Universidad Complutense Madrid

MODELOS OPERATIVOS DE GESTIÓN

Begoña Vitoriano

Septiembre 2009

ÍNDICE

1 Modelos de programación matemática para gestión

1.1 Modelos característicos de programación matemática para gestión

A continuación se presentan algunos problemas característicos de programación lineal y entera. Éstos se utilizan como referencia y clasificación para otros problemas. En particular, para los problemas enteros existen numerosas referencias de investigación dedicadas a la solución de los mismos.

1.1.1 Problema de la dieta

El problema por excelencia de programación lineal es el de asignación óptima de recursos. Un caso particular de éste es el denominado problema de la dieta. Consiste en determinar la composición de la dieta de mínimo coste que satisface las necesidades específicas de nutrientes. Pongamos un caso particular muy sencillo de alimentación de ganado bovino.

Aprovechamos este ejemplo para seguir paso a paso las etapas en el desarrollo de un modelo.

- En primer lugar hay que ***identificar el problema***.

  Se ha determinado que las necesidades mínimas diarias en la alimentación de una ternera son de 700 g de proteínas, 28 g de calcio y 150 mg de vitaminas. Los alimentos disponibles son pienso y forraje con un coste unitario de 0.30 y 0.35 €/kg respectivamente. La composición nutritiva por kg de alimento se muestra en la siguiente tabla.

	Proteínas (g)	Calcio (g)	Vitaminas (mg)
Pienso	30	2	10
Forraje	45	1	5

  Se trata de determinar la cantidad diaria óptima de cada alimento para minimizar el coste total de alimentación.

- A continuación ***se especifica matemáticamente y se formula*** el problema.

Para ello analizamos y organizamos los **datos** del problema. Sean los alimentos disponibles (pienso y forraje) y sean j los nutrientes (proteínas, calcio y vitaminas). Sea b_j la cantidad mínima diaria requerida de cada nutriente. Sea a_{ij} la cantidad de nutriente por kg de alimento correspondiente a los valores de la tabla dada. Sea c_i el coste unitario de cada alimento. A continuación definimos las **variables**. Sea x_i la cantidad diaria en kg de cada alimento . Además indicamos la **función objetivo** y las **restricciones** del problema. La función objetivo es la minimización del coste diario de la dieta

$$\min_{x_i} \sum_i c_i x_i \qquad (1.1)$$

Las restricciones corresponden a satisfacer con la mezcla de alimentos las necesidades mínimas diarias de cada nutriente y, por consiguiente, habrá tantas restricciones de este tipo como nutrientes.

$$\sum_i a_{ij} x_i \geq b_j \quad \forall j \qquad (1.2)$$

Además hay que añadir la restricción natural de que la cantidad de cada alimento ha de ser no negativa.

$$x_i \geq 0 \qquad (1.3)$$

Particularizando estas ecuaciones para los datos previos se obtiene.

$$\begin{aligned}
&\min_{x_1, x_2} 0.30x_1 + 0.35x_2 \\
&30x_1 + 45x_2 \geq 700 \\
&2x_1 + x_2 \geq 28 \\
&10x_1 + 5x_2 \geq 150 \\
&x_1 \geq 0 \\
&x_2 \geq 0
\end{aligned}$$

- Después viene la ***resolución***.

  Vamos a resolver gráficamente el problema. Para ello se dibujan las ecuaciones en forma de igualdad en el espacio de las variables y se indica la región factible del problema. Es decir, el conjunto de puntos que cumple todas las restricciones. Se traza la recta de la función objetivo para un valor cualquiera y se desplaza paralela a sí misma en el sentido de minimizar dicho valor hasta el último punto de la región factible. Dicho punto será el óptimo del problema.

- Las etapas de ***verificación*** (comprobación de que el modelo es correcto) y ***validación*** (comprobación de que la realidad se representa adecuadamente) son inmediatas en un modelo tan sencillo como éste.

- Seguidamente se realiza la ***interpretación*** y ***análisis*** de los resultados.

  Los resultados indican que la decisión óptima es comprar 10.833 kg de pienso y 8.333 kg de forraje cada día. Con estas decisiones el coste diario de los alimentos es de 6.1667 €.

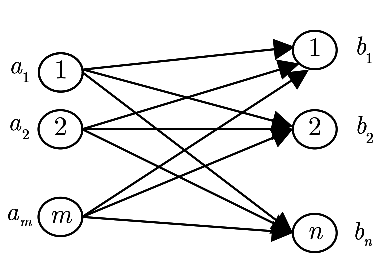
  Al ganadero le ha llegado una oferta de otro fabricante de piensos a un precio de 0.25 €/kg pero con menor contenido en calcio, 1.5 g de calcio por kg de pienso, y tiene interés en analizar si le interesa comprar o no a dicho fabricante. Para ello planteamos este nuevo problema de optimización

  La solución óptima para este nuevo problema es comprar 14.933 kg de pienso y 5.6 kg de forraje diariamente con un coste de 5.6933 €. Luego, esta oferta es atractiva económicamente.

- La etapa de ***implantación, documentación y mantenimiento*** se da por satisfecha en este modelo sencillo con este apartado donde se explica el modelo.

1.1.2 Problema de transporte

Se trata de minimizar el coste total de transporte de un cierto producto desde los diferentes orígenes a los destinos, satisfaciendo la demanda de cada destino sin superar la oferta disponible en cada origen (ver ejemplo en 1.2.3.1 Ejemplo de transporte). Se supone que todos los m orígenes están conectados con los todos los n destinos. Sea a_i la oferta de producto en el origen i, b_j la demanda de producto en el destino j y c_{ij} el coste unitario de transporte desde el origen i al destino j.



El problema de optimización consiste en determinar las unidades de producto $x_{ij} \geq 0$ transportadas desde i hasta j, $\forall i, j$, que minimizan los costes de transporte sujeto a las restricciones de oferta disponible en cada origen i (m restricciones de oferta) y demanda en cada destino j (n restricciones de demanda)

$$
\begin{aligned}
&\min_{x_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
&\sum_{j=1}^{n} x_{ij} = a_i \quad i = 1, \ldots, m \\
&\sum_{i=1}^{m} x_{ij} = b_j \quad j = 1, \ldots, n \\
&x_{ij} \geq 0
\end{aligned}
\tag{1.4}
$$

Implícitamente en esta formulación, se supone que la oferta del producto es igual a la demanda del mismo $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$. Si $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$ se añade un sumidero universal con coste nulo. Si $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$ se añade una fuente universal conectada con todos los destinos con coste muy elevado.

La estructura que presenta la matriz de restricciones del problema tiene el siguiente aspecto.

$$
\begin{array}{c|cccc|cccc|c|cccc|}
 & x_{11} & x_{12} & \cdots & x_{1n} & x_{21} & x_{22} & \cdots & x_{2n} & \cdots & x_{m1} & x_{m2} & \cdots & x_{mn} \\
1 & 1 & 1 & \cdots & 1 & & & & & & & & & \\
2 & & & & & 1 & 1 & \cdots & 1 & & & & & \\
\vdots & & & & & & & & & \ddots & & & & \\
m & & & & & & & & & & 1 & 1 & \cdots & 1 \\
\hline
1 & 1 & & & & 1 & & & & \cdots & 1 & & & \\
2 & & 1 & & & & 1 & & & \cdots & & 1 & & \\
\vdots & & & \ddots & & & & \ddots & & \cdots & & & \ddots & \\
n & & & & 1 & & & & 1 & \cdots & & & & 1
\end{array}
$$

Si tanto las ofertas a_i como las demandas de los productos b_j son números enteros, entonces el valor óptimo de x_{ij} es entero por ser la matriz totalmente unimodular[1], por lo que no se necesita recurrir a métodos específicos de resolución de problemas de programación entera.

1.1.3 Problema de transbordo

Consiste en determinar en una red con n nodos las cantidades óptimas para llevar unidades de un producto desde sus orígenes a sus destinos pasando por puntos de transbordo intermedios.

Cada *origen* genera $b_i > 0$ unidades, cada *destino* consume $b_i < 0$ unidades y cada *transbordo* ni genera ni consume unidades $b_i = 0$. El coste unitario de transporte desde el origen i hasta el destino j en dicho sentido es c_{ij}.

Hay que determinar las unidades de producto transportadas desde i a j, $x_{ij} \geq 0$, $\forall i, j$, que minimizan los costes de transporte teniendo en cuenta la restricción de balance o conservación del flujo en cada nudo i.

$$\begin{aligned} &\min_{x_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \\ &\sum_{j=1}^{n} x_{ij} - \sum_{k=1}^{n} x_{ki} = b_i \quad i = 1, \ldots, n \\ &x_{ij} \geq 0 \end{aligned} \tag{1.5}$$

Implícitamente en esta formulación, se supone que la oferta es igual a la demanda del producto, es decir, $\sum_{i=1}^{n} b_i = 0$.

Esta matriz también es totalmente unimodular por lo que el problema también puede ser resuelto mediante programación lineal.

[1] Una matriz es *totalmente unimodular* si toda submatriz cuadrada tiene determinante 0, 1 ó –1. Si la matriz de un problema lineal es totalmente unimodular y las cotas de las restricciones son enteras, entonces todos los puntos extremos del poliedro tienen coordenadas enteras (se denomina *politopo entero*).

1.1.4 Problema de asignación

Se trata de asignar la realización de n tareas a n personas (máquinas, etc.). Este problema es un caso particular del problema de transporte. Por consiguiente las variables toman valores enteros sin exigir esta condición en la formulación del problema.

Consiste en minimizar el coste total de realizar las tareas sabiendo que cada tarea i debe ser hecha por una sola persona y cada persona j debe realizar una única tarea, siendo c_{ij} el coste de realizar la tarea i por la persona j. Las variables del problema son $x_{ij} = \begin{cases} 1 & \text{si se asigna la tarea } i \text{ a la persona } j \\ 0 & \text{en cualquier otro caso} \end{cases}, \ \forall i, j$.

$$
\begin{aligned}
&\min_{x_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \\
&\sum_{j=1}^{n} x_{ij} = 1 \quad i = 1, \ldots, n \\
&\sum_{i=1}^{n} x_{ij} = 1 \quad j = 1, \ldots, n \\
&x_{ij} \geq 0
\end{aligned}
\tag{1.6}
$$

1.1.5 Problema de la mochila (*knapsack*)

Se trata de maximizar el valor total de la elección de un conjunto de n proyectos sin sobrepasar el presupuesto b disponible, siendo v_j y c_j el valor y coste de cada proyecto j respectivamente. El nombre procede de la decisión que toma un montañero que trata de maximizar el valor de lo que introduce en su mochila con una restricción de máximo peso admisible. Las variables del problema son $x_j = \begin{cases} 1 & \text{si se elije el proyecto } j \\ 0 & \text{en cualquier otro caso} \end{cases}$. Ésta es una utilización habitual de las variables binarias como forma de seleccionar una alternativa, un proyecto en este caso. La formulación del problema es la siguiente

$$\begin{aligned}
&\max_{x_j} \sum_{j=1}^{n} v_j x_j \\
&\sum_{j=1}^{n} c_j x_j \le b \qquad (1.7)\\
&x_j \in \{0,1\}
\end{aligned}$$

1.1.6 Problema de recubrimiento (*set covering*)

Existen m características y n combinaciones (subconjuntos) de dichas características. La elección de una combinación implica realizar todas las características de la misma. Se trata de minimizar el coste total de las combinaciones elegidas de manera que se cubra o posea cada característica i *al menos* una vez. Los datos son c_j el coste de elegir la combinación j y la matriz de pertenencia de cada característica i a cada combinación j, $a_{ij} = \begin{cases} 1 & \text{si } i \text{ pertenece a } j \\ 0 & \text{si no pertenece} \end{cases}$. Denominamos las variables $x_j = \begin{cases} 1 & \text{si se elige la combinación } j \\ 0 & \text{en cualquier otro caso} \end{cases}$. El problema se formula de la siguiente manera

$$\begin{aligned}
&\min_{x_j} \sum_{j=1}^{n} c_j x_j \\
&\sum_{j=1}^{n} a_{ij} x_j \ge 1 \quad i = 1, \ldots, m \qquad (1.8)\\
&x_j \in \{0,1\}
\end{aligned}$$

En la siguiente figura se representa gráficamente el problema de recubrimiento así como los de empaquetado y partición que se explican a continuación.

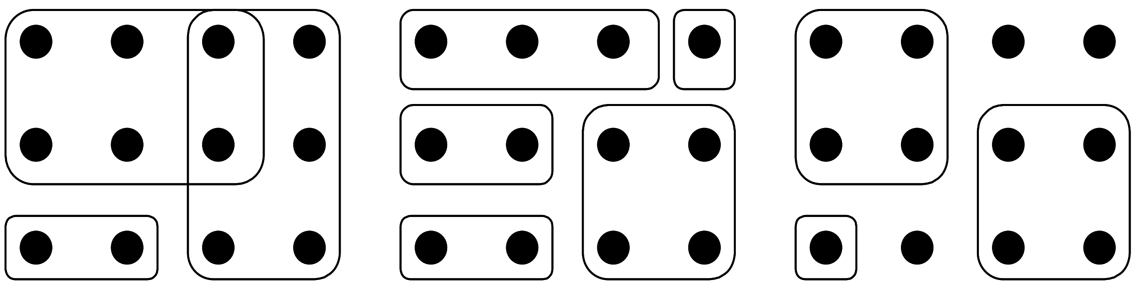


Figura 1.1 Representación gráfica de un recubrimiento, una partición y un empaquetado, respectivamente.

Veamos a continuación un ejemplo de recubrimiento: *asignación de tripulaciones*, tomado de [Hillier y Lieberman, 2002]. Una compañía aérea necesita asignar sus tripulaciones para cubrir todos sus vuelos. En particular, quiere resolver el problema de asignar tres tripulaciones con base en San Francisco a los vuelos listados en la primera columna de la tabla. Las otras columnas muestran las 12 secuencias factibles de vuelos para una tripulación cualesquiera. Los números de cada columna indican el orden de los vuelos. Se necesita elegir tres secuencias (una por tripulación) de manera que se cubran todos los vuelos. Se permite tener más de una tripulación en un vuelo, donde la/s tripulación/es extra viajan como pasajeros, pero por convenio laboral la tripulación extra cobra como si estuviera trabajando. El coste de asignación de una tripulación a cada secuencia de vuelos se da en millones de euros en la última fila. El objetivo es minimizar el coste total de asignación de las tres tripulaciones para cubrir todos los vuelos. Resolver el mismo problema para el caso en que no se permite el vuelo de una tripulación fuera de servicio en un vuelo.

	Secuencias factibles											
	1	2	3	4	5	6	7	8	9	10	11	12
SF – LA	1			1			1			1		
SF – Denver		1			1			1			1	
SF – Seattle			1			1			1			1
LA – Chicago				2			2		3	2		3
LA – SF	2					3				5	5	
Chicago – Denver				3	3				4			
Chicago – Seattle							3	3		3	3	4
Denver – SF		2		4	4				5			
Denver – Chicago					2			2			2	
Seattle – SF			2				4	4				5
Seattle – LA						2			2	4	4	2

Coste (M€)	2	3	4	6	7	5	7	8	9	9	8	9

Se definen las variables del problema como $x_j = \begin{cases} 1 & \text{si se asigna la secuencia } j \\ 0 & \text{en cualquier otro caso} \end{cases}$, $x_j \in \{0,1\} \quad j = 1,\ldots,12$

La función objetivo será

$$\min 2x_1 + 3x_2 + 4x_3 + 6x_4 + 7x_5 + 5x_6 + 7x_7 + 8x_8 + 9x_9 + 9x_{10} + 8x_{11} + 9x_{12}$$

Cobertura de cada vuelo al menos una vez

$$\begin{gathered} x_1 + x_4 + x_7 + x_{10} \geq 1 \ \text{(SF-LA)} \\ x_2 + x_5 + x_8 + x_{11} \geq 1 \ \text{(SF-Denver)} \\ x_3 + x_6 + x_9 + x_{12} \geq 1 \ \text{(SF-Seatlle)} \\ \vdots \end{gathered}$$

Asignación de las tres tripulaciones

$$\sum_{j=1}^{12} x_j = 3$$

Las soluciones óptimas son $x_3 = x_4 = x_{11} = 1$ y el resto 0 ó $x_1 = x_5 = x_{12} = 1$ y el resto 0, ambas con coste 18 millones de €.

Si no se permite que una tripulación fuera de servicio vuele en un avión las restricciones de cobertura de mayor o igual pasan a ser de igualdad. Luego, se trata de un problema de partición, cuya formulación se verá a continuación.

1.1.7 Problema de empaquetado (*set packing*)

Se tienen que realizar m proyectos divididos en n paquetes. La elección de un paquete implica realizar todos los proyectos del mismo. Se trata de maximizar el beneficio total de manera que cada proyecto i del conjunto de todos los paquetes que lo incluyen no pueda ser elegido más de una vez. c_j es el beneficio de elegir el paquete j, la matriz de pertenencia de cada proyecto i a cada paquete j es $a_{ij} \begin{cases} 1 & \text{si } i \text{ pertenece a } j \\ 0 & \text{si no pertenece} \end{cases}$. Las variables del problema son $x_j \begin{cases} 1 & \text{si se elige el paquete } j \\ 0 & \text{en cualquier otro caso} \end{cases}$. La formulación del problema es la siguiente

$$
\begin{aligned}
&\max_{x_j} \sum_{j=1}^{n} c_j x_j \\
&\sum_{j=1}^{n} a_{ij} x_j \le 1 \quad i = 1,\ldots,m \qquad (1.9)\\
&x_j \in \{0,1\}
\end{aligned}
$$

1.1.8 Problema de partición (*set partitioning*)

La formulación es similar al problema anterior pero en este caso exactamente una característica (proyecto) del conjunto de combinaciones (paquetes) que la contienen debe ser elegida.

$$
\begin{aligned}
&\max_{x_j} \sum_{j=1}^{n} c_j x_j \\
&\sum_{j=1}^{n} a_{ij} x_j = 1 \quad i = 1,\ldots,m \qquad (1.10)\\
&x_j \in \{0,1\}
\end{aligned}
$$

1.1.9 Problema del viajante de comercio (*Traveling Salesman Problem TSP*)

El problema consiste en hacer un recorrido que pase por n ciudades sin repetir ninguna y volviendo a la ciudad de partida de manera que la distancia (o tiempo o coste) total sea mínima. Es un problema de asignación pero con la condición de que la asignación sea un *ciclo*. Es uno de los problemas más importantes en la historia de la programación matemática por todas las investigaciones a las que ha dado lugar y por todas las aplicaciones que tiene, tanto directamente o apareciendo como subproblema dentro de otros más complejos. En una noticia de *OR/MS Today* (publicada por el *Institute of Operations Research and the Management Sciences* (*INFORMS*)) de junio de 2004, mencionaba que se había conseguido resolver un problema del viajante con 24978 ciudades. Los problemas de enrutamiento de vehículos (expedición o recogida de mercancías) pueden ser formulados de esta manera.

Una de las características más interesantes de este problema es que existen muchas formulaciones conocidas para el mismo, ver [Williams, 1999] y [Nemhauser, 1999]. Una de ellas es la siguiente. Sea c_{ij} la distancia entre las ciudades i y j.

Se definen las variables

$$x_{ij} = \begin{cases} 1 & \text{si se va de la ciudad } i \text{ a la ciudad } j \\ 0 & \text{en otro caso} \end{cases}$$

$$T_j : \text{ instante de llegada a ciudad } j$$

La formulación del problema es:

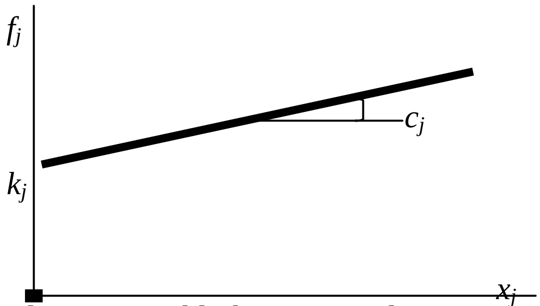
$$\begin{aligned} & \min_{x_{ij}} \sum_{i,j} c_{ij} x_{ij} \\ & \sum_i x_{ij} = 1 \quad \forall j \\ & \sum_j x_{ij} = 1 \quad \forall i \\ & T_j \geq T_i + c_{ij} - m(1 - x_{ij}) \quad \forall i, j \\ & T_j \geq c_{1j} - m(1 - x_{1j}) \quad \forall j \neq 1 \\ & x_{ij} \in \{0,1\}, T_j \geq 0 \end{aligned} \tag{1.11}$$

La primera restricción indica que a una ciudad j sólo se puede llegar una vez desde cualquier ciudad i. La segunda dice que desde una ciudad i sólo se puede salir una vez a cualquier otra ciudad j. Sólo con estas variables no es suficiente para formular el problema, ya que se pueden formar subciclos. La forma de evitarlos es añadiendo las variables continuas.

1.1.10 Problema de coste fijo

Los problemas de coste fijo aparecen cuando el coste de una variable tiene un término fijo con valor diferente de 0 si la variable toma un valor estrictamente positivo. Es una función no lineal y discontinua.

$$f_j(x_j) = \begin{cases} 0 & x_j = 0 \\ k_j + c_j x_j & x_j > 0 \end{cases}$$



Este coste se puede modelar con ayuda de una variable binaria auxiliar $y_j \in \{0,1\}$ definida como $y_j = \begin{cases} 1 & x_j > 0 \\ 0 & x_j = 0 \end{cases}$, que indica la realización de la actividad x_j.

Introduciendo la condición $x_j \le My_j$, , siendo M una constante, cota superior de x_j, cuyo valor dependerá del problema, se distingue entre no realizar la actividad y realizarla al menos infinitesimalmente. El valor de la constante M debe ser el menor posible ya que esto es computacionalmente beneficioso. El problema lineal entero se formula como sigue

$$\begin{aligned}
&\min_{x_j, y_j} \sum_{j=1}^{n} f_j(x_j) = \sum_{j=1}^{n} \left(k_j y_j + c_j x_j \right) \\
&x_j \le M y_j \\
&x_j \ge 0 \\
&y_j \in \{0,1\}
\end{aligned}$$

1.1.11 Modelado de algunas restricciones especiales

Supongamos que necesitamos considerar en un problema la condición de que si se produce el producto A también se debe producir el producto B. La condición de producción de un producto j la representamos por la restricción $x_j \ge 1$. Entonces, la implicación es

$$x_A \ge 1 \rightarrow x_B \ge 1$$

Esta condición no se puede introducir directamente en un problema lineal porque hace que la estructura del problema (el que se considere o no una restricción más $x_B \ge 1$) depende de que se cumpla otra ($x_A \ge 1$) y esto sólo se conoce una vez que se ha determinado la solución óptima. Un problema de optimización no se puede redefinir endógenamente, es decir, en función de los propios valores que toman las variables del problema.

En este apartado se van a modelar en un problema de optimización algunas condiciones especiales (las restricciones lógicas entre ellas) que requieren el uso de variables binarias para detectar o forzar el cumplimiento de restricciones.

1.1.11.1 Disyunciones

Las disyunciones implican una pareja de restricciones donde sólo una (cualquiera de las dos) debe satisfacerse, mientras que la otra no es necesario que se cumpla. Debe cumplirse una al menos pero no necesariamente las dos.

$$f(x) \le 0 \text{ ó } g(x) \le 0$$

Supongamos el ejemplo de esta disyunción

$$3x_1 + 2x_2 - 18 \le 0 \text{ ó } x_1 + 4x_2 - 16 \le 0$$

Veamos cómo estas restricciones se pueden incorporar en un problema de optimización. Añadir una constante de valor elevado M a una restricción es equivalente a eliminar (relajar) dicha restricción (se supone que las variables son positivas en estas restricciones), dado que los coeficientes de las variables son también positivos.

$$\begin{matrix} 3x_1 + 2x_2 - 18 \le 0 \\ x_1 + 4x_2 - 16 \le M \end{matrix} \text{ ó } \begin{matrix} 3x_1 + 2x_2 - 18 \le M \\ x_1 + 4x_2 - 16 \le 0 \end{matrix}$$

Se define la variable binaria auxiliar y que selecciona la ecuación correspondiente, $y = \begin{cases} 1 & \text{se relaja la ecuación 1} \\ 0 & \text{se relaja la ecuación 2} \end{cases}$. Luego las restricciones disyuntivas se modelan en un problema de optimización como

$$\begin{aligned} 3x_1 + 2x_2 - 18 &\le My \\ x_1 + 4x_2 - 16 &\le M(1-y) \end{aligned}$$

Si $y = 1$ se relaja la restricción 1 pero se obliga a cumplir la 2 y viceversa para $y = 0$.

Algunas implicaciones son un caso semejante a las restricciones disyuntivas

$$f(x) > 0 \quad g(x) \le 0$$

es equivalente a

$$f(x) \le 0 \text{ ó } g(x) \le 0$$

ya que $P \to Q$ es equivalente a (No P) ó Q.

1.1.11.2 Cumplir *k* de *N* ecuaciones

Se tiene un conjunto de N ecuaciones de las cuales se han de satisfacer al menos k, siendo $k < N$. Las disyunciones son un caso particular de éste para $k = 1$ y $N = 2$.

Sea el conjunto de N ecuaciones

$$f_1(x_1,\ldots,x_n) \le 0$$
$$f_2(x_1,\ldots,x_n) \le 0$$
$$f_N(x_1,\ldots,x_n) \le 0$$

añadiendo una constante M y una variable binaria y_i para cada ecuación tenemos

$$f_1(x_1,\ldots,x_n) \le My_1$$
$$f_2(x_1,\ldots,x_n) \le My_2$$
$$f_N(x_1,\ldots,x_n) \le My_N$$

donde además se impone la condición de seleccionar solamente k ecuaciones.

$$\sum_{i=1}^{N} y_i = N-k$$

$$y_i \in \{0,1\} \quad i=1,\ldots,N$$

1.1.11.3 Seleccionar entre *N* valores

Sea una función con múltiples posibles valores y se desea elegir uno de ellos.

$$f(x_1,\ldots,x_n) = \begin{cases} d_1 \\ d_2 \\ \vdots \\ d_N \end{cases}$$

La manera de modelarlo es introduciendo una variable binaria auxiliar y_i por cada valor y la condición de elección única.

$$f(x_1,\ldots,x_n) = \sum_{i=1}^{N} d_i y_i$$
$$\sum_{i=1}^{N} y_i = 1$$
$$y_i \in \{0,1\} \quad i=1,\ldots,N$$

1.1.11.4 Implicaciones sencillas

Retomemos el ejemplo de la restricción que aparecía en el problema de coste fijo

$$x \le M\delta$$

siendo M una cota superior positiva de x (por ejemplo, 10^6), $m \leq x \leq M$ y δ la variable binaria. Por claridad en la explicación en este apartado se utiliza la letra griega δ para denominar a la variable binaria auxiliar.

Si $\delta = 1$ la restricción no obliga a nada ya que $x \leq M$ se cumple por definición. Si $\delta = 0$ entonces $x \leq 0$. Luego esta restricción permite modelar la implicación

$$\delta = 0 \rightarrow x \leq 0 \text{ (si } \delta = 0 \text{ entonces se cumple que } x \leq 0\text{)}$$

Por otra parte, si $x > 0$ entonces $\delta = 1$. Si $x \leq 0$ la restricción no obliga a nada.

$$x > 0 \rightarrow \delta = 1 \text{ (si } x > 0 \text{ entonces se cumple que } \delta = 1\text{)}$$

Ambas son implicaciones equivalentes puesto que $P \rightarrow Q$ es equivalente a $\text{No } Q \rightarrow \text{No } P$. Luego, la restricción lineal $x \leq M\delta$ nos permite representar dichas implicaciones en un problema lineal.

De forma análoga veamos la restricción

$$x \geq m\delta$$

siendo m una cota inferior negativa de x (por ejemplo, -10^6), $m \leq x \leq M$ y δ la variable binaria.

Si $\delta = 1$ la restricción no obliga a nada ya que $x \geq m$ se cumple por definición. Si $\delta = 0$ entonces $x \geq 0$. Luego esta restricción permite modelar la implicación

$$\delta = 0 \rightarrow x \geq 0 \text{ (si } \delta = 0 \text{ entonces se cumple que } x \geq 0\text{)}$$

Por otra parte, si $x < 0$ entonces $\delta = 1$. Si $x \geq 0$ la restricción no obliga a nada.

$$x < 0 \rightarrow \delta = 1 \text{ (si } x < 0 \text{ entonces se cumple que } \delta = 1\text{)}$$

Nuevamente ambas son implicaciones equivalentes puesto que $P \rightarrow Q$ es equivalente a $\text{No } Q \rightarrow \text{No } P$.

En resumen, hasta ahora hemos visto la representación en un problema lineal de las siguientes implicaciones

$$\left.\begin{array}{r} \delta = 0 \rightarrow x \leq 0 \\ x > 0 \rightarrow \delta = 1 \end{array}\right\} x \leq M\delta$$

$$\left.\begin{array}{r} \delta = 0 \rightarrow x \geq 0 \\ x < 0 \rightarrow \delta = 1 \end{array}\right\} x \geq m\delta$$

A continuación, vamos a generalizar la representación de implicaciones para cualquier tipo de restricción genérica.

1.1.11.5 Implicaciones de una restricción $\leq$

La implicación

$$\delta = 1 \rightarrow \sum_j a_j x_j \leq b$$

es equivalente a

$$\sum_j a_j x_j \leq b + M(1-\delta)$$

siendo una cota superior de la restricción para cualquier valor de cualquier x_j, $\sum_j a_j x_j - b \leq M$. Efectivamente de manera directa se deduce que si $\delta = 1$ se impone la restricción original y si $\delta = 0$ no implica nada (se relaja la restricción original). Análogamente al caso anterior esta restricción también representa la implicación

$$\sum_j a_j x_j > b \rightarrow \delta = 0$$

La implicación

$$\sum_j a_j x_j \leq b \rightarrow \delta = 1$$

se puede transformar en $\delta = 0 \rightarrow \sum_j a_j x_j > b$ o bien en $\delta = 0 \rightarrow \sum_j a_j x_j \geq b + \varepsilon$ que es equivalente a

$$\sum_j a_j x_j \geq b + \varepsilon + (m - \varepsilon)\delta$$

siendo m una cota inferior de la restricción para cualquier valor de cualquier x_j, $\sum_j a_j x_j - b \geq m$.

1.1.11.6 Implicaciones de una restricción $\geq$

De manera simétrica se pueden representar las implicaciones con restricciones de tipo mayor o igual.

La implicación

$$\delta = 1 \rightarrow \sum_j a_j x_j \geq b$$

es equivalente a

$$\sum_j a_j x_j \geq b + m(1-\delta)$$

siendo m una cota inferior de la restricción para cualquier valor de cualquier x_j, $\sum_j a_j x_j - b \geq m$. Efectivamente de manera directa se deduce que si $\delta = 1$ se impone la restricción original y si $\delta = 0$ no implica nada (se relaja la restricción original). Análogamente al caso anterior esta restricción también representa la implicación

$$\sum_j a_j x_j < b \rightarrow \delta = 0$$

La implicación

$$\sum_j a_j x_j \geq b \rightarrow \delta = 1$$

se puede transformar en $\delta = 0 \rightarrow \sum_j a_j x_j < b$ o bien en $\delta = 0 \rightarrow \sum_j a_j x_j \leq b - \varepsilon$ que es equivalente a

$$\sum_j a_j x_j \leq b - \varepsilon + (M + \varepsilon)\delta$$

siendo M una cota superior de la restricción para cualquier valor de cualquier x_j, $\sum_j a_j x_j - b \leq M$.

1.1.11.7 Implicaciones de una restricción $=$

Para deducir las implicaciones de restricciones igualdad se transforman en ecuaciones de tipo mayor o igual y menor o igual simultáneamente.

La implicación

$$\delta = 1 \rightarrow \sum_j a_j x_j = b$$

es equivalente a

$$\delta = 1 \rightarrow \sum_j a_j x_j \leq b$$
$$\delta = 1 \rightarrow \sum_j a_j x_j \geq b$$

Luego se representa por las ecuaciones

$$\sum_j a_j x_j \leq b + M(1-\delta)$$
$$\sum_j a_j x_j \geq b + m(1-\delta)$$

Efectivamente para $\delta = 1$ se cumplen ambas restricciones y para $\delta = 0$ ambas se relajan.

La implicación

$$\sum_j a_j x_j = b \rightarrow \delta = 1$$

es una combinación de los casos anteriores simultáneamente

$$\begin{array}{l} \sum_j a_j x_j \leq b \rightarrow \delta' = 1 \\ \sum_j a_j x_j \geq b \rightarrow \delta'' = 1 \end{array} \text{ y además } \delta' = 1 \text{ y } \delta'' = 1 \rightarrow \delta = 1$$

que se modela con las restricciones

$$\sum_j a_j x_j \geq b + \varepsilon + (m - \varepsilon)\delta'$$
$$\sum_j a_j x_j \leq b - \varepsilon + (M + \varepsilon)\delta''$$

y la restricción adicional que indica el cumplimiento de ambas.

$$\delta' + \delta'' - \delta \leq 1$$

1.1.11.8 Implicaciones dobles

Para formular implicaciones dobles éstas se desdoblan en las implicaciones unidireccionales correspondientes.

$$\delta = 1 \leftrightarrow \sum_j a_j x_j \leq b \text{ es equivalente a } \begin{cases} \delta = 1 \rightarrow \sum_j a_j x_j \leq b \\ \sum_j a_j x_j \leq b \rightarrow \delta = 1 \end{cases}$$

y lo mismo para los otros tipos de restricciones.

1.1.12 Problemas de producción con elasticidad en los precios y/o costes

En algunos problemas de producción se puede suponer que hay una ganancia unitaria fija asociada a cada producto, con lo que la función objetivo de beneficio que se obtiene es lineal. Sin embargo, en otros problemas ciertos factores introducen no linealidades en la función objetivo. Por ejemplo, un gran fabricante puede encontrar *precios elásticos* mediante los cuales la cantidad que se puede vender de un producto va en relación inversa con el precio que se cobra. La curva precio-demanda, $p(x)$, que representa el precio unitario que se necesita para poder vender x unidades, sería una función no lineal decreciente, nunca inferior al coste unitario de producción c.

Así el margen de contribución de la empresa (ingreso bruto menos coste de producción, beneficio neto, EBITDA) vendría determinado por

$$P(x) = xp(x) - cx$$

Si, además, la empresa tiene una función semejante para cada uno de los n productos que puede fabricar la función objetivo global sería una suma de funciones no lineales.

$$f(x) = \sum_{j=1}^{n} P_j(x) = \sum_{j=1}^{n} \left[x_j p_j(x_j) - c_j x_j \right]$$

Otra razón por la que pueden surgir no linealidades en la función objetivo es a causa de los costes de producción, ya que éstos pueden variar con el nivel de producción. Por ejemplo, el coste puede decrecer cuando aumenta el nivel de producción gracias al efecto de una *curva de aprendizaje* (mayor eficiencia con más experiencia) o aumentar por necesidad de tiempos extra o instalaciones más costosas.

Las restricciones también se pueden ver afectadas por estos tipos de no linealidades. Una que surge inmediatamente es la restricción de presupuesto, si existe, cuando los costes de producción varían como se ha descrito anteriormente. También serán funciones no lineales las asociadas a los recursos, siempre que el uso de un determinado recurso no sea proporcional a los niveles de los respectivos productos.

1.1.13 Problema de transporte con descuentos por volumen

El problema de transporte que se ha considerado hasta el momento supone que el coste por unidad enviada de un origen a un destino dados es fijo, independientemente de la cantidad mandada. Sin embargo, una situación muy habitual es que se disponga de *descuentos por cantidad* para volúmenes grandes, con lo que la función de coste unitaria sería una función no lineal con pendiente no creciente. Una alternativa es aproximar esta función no lineal por una poligonal.

Así pues, el coste de embarcar x unidades viene dado por una función *poligonal*, $C(x)$, continua, con pendiente en cada tramo igual al coste unitario de transporte. En consecuencia, si cada combinación de origen y destino tiene una función semejante, la función objetivo sería

$$f(x) = \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij}(x_{ij})$$

Al ser una función poligonal cóncava en un problema de minimización se modelará introduciendo variables binarias de selección del segmento de la poligonal.

1.1.14 Selección de una cartera de inversiones

Actualmente, cuando se plantea la selección de una cartera de inversiones, los inversores se preocupan tanto por el rendimiento esperado como por el riesgo asociado a su inversión y para obtener un modelo que permita determinar una cartera que, con ciertas suposiciones, combine de forma óptima estos factores se utiliza la programación no lineal.

Supongamos que se están considerando n tipos de acciones para incluirlas en la cartera; las variables de decisión x_j, $j = 1, \ldots, n$ representan el número de acciones j que se van a incluir. Sean μ_j y σ_{jj} la media y la varianza del rendimiento sobre cada acción de tipo j, en donde σ_{jj} es una medida del riesgo de estas acciones. Sea σ_{ij} la covarianza del rendimiento sobre una acción de cada tipo i y j. Entonces, el valor esperado $R(x)$ y la varianza $V(x)$ del rendimiento total de la cartera son

$$R(x) = \sum_{j=1}^{n} \mu_j x_j$$

$$V(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j$$

con lo que la función objetivo del modelo resultante es

$$f(x) = R(x) - \beta V(x)$$

donde β se denomina factor de aversión al riesgo, ya que cuanto mayor sea mayor importancia (negativa) se le da en la función objetivo a la variabilidad (la volatilidad del rendimiento no es más que su desviación estándar) de la inversión final.

Como restricción se incluye la restricción del presupuesto y la no negatividad de las variables (P_j representa el coste de cada acción de tipo j y B es el presupuesto):

$$\sum_{j=1}^{n} P_j x_j \le B$$
$$x_j \ge 0 \quad j = 1, \ldots, n$$

1.1.15 Referencias

Hillier, F.S., Lieberman, G.J. (2002) *Investigación de Operaciones*. 7ª edición. McGraw Hill.

Nemhauser, G.L. and Wolsey, L.A. (1999) *Integer and Combinatorial Optimization*. John Wiley and Sons.

Williams, H.P. (1999) *Model Building in Mathematical Programming*. 4th Edition. John Wiley and Sons.

Wolsey, L.A. (1998) *Integer Programming*. John Wiley and Sons.

1.1.16 Biblioteca de problemas

PROBLEMA: AYUDA EN EMERGENCIAS

Tienen que transportarse sacos con alimentos mediante tres tipos de aviones A1, A2, A3, desde un aeropuerto y arrojarse en las aldeas V1, V2, V3, V4, V5, afectadas por inundaciones. La cantidad de alimentos (en unidades adecuadas) que cada avión puede transportar a cada aldea en cada viaje, se da en la siguiente tabla. El número de viajes que puede hacer cada avión se da en la última columna y el número máximo de aviones que puede recibir diariamente cada aldea en la última fila. Encontrar el número de viajes que deberá hacer cada avión a cada aldea de forma que se maximice la cantidad de alimento distribuido por día.

	V1	V2	V3	V4	V5	
A1	10	8	6	9	12	50
A2	5	3	8	4	10	90

A3	7	9	6	10	4	60
	100	80	70	40	20	

PROBLEMA: CONSTRUCCIÓN DE ALMACENES

Una compañía planea construir varios almacenes para guardar un cierto producto. Estos almacenes surtirán a dos grandes clientes con las unidades demandadas mensualmente apuntadas en la última fila de la tabla. Se pueden construir hasta tres almacenes, que se tienen como candidatos, con capacidades expresadas en la última columna. Usando el coste estimado de construcción de los almacenes, su vida útil y el valor del dinero en el tiempo, los costes de construcción por mes para los tres almacenes se han estimado en 8000, 12000 y 7000. A continuación se dan los costes de transporte por unidad desde los tres almacenes candidatos a los clientes.

	Cliente 1	Cliente 2	Capacidad
Almacén 1	1.50	2.00	4000
Almacén 2	2.00	1.50	5000
Almacén 3	2.50	2.25	6000
Demanda	3000	5000	

Determinar qué almacenes se deben construir y cómo se ha de satisfacer la demanda de los clientes.

