and Economics* 15, no. 3 (1997): 241-258
- Wang 1999. A fuzzy set approach to activity scheduling for product development / J. R. Wang. *Journal of the Operational Research Society* 50, no. 12 (1999): 1217-1228
- Wang 2002. A fuzzy project scheduling approach to minimize schedule risk for product development / Juite Wang. *Fuzzy Sets and Systems* 127, no. 2 (2002): 99-116
- Wang 2004. A fuzzy robust scheduling approach for product development projects / Juite Wang. *European Journal of Operational Research* 152, no. 1 (2004): 180-194
- Wang 2005. Constraint-based schedule repair for product development projects with time-limited constraints / Juite Wang. *International Journal of Production Economics* 95, no. 3 (2005): 399-414
- Wang; Lu 2002. Genetic algorithm optimized resource activity critical path method / Peng-Yi Wang; Ming Lu. in Proceedings of 2002 International Conference on Machine Learning and Cybernetics, 2002. 1978-1982
- Wendling; Lorance 1999. Basic Techniques for Analyzing and Presenting Schedule Risk Analysis / Robert V. Wendling; Randal B. Lorance. *AACE International Transactions* (1999): risk.08.1-risk.08.8
- Wendling; Lorance 2000. Integration of Schedule and Cost Risk Models / Robert V. Wendling; Randal B. Lorance. *Cost Engineering* 42, no. 4 (2000): 38-43

- Wickwire et al. 2001. *Construction scheduling: Preparation, liability, and claims, 2001 cumulative supplement* / J. M. Wickwire; T. J. Driscoll; S. B. Hurlbut. Aspen Law & Business, New York, 2001
- Wiest 1964. Some Properties of Schedules for Large Projects with Limited Resources / J. D. Wiest. *Operations Research* 12 (1964): 395-418
- Williams 1937. The Theory of Investment Value / John Burr Williams. Harvard University, Ph. D. Thesis, 1937. Publicada por Harvard University Press, 1938, y reimpressa Fraser, Flint Hill, Virginia, 1997
- Williams 1992a. Criticality in stochastic networks / Terry M. Williams. *Journal of the Operational Research Society* 43 (1992): 353-357
- Williams 1992b. Practical Use of Distributions in Network Analysis / Terry M. Williams. *Journal of the Operational Research Society* 43, no. 3 (1992): 265-270
- Williams 1993. Risk-management infrastructures / Terry M. Williams. *International Journal of Project Management* 11, no. 1 (1993): 5-10
- Williams 1997. Empowerment vs risk management? / Terry M. Williams. *International Journal of Project Management* 15, no. 4 (1997): 219-222
- Williams 1999a. The need for new paradigms for complex projects / Terry M. Williams. *International Journal of Project Management* 17, no. 5 (1999): 269-273
- Williams 1999b. Seeking optimum project duration extensions / Terry M. Williams. *Journal of the Operational Research Society* 50, no. 5 (1999): 460-467
- Williams 2000. Systematic project risk management / Terry M. Williams. *International Journal of Risk Assessment and Management* 1, no. 1 (2000): 149-159
- Williams 2003. The contribution of mathematical modelling to the practice of project management / Terry Williams. *IMA Journal of Management Mathematics* 14, no. 1 (2003): 3-30
- Wilson 1984. Variance reduction techniques for digital simulation / James R. Wilson. *American Journal of Mathematical and Management Sciences* 4, no. 3-4 (1984): 277-312
- Whitehouse; Pritsker 1969. GERT: Part III - Further Statistical Results; Counters, Renewal Times and Correlations / Gary E. Whitehouse; A. A. B. Pritsker. *AIIE Transactions* 1, no. 1 (1969): 45-50
- Wolsey 1998. Integer Programming / Laurence Wolsey. Wiley, New York, New York, 1998
- Woodworth; Shanahan 1988. Identifying the critical sequence in a resource-constrained project / B. M. Woodworth; S. Shanahan. *International Journal of Project Management* 6 (1988): 89-96
- Yates 1992. Computerized Inquiry-Feedback Knowledge Engineering System / J. K. Yates. *AACE Transactions* 2 (1992): R1.1-R.1.7
- Yassine; Braha 2003. Complex Concurrent Engineering and the Design Structure Matrix Method / Ali Yassine; Dan Braha. *Concurrent Engineering* 11, no. 3 (2003): 165-176
- Yassine et al. 1999. A Decision Analytic Framework for Evaluating Concurrent Engineering / Ali A. Yassine; Kenneth R. Chelst; Donald R. Falkenburg. *IEEE Transactions on Engineering Management* 46, no. 2 (1999): 144-157

- Yi et al. 2002. Network Creation and Development for Repetitive-Unit Projects / Kyoo Jin Yi; Hyun-Soo Lee; Yoon Ki Choi. *Journal of Construction Engineering and Management* 128, no. 3 (2002): 257-264
- Yunus et al. 1990. Development of a Knowledge-Based Schedule Planning System / N. B. Yunus; D. L. Babcock; C. O. Benjamin. *Project Management Journal* (1990)
- Zadeh 1965. Fuzzy sets / Lotfi A. Zadeh. *Information Control* 8 (1965): 338-353
- Zafra-Cabeza et al. 2004a. An algorithm for optimal scheduling and risk assessment of projects / Ascensión Zafra-Cabeza; Miguel A. Ridao; Eduardo F. Camacho. *Control Engineering Practice* 12, no. 10 (2004): 1329-1338
- Zafra-Cabeza et al. 2004b. Chance constrained project scheduling under risk / Ascensión Zafra-Cabeza; Miguel A. Ridao; Eduardo F. Camacho. in 2004 IEEE International Conference on Systems, Man and Cybernetics, vol. 2, 2004. 1789-1794
- Zafra-Cabeza et al. 2005. A stochastic predictive control approach to project risk management / Ascensión Zafra-Cabeza; Miguel A. Ridao; Eduardo F. Camacho. in Proceedings of 16th IFAC World Congress, Elsevier, Amsterdam, Nederlanden, 2005
- Zhang et al. 2006. Particle swarm optimization for resource-constrained project scheduling / Hong Zhang; Heng Li; C. M. Tam. *International Journal of Project Management* 24, no. 1 (2006): 83-92
- Zheng et al. 2002. Applying genetic algorithm techniques for time-cost optimization / Daisy X. M. Zheng; S. Thomas Ng; Mohan M. Kumaraswamy. in Proceedings of the 18th Annual Conference of ARCOM / D. Greenwood (ed.), 2002. 801-810
- Zheng et al. 2003. GA-Based Multiobjective Technique for Multi-Resource Leveling / Daisy X. M. Zheng; S. Thomas Ng; Mohan M. Kumaraswamy. in Construction Research Congress, Winds of Change, 2003. 237-244
- Zhuang; Yassine 2004. Task scheduling of parallel development projects using genetic algorithms / Miao Zhuang; Ali A. Yassine. in Proceedings of the ASME Design Engineering Technical Conference and Computers and Information in Engineering Conference 2004, ASME, New York, 2004. 215-223