PROBLEMA: TRANSPORTE DE ELECTRODOMÉSTICOS

Una compañía tiene dos fábricas, una en Alicante y otra en Huelva. Las dos fábricas producen frigoríficos y lavadoras. Las capacidades de producción de estos artículos en Alicante son de 5000 y 7000, respectivamente, y en Huelva de 8000 y 4000. La compañía entrega estos productos a tres grandes clientes en las ciudades de Barcelona, A Coruña y Valencia, siendo las demandas:

Demanda/Cliente	Barcelona	A Coruña	Valencia
Frigoríficos	4000	5000	4000
Lavadoras	3000	3000	4000

Los artículos se transportan por ferrocarril. En la tabla siguiente se muestran los costes unitarios de transporte y las limitaciones para enviar cualquiera de los dos productos de cada fábrica a cada cliente:

		Barcelona	A Coruña	Valencia
Alicante	Coste unitario	6	14	7
	Máximo unidades	6000	3000	7500
Huelva	Coste unitario	10	8	15
	Máximo unidades	3000	9000	3000

Se desea minimizar el coste total de transporte.

PROBLEMA: LOGÍSTICA

Una empresa tiene dos factorías, F1 y F2, con las que abastece a tres almacenes de distribución, D1, D2 y D3, de dos artículos, A1 y A2.

Los costes de transporte de una unidad de cualquiera de los dos artículos desde cada factoría a cada almacén se dan en la tabla izquierda, en tanto que los precios de venta unitarios de cada artículo en cada almacén se dan en la tabla derecha.

Coste Tr	D1	D2	D3
F1	4	7	5
F2	6	5	7

Precio	D1	D2	D3
A1	17	20	18
A2	19	17	21

El tiempo, expresado en minutos, que se tarda en fabricar una unidad de cada artículo en cada una de las factorías se refleja en la tabla izquierda, en tanto que los costes unitarios de fabricación de cada artículo en cada factoría aparecen en la tabla derecha.

Tiempo	A1	A2
F1	6	7.5
F2	10	5

Coste Fb	A1	A2
F1	8	6
F2	5	10

La capacidad de producción de la factoría 1 es de 260 horas y la de la factoría 2 de 240 horas.

Las demandas mínimas de cada uno de los artículos que en cada almacén deben ser satisfechas son expresadas en la tabla siguiente.

	D1	D2	D3
A1	600	800	500
A2	700	500	1200

Por último y por cuestiones de tipo técnico y de política de empresa, nunca se pueden producir en cualquiera de las factorías más de 500 unidades de un artículo que de otro.

Se trata de elaborar un modelo que proporcione el mejor programa de producción y distribución para maximizar el beneficio neto.

¿Se modifica la solución si el tiempo de ejecución de A2 en F1 se reduce en medio minuto?

PROBLEMA: GESTIÓN DE AUTOBUSES

En una ciudad se intenta disminuir la contaminación reduciendo la circulación interurbana. Un primer estudio busca determinar el mínimo número de autobuses que satisfagan las necesidades de transporte. Después de recoger la información se observa que este número varía según la hora del día, pero se puede considerar constante en intervalos sucesivos de cuatro horas:

00:00 a.m. – 4:00 a.m.	4
4:00 a.m. – 8:00 a.m.	8
8:00 a.m. – 12:00 m.	10

12:00 m. – 4:00 p.m.	7
4:00 p.m. – 8:00 p.m.	12
8:00 p.m. – 00:00 a.m.	4

Los turnos de autobuses funcionan durante ocho horas seguidas y pueden comenzar al principio de cualquiera de los seis periodos descritos anteriormente. Además, si en el turno que comienza a las 8:00 p.m. hay estrictamente más de 4 autobuses, en el siguiente ha de haber también estrictamente más de 4. Plantear un problema de programación lineal entera para determinar el mínimo número de autobuses diario que satisface las necesidades anteriores.

PROBLEMA: ADQUISICIÓN DE CAMIONES

Una compañía de transportes tiene 10 camiones con capacidad 40000 kg y 5 camiones de 30000 kg. Los camiones grandes tienen un coste variable de combustible de 0.30 €/km y los pequeños de 0.25 €/km.

En una semana la empresa debe transportar 400000 kg en un recorrido de 800 km. La posibilidad de otros compromisos recomienda que por cada dos camiones pequeños mantenidos en reserva debe quedarse por lo menos uno de los grandes.

¿Cuál es el número óptimo de camiones de ambas clases que deben movilizarse para ese transporte y teniendo en cuenta las restricciones?

PROBLEMA: PLANIFICACIÓN DEL METRO

En una determinada ciudad se va a construir la red del metro. La empresa encargada ha de decidir qué líneas construir y para ello tiene varias opciones. Existen 10 puntos claves por los que ha de pasar la red y se ha visto que son 8 las posibles líneas a construir. Las líneas posibles, los puntos clave por los que pasaría cada una y su coste estimado de construcción en unidades apropiadas, son:

	Puntos clave	Coste
L1	P1 P2 P3 P4	4
L2	P1 P3 P5 P7	4
L3	P2 P3 P4 P6	4
L4	P5 P7 P9 P10	4
L5	P2 P7 P8	3
L6	P1 P4 P5 P10	4
L7	P3 P8 P9	3
L8	P2 P6 P10	3

Además, por el punto P2 han de pasar al menos dos líneas; y, si los puntos P3 y P7 no quedan conectados por una línea directa, entonces debe existir un transbordo en P8 de modo que pase una línea que una este punto con el P3 y otra con el P7. Plantear como un problema de programación lineal entera el problema de decidir qué líneas construir de la forma más económica con estas restricciones teniendo en cuenta que por cada punto clave debe pasar al menos una línea.

PROBLEMA: OFICINA DE CORREOS

Una oficina de correos necesita distinto número de empleados de jornada completa para cada día de la semana, tal como se da en la tabla adjunta. Las reglas sindicales señalan que cada empleado de jornada completa tiene que trabajar durante cinco días consecutivos y, a continuación, descansar dos días. Por ejemplo, un empleado que trabaje de lunes a viernes tiene que descansar sábado y domingo. La oficina de correos quiere cumplir con sus requerimientos diarios y utilizar sólo empleados de jornada completa. Formular mediante programación matemática un modelo que pueda utilizar la oficina de correos para minimizar el número de empleados de jornada completa a contratar.

	Empleados
Lunes	17
Martes	13
Miércoles	15
Jueves	19
Viernes	14
Sábado	16
Domingo	11

PROBLEMA: ABASTECIMIENTO

Una empresa abastecedora de agua tiene que llevar agua de un punto s a un punto t y para realizar la conexión entre ambos puntos ha de pasar por unos puntos intermedios. Cada conexión entre un par de puntos tiene un coste estimado de construcción y, una vez construida, un coste unitario de envío de cada litro y una capacidad por hora que se recogen en la siguiente tabla:

Conexión	Coste construcción	Coste envío l/min	Capacidad l/min
s–1	100000	40	100

s–2	200000	50	200
1–3	80000	60	50
1–*t*	100000	70	30
2–3	200000	40	20
2–*t*	200000	70	100
3–*t*	150000	60	60

Plantear un problema de programación matemática si se quieren enviar 180 litros por minuto de la forma más económica posible, teniendo en cuenta que si se construye la conexión de s a 2 ha de hacerse la de 2 a t.

PROBLEMA: ADQUISICIÓN DE MÁQUINAS TROQUELADORAS

Una compañía tiene tres tipos de máquinas troqueladoras de diferente velocidad y precisión:

	Velocidad (piezas/hora)	Precisión (%)	Coste (€/hora)
Tipo 1	20	99	2.00
Tipo 2	15	95	1.75
Tipo 3	10	99	1.50

Cada día (8 horas) se deben procesar por lo menos 3500 piezas y hay disponibles 8 máquinas del tipo 1, 10 del tipo 2 y 20 del tipo 3.

Si cada pieza errónea le cuesta a la compañía 1 céntimo. ¿Cuántas máquinas de cada tipo se deben utilizar para minimizar los costes?

PROBLEMA: SECUENCIACIÓN DE TRABAJOS EN UNA MÁQUINA

Dados unos trabajos que realizar, una duración de éstos y una fecha de entrega prevista, plantear un problema de programación lineal entera para encontrar la secuencia que minimiza el retraso o demora media con que los trabajos son entregados, con los siguientes datos:

Tarea	T1	T2	T3	T4
Tiempo de proceso	9	12	7	14
Fecha de entrega	15	19	23	31

PROBLEMA: PRODUCCIÓN II

En una empresa familiar se producen dos tipos de productos, 1 y 2, procesando materia prima. Se pueden comprar hasta 90 kg de materia prima a un coste de 10 €/kg. Se puede usar 1 kg de materia prima para producir 1 kg de producto 1 o para producir 1/2 kg del producto 2. Usar 1 kg de materia prima para producir el producto 1 requiere 2 horas de mano de obra. Usar 1 kg de materia prima para procesar el producto 2 requiere 3 horas de mano de obra. Se dispone de 300 horas de mano de obra a 3 €/hora. Se pueden vender a lo sumo 40 kg del producto 2. El producto 1 se vende a 29 €/kg y el producto 2 a 69 €/kg.

Además existe una limitación inferior y superior en caso de que se produzca alguna cantidad de cada artículo. Es decir, si se produce algo del producto 1 ha de ser más de 15 y menos de 30 kg y si se produce algo del producto 2 ha de ser más de 10 y menos de 20 kg. Plantear el problema y obtener la solución óptima.

PROBLEMA: PRODUCCIÓN E INVENTARIO

Una empresa desea planear su política de producción/inventario para los meses de agosto, septiembre, octubre y noviembre. La demanda estimada del producto para esos meses es de 500, 600, 800 y 1000 unidades, respectivamente. En la actualidad, la capacidad de producción mensual es de 600 unidades con un coste de 2500 €. La administración ha decidido instalar un nuevo sistema de producción con capacidad mensual de 1100 unidades a un coste por unidad de 3000 €. Sin embargo, el nuevo sistema no puede ser instalado hasta noviembre. Supóngase que el inventario inicial es de 250 unidades y que, durante cualquier mes dado, se pueden almacenar a lo sumo 400 unidades. Si el coste mensual por unidad por mantener en inventario es de 300 €, minimizar el coste total de producción e inventario. Suponer que se debe satisfacer la demanda y que se requiere tener 100 unidades en inventario al final de noviembre.

PROBLEMA: MISIÓN PACÍFICA

En una misión pacífica de las Naciones Unidas se dispone de 5 aviadores para formar las tripulaciones de dos aviones biplaza. Estos aviadores son de distintas nacionalidades: Español, Francés, Italiano, Griego y Portugués. Como en toda cuestión diplomática las relaciones internacionales son de gran peso, cada una de las distintas composiciones de las tripulaciones conlleva un beneficio, siendo éstos:

	Francés	Italiano	Griego	Portugués
Español	2	5	4	3
Francés		4	4	2
Italiano			5	4
Griego				3

Por otra parte, estas mismas relaciones internacionales hacen que si una tripulación está formada por el aviador español y el italiano la otra ha de estar formada por el aviador francés y el griego. Formular el problema de programación lineal entera.

PROBLEMA: MEZCLA DE CRUDO

La empresa Sunco Oil produce dos tipos de gasolina (1 y 2), cada una de ellas mezclando dos tipos de crudo (1 y 2). Los precios de venta de cada barril de gasolina son 7000 y 6000 €, respectivamente. Por su parte, los precios de compra de los dos tipos de crudo son de 4500 y 3500 € por barril, respectivamente. Se pueden comprar hasta 5000 barriles de cada crudo diarios. Los dos tipos de gasolina difieren en su índice de octano y en su contenido en azufre. La mezcla del petróleo crudo que se utiliza para obtener la gasolina 1 ha de tener un índice de octano promedio de al menos 10 y a lo sumo un 1 % de azufre. La mezcla que se obtiene para la gasolina 2 ha de tener un índice promedio de octano de por lo menos 8 y a lo sumo un 2 % de azufre. Los índices de octano y el contenido en azufre de los dos tipos de crudo son:

Crudo	Octano	Azufre
1	12	0.5
2	6	2.0

La transformación de un barril de petróleo en un barril de gasolina cuesta 400 € y la refinería de Sunco puede producir diariamente hasta 9000 barriles de gasolina.

Los clientes de Sunco actualmente demandan 3000 barriles de la gasolina 1 y 2000 de la gasolina 2. Sin embargo, Sunco tiene la posibilidad de estimular la demanda mediante la publicidad, de modo que cada euro invertido en la publicidad de cada tipo de gasolina, aumenta la demanda diaria de ese tipo de gasolina en 0.1 barriles (si por ejemplo gasta 1000 € en publicidad de la gasolina 1 aumenta la demanda de gasolina 1 en 100 barriles). Formular un problema de programación lineal que permita a Sunco maximizar sus ganancias diarias.

PROBLEMA: PRODUCCIÓN VI

Una planta de producción dispone de m máquinas para llevar a cabo su producción. La demanda semanal del producto es conocida para las siguientes n semanas, dem_j siendo j cada una de las semanas, y ha de ser satisfecha. Cada una de las máquinas i puede estar arrancada y produciendo durante cada semana o no, pero si lo está tiene un coste fijo por estar arrancada de cf_i €, siendo su producción máxima pm_i. Además, el coste unitario de producción con cada una de las máquinas es variable con las semanas, siendo cv_{ij} € por unidad de producto, y el coste de almacenamiento de una semana a otra está estimado en $calm$ € por unidad de producto. Por otra parte, arrancar una máquina para acoplarla una semana si no lo estaba la anterior tiene un coste de arranque $carr_i$. Se supone que todas las máquinas inicialmente están arrancadas (no hay coste de arranque para la primera semana).

a) Plantear un modelo lineal para optimizar la planificación de la producción siendo el horizonte de planificación las n semanas.

b) Sobre la formulación anterior supóngase ahora que cuando una máquina para, ha de hacerlo al menos dos semanas consecutivas por razones técnicas, ¿cómo se modelaría esta nueva condición?

PROBLEMA: TRANSPORTE POR FERROCARRIL

Una empresa de transporte opera en una línea ferroviaria con un determinado material rodante que puede transportar un volumen máximo V y un peso máximo P de mercancías, transportando distintas mercancías de otras empresas. Para un determinado día dispone de distintas solicitudes de mercancías, de modo que de cada mercancía i

conoce la cantidad máxima a transportar mx_i, el volumen v_i y peso p_i unitarios, el pago que el propietario de la mercancía está dispuesto a pagar a la empresa por cada unidad transportada, b_i, y el coste que estima la empresa por cada unidad a transportar, c_i. La empresa de transporte desea tener un modelo que le permita elegir cada día las cantidades de las mercancías que le proporcionen mayor beneficio a partir de estos datos diarios.

A su vez, puede variar la capacidad del material rodante, tanto en volumen o peso, de modo que desea saber en cuánto puede valorar cada unidad extra de volumen y de peso de la que puede disponer, para saber cuánto puede estar dispuesto a pagar por un aumento de éstos.

Por otra parte, a los propietarios de las mercancías que no son transportadas, puede y debe informarles de la cantidad en la que deberían aumentar su oferta para que su mercancía fuera transportada (en detrimento de otras).

a) Presentar un modelo lineal para este problema, suponiendo que las cantidades se pueden fraccionar. ¿Qué elementos del modelo darías como resultado para responder a las distintas preguntas planteadas?

b) Supóngase ahora que la oferta conlleva un descuento por volumen de modo que la cantidad que un propietario paga por cada unidad es de la forma $a_i - g_i x_i$, donde x_i es la cantidad transportada. Plantear el nuevo modelo.

c) Sobre el planteamiento del apartado a), supóngase que lo que hay es una capacidad de volumen y peso por vagón, $\overline{V}$ y $\overline{P}$, y que hay que decidir además de cuánto transportar, cuántos vagones hay que enganchar, existiendo un coste unitario por vagón utilizado, $cvag$. Plantear el nuevo modelo.

PROBLEMA: EDICIÓN DE CDS

Una compañía discográfica está pensando en editar una colección de grandes éxitos de uno de sus cantantes más famosos. Todas sus canciones han sido agrupadas en doce lotes. Cada lote ocupa tiene un cierto tamaño expresado en MB y tiene asociado un índice de marketing (relacionado con su demanda esperada).

Lote	Tamaño	Índice marketing
1	350	15

2	300	32
3	160	20
4	310	25
5	260	35
6	250	10
7	400	40
8	100	34
9	520	24
10	170	16
11	300	36
12	360	26

Los lotes van a ser grabados en una serie de CDs, cada uno con una capacidad máxima de 700 MB. Además, para que el CD funcione en el mercado, se estima que debe tener un índice de marketing superior a 45. Por otro lado, y dado el contenido de sus canciones, los lotes 1, 2 y 3 se han de grabar en CDs distintos. El objetivo de la compañía es maximizar el número de CDs editados, utilizando todos los lotes de canciones y sin editar más de una vez el mismo lote.

PROBLEMA: VUELOS CHARTER

Una compañía aérea tiene una flota de 15 aeronaves: 5 de cada uno de tres tipos A, B y C, cuyas respectivas capacidades para el transporte de viajeros son de 80, 68 y 55 personas. Una agencia de viajes le solicita presupuesto para trasladar a 372 personas. La compañía analiza sus costes, que dependen del número de aviones de cada tipo que quiera utilizar para transportar a esas personas, datos que se dan en la siguiente tabla en miles de euros

Tipo	1	2	3	4	5
A	11	20	30	40	50
B	9	17	24	34	45
C	8	15	21	26	31

Además la compañía aérea incurre en un coste fijo adicional de 6 k€ por cada tipo de aviones que utilice.

Proponer un modelo de programación lineal entera cuyo objetivo sea determinar la composición óptima de la flotilla de aviones que va a realizar el transporte para minimizar los costes de la operación.

PROBLEMA: PROVEEDORES

Tres almacenes deben abastecer a cuatro mercados de cierto número de unidades de un producto. Los costes unitarios de transporte (en €), las existencias en los almacenes y las necesidades de los mercados se dan en la tabla siguiente.

	M1	M2	M3	M4	Existencias en almacenes
A1	6	7	9	5	220
A2	7	10	9	6	350
A3	5	8	7	5	270
Demandas en mercados	80	90	100	135	

Cuando un almacén abastece a un mercado ha de firmar un contrato para establecerse como proveedor del mercado, de modo que ha de pagar una cantidad por el único hecho de ser proveedor, establecida en 100 €. La demanda de los mercados ha de ser satisfecha.

1. Establecer un programa lineal entero que permita obtener el plan de abastecimiento más económico.
2. Sobre el planteamiento anterior, si de un almacén cualquiera se envía a todos los mercados un total de menos de 100 unidades, sus costes unitarios de la tabla anterior se incrementan en 2 €, es decir, el coste unitario de la tabla es para todas las unidades enviadas desde un almacén a un mercado si la cantidad total que el almacén envía es mayor o igual que 100. Establecer un programa lineal entero que permita obtener el plan de abastecimiento más económico.

PROBLEMA: COMPOSICIÓN DE PIENSO

Un pequeño ganadero alimenta sus reses con una mezcla de dos piensos compuestos, P1 y P2, que él mismo elabora, y en los que es posible encontrar tres nutrientes N1, N2 y N3, de acuerdo con lo reflejado en la tabla, donde se da el contenido, en gramos, de cada nutriente por kilo de pienso compuesto.

	N1	N2	N3	Aditivos
P1	100	300	400	200
P2	200	250	300	250

Los costes de fabricación de un kilo de cada pienso son de 0.3 € para P1 y de 0.36 € para P2. Las necesidades diarias de una res respecto a los nutrientes considerados son:

N1: entre 250 y 300 gramos, con una cantidad óptima de 250 gramos

N2: entre 325 y 460 gramos, con una cantidad óptima de 400 gramos

N3: entre 450 y 600 gramos, con un óptimo de 500 gramos.

La desviación, en más o en menos, de la cantidad de nutrientes proporcionados a una res respecto al valor óptimo antes indicado requiere un tratamiento compensatorio, con costes de 0.01 € por gramo, en el caso de N1, 0.005 € por gramo en el caso de N2 y de 0.008 € por gramo en el caso de N3.

1. Plantear un modelo de programación lineal para determinar qué cantidad de cada pienso hay que proporcionar a cada res para que el coste de su alimentación sea mínimo
2. Además, si la cantidad utilizada de P1 fuera estrictamente mayor que el doble de la de P2 el coste de P1 se incrementaría en 0.05 €/kg. Modificar el modelo anterior añadiendo este supuesto y sin que deje de ser lineal.

PROBLEMA: EXPLOTACIONES GANADERAS

En dos explotaciones ganaderas G1 y G2 se crían vacas y cerdos, con los cuales se intenta satisfacer las demandas de tres mataderos industriales M1, M2 y M3.

Las disponibilidades y demandas actuales de vacas y cerdos en explotaciones y mercados, así como los costes de transporte unitarios entre ellos, en €, se resumen en las siguientes tablas.

	Vacas	Cerdos

	M1	M2	M3	Disponibilidad	M1	M2	M3	Disponibilidad
G1	25	21	20	**25**	6	7	8	**130**
G2	27	19	25	**32**	8	6	7	**100**
Demanda	**17**	**6**	**22**		**45**	**85**	**76**	

En cada explotación hay 6 vehículos especialmente adaptados para el transporte de vacas y 8 para el de cerdos.

Un vehículo para vacas tiene una capacidad en volumen de 100 unidades y uno para cerdos tiene una capacidad de 80 unidades de volumen. Con independencia de la explotación ganadera y del matadero entre los que puedan hacer un viaje, el uso de un camión para vacas tiene un coste fijo de 200 €, además del coste directamente asociado a las vacas transportadas. Por lo que hace a los camiones para cerdos, dicho coste fijo es de 165 €.

Una vaca ocupa 12 unidades de volumen y un cerdo 3.

1. Elaborar un modelo de programación lineal entera que satisfaga las demandas de los mataderos con un coste de transporte mínimo.
2. Supóngase ahora que también es posible, si conviene, transportar cerdos en un camión para vacas, aunque ello supone ciertas adaptaciones que incrementan el coste unitario de transporte de los cerdos en 2 €. De la misma forma, es posible transportar vacas en los vehículos para cerdos, con un coste unitario adicional de 10 €. Suponiendo que en un camión pueden ir vacas o cerdos simultáneamente, elaborar un modelo de programación lineal entera que satisfaga las demandas de los mataderos con un coste de transporte mínimo.

PROBLEMA: CORTE DE BOBINAS

La empresa EIVISSA se dedica a la fabricación de bobinas madre de plástico con un ancho de 6000 mm. Sus clientes son empresas de envasado cuyos pedidos son bobinas hija de anchos inferiores, tal como aparecen en la siguiente tabla:

Pedido	Anchura [mm]	Fecha de entrega [horas]
1	1500	6
2	1200	10

3	800	7
4	750	8
5	915	8
6	315	10
7	1450	5
8	650	4
9	725	7
10	1800	12

Se trata de plantear un modelo de optimización para determinar los cortes a realizar en las bobinas madre para satisfacer los pedidos de bobinas hija con estas posibles diferentes funciones objetivo alternativas:

1. Plantear el modelo de optimización completo para minimizar el número total de bobinas madre necesitadas
2. Minimizar la suma del desperdicio lateral sobrante de todas las bobinas
3. Minimizar la suma de los retrasos de cada pedido con respecto a las fechas de entrega teniendo en cuenta que el corte de una bobina madre en bobinas hija dura 6 horas y que sólo se dispone de una máquina de corte de bobinas madre por lo que el corte de cada bobina se hace consecutivamente

PROBLEMA EMPRESA DISTRIBUIDORA

Una empresa distribuidora tiene dos almacenes centrales en las ciudades C1 y C2 desde los que abastece a tres centros de distribución situados en D1, D2 y D3. Desde estos centros de distribución se aprovisionan a cuatro mercados M1, M2, M3 y M4.

Sean e_1 y e_2 las existencias en los almacenes centrales y d_1, d_2, d_3 y d_4 las demandas de los mercados. El coste de transporte unitario desde el almacén central i al centro de distribución j es c_{ij} y el de transportar una unidad desde el centro de distribución j al mercado k es w_{jk}. Cada unidad del artículo gestionado tiene un peso p.

1. Elaborar un modelo de programación lineal que permita satisfacer las demandas de los mercados al menor coste.
2. ¿Cómo se modifica dicho modelo si M1 no puede ser aprovisionado simultáneamente desde D1 y D2?
3. ¿Qué modificación adicional hay que introducir cuando el coste unitario de transporte entre un centro de distribución y un mercado cualesquiera de todas las unidades que excedan de un mínimo m, y sólo de ellas, se reduce en una cantidad h?
4. ¿Qué nueva modificación experimenta el modelo si las unidades se transportan en camiones que pueden transportar un peso máximo P, tal que P no es múltiplo de p? El coste de cada camión es independiente del trayecto realizado y asciende a la cantidad C. Este coste se añade al coste unitario de transporte de las unidades.

PROBLEMA PROGRAMACIÓN DE RUTAS

Un comerciante dispone de 1000 kg de una mercancía. Tiene 4 posibles mercados para vender esa mercancía, con distinto beneficio unitario en cada uno de ellos. Sin embargo para acceder a cada mercado tiene una única posible ruta, con un coste por cada viaje entre mercados. En la siguiente figura se muestra el esquema de las posibles rutas y el beneficio unitario en cada mercado:

A: origen	B	C	D
3 €	4 €	5.5 €	6.5 €

Se pide formular un problema de programación lineal en los siguientes supuestos:

1. En cada conexión hay un coste unitario de transporte: A-B: 0.75 €, B-C: 1 €, C-D: 0.75 €. Para este apartado, dar además la solución óptima.
2. Además del coste variable, en cada conexión hay un coste fijo de transporte: A-B: 300 €, B-C: 400 €, C-D: 250 €
3. Además del coste variable, el coste fijo de transporte depende de la cantidad transportada, siendo los valores anteriores si se transportan por la conexión menos de 300 kg, y de 100 € menos si se transportan 300 o más kg.

4. Además del coste fijo del apartado anterior, el coste variable es función de lo transportado, siendo los valores del primer apartado si son menos de 300 kg, o 0.25 € menos si se transportan 300 o más kg.

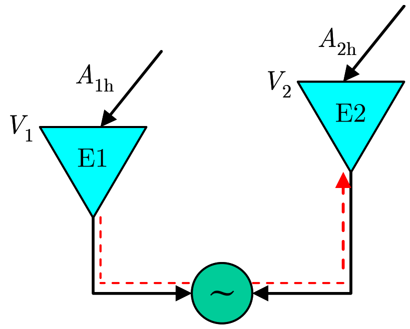
PROBLEMA CENTRAL HIDROELÉCTRICA

Una central hidráulica se abastece de dos embalses, E1 y E2. La central en cada hora h del día puede estar turbinando de un embalse (flecha continua de la izquierda), del otro embalse (flecha continua de la derecha), de ambos o no hacer nada. Los embalses tienen una capacidad máxima V_1 y V_2 (en hm^3) y reciben unas aportaciones horarias A_{1h} y A_{2h} (en hm^3) del río correspondiente. La central tiene una producción máxima P (en MW) si está turbinando. La producción desde cada embalse tiene un rendimiento asociado CE_1 y CE_2 (en MWh/hm^3 que se supone constante). Los embalses pueden verter energía almacenada si fuera necesario. Además, los embalses tienen un volumen inicial y final del día VI_1, VI_2, VF_1 y VF_2 (en hm^3) que se deben respetar. Se dispone además de un perfil de precios horarios de la energía (en €/MWh).

a) Definir las variables, las restricciones y la función objetivo del problema de optimización lineal que maximiza la remuneración diaria de su producción de energía y venta a los precios conocidos.

b) Suponer ahora que no se puede turbinar simultáneamente de los dos embalses y además existe una limitación técnica por la cual el cambio para turbinar de un embalse a turbinar del otro (cierre y apertura de las válvulas correspondientes) requiere 2 horas de operación para hacer la maniobra y en ese intervalo no se puede turbinar. Plantear las modificaciones a introducir en el nuevo problema de optimización.

c) Suponer que se puede bombear del embalse E1 al E2 (flecha discontinua) con un rendimiento conocido de CB_{12} (expresado en MWh/hm^3) y un consumo máximo como bomba B (en MW). Plantear las modificaciones a introducir en el nuevo problema de optimización.



1.1.17 Resultados de la biblioteca de problemas

RESULTADO DEL PROBLEMA AYUDA EN EMERGENCIAS

$$
\begin{aligned}
&\max_{x_{ij}} \sum_{i,j} a_{ij} x_{ij} \\
&\sum_j x_{ij} \le a_i \quad \forall i \\
&\sum_i x_{ij} \le v_j \quad \forall j \\
&x_{ij} \ge 0
\end{aligned}
$$

El avión A1 hace 50 viajes a la aldea V1, el avión A2 hace 40 a la aldea V3 y 20 a la V5 y el avión A3 hace 50 a la aldea V2 y 40 a la V4. La cantidad total de alimentos repartidos es 1870.

RESULTADO DEL PROBLEMA CONSTRUCCIÓN DE ALMACENES

$$
\begin{aligned}
&\min_{x_{ij}, y_i} \sum_{i,j} v_{ij} x_{ij} + \sum_i f_i y_i \\
&\sum_j x_{ij} \le c_i y_i \qquad \forall i \\
&\sum_i x_{ij} \ge d_j \qquad \forall j \\
&x_{ij} \ge 0, y_i \in \{0,1\}
\end{aligned}
$$

Construir los almacenes 1 y 3 y servir 3000 unidades del almacén 1 al cliente 1 y 1000 unidades al cliente 2 y del almacén 3 4000 unidades al cliente 2.

RESULTADO DEL PROBLEMA TRANSPORTE DE ELECTRODOMÉSTICOS

$$\min_{x_{ijk}} \sum_{i,j,k} c_{ij} x_{ijk}$$

$$\sum_j x_{ijk} \le a_{ik} \qquad \forall i,k$$

$$\sum_i x_{ijk} \ge b_{jk} \qquad \forall j,k$$

$$\sum_k x_{ijk} \le t_{ij} \qquad \forall i,j$$

$$x_{ijk} \ge 0$$

Frigoríficos	Barcelona	A Coruña	Valencia
Alicante	1500	0	3500
Huelva	2500	5000	500

Lavadoras	Barcelona	A Coruña	Valencia
Alicante	3000	0	4000
Huelva	0	3000	0

El coste total de transporte es de 176000 u.m.

RESULTADO DEL PROBLEMA LOGÍSTICA

$$\max_{x_{ijk}} \sum_{i,j,k} v_{jk} x_{ijk} - \sum_{i,j,k} f_{ik} x_{ijk} - \sum_{i,j,k} c_{ij} x_{ijk}$$

$$\sum_{j,k} t_{ik} x_{ijk} \le h_i \qquad \forall i$$

$$\sum_i x_{ijk} \ge b_{jk} \qquad \forall j,k$$

$$-500 \le \sum_j x_{ij1} - \sum_j x_{ij2} \le 500 \quad \forall i$$

$$x_{ijk} \ge 0$$

A1	D1	D2	D3
F1	600	0	500
F2	0	840	0

A1	D1	D2	D3
F1	0	0	1200
F2	700	500	0

El beneficio neto es de 29000.

RESULTADO DEL PROBLEMA GESTIÓN DE AUTOBUSES

$$
\begin{aligned}
&\min x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \\
&x_1 + x_6 \geq 4 \\
&x_1 + x_2 \geq 8 \\
&x_2 + x_3 \geq 10 \\
&x_3 + x_4 \geq 7 \\
&x_4 + x_5 \geq 12 \\
&x_5 + x_6 \geq 4 \\
&x_5 + x_6 \leq 4 + 12\delta \\
&x_1 + x_6 \geq 5\delta \\
&x_i \in \mathbb{Z}^+, \delta \in \{0,1\}
\end{aligned}
$$

El número mínimo de autobuses es de 26.

RESULTADO DEL PROBLEMA ADQUISICIÓN DE CAMIONES

$$
\begin{aligned}
&\min(0.30x + 0.25y)800 \\
&x \leq 10 \\
&y \leq 5 \\
&40x + 30y \geq 400 \\
&2(10 - x) \geq 5 - y \\
&x, y \in \mathbb{Z}^+
\end{aligned}
$$

RESULTADO DEL PROBLEMA PLANIFICACIÓN DEL METRO

$$\min 4x_1 + 4x_2 + 4x_3 + 4x_4 + 3x_5 + 4x_6 + 3x_7 + 3x_8$$
$$x_1 + x_2 + x_6 \ge 1$$
$$x_1 + x_3 + x_5 + x_8 \ge 2$$
$$x_1 + x_2 + x_3 + x_7 \ge 1$$
$$x_1 + x_3 + x_6 \ge 1$$
$$x_2 + x_4 + x_6 \ge 1$$
$$x_3 + x_8 \ge 1$$
$$x_2 + x_4 + x_5 \ge 1$$
$$x_5 + x_7 \ge 1$$
$$x_4 + x_7 \ge 1$$
$$x_4 + x_6 + x_8 \ge 1$$
$$\left.\begin{array}{r} x_2 + \delta \ge 1 \\ x_5 + x_7 - 2\delta \ge 0 \end{array}\right\} \text{ o bien } \left\{ x_5 + x_7 \ge 2 - 2\,x_2 \right.$$
$$x_i, \delta \in \{0,1\}$$

RESULTADO DEL PROBLEMA OFICINA DE CORREOS

$$\min \sum_{i=1}^{7} x_i$$
$$\begin{array}{rcl} x_1 + x_4 + x_5 + x_6 + x_7 & \ge & 17 \\ x_1 + x_2 + x_5 + x_6 + x_7 & \ge & 13 \\ x_1 + x_2 + x_3 + x_6 + x_7 & \ge & 15 \\ x_1 + x_2 + x_3 + x_4 + x_7 & \ge & 19 \\ x_1 + x_2 + x_3 + x_4 + x_5 & \ge & 14 \\ x_2 + x_3 + x_4 + x_5 + x_6 & \ge & 16 \\ x_3 + x_4 + x_5 + x_6 + x_7 & \ge & 13 \end{array}$$
$$x_i \in \mathbb{Z}^+$$

RESULTADO DEL PROBLEMA ABASTECIMIENTO

$$
\begin{aligned}
&\min 40x_{s1} + 50x_{s2} + 60x_{13} + 70x_{1t} + 40x_{23} + 70x_{2t} + 60x_{3t} + \\
&+100000y_{s1} + 200000y_{s2} + 80000y_{13} + \\
&+10000y_{1t} + 200000y_{23} + 200000y_{2t} + 150000y_{3t} \\
&x_{s1} + x_{s2} = 180 \\
&x_{s1} = x_{13} + x_{1t} \\
&x_{s2} = x_{23} + x_{2t} \\
&x_{13} + x_{23} = x_{3t} \\
&x_{1t} + x_{2t} + x_{3t} = 180 \\
&x_{s1} \le 100y_{s1} \\
&x_{s2} \le 200y_{s2} \\
&x_{13} \le 50y_{13} \\
&x_{1t} \le 30y_{1t} \\
&x_{23} \le 20y_{23} \\
&x_{2t} \le 100y_{2t} \\
&x_{3t} \le 60y_{3t} \\
&y_{2t} \ge y_{s2} \\
&x_{ij} \ge 0, y_{ij} \in \{0,1\} \quad \forall i, j
\end{aligned}
$$

Se eligen las conexiones (s–1), (s–2), (1–3), (1–t), (2–t) y (3–t) y pasa un flujo de 80, 100, 50, 30, 100 y 50 respectivamente. El coste total es 853300.

RESULTADO DEL PROBLEMA ADQUISICIÓN DE MÁQUINAS TROQUELADORAS

$$
\begin{aligned}
&\min(2x + 1.75y + 1.5z + 0.01\cdot 0.01\cdot 20x + 0.01\cdot 0.05\cdot 15y + 0.01\cdot 0.01\cdot 10z)8 \\
&20\cdot 8x + 15\cdot 8z + 10\cdot 8z \ge 3500 \\
&x \le 8 \\
&y \le 10 \\
&z \le 20 \\
&x, y, z \in \mathbb{Z}^+
\end{aligned}
$$

RESULTADO DEL PROBLEMA SECUENCIACIÓN DE TRABAJOS EN UNA MÁQUINA

Denominamos d_j al tiempo de proceso del trabajo j y r_j a la fecha de entrega del trabajo j.

Definimos las variables del problema como

$$x_{ij} = \begin{cases} 1 & \text{si el trabajo } j \text{ se hace en la posición } i \\ 0 & \text{en otro caso} \end{cases}$$

La función objetivo será la minimización de la demora media

$$\min \frac{1}{4} \sum_i p_i$$

sujeto a estas restricciones.

Cada trabajo se hace una vez

$$\sum_i x_{ij} = 1 \quad \forall j$$

En cada posición sólo un trabajo

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

Para cada posición i se acaba un trabajo en ella y su fecha de entrega es $\sum_j r_j x_{ij}$.

Por otra parte, el trabajo j que acaba en esa posición acaba en el instante $\sum_j d_j \sum_{k \le i} x_{kj}$. Las variables n_i y p_i, cuentan si acaba antes de tiempo (adelantado) o después (retrasado). Por eso p_i, que es la demora, es la que aparece en la función objetivo

$$\sum_j d_j \sum_{k \le i} x_{kj} + n_i - p_i = \sum_j r_j x_{ij} \quad \forall i$$

$$n_i, p_i \ge 0 \quad x_{ij} \in \{0,1\}$$

RESULTADO DEL PROBLEMA PRODUCCIÓN II

$$
\begin{aligned}
&\max 29y_1 + 69y_2 - 6y_1 - 18y_2 - 10y_1 - 20y_2 \\
&y_1 + 2y_2 \le 90 \\
&2y_1 + 6y_2 \le 300 \\
&y_2 \le 40 \\
&15u_1 \le y_1 \le 30u_1 \\
&10u_2 \le y_2 \le 20u_2 \\
&y_1, y_2 \ge 0, u_1, u_2 = \{0,1\}
\end{aligned}
$$

$$
\begin{aligned}
&\max \sum_k p_k y_k - e\sum_k t_k x_k - c\sum_k x_k \\
&\sum_k x_k \le \overline{x} \\
&\sum_k t_k x_k \le \bar{t} \\
&y_k = a_k x_k, \forall k \\
&u_k \underline{y}_k \le y_k \le u_k \overline{y}_k, \forall k \\
&x_k, y_k \ge 0, u_k = \{0,1\}, \forall k
\end{aligned}
$$

RESULTADO DEL PROBLEMA PRODUCCIÓN E INVENTARIO

$$
\begin{aligned}
&\min \sum_j \left(c\, i_j + c'_j\, p_j \right) \\
&i_j + p_j - d_j = i_{j+1} \\
&p_j \le \overline{p}_j \\
&i_j \le \overline{i}_j \\
&i_1 = 250, i_5 = 100 \\
&i_j, p_j \ge 0
\end{aligned}
$$

RESULTADO DEL PROBLEMA MISIÓN PACÍFICA

$$\max 2x_{ef}+5x_{ei}+4x_{eg}+3x_{ep}+4x_{fi}+4x_{fg}+2x_{fp}+5x_{ig}+4x_{ip}+3x_{gp}$$
$$x_{ef}+x_{ei}+x_{eg}+x_{ep}\le 1$$
$$x_{ef}+x_{fi}+x_{fg}+x_{fp}\le 1$$
$$x_{ei}+x_{fi}+x_{ig}+x_{ip}\le 1$$
$$x_{eg}+x_{fg}+x_{ig}+x_{gp}\le 1$$
$$x_{ep}+x_{fp}+x_{ip}+x_{gp}\le 1$$
$$x_{ef}+x_{ei}+x_{eg}+x_{ep}+x_{fi}+x_{fg}+x_{fp}+x_{ig}+x_{ip}+x_{gp}=2$$
$$x_{ei}\le x_{fg}$$
$$x_{ef},x_{ei},x_{eg},x_{ep},x_{fi},x_{fg},x_{fp},x_{ig},x_{ip},x_{gp}\in\{0,1\}$$

RESULTADO DEL PROBLEMA MEZCLA DE CRUDO

$$\max 7000(x_{11}+x_{21})+6000(x_{12}+x_{22})-4500(x_{11}+x_{12})-3500(x_{21}+x_{22})$$
$$-400(x_{11}+x_{12}+x_{21}+x_{22})-y_1-y_2$$
$$x_{11}+x_{12}\le 5000$$
$$x_{21}+x_{22}\le 5000$$
$$x_{11}+x_{12}+x_{21}+x_{22}\le 9000$$
$$12x_{11}+6x_{21}\ge 10(x_{11}+x_{21})$$
$$12x_{12}+6x_{22}\ge 8(x_{12}+x_{22})$$
$$0.5x_{11}+2x_{21}\le x_{11}+x_{21}$$
$$0.5x_{12}+2x_{22}\le 2(x_{12}+x_{22})$$
$$x_{11}+x_{21}=3000+0.1y_1$$
$$x_{12}+x_{22}=2000+0.1y_2$$
$$x_{11},x_{12},x_{21},x_{22},y_1,y_2\ge 0$$

RESULTADO DEL PROBLEMA PRODUCCIÓN VI

a)

$$\min \sum_j \left[calmAl_j + \sum_i \left(cv_{ij} P_{ij} + cf_i \lambda_{ij} + carr_i Arr_{ij} \right) \right]$$
$$\sum_i P_{ij} + Al_{j-1} = dem_j + Al_j \quad \forall j \quad (Al_0 = 0)$$
$$P_{ij} \le pm_i \lambda_{ij} \quad \forall i, j$$
$$Arr_{ij} - Par_{ij} = \lambda_{ij} - \lambda_{ij-1} \quad \forall i, \forall j \ge 2$$
$$\lambda_{i0} = 1$$
$$P_{ij}, Al_j \ge 0, \lambda_{ij} \in \{0,1\}$$
$$Arr_{ij} \in \{0,1\} \quad \text{ó} \quad Arr_{ij} \ge 0$$

b)

$$\lambda_{ij-1} + \lambda_{ij+1} - \lambda_{ij} \le 1 \quad \forall i, 2 \le j \le n-1$$

o bien

$$\lambda_{ij} + \lambda_{ij+1} \le 2 - 2Par_{ij} \quad \forall i, 1 \le j \le n-1$$

RESULTADO DEL PROBLEMA TRANSPORTE POR FERROCARRIL

a)

$$\max z = \sum_i (b_i - c_i) x_i$$
$$\sum_i v_i x_i \le V$$
$$\sum_i p_i x_i \le P$$
$$0 \le x_i \le mx_i \quad \forall i$$

Se deben dar las cantidades de las mercancías a transportar, el valor de la variable dual de la primera restricción para dar el valor de cada unidad de volumen, el valor de la variable dual de la segunda restricción para dar el valor de cada unidad de peso y a cada propietario del que no se transporte nada el coste reducido de su variable correspondiente que sería en lo que tendría que aumentar su oferta para que se transporte algo de su mercancía.

b)

$$
\begin{aligned}
&\max z = \sum_i (a_i - g_i x_i - c_i) x_i \\
&\sum_i v_i x_i \le V \\
&\sum_i p_i x_i \le P \\
&0 \le x_i \le mx_i \quad \forall i
\end{aligned}
$$

c)

$$
\begin{aligned}
&\max z = \sum_{i,j} (b_i - c_i) x_{ij} - cvag \sum_j y_j \\
&\sum_i v_i x_{ij} \le \bar{V} y_j \quad \forall j \\
&\sum_i p_i x_{ij} \le \bar{P} y_j \quad \forall j \\
&\sum_j x_{ij} \le mx_i \quad \forall i \\
&x_{ij} \ge 0, \quad y_j \in \{0,1\}
\end{aligned}
$$

o bien

$$
\begin{aligned}
&\max z = \sum_i (b_i - c_i) x_i - cvag \cdot y \\
&\sum_i v_i x_i \le \bar{V} y \\
&\sum_i p_i x_i \le \bar{P} y \\
&0 \le x_i \le mx_i \quad \forall i \\
&y \in \mathbb{Z}^+
\end{aligned}
$$

RESULTADO DEL PROBLEMA EDICIÓN DE CDS

$$
\begin{aligned}
&\max \sum_j y_j \\
&\sum_j x_{ij} = 1 \quad \forall i \\
&\sum_i tam_i x_{ij} \le 700 y_j \quad \forall j \\
&\sum_i indi_i x_{ij} \ge 45 y_j \quad \forall j \\
&x_{1j} + x_{2j} + x_{3j} \le 1 \quad \forall j \\
&x_{ij}, y_j \in \{0,1\}
\end{aligned}
$$

RESULTADO DEL PROBLEMA VUELOS CHARTER

$$\begin{array}{lcl} \min \sum_{i,j} C_{ij} x_{ij} + \sum_{i} 6 y_i & & \min \sum_{i,j} C_{ij} x_{ij} + 6 \sum_{i,j} x_{ij} \\ \sum_{i} CAP_i \sum_{j} j \cdot x_{ij} \geq D & & \sum_{i} CAP_i \sum_{j} j \cdot x_{ij} \geq D \\ \sum_{j} x_{ij} \leq y_i \quad \forall i & \text{o bien} & \sum_{j} x_{ij} \leq 1 \quad \forall i \\ x_{ij}, y_i \in \{0,1\} & & x_{ij} \in \{0,1\} \end{array}$$

RESULTADO DEL PROBLEMA PROVEEDORES

1.

$$\min \sum_{i,j} \left(c_{ij} X_{ij} + ccont_{ij} Y_{ij} \right)$$
$$\sum_{j} X_{ij} \leq a_i \quad \forall i$$
$$\sum_{i} X_{ij} = d_j \quad \forall j$$
$$X_{ij} \leq MY_{ij} \quad \forall i, j \ (M \text{ cota suf. grande, por ejemplo, } M = \max(d_j) = 135)$$
$$X_{ij} \geq 0, Y_{ij} \in \{0,1\}$$

2.

$$\min \sum_{i,j} \left(c_{ij} X_{ij}^{+} + (c_{ij} + penal_i) X_{ij}^{-} + ccont_{ij} Y_{ij} \right)$$
$$\sum_{j} X_{ij} \leq a_i \quad \forall i$$
$$\sum_{i} X_{ij} = d_j \quad \forall j$$
$$X_{ij} \leq MY_{ij} \quad \forall i, j \ (M \text{ cota suf. grande, por ejemplo, } M = \max(d_j) = 135)$$
$$X_{ij} = X_{ij}^{+} + X_{ij}^{-} \quad \forall i, j$$
$$\sum_{j} X_{ij} \geq k_i (1 - Z_i) \quad \forall i$$
$$X_{ij}^{-} \leq m_{ij} Z_i \quad \forall i, j$$
$$X_{ij}^{+} \leq m_{ij} (1 - Z_i) \quad \forall i, j \ (m_{ij} \text{ cota suf. grande, por ejemplo, } d_j)$$
$$X_{ij}, X_{ij}^{-}, X_{ij}^{+} \geq 0, Y_{ij}, Z_{ij} \in \{0,1\}$$

RESULTADO DEL PROBLEMA COMPOSICIÓN DE PIENSO

a)

$$\min \sum_i c_i X_i + \sum_j p_j (D_j + E_j)$$
$$\sum_i a_{ij} X_i \geq \min_j \quad \forall j$$
$$\sum_i a_{ij} X_i \leq \max_j \quad \forall j$$
$$\sum_i a_{ij} X_i + D_j - E_j = opt_j \quad \forall j$$
$$X_i, D_j, E_j \geq 0$$

b) Dos modelos alternativos:

$$\min c_1 X_1^- + c_2 X_2 + (c_1 + 0.05) X_1^+ + \sum_j p_j (D_j + E_j)$$
$$\sum_i a_{ij} X_i \geq \min_j \quad \forall j$$
$$\sum_i a_{ij} X_i \leq \max_j \quad \forall j$$
$$\sum_i a_{ij} X_i + D_j - E_j = opt_j \quad \forall j$$
$$X_1 = X_1^- + X_1^+$$
$$X_1 - 2X_2 \leq M\delta$$
$$X_1^- \leq M(1-\delta)$$
$$X_1^+ \leq M\delta$$
$$X_i, X_1^-, X_1^+, D_j, E_j \geq 0 \quad \delta \in \{0,1\}$$

$$\min \sum_i c_i X_i + Extra + \sum_j p_j (D_j + E_j)$$
$$\sum_i a_{ij} X_i \geq \min_j \quad \forall j$$
$$\sum_i a_{ij} X_i \leq \max_j \quad \forall j$$
$$\sum_i a_{ij} X_i + D_j - E_j = opt_j \quad \forall j$$
$$X_1 - 2X_2 \leq M\delta$$
$$Extra \geq 0.05X_1 - M(1-\delta)$$
$$X_i, D_j, E_j, Extra \geq 0 \quad \delta \in \{0,1\}$$

RESULTADO DEL PROBLEMA EXPLOTACIONES GANADERAS

a) i explotaciones ganaderas, granjas (G1 y G2)

j mataderos (M1, M2 y M3)

k tipo de animal (vaca y cerdo)

N_{ik} número de vehículos para transportar animales tipo k en explotación i

$DISP_{ik}$ disponibilidad de animales k en las explotaciones ganaderas i

DEM_{jk} demanda en cada matadero j de cada animal k

$COSTE_{ijk}$ coste unitario de transporte de explotaciones ganaderas i a mataderos j para cada animal k

CAP_k capacidad de cada tipo de vehículo k

$FIJO_k$ coste fijo de uso del camión para transportar animales tipo k

VOL_k volumen unitario de cada tipo de animal k

Transformar un dato: $ANIM_k = \lfloor CAP_k / VOL_k \rfloor$: número animales que caben en un camión de tipo k

Podrían ser dos problemas separados de transporte (nada los liga)

Variables:

x_{ijk} animales tipo k transportados entre la explotación i y el matadero j

z_{ijk} vehículos para animales tipo k usados entre explotación i y matadero j

$$\min \quad \sum_k \left(\sum_{i,j} \left(COSTE_{ijk} x_{ijk} + FIJO_k z_{ijk} \right) \right)$$

$$\sum_j x_{ijk} \leq DISP_{ik} \qquad \forall i,k$$

$$\sum_i x_{ijk} \geq DEM_{jk} \quad \forall j,k$$

$$\sum_j z_{ijk} \leq N_{ik} \qquad \forall i,k$$

$$x_{ijk} \leq ANIM_k z_{ijk} \qquad \forall i,j,k \quad Cuasicorrecto: VOL_k x_{ijk} \leq CAP_k z_{ijk} \quad \forall i,j,k$$

$$x_{ijk}, z_{ijk} \in \mathbb{Z}^+$$

b)

i explotaciones ganaderas, granjas (G1 y G2)

j mataderos (M1, M2 y M3)

k tipo de animal (vaca y cerdo)

n vehículo

N_{ik} número de vehículos para transportar animales tipo k en explotación i

$DISP_{ik}$ disponibilidad de animales k en las explotaciones ganaderas i

DEM_{jk} demanda en cada matadero j de cada animal k

$COSTE_{ijk}$ coste unitario de transporte de explotaciones ganaderas i a mataderos j para cada animal k

CAP_k capacidad de cada tipo de vehículo k

$FIJO_k$ coste fijo de uso del camión para transportar animales tipo k

VOL_k volumen unitario de cada tipo de animal k

$INCCOSTE_k$ incremento de coste unitario de transporte para cada animal k

Variables

x_{ijk}^{kn} cantidad de animales de tipo k transportados entre la explotación ganadera i y el matadero j transportados en el vehículo número n del tipo k

$y_{ijk}^{k'n}$ cantidad de animales de tipo k transportados entre la explotación ganadera i y el matadero j transportados en el vehículo número n del tipo k' (en este caso k y k' son diferentes)

v_{ij}^{nk} uso o no del vehículo n diseñado para el transporte del animal k para transportar animales entre la explotación ganadera i y el matadero j

$$\min \sum_{knij} \left(COSTE_{ijk} x_{ijk}^{kn} + (COSTE_{ijk} + INCCOSTE_k) y_{ijk}^{k'n} + FIJO_k v_{ij}^{kn} \right)$$

$$VOL_k x_{ijk}^{kn} + VOL_{k'} y_{ijk'}^{kn} \le CAP_k v_{ij}^{kn} \quad \forall knij$$

$$\sum_{ni} (x_{ijk}^{kn} + y_{ijk}^{k'n}) \ge DEM_{jk} \quad \forall jk$$

$$\sum_{nj} (x_{ijk}^{kn} + y_{ijk}^{k'n}) \le DISP_{ik} \quad \forall ik$$

$$\sum_{j} v_{ij}^{kn} \le 1 \quad \forall ikn$$

$$x_{ijk}^{kn}, y_{ijk}^{kn} \in \mathbb{Z}^+$$

$$v_{ij}^{kn} \in \{0,1\}$$

La solución óptima con una tolerancia relativa del 3 % tiene un coste de 4943 €.

		Vehículos de vacas				Vehículos de cerdos	
	M1	M2	M3		M1	M2	M3
G1			(8,0) (8,0) (6,0)		(0,23) (**1**,22)	(0,26)	(0,24)
G2	(8,0) (8,0)	(6,**7**)				(0,26) (0,26)	(0,26) (0,26)

RESULTADO DEL PROBLEMA CORTE DE BOBINAS

Definimos i como el índice de cada pedido o bobina hija, I es el número total de pedidos, j como el índice de las bobinas madre, J es el número máximo de bobinas madre posibles (una cota superior es el número de pedidos). A_i es la anchura de cada pedido, F_i es la fecha de entrega de cada pedido y $\bar{A}$ es la anchura de la bobina madre.

Definimos la variable

$$X_{ij} = \begin{cases} 1 & \text{si el pedido } i \text{ se obtiene de la bobina madre } j \\ 0 & \text{en otro caso} \end{cases},$$

$$Y_j = \begin{cases} 1 & \text{si se usa la bobina madre } j \\ 0 & \text{en otro caso} \end{cases}, \ D_j \text{ el desperdicio de la bobina } j.$$

La función objetivo de minimización del número total de bobinas se expresa como

$$\min \sum_{j=1}^{J} Y_j$$

Satisfacción de cada pedido i

$$\sum_{j=1}^{J} X_{ij} = 1 \quad \forall i$$

Los pedidos de una bobina no pueden exceder su ancho

$$\sum_{i=1}^{I} A_i X_{ij} \le \bar{A} Y_j \quad \forall j$$

La función objetivo de minimización del desperdicio de las bobinas madre se expresa como

$$\min \sum_{j=1}^{J} D_j$$

El desperdicio se calcula cambiando la restricción anterior

$$\sum_{i=1}^{I} A_i X_{ij} + D_j = \bar{A} Y_j \quad \forall j$$

La función objetivo de minimización del retraso en la fecha de entrega de los pedidos se expresa como

$$\min \sum_{i=1}^{I} RT_i$$

siendo AD_i y RT_i el adelanto o retraso en la fecha de entrega del pedido i

El retraso en la fecha de entrega de cada pedido se calcula en esta ecuación

$\sum_{j=1}^{J} 6jX_{ij} + AD_i - RT_i = F_i \quad \forall i$ o bien $\sum_{j=1}^{J} 6jX_{ij} - RT_i \le F_i \quad \forall i$

RESULTADO DEL PROBLEMA EMPRESA DISTRIBUIDORA

1. Variables

X_{ij} : número de unidades enviadas de i a j

Y_{ij} : número de unidades enviadas de j a k

$$\begin{aligned}
&\min \sum_{ij} c_{ij} X_{ij} + \sum_{jk} w_{jk} Y_{jk} \\
&\sum_{j} X_{ij} \le e_i \quad \forall i \\
&\sum_{j} Y_{jk} = d_k \quad \forall k \\
&\sum_{i} X_{ij} = \sum_{k} Y_{jk} \quad \forall j \\
&X_{ij}, Y_{jk} \ge 0 \quad X_{ij}, Y_{jk} \in \mathbb{Z}??
\end{aligned}$$

2. Nueva variable: $Z = \begin{cases} 1 & \text{le provee distribuidor 1} \\ 0 & \text{le provee distribuidor 2} \end{cases}$

   Añadir al modelo las restricciones:

$$\begin{aligned}
&Y_{11} \le MZ \\
&Y_{21} \le M(1-Z) \\
&Z \in \{0,1\}
\end{aligned}$$

   siendo, por ejemplo, $M = d_1$

3. Entiendo por el enunciado que las m primeras siempre cuestan lo mismo y las que se pasen son las que tienen descuento (no todas una vez se pasan).

   Nuevas variables

   N_{jk}, P_{jk} : exceso y defecto sobre el valor m

$$\delta_{jk} = \begin{cases} 1 & \text{si } N_{jk} > 0 \\ 0 & \text{si } P_{jk} > 0 \end{cases}$$

Modificar función objetivo añadiendo el término $-\sum_{jk} hN_{jk}$

Añadir las restricciones:

$$\begin{aligned} &Y_{jk} = m + N_{jk} - P_{jk} \quad \forall j,k \\ &N_{jk} \le M\delta_{jk} \quad \forall j,k \\ &P_{jk} \le M(1-\delta_{jk}) \quad \forall j,k \end{aligned}$$

(Estas dos últimas son necesarias pues si no pueden ser distintas de 0 ambas a la vez, para un mismo valor de Y y lograr menor coste).

4. Sea $cap = \left\lfloor \frac{P}{p} \right\rfloor$ número de artículos que caben en un camión. Sean las variables:

T_{ij}: número de camiones enviados de i a j

S_{jk} T_{ij}: número de camiones enviados de j a k

Modificar la función objetivo añadiendo el término $+C\left(\sum_{ij} T_{ij} + \sum_{jk} S_{jk}\right)$

Añadir las restricciones:

$$\begin{aligned} &X_{ij} \le cap\, T_{ij} \quad \forall i,j \\ &Y_{jk} \le cap\, S_{jk} \quad \forall j,k \end{aligned}$$

RESULTADO DEL PROBLEMA PROGRAMACIÓN DE RUTAS

1.

$$\begin{aligned} &\max\ f_1 = 3A + 4B + 5.5C + 6.5D - 0.75(B + C + D) - 1(C + D) - 0.75D \\ &A + B + C + D \le 1000 \\ &A, B, C, D \ge 0 \end{aligned}$$

Solución óptima: coeficiente en objetivo A: 3, B: 3.25, C: 3.75, D: 4. Todos aportan igual en restricciones, así que vender todo en D.

2.

$$\begin{aligned}
&\max\ f_2 = f_1 - 300\delta_1 - 400\delta_2 - 250\delta_3 \\
&A + B + C + D \le 1000 \\
&B + C + D \le 1000\delta_1 \qquad C + D \le 1000\delta_2 \qquad D \le 1000\delta_3 \\
&A, B, C, D \ge 0 \quad \delta_i \in \{0,1\}
\end{aligned}$$

3.

$$\begin{aligned}
&\max\ f_3 = f_2 + 100(Y_1 + Y_2 + Y_3) \\
&A + B + C + D \le 1000 \\
&B + C + D \le 1000\delta_1 \qquad C + D \le 1000\delta_2 \qquad D \le 1000\delta_3 \\
&B + C + D \ge 300Y_1 \qquad C + D \ge 300Y_2 \qquad D \ge 300Y_3 \\
&A, B, C, D \ge 0 \quad \delta_i \in \{0,1\} \quad Y_i \in \{0,1\}
\end{aligned}$$

4.

$$\begin{aligned}
&\max\ f_4 = f_3 + E_1 + E_2 + E_3 \\
&A + B + C + D \le 1000 \\
&B + C + D \le 1000\delta_1 \qquad C + D \le 1000\delta_2 \qquad D \le 1000\delta_3 \\
&B + C + D \ge 300Y_1 \qquad C + D \ge 300Y_2 \qquad D \ge 300Y_3 \\
&E_1 \le 0.25(B + C + D) \quad E_2 \le 0.25(C + D) \quad E_3 \le 0.25D \\
&E_1 \le 250Y_1 \qquad E_2 \le 250Y_2 \qquad E_3 \le 250Y_3 \\
&A, B, C, D, E_i \ge 0 \quad \delta_i \in \{0,1\} \quad Y_i \in \{0,1\}
\end{aligned}$$

RESULTADO DEL PROBLEMA CENTRAL HIDROELÉCTRICA

a)

Índices

e : embalses, E1 y E2.

h : horas, 1 a 24.

Parámetros

V_1, V_2 : volumen máximo [hm^3].

P : producción máxima [MW].

A_{1h}, A_{2h} : aportaciones horarias [hm^3].

CE_1, CE_2: coeficiente energético [MWh/hm^3].

VI_1, VI_2, VF_1, VF_2: volumen inicial y final del día [hm^3].

PR_h: precio en hora h [€/MWh].

Variables

p_{eh}: producción de la central desde el embalse e en la hora h [MW].

v_{eh}: volumen del embalse en la hora h [hm^3].

Función objetivo

Maximización de la remuneración de la turbinación de la central

$$\max \sum_{eh} PR_h p_{eh}$$

Restricciones

Continuidad en el volumen de cada embalse

$$v_{eh-1} + A_{eh} - \frac{p_{eh}}{CE_e} \geq v_{eh} \quad \forall eh$$

$$v_{e0} = VI_e \quad \forall e$$
$$v_{e24} = VF_e \quad \forall e$$

Cotas a la producción del grupo

$$0 \leq p_{eh} \leq P \quad \forall eh$$
$$\sum_e p_{eh} \leq P \quad \forall h$$

Cotas al volumen de los embalses

$$0 \leq v_{eh} \leq V_e \quad \forall eh$$

b)

Se necesita crear una variable binaria que indique si se turbina desde un embalse o desde otro pero no de ambos simultáneamente.

t_{eh} : selección de turbinación desde el embalse e en la hora h {0,1}.

Restricción de no simultaneidad en la turbinación desde los embalses con un retraso de hasta dos horas

$$\begin{aligned} &t_{1h} + t_{2h} \le 1 \quad \forall h \\ &t_{1h} + t_{2h+1} \le 1 \quad \forall h \ \text{ y } \ t_{2h} + t_{1h+1} \le 1 \quad \forall h \\ &t_{1h} + t_{2h+2} \le 1 \quad \forall h \qquad t_{2h} + t_{1h+2} \le 1 \quad \forall h \end{aligned}$$

Relación entre producción y variable binaria de selección de embalse

$$0 \le p_{eh} \le Pt_{eh} \quad \forall eh$$

c)

Parámetros

CB_{12} : rendimiento el bombeo [MWh/hm^3].

B : consumo máximo como bomba [MW]

Variables

b_h : consumo de la central en la hora h [MW].

Función objetivo

$$\max \sum_{eh} PR_h \left(p_{eh} - b_h \right)$$

Restricciones

Continuidad en el volumen de cada embalse

$$v_{1h-1} + A_{1h} - \frac{p_{1h}}{CE_1} - \frac{b_h}{CB_{12}} \ge v_{1h} \quad \forall h$$

$$v_{2h-1} + A_{2h} - \frac{p_{2h}}{CE_2} + \frac{b_h}{CB_{12}} \ge v_{2h} \quad \forall h$$

Cotas al consumo del grupo

$$0 \le b_h \le B \quad \forall h$$

1.2 Codificación de problemas de optimización

1.2.1 Lenguajes de modelado

Las principales alternativas actuales para el desarrollo de modelos de optimización suelen ser, Sharda (1995):

- *Lenguajes de programación de propósito general* (C, C++, Java, Visual Basic, FORTRAN 90) que llaman a una biblioteca de optimización

  Tienen sentido cuando el tiempo de solución es crítico o el modelo es ejecutado con mucha frecuencia o cuando se necesitan interfaces a medida para la entrada de datos o salida de resultados o cuando el modelo tiene que ser integrado en otra aplicación o se necesitan algoritmos de optimización específicos. Además permiten la implantación del modelo en un entorno software o hardware especial. Como contrapartida requiere un tiempo de desarrollo muy elevado y, sobre todo, presenta una gran dificultad y consumo de recursos para el mantenimiento del código.

  Actualmente ya existen bibliotecas de componentes orientados a objetos (clases C++) dedicadas exclusivamente a optimización, por ejemplo, Concert de ILOG, LINDO API de LINDO Systems, OptiMax 2000 de Maximal Software, FLOPC++ de Universidade de Aveiro. Es necesario también mencionar la iniciativa de desarrollo de software abierto para investigación operativa denominada *Computational Infrastructure for Operations Research* (COIN-OR) (www.coin-or.org).

- *Lenguajes o entornos de cálculo numérico o simbólico* (hojas de cálculo, lenguajes para cálculo numérico intensivo, como MATLAB, o para cálculo simbólico, como Maple o Mathematica, etc.)

  Los optimizadores de las hojas de cálculo, por ser aplicaciones muy comunes y conocidas, pueden ser un vehículo eficaz de difusión de un modelo entre cierto tipo de usuarios y facilitan el manejo de datos que se encuentren ya en dicho formato [Ragsdale, 1998]. Como ventajas específicas se pueden mencionar: su facilidad de uso, su integración total con la hoja de cálculo, la familiaridad con el entorno que facilita la explicación del modelo y de sus resultados, así como la facilidad de presentación de resultados en gráficos. Sin embargo, no inducen una buena práctica de programación, presentan la dificultad de su desarrollo, verificación, validación, actualización, documentación y, en general, el mantenimiento del modelo y no permiten modelar problemas complejos o de gran tamaño [Gass, 1995]. Frontline Systems (www.solver.com) ha desarrollado optimizadotes para Microsoft Excel.

Los lenguajes de cálculo numérico o simbólico no son específicos de problemas de optimización pero facilitan la manipulación numérica o simbólica de matrices y vectores. También disponen de funciones de optimización.

Todas estas alternativas pueden ser utilizadas para desarrollo rápido de un prototipo o una demostración ya que presentan capacidades de presentación gráfica que pueden ser aprovechadas. Son difícilmente utilizables cuando se plantean problemas de optimización de tamaño medio o superior.

- *Lenguajes algebraicos de modelado*

Son las alternativas más complejas y potentes por su capacidad de indexación de las variables y ecuaciones, permiten cambiar sin dificultad las dimensiones del modelo, de forma natural separan datos de resultados. Desde el punto de vista del modelador permiten la detección de errores de consistencia en la definición y verificación del modelo. Desde el punto de vista del usuario simplifican drásticamente su mantenimiento. Entre los lenguajes de modelado más conocidos se pueden mencionar: GAMS (www.gams.com), AMPL (www.ampl.com) de origen estadounidense y MPL (www.maximalsoftware.com) y AIMMS (www.aimms.com) y XPRESS-MP (www.dashoptimization.com) de origen europeo, por citar algunos. De algunos de ellos se pueden descargar versiones de estudiante desde sus páginas web. GAMS es el más antiguo, pero con el conjunto de usuarios más amplio, quizá por eso con algunas limitaciones en sus capacidades de modelado. AMPL es más nuevo, muy potente para el modelado pero con un conjunto reducido de usuarios. MPL es otro lenguaje de modelado robusto, cuya versión de estudiante acompaña al libro [Hillier y Lieberman, 2002].

Existe una herramienta integrada denominada OPLStudio (www.ilog.com), en la que se dispone de un lenguaje de modelado (OPL) y varios optimizadores dependiendo del modelo propuesto. Está especialmente desarrollada para problemas de programación (*scheduling*) y planificación, aunque admite también cualquier modelo de optimización lineal y lineal entera mixta. Es una herramienta integrada ya que además del lenguaje de modelado, incluye sus propios optimizadores, Scheduler, Solver, CPLEX[2], estando los dos primeros basados en la programación de restricciones[3] y el último en programación matemática.

GAMS es el lenguaje más ampliamente difundido comercialmente con su propia lista de discusión de usuarios (gams-l@listserv.gmd.de) mientras que AMPL se está potenciando mucho en las universidades estadounidenses. Existe un proyecto denominado NEOS Server for Optimization (http://neos.mcs.anl.gov/neos/solvers/index.html) para el cálculo distribuido que permite el envío de problemas de optimización escritos en AMPL o GAMS a través de internet y éstos son resueltos por optimizadores específicos para el tipo de problema enviado en servidores de la red devolviendo los resultados de la optimización.

Existen libros específicos que describen sus características y que sirven como guías de usuario tanto para el lenguaje GAMS [Brooke, 1998], [McCarl, 1998], para AMPL, [Fourer, 2000], o para OPL [Van Hentenryck, 1999]. Incluso en España se ha publicado un libro de optimización que se apoya en GAMS para la presentación de ejemplos [Mocholí, 1996]. Los campos de aplicación de estos lenguajes son tan amplios como los de la optimización propiamente dicha. Abarcan desde la micro y macroeconomía, a la economía de la energía, a la planificación energética o eléctrica, a la ingeniería química o forestal, a la planificación del desarrollo económico o del comercio internacional, a la cobertura de riesgos financieros, a problemas de transporte y comunicaciones, a organización de la producción o fabricación o a la planificación de grandes proyectos. En el caso de la programación de restricciones ésta aparece especialmente en problemas combinatorios para modelar restricciones lógicas.

1.2.2 Lenguajes algebraicos de modelado

Los lenguajes algebraicos son lenguajes de alto nivel que han sido diseñados específicamente para el desarrollo e implantación de modelos de optimización de forma más directa para los programadores y más inteligible para los usuarios. En

[2] CPLEX es problablemente el mejor optimizador existente en la actualidad para problemas LP y MIP.

[3] Se denomina programación de restricciones a un tipo de programación lógica donde el dominio de las variables viene definido por relaciones lógicas y por restricciones.

consecuencia, el campo de actuación y utilidad de los modelos de optimización se ha ampliado tremendamente al utilizar estos lenguajes. Entre sus *características* y *ventajas* principales destacan las siguientes:

- Proporcionan una formulación sencilla de modelos grandes y complejos
- Facilitan sobremanera el desarrollo de prototipos
- Mejoran sustancialmente la productividad de los modeladores al permitir dedicar más tiempo al diseño, ejecución del modelo y análisis de los resultados y menos a la codificación del mismo
- Estructuran los buenos hábitos de modelado al exigir una representación concisa y exacta de los parámetros/variables y sus relaciones
- Recogen simultáneamente la estructura del modelo y su documentación
- Separan de manera natural los datos de la estructura del modelo y ésta de los algoritmos de solución
- La formulación del problema es independiente del tamaño. Permiten el uso de la estructura del modelo para diferentes casos[4]
- Los optimizadores pueden ser intercambiados sin dificultad, se pueden probar nuevos optimizadores, nuevos métodos o nuevas versiones
- Por ejemplo, en el lenguaje GAMS se encuentran entre otros disponibles los optimizadores CPLEX, OSL, XA y XPRESS para problemas LP y MIP, MINOS y CONOPT para problemas NLP, DICOPT para problemas MINLP y MILES y PATH para problemas MCP.
- Permiten la realización de cambios en el modelo de manera sencilla y segura, es decir, se puede afrontar un refinamiento continuo en la formulación del problema
- Cualquier tipo de problemas de programación lineal, no lineal, flujos en redes o mixta complementaria resulta muy fácil implantar su formulación

[4] Una manera habitual de desarrollar es utilizar una maqueta (caso ejemplo) para la depuración y verificación del modelo y una vez comprobada su validez utilizar el caso real a ser resuelto.

- Permiten la implantación de algoritmos avanzados, que incluyan varias llamadas al optimizador o procedimientos específicos para el problema (como por ejemplo los métodos de descomposición)
- Permiten la portabilidad de los modelos entre plataformas y sistemas operativos

Como *desventajas* principales se pueden mencionar las siguientes:

- No son adecuados para la resolución de problemas de pequeño tamaño por parte de usuarios esporádicos por la barrera de entrada que supone el aprendizaje de un nuevo lenguaje
- No pueden utilizarse para la resolución directa de problemas gigantescos cuya formulación completa incluso no se puede realizar (por ejemplo, a partir de 1 millón de restricciones y/o variables)
- En la ejecución se incluye un tiempo de creación del modelo y de interfaz con el optimizador que ralentiza la obtención de la solución, por lo tanto no es recomendable cuando el tiempo de ejecución es un factor crítico.

Las *tendencias* o características más actuales en el desarrollo de lenguajes algebraicos se mueven hacia:

- Interfaces de entrada y salida de datos más estrechamente relacionadas con bases de datos u hojas de cálculo
- El desarrollo de interfaces gráficas que faciliten al usuario la formulación visual y el entendimiento de problemas de optimización
- Interfaz con lenguajes de propósito general para la incorporación de funciones externas definidas por el usuario dentro de la optimización
- El avance en las capacidades de resolución directa de problemas estocásticos (con adición de características específicas en el propio lenguaje y uso de algoritmos de descomposición en el optimizador) o problemas no lineales complejos
- La posibilidad de ocultar el código fuente produciendo versiones ejecutables para usuarios finales
- La selección automática del método y optimizador

1.2.2.1 Referencias

Brooke, A., Kendrick, D., Meeraus, A. and Raman, R. (1998) *GAMS A User's Guide.* GAMS Development Co.

Fourer, R., Gay, D.M. and Kernighan, B.W. (2000) *AMPL: A Modeling Language for Mathematical Programming.* The Scientific Press. 2nd ed.

Ragsdale, C. T. (1998) *Spreadsheet modeling and decision analysis: a practical introduction to management science*. South-Western College. 2nd ed.

Gass, S.I., Hirshfeld, D.S. and Wasil, E.A. (1995) "Model World: The Spreadsheeting of OR/MS" *Interfaces* pp. 72-81. September-October.

McCarl, B.A. and Spreen, Th.H. (1998) *Applied Mathematical Programming using Algebraic Systems*. Technical Report.

Mocholí, M. y Sala, R. (1996) *Decisiones de optimización* Tirant lo Blanch. Valencia.

Sharda, R. and Rampal, G. (1995) "Algebraic Modeling Languages on PCs" *OR/MS Today* pp. 58-63. June.

Van Hentenryck, P. (1999) *The OPL Optimization Programming Language*. The MIT Press.

1.2.3 Modelado en GAMS

En este apartado se presentan varios ejemplos sencillos que permiten mostrar algunas de las características del lenguaje GAMS. Sin embargo, el manual de usuario contiene un capítulo tutorial y la referencia de todas las características del lenguaje.

El tiempo de ejecución de un modelo escrito en GAMS se puede descomponer en estos tres tipos principales[5]:

- tiempo de *creación*

  formulación del problema de optimización específico, es decir, creación de las variables y de las restricciones.

- tiempo de *interfaz*

---

[5] Esta clasificación del tiempo de ejecución de un modelo en tres componentes es relevante para los modelos escritos en GAMS. Quizá con otros lenguajes de modelado alguno de estos tiempos puede ser despreciable.

escritura del problema de optimización en disco para su lectura por el optimizador y viceversa.

- tiempo de optimización

  resolución del problema de optimización por parte del optimizador.

Además de éstos hay que añadir el tiempo de compilación del modelo. Sin embargo, este tiempo se da únicamente una vez al comienzo y habitualmente es despreciable frente al resto.

El valor e importancia de cada uno de estos tiempos se puede conocer con las opciones stepsum, que resume el consumo de tiempo entre llamadas al optimizador, y profile, que informa sobre el consumo de tiempo y memoria en cada instrucción del código. Antes de iniciar las acciones de mejora es necesario realizar un análisis de los consumos de tiempo del modelo y de cómo se reparten.

1.2.3.1 Ejemplo de transporte

Veamos a continuación un caso típico de un problema de optimización lineal clásico y cómo este problema se codifica en el lenguaje GAMS. En el apartado de modelado en programación lineal entera mixta se presenta formalmente este problema y sus características. Sean i fábricas de envasado y j mercados de consumo. Cada fábrica tiene una capacidad máxima de producción de a_i cajas y cada mercado demanda una cantidad b_j de cajas (se supone que la capacidad de producción total de las fábricas es superior a la demanda total para que el problema sea factible). El coste de transporte entre cada fábrica i y cada mercado j por cada caja es c_{ij}. Se desea satisfacer la demanda de cada mercado al mínimo coste. Las variables de decisión del problema serán las cajas transportadas entre cada fábrica i y cada mercado j, x_{ij}. Las ecuaciones que deben satisfacerse son:

Límite de capacidad máxima de producción de cada fábrica

$$\sum_j x_{ij} \le a_i \text{ para cada fábrica } i$$

Satisfacción de la demanda de cada mercado

$$\sum_i x_{ij} \ge b_j \text{ para cada mercado } j$$

La función objetivo será la minimización de los costes totales de transporte

$$\sum_i \sum_j c_{ij} x_{ij}$$

Ésta es la forma algebraica de representación de este problema de optimización. La codificación en lenguaje GAMS aparece a continuación.

```
$TITLE MODELO DE TRANSPORTE

SETS
   I fábricas de envasado / VIGO, ALGECIRAS /
   J mercados de consumo  / MADRID, BARCELONA, VALENCIA /

PARAMETERS
   A(i) capacidad de producción de la fábrica i [cajas]
       / VIGO      350
         ALGECIRAS 700 /

   B(j) demanda del mercado j [cajas]
       / MADRID    400
         BARCELONA 450
         VALENCIA  150 /

TABLE C(i,j) coste unitario transporte entre i y j [miles de euros por caja]
          MADRID BARCELONA VALENCIA
VIGO       0.06     0.12     0.09
ALGECIRAS  0.05     0.15     0.11

VARIABLES
   X(i,j) cajas transportadas entre fábrica i y mercado j [cajas]
   CT     coste de transporte [miles de euros]

POSITIVE VARIABLE X

EQUATIONS
   COSTE        coste total de transporte     [miles de euros]
   CAPACIDAD(i) capacidad máxima de cada fábrica i     [cajas]
   DEMANDA(j)   satisfacción demanda de cada mercado j [cajas] ;

COSTE ..        CT =E= SUM[(i,j), C(i,j) * X(i,j)] ;

CAPACIDAD(i) .. SUM[j, X(i,j)] =L= A(i) ;

DEMANDA(j) ..   SUM[i, X(i,j)] =G= B(j) ;

MODEL TRANSPORTE / COSTE, CAPACIDAD, DEMANDA /

SOLVE TRANSPORTE USING LP MINIMIZING CT
```

Se ha puesto especial cuidado en presentar de forma clara, concisa y limpia el código. El contenido de este código resulta prácticamente autoexplicativo. El resultado de la ejecución del modelo de transporte se presenta a continuación.

```
C o m p i l a t i o n
```

```
COMPILATION TIME     =        0.000 SECONDS     0.7 Mb      WIN-19-115

Equation Listing    SOLVE TRANSPORTE USING LP FROM LINE 39

---- COSTE  =E=  coste total de transporte   [miles de euros]

COSTE..  - 0.06*X(VIGO,MADRID) - 0.12*X(VIGO,BARCELONA) - 0.09*X(VIGO,VALENCIA)

     - 0.05*X(ALGECIRAS,MADRID) - 0.15*X(ALGECIRAS,BARCELONA)

     - 0.11*X(ALGECIRAS,VALENCIA) + CT =E= 0 ; (LHS = 0)

---- CAPACIDAD  =L=  capacidad máxima de cada fábrica i     [cajas]

CAPACIDAD(VIGO)..  X(VIGO,MADRID) + X(VIGO,BARCELONA) + X(VIGO,VALENCIA) =L= 350 ;

     (LHS = 0)

CAPACIDAD(ALGECIRAS)..  X(ALGECIRAS,MADRID) + X(ALGECIRAS,BARCELONA)

     + X(ALGECIRAS,VALENCIA) =L= 700 ; (LHS = 0)

---- DEMANDA  =G=  satisfacción demanda de cada mercado j [cajas]

DEMANDA(MADRID)..  X(VIGO,MADRID) + X(ALGECIRAS,MADRID) =G= 400 ;

     (LHS = 0, INFES = 400 ***)

DEMANDA(BARCELONA)..  X(VIGO,BARCELONA) + X(ALGECIRAS,BARCELONA) =G= 450 ;

     (LHS = 0, INFES = 450 ***)

DEMANDA(VALENCIA)..  X(VIGO,VALENCIA) + X(ALGECIRAS,VALENCIA) =G= 150 ;

     (LHS = 0, INFES = 150 ***)

Column Listing      SOLVE TRANSPORTE USING LP FROM LINE 39

---- X  cajas transportadas entre fábrica i y mercado j [cajas]

X(VIGO,MADRID)
            (.LO, .L, .UP = 0, 0, +INF)
      -0.06    COSTE
       1       CAPACIDAD(VIGO)
       1       DEMANDA(MADRID)

X(VIGO,BARCELONA)
            (.LO, .L, .UP = 0, 0, +INF)
      -0.12    COSTE
       1       CAPACIDAD(VIGO)
       1       DEMANDA(BARCELONA)

X(VIGO,VALENCIA)
            (.LO, .L, .UP = 0, 0, +INF)
      -0.09    COSTE
       1       CAPACIDAD(VIGO)
       1       DEMANDA(VALENCIA)

REMAINING 3 ENTRIES SKIPPED

---- CT  coste de transporte [miles de euros]

CT
            (.LO, .L, .UP = -INF, 0, +INF)
       1       COSTE

Model Statistics    SOLVE TRANSPORTE USING LP FROM LINE 39

MODEL STATISTICS
```

```
BLOCKS OF EQUATIONS       3     SINGLE EQUATIONS        6
BLOCKS OF VARIABLES       2     SINGLE VARIABLES        7
NON ZERO ELEMENTS        19

GENERATION TIME      =        0.140 SECONDS     1.4 Mb      WIN-19-115

EXECUTION TIME       =        0.140 SECONDS     1.4 Mb      WIN-19-115

           S O L V E      S U M M A R Y

    MODEL   TRANSPORTE          OBJECTIVE  CT
    TYPE    LP                  DIRECTION  MINIMIZE
    SOLVER  CPLEX               FROM LINE  39

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL
**** OBJECTIVE VALUE              93.5000

 RESOURCE USAGE, LIMIT          0.401     1000.000
 ITERATION COUNT, LIMIT         5        10000

GAMS/Cplex    May 18, 2000 WIN.CP.CP 19.3 016.014.038.WAT For Cplex 6.6
Cplex 6.6.1, GAMS Link 16, Using a GAMS/Cplex demo license installed at runtime.

Optimal solution found.

Objective :          93.500000

                  LOWER     LEVEL     UPPER    MARGINAL

---- EQU COSTE           .         .         .        1.000

  COSTE  coste total de transporte   [miles de euros]

---- EQU CAPACIDAD  capacidad máxima de cada fábrica i     [cajas]

          LOWER     LEVEL     UPPER    MARGINAL

VIGO          -INF    350.000   350.000    -0.030
ALGECIRAS     -INF    650.000   700.000      .

---- EQU DEMANDA  satisfacción demanda de cada mercado j [cajas]

          LOWER     LEVEL     UPPER    MARGINAL

MADRID      400.000   400.000     +INF      0.050
BARCELONA   450.000   450.000     +INF      0.150
VALENCIA    150.000   150.000     +INF      0.110

---- VAR X  cajas transportadas entre fábrica i y mercado j [cajas]

                  LOWER     LEVEL     UPPER    MARGINAL

VIGO     .MADRID         .         .       +INF      0.040
VIGO     .BARCELONA      .     350.000     +INF       .
VIGO     .VALENCIA       .         .       +INF      0.010
ALGECIRAS.MADRID         .     400.000     +INF       .
ALGECIRAS.BARCELONA      .     100.000     +INF       .
ALGECIRAS.VALENCIA       .     150.000     +INF       .

                  LOWER     LEVEL     UPPER    MARGINAL

---- VAR CT            -INF     93.500     +INF       .

  CT  coste de transporte [miles de euros]

**** REPORT SUMMARY :        0     NONOPT
                       0 INFEASIBLE
```

```
                 0  UNBOUNDED

EXECUTION TIME      =       0.030 SECONDS    0.7 Mb      WIN-19-115

**** FILE SUMMARY

INPUT      D:\TR.GMS
OUTPUT     D:\TR.LST
```

1.2.3.2 Ejemplo de planificación de la producción

A continuación se presenta un ejemplo de planificación de la producción de una fábrica de papel. Se dispone de varias máquinas para producir diferentes tipos de papel. Se trata de determinar cuáles son las cantidades óptimas a producir de cada tipo de papel en cada máquina para maximizar el beneficio. La demanda de cada tipo de papel se considera fija y conocida y existen limitaciones en el tiempo de producción disponible en cada máquina.

```
$TITLE Planificación de la producción de una papelera

* la papelera puede producir cuatro tipos diferentes de papel en tres máquinas
* distintas. Dada una demanda fija se trata de determinar la producción que
* maximiza los beneficios mensuales

SETS
   M máquinas       / maquina1 * maquina3 /
   P tipos de papel / prensa, folio, imprenta, reciclado /

TABLE TASAPROD(p,m) tasa de producción (t por h)
                 maquina1  maquina2  maquina3
      prensa        53        52        49
      folio         51        49        44
      imprenta      52        45        47
      reciclado     42        44        40

TABLE COSTEPROD(p,m) coste de producción (€ por t)
                 maquina1  maquina2  maquina3
      prensa        76        75        73
      folio         82        80        78
      imprenta      96        95        92
      reciclado     72        71        70

TABLE DATDEM(p,*) demanda y precio de venta
                  demanda   precio
*              (t por mes)(€ por t)
      prensa        30000      77
      folio         20000      81
      imprenta      12000      99
      reciclado      8000     105

PARAMETER TIEMPOMAQ(m) tiempo disponible al mes de cada máquina (h)
                   / maquina1   672
                     maquina2   600
```

```
                     maquina3   480 /

VARIABLES
   PRODUCC(p,m) producción de cada tipo papel en cada máquina (t por mes)
   BENEFICIO    beneficio  (€ por mes)

POSITIVE VARIABLE PRODUCC

EQUATIONS
   CAPACMAQ(m) capacidad de cada máquina     (h  por mes)
   DEMANDAP(p) demanda de cada tipo de papel (t por mes)
   BENEF       beneficio                     (€  por mes) ;

CAPACMAQ(m) .. SUM[p, PRODUCC(p,m)/TASAPROD(p,m)] =L= TIEMPOMAQ(m) ;

DEMANDAP(p) .. SUM[m, PRODUCC(p,m)] =E= DATDEM(p,'demanda') ;

BENEF       .. BENEFICIO =E= SUM[p, DATDEM(p,'demanda')*DATDEM(p,'precio')]
                           - SUM[(p,m), COSTEPROD(p,m)*PRODUCC(p,m)] ;

MODEL PAPELERA / ALL /

SOLVE PAPELERA USING LP MAXIMIZING BENEFICIO
```

1.2.3.3 Ejemplo de secuenciación de órdenes de trabajo

Dada una máquina y 5 trabajos que hay que realizar en ella, en cualquier orden, se dispone del tiempo de ejecución de cada trabajo

TR1	TR2	TR3	TR4	TR5
15	13	12	14	16

y del tiempo de ajuste de la máquina para pasar de ejecutar el trabajo i (fila) a ejecutar el trabajo j (columna)

	TR1	TR2	TR3	TR4	TR5
TR1		2	5	1	6
TR2	3		4	2	5
TR3	4	2		3	4
TR4	5	3	6		5
TR5	4	4	4	3	

Resolver el problema de determinar cuál es el menor tiempo posible para completar los 5 trabajos y cómo hacerlo. Se considera un ciclo de trabajo cerrado, que se repite y vuelve a comenzar.

```
$TITLE Secuenciación de órdenes de trabajo

SETS
   I trabajos que se van a ejecutar / TR1 * TR5 /

ALIAS (i,j)

TABLE C(i,j) tiempo de ajuste para pasar del trabajo i al trabajo j
        TR1    TR2    TR3    TR4    TR5
TR1             2      5      1      6
TR2      3             4      2      5
TR3      4      2             3      4
TR4      5      3      6             5
TR5      4      4      4      3

PARAMETERS
m cota superior de tiempo del problema;
VARIABLES
   X(i,j) paso del trabajo i al trabajo j
   T(i) instante de comienzo de trabajo i
   TT     tiempo total en completar los trabajos

BINARY VARIABLE X
POSITIVE VARIABLE T

EQUATIONS
   TIEMPOTOT       tiempo total de trabajo
   ANTERIOR(i)  de cada trabajo se parte una vez
   POSTERIOR(j) a  cada trabajo se llega una vez
   TIEMPO(i,j) tiempo inicio trabajo j despues de i
   TINICIO(j)  tiempo inicio segundo trabajo;

TIEMPOTOT       .. TT =E= SUM[(i,j) $(NOT SAMEAS(i,j)), C(i,j)*X(i,j)] ;
ANTERIOR(i)  .. SUM[j $(NOT SAMEAS(i,j)), X(i,j)] =E= 1 ;
POSTERIOR(j) .. SUM[i $(NOT SAMEAS(i,j)), X(i,j)] =E= 1 ;
TIEMPO(i,j) $(NOT SAMEAS(i,j) AND ORD(i)>1) .. T(j) =G= T(i) + C(i,j)-m*(1-X(i,j)) ;
TINICIO(j)$(ORD(j)>1).. T(j) =G= SUM(i$(ORD(i)=1),C(i,j)-m*(1-X(i,j)));

MODEL AJUSTE / TIEMPOTOT, ANTERIOR, POSTERIOR, TIEMPO,TINICIO/;
m=SUM((i,j),C(i,j));

SOLVE AJUSTE USING MIP MINIMIZING TT
```

1.2.3.4 Ejemplo del viajante de comercio

Veamos tres posibles formulaciones del problema del viajante de comercio escritas en GAMS. La primera formulación es la denominada de Miller, Tucker y Zemlin

$$\begin{aligned}
&\min_{x_{ij},u_i} \sum_{i,j} c_{ij} x_{ij} \\
&\sum_i x_{ij} = 1 \quad \forall j \\
&\sum_j x_{ij} = 1 \quad \forall i \\
&u_i - u_j + [\mathrm{card}(i) - 1] x_{ij} \le \mathrm{card}(i) - 2 \quad \forall i, j \\
&x_{ij} \in \{0,1\}, u_i \ge 0
\end{aligned} \tag{1.12}$$

y las dos últimas a las presentadas previamente en la sección I.3.2 de formulación de problemas de optimización lineal entera.

```
$TITLE Problema del viajante de comercio

SETS
I Ciudades
K Etapas

ALIAS(I,J)

PARAMETER
COSTE(i,j)       Coste de ir de la ciudad i a la ciudad j [€]
m                COTA PARA EL MÁXIMO TIEMPO

VARIABLE
FOBJ             Función objetivo [€]

BINARY VARIABLES
X(i,j)          Indica si se viaja de la ciudad i a la ciudad j

POSITIVE VARIABLE
T(i)             Instante en que llega a una cierta ciudad

EQUATIONS
E_FOBJ1              Función objetivo
E_ORIGEN1(i)         Cada ciudad es origen  una sola vez
E_DESTINO1(j)        Cada ciudad es destino una sola vez
E_TIEMPO(i,j)     Restricciones para eliminar subciclos relacionando tiempo
E_TINICIO(j)     la primera en visitar...
;

E_FOBJ1       .. FOBJ =E= SUM[(i,j) $(NOT SAMEAS(i,j)), COSTE(i,j) * X(i,j)];

E_ORIGEN1(i)  .. SUM[j $(NOT SAMEAS(i,j)), X(i,j)] =E= 1;

E_DESTINO1(j) .. SUM[i $(NOT SAMEAS(i,j)), X(i,j)] =E= 1;

E_TIEMPO(i,j) $(NOT SAMEAS(i,j) AND ORD(i)>1) .. T(j) =G= T(i) + COSTE(i,j)-m*(1-X(i,j))
;

E_TINICIO(j)$(ORD(j)>1).. T(j) =G= SUM(i$(ORD(i)=1),COSTE(i,j)-m*(1-X(i,j)));

MODEL TSP1  / E_FOBJ1, E_ORIGEN1, E_DESTINO1, E_TIEMPO, E_TINICIO /

********************************************************************************
* Caso ejemplo: Schrage (1997) p. 319
* Esta información se podría introducir mediante ficheros de texto
* utilizando la instrucción $INCLUDE nombrefichero

SETS
I        Ciudades
/ Atl, Chi, Cin, Hou, LA, Mon, NY, Phi, Pit, StL, SD, SF /
```

```
TABLE COSTE(i,j) Coste de ir de la ciudad i a la ciudad j [€]
      Atl    Chi    Cin    Hou    LA     Mon    NY     Phi    Pit    StL     SD     SF
Atl   0      702    454    842    2396   1196   864    772    714    554    2363   5679
Chi   702    0      324    1093   2136   764    845    764    459    294    2184   2187
Cin   454    324    0      1137   2180   798    664    572    284    338    2228   2463
Hou   842    1093   1137   0      1616   1857   1706   1614   1421   799    1521   2021
LA    2396   2136   2180   1616   0      2900   2844   2752   2464   1842   95     405
Mon   1196   764    798    1857   2900   0      396    424    514    1058   2948   2951
NY    864    845    664    1706   2844   396    0      92     386    1002   2892   3032
Phi   772    764    572    1614   2752   424    92     0      305    910    2800   2951
Pit   714    459    284    1421   2464   514    386    305    0      622    2512   2646
StL   554    294    338    799    1842   1058   1002   910    622    0      1890   2125
SD    2363   2184   2228   1521   95     2948   2892   2800   2512   1890   0      500
SF    2679   2187   2463   2021   405    2951   3032   2951   2646   2125   500    0     ;

COSTE(i,j) = COSTE(i,j) / 1e3 ;

m=SUM((i,j),COSTE(i,j)) ;

OPTION OPTCR = 0

SOLVE TSP1 USING MIP MINIMIZING FOBJ
DISPLAY X.L, T.L
```

1.2.3.5 Ejemplo de asignación de grupos térmicos

El problema de la asignación de grupos térmicos de producción de electricidad consiste en la decisión de qué grupos térmicos hay que acoplar en cada hora del día (o semana) de manera que:

- Se minimicen los costes variables de generación (incluyendo costes de combustible y costes de arranque y parada)
- Se suministre la demanda en cada hora
- Se mantenga un cierto nivel de reserva rodante
- Se respeten los parámetros de funcionamiento de los grupos térmicos (mínimos técnicos, potencia nominal, rampas de subida y bajada)

Datos

D_h demanda térmica en la hora h [MW]

R coeficiente de reserva rodante con respecto a la demanda [p.u.]

a_t término lineal del coste de combustible del grupo térmico t [€/MWh]

b_t término fijo del coste de combustible del grupo térmico t [€/h]

ca_t coste de arranque del grupo térmico t [€]

cp_t coste de parada del grupo térmico t [€]

$\overline{P}_t$ potencia máxima del grupo térmico t [MW]

$\underline{P}_t$ potencia mínima del grupo térmico t [MW]

rs_t rampa de subida del grupo térmico t [MW/h]

rb_t rampa de bajada del grupo térmico t [MW/h]

Variables

P_{ht} potencia producida por el grupo térmico t en la hora h [MW]

A_{ht} acoplamiento del grupo térmico t en la hora h [0,1]

AR_{ht} arranque del grupo térmico t en la hora h [0,1]

PR_{ht} parada del grupo térmico t en la hora h [0,1]

$$\min \sum_{h=1}^{H} \sum_{t=1}^{T} \left(a_t P_{ht} + b_t A_{ht} + ca_t AR_{ht} + cp_t PR_{ht} \right)$$

$$\sum_{t=1}^{T} P_{ht} = D_h \qquad H$$

$$\sum_{t=1}^{T} (\overline{P}_t A_{ht} - P_{ht}) = RD_h \qquad H$$

$$\underline{P}_t A_{ht} \le P_{ht} \le \overline{P}_t A_{ht} \qquad 2HT$$

$$A_{ht} - A_{h-1t} = AR_{ht} - PR_{ht} \qquad (H-1)T$$

$$P_{ht} - P_{h-1t} \le rs_t \qquad (H-1)T$$

$$P_{h-1t} - P_{ht} \le rb_t \qquad (H-1)T$$

$$P_{ht} \ge 0 \;\; A_{ht}, AR_{ht}, PR_{ht} \in \{0,1\}$$

```
$TITLE ASIGNACIÓN HORARIA DE GRUPOS TÉRMICOS

SETS
T grupos térmicos /GALICIA, CATALUNA, MADRID, VALENCIA,
                   EXTREMAD, ANDALUCI, CASTLEON/
H hora h /h1 * h5/

SCALAR
r porcentaje de reserva rodante sobre la demanda [p.u.] /0.2/

PARAMETERS
d(h) demanda cada hora [MW]
/h1 1000 , h2 1400 , h3 2400 , h4 2000 , h5 1000/
pmax(t) pot máxima de cada térmico [MW]
/GALICIA 400, CATALUNA 500, MADRID 700, VALENCIA 400,
 EXTREMAD 300, ANDALUCI 800, CASTLEON 800/
pmin(t) pot mínima de cada térmico [MW]
/GALICIA 100, CATALUNA 150, MADRID 150, VALENCIA 50,
```

```
 EXTREMAD 50, ANDALUCI 400, CASTLEON 200 /
rs(t) rampa de subida [MW por hora]
/GALICIA 200, CATALUNA 300, MADRID 500, VALENCIA 300,
 EXTREMAD 100, ANDALUCI 500, CASTLEON 400/
rb(t) rampa de bajada [MW por hora]
/GALICIA 300, CATALUNA 300, MADRID 200, VALENCIA 100,
 EXTREMAD 100, ANDALUCI 500, CASTLEON 400/
c(t)  coste lineal de producción [€ por MWh]
/GALICIA 4, CATALUNA 4, MADRID 4, VALENCIA 4,
 EXTREMAD 3, ANDALUCI 2, CASTLEON 7/
b(t)  coste fijo de producción [€]
/GALICIA 50, CATALUNA 30, MADRID 30, VALENCIA 25,
 EXTREMAD 30, ANDALUCI 80, CASTLEON 70/
ca(t) coste de arranque
/GALICIA 10, CATALUNA 20, MADRID 10, VALENCIA 15,
 EXTREMAD 20, ANDALUCI 10, CASTLEON 15/
cp(t) coste de parada
/GALICIA 5, CATALUNA 10, MADRID 5, VALENCIA 10,
 EXTREMAD 5, ANDALUCI 15, CASTLEON 10/

VARIABLES
CT      coste variable total del sistema [M€]
A(t,h)  acoplamiento del grupo t a las h horas [0-1]
AR(t,h) arranque     del grupo t a las h horas [0-1]
PR(t,h) parada       del grupo t a las h horas [0-1]
P(t,h)  generación producida por el grupo t a las h horas [MW]

BINARY VARIABLE A,AR,PR
POSITIVE VARIABLE P

EQUATIONS
COSTE          costes variables de generación-función objetivo [€]
DEMANDA(h)     abastecimiento de la demanda [MW]
RESERVA(h)     reserva rodante del sistema [MW]
COTASUP(t,h)   cota superior de producción del grupo t [MW]
COTAINF(t,h)   cota inferior de producción del grupo t [MW]
RAMPASUB(t,h)  limitación de rampa de subida del grupo t [MW]
RAMPABAJ(t,h)  limitación de rampa de bajada del grupo t [MW]
LOGICA(t,h)    relación lógica entre variables de acoplamiento arranque y parada ;

COSTE .. CT =E= SUM[(T,H), c(t)*P(t,h)+b(t)*A(t,h)+ca(t)*AR(t,h)+cp(t)*PR(t,h)] ;

DEMANDA(h) .. SUM[T, P(t,h)] =E= d(h) ;

RESERVA(h) .. SUM[T, A(t,h)*pmax(t)-P(t,h)] =G= d(h)*r;

COTASUP(t,h) .. P(t,h) =L= pmax(t)*A(t,h);

COTAINF(t,h) .. P(t,h) =G= pmin(t)*A(t,h);

RAMPASUB(t,h) .. P(t,h)-P(t,h-1) =L= rs(t);

RAMPABAJ(t,h) .. P(t,h-1)-P(t,h) =L= rb(t);

LOGICA(t,h) .. A(t,h)-A(t,h-1) =E= AR(t,h)-PR(t,h);

MODEL ASIGNA /COSTE,DEMANDA,RESERVA,COTASUP,COTAINF,RAMPASUB,RAMPABAJ,LOGICA/ ;

P.UP(t,h) = pmax(t)

OPTION OPTCR = 0

SOLVE ASIGNA USING MIP MINIMIZING CT
```

1.2.4 Elementos de estilo de programación

> "En los últimos años se ha reconocido la programación de computadores como una disciplina cuyo dominio es básico y crucial para el éxito de muchos proyectos de ingeniería" Niklaus Wirth (1976).

La programación no es un castigo divino para los humanos, es *ciencia* y *arte*. Es ciencia en la medida que se pueden implantar modelos matemáticos complejos y que el pensamiento, la disciplina, la rigurosidad y la experimentación acompañan este desarrollo. El resultado es arte por la belleza, elegancia, sensación que puede transmitir un modelo y la profesionalidad de su creador.

Una forma de aprender a escribir con estilo y estructura ordenada es mediante la *lectura* de ejemplos ilustrativos o de código ajeno. Una manera de programar es por *refinamiento gradual* de los detalles. Es importante recordar que en el desarrollo de una aplicación el diablo se esconde en los detalles.

La potencia y concisión de los modelos escritos en un lenguaje de modelado hacen que el propio código forme parte de la documentación. De hecho la reutilización de modelos fue una de las causas que dieron origen a los lenguajes de modelado. La etapa de diseño del modelo cobra gran importancia para permitir posteriores ampliaciones. Por esta razón es importante el estilo en la programación, que incide en la *calidad* y *mantenibilidad*[6] del código desarrollado. Piénsese que el tiempo dedicado a mantenimiento y ampliación de un modelo es muy superior al inicial de desarrollo. El desarrollo y la depuración del modelo se debe hacer con una maqueta (caso ejemplo sencillo) para finalmente utilizar un problema real.

He aquí algunas *recomendaciones* para la escritura de un modelo que inciden en la *calidad* del desarrollo:

MODULARIDAD

Estructurar el modelo en diversos módulos con diferentes propósitos. Por ejemplo, la inclusión de los datos[7] y la escritura de resultados deben separarse en diferentes ficheros que son convenientemente insertados mediante la instrucción $include en el módulo principal, que contiene la formulación del problema de optimización.

---

[6] Se entiende por mantenibilidad la reutilización, reparación o modificación de un modelo.

[7] Se recomienda la introducción de los datos tal como son recogidos y entendidos por el usuario y se hacen en el modelo los cálculos auxiliares que sean necesarios.

Utilización de las entidades (parámetros, escalares, etc.) con el mismo propósito y significado en las diferentes partes del código. Es decir, mantener la definición y uso de cada parámetro y escalar en todo el código para evitar la confusión del lector.

Comprobación de la pertenencia de un subconjunto a un conjunto de forma explícita en su definición, es decir, evitar el uso de índices comodín en vectores y matrices. Esta es una manera de validar y evitar errores en la introducción de los datos.

ESCRIBIR CÓDIGO PARA FACILITAR SU LECTURA

Estas otras *recomendaciones* están orientadas al cuidado exquisito de la estética. Es el primer paso en el desarrollo profesional de un modelo. Son fundamentales, aunque aparentemente carecen de importancia para el desarrollador, pero se hacen imprescindibles para su *mantenimiento* y ampliación. El código debe ser limpio y claro para que pueda ser mantenido.

Mantener una coherencia en las reglas de escritura, de manera que se observe una norma sistemática en todo el código. Por ejemplo, endentación en las instrucciones repetitivas, sangría de tres espacios cada vez que se realiza una instrucción tipo LOOP, IF. Las palabras reservadas del lenguaje van en mayúsculas (LOOP, IF, THEN, ELSE, SET, SCALAR, PARAMETER, TABLE, etc.). La coma del final de instrucción va separada por un blanco. El signo de igualdad en las asignaciones se separa por espacios en blanco a ambos lados.

Establecer paralelismos o réplicas entre instrucciones consecutivas semejantes.

Las líneas de código deben tener una longitud aproximada de 100 columnas, no sobrepasando nunca las 110. Romper la instrucción en cuantas líneas sea necesario para cumplir esta recomendación.

Los comentarios deben ser suficientemente ilustrativos del contenido y estar bien localizados. Deben ayudar a documentar la naturaleza y origen de los datos

Se deben utilizar nombres largos y descriptivos para las entidades del modelo.

Los nombres y los índices de los parámetros, variables y ecuaciones han de ser acrónimos que representen su significado. Se recomienda una longitud de hasta 10 caracteres para los primeros y de hasta 2 para los segundos. Los comentarios explicativos pueden hacerse de hasta 80 caracteres.

Las definiciones de las entidades del modelo deben llevar las dimensiones físicas del problema.

Hacer un uso sistemático de mayúsculas y minúsculas con algún criterio predefinido, que debe ser coherente y mantenerse a lo largo de todo el programa. Por ejemplo, los nombres de los parámetros, variables y ecuaciones van en mayúsculas. Los nombres de sus índices van en minúsculas.

REFORMULACIÓN MANUAL DEL PROBLEMA

Un primer estadio en la formulación de un problema está en la elección de la propia formulación. A veces se pueden utilizar formulaciones semejantes con coste computacional muy diferente. Por ejemplo, para la representación de las pérdidas en un circuito eléctrico se puede utilizar una función no lineal o una poligonal aproximada. Habitualmente la formulación poligonal convexa requiere mucho menos tiempo.

Diferentes formulaciones matemáticamente equivalentes de un mismo problema de optimización pueden requerir tiempos de optimización muy distintos. Esta afirmación es especialmente relevante en problemas de programación lineal entera mixta y programación no lineal. Por esta razón, siempre es conveniente un ejercicio continuo de experimentación y reformulación de los problemas.

Veamos estas tres formulaciones de un problema NLP

$$\min \sum_{i=1}^{n} \sum_{j=i+1}^{n} q_{ij} x_i x_j$$

$$\sum_{j=1}^{n} x_j = 1$$

$$\sum_{j=1}^{n} r_j x_j = r_0$$

$$\min \sum_{i=1}^{n} x_i \sum_{j=i+1}^{n} q_{ij} x_j$$

$$\sum_{j=1}^{n} x_j = 1$$

$$\sum_{j=1}^{n} r_j x_j = r_0$$

$$\min \sum_{i=1}^{n} x_i w_i$$

$$w_i = \sum_{j=i+1}^{n} q_{ij} x_j$$

$$\sum_{j=1}^{n} x_j = 1$$

$$\sum_{j=1}^{n} r_j x_j = r_0$$

La ventaja de la formulación segunda con respecto a la primera es inmediata. La formulación 1 requiere para evaluar la función objetivo aproximadamente $2n n/2$ multiplicaciones. En la formulación 2 se necesitan $n + n n/2$ aproximadamente. La tercera formulación tiene esencialmente las mismas multiplicaciones pero aparecen en restricciones lineales. El número de restricciones aumenta sustancialmente pero todas son lineales y los métodos de manipulación de restricciones lineales son extremadamente eficientes. La formulación 3 resulta ser la más eficiente.

De hecho, algunos optimizadores realizan una etapa previa de preproceso[8] del problema antes de su resolución (parámetro de control presolve en el CPLEX). Como ejemplo, el impacto en el tamaño de dos problemas LP debido al preproceso realizado por el optimizador CPLEX 6.0 se muestra en la tabla 1.3.

	Caso 1			**Caso 2**		
	Restricc.	Variables	Elementos	Restricc.	Variables	Elementos
Sin preproceso	19047	27262	81215	48971	63935	187059
Con preproceso	15744	21982	51079	40794	56133	135361
Decremento	17%	19%	37%	17%	12%	28%

Tabla 1.3 Reducción de tamaños con la opción de preproceso.

En la formulación del problema la opción profile muestra el tiempo y memoria consumidos y el número de asignaciones realizadas o restricciones creadas.

[8] El desarrollo de las técnicas de preproceso y reformulación han originado avances muy importantes en la resolución de problemas MIP.

Entre las principales consideraciones para mejorar la formulación de un problema de optimización se pueden citar:

- Cálculo analítico del número de restricciones y variables

  Éste es una ayuda para ser consciente del tamaño esperable del problema y ver su dependencia en función de los elementos básicos que lo componen. El número real de restricciones para un caso concreto se muestra con la opción profile. Puede ser utilizado para detectar errores en la formulación. Por ejemplo, por excesivo número de ecuaciones al haber puesto dimensiones superfluas no controladas convenientemente con conjuntos dinámicos.

  Es conveniente también conocer la estructura de la matriz de restricciones, es decir, los bloques que la componen. Existe alguna utilidad, citada en el siguiente apartado, que lo permite hacer.

- No crear variables ni ecuaciones superfluas.

  Hay que tener cuidado con lo que se entiende por superfluas porque algunas condiciones redundantes pueden realmente llevar a obtener un modelo más fuerte en el contexto de programación entera. Sin embargo, el conocimiento de la naturaleza del problema permite introducir condiciones lógicas (mediante el uso del operador $) que eliminan algunas de ellas en la escritura de las ecuaciones o de las variables. Por ejemplo, en el caso de una red se suprimen variables o ecuaciones asociadas a líneas entre nudos no conectados entre sí. Aunque los optimizadores pueden detectar algunas de estas ecuaciones/variables superfluas, es más eficiente evitarlo mediante condiciones expresas.

  La opción solprint=on o la utilidad gamschk puede ayudar en la detección de las variables o ecuaciones superfluas (porque toman valor 0 o conocido bajo toda circunstancia en la solución).

- Reducción del número de restricciones y/o elementos de la matriz aun a costa de aumentar el número de variables.

  Una manera de reducir el número de ecuaciones o de variables es introduciendo expresamente el conocimiento que se tiene del problema real (casos particulares que pueden aparecer y sus implicaciones). Es más conveniente hacerlo manualmente a dejar que lo intente el preproceso del optimizador.

Se puede hacer mediante sustitución, definición de nuevas variables, reformulación en general se debe intentar reducir el número de restricciones y/o de elementos de la matriz.

Como norma general para la formulación de problemas lineales es conveniente saber que el tiempo necesario para su solución por el método simplex depende aproximadamente del cubo del número de restricciones, no siendo demasiado influyente el número de variables. En el método de punto interior el tiempo de ejecución depende principalmente del número de elementos (densidad) de la matriz de restricciones.

- Escalación tanto de variables como de coeficientes y valores de restricciones a números alrededor de 1

  Esto mejora el comportamiento numérico en la resolución del problema y reduce el tiempo de ejecución. La escalación resulta muy conveniente en problemas LP de gran tamaño pero es imprescindible en problemas NLP. Implícitamente los valores por omisión de los parámetros de control de los optimizadores están fijados suponiendo que el problema está bien escalado alrededor de 1. Con una escalación razonable puede haber como mucho 6 órdenes de magnitud de diferencia (por ejemplo, coeficientes de las variables entre 0.001 y 1000). La utilidad gamschk es una herramienta muy útil para observar los intervalos de variación de los coeficientes de las variables en las restricciones y de las costas de éstas para detectar potenciales problemas de escalado.

  La escalación se puede hacer *manualmente* –expresando las variables, parámetros y ecuaciones en unidades naturales con sentido físico para el problema– o *automáticamente* mediante las opciones disponibles en el lenguaje (nombre_modelo.scaleopt=1) o en el optimizador (scale). La escalación manual requiere más cuidado y control pero es preferible porque conserva la naturaleza física del problema dentro del código y es igual de efectiva que la automática.

  En cualquier caso hay que tener cuidado al realizar el escalado, especialmente en problemas no lineales, donde pueden existir efectos no lineales que invaliden la nueva formulación.

- Acotamiento de las variables.

  Las cotas en las variables no cuentan como restricciones desde el punto de vista del tiempo de cálculo, ya que los algoritmos de optimización las tratan de forma específica. Las cotas pueden tener sentido *físico* (y, por tanto, forman parte de la naturaleza del problema) o ser *algorítmicas* (es decir, cotas superfluas que nunca deben ser activas en la solución óptima pero que reducen el tiempo de optimización).

El preproceso generalmente incluye procedimientos para el fortalecimiento de las cotas de las variables (reducción de las cotas superiores y aumento de las inferiores).

TRATAMIENTO EXPLÍCITO DE CONJUNTOS ORDENADOS SOSN

Los conjuntos ordenados (*Special Ordered Sets SOS*) son conjuntos de variables que cumplen las siguientes condiciones:

Como mucho n elementos del conjunto toman valores diferentes de 0. El resto de elementos ha de ser 0

Si hay n elementos que son diferentes de 0 deben ser contiguos

Los conjuntos ordenados tienen un tratamiento especial en la optimización, por lo que su definición puede mejorar mucho el tiempo requerido para la resolución.

SELECCIÓN DEL OPTIMIZADOR Y TIPO ALGORITMO DE OPTIMIZACIÓN

Un lenguaje de modelado permite utilizar diferentes optimizadores para la resolución de un mismo problema de optimización. Esta característica representa una gran ventaja por la flexibilidad que aporta en la selección del optimizador más adecuado a las características del problema.

En Internet (www-c.mcs.anl.gov/otc/guide/faq/linear-programming.html) y en la revista *OR/MS Today*, Fourer (2003), se pueden encontrar opiniones y revisiones del software disponible para la resolución de problemas de optimización de todo tipo. Entre los optimizadores a los que se ha tenido acceso destacan CPLEX y OSL para LP, MINOS y CONOPT para NLP y MILES y PATH para MCP.

El mejor método para un problema concreto depende de las características del problema, de los detalles de implantación del método simplex o del punto interior y del ordenador utilizado. Por esta razón los paquetes comerciales de LP importantes incluyen métodos de punto interior (habitualmente primal-dual predictivo-correctivo), métodos simplex (en su versión primal y dual) y de resolución de flujos de redes (simplex de red). Se debe utilizar el método de optimización (punto interior o barrera, simplex primal o simplex dual) más adecuado al tipo o tamaño del problema.

Como recomendación general, para problemas de tamaño medio (hasta aproximadamente de 10000 x 10000) el método más adecuado es el simplex y para problemas de gran tamaño (desde 10000 x 10000 hasta 100000 x 100000) el mejor método es el de punto interior (especialmente en problemas degenerados). Para problemas de tamaño superior se requiere el uso de técnicas de optimización específicas (como, por ejemplo, las de descomposición entre otras [Ramos, 1996]). La selección de un método u otro se debe realizar principalmente en función del tamaño del problema. [Bixby, 2000] es un artículo práctico reciente donde se presentan algunas comparaciones entre métodos de solución del optimizador CPLEX tanto para problemas lineales como enteros mixtos.

El método simplex también resulta adecuado en la realización de análisis de sensibilidad, es decir, cuando se trata de resolver problemas similares disponiendo de una solución próxima y una base previa, como sucede en el método de ramificación y acotamiento para resolver problemas lineales enteros.

A continuación se presenta la tabla 1.4 de comparación entre varios optimizadores y métodos de optimización. En la tabla 1.5 se muestra la diferencia de funcionamiento entre las opciones de preproceso de dos optimizadores.

		Caso 1			**Caso 2**		
		Tiempo	Índice	Iter.	Tiempo	Índice	Iter.
CPLEX 6.0	Punto interior	41.8	1.0	32	237.3	1.0	35
	Simplex dual	99.8	1.4	12692	1812.6	6.6	48695
	Simplex primal	156.2	3.7	21622	1217.5	5.1	50280
MINOS 5.3	Simplex primal	1863.6	44.6	23927	–	–	–
OSL 2.1	Punto interior	163.9	3.9	10798	774.4	3.3	19524
	Simplex primal	530.9	12.7	12685	7426.6	31.3	62019

Tabla 1.4 Comparación entre diferentes optimizadores en problemas LP.

	Caso 1			**Caso 2**		
	Restricc.	Variables	Elementos	Restricc.	Variables	Elementos
Sin prep	19047	27847	82295	49715	64679	189477

Prep CPLEX	–14,8%	–19,3%	–36,2%	–17,9%	–13,2%	–28,6%
Prep OSL	–4,9%	0,0%	–2,4%	–15,6%	0,0%	–9,1%

Tabla 1.5 Comparación entre diferentes preprocesos.

Las diferencias en tiempo de resolución que pueden encontrarse entre métodos de optimización o entre implantaciones de un mismo método llegan a ser significativas (de hasta 45 veces para una comparación entre CPLEX 6.0 utilizando un método de punto interior y MINOS 5.3 utilizando el método simplex para un problema de 19000 restricciones, 28000 variables y 82000 elementos no nulos). Para un mismo método de optimización se han encontrado diferencias de hasta 3 veces entre implantaciones.

UTILIZACIÓN DE ÚLTIMAS VERSIONES

En general, las últimas versiones aportan mejoras de tiempo o funcionalidad con respecto a versiones previas.

En particular, una característica muy atractiva de los lenguajes de modelado es la posibilidad de actualizar la versión del optimizador o cambiar de optimizador sin necesidad de realizar modificaciones en el código del modelo. Ser consciente de ello y aprovecharlo forma parte de un uso avanzado del lenguaje.

AJUSTE DE PARÁMETROS DE CONTROL DEL OPTIMIZADOR

Habitualmente los parámetros de control de un optimizador toman unos valores por omisión generalmente adecuados para un problema estándar de optimización. Sin embargo, cuando se trata de problemas difíciles, como pueden ser los LP de muy gran tamaño o los NLP o MIP, son convenientes pruebas específicas de ajuste con algunos parámetros. En particular, algunos relacionados con la eficiencia y estabilidad numérica del algoritmo.

Los parámetros son propios de cada optimizador y también pueden serlo de cada método de optimización. Por mencionar algunos que pueden ser importantes en MINOS (linesearch tolerance, penalty, major iterations, minor iterations, factorization frequency) y en CPLEX (epopt, eprhs, epmrk).

Como consejo para evitar errores o confusiones es conveniente la creación de los ficheros de parámetros de control del optimizador dentro del código en lugar de editarlos manualmente.

USO DE SOLUCIONES INICIALES Y/O BASES PREVIAS

El uso de puntos iniciales es particularmente importante en el caso de problemas no lineales, donde se debe ejecutar un problema lineal cuya solución resulte cercana a la previsible solución del problema no lineal.

Cuando se trata de ejecuciones sucesivas es conveniente, desde el punto de vista de cálculo, aprovechar en el algoritmo del simplex las bases de soluciones previas del mismo problema u otro similar que haya sido resuelto previamente. La base contiene la información relativa a las variables primales y duales del problema. El aprovechamiento se controla con la opción bratio que marca un criterio de aceptación o rechazo de la misma.

Como ejemplo del impacto en el tiempo de optimización del aprovechamiento de la base, un problema LP de 8000 restricciones, 10000 variables y 30000 elementos requiere 10.3, 4.4, 4.7 y 2.6 segundos en sucesivas resoluciones.

A pesar de ello esta ventaja puede no ser suficiente para ciertos tamaños como para superar al método de punto interior. Por ejemplo, aproximadamente a partir de 20000 restricciones por 20000 variables el método de punto interior resulta más competitivo que el simplex aun comenzando éste con una base previa de un problema anterior.

DETECCIÓN DE INFACTIBILIDADES

Un método muy sencillo aunque laborioso y que puede producir problemas de consistencia y de dimensiones, es introducir variables de holgura en cada restricción y penalizarlas en la función objetivo. Este procedimiento hace las *restricciones elásticas*.

Alternativamente algunos optimizadores tienen un parámetro que detecta el núcleo menor de restricciones infactibles de un problema (parámetro *Irreducible Infeasible Subsets* iis) y, por consiguiente, ayudan a localizar su posible causa. Una vez conocidas el desarrollador debe modificar o eliminar alguna del conjunto para que el problema se haga factible. Un artículo reciente de John W. Chinneck sobre algoritmos para encontrar este conjunto mínimo es [Guieu, 1999]

ANÁLISIS DE SENSIBILIDAD

Proporciona información adicional sobre la solución de un problema de optimización lineal.

Algunos optimizadores permiten realizar directamente un análisis de sensibilidad a cambios en los coeficientes de la función objetivo que no producen una alteración de la base óptima o a cotas de las restricciones que no producen pérdida de factibilidad (parámetros objrng, rhsrng).

1.2.5 Referencias

Bixby, R.E., Fenelon, M., Gu, Z., Rothberg, E. and Wunderling, R. (2000) *MIP: Theory and Practice - Closing the Gap*. Technical Report.

Guieu, O. and Chinneck, J.W. (1999) "Analyzing Infeasible Mixed-Integer and Integer Linear Programs", INFORMS Journal on Computing, vol. 11, no. 1, pp. 63-77.

Fourer, R. (2003) "Linear Programming" *OR/MS Today*. December.

McCarl, B. A. (1998) *So Your GAMS Model Didn't Work Right. A Guide to Model Repair*. Technical Report.

Jacobs, J., Freeman, G., Grygier, J., Morton, D., Schultz, G., Staschus, K. and Stedinger, J. (1995) "SOCRATES: A system for scheduling hydroelectric generation under uncertainty" *Annals of Operations Research* 59. pp. 99-133.

Ramos, A., Muñoz, L., Rupérez, I., Martínez-Córcoles, F. and Martín-Corrochano, V. (1996) "Computational Experience with Optimization for a Bulk Production Cost Model" *12th PSCC*. Dresden, Germany.

2 Teoría de la decisión

En la vida real, y tanto en el ámbito profesional como el personal, nos vemos enfrentados a multitud de situaciones en las que tenemos que decidir entre varias alternativas. La propia optimización no es más que una forma de tomar una decisión entre unas alternativas factibles.

Así, en su dimensión más básica, un proceso de toma de decisión puede entenderse como la elección de lo "mejor" entre lo "posible". Ahora bien, según se defina qué es lo mejor y qué es lo posible nos enfrentaremos a distintas situaciones de decisión.

La optimización clásica tiene como característica general que lo mejor, el objetivo, es único y está claramente determinado (excepto en optimización multiobjetivo) y que lo posible, las soluciones factibles, no vienen expresadas explícitamente sino en forma de restricciones y sin incertidumbre (excepto en optimización estocástica, que no es precisamente clásica)

Pero además de estos contextos de decisión de optimización clásica, existen otros que configuran lo que se suele denominar en términos amplios la teoría de la decisión. Tres grandes bloques son los que se suelen abordar en este análisis:

a) La *teoría de la decisión con incertidumbre o riesgo*, en la que se analiza la toma de decisiones con aleatoriedad o incertidumbre en los resultados, de modo que las consecuencias de una decisión no están determinadas de antemano, sino que están sujetas al azar.

b) La *decisión multicriterio*, en la que si bien dada una decisión sus consecuencias están perfectamente determinadas, lo que no está definido tan claramente es qué es lo mejor, existiendo varios objetivos en conflicto.

c) La *teoría de juegos*, en la que las consecuencias de una decisión no dependen únicamente de la decisión adoptada, sino, también de la que elijan otros jugadores. En este contexto, los problemas de decisión con aleatoriedad del bloque anterior suelen ser denominados juegos frente a la naturaleza.

A continuación, se presenta una introducción a estos tres enfoques de decisión.

2.1 Teoría de la decisión con incertidumbre o riesgo

Una de las situaciones que más dificultad lleva a la hora de tomar una decisión es aquella en la que las consecuencias de las decisiones no pueden ser controladas, sino que están sujetas a la aleatoriedad; esta aleatoriedad puede provenir, tanto porque el proceso pueda estar gobernado por el azar, como por una falta de información que nos impida determinar con exactitud cuáles son esas consecuencias.

El contexto en que nos encontramos por lo tanto, es aquél en que el decisor ha de tomar una decisión ante una situación con diversos estados gobernados por el azar.

Los elementos que intervienen en un proceso de decisión de estas características son

- $E = \{E_1, ..., E_m\}$: conjunto de estados de la naturaleza o posibles escenarios.
- $A = \{A_1, ..., A_n\}$: conjunto de posibles alternativas o decisiones
- X_{ij} : consecuencia de tomar la decisión A_i y se dé el estado E_j

En ocasiones también intervienen las probabilidades a la hora de tomar una decisión:

- p_j : probabilidad de que se dé el estado E_j; este valor en muchas ocasiones no es conocido.

Si estas probabilidades son conocidas (o han sido estimadas) antes de tomar la decisión, se dice que es un proceso de *decisión bajo riesgo*, mientras que si son desconocidas se habla de *decisión bajo incertidumbre.*

Con estos elementos, cuando el proceso se define en una sola etapa, es decir, hay una única decisión que tomar en un momento dado, y los conjuntos de estados y alternativas son finitos, para facilitar la comprensión de la situación, se representa el problema mediante una *tabla de decisión*:

	E_1	E_2	$\cdots$	E_m
	p_1	p_2	$\cdots$	p_m
A_1	X_{11}	X_{12}	$\cdots$	X_{1m}
A_2	X_{21}	X_{22}	$\cdots$	X_{2m}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
A_n	X_{n1}	X_{n2}	$\cdots$	X_{nm}

A la matriz central formada por las consecuencias, se le suele denominar *matriz de pagos o consecuencias,* tomada esta denominación más del contexto de la teoría de juegos que de la decisión clásica.

2.1.1 Criterios para valorar las posibles decisiones

La mayor dificultad en este contexto es cómo valorar una decisión o alternativa para poder compararla con otras. Así se presentan distintos criterios para valorar las alternativas y, según sea el criterio adoptado, decidir cuál es la decisión óptima.

Los criterios se clasifican según utilicen las probabilidades de los distintos estados o no. Los primeros está claro que sólo pueden ser utilizados cuando estas probabilidades son conocidas, mientras que los segundos pueden ser aplicados en cualquier caso.

Criterios utilizando las probabilidades de los estados de la naturaleza.

- *Criterio del valor esperado:*

  Este criterio supone seleccionar aquella alternativa cuyo pago esperado o medio sea mejor (si los pagos son beneficios la de mayor beneficio esperado y si son costes la de menor coste esperado).

  Este criterio es el más común cuando las probabilidades son conocidas, pero no tiene por qué ser el más apropiado. Obsérvese que si el proceso de decisión se repite muchas veces en idénticas condiciones las leyes de los grandes números aseguran que en el límite el pago medio es la esperanza. Así pues este criterio es apropiado cuando el proceso se va a repetir muchas veces, pero puede no serlo cuando se presenta una situación única, en la que el proceso no va a ser repetido.

- *Criterio de lo más probable*

  Este criterio supone elegir la alternativa con mejor valor para el estado más probable, es decir, visto cuál es el estado más probable elegir la alternativa con mejor valor en ese estado.

  Este criterio se suele utilizar más cuando el proceso de decisión no es iterativo, es decir, se lleva a cabo una única vez.

- *Criterio del escenario medio*

  En ocasiones, cuando el espacio de estados es numérico, también es posible establecer un escenario medio y buscar aquella alternativa óptima para este escenario. Tiene sentido hacerlo sobre todo con distribuciones continuas (espacio de estados infinito). Si las consecuencias son proporcionales al estado, este criterio es equivalente al del valor esperado. No es un criterio muy aconsejable, pues, el escenario medio puede distar mucho de los escenarios reales, aunque en ocasiones se utilice para simplificar el procedimiento.

- *Criterio del valor en riesgo VaR (value at risk)*

  Este criterio es especialmente útil cuando el conjunto de estados de la naturaleza es continuo o al menos tiene un número de posibles escenarios muy elevado. Se basa en que normalmente el decidor siente aversión por el riesgo, mientras que para cantidades que superan un umbral las diferencias no le importan tanto. El VaR es un percentil de la distribución de la función de utilidad. Así si se trata del rendimiento de una inversión el VaR sería el percentil α, de modo que el $(100-\alpha)$% de las veces se espera superar ese valor, con lo que se elegirá la opción que tenga un VaR mayor (al revés si fueran costes). Habitualmente en finanzas se maneja el VaR como del 5%, y en ocasiones del 10%. Este criterio no suele usarse solo sino en compañía del criterio del valor esperado, entrando entonces en el mundo de la decisión multicriterio.

Criterios sin utilizar las probabilidades de los estados de la naturaleza

Estos criterios se utilizan cuando las probabilidades son desconocidas o ignoradas:

- *Criterio de Wald o minimax-maximin o pesimista*

  Para cada alternativa se supone que va a pasar lo peor, y elige aquella alternativa que dé mejor valor. De esta forma se asegura que en el peor de los casos se obtenga lo mejor posible, que corresponde a una visión pesimista de lo que puede ocurrir. En el caso de que los pagos sean costes esta filosofía supone elegir el mínimo de los máximos denominándose minimax, mientras que si son ganancias será el máximo de los mínimos, denominándose maximin.

- *Criterio optimista*

  Es el criterio justamente opuesto al anterior, para cada alternativa se supone que pasará lo mejor, y se elige la que dé mejor valor. Este criterio apenas es utilizado ya que no tiene en cuenta en ningún momento los riesgos que se corren al tomar una decisión.

- *Criterio de Hurwicz*

Este criterio combina las actitudes pesimista y optimista, valorando cada alternativa con una ponderación entre lo mejor y lo peor posible. Esta ponderación se hace multiplicando lo mejor por un factor α entre 0 y 1, denominado *índice de optimismo*, y lo peor por $1-\alpha$, sumando ambas cantidades. Se elegirá la alternativa que mejor valor dé. Este criterio presenta la dificultad de estimar el valor del índice de optimismo del decisor, de modo que habitualmente se obtiene la solución para todos los posibles valores de este índice y se intenta situar al decisor en alguno de los intervalos resultantes del índice de optimismo.

- *Criterio de Savage o costes de oportunidad*

  Este criterio toma en consideración el coste de oportunidad o penalización o arrepentimiento por no prever correctamente el estado de la naturaleza. Estos costes de oportunidad se evalúan para cada alternativa y cada estado, haciendo la diferencia entre lo mejor de ese estado y lo que proporciona esa alternativa para ese estado, construyendo la llamada matriz de penalizaciones o costes de oportunidad. Sobre esta matriz se aplican los criterios anteriores, pudiendo aplicarse el del coste esperado, o, lo que es más habitual, el criterio minimax conociéndose entonces también como criterio de minimizar el máximo arrepentimiento.

Vamos a ver un ejemplo para clarificar todos estos conceptos. Supóngase que en la demanda prevista para el mes siguiente de un determinado producto es 1, 2, 3 o 4, con probabilidades 0.1, 0.3, 0.4, y 0.2, respectivamente. Si un producto que es fabricado un mes se vende ese mismo mes el precio de venta será de 6500 euros, mientras que si ha de venderse el mes siguiente será de 4000. Los costes unitarios de producción son de 5000 euros.

Con estos datos se forma la matriz de decisión:

	$E_1=1$	$E_2=2$	$E_3=3$	$E_4=4$
	$p_1=0.1$	$p_2=0.3$	$p_3=0.4$	$p_4=0.2$
$A_1=1$	1500	1500	1500	1500
$A_2=2$	500	3000	3000	3000
$A_3=3$	-500	2000	4500	4500

$A_4 = 4$	-1500	1000	3500	6000

Las distintas alternativas seleccionadas para los distintos criterios serán:

- *Criterio de la ganancia esperada.*

Las esperanzas de ganancia son las siguientes: para A_1 1500, para A_2 2750, para A_3 3250 y para A_4 2750. Con lo que la decisión óptima es producir 3 artículos.

- *Criterio de lo más probable*

El estado más probable es E_3, y para este estado la mejor alternativa es pedir 3 artículos.

- *Criterio del escenario medio*

El escenario medio resulta ser : 1x0.1+ 2x0.3 + 3x0.4 + 4x0.2 = 2.7, cómo se ve no corresponde a ninguno de los estados ya que no es entero. Los beneficios serían para A_1 1500, para A_2 3000, para A_3 3750 y para A_4 2750, que como se ve no coinciden con los valores de la ganancia esperada. La alternativa elegida sería A_3.

- *Criterio de Wald.*

Los mínimos para cada decisión son 1500, 500, -500 y –1500, respectivamente, luego, la alternativa preferida sería producir 1 artículo.

- *Criterio optimista.*

En este caso los máximos son 1500, 3000, 4500, -1500, y por lo tanto, se elegiría producir 4 artículos.

- *Criterio de Hurwicz*

En este caso, para cada alternativa tenemos: A_1 1500, A_2 3000α+500(1-α), A_3 4500α-500(1-α) y A_4 6000α-1500(1-α). Si α<0.4, la alternativa sería producir 1 artículo, mientras que si es superior sería producir 4 artículos.

- *Criterio de Savage*

El primer paso es construir la matriz de penalizaciones o costes de oportunidad. La matriz la formamos por columnas, obteniendo el máximo de la columna y restándole a este valor el pago de cada alternativa. Así la matriz obtenida es:

0	1500	3000	4500
1000	0	1500	3000
2000	1000	0	1500
3000	2000	1000	0

Ahora aplicamos el criterio minimax, para minimizar la máxima penalización, obteniendo que los máximos son 4500, 3000, 2000 y 3000, por lo que la alternativa sería producir 3 artículos.

2.1.2 Valor esperado de la información perfecta (VEIP)

La idea en esta sección es que podría modificarse el conocimiento que se tiene acerca de los estados de la naturaleza. Esa modificación puede conllevar un coste y la pregunta es, ¿qué valor tiene disponer de esa información? ¿cuánto estamos dispuestos a pagar por ella? Hay que tener en cuenta que con mayor información la ganancia esperada será mayor.

Así, se define la ganancia esperada con información perfecta a la esperanza de la ganancia tomando para cada estado la mejor opción. Para el ejemplo de la sección anterior sería $0.1\cdot 1500+0.3\cdot 3000+0.4\cdot 4500+0.2\cdot 6000=4050$.

Por otro lado, se estima la ganancia esperada con incertidumbre, es decir, la ganancia esperada con la decisión que se haya tomado con alguno de los criterios anteriores. Así, si la decisión seleccionada es la A_3, la ganancia esperada es 3250.

Por último se define el valor esperado de la información perfecta, denotado por VEIP, a la diferencia entre ambas ganancias, es decir, la diferencia entre la ganancia esperada con información perfecta y la esperada con incertidumbre. Para el ejemplo sería VEIP= 4050 – 3250 = 800.

Obsérvese que es equivalente a utilizar el criterio de Savage con la mínima penalización esperada.

2.1.3 Procesos decisión polietápicos: Árboles de decisión

Otra situación muy habitual en un proceso de decisión es que no sea un proceso estático, sino dinámico, es decir, que sea un proceso secuencial de decisión-azar, donde secuencialmente se van tomando decisiones y va actuando el azar condicionando las decisiones posteriores. Esta situación es absolutamente análoga a la de la programación estocástica, y en particular, a la de la programación dinámica estocástica.

Para representan la aleatoriedad por etapas se suele utilizar los denominados *árboles de escenarios.* Sin embargo, para nuestro contexto no es suficiente representar la aleatoriedad, ya que las decisiones tomadas en el proceso condicionan lo que puede o no darse posteriormente, representándose en un *árbol de decisión* la secuencia de aleatoriedad y decisiones que conforman el proceso. Esta representación es apropiada cuando tanto el espacio de estados como el de las decisiones es discreto.

Para representar gráficamente en un árbol de decisión el proceso se utilizan los siguientes elementos y notación:

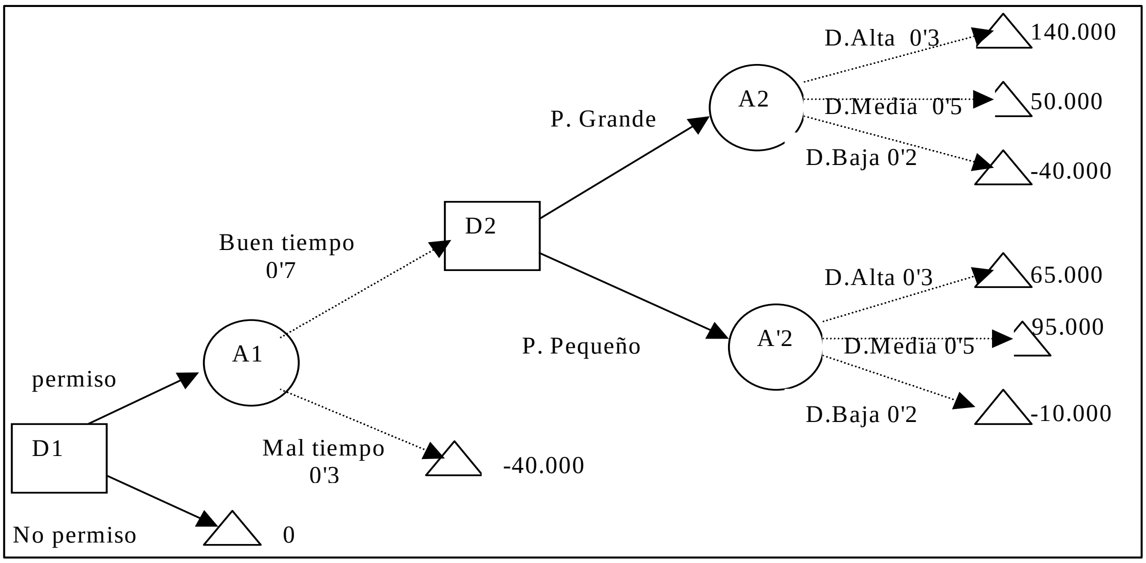
- Vértice de azar: son vértices que representan puntos en los que la naturaleza elige un estado. De estos vértices salen tantos arcos como estados de la naturaleza posibles hay en ese punto, y se representan mediante un círculo.
- Vértice de decisión: son vértices que representan puntos en los que hay que tomar una decisión. De ellos salen tantos arcos como alternativas posibles hay en ese punto, y se representan mediante un cuadrado.
- Vértice inicial o raiz: es la raíz del árbol, de donde salen tantos arcos como decisiones iniciales hay, ya que en un proceso de estas características lo primero es tomar una decisión.
- Vértice terminal u hoja: son los vértices finales de una rama que es sucesión de estados y decisiones. Se les asigna el coste o beneficio (según sea la función objetivo a evaluar) del camino seguido para llegar a él, y se representan por un triángulo.

El árbol se construye de la raíz a las hojas, mostrando el proceso secuencial que es seguido. Una vez acabado se valora de las hojas a la raíz de la siguiente forma:

- Nodos de azar: se valoran con alguno de los criterios mostrados para valorar decisiones, en general, suele ser el del valor medio, pero, no tiene por qué ser así.
- Nodos de decisión: se valoran eligiendo la mejor decisión según el criterio considerado. Las decisiones no seleccionadas se consideran rechazadas y con ello todos los caminos que salgan de ese arco.

Veamos el siguiente ejemplo. Un vendedor ambulante ha de decidir en el mes de Enero si va a acudir a una feria que se celebra en el mes de septiembre, porque en el caso en que sea que sí tendrá que pagar 40.000 euros de licencia para poder montar su puesto en la feria. Un mes antes de la feria se conocen las previsiones meteorológicas para el mes de septiembre, de modo que si estas son de mal tiempo el vendedor sabe que lo más rentable es no ir a la feria. Por experiencias de años anteriores sabe que el 30% de las veces estas previsiones son de mal tiempo. En el caso de que sean buenas las previsiones, el vendedor decidirá ir a la feria y hacer el pedido de productos a vender. Puede hacer dos tipos de pedido: un pedido grande de 900 unidades, con un precio unitario de compra de 100 euros y un precio de venta de 300 euros, o un pedido pequeño de 600 unidades que comprará a 125 euros y venderá a 350. Una vez en la feria, este vendedor estima que la demanda puede ser de tres tipos: demanda alta de 900 unidades, media de 600 unidades, y baja de 300 unidades, con probabilidades 0.3, 0.5 y 0.2, respectivamente. Sin embargo, se da la circunstancia de que si la demanda es mayor que la cantidad de productos que ha llevado el precio de venta se verá reducido en 50, en concepto de penalización.

Este problema es claramente un proceso secuencial de decisiones y aleatoriedad que puede ser representado gráficamente mediante el siguiente árbol de decisión:



La valoración de los nodos se hace como sigue:

- Nodo A2: habrá que valorarlo según alguno de los criterios vistos, por ejemplo si utilizamos el del valor medio se valorará con 59.000, si fuera el de Wald con –

40.000. En cualquier caso hay que definir uno de ellos para todos los nodos, así que lo desarrollaremos según el criterio del valor medio.

- Nodo A'2: puesto que se ha decidido hacerlo según el criterio del valor medio, será 65.000.
- Nodo D2: al ser un nodo de decisión se valora con el mejor valor de los nodos conectados a él, es decir, el mejor de A2 y A'2, con 65.000
- Nodo A1: es un nodo de azar, luego se valora por el criterio adoptado con la media del nodo 45.500.
- El otro nodo es un nodo terminal valorado con 0.
- Nodo D1: es un nodo de decisión en el que se elige entre 45.500 y 0, con lo que resulta valorado 3n 45.500. Además es el nodo inicial, con lo que ya se ha valorado todo el árbol.

Así la política óptima resulta ser pedir el permiso en Enero, y si hace buen tiempo ir con pedido pequeño. Esta política óptima nos da una ganancia esperada de 45.500 euros.

En bastantes situaciones es posible incorporar información al árbol de decisión de manera parcial. Para llevar a cabo esta incorporación ha de utilizarse el *análisis bayesiano*. Este análisis modifica las probabilidades de los escenarios en función de la nueva información de que se dispone.

El análisis bayesiano se fundamenta en los dos resultados de probabilidad más utilizados en el cálculo de probabilidades condicionadas: el teorema de Bayes y el teorema de la probabilidad total.

El teorema de la probabilidad total permite calcular la probabilidad de un suceso a partir de las probabilidades de este suceso condicionadas a otros sucesos que formen un sistema completo, es decir, cuya unión sea todo el espacio muestral y sean disjuntos. Así si $\{B_1,...,B_n\}$ son tales que $\bigcup B_i = \Omega$ y $B_i \cap B_j = \varnothing \quad \forall i \neq j$, para cualquier suceso (no necesariamente sobre el mismo espacio) se tiene

$$P(A) = \sum_i P(A/B_i) \cdot P(B_i)$$

Por otra parte el teorema de Bayes establece la probabilidad de un suceso A condicionada a la ocurrencia de otro suceso, cuando la información de la que se dispone es del condicionado inverso:

$$P(B/A) = \frac{P(A/B)\cdot P(B)}{P(A)}$$

Supóngase que en el ejemplo anterior, el vendedor dado la suma tan alta que está en juego busca desesperadamente alguien que pueda darle información adicional en Enero acerca de lo que va a ocurrir con el clima en septiembre. Al fin encuentra a un experto meteorólogo que le dice que él puede hacer una predicción por 10.000 euros de nada, y que ya lo ha hecho más veces con los siguientes resultados:

- cuando hizo bueno acertó 3 de cada 5 veces
- cuando hizo malo acertó 2 de cada 5 veces.

Al vendedor entonces se le presenta una decisión previa a decidir si va a pedir el permiso que es si consultar al experto. Así ahora el nodo inicial es otro del que salen dos arcos, uno que es consultar al experto y otro que es no hacerlo. En el caso de que no le consulte todo se mantiene igual exactamente que antes, con lo que el nodo que sale de ahí ya estaría valorado: 45.500. Quedaría por valorar el otro nodo, pero hay que tener en cuenta que tras la consulta viene un nodo de azar acerca del resultado de la consulta, y que además según sea este resultado modifica las posteriores probabilidades de ocurrencia de buen tiempo y mal tiempo (todo ello fiándonos de lo que dice el experto, claro está).

Así, primero necesitamos obtener las probabilidades de lo que dirá el experto.

Empecemos por recopilar y denotar la información disponible. Llamaremos B al suceso hace buen tiempo en septiembre y M al suceso hace mal tiempo. Llamaremos DB al suceso el experto dice que va a hacer buen tiempo, y DM al suceso el experto dice que va a hacer mal tiempo.

La información de la que disponíamos ya es $P(B) = 0.7$ y $P(M) = 0.3$. Por otra parte, ahora nos dicen $P(DB/B) = 3/5 = 0.6$ y $P(DM/M) = 2/5 = 0.4$, de los que se deducen por complementariedad, $P(DM/B) = 1-3/5 = 0.4$ y $P(DB/M) = 1-2/5 = 0.6$. De estos datos, aplicando el teorema de la probabilidad total podemos obtener directamente la probabilidad de que el experto diga que hará bueno o que diga que hará malo:

$$P(DB) = P(DB/B)P(B) + P(DB/M)P(M) = 0.6\times 0.7 + 0.6\times 0.3 = 0.6$$

$$P(DM) = P(DM/B)P(B) + P(DM/M)P(M) = 0.4\times 0.7 + 0.4\times 0.3 = 0.4$$

Este último valor podía haberse obtenido directamente por complementariedad.

Estas serán las probabilidades de los arcos que salen del nodo del experto, pero según sea una u otra afectará a las probabilidades que habrá que poner en los arcos posteriores cuando se llegue al nodo de la previsión meteorológica de septiembre, ya que las probabilidades han de ser probabilidades condicionadas a los valores que la aleatoriedad haya tomado previamente. Así en los arcos posteriores al que dice que el experto dice que hará bueno, las probabilidades que hay que situar son $P(B/DB)$ y $P(M/DB)$, mientras que en las que derivan del estado dice el experto que hará malo serán condicionadas al suceso DM. Por lo tanto, hay que calcular estas probabilidades condicionadas ya que no las tenemos y para ello se utiliza el teorema de Bayes:

$$P(B/DB) = \frac{P(DB/B)P(B)}{P(DB)} = \frac{0.6 \times 0.7}{0.6} = 0.7$$

$$P(M/DB) = \frac{P(DB/M)P(M)}{P(DB)} = \frac{0.6 \times 0.3}{0.6} = 0.3$$

$$P(B/DM) = \frac{P(DM/B)P(B)}{P(DM)} = \frac{0.4 \times 0.7}{0.4} = 0.7$$

$$P(M/DM) = \frac{P(DM/M)P(M)}{P(DM)} = \frac{0.4 \times 0.3}{0.4} = 0.3$$

Estas probabilidades son las que habría que poner entonces en las ramas correspondientes y valorar de nuevo toda esta parte del árbol. Sin embargo, en este caso no es necesario, ya que resulta evidente que las probabilidades de que haga buen o mal tiempo no se ven afectadas por lo que diga el experto, de modo que éste no es más que un embaucador que pretende engañar al vendedor y aprovecharse de su buena fe. Suerte que nuestro vendedor sabe de cálculo de probabilidades y de teoría de la decisión.

2.1.4 Utilidad: concepto y funciones de utilidad

Por último, dentro de esta introducción a la teoría de la decisión vamos a hablar de un concepto muy utilizado en este contexto, la utilidad.

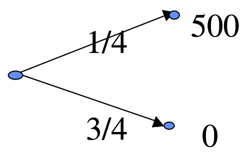
En lo que hemos visto siempre hemos supuesto que las alternativas tienen pagos cuantificables, sin embargo, no siempre es cierto. Valoraciones de tipo calidad, prestigio, etc. no son valores numéricos. Por otra parte, aunque los pagos sean numéricos el valor que nosotros damos a una cantidad, que es personal, no siempre es proporcional a ella. Obsérvese que para una persona determinada y en un contexto determinado, no es lo mismo ganar de repente un millón de euros teniendo un capital de 10 euros, que disponiendo ya de 100 millones de euros ganar ese millón. O visto de forma negativa, no es lo mismo perder un millón cuando no tienes nada, que perderlo cuando tienes 100 millones.

Para reflejar esta valoración de los pagos, no por el pago en sí, sino por el valor que tiene éste para el decisor, se utiliza la *utilidad*, que es una valoración personal de una cantidad que no varía proporcional al importe de esa cantidad.

La utilidad se plasma en la *función de utilidad* que es una función que resume la importancia que esa persona asocia a diferentes cantidades. Se trata de un índice o escala personal del decisor, y por lo tanto, diferente para cada persona que lo plantee, que ha de ser no decreciente (a mayor importe, mayor utilidad).

La formalización de lo que ha de cumplir una función para poder ser una función de utilidad, se hace axiomáticamente. No hay una única forma de establecer los axiomas que ha de cumplir una función de utilidad. A continuación, se presenta la axiomática planteada por Von Neumann y Morgenstern cuando introdujeron este concepto. Para ello definiremos lo que son las loterías y la relación de equivalencia entre ellas, para poder definir la relación de preferencia que ha de reflejar una función de utilidad.

Una lotería viene dada por un conjunto de valores posibles de ganancia y unas probabilidades. Formalmente se describe como $(p_1, r_1; ...; p_n, r_n)$, donde p_i denota la probabilidad de obtener una ganancia r_i. Por ejemplo, dada una lotería que gráficamente sería



se denotaría por (1/4, 500; 3/4, 0).

Se define la relación $L_1 p L_2$ si la lotería L_1 es preferida a la lotería L_2, y la relación de equivalencia $L_1 i L_2$ si la lotería L_1 es indiferente a la lotería L_2, es decir, son equivalentes.

Dadas estas definiciones, podemos definir formalmente la utilidad de una cantidad r_i en función de una lotería y que denotaremos por $u(r_i)$ como el valor q_i tal que es equivalente recibir con seguridad la cantidad r_i, que sería la lotería $(1, r_i)$ a la lotería en que se recibe con probabilidad q_i el valor más favorable de la lotería de referencia y con probabilidad $1-q_i$ el resultado menos favorable.

La función de utilidad sería la función que especifica todos los pagos de la lotería, y la utilidad esperada de una lotería sería $\sum_i p_i u(r_i)$. Así pues, las alternativas de un problema de decisión se pueden ver como loterías, ya que se tienen unos pagos con ciertas probabilidades cuando es decisión bajo riesgo.

Dadas estas definiciones, se pueden mostrar los axiomas que Von Neumann y Morgestern establecieron para que una relación de preferencia/indiferencia pueda definir una función de utilidad.

Axiomas de Von Neumann y Morgestern

De ordenación completa

Dados r_1 y r_2 r2 se cumple: $r_1 p r_2$ o $r_2 p r_1$ o $r_1 i r_2$. Además esta ordenación ha de cumplir la propiedad de transitividad ($r_1 p r_2$ y $r_2 p r_3$ implica que $r_1 p r_3$).

De continuidad

Si $r_1 p r_2$ y $r_2 p r_3$ entonces existe c tal que $(1, r_2) i (c, r_1; 1-c, r_3)$

De independencia

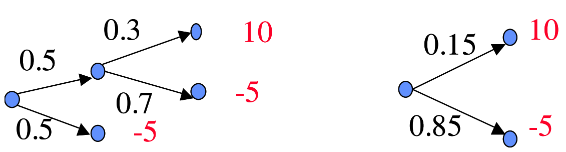
Si $r_1 i r_2$ entonces $\forall c \in (0,1)$ son indiferentes $(c, r_1; 1-c, r_3)$ y $(c, r_2; 1-c, r_3)$

De probabilidad desigual

Si $r_1 p r_2$ entonces $(c, r_1; 1-c, r_2) p (c', r_1; 1-c', r_2)$ si $c > c$'

De probabilidad desigual

Dada una lotería compuesta existe una equivalente simple. Por ejemplo, son equivalentes las siguientes dos loterías:



La función de utilidad aunque no vaya entre 0 y 1 puede ser transformada a este rango. Sea $u(x)$ una función de utilidad, y sea $v(x) = a\,u(x) + b$ ($a > 0$). Entonces:

• $L_1 p L_2$ usando $u(x)$ si y sólo si $L_1 p L_2$ usando $v(x)$

• $L_1 i L_2$ usando $u(x)$ si y sólo si $L_1 i L_2$ usando $v(x)$

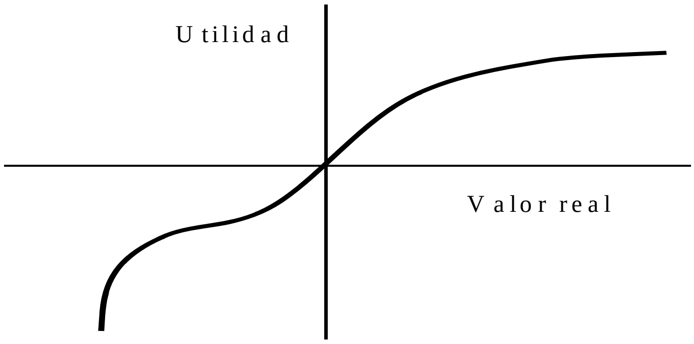
De modo que si la función de utilidad de un individuo no va entre 0 y 1, se puede transformar a este rango utilizando la función de utilidad $v(x) = \dfrac{x - \min(u(x))}{\max(u(x)) - \min(u(x))}$.

Por último, restaría saber como estimar la función de utilidad de un individuo. Para ello, se propone ir comparando loterías, de modo que en unas se obtenga un valor seguro y en otras ciertos valores con determinadas probabilidades, o viceversa. Por ejemplo, se parte de $u(\min(x)) = 0$ y $u(\max(x)) = 1$. A continuación, para saber qué valor tendrá utilidad ½, se pide que el individuo diga cuál es la cantidad en que para él es indiferente recibir esa cantidad con seguridad frente a jugar una lotería en que puede obtener el máximo con probabilidad ½ y el mínimo con probabilidad también ½ (es decir, sería la lotería $(1/2, \max(x); 1/2, \min(x))$). A continuación, hacer lo mismo con el de ¼ (pero utilizando en la lotería de comparación en lugar de el máximo el valor obtenido antes de utilidad ½), con el de ¾,... y así sucesivamente, hasta describir la función de utilidad.

Una función de utilidad de un individuo ha de representar la conducta ante el riesgo de este individuo. Para ello, se define el *equivalente de certeza* de una lotería ($CE(L)$) como el valor en que es indiferente ese valor seguro a la lotería L, y v*entaja de riesgo* ($RP(L)$) a la diferencia entre el valor esperado de la lotería y su equivalente de certeza, es decir, $E[L] - CE(L)$. Según sea el signo de este valor supondrá que hay preferencia por el riesgo o por el contrario aversión al riesgo. Más concretamente, se puede resumir en los siguientes tres tipos de actitud ante el riesgo:

- Contrario a los riesgos o aversión al riesgo: $RP(L) > 0$ (la función de utilidad sería cóncava)
- Neutral frente a riesgos: $RP(L) = 0$ (la función de utilidad sería una línea recta)
- Preferencia por el riesgo: $RP(L) < 0$ (la función de utilidad sería convexa)

Una gráfica de una posible función de utilidad es la que se tiene a continuación. Las zonas cóncavas (en la gráfica las extremas) suponen zonas de aversión al riesgo ya que dados dos puntos la cuerda que los une, que representaría la neutralidad queda por debajo, lo que supone valorar más la cantidad que su propio valor. Sin embargo las zonas convexas representan preferencia por el riesgo ya que valoran por debajo las cantidades.



Es muy habitual aplicar todos los conceptos de decisión vistos previamente con utilidades más que con las cantidades reales para reflejar claramente el carácter del decisor.

2.2 Decisión multicriterio

2.2.1 Introducción: conceptos básicos

Se puede decir que un problema general de decisión consiste en elegir lo mejor entre lo posible. Esta definición, como ya se ha visto, implica definir qué es lo mejor y qué es lo posible.

Respecto a lo posible, se trata de establecer las alternativas o puntos factibles existentes. El conjunto puede ser discreto o continuo. En general, se considera discreto y se aplica la metodología apropiada cuando es factible enumerar y tratar explícitamente cada uno de las alternativas posibles. En el caso continuo o caso discreto donde no viene explícitamente definido el conjunto de alternativas es cuando se habla de conjunto o región factible. Este conjunto o región factible, a su vez, puede venir definido de forma rígida mediante restricciones o de forma más flexible mediante lo que se conoce como *niveles de aspiración.*

Respecto a lo mejor, se puede definir según un único criterio o según varios criterios. Los problemas de decisión con un único criterio y conjunto factible continuo (entendiendo por éste la extensión a conjuntos discretos no definidos explícitamente) son básicamente problemas de optimización "clásica": lineal, entera o no lineal. Si además incluyen aleatoriedad, estaríamos ante un problema de optimización estocástica. Si el conjunto factible es discreto, sólo tiene sentido plantearse el problema en el caso de que haya aleatoriedad, siendo entonces un problema de los conocidos como problemas clásicos de decisión[9] y visto en la sección anterior.

En el caso en que haya varios criterios, si la región factible es continua, se puede resolver el problema mediante métodos denominados de *optimización multiobjetivo* o mediante *métodos satisfacientes* (*programación por metas*). Si lo posible viene definido por un conjunto discreto de alternativas (pudiendo incluso no ser numérico el valor de los criterios), existen métodos multicriterio discretos para resolver el problema.

De forma general un problema de decisión multicriterio vendría formulado de la siguiente forma:

$$\begin{aligned}&\text{opt}\ z(x) = (z_1(x), ..., z_p(x))\\ &x \in F\end{aligned}$$

donde F es el espacio de decisiones o soluciones (si es continuo, se denomina región factible, $F \subseteq \mathbb{R}^n$).

Al conjunto $z(F)$ se le denomina *espacio de objetivos* o *resultados* (en el caso de que sean criterios numéricos $z(F) \subseteq \mathbb{R}^p$).

A continuación se exponen algunos conceptos básicos de la decisión multicriterio:

Atributo: "valor" observado (medido) de una decisión independientemente del decisor. Los atributos suelen ser competetidores o contradictorios entre sí.

Objetivo: dirección de mejora de un atributo. Esta dirección será de maximización o minimización en el caso de atributos numéricos y en el caso de atributos no numéricos vendrá dado por un sistema de preferencias (por ejemplo, si el problema fuera la selección de un automóvil, el color sería un atributo no numérico y se establecería un sistema de preferencias sobre este atributo).

Nivel de aspiración: es un nivel aceptable de logro para un atributo.

[9] No tiene sentido sin aleatoriedad, ya que si el conjunto está definido explícitamente y no hay aleatoriedad, basta una exploración de las alternativas para ver cuál es la mejor.

Meta: es la combinación de un atributo con su nivel de aspiración.

Criterio: son los atributos, objetivos o metas relevantes en un problema de decisión.

Dadas estas definiciones previas, lo primero que se necesita en un problema de decisión multicriterio es dar un concepto de solución. El concepto de solución utilizado habitualmente es el concepto de óptimo de Pareto. Este concepto está basado en el *criterio de optimalidad paretiana*, enunciado por Pareto en 1896:

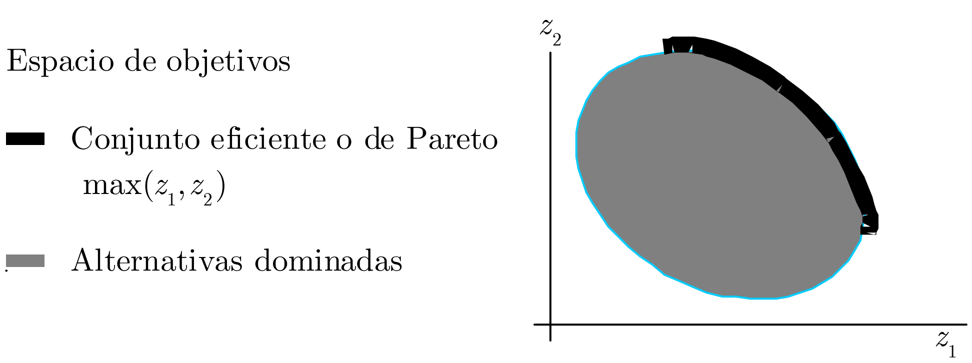
"Una alternativa es *eficiente* (o *Pareto óptima*) si toda alternativa que proporcione una mejora en un atributo produce un empeoramiento en al menos otro de los atributos."

De esta definición, se deriva la de *alternativa dominada* o *no eficiente*, como una alternativa para la que existe otra alternativa con todos los atributos mejores.

Así se define el *conjunto eficiente* o *de Pareto* o también llamada *frontera de Pareto* (por su representación en el caso continuo como parte de la frontera del espacio de objetivos o resultados) para atributos numéricos con objetivos de maximización como

$$\varepsilon = \left\{ x \in F : \nexists x' \in F / z_k(x') \geq z_k(x)\ \forall k \text{ y } z_t(x') > z_t(x) \text{ para al menos un } t \in \{1, \ldots, p\} \right\}$$

Un ejemplo gráfico para dos atributos podría ser el siguiente:



Sin embargo, en la mayoría de las situaciones, el fin último es dar una única solución, no un conjunto de posibles soluciones. Se denomina *solución de mejor compromiso* a la solución del conjunto eficiente que es seleccionada por el decisor.

A continuación, vamos a ver diferentes métodos usados en decisión multicriterio continua. Básicamente hay dos enfoques dentro de estos métodos: los métodos de optimización multiobjetivo y los métodos satisfacientes o programación por metas.

2.2.2 Métodos de optimización multiobjetivo

Estos métodos buscan optimizar los objetivos satisfaciendo unas restricciones rígidas que determinan la región factible. El planteamiento del problema sería

$$\max z = (z_1(x), ..., z_p(x))$$
$$x \in F$$

donde $z_i(x)$ es la función matemática que describe el atributo i-ésimo, $x \subseteq \mathbb{R}^n$ es el vector de variables de decisión y F es el conjunto de restricciones que definen las posibles soluciones.

Dentro de los métodos de optimización multiobjetivo existen métodos para generar el conjunto eficiente en su totalidad y métodos para dar una solución compromiso.

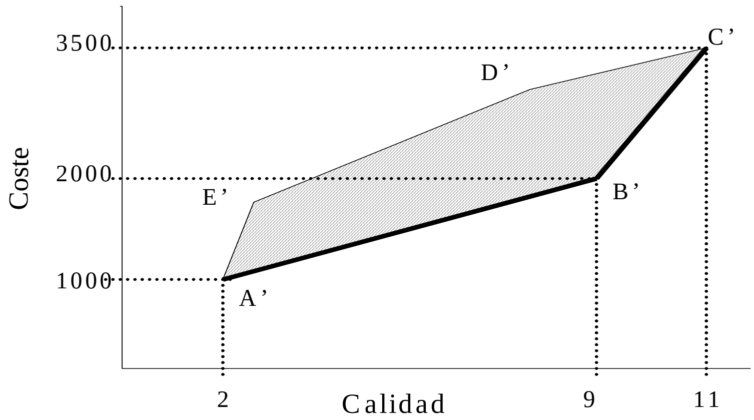
Antes de ver estas técnicas se van a presentar dos conceptos que serán de gran utilidad para comprender e interpretar el problema planteado: la matriz de pagos y las tasas de intercambio.

La *matriz de pagos* (pay-off matrix) es una matriz donde se representan por filas el valor óptimo de un objetivo sin considerar el resto de objetivos (resolviendo el problema independientemente), y los valores que resultarían para los demás objetivos con esa solución. Esta matriz representa el grado de conflicto que hay entre los objetivos propuestos.

Por ejemplo, supóngase un problema en que hay que determinar la composición de una mezcla a partir de unas ciertas componentes básicas. Supóngase que se desea que el coste unitario de la mezcla sea mínimo pero que la calidad sea máxima, medida ésta como una función de las componentes. Se tendrían entonces dos objetivos contrapuestos: minimizar el coste y maximizar la calidad. La matriz de pagos sería de la forma

	Coste	Calidad
Coste	m_{11}	m_{12}
Calidad	m_{21}	m_{22}

donde m_{11} es el mínimo coste que se puede lograr con las restricciones impuestas en el problema y m_{12} es la calidad para la composición que da el mínimo coste; en la siguiente fila sería al revés, es decir, m_{22} sería la máxima calidad que se puede lograr de la mezcla, y m_{21} sería el coste de esa mezcla. Así, si el espacio de objetivos fuera el siguiente, donde el conjunto eficiente está marcado en negrita,



la matriz de pagos sería

	Coste	Calidad
Coste	**1000**	2
Calidad	3500	**11**

Por otra parte, las *tasas de intercambio* (trade-offs o costes de oportunidad) entre los atributos representan lo que se está dispuesto a empeorar de un objetivo por mejorar en una unidad otro objetivo. Serían las pendientes de los segmentos que forman el conjunto eficiente. Así, en el segmento A'B' la tasa de intercambio entre el coste y la calidad será $T_{A'B'} = \frac{2000-1000}{9-2} = 142.86$, es decir, en ese segmento cada unidad más de calidad "cuesta" 142.86 unidades. Análogamente, para el segmento B'C' la tasa será $T_{B'C'} = \frac{3500-2000}{11-9} = 750$. Es decir, cada unidad más de calidad "cuesta" 750 unidades monetarias.

2.2.2.1 Técnicas generadoras del conjunto eficiente

Son técnicas de carácter mecánico, en las que no se incluyen las preferencias del decisor. El fin para el que están diseñadas es la obtención de todo el conjunto eficiente, en general, tras aplicar métodos de programación paramétrica.

Método de las ponderaciones [Zadeh, 1963]

Este método consiste en multiplicar cada objetivo por un peso o factor no negativo y agregarlos en una única función. Variando los pesos se puede obtener todo el conjunto eficiente, resolviendo los distintos problemas planteados mediante programación paramétrica.

$$\left.\begin{array}{l} \max \sum_{i=1}^{p} \lambda_i z_i(x) \\ x \in F \\ \lambda \geq 0 \end{array}\right\} P(\lambda)$$

Uno de los resultados en que se fundamenta este método dice que si $\lambda_i > 0 \quad \forall i$, entonces cualquier solución óptima del problema $P(\lambda)$ es eficiente. El recíproco es cierto sólo bajo ciertas condiciones (por ejemplo, si todas las funciones objetivo y las restricciones son lineales).

En cualquier caso, hay que tener en cuenta que para aplicar este método (y no sólo éste) es conveniente haber normalizado previamente los criterios (para que no influya la diferencia de unidades de los criterios).

Método de las ε-restricciones [Marglin, 1967]

Consiste en optimizar uno de los objetivos e incorporar el resto como restricciones paramétricas, resolviendo el problema resultante mediante programación paramétrica (variando los términos de la derecha se obtiene todo el conjunto eficiente).

$$\left.\begin{array}{l} \max z_l(x) \\ x \in F \\ z_k(x) \geq \varepsilon_k \quad k = 1, \ldots, l-1, l+1, \ldots, p \end{array}\right\} P_l(\varepsilon)$$

El fundamento de este procedimiento se recoge en los dos resultados siguientes:

Teorema: Si la solución del problema $P_l(\varepsilon)$ es única, entonces es una solución eficiente.

Teorema: Si x^* es eficiente, $\forall l \;\; \exists \varepsilon_k$ tales que x^* es solución óptima de $P_l(\varepsilon)$.

Método simplex multiobjetivo [Zeleny, 1973]

Este método sólo es aplicable para objetivos y restricciones lineales. Se trata de una extensión del método simplex que evalúa en cada iteración la eficiencia de las soluciones básicas obtenidas (puntos extremos), obteniendo así todos los puntos extremos eficientes. El conjunto eficiente serán todas las combinaciones lineales convexas de puntos extremos eficientes que sean adyacentes.

2.2.3 Programación compromiso

En la programación compromiso el fin último es seleccionar un punto del conjunto eficiente y, para ello, naturalmente, ha de incluir las preferencias del decisor. Las técnicas que se presentan fueron desarrolladas inicialmente en los trabajos de Yu(1973) y Zeleny (1973, 1974).

Se define el *punto* o *alternativa ideal*, como un punto del espacio de objetivos que recoge los valores óptimos de los objetivos individualmente tratados y se denota por $z^* = (z_1^*, ..., z_i^*, ..., z_p^*)$, siendo $z_i^* = \max_{x \in F} z_i(x)$, denominándose este valor individual *punto ancla*.

Se define la *solución óptima* o *mejor solución compromiso*, como la solución eficiente más próxima al punto ideal (axioma de [Zeleny, 1973]).

La cuestión ahora es definir una distancia que ha de ser minimizada sobre el conjunto eficiente. Para ello se define el grado de proximidad del objetivo i-ésimo normalizado como $d_i(x) = \dfrac{z_i^* - z_i(x)}{z_i^* - z_{*i}}$, siendo z_{*i} el antiideal del objetivo, es decir, el peor valor posible para el objetivo sobre el conjunto eficiente.

Estos grados de proximidad son agregados para definir una métrica:

$$\min_{x \in F} L_\pi = \left[\sum_{i=1}^{p} w_i^\pi \left(\frac{z_i^* - z_i(x)}{z_i^* - z_{*i}} \right)^\pi \right]^{1/\pi}$$

donde el conjunto de los pesos w_i suponen una ponderación preferencial de los criterios (ponderación subjetiva dada por el decisor, este sistema de ponderación se puede intentar obtener del decisor por varios métodos descritos en la literatura, como son el método de ordenación, el método de Saaty, ...).

Para los distintos valores de π se obtienen diferentes métricas. Así $\pi = \infty$, denota la distancia de Tchebychev o lo que es lo mismo minimizar la máxima distancia (así se obtiene una solución equilibrada, en que todos los grados de proximidad individuales están acotados, siendo a su vez un problema lineal).

Para el caso $\pi = 1$, se trata de una agregación lineal de distancias ponderadas, que también mantiene el carácter lineal del problema.

En general, para diferentes valores de π se obtiene diferente solución, de ahí que se haya definido el *conjunto compromiso*, como el conjunto de soluciones que se obtienen al variar π. Para el caso de dos objetivos, es usual dar el conjunto definido por el segmento $\left[L_1, L_\infty\right]$ (es decir, el segmento que une las soluciones obtenidas con las dos métricas comentadas anteriormente), ya que bajo ciertas condiciones es cierto que el conjunto compromiso coincide con este segmento.

2.2.4 Métodos satisfacientes: programación por metas

Los métodos satisfacientes se basan en la denominada lógica satisfaciente enunciada por Simon en 1955:

"en los contextos actuales de decisión (información incompleta, recursos limitados, conflictos de intereses, ...) el decisor más que optimizar unas funciones objetivo intenta que una serie de metas se aproximen lo más posible a unos niveles de aspiración prefijados."

De ahí que se desarrollara, basándose en esta filosofía, la denominada *programación por metas* ([Charnes y Cooper, 1961], [Lee, 1972] e [Ignizio, 1976]).

La idea de este método es que si un atributo tiene por expresión matemática $z_i(x)$ y para este atributo se tiene un *nivel de aspiración*, es decir, un nivel aceptable de logro para un atributo, $\hat{z}_i$, entonces una *restricción meta* sería una restricción de la forma $z_i(x) \geq \hat{z}_i$.

Sin embargo, dado que son niveles de aspiración, una restricción meta, en general, no puede ser incluida como una restricción rígida en el problema, ya que soluciones que no logren ese nivel de aspiración también son soluciones admisibles. Así, una restricción meta se formula utilizando *variables de desviación*: $z_i(x) + n_i - p_i = \hat{z}_i$. De estas variables, una de ellas es una variable no deseada: si la meta es "al menos" un valor, la variable no deseada será n_i, y si es del tipo "a lo sumo", la variable de desviación no deseada será p_i.

Así el problema a resolver, suponiendo que todos los objetivos fueran de tipo maximizar sería:

$$\min_{x\in F\cap \text{restricciones meta}} \sum_{i=1}^{p} n_i$$

Este procedimiento se puede aplicar también a las restricciones del problema para relajarlas, admitiendo soluciones "cercanas" a la región factible (el mismo concepto se ha aplicado en lógica difusa para resolver problemas de programación lineal).

La programación por metas aplicada en la realidad muestra que se están obteniendo valiosos resultados, siendo una de las técnicas de decisión multicriterio que está proporcionando mejores resultados. Sin embargo, no está exenta de detractores que critican aparentes debilidades del modelo que se presentarán más adelante.

Algunas variantes de este método se han desarrollado con gran éxito y se exponen a continuación.

2.2.4.1 Variantes de la programación por metas

Programación por metas ponderadas

$$\begin{aligned}
&\min \sum_{i=1}^{p} (\alpha_i n_i + \beta_i p_i) \\
&z_i(x) + n_i - p_i = \hat{z}_i \quad i = 1, \ldots, p \\
&x \in F \\
&n_i \geq 0,\ p_i \geq 0 \quad i = 1, \ldots, p
\end{aligned}$$

La idea básica es ponderar las variables de desviación, ya que pueden tener diferente relevancia las metas o diferentes unidades si no han sido previamente normalizados los criterios.

Los distintos criterios, en general, vendrán dados en distintas unidades, así que la función objetivo estaría agregando valores de distintas unidades. Una primera medida, muy deseable, es ponderar las desviaciones dividiendo por el nivel de aspiración, con lo que serían desviaciones porcentuales que no tienen unidades y que corrigen el efecto de las distintas magnitudes de éstas.

Por otra parte, si sólo se ponderan dividiendo por el nivel de aspiración, implícitamente lo que se está haciendo es dar una misma importancia a todos los criterios. Si no es ése el caso, se deben multiplicar por pesos que muestren la relevancia dada por el decisor a cada meta.

Programación por metas MINIMAX o Tchebychev

En este caso se busca una solución "equilibrada", de modo que ninguna de las metas se desvíe en exceso de su nivel de aspiración. Para ello se minimiza la máxima distancia a este nivel, es decir, el problema se plantearía en estos términos

$$\begin{aligned}
&\min d \\
&\alpha_i n_i + \beta_i p_i \leq d \quad i = 1, \ldots, p \\
&z_i(x) + n_i - p_i = \hat{z}_i \quad i = 1, \ldots, p \\
&x \in F \\
&n_i \geq 0,\ p_i \geq 0 \quad i = 1, \ldots, p
\end{aligned}$$

Programación por metas lexicográficas

Este procedimiento establece niveles de prioridad en las metas y resuelve secuencialmente el problema para cada nivel de prioridad, manteniendo los valores previamente obtenidos para metas con mayor nivel de prioridad. Así, si en un problema se formulan k metas es posible que se consideren agrupadas en $h\,(\leq k)$ niveles de prioridad. Por ejemplo, si el problema incluye 6 metas, que denominaremos $g_i(n, p), i = 1, \ldots, 6$, agrupadas en 3 niveles de prioridad de la forma las más importantes las dos primeras, después la tercera, y por último las 3 últimas, el problema se representa como

$$Lex \min a = [g_1(n, p) + g_2(n, p), g_3(n, p), g_4(n, p) + g_5(n, p) + g_6(n, p)]$$

Para resolver el problema, se resolvería primero con las dos primeras metas. Obtenidos los valores de las variables de desviación no deseadas para éstas, se fijan y se resuelve el problema para la tercera meta (segundo nivel de prioridad). Una vez resuelto, se fija el valor de la variable no deseada también para esta meta, y se resuelve el tercer nivel incluyendo las tres últimas metas.

2.2.4.2 Temas críticos en programación por metas

A pesar de que la programación por metas resulta un potente instrumento de gran aplicabilidad e indudable éxito en la realidad, diferentes autores han apuntado algunas aparentes debilidades de los modelos de metas. Estas consideraciones han llevado a denominar *temas críticos en programación por metas* a anomalías aparentemente causadas por debilidades lógicas de la programación por metas, aunque en realidad se deben al uso insatisfactorio del enfoque. La mayoría se presentan en las metas lexicográficas. Algunas de ellas se exponen a continuación.

1. *Posible equivalencia de soluciones entre modelos de programación por metas y modelos de optimización de un solo criterio.*

   Es cierto, que, por ejemplo, en las metas lexicográficas si se fijan unos niveles muy pesimistas en los primeros niveles (fáciles de alcanzar) y muy optimistas en los últimos (difíciles de alcanzar), al resolverlo para los primeros niveles se obtendrá un valor 0 de las desviaciones, y puede llegar un momento en que para una meta no haya óptimos alternativos y, por lo tanto, fijar el valor de su desviación es ya dar una solución haciendo redundantes las metas de prioridades más bajas; de este modo, sería equivalente a fijar las metas de los niveles anteriores en su nivel de aspiración y optimizar la meta para la que no hay óptimos alternativos. Sin embargo, esta equivalencia se da por un mal planteamiento de los niveles de aspiración. La equivalencia puede darse también en metas ponderadas, por ejemplo, si se plantean unos niveles de aspiración muy pesimistas para las metas, excepto para una en que resulte muy optimista, prácticamente inalcanzable. En tal caso, es cierto que el problema puede ser equivalente a optimizar ese atributo incluyendo las demás metas como restricciones. En cualquier caso, estos problemas se derivan de la formulación del modelo y no a debilidades lógicas de la metodología.

2. *Conclusiones equivocadas por agregar metas lexicográficas en una función objetivo*

   Cuando se plantea un modelo de programación por metas lexicográficas, suele haber una tendencia (casi natural) a intentar plantear un modelo equivalente que se resuelva en una sola iteración, sin tener que resolver un problema para cada nivel de prioridad. Así es común ver que se plantea una función objetivo como en el procedimiento de metas ponderadas donde se multiplica cada variable de desviación por un peso asignado a su nivel de prioridad (así para el primer nivel el peso es mucho mayor que para el segundo, etc.). En general, este planteamiento, además de ser lógicamente incorrecto, puede llevar a obtener soluciones que difieran mucho de las que se obtendrían con el modelo lexicográfico, y por lo tanto, si se intenta razonar sobre ellas como si fueran soluciones del modelo lexicográfico, a conclusiones claramente incorrectas.

3. *Problemas derivados de la omisión de una variable de desviación*

   Otros de los problemas que han surgido en la aplicación, o mejor dicho, en la incorrecta aplicación de la programación por metas, han sido los derivados por no incluir una variable de desviación. En ocasiones, se ha prescindido de la variable de desviación que no es considerada, por ejemplo, si se desea que un atributo sea superior a un nivel de aspiración, la variable no deseada es la de holgura al nivel de aspiración, n, sin embargo de la otra variable, la de exceso p, no se considera en el objetivo. Obsérvese que prescindir de ella implica que el atributo no pueda tomar valores superiores al nivel de aspiración, sólo inferiores (no deseados) o igual, lo que restringe el espacio de soluciones, eliminando del espacio de objetivos toda la región en que el atributo sea superior. Por lo tanto, la variable de desviación que no es la no deseada, también debe aparecer en el modelo o formular la meta como una desigualdad y no una igualdad. Ante el riesgo que puede presentar esta última formulación, se aconseja mantener ambas variables con la formulación lógica clásica de las metas, además de que, en general, no incluirlas no conlleva una mejora computacional (los métodos de resolución suelen añadir una variable en las desigualdades como primera medida).

4. *Problemas derivados de la inclusión de metas con dos lados innecesariamente*

   Las metas con dos lados no deseados son menos habituales que las de un solo lado, ya que es raro que un decisor desee el logro exacto de una meta más que superar, o no hacerlo, un determinado nivel. Incluir entre las no deseadas una desviación que no lo es, puede llevar a obtener soluciones subóptimas. Es claro, que hay que tener un especial cuidado en identificar cuáles son las variables de desviación no deseadas.

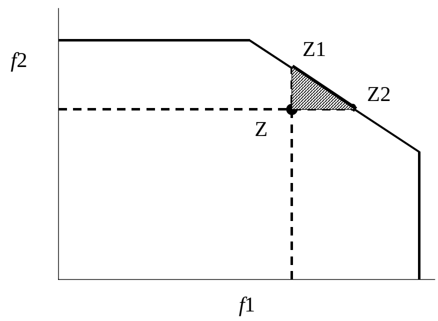
5. *Incompatibilidad entre ordenaciones lexicográficas y la existencia de una función de utilidad*

   Una de las críticas más fuertes al modelo lexicográfico ha sido su incompatibilidad con una función de utilidad del decisor. Esta incompatibilidad demostrada, que aquí no se niega, se deriva de la incompatibilidad con uno de los axiomas que se atribuye a una función de utilidad. Los axiomas que se supone que deben ser cumplidos por una función de utilidad son comparabilidad, reflexividad, transitividad y continuidad. De ellos, el que resulta incompatible con las ordenaciones lexicográficas es el de la continuidad. Según este axioma, los conjuntos de nivel de la función de utilidad son superficies continuas, es decir, dada una combinación de dos elecciones posibles de un decisor, siempre se podrá reducir la cantidad de una elección encontrando un incremento de la otra elección que compense exactamente la reducción. Es fácil comprobar que en las metas lexicográficas los conjuntos de nivel o superficies de indiferencia están formadas por un único punto (ver estos conjuntos como intersección del conjunto mejor y el peor respecto a un punto), por lo tanto, no es posible obtener el mismo valor con dos puntos distintos, lo que supone que no se cumple el axioma. Sin embargo, la continuidad de las preferencias dista mucho de ser un hecho, ni tan siquiera una hipótesis corroborada de forma empírica. Es tan sólo una hipótesis necesaria para axiomatizar la teoría neoclásica del consumo y asegurar la existencia de un equilibrio competitivo. Es más, hay situaciones en las que suponer válido este axioma puede ser un disparate; por ejemplo, supóngase un problema de planificación forestal donde se tienen en cuenta varios criterios, por ejemplo, uno económico que fuera la producción de madera y otro que fuera un índice que mida el riesgo de colapso ecológico del bosque. En este caso el supuesto de continuidad diría que siempre existe un incremento en el volumen de madera producida que compense un incremento del riesgo de colapso del bosque, por alto que éste sea, lo que puede ser un auténtico disparate. Es obvio, que en este caso y en otros muchos, las preferencias no son continuas y por lo tanto no existe una función de utilidad para representarlas. Es decir, no es absolutamente necesario aceptar la continuidad de las preferencias para poder representar el problema de decisión.

6. *Posible ineficiencia paretiana de la solución obtenida por metas, y forma de resolverlo*

   La programación por metas no está pensada para obtener soluciones eficientes sino para obtener soluciones satisfacientes. En este sentido, es posible que la solución de un modelo de programación por metas no sea eficiente. Para ello, lo que tiene que pasar es que haya óptimos alternativos del problema de metas ponderadas o del último nivel de metas lexicográficos (esta condición es necesaria aunque no es suficiente). Para algunos autores es un gran inconveniente, pero no ha de olvidarse que el objetivo de la programación por metas, como ya se ha dicho, no es obtener soluciones eficientes sino satisfacientes. Para comprobar si una solución obtenida mediante programación por metas es eficiente, se fijan los valores de las variables de desviación no deseadas obtenidos, y se maximizan las variables opuestas; si la solución no cambia, es que la solución es eficiente, mientras que si varía, la nueva solución será una solución eficiente con los mismos valores de las variables no deseadas. Existe un test propuesto por Hannan (1980) que permite a partir de una solución obtenida por metas, obtener todas las soluciones eficientes que tienen el mismo valor de las variables de desviación no deseadas. Este test consiste en aplicar la técnicas multiobjetivo a un problema que sea el original pero incluyendo los valores de las variables de desviación (tanto las deseadas como las no deseadas) obtenidas previamente mediante programación por metas. Así si una meta es que x_1 sea al menos 5, y al resolverlo por metas se ha obtenido $n_1 = 0, p_1 = 1$, se formularía una restricción para el test de Hannan que fuera:

   $x_1 \geq 5 - n_1^* + p_1^* = 5 - 0 + 1 = 6$. Gráficamente, si Z es el punto obtenido por programación por metas, toda la zona rayada son puntos mejores, y Z1 y Z2 serían puntos eficientes mejores que Z y todas las combinaciones lineales convexas de ellos también:



De aquí se deriva un procedimiento para resolver modelos de programación por metas si se desea que además la solución sea eficiente:

1. Resolver el modelo de metas inicial. Si no hay óptimos alternativos la solución es eficiente, en otro caso ir al paso 2.
2. Si se desea una única solución eficiente ir a 3, si se desea explorar un conjunto de soluciones eficientes ir a 4.
3. Maximizar la suma de las variables desviacionales opuestas sin empeorar los valores alcanzados. La solución será eficiente
4. Transformar el modelo de metas en uno multiobjetivo con el procedimiento de Hannan para obtener todos los puntos eficientes del correspondiente modelo de metas.

7. *El concepto de meta redundante y repercusión en las metas lexicográficas*

El concepto de metas redundante puede surgir tanto en metas ponderadas como en lexicográficas, aunque con distinto sentido. Así en metas ponderadas puede surgir porque se plantee una meta inalcanzable y las demás de modo pesimista, con lo que el problema puede desviarse hacia la meta inalcanzable olvidando las demás. Sin embargo, donde es más relevante y claro este concepto es en las metas lexicográficas, ya que si en un determinado nivel de prioridad se llega a un único óptimo, todas las metas incluidas en niveles inferiores son un mero ornamento, ya que no pueden variar la solución. Se define como meta redundante aquella cuya omisión no influye en el resultado. La existencia de metas redundantes puede darse principalmente por tres causas:

1. Una excesiva priorización de las metas, es decir, una agrupación en un número excesivamente elevado de niveles de prioridad
2. Ser excesivamente optimista, fijando los niveles de aspiración muy próximos al ideal del atributo.
3. Incluir muchas metas con dos lados, obligando a la minimización de ambas variables de desviación.

Cuando se da la redundancia es interesante saber en qué nivel de prioridad se ha dado, y si parece oportuno, replantear el problema. El replanteamiento puede hacerse redefiniendo los niveles de aspiración de las metas anteriores o viendo si se han incluido metas de dos lados innecesariamente, o agrupando en distintos niveles (por ejemplo, se puede plantear un último nivel con la última meta no redundante ponderada para darle más importancia y las restantes metas, las que resultaron redundantes, con ponderación menor).

2.2.5 Métodos de decisión multicriterio discretos

Estos métodos se aplican cuando el conjunto de alternativas que se considera es discreto y es factible enumerar y tratar explícitamente cada uno de las alternativas posibles.

Existen distintos métodos para abordar estos problemas. En esta introducción se van a mostrar dos de ellos, aunque no son los únicos: el método ELECTRE y los procesos analíticos jerarquizados (AHP).

2.2.5.1 Procesos analíticos jerarquizados (método AHP)

Este método fue introducido por Saaty en 1977, y tuvo un gran impacto teórico y aplicado. Este método considera el problema dividido en 3 *niveles o jerarquías:*

Nivel 1: *Propósito* del problema

Nivel 2*: Criterios*

Nivel 3: *Alternativas*

Por ejemplo, si el problema es elegir el trazado de una carretera con tres posibles alternativas A, B, C, evaluadas por su coste, impacto medioambiental y tiempo de ejecución, los tres niveles serían:

Nivel 1: trazado de una carretera

Nivel 2*:* Coste, Impacto medioambiental, tiempo ejecución

Nivel 3: A, B, C

Una vez conceptualizada la estructura jerárquica del problema, se establece una fuerte interacción con el decisor en cada uno de los niveles para que emita su juicio de valor o preferencias. Esto supone hacer comparaciones por parejas de criterios y alternativas:

- Nivel 2:

La interacción con el decisor consiste en comparar por parejas los criterios. El procedimiento es el que se describió de Saaty en el punto 1 del método Electre: construye una matriz de comparación de criterios asignando un 1 si ambos son de la misma importancia, un 3 si hay una moderada importancia de un criterio respecto a otro, un 7 si hay una demostrada importancia y un 9 si hay extrema importancia. Con ello se forma una matriz cuadrada a_{ij} que valora la importancia del criterio *i* respecto al *j*. Si la importancia de un criterio respecto a otro es a_{ij}, a la inversa será $1/a_{ij}$. Obtenida esta matriz, se busca un vector de pesos para los criterios que sea solución del sistema $\frac{W_i}{W_j} = a_{ij}$, o lo que es lo mismo, $W_i - a_{ij}W_j = 0$ con la condición añadida de que sumen 1 para evitar la solución trivial de todos los pesos iguales a 0. Lamentablemente, ese sistema no suele tener solución dadas las normales inconsistencias del decisor y hay que buscar los que más se aproximen, por ejemplo mediante la programación por metas.

En el ejemplo del trazado de la autopista, supóngase que el centro decisor ha emitido sus juicios de valor obteniendo la siguiente matriz de comparaciones

	Coste	Impacto Medioambiental	Tiempo ejecución
Coste	1	2	5
Impacto Medioambiental	1/2	1	3
Tiempo ejecución	1/5	1/3	1

A partir de esta matriz se buscan los pesos preferenciales de los criterios resolviendo el siguiente problema donde se han introducido las variables de desviación para resolverlo:

$$\min \sum_{i=1}^{3} (n_i + p_i)$$
$$w_1 - 2w_2 + n_1 - p_1 = 0$$
$$w_1 - 5w_3 + n_2 - p_2 = 0$$
$$w_2 - 3w_3 + n_3 - p_3 = 0$$
$$w_1 + w_2 + w_3 = 1$$

La solución obtenida es $(w_1, w_2, w_3) = (.588,.294,.118)$, siendo las variables de desviaciones $n_1 = p_1 = p_2 = n_2 = p_3 = 0, n_3 = 0.06$.

- Nivel 3:

La interacción con el decisor en este nivel supone que el decisor emita sus juicios de valor cuando se comparan las alternativas por parejas para un determinado criterio. De esta forma se obtiene una tabla de comparaciones de alternativas para cada criterio. Estas comparaciones se valoran igual que en el caso anterior. Supongamos que para el ejemplo, se han obtenido las siguientes tres tablas:

	A	B	C
A	1	6	3
B	1/6	1	½
C	1/3	2	1

Coste

.

	A	B	C
A	1	1/9	1/5
B	9	1	2
C	5	½	1

Imp. Medioamb.

.

	A	B	C
A	1	½	¼
B	2	1	½
C	4	2	1

Tiempo ejecuc.

De nuevo se plantean sendos problemas de programación por metas para obtener unos pesos consistentes con las preferencias del decisor. En la tabla siguiente, se recogen estos pesos de las alternativas con cada criterio, donde se han puesto también en los criterios los pesos obtenidos en el nivel anterior:

Criterios / Alternativas	Coste 0.588	Impacto medioambiental 0.294	Tiempo ejecución 0.118
A	0.667	0.069	0.143
B	0.111	0.621	0.286

C	0.222	0.310	0.571

Por último, se obtienen los pesos globales de las alternativas para ambos niveles jerárquicos mediante una agregación multiplicativa, es decir, para cada alternativa se suman los productos de los pesos de ésta por el peso del criterio correspondiente. Así, para la alternativa A se obtendría 0.667x0.588+0.069x0.294+0.143x0.118=0.429. De esta forma se obtienen los pesos globales 0.429 para A, 0.282 para B y 0.289 para C. De este modo se obtiene una ordenación de las alternativas, resultando la A la preferida.

2.2.5.2 Método Electre

El **Método Electre** fue desarrollado por Benayoun, Roy y Sussman, en 1966. El método, básicamente, pretende reducir el tamaño del conjunto de soluciones eficientes, realizando una partición del conjunto eficiente en alternativas más favorables (*núcleo*) y menos favorables. Esta partición se lleva a cabo mediante una *relación de sobreclasificación* entre las alternativas. Una relación de sobreclasificación se basa en la siguiente relación entre alternativas:

Una alternativa E_i *sobreclasifica* (*outranks*) a otra E_k si para los atributos considerados el enunciado "la alternativa E_i es al menos tan buena como la alternativa E_k" es válido.

Esa definición de "al menos tan buena" en el método Electre se define a partir de la concordancia y discordancia entre alternativas:

- La *concordancia* de una alternativa E_i y otra E_k cuantifica hasta qué punto en un número alto de atributos E_i es preferida a E_k.
- La *discordancia* cuantifica hasta qué punto no hay atributo en que E_k es mucho mejor que E_i

Para que una alternativa *sobreclasifique* a otra y forme parte del *núcleo*, ha de superar un umbral mínimo de concordancia ($\bar{c}$)y no superar otro umbral de discordancia ($\bar{d}$).

Obsérvese que la sobreclasificación es distinta de la dominancia, en el sentido paretiano. Por otra parte, se trata de una relación que no tiene la propiedad de la transitividad, es decir, una alternativa puede sobreclasificar a otra y ésta a una tercera, y sin embargo, la primera no sobreclasificar a la tercera. Esta falta de transitividad se presenta como una ventaja ya que las razones por las que un centro decisor puede preferir la primera alternativa a la segunda y las que llevan a preferir la segunda a la tercera pueden ser muy diferentes y no llevar a que la primera sea preferida a la tercera.

El método Electre puede recogerse en el siguiente algoritmo.

Algoritmo Electre

1.- Formar la *matriz decisional* (E_i,A_j), es decir, una matriz donde las filas son las posibles elecciones y las columnas los atributos, siendo el “valor” de un elemento de la matriz el valor de ese atributo para esa elección.

Dar un *vector de pesos preferencial de los atributos* W, es decir, dar un vector que ordene la importancia de los criterios. La forma más sencilla de obtenerlo es pedirle al decisor que clasifique los criterios por orden de importancia, de modo que si hay n criterios, al más importante le asigne el valor 1 y al menos importante el valor n; a continuación, con el fin de que la suma de los pesos sea 1 se le asigna al criterio en posición j-ésima el peso $W_j = \frac{1/r_j}{\sum_{i=1}^{n} r_i}$ o el peso $W_j = \frac{n - r_j + 1}{\sum_{i=1}^{n}(n - r_i + 1)}$. El problema de este método es que no tiene en cuenta la diferencia de importancia entre criterios, es decir, cuánto más importante es uno que el siguientes. Para evitarlo Saaty propone otro método que consiste en comparar los criterios por parejas, de modo que se asigne un 1 si ambos son de la misma importancia, un 3 si hay una moderada importancia de un criterio respecto a otro, un 7 si hay una demostrada importancia y un 9 si hay extrema importancia. Con ello se forma una matriz cuadrada a_{ij} que valora la importancia del criterio *i* respecto al *j*. Si la importancia de un criterio respecto a otro es a_{ij}, a la inversa será $1/a_{ij}$. Obtenida esta matriz, se busca un vector de pesos para los criterios que sea solución del sistema $\frac{W_i}{W_j} = a_{ij}$. Lamentablemente, ese sistema no suele tener solución dadas las normales inconsistencias del decisor y hay que buscar los que más se aproximen.

Aunque no puede hablarse de cuál es el mejor método para estimar pesos preferenciales, siempre que la situación lo permita parecen tener más solidez los métodos propuestos por Saaty que los anteriores.

2.- Cálculo de la *matriz de índices de concordancia,* $c(i,k)$*:*

$c(i,k)$ = suma de los pesos de los atributos en que E_i es mejor que E_k

(en caso de empate en un atributo se suma la mitad del peso)

3.- *Normalización de la matriz decisional*: consiste en normalizar los criterios para evitar el posible efecto de trabajar con distintas unidades. Hay varios métodos para la normalización: dividir los valores por el mejor que haya en ese criterio, o dividir por el rango o recorrido del criterio, o restar al mejor valor del criterio el de esa alternativa y dividir después por el rango (con este procedimiento los valores quedan entre 0 y 1, siendo 0 el mejor y 1 el peor).

4.- Cálculo de la *matriz de decisión normalizada y ponderada*: Multiplicar cada columna (atributo) de la matriz de decisión por el peso preferencial correspondiente a ese atributo.

5.- Cálculo de la *matriz de índices de discordancia,* $d(i,k)$*:*

$d(i,k)$ = diferencia mayor entre los criterios en que E_i es dominada por E_k dividida entre la diferencia mayor en valor absoluto entre los valores de un criterio cualquiera.

6.- Fijar un *umbral mínimo de concordancia,* $\overline{c}$, y un umbral máximo de discordancia, $\overline{d}$.

7.- Calcular la *matriz de dominancia concordante*: poner un 1 si el índice de concordancia es mayor que el umbral, un 0 si no.

8.- Calcular la *matriz de dominancia discordante*: poner un 1 si el índice de discordancia es menor que el umbral, un 0 si no.

9.- Calcular la *matriz de dominancia agregada*: multiplicar términos homólogos de las matrices de dominancia concordante y discordante. Así si un elemento de esa matriz es 1 es porque la alternativa de esa fila es mejor que la de la columna en un número de importante de criterios y no es claramente peor en ningún criterio. Si hay un 0 es que o bien no es mejor en un número importante de criterios o es claramente peor en algún criterio o ambas.

10.- Obtener el ***núcleo***: se eliminan las alternativas que están sobreclasificadas por alguna otra, es decir, las que tienen al menos un 1 en su columna en la matriz de dominancia agregada. Si se representa en un grafo en los nodos las alternativas y en los arcos las sobreclasificaciones (el origen es la alternativa que sobreclasifica a otra que es el destino), el núcleo será único si este grafo no tiene circuitos.

Obsérvese que según se fijen los umbrales de concordancia y discordancia el resultado será distinto.

Por ejemplo, sea el problema es elegir el trazado de una carretera con cinco posibles alternativas A, B, C, D y E, evaluadas por su coste, impacto medioambiental y tiempo de ejecución, arrojando los siguientes valores:

$$\begin{bmatrix} & \text{Coste(cien millones€)} & \text{Medioambiental (u. más alto peor)} & \text{Tiempo ejec(a.)} \\ A & 1 & 8 & 1 \\ B & 2 & 6 & 2 \\ C & 2.5 & 7 & 1.5 \\ D & 3 & 4 & 2 \\ E & 4 & 2 & 2.5 \end{bmatrix}$$

El proceso sería el siguiente:

1.- Formar la *matriz decisional* (matriz anterior), y dar un *vector de pesos preferencial de los atributos* W: mediante el método descrito por Saaty y utilizando programación por metas, una posible tabla de comparaciones sería

	Coste	Impacto Medioambiental	Tiempo ejecución
Coste	1	2	5
Impacto Medioambiental	1/2	1	3
Tiempo ejecución	1/5	1/3	1

A partir de esta matriz se buscan los pesos preferenciales de los criterios resolviendo el siguiente problema donde se han introducido las variables de desviación para resolverlo:

$$\min \sum_{i=1}^{3} (n_i + p_i)$$
$$w_1 - 2w_2 + n_1 - p_1 = 0$$
$$w_1 - 5w_3 + n_2 - p_2 = 0$$
$$w_2 - 3w_3 + n_3 - p_3 = 0$$
$$w_1 + w_2 + w_3 = 1$$

La solución obtenida es $(w_1, w_2, w_3) = (.588,.294,.118)$, siendo las variables de desviaciones $n_1 = p_1 = p_2 = n_2 = p_3 = 0, n_3 = 0.06$.

2.- Cálculo de la *matriz de índices de concordancia, c*(*i,k*)*:*

c(*i,k*) = suma de los pesos de los atributos en que E_i es mejor que E_k

(en caso de empate en un atributo se suma la mitad del peso)

$$\begin{bmatrix} & A & B & C & D & E \\ A & - & .71 & .71 & .71 & .71 \\ B & .29 & - & .88 & .65 & .71 \\ C & .29 & .12 & - & .71 & .71 \\ D & .29 & .35 & .29 & - & .71 \\ E & .29 & .29 & .29 & .29 & - \end{bmatrix}$$

3.- *Normalización de la matriz decisional*: se selecciona que el rango sea 0 a 1, y que 0 sea siempre el peor valor.

4.- Cálculo de la *matriz de decisión normalizada y ponderada*: Multiplicar cada columna (atributo) de la matriz de decisión por el peso preferencial correspondiente a ese atributo.

$$\begin{bmatrix} & \text{C.} & \text{E.} & \text{I.} \\ A & 0 & .29 & 0 \\ B & .19 & .19 & .08 \\ C & .29 & .24 & .04 \\ D & .39 & .10 & .08 \\ E & .59 & 0 & .12 \end{bmatrix}$$

5.- Cálculo de la *matriz de índices de discordancia, d*(*i,k*)*:*

d(*i,k*) = diferencia mayor entre los criterios en que E_i es dominada por E_k dividida entre la diferencia mayor en valor absoluto entre los valores de un criterio.

$$\begin{bmatrix} & A & B & C & D & E \\ A & - & 0.16 & .08 & .33 & .5 \\ B & .33 & - & .07 & .16 & .33 \\ C & .5 & .16 & - & .25 & .42 \\ D & .66 & .33 & .16 & - & .16 \\ E & 1 & .66 & .5 & .33 & - \end{bmatrix}$$

6.- Fijar un *umbral mínimo de concordancia,* $\overline{c}$ *=0.7*, y un umbral máximo de discordancia, $\overline{d}$ *= 0.2*.

7.- Calcular la *matriz de dominancia concordante*: poner un 1 si el índice de concordancia es mayor que el umbral, un 0 si no.

$$\begin{bmatrix} & A & B & C & D & E \\ A & - & 1 & 1 & 1 & 1 \\ B & 0 & - & 1 & 0 & 1 \\ C & 0 & 0 & - & 1 & 1 \\ D & 0 & 0 & 0 & - & 1 \\ E & 0 & 0 & 0 & 0 & - \end{bmatrix}$$

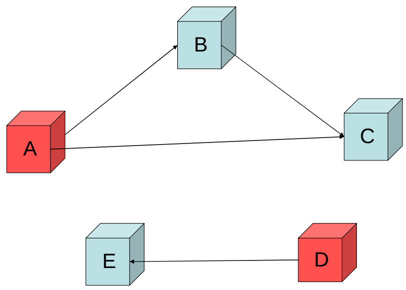
8.- Calcular la *matriz de dominancia discordante*: poner un 1 si el índice de discordancia es menor que el umbral, un 0 si no.

$$\begin{bmatrix} & A & B & C & D & E \\ A & - & 1 & 1 & 0 & 0 \\ B & 0 & - & 1 & 1 & 0 \\ C & 0 & 1 & - & 0 & 0 \\ D & 0 & 0 & 1 & - & 1 \\ E & 0 & 0 & 0 & 0 & - \end{bmatrix}$$

9.- Calcular la *matriz de dominancia agregada*: multiplicar términos homólogos de las matrices de dominancia concordante y discordante.

$$\begin{array}{cc} & \begin{array}{ccccc} A & B & C & D & E \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \\ E \end{array} & \begin{bmatrix} - & 1 & 1 & 0 & 0 \\ 0 & - & 1 & 0 & 0 \\ 0 & 0 & - & 0 & 0 \\ 0 & 0 & 0 & - & 1 \\ 0 & 0 & 0 & 0 & - \end{bmatrix} \end{array}$$

10.- Obtener el núcleo: es único y está formado por las alternativas A y D. Según el grafo:



2.3 Referencias

Benayoun, R., Roy, B., Sussman, B. Electre: Une Méthode pour Guider le Choix en Présence de Vue Multiples. *Sema* (*Metra International*), Direction Scientifique. Note de travail, 49.

Charnes, A., Cooper, W.W (1961) *Management Models and Industrial Applications of Linear Programming.* John Wiley and Sons.

Cournot, A. (1838) *Recherches su le Principes Mathématiques de la Théorie des Richesses.*

DeGroot, M.H. (1970) *Optimal Statistical Decisions*, McGraw Hill, New York

French, S. (1986) *Decision Theory: An introduction to the mathematics of rationality*, Ellis Horwood, Chichester.

Hannan, E.L. (1980) Nondominance in Goal Programming. *INFOR, Canadian Journal of Operational Research and Information Processing* **18**

Ignizio, J. P. (1976) *Goal Programming and Extensions.* Lexington Books.

Lee, S. M. (1972) *Goal Programming for Decision Analysis*. Auerbach Publishers.

Keeney, R.L. and Raiffa, H. (1976) *Decisions with Multiple Objectives: Preferences and Value Trade-Offs.* John Wiley and Sons.

Ríos, S., M.J. Ríos-Insúa, S. Ríos-Insúa. (1989) *Procesos de Decisión Multicriterio*. Eudema.

Romero, C. (1991) *Handbook of Critical Issues in Goal Programming.* Pergamon Press.

Romero, C. (1993) *Teoría de la Decisión Multicriterio: Conceptos, Técnicas y Aplicaciones.* Alianza Universidad.

Saaty, T. L. (1977) A Scaling Method for Priorities in Hierarchical Structures. *Journal of Mathematical Psychology*, **15,** 234-281.

Saaty, T.L. (1980) *The Analytic Hierarchy Process.* McGraw Hill

Simon, H. A. (1955) A Behavioral Model of Rational Choice. *Quarterly Journal of Economics*, **69,** 99-118.

Yu, P.L. (1973) A Class of Solutions for Group Decision Problems. *Management Science*, **19,** 936-946.

Yu, P.L. (1985) *Multiple criteria decision making: Concepts, techniques and extensions*. Plenum, New York.

Zadeh, L.A. (1963) Optimality and Non-Scalar-Valued Performance Criteria. *IEEE Transactions on Automatic Control*, **8**, 59-60.

Zeleny, M. (1973) Compromise Programming, in *Multiple Criteria Decision Making* (Cochrane, J.L. and Zeleny, M. ed.). University of South Carolina Press, 262-301

Zeleny, M. (1974) A Concept of Compromise Solutions and the Method of the Displaced Ideal. *Computers and Operations Research*, **1,** 479-496.

Zeleny, M. (1982) *Multiple criteria decision making*. McGraw-Hill, New York.

2.4 Biblioteca de problemas

PROBLEMA 1

El vendedor de periódicos Chema Pamundi vende en la esquina de la Avenida de Roquetas y la Avenida de Aletas de la Frontera, y cada día debe determinar cuántos periódicos pedir. Chema compra a 100 céntimos de euro cada periódico y lo vende a 125. Los periódicos que no vende al final del día no tienen valor. Chema sabe que cada día puede vender entre 6 y 10 periódicos, siendo igual cada probabilidad. Dar la decisión óptima según cada uno de los criterios.

PROBLEMA 2

Una compañía ha diseñado un nuevo circuito integrado que le permitirá entrar en el mercado de los microordenadores si así lo desea. En otro caso, puede vender sus derechos por 80 millones. Si elige construir ordenadores, la rentabilidad de este proyecto depende de la habilidad de la compañía para comercializarlas durante el primer año. Tiene suficiente acceso a los distribuidores para asegurar la venta de 1000 ordenadores. Por otro lado, si tiene éxito puede llegar a vender hasta 10000 máquinas. La compañía piensa que ambas alternativas de venta son igualmente probables y que cualquier otra puede ignorarse. El coste de instalar la línea de producción es de 60 millones. La diferencia entre el precio de venta y el coste de cada ordenador es de 60000 euros. Determinar según los diferentes criterios cuál es la decisión óptima.

Supóngase ahora que se puede realizar un estudio de mercado a un coste de 40 millones para predecir cuál de los dos niveles de demanda es más probable que se dé. La experiencia indica que esta investigación de mercado es correcta dos tercios de las veces. Determinar la política óptima a seguir, según el criterio del valor medio.

PROBLEMA 3

Colaco tiene en la actualidad activos de 15 millones de euros y desea decidir si vende o no un refresco con sabor a chocolate, la Chocola. Colaco tiene tres opciones: 1) Probar en forma local el mercado de Chocola y, a continuación, usar los resultados del estudio de mercado para decidir si vende la Chocola a nivel nacional o no. 2) Directamente vender la Chocola a nivel nacional. 3) Decidir directamente no vender la Chocola.

A falta de un estudio de mercado, Colaco cree que Chocola tiene un 55% de posibilidades de éxito nacional, y 45% de fracaso absoluto. Si es un éxito nacional, el beneficio será de 30 millones y si es un fracaso, se perderán 10.

Si Colaco decide hacer el estudio previo a un coste de 3 millones, hay un 60% de posibilidades de que sea un éxito local y un 40% de fracaso local. Si obtiene éxito local hay un 85% de posibilidades de que Chocola sea éxito nacional. Si se obtiene un fracaso local hay sólo un 10% de que Chocola sea éxito nacional. Si Chocola es neutral respecto a riesgos, ¿qué estrategia debe seguir? ¿Y si Colaco valora sus ganancias de forma no lineal asignando las siguientes utilidades: u(45 millones)=1, u(42 millones)=.99, u(22.6 millones)=.6649, u(15 millones)=.48, u(12 millones)=.40, u(5 millones)=.19 y u(2 millones)=0?

PROBLEMA 4

Una compañía es dueña de unos terrenos en los que puede haber petróleo. Un geólogo consultor ha informado a la gerencia que piensa que existe una posibilidad de 1 entre 4 de encontrar petróleo. Debido a esta posibilidad, otra compañía petrolera ha ofrecido comprar las tierras por 9 millones de euros. Sin embargo, la compañía está considerando conservarla para perforar ella misma. Si encuentra petróleo, espera ganar aproximadamente 70 millones, pero perderá 10 si el pozo está seco.

1. Dar la decisión óptima según los distintos criterios de decisión. Para el criterio de Hurwicz determinar el grado de optimismo que separa cada una de las decisiones. ¿Cuál darías como solución si sabes que la empresa está operando con poco capital y la pérdida de 10 millones resulta importante?
2. A la empresa, se le ofrece una opción anterior a tomar una decisión que consiste en llevar a cabo una exploración sísmica detallada en el área para obtener una mejor estimación de la probabilidad de encontrar petróleo, y cuyo coste es de 3 millones. Se sabe que éste sondeo cuando hay petróleo lo predice correctamente seis de cada 10 veces, mientras que si no lo hay, acierta 8 de cada 10.

   Dar la política óptima de estas decisiones, y dar el beneficio esperado de esa política.
3. Dado que hay problemas de capital, se observa que la pérdida de 10 millones, y otros 3 si se encargó el sondeo previo, puede resultar muy grave para la empresa. Por ello se plantea la posibilidad de sustituir los beneficios/costes reales por utilidades de éstos. El dueño de la compañía da la siguiente tabla de utilidades:

Beneficio	-13	-10	6	9	67	70
Utilidad	-15	-10.5	6	9	58	60

   Replantear el problema con estas decisiones.

4. Antes de llevar a cabo la política de decisiones, la empresa es vendida a un empresario con mayor aversión al riesgo que plantea una nueva tabla de utilidades:

Beneficio	-13	-10	6	9	67	70
Utilidad	-20	-13	6	9	44	45

¿Cuál será la decisión del nuevo dueño?

PROBLEMA 5

La empresa HeladoSA desea lanzar un nuevo producto al que permita introducirle en nuevos mercados. En la siguiente tabla se muestra los beneficios (o pérdidas en su caso) esperados de los tres productos candidatos dependiendo del impacto de la campaña publicitaria.

	Alto Impacto	Impacto Medio	Bajo Impacto
Chocolight	25	10	15
Chocochufa	15	15	15
Chocodisco	20	10	-10

Determine qué producto sacaría al mercado si aplica el criterio de Savage, el de Wald y el de Hurwicz (para los distintos valores de α).

PROBLEMA 6

Hay un concurso de televisión que funciona como sigue: primero se me pregunta algo acerca de Stupid Videos. Si contesto bien gano 10000 euros. Creo tener una probabilidad de 0.8 de contestar bien esa pregunta. Si la contesto mal, se acabó y no gano nada. Si contesto bien me puedo quedar las 10000 euros o proseguir el juego y contestar alguna pregunta de Stupid TV Shows. Si contesto bien gano otras 30000 euros, pero si contesto mal pierdo lo anterior. Creo poder responder bien con una probabilidad de 0.6. Si la contesto bien, puedo irme con mi ganancia o proseguir y responder a una pregunta de Estadística. Si la contesto bien gano otras 50000 euros, pero si contesto mal pierdo todo lo anterior. La probabilidad de acertar esta pregunta es de 0.4. ¿Cómo puedo maximizar mi ganancia esperada y cuál es esta ganancia? Usar resultados de decisión.

PROBLEMA 7

Una empresa de suministros, "Sumis, S.A." desea analizar varias opciones de expansión de la empresa, fundamentalmente asociadas a su proceso productivo. Por una parte, tiene la opción de mejorar el material que está utilizando haciendo una fuerte inversión en investigación sobre materiales, logrando un mayor beneficio unitario posterior de los productos. Por otra parte, tiene la opción de ampliar la capacidad de producción haciendo una ampliación de la planta de producción ya existente. Además tiene otra posibilidad para ampliar la capacidad que consiste en abrir una nueva planta de producción. El presupuesto disponible no le permite abordar varias estrategias de expansión a la vez, de modo que ha de elegir una de las alternativas.

Por otra parte, las consecuencias derivadas de cada una de las decisiones son distintas según sea la coyuntura que rodee al momento de la expansión. Así, pueden darse distintos escenarios de demanda de los suministros en función fundamentalmente de lo que una empresa competidora, "Multi" pueda hacer. La otra empresa puede construir una nueva fábrica en el país, construir una nueva fábrica fuera del país, o no aumentar su producción.

Los beneficios que considera la empresa para cada una de sus opciones según se den las opciones de la otra empresa se recogen en la siguiente tabla, donde las filas representan las opciones de la empresa "Sumis, S.A." y los números su beneficio.

	Nueva fábrica en país	Nueva fábrica fuera de país	No hace nada
Nuevo material	10	-1	10
Amplía planta	0	3	10
Nueva planta	5	-2	6
No hace nada	-15	2	5

¿Qué aconsejarías a la empresa “Sumis, S.A.” hacer si las estrategias de la otra empresa son conocidas pero se desconoce absolutamente la valoración que esta empresa puede hacer de ellas, ya que además opera en otros mercados? ¿Cambiarías la opinión si supieras que las probabilidades de que la otra empresa lleve a cabo cada una de sus estrategias son de 0.1, 0.8 y 0.1, respectivamente? Justificar.

PROBLEMA 8

El gestor de las líneas de ferrocarril de un país desea instalar un sistema de seguridad en una nueva línea férrea en construcción. Para ello ha recibido la oferta de 5 empresas. Cada una de las ofertas está compuesta por tres partes: la oferta técnica, la oferta económica y el plan de desarrollo. Cada oferta es evaluada por expertos, y tras un análisis en que se considera que la oferta de la Empresa 5 no es viable, y sí las del resto, conceden una “puntuación” en cada una de las tres partes.

Se asigna una puntuación a cada parte y empresa, desestimando la mejor y la peor de las puntuaciones obtenidas, y haciendo la media de las otras. A continuación se muestra la tabla de puntuaciones asignadas:

	Oferta técnica	Oferta económica	Plan de desarrollo
Empresa 1	10	4	8
Empresa 2	6	9	5
Empresa 3	8	6	7
Empresa 4	7	7	9

El gestor ha de decidir a la vista de las puntuaciones obtenidas, qué empresa desarrollará el sistema. Suponiendo que las preferencias del gestor mantienen la relación lineal (es decir, una diferencia en un criterio de 10 a 8, es igual que de 8 a 6, por ejemplo), dar la opción en tres supuestos:

a) Los pesos asignados a los tres criterios son 0.4, 0.35 y 0.25, respectivamente

b) La matriz de comparación entre criterios es:

	O. Técnica	O. Económica	P. Desarrollo
O. Técnica	1	3	5
O. Económica	1/3	1	3
P. Desarrollo	1/5	1/3	1

c) Plantearlo con valoraciones propias de los criterios

PROBLEMA 9

La planificación energética se ha venido realizando tradicionalmente, en función de los costes del productor. Sin embargo, la asignación de recursos en base a este criterio no siempre es eficiente, ya que existen otros costes asociados a las distintas opciones, que no aparecen reflejados en el coste de generación.

Este es el caso de los costes medioambientales. La generación eléctrica conlleva unos impactos sobre el medio ambiente que, sin embargo, no son tenidos en cuenta a la hora de asignar eficientemente los recursos, desde un punto de vista social. Esta ineficiencia hace que la alteración del medio se siga produciendo, a pesar de la cada vez mayor sensibilización por parte de la opinión pública, sin que las fuerzas del mercado sean capaces de impedirlo.

Para evitar estas debilidades del mercado, es necesario introducir otros criterios para la toma de decisiones económicas. Una de las formas de hacerlo es la aplicación de la decisión multicriterio. Esta metodología presenta la ventaja de que no es necesario determinar un valor monetario para los impactos medioambientales. Su desventaja, sin embargo, radica en que esta heterogeneidad hace necesario analizar cada caso por separado, con la complejidad analítica que ello supone, lo que sería innecesario si todos los costes estuvieran medidos en las mismas unidades.

En este caso se presenta una planificación eléctrica muy simplificada, ya que ha sido necesario hacer supuestos, no siempre evidentes, para reducir la complejidad operativa. Esta simplificación se debe a que el objetivo buscado es simplemente mostrar cómo los resultados de una planificación eléctrica tradicional pueden verse alterados si se introducen otros criterios. La introducción de los costes medioambientales se ha hecho a través de una variable asociada, como las emisiones de CO_2, ya que estas emisiones son un aspecto medioambiental de gran relevancia en la actualidad.

Se supone una demanda eléctrica anual de 150000 GWh, que puede ser satisfecha mediante el empleo de las siguientes fuentes de energía: carbón, energía nuclear, gas natural, energía eólica, biomasa procedente de cultivos energéticos, y energía hidráulica. El coste de generación, así como los costes medioambientales producidos deben ser mínimos.

Las posibilidades de utilización de cada una de las opciones están determinadas por el potencial existente, o por condicionantes de otro tipo, que se detallan a continuación:

- la energía nuclear está sujeta a una moratoria que impide la expansión de la potencia instalada, con una producción anual de 53000 GWh.
- respecto al gas natural, se supone una potencia máxima instalada de 1.835 MW, establecida por el Plan Energético Nacional. Suponiendo una utilización de 7.000 h al año, la producción potencial sería de 12.845 GWh.
- el potencial de la energía eólica en nuestro país ha sido evaluado en unos 2.000 MW, que, para una utilización media de 2.500 h, produciría un máximo de 5.000 GWh.
- los recursos de biomasa para producción energética han sido estimados en 22 Mtep, que equivaldrían a unos 40000 GWh de producción eléctrica.
- por último, se supone un potencial hidráulico de 31.755 GWh.

Estos supuestos simplifican bastante la operación del problema, aunque se ajustan bastante a la realidad. En primer lugar, se considera que toda la demanda debe ser satisfecha con estas opciones, sin tener en cuenta otras como el fuel-oil, la energía solar, la cogeneración, etc. En segundo lugar, la planificación se va a realizar en base a la producción, no a la potencia instalada. Para ello se consideran unas horas de utilización determinadas, lo que en lógica, debería venir definido por la posterior optimización de los costes de generación, ya que es a partir de ésta cuando debe calcularse el tiempo de funcionamiento, y no antes. En cualquier caso, es de esperar que los supuestos no alteren demasiado el carácter ilustrativo de este ejercicio

Los costes de generación que se van a utilizar son costes medios aproximados de la generación eléctrica en España, para una utilización normal durante el año, según la planificación tradicional. En la tabla 1, aparecen los costes de generación, y emisiones de CO_2 para las opciones consideradas. Para poder referirlas a la producción de electricidad, ambos vienen establecidos con relación al kWh generado. Hay que recordar que algunos costes son simplemente estimaciones. El coste del carbón corresponde al carbón importado.

Tabla 1. Costes y emisiones de CO_2 de la generación eléctrica

Fuentes de energía	**Coste (c€/kWh)**	**Emisiones de CO_2 (g/kWh)**
Carbón	5,85	1015
Nuclear	8,24	-
Gas natural	5,00	401
Eólica	9,00	-
Biomasa	12,00	-
Hidráulica	6,00	-

Las tecnologías consideradas para estimar las emisiones son, para el carbón, combustión con combustible pulverizado, para el gas natural, un ciclo combinado, y para la biomasa, combustión en lecho fluidizado. En este último caso, aunque hay emisiones de CO2 en la combustión, las emisiones netas se consideran nulas debido a la fijación previa del CO2 por los cultivos energéticos empleados para la producción de electricidad.

2.5 Resultados de la biblioteca de problemas

RESULTADO DEL PROBLEMA 1

Por el criterio del valor esperado, son indiferentes 6 y 7 periódicos, siendo el beneficio esperado de 150 centieuros.

Por el criterio de Wald, se piden 6 periódicos.

Por el criterio de Savage, son indiferentes 6 y 7 periódicos siendo 100 la pérdida.

Por el criterio de Hurwicz, para $\alpha < 0.8$ se piden 6 periódicos y para $\alpha > 0.8$ se piden 10 periódicos.

RESULTADO DEL PROBLEMA 2

Por el criterio del valor esperado, se elige construir con un beneficio esperado de 270 millones de euros.

Por el criterio de Wald, se venden los derechos.

Por el criterio de Savage, elige construir con una posible pérdida de 80 millones.

Por el criterio de Hurwicz, para $\alpha < 0.148$ se venden los derechos y para $\alpha > 0.148$ se construyen ordenadores.

Si tiene la posibilidad de encargar el estudio, al fin decide no encargarlo e invertir directamente, con el beneficio esperado de 270 millones.

RESULTADO DEL PROBLEMA 3

Colaco debe lanzar la nueva bebida a nivel nacional sin estudio local, siendo la cantidad esperada al final de sus activos de 27 millones de euros.

Sin embargo, si emplea las utilidades, debe encargar el estudio local, y si resulta un éxito local lanzarlo a nivel nacional, y si no, no lanzarlo. La utilidad esperada en este caso es .6649.

RESULTADO DEL PROBLEMA 4

1. Por el criterio del valor esperado, debe perforar esperando ganar 10 millones.

   Por el criterio de Wald, decide vender los terrenos.

   Por el criterio de Savage, decide perforar siendo el coste de oportunidad de 19 millones.

   Por el criterio de Hurwicz, para $\alpha < 0.2375$ vende los terrenos y para $\alpha > 0.2375$ decide perforar.

2. Decide hacer el sondeo previo y si resulta favorable perforar, y si no, vender, esperando ganar 12.3 millones.

3. Decide seguir la misma estrategia, aunque ahora su utilidad es de 10.65.

   El nuevo dueño, decide vender recibiendo los 9 millones.

RESULTADO DEL PROBLEMA 5

Según el criterio de Wald, sacará Chocochufa; según el de Savage, sacará Chocolight; y según el criterio de Hurwicz, para $\alpha < 1/3$ será Chocochufa, y para $\alpha > 1/3$ será Chocolight.

RESULTADO DEL PROBLEMA 6

La ganancia esperada máxima es 19200 euros, decidiendo responder la primera pregunta, si la acierto responder la segunda, y después de ésta, en ningún caso continuar.

RESULTADO DEL PROBLEMA 8

a) Puntuaciones de las ofertas de las empresas: 7.4, 6.8, 7.05 y 7.5. Se elegiría la oferta de la Empresa 4

b) Pesos obtenidos mediante metas para los criterios: 0.65, 0.22 y 0.13, con una desviación del segundo criterio respecto al tercero de 0.174. Con estos pesos las puntuaciones de las ofertas de las empresas son: 8.42, 6.53, 7.43 y 7.26. Se elegiría la oferta de la Empresa 1.

3 Técnicas de planificación y control de proyectos

En un negocio o industria es habitual que se establezcan tareas que hay que desarrollar existiendo un determinado orden total o parcial para llevarlas a cabo. Estas tareas pueden ser tareas pertenecientes a un proyecto y lo que se busca es determinar el menor tiempo posible para acabar ese proyecto, o pueden ser, por ejemplo, procesos productivos que requieren el uso de determinados recursos y hay que decidir cómo secuenciar los procesos en los recursos con distintos objetivos o metas a alcanzar. Lo que resulta común a ambos planteamientos es la existencia de unas tareas que desarrollar y unas precedencias entre ellas, pudiendo considerarse todo ello planificación de tareas, con o sin recursos. Sin embargo, la planificación de proyectos tiene una entidad propia y un uso generalizado por lo que ha sido objeto de análisis pormenorizados y ha dado lugar a técnicas específicas, por lo que debe tener un tratamiento diferenciado. Existen aplicaciones ampliamente utilizadas en la industria que ayudan en la planificación y control de proyectos como son Microsoft Project (www.microsoft.com) y Gantt Project (ganttproject.sourceforge.net), esta última de domino público.

Por lo tanto, en esta sección se verán en primer lugar las técnicas más usuales de planificación de proyectos por su relevancia y en la sección siguiente los modelos de secuenciación sobre máquinas. Sin embargo, es conveniente observar que las técnicas que se presentarán para modelar los problemas de secuenciación son aplicables y extensibles a cualquier situación donde se busque una planificación de tareas con precedencias entre ellas.

3.1 Introducción

Las técnicas que vamos a ver en esta sección se utilizan cuando se desea planificar un proyecto con múltiples actividades, permitiendo no sólo llevar a cabo la programación de las actividades que configuran el proyecto, sino además detectar cuellos de botella, estimar la probabilidad de cumplir los plazos de entrega establecidos, evaluar el impacto en la planificación de realizar cambios de programa, etc.

Se entiende por *proyecto* un conjunto de actividades interrelacionadas, cada una con una duración y unos recursos necesarios para llevarla a cabo. Así el primer paso para planificar un proyecto es definirlo. Esto supone definir todas las actividades que configuran el proyecto, sus relaciones de precedencia y sus requerimientos de tiempo y recursos para llevarlas a cabo.

Estas condiciones configurarían las restricciones que la planificación debe cumplir, pero es necesario determinar cuál es el objetivo de la planificación. Se pueden plantear distintos objetivos, siendo fundamentalmente dos: objetivos de tiempo y objetivos de coste. Las técnicas básicas se plantean el objetivo de encontrar el tiempo mínimo en que puede ser completado el proyecto, sin embargo veremos algunas variantes donde se incluyen objetivos de coste o de asignación de los recursos.

Para resolver el problema del tiempo mínimo en los años 50 se desarrollaron dos técnicas que difieren en la consideración de la duración de las actividades, siendo una para duraciones deterministas y otra para duraciones estocásticas. Estas técnicas son:

- CPM (*Critical Path Method*): Método del camino crítico
- PERT (*Program Evaluation and Review Technique*)

Para aplicar ambas técnicas se requiere primero representar el proyecto mediante una red de actividades.

3.2 Red de actividades

Un proyecto se representa mediante una red que visualiza gráficamente las relaciones de precedencia en la realización de las actividades. Existen dos formas de llevar a cabo esta representación, una en que las actividades se localizan en los nodos y otra en la que las actividades se localizan en los arcos.

La técnica de búsqueda del tiempo mínimo de finalización del proyecto representando el proyecto con las actividades en los nodos es conocida como método ROY y su red como grafo ROY. La representación del proyecto mediante un grafo ROY suele resultar más intuitiva, pero cuando hay muchas actividades y relaciones de precedencia entre ellas puede resultar bastante más compleja y, por tanto, mucho menos útil. De ahí que en este contexto se presenten las técnicas sobre una red de actividades donde éstas se localizan en los arcos.

En la red cada actividad se representa por un arco con orientación la de progreso del proyecto. La duración de la actividad o su duración media, según el caso, determina la longitud del arco.

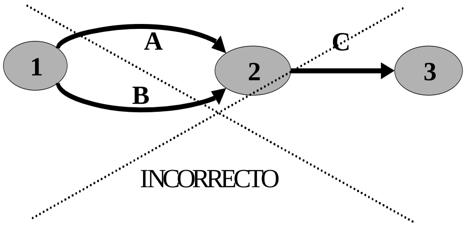
Los nodos establecen las relaciones de precedencia, de modo que los arcos que llegan a un nodo son las actividades que tienen que haber terminado para que puedan comenzar aquéllas que vienen representadas por arcos que salen de ese nodo. Así pues un nodo es identificado con un evento, entendiéndose por éste el fin de las tareas que llegan a ese nodo o el inicio de las tareas que salen de ese nodo.

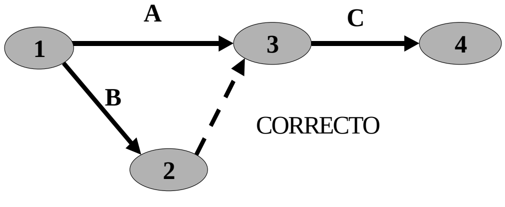
Como tales eventos que son considerados los nodos, en toda red deben existir un nodo inicial y un nodo final que representan el inicio del proyecto y su finalización.

Después de estas consideraciones se construye la red de actividades manteniendo dos propiedades:

1. Cada actividad es representada por uno y sólo un arco de la red
2. Dos nodos pueden estar conectados por a lo sumo un arco (ya que una actividad vendrá representada por sus nodos inicial y final)

Es posible que al construir la red no se puedan mantener estas propiedades y representar correctamente la relación entre las actividades sin añadir en algunos casos actividades ficticias. Una actividad ficticia es una actividad que se añade sin duración alguna para establecer relaciones de precedencia que de otra forma no se podría o para evitar violar las propiedades anteriores. Por ejemplo, si dos actividades tienen exactamente los mismos predecesores y son antecesoras de las mismas actividades habría que poner dos arcos entre dos nodos, lo que violaría la propiedad 2, teniendo que usar una actividad ficticia como se muestra en la figura.

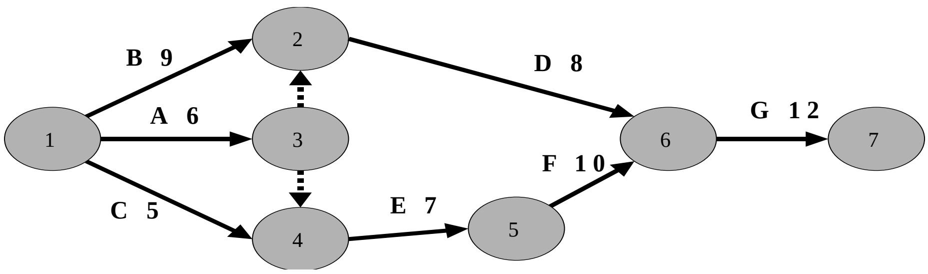




Otro caso donde también sería necesario introducir actividades ficticias es si dos actividades tienen distintos predecesores pero al menos uno común. Veamos el siguiente ejemplo de proyecto y su red de actividades correspondiente.

Un determinado artículo está compuesto por dos productos ensamblados. Se desea hacer la planificación para saber cuándo será posible disponer del artículo ya terminado habiéndose determinado que las actividades a realizar, sus precedencias y sus duraciones son las que se recogen en la siguiente tabla:

Actividad	Descripción	Predecesores	Duración
A	Capacitar trabajadores	–	6
B	Comprar materia prima producto 1	–	9
C	Comprar materia prima producto 2	–	5
D	Fabricar producto 1	A, B	8
E	Fabricar producto 2	A, C	7
F	Probar producto 2	E	10
G	Ensamblar productos 1 y 2	D, F	12



3.3 Método del camino crítico (CPM)

Este método se aplica cuando las duraciones de las actividades se suponen conocidas o estimadas con exactitud. El método obtiene la duración total necesaria para completar el proyecto, la programación de las actividades y la clasificación de éstas en actividades críticas y no críticas.

Una actividad se denomina *crítica* cuando no hay holgura al determinar sus instantes de inicio y final, es decir, cuando cualquier retraso en la finalización de esta actividad (tanto porque empezara más tarde que el instante previsto como porque se alargara su duración) supone un retraso en el final del proyecto.

Una actividad no crítica, sin embargo, tendrá holgura, pudiendo ser adelantada o retrasada dentro de unos límites sin afectar a la duración total del proyecto.

Se denomina *camino crítico* a un camino en la red desde el inicio hasta el fin formado por actividades críticas. Obsérvese que en una red puede haber más de un camino crítico.

El método del camino crítico se lleva a cabo en dos pasadas, una hacia delante en la red y otra hacia atrás. En la primera se determinan los instantes más tempranos en que pueden comenzarse las actividades y termina determinando la duración total del proyecto. En la segunda se calculan, yendo hacia atrás, los instantes más tardíos en que pueden acabar las actividades. Una vez obtenidos estos valores es fácil determinar cuáles son las actividades críticas.

Se usará la siguiente notación:

- d_{ij}: duración de la actividad que va del nodo i al nodo j
- t_i: instante más temprano del nodo i (es el instante más temprano en que pueden empezar las actividades que salen de ese nodo)
- T_i: instante más tardío del nodo i (es el instante más tardío en que pueden acabar las actividades que llegan a ese nodo)

Fase hacia delante* (*determinación de instantes más tempranos*)*:

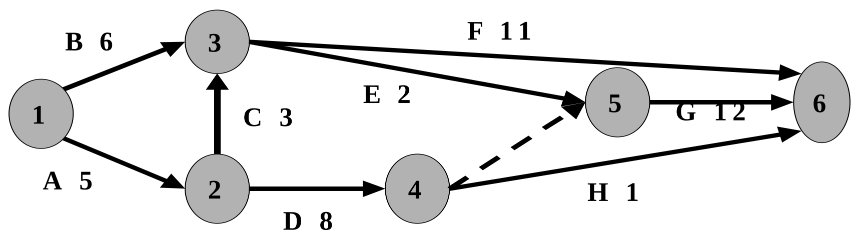
1. Etiquetar el nodo inicial con tiempo 0, es decir, $t_1 = 0$.
2. Elegir un nodo j tal que todos los nodos anteriores unidos directamente a él por un arco ya hayan sido etiquetados ($p,q,...,v$). Etiquetar el nodo j con el máximo de las etiquetas de estos nodos más la longitud del arco que los une, es decir, $t_j = \max\left\{t_p + d_{pj}, t_q + d_{qj}, ..., t_v + d_{vj}\right\}$.
3. Repetir el paso 2 hasta etiquetar el nodo final n, entonces t_n es la duración mínima del proyecto.

***Fase hacia atrás* (*determinación de instantes más tardíos*):**

1. Etiquetar el nodo final con tiempo el del proyecto, es decir, $T_n = t_n$.
2. Elegir un nodo j tal que todos los nodos posteriores unidos directamente a él por un arco ya hayan sido etiquetados ($p, q, ..., v$). Etiquetar el nodo j con el mínimo de las etiquetas de estos nodos menos la longitud del arco que los une, es decir, $T_j = \min\{T_p - d_{jp}, T_q - d_{jq}, ..., T_v - d_{jv}\}$.
3. Repetir el paso 2 hasta etiquetar el nodo inicial (que debe ser etiquetado con $T_0 = 0$, pues si no fuera así no sería correcto y se habría cometido algún error).

Con esto acaba el procedimiento de cálculo de los instantes más tempranos y más tardíos. Ahora es fácil identificar las actividades críticas que son aquellas actividades que no tienen holgura, es decir, una actividad (i, j) (denotada por sus nodos inicial y final) será crítica si se verifica $T_j = t_j$, $T_i = t_i$ y $T_j - T_i = t_j - t_i = d_{ij}$. Las actividades críticas han de formar un camino (al menos) desde el inicio al final. Puede haber más de un camino crítico.

Para ilustrar el procedimiento y los conceptos que se han introducido y los que se comentarán después veamos un ejemplo. Sea la siguiente red de actividades:



cuyos datos son los siguientes

Actividad	Predecesores	Duración
A	–	5
B	–	6

C	A	3
D	A	8
E	B, C	2
F	B, C	11
G	D, E	12
H	D	1

El procedimiento sería el siguiente:

Fase hacia delante:

1. $t_1 = 0$

2. $t_2 = t_1 + d_{12} = 0 + 5 = 5$

2. $t_3 = \max\{t_1 + d_{13}, t_2 + d_{23}\} = \max\{0 + 6, 5 + 3\} = 8$

2. $t_4 = t_2 + d_{24} = 5 + 8 = 13$

2. $t_5 = \max\{t_3 + d_{35}, t_4 + d_{45}\} = \max\{8 + 2, 13 + 0\} = 13$

2. $t_6 = \max\{t_3 + d_{36}, t_4 + d_{46}, t_5 + d_{56}\} = \max\{8 + 11, 13 + 1, 13 + 12\} = 25$

Luego, la duración del proyecto será de 25 unidades de tiempo.

Fase hacia atrás:

1. $T_6 = 25$

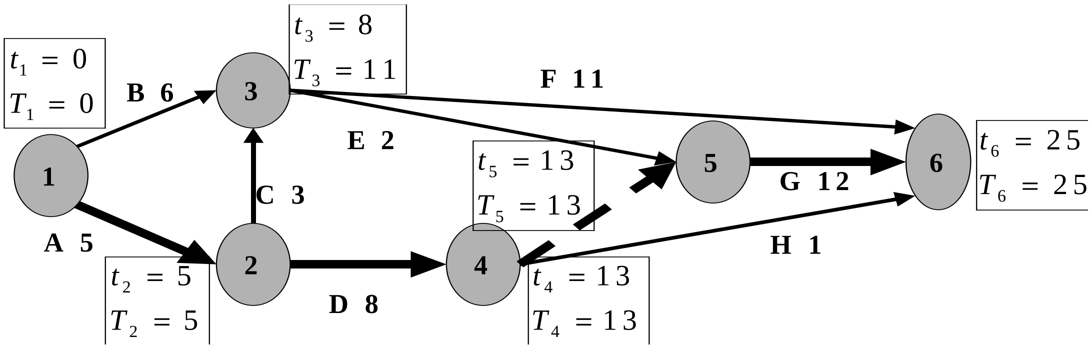
2. $T_5 = T_6 - d_{56} = 25 - 12 = 13$

2. $T_4 = \min\{T_6 - d_{46}, T_5 - d_{45}\} = \min\{25 - 1, 13 - 0\} = 13$

2. $T_3 = \min\{T_6 - d_{36}, T_5 - d_{35}\} = \min\{25 - 11, 13 - 2\} = 11$

2. $T_2 = \min\{T_3 - d_{23}, T_4 - d_{24}\} = \min\{11 - 3, 13 - 8\} = 5$

2. $T_1 = \min\{T_3 - d_{13}, T_2 - d_{12}\} = \min\{11-6, 5-5\} = 0$



Evidentemente, las actividades críticas son A, D y G, siendo el camino crítico el que pasa por los nodos 1, 2, 4, 5 y 6. Obsérvese que para la actividad H se verifica que los instantes más tempranos y más tardíos de sus nodos inicial y final coinciden, pero que no es una actividad crítica ya que la diferencia entre ambos es superior a la duración de la actividad. Es decir, no es suficiente para que una actividad sea crítica que sus nodos inicial y final no tengan holgura, la que no debe tenerla es la actividad.

Respecto a lo que se entiende por holgura de una actividad, existen dos tipos de holgura:

- *Holgura total* de la actividad (i, j), denotada por TF_{ij}: cantidad en que se puede retrasar el inicio de la actividad (i, j) más allá de su instante más temprano posible sin retrasar el proyecto (suponiendo que no se retrasan otras actividades). Su valor viene dado por la expresión $TF_{ij} = T_j - t_i - d_{ij}$. También se interpreta como lo que puede ser incrementada la duración de la actividad en esas condiciones.
- *Holgura libre* de la actividad (i, j), denotada por FF_{ij}: cantidad en que se puede retrasar el inicio de la actividad (i, j) (o su duración) más allá de su instante más temprano posible sin afectar al inicio de cualquier actividad posterior. Su valor viene dado por $FF_{ij} = t_j - t_i - d_{ij}$. Por definición, se verifica $FF_{ij} \le TF_{ij}$.

Por lo tanto si para una actividad se cumple que $FF_{ij} = TF_{ij}$, entonces la actividad puede ser programada en cualquier instante del intervalo de tiempo $[t_i, T_i]$ sin afectar a ninguna otra actividad. Si por el contrario se cumple que $FF_{ij} < TF_{ij}$, entonces cualquier retraso superior a FF_{ij} afectará a todos los eventos y actividades posteriores, teniendo que retrasar en la cantidad que exceda a la holgura libre el comienzo de todas las actividades que salen del nodo j.

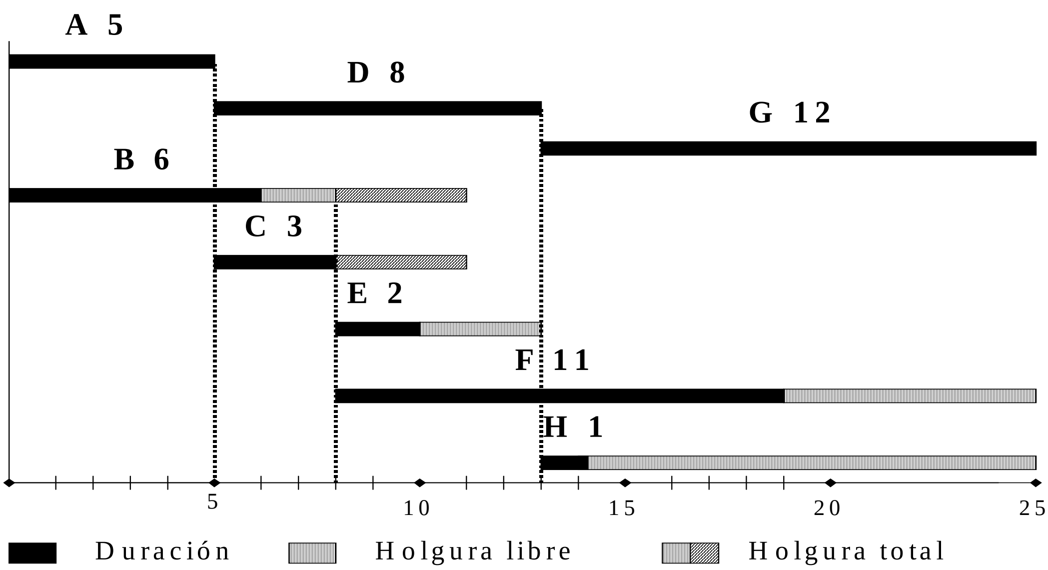
Obsérvese que si se han añadido actividades ficticias éstas pueden modificar la holgura libre de algunas actividades ya que a las que afectaría modificar el instante final de una variable pueden ser ficticias, lo cual no es real. Así pues, las actividades ficticias sólo se deben utilizar cuando sea absolutamente necesario y si al calcular las holguras alguna de estas actividades ficticias resultan con holgura dársela a la correspondiente actividad real a la que pertenece dicha holgura (generalmente, una actividad que acabe en el nodo de inicio de la ficticia).

Veamos el valor de las holguras en el ejemplo anterior. En primer lugar, para las actividades críticas no existe ningún tipo de holgura, luego sólo hay que calcularlas para las actividades no críticas. En la siguiente tabla se recogen esas holguras:

Act. no críticas	Duración d_{ij}	Holgura total TF_{ij}	Holgura libre FF_{ij}
B (1,3)	6	11 – 0 – 6 = 5	8 – 0 – 6 = 2
C (2,3)	3	11 – 5 – 3 = 3	8 – 5 – 3 = 0
E (3,5)	2	13 – 8 – 2 = 3	13 – 8 – 2 = 3
F (3,6)	11	25 – 8 – 11 = 6	25 – 8 – 11 = 6
H (4,6)	1	25 – 13 – 1 = 11	25 – 13 – 1 = 11

Como se puede observar, ambas holguras coinciden para las actividades E, F y H y no coinciden para B y C. Esto supone que así como las tres últimas actividades se pueden programar en cualquier punto entre su inicio más temprano y su inicio más tardío sin afectar a ninguna otra programación de actividad, no ocurre lo mismo para las actividades B y C. Es más, puesto que la actividad C tiene holgura libre 0 cualquier retraso en ella supone una modificación de las actividades posteriores a ella. Si este retraso es menor o igual que 3 se podrán reprogramar las actividades sin alargar la duración del proyecto y si es superior necesariamente se acabará más tarde el proyecto.

Para visualizar claramente la planificación obtenida y las holguras de las actividades se utiliza un formato gráfico, denominado *diagrama de Gantt.* En este diagrama, el eje X representa el tiempo y cada actividad se representa mediante una barra horizontal que comienza en su instante más temprano y de longitud la duración. Las actividades críticas se colocan arriba y las no críticas debajo. Además, en ocasiones, se alarga la barra añadiendo al final de la duración la holgura libre y detrás la diferencia entre la holgura total y la libre. Estos valores añadidos se representan con líneas de otro color. Veamos el diagrama de Gantt para el ejemplo anterior.



3.4 Método PERT

Este método se aplica cuando se considera aleatoria la duración de las actividades. El objetivo del método es dar una distribución de la duración total del proyecto, con la que se obtenga la duración esperada, su varianza y se pueda dar respuestas a preguntas del tipo probabilidad de terminar el proyecto en una fecha programada, etc.

En este contexto la duración de cada actividad, D_{ij}, es una variable aleatoria con esperanza $E\left[D_{ij}\right]$ y varianza $V\left[D_{ij}\right]$ y la duración total del proyecto será otra variable aleatoria, CP, cuya esperanza denotaremos por $E[CP]$ y su varianza por $V[CP]$.

El método asume ciertas hipótesis que pueden ser muy discutidas. Éstas son:

Día 1: *Las duraciones de las actividades son variables aleatorias independientes.*

Para cualquier camino desde el inicio hasta el final la duración esperada de ese camino es la suma de las duraciones esperadas de las actividades que lo forman, pero además por ser independientes las variables aleatorias, la varianza del camino es la suma de las varianzas. Es decir, en cualquier caso la duración esperada de un camino c es $\sum_{(i,j)\in c} E\left[D_{ij}\right]$, pero, además por ser independientes la varianza de la duración del camino es $\sum_{(i,j)\in c} V\left[D_{ij}\right]$.

Por otra parte, como la duración total de un proyecto viene dada por la duración de un camino crítico cc, es decir, $CP = \sum_{(i,j)\in cc} D_{ij}$, se tiene de forma inmediata que $E[CP] = \sum_{(i,j)\in cc} E\left[D_{ij}\right]$ y $V[CP] = \sum_{(i,j)\in cc} V\left[D_{ij}\right]$.

Día 2: *El camino crítico no cambia, siempre es el de mayor duración esperada.*

En el caso en que haya más de un camino con longitud esperada igual, se considera crítico el de mayor varianza.

La consecuencia inmediata de esta hipótesis junto con las consideraciones previas, es que para determinar la duración esperada de un proyecto y su varianza, es suficiente determinar el camino crítico aplicando el CPM, visto en la sección anterior, utilizando las duraciones esperadas de las actividades. Una vez determinado el camino crítico la esperanza de la duración es la suma de las esperanzas de las duraciones de las actividades que lo forman y la varianza también la suma de las varianzas de las actividades del camino.

Día 3: La variable duración del proyecto, CP, tiene distribución normal.

Esta hipótesis, en principio arbitraria, viene justificada porque al ser una variable que es suma de otras, aplicando el teorema central del límite, su distribución se aproxima suficientemente por una distribución normal. Por lo tanto, de las tres hipótesis se concluye que la distribución de la variable aleatoria CP es $N(\mu = \sum_{(i,j)\in cc} E\left[D_{ij}\right], \sigma^2 = \sum_{(i,j)\in cc} V\left[D_{ij}\right])$.

Por lo tanto, el procedimiento para obtener la distribución de la duración del proyecto sería el siguiente:

PASO 1: Determinar las esperanza y varianza de todas las actividades del proyecto.

PASO 2: Encontrar el camino crítico utilizando las duraciones esperadas de las actividades (en caso de haber más de uno, se elige el de mayor varianza).

PASO 3: Obtener la varianza del camino crítico sumando las varianzas de las actividades que formen el camino.

Así, por ejemplo, si se obtiene una ruta crítica formada por cinco actividades (una de ellas ficticia) cuyas esperanzas son 9, 0, 7, 10 y 12 y sus varianzas son 1.78, 0, 4, 0.44 y 1, la distribución de la duración del proyecto será $N(38,7.22)$. Con esta conclusión se puede dar respuesta a preguntas como ¿cuál es la probabilidad de acabar el proyecto antes de 35 días? Para responder, se calcula esta probabilidad por los métodos clásicos, siendo Z una variable con distribución normal de media 0 y varianza 1:

$$P(CP \leq 35) = P(\frac{CP-38}{\sqrt{7.22}} \leq \frac{35-38}{\sqrt{7.22}}) = P(Z \leq -1.12) = 0.13$$

Otra posible pregunta sería, ¿qué duración debo decir para tener una probabilidad del 97.5 % de acabar a tiempo? Si x es la incógnita, el planteamiento y la solución sería

$$\begin{aligned}
&P(CP \leq x) = 0.975 \Rightarrow \\
&\Rightarrow P(\frac{CP-38}{\sqrt{7.22}} \leq \frac{x-38}{\sqrt{7.22}}) = 0.975 \Rightarrow P(Z \leq \frac{x-38}{\sqrt{7.22}}) = 0.975 \Rightarrow \\
&\Rightarrow \frac{x-38}{\sqrt{7.22}} = 1.96 \Rightarrow x = 38 + 1.96\sqrt{7.22} = 43.27
\end{aligned}$$

de donde se deduce que habría que decir al menos 44 días.

Obsérvese que para aplicar este método no es necesario ni siquiera suponer distribución alguna para la duración de las actividades, sólo es necesario conocer su esperanza y varianza. Sin embargo, en la realidad no es sencillo determinar estos valores. ¿Cuál es la esperanza o media de la duración de una actividad? Es una pregunta que no es fácil responder a no ser que se disponga de datos históricos, pero, mucho menos lo es si preguntamos por la varianza, un valor mucho menos intuitivo.

Para solventar esta dificultad tradicionalmente se ha supuesto una determinada distribución para la que se puede obtener la media y la varianza a partir de valores mucho más intuitivos y fáciles de determinar: la distribución beta. Esta distribución supone que la variable toma valores entre un mínimo y un máximo y además, de todas las posibles, se utilizan las que tienen una única moda. Conociendo estos tres valores es posible determinar la esperanza y varianza de la distribución. Así pues, haciendo esta suposición, lo que es necesario conocer es la duración más optimista de la actividad (a: el mínimo), la duración más pesimista (b: el máximo) y la duración más probable (m: la moda), que son conceptos mucho más intuitivos y fáciles de determinar. A partir de ellos se obtienen los valores de la esperanza y la varianza como $E\left[D_{ij}\right]=\frac{a+b+4m}{6}$ y $V\left[D_{ij}\right]=\frac{(b-a)^2}{36}$.

El procedimiento es sencillo y de ahí su uso tan generalizado, pero, como ya se ha dicho, las hipótesis que se formulan no están siempre justificadas. A continuación se presentan algunos comentarios a estas hipótesis:

1. La hipótesis de independencia de las variables no siempre es razonable, ya que esto supone que si una variable se alarga o se reduce en tiempo, no va a afectar a la distribución de las otras variables, cuando en la realidad puede haber razones que hacen que se alargue que también hace que otras lo hagan (factores meteorológicos, por ejemplo), o al revés, al ver que una se ha demorado, que las siguientes se hagan con mayor velocidad para compensar la demora.
2. La hipótesis de que el camino crítico es siempre el mismo tampoco está del todo justificada, ya que en muchos casos se puede esperar un camino crítico pero la duración de otras tareas puede dar lugar a que actividades que por su duración media no sean críticas, se demoren, produciendo otro camino con mayor duración.
3. La hipótesis de normalidad de la distribución de la duración del camino se justifica por el teorema central del límite, pero este resultado es asintótico, es decir, se verifica en el límite, y puede ser utilizado para aproximar distribuciones cuando es un número grande de variables las que se suman. Por lo tanto, si en el camino crítico no resulta un número grande de actividades, la distribución de la suma puede estar muy lejos de aproximarse a una distribución normal.

4. La hipótesis de seguir una distribución beta las duraciones de las actividades es, probablemente, la hipótesis más fácil de asumir. Por una parte, esta hipótesis no siempre es necesaria, ya que si la esperanza y varianza de la distribución es conocida o estimada no es necesario hacerla. Realmente, sólo es necesaria cuando los datos que se conocen son el tiempo mínimo, el máximo y el más probable, e incluso en estos casos, la distribución beta se presenta como la más apropiada para ajustar una distribución de una variable que toma valores en un intervalo acotado, ya que sus dos parámetros y su expresión la hacen muy flexible, pudiendo tomar formas muy diversas.

A pesar de los comentarios anteriores, el método PERT es un método muy apropiado para aproximar la duración de un proyecto, sobre todo por la facilidad de sus cálculos. Sin embargo, hay que tener en cuenta que es una aproximación, que en algunos casos puede estar muy lejos de la realidad, tanto más cuanto menos asumibles sean las hipótesis anteriores. En el caso de querer obtener algo más preciso sin ajustarse a estas hipótesis, la alternativa más razonable es utilizar la simulación para hacerlo.

3.5 Penalizaciones en método PERT: el problema de establecer una fecha de finalización ante riesgo o incertidumbre

La duración de un proyecto, como se ha visto en el apartado anterior, puede ser considerada una variable aleatoria. El método PERT da una estimación (media) de esta duración y puede utilizarse para determinar fechas de finalización que serán superadas sólo con cierta probabilidad determinada a priori.

Sin embargo, esta probabilidad es un parámetro de diseño que puede venir determinada por diversos condicionantes. Una de las situaciones más habituales es que existan costes o penalizaciones asociadas al incumplimiento de la fecha de finalización.

Un caso ejemplo muy claro es cuando se plantea un concurso para adjudicar una obra. Por una parte, una empresa que desee concursar sabe que cuánto menor sea el plazo de finalización más probable es que le sea adjudicado el proyecto. Pero, por otra parte, en este tipo de concursos lo más habitual es que se establezcan penalizaciones por no acabarlo en el plazo previsto, estando la empresa adjudicataria obligada a pagar una multa por ello. Así pues, si la empresa no desea incurrir en una penalización y alarga el plazo de entrega, en general, tendrá que estar dispuesta a hacer una "rebaja" para mejorar su posición frente a otros adjudicatarios. La rebaja habrá de ser mayor cuánto mayor sea el plazo de entrega. Sin embargo, si la empresa realiza el proyecto y por causas aleatorias acaba antes de la fecha prevista, pensará que ha dejado de ganar una cierta suma de dinero, lo que se conoce como "coste de rebaja". ¿Cuál será la duración que establezca el mejor compromiso entre los costes de penalización y de rebaja?

Esta duración será la que minimice la suma del coste de rebaja más el coste de penalización. Sin embargo, esta minimización dado que las duraciones no son exactas se puede plantear en dos contextos muy distintos:

- En situación de *riesgo*: se considera que la distribución de probabilidad de la duración del proyecto es conocida.
- En situación de *incertidumbre*: se considera desconocida la distribución de probabilidad que sigue la duración del proyecto.

Los costes de rebaja o penalización correspondientes a una fecha de entrega establecida y una fecha real de finalización del proyecto se considerarán conocidos o estimados para poder determinar cuál es esa fecha de mejor compromiso, tanto en un contexto como en otro.

3.5.1 Situación de riesgo

En esta sección se considerarán lineales los costes de rebaja y penalización. Así, denotaremos por α al coste unitario de rebaja en que incurre la empresa si finaliza el proyecto antes de la fecha pactada (por ejemplo, en €/día) y por β a la penalización unitaria en caso de que finalice el proyecto después de esa fecha pactada (podrían ser también €/día). Esta hipótesis no es absolutamente necesaria, pero simplifica mucho los cálculos a la hora de presentar el procedimiento.

También se va a considerar que a la empresa le es indiferente perder dinero que dejar de ganarlo, es decir, le resulta tan gravoso perder c € por penalización como dejar de ganar c € por haber rebajado el precio previamente. Esta hipótesis se puede modificar de forma sencilla incrementando uno de los costes unitarios; por ejemplo, si le resulta peor tener que pagar el dinero al final de una penalización, el parámetro β puede ser incrementado.

Sea $f(x)$ la función de densidad conocida de la distribución de la duración del proyecto (en el método PERT esta distribución se asume como normal). Sea a su vez Z la duración establecida de antemano que se desea determinar como óptima para minimizar la suma de los costes esperados de rebaja y penalización.

Si la duración real del proyecto x resulta anterior a la duración establecida de antemano ($x < Z$) el coste en concepto de rebaja será $\alpha(Z-x)$ y, por lo tanto, el coste esperado de rebaja para una fecha dada Z será

$$\int_{-\infty}^{Z} \alpha(Z-x) f(x)dx$$

Por otra parte, si la duración real es posterior a la fecha establecida ($x > Z$), el coste por penalización será $\beta(x-Z)$ y su coste esperado por lo tanto será

$$\int_{Z}^{\infty} \beta(x-Z) f(x)dx$$

Así, el coste esperado que se desea minimizar es la suma de ambos costes

$$\min \int_{-\infty}^{Z} \alpha(Z-x) f(x)dx + \int_{Z}^{\infty} \beta(x-Z) f(x)dx$$

Para obtener el valor óptimo de Z basta con derivar respecto a esta variable e igualar a 0. Aplicando la regla de Leibnitz (dado que en ambos casos la función dentro de la integral es 0 en Z) la ecuación resultante es:

$$\alpha \int_{-\infty}^{Z} f(x)dx - \beta \int_{Z}^{\infty} f(x)dx = 0$$

Reordenando términos y dado que $\int_{-\infty}^{\infty} f(x)dx = 1$ por ser función de densidad, el valor que buscamos de duración del proyecto es Z tal que

$$\int_{-\infty}^{Z} f(x)dx = P(x \leq Z) = \frac{\beta}{\alpha + \beta}$$

Este valor en el caso de la normal puede ser buscado en las tablas o calculado por algún programa estadístico. Para cualquier otra distribución puede buscarse en sus tablas, si está tabulada, o calculada directamente.

3.5.2 Situación de incertidumbre

En este caso, la distribución de probabilidad de la duración del proyecto se considera desconocida. Para poder hacer un análisis de las opciones y tomar una decisión, este problema se debe plantear como un problema de decisión con incertidumbre.

Para ello, se definen las alternativas o estrategias como las posibles duraciones del proyecto a considerar y como estados de la naturaleza o escenarios las posibles duraciones reales que pueden darse o que se consideren relevantes. Al fin, debe quedar una matriz cuadrada pues todas las opciones que se puedan dar podrían ser duraciones a establecer de antemano.

En la tabla de decisión, se pondrían los costes asociados a esa decisión y ese estado de la naturaleza (si la decisión es inferior al estado de la naturaleza será una penalización y si es superior será un coste de rebaja).

Por ejemplo, si un proyecto puede durar con duraciones mínimas de las actividades 40 unidades de tiempo y con duraciones máximas de las actividades 50 unidades de tiempo, la tabla de decisión tendría 11 posibles escenarios que serían las posibles duraciones. A su vez, como decisiones se pueden considerar todas las posibles duraciones o una selección de ellas. La valoración de una decisión bajo un escenario concreto será la penalización en que se incurra, bien por coste de rebaja o por penalización por demora. Supóngase que estos costes son de 10 si es de rebaja y de 20 por demora, la tabla sería la siguiente:

Escenarios / Decisiones	40	41	42	43	44	45	46	47	48	49	50
40	0	20	40	60	80	100	120	140	160	180	200
41	10	0	20	40	60	80	100	120	140	160	180
42	20	10	0	20	40	60	80	100	120	140	160
43	30	20	10	0	20	40	60	80	100	120	140
44	40	30	20	10	0	20	40	60	80	100	120

45	50	40	30	20	10	0	20	40	60	80	100
46	60	50	40	30	20	10	0	20	40	60	80
47	70	60	50	40	30	20	10	0	20	40	60
48	80	70	60	50	40	30	20	10	0	20	40
49	90	80	70	60	50	40	30	20	10	0	20
50	100	90	80	70	60	50	40	30	20	10	0

Aplicando los criterios de teoría de la decisión con incertidumbre se obtienen distintas decisiones, de modo que al fin se ha de llegar a un compromiso con una de las decisiones. Así, por ejemplo, si se aplica el criterio de Wald o pesimista, las decisiones se valorarían como:

Decisiones	40	41	42	43	44	45	46	47	48	49	50
Valor	200	180	160	140	120	100	80	70	80	90	100

A la vista de esta tabla, la decisión óptima sería dar una duración prevista de 47 unidades de tiempo.

El criterio optimista no tendría sentido ya que en el mejor de los casos todas las opciones tienen una penalización de 0.

Por último, aplicar el criterio de Savage tampoco tiene sentido ya que es el mismo de Wald, puesto que dado un escenario la solución óptima para él tiene valor 0 y, por lo tanto, los valores que aparecen en la tabla ya son de por sí, penalizaciones o costes de oportunidad, por lo que la tabla no cambia.

3.6 Coste en el método del camino crítico: alternativas en el desarrollo de una actividad

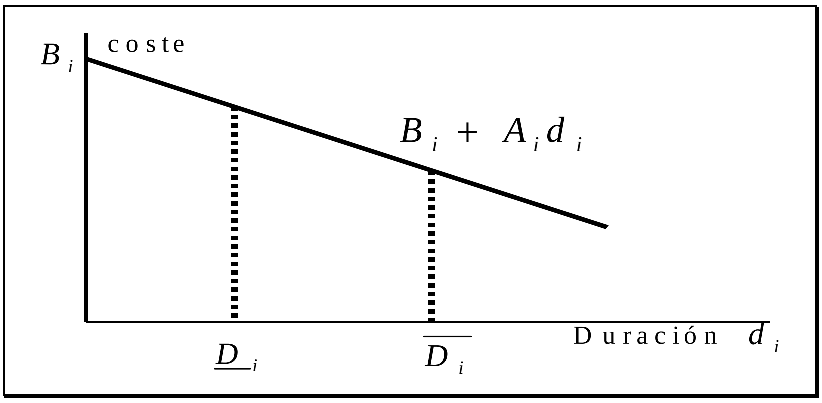
Es muy habitual que la duración de una actividad no esté prefijada de antemano, pero no por ser aleatoria, sino porque existan diversas alternativas con distintos costes, dependiendo de los recursos asignados.

Existen técnicas heurísticas para resolver el problema de elegir la duración óptima de las actividades y, por lo tanto, la duración y coste del proyecto asociados a esas duraciones. Uno de los algoritmos diseñados con tal fin es del Ackoff y Sasieni. Sin embargo, es con la programación matemática con la que mejores resultados se pueden obtener, pudiendo hacer modelos adaptados al caso específico que se esté tratando.

A continuación se presenta un modelo a modo de ejemplo de cómo formular mediante programación matemática un problema de planificación.

Sea un proyecto con un conjunto de tareas I y unas precedencias definidas entre estas tareas P, de modo que si $(i, j) \in P$ quiere decir que la actividad j no puede comenzar hasta que no haya acabado la tarea i.

Las tareas tienen asociado un coste a su duración, que vamos a suponer lineal, de modo que cuanto menor sea esta duración mayor será su coste, es decir, la función de coste es de la forma $B_i + A_i d_i$ ($A_i \leq 0$) dentro de un intervalo $\left[\underline{D_i}, \overline{D_i}\right]$.



Se supone además que existe una duración total de proyecto que no puede ser superada T y el objetivo es obtener una planificación de mínimo coste que cumpla las precedencias establecidas y no supere el tiempo máximo de proyecto.

A continuación se presenta el modelo, donde las incógnitas son d_i la duración de las actividades y x_i el instante más temprano en que puede comenzarse una actividad (que será el instante para el que quede programada):

$$\min \sum_{i \in I} (B_i + A_i d_i)$$
$$x_i + d_i \le x_j \quad \forall (i, j) \in P$$
$$\underline{D}_i \le d_i \le \overline{D}_i \quad \forall i \in I$$
$$x_i + d_i \le T \quad \forall i \in I$$
$$x_i, d_i \ge 0 \quad \forall i \in I$$

El problema puede ser considerado como de programación lineal paramétrica si la duración total del proyecto T es considerada un parámetro. En tal caso, la solución del problema serían unas duraciones y costes de actividades asociados a cada una de las duraciones consideradas para el parámetro.

3.7 Programación de proyectos con recursos limitados: nivelación y asignación de recursos

En los métodos de programación y control de proyectos vistos anteriormente existe la hipótesis implícita de que los distintos recursos necesarios para desarrollar las actividades existen en cantidades ilimitadas. Obviamente, se trata de un supuesto muy fuerte y en muchos casos poco realista. En esta sección, se van a analizar dos problemas asociados a los recursos, el problema de nivelación y el problema de asignación de recursos limitados.

3.7.1 Nivelación de recursos

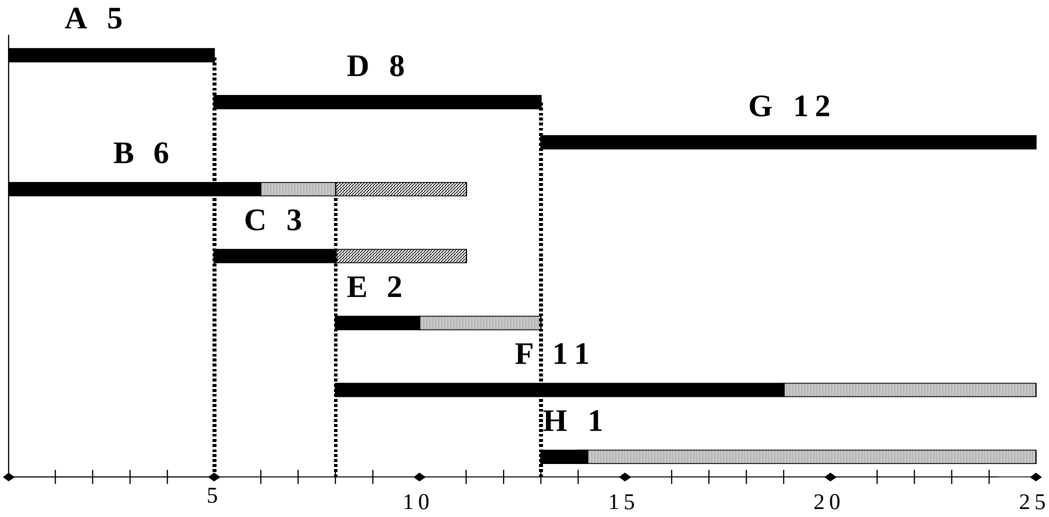
Este problema se plantea cuando, aun existiendo los recursos sin ser una restricción, éstos son utilizados de forma desigual en el tiempo. El objetivo es nivelar o repartir el uso de estos recursos en el tiempo de la forma más equilibrada posible sin alargar la duración del proyecto, es decir, la duración dada por el camino crítico.

Para abordar este problema, dada una programación de un proyecto, lo primero es hacer una representación gráfica y clara de la carga a lo largo del tiempo del recurso que se esté analizando. Para ello, en el gráfico Gantt que representa el calendario del proyecto se añade una fila final en la que se contabiliza esta carga por periodos (días, por ejemplo) y a partir de esta fila se hace una representación gráfica en la que en el eje de abcisas se representa el tiempo y en el eje de ordenadas la carga de trabajo.

Para ilustrar esta representación, supongamos que en el ejemplo de la sección 3.3 se considera la mano de obra como un recurso que se desea que esté nivelado y los requerimientos de mano de obra de cada una de estas tareas son los que se recogen en la siguiente tabla:

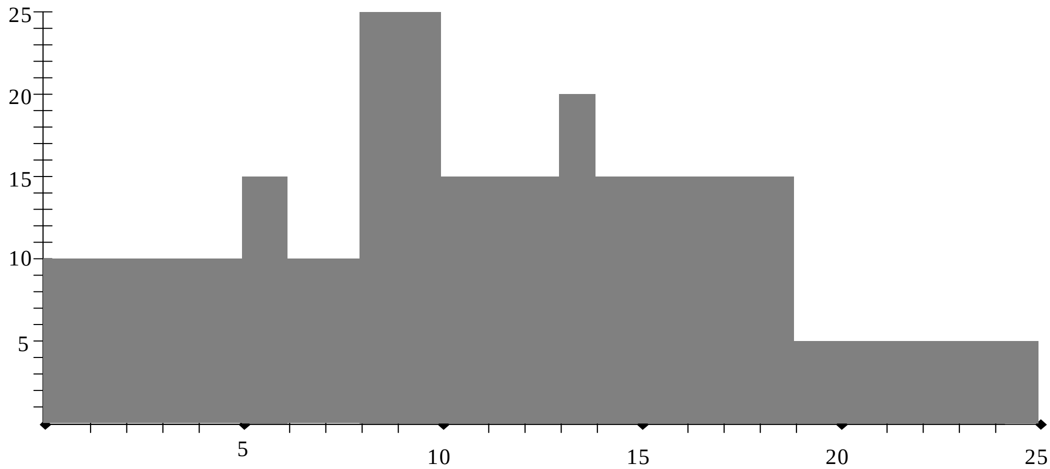
Tareas	A	B	C	D	E	F	G	H
Mano de obra	5	5	5	5	10	10	5	5

La carga de trabajo para la programación de tiempo más temprano mostrada en el siguiente diagrama de Gantt se recoge en la tabla presentada a continuación:



Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Carga	10	10	10	10	10	15	10	10	25	25	15	15	15	20	15	15	15	15	15	5	5	5	5	5	5

Gráficamente, esta carga se representa de la siguiente forma:



El objetivo de la nivelación de recursos será obtener una carga lo más uniforme posible a lo largo del tiempo. La programación ideal del proyecto sería que esa carga fuera siempre la misma, lo que en el ejemplo corresponde a un valor de 305 (que es la carga total) entre los días que dura el proyecto que son 25. Es decir, el ideal sería tener una carga constante diaria de 12.2 operarios. Sin embargo, resulta obvio que en este caso es imposible, de modo que un posible objetivo es minimizar la varianza de la programación resultante. Esta varianza puede ser calculada como la media de la suma de los cuadrados de las cargas menos la carga media al cuadrado, es decir, sumar los cuadrados de las cargas y dividir por el número de días y a ese valor restarle el cuadrado de la carga media (en el ejemplo, 12.2^2). Sin embargo, optimizar esta función u optimizar esta función sin la constante y sin dividir nos va a dar el mismo resultado, por lo que se puede plantear el problema como minimizar la suma de los cuadrados de las cargas diarias.

Para resolver este problema, plantear un modelo de programación matemática es el método más eficiente y capaz de asegurarnos que la solución alcanzada es la óptima. Sin embargo, también existen algunos algoritmos heurísticos que pueden dar una solución razonable en poco tiempo y sin tener que resolver un problema de optimización. A continuación, se muestra un algoritmo heurístico de este tipo debido a Burgess y Killebrew, que siendo uno de los pioneros en este tema también está considerado uno de los más eficientes.

3.7.1.1 Algoritmo Burgess-Killebrew

PASO 1:

Elegir la actividad no crítica con mayor o más avanzado instante más temprano de finalización. Retrasar esta actividad de unidad en unidad de tiempo hasta lo que le permita su holgura total, eligiendo como fecha de inicio aquélla que dé menor valor para la suma de los cuadrados de las cargas diarias.

PASO 2:

Repetir el paso 1 una por una para las actividades no críticas con mayor instante más temprano de finalización, pero que no hayan sido analizadas hasta el momento, hasta que todas las actividades no críticas hayan sido analizadas. En caso de empate, tomar primero la que tenga mayor holgura. (Atención a las relaciones de precedencia al entrar en retrasos en la parte de la holgura total que no es holgura libre).

PASO 3:

Repetir los pasos 1 y 2 hasta que no haya ninguna disminución en los cuadrados de las cargas.

Apliquémoslo a nuestro ejemplo.

PASO 1: La actividad no crítica que se selecciona en primer lugar es la actividad F. Las cargas de trabajo, si no se retrasa su inicio, y sus cuadrados serán

Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Carga	10	10	10	10	10	15	10	10	25	25	15	15	15	20	15
Carga^2	100	100	100	100	100	225	100	100	625	625	225	225	225	400	225

Días	16	17	18	19	20	21	22	23	24	25	SUMA
Carga	15	15	15	15	5	5	5	5	5	5	
Carga^2	225	225	225	225	25	25	25	25	25	25	4525

Si se retrasa 1 día estos valores resultan ser:

Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Carga	10	10	10	10	10	15	10	10	15	25	15	15	15	20	15
Carga^2	100	100	100	100	100	225	100	100	225	625	225	225	225	400	225

Días	16	17	18	19	20	21	22	23	24	25	SUMA
Carga	15	15	15	15	15	5	5	5	5	5	
Carga^2	225	225	225	225	225	25	25	25	25	25	4325

Con 2 días sería:

Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Carga	10	10	10	10	10	15	10	10	15	15	15	15	15	20	15
Carga^2	100	100	100	100	100	225	100	100	225	225	225	225	225	400	225

Días	16	17	18	19	20	21	22	23	24	25	SUMA
Carga	15	15	15	15	15	15	5	5	5	5	
Carga^2	225	225	225	225	225	225	25	25	25	25	4125

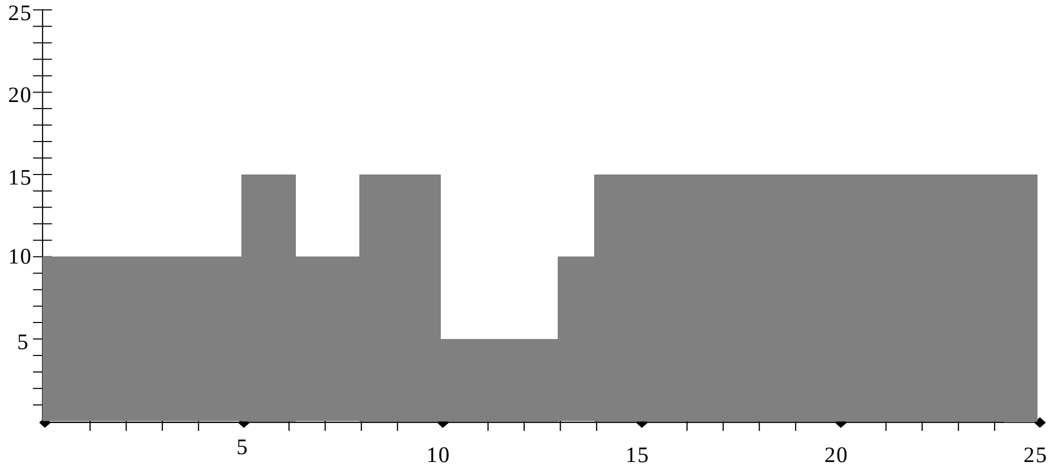
Si se retrasa 3 días el resultado de la suma de los cuadrados no varía, al igual que si se retrasa 4 o 5 días, siendo en todos los casos 4125. Si se retrasa 6 días, la tabla sería:

Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Carga	10	10	10	10	10	15	10	10	15	15	5	5	5	10	15
Carga^2	100	100	100	100	100	225	100	100	225	225	25	25	25	100	225

Días	16	17	18	19	20	21	22	23	24	25	SUMA

Carga	15	15	15	15	15	15	15	15	15	15	
Carga^2	225	225	225	225	225	225	225	225	225	225	4025

De todas las opciones se toma la última ya que es la que resulta en una suma de cuadrados menor. De este modo la carga de trabajo queda así:



PASO 2: Se repite el proceso, eligiendo ahora la siguiente actividad que sería la H. Esta actividad si se retrasa en una unidad daría la tabla:

Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Carga	10	10	10	10	10	15	10	10	15	15	5	5	5	5	20
Carga^2	100	100	100	100	100	225	100	100	225	225	25	25	25	25	400

Días	16	17	18	19	20	21	22	23	24	25	SUMA
Carga	15	15	15	15	15	15	15	15	15	15	
Carga^2	225	225	225	225	225	225	225	225	225	225	4125

El valor de la suma de cuadrados empeora y para retrasos mayores sería el mismo valor, luego no interesa retrasar esta actividad.

La siguiente actividad sería la E. Es fácil comprobar que moverla dentro de su holgura no modifica la suma de cuadrados, luego la dejamos como está.

A continuación sería la C. Si se retrasa esta actividad en una unidad la nueva tabla (teniendo en cuenta que habría que retrasar la E y la F, aunque esta última ya está retrasada) resulta ser:

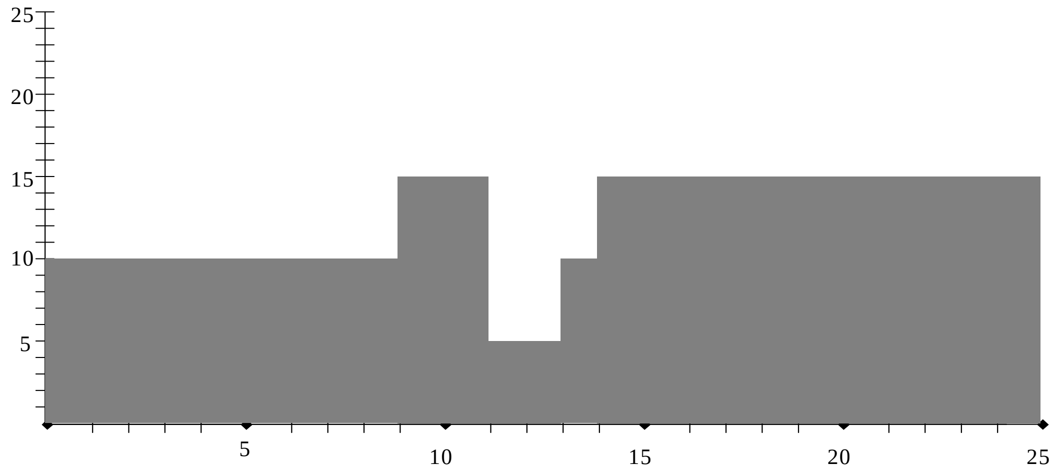
Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Carga	10	10	10	10	10	10	10	10	10	15	15	5	5	10	15
Carga^2	100	100	100	100	100	100	100	100	100	225	225	25	25	100	225

Días	16	17	18	19	20	21	22	23	24	25	SUMA
Carga	15	15	15	15	15	15	15	15	15	15	
Carga^2	225	225	225	225	225	225	225	225	225	225	3975

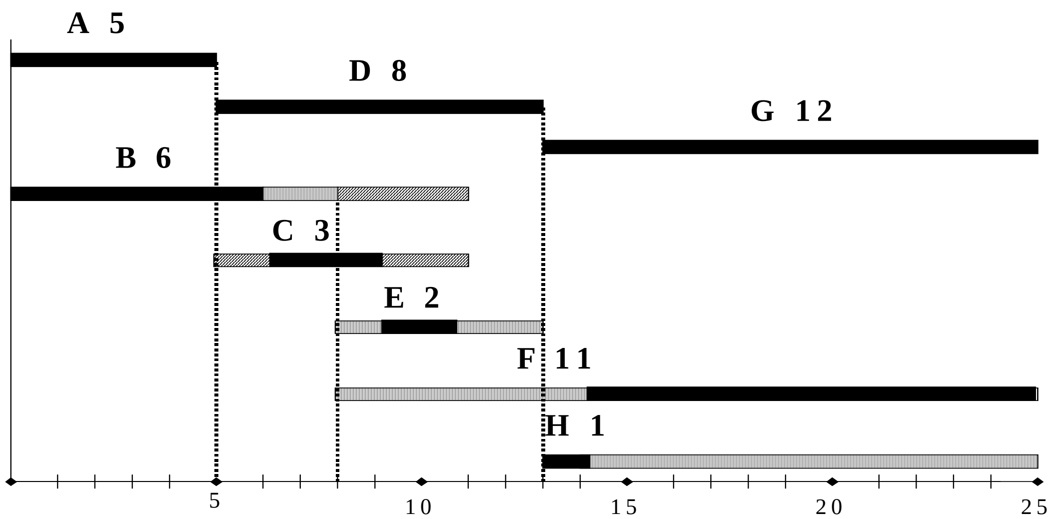
De modo, que interesa. Para retrasos mayores se obtienen sumas de cuadrados de igual valor de modo que nos quedamos con esta última programación.

La siguiente y última actividad de esta iteración, es la actividad B. Para esta actividad es fácil comprobar que cualquier retraso o produce la misma suma de cuadrados o empeora (con el cuidado de retrasar la actividad E cuando sea necesario).

De esta forma la tabla final de cargas de esta iteración que tendríamos es la de la tabla anterior que corresponde al gráfico siguiente:



El diagrama de Gantt asociado sería:



PASO 3: Hay que repetir el proceso anterior sobre esta nueva programación para ver si se puede reducir la suma de los cuadrados de las cargas. Es fácil comprobar que este resultado no es mejorable.

3.7.1.2 Nivelación de recursos mediante programación matemática

A continuación, se muestra un modelo de programación matemática para este mismo problema y la solución aplicada a este caso ejemplo. La primera ventaja de la programación matemática es que nos asegura que llegaremos al óptimo, pero además nos permite hacer modelos más flexibles pudiendo nivelar más de un recurso e incluso incorporar la opción de fraccionar las actividades.

Utilizaremos la siguiente notación:

j : actividades del proyecto

k : recursos a nivelar

p : periodos de tiempo (de 1 a la duración del proyecto)

d_j : duración de la actividad j

car_{jk} : carga del recurso k que utiliza la actividad j por unidad de tiempo

$G=(J,Q)$: grafo de precedencias, de modo que los nodos son las actividades y los arcos las relaciones de precedencia directas, es decir, existe un arco en Q si el nodo inicial corresponde a una actividad que ha de acabar antes que la correspondiente al nodo final.

Como variables vamos a considerar:

$$X_{jp}=\begin{cases}1 & \text{si la actividad } j \text{ se realiza durante periodo } p\\ 0 & \text{en otro caso}\end{cases}$$

Las restricciones serán:

a) Todas las actividades han de hacerse dentro del tiempo permitido.

$$\sum_p X_{jp}=d_j \quad \forall j$$

b) Respetar las relaciones de precedencia del grafo G .

Estas prelaciones pueden representarse de muy diversas formas. Una de ellas es mediante el siguiente conjunto de restricciones

$$\sum_{p'<p} X_{j'p'} \geq d_{j'} X_{jp} \quad \forall p, \forall (j',j) \in Q$$

c) Las actividades se han de hacer sin interrupción. Obsérvese que esta hipótesis está subyacente en todo lo visto anteriormente, pero dada la flexibilidad de la programación matemática, podría no incluirse en este caso. Así pues, para las actividades para las que sea un requisito imprescindible se añadirán las siguientes restricciones:

$$X_{jp}+X_{j\,p+d_j}\leq 1 \quad \forall j,p$$

Por último, hay que expresar la función objetivo en términos de nuestras variables. Como se vio al principio de esta sección, el objetivo puede ser minimizar la suma de los cuadrados de las cargas, es decir,

$$\min \sum_p \sum_k \left(\sum_j car_{jk} X_{jp} \right)^2$$

El modelo así planteado resulta de programación cuadrática. Existiría una formulación lineal alternativa que minimiza las desviaciones al valor medio. Para ello habría que añadir unas variables de desviaciones:

N_{pk}: desviación inferior (por debajo) durante el periodo p al valor medio de la carga del recurso k

S_{pk}: desviación superior (por encima) durante el periodo p al valor medio de la carga del recurso k

y unas restricciones de desviaciones:

$$\sum_j car_{jk} X_{jp} + N_{pk} - S_{pk} = \overline{car}_k \quad \forall p, k$$

donde $\overline{car}_k$ representa la carga media del recurso k. Con esta formulación la función objetivo sería:

$$\min \sum_p \sum_k (N_{pk} + S_{pk})$$

Para el caso ejemplo que estamos tratando, la formulación en GAMS de este modelo se presenta a continuación.

```
$TITLE Nivelación de recursos


OPTION OPTCR = 0


SETS J / A, B, C, D, E, F, G, H /

     K / MOBRA /

     P / p1*p25 /

     preced(j,j) precedencia de j1 con respecto a j2

       / A.C, A.D, B.E, B.F, C.E, C.F, D.G, D.H, E.G /
```

```
ALIAS (jj,j), (pp,p)

PARAMETERS
d(j)      duración  / A 5, B 6, C 3, D 8, E 2, F 11, G 12, H 1 /
cuadrados cuadrados de las cargas

TABLE car(j,k) carga de recursos por actividad
   MOBRA
A    5
B    5
C    5
D    5
E   10
F   10
G    5
H    5

binary   variables X(j,p)
positive variables N(p,k), S(p,k)
         variables objetivo

EQUATIONS
dura(j)             duración total de actividad j
precedencias(p,j,j) relaciones de precedencia
nointerrumpir(j,p)  no interrumpir las actividades
desviaciones(k,p)   desviaciones a media de la carga del recurso
fobjetivo           función objetivo ;

dura(j)..                 sum(p, X(j,p)) =E= d(j) ;
precedencias(p,preced(jj,j))..
                          sum(pp $(ord(pp)<ord(p)), X(jj,pp)) =G= d(jj)*X(j,p) ;
nointerrumpir(j,p+d(j)).. X(j,p) + X(j,p+d(j)) =L= 1 ;
desviaciones(k,p)..       sum(j, car(j,k)*X(j,p)) + N(p,k) - S(p,k) =E=
                          sum(j, car(j,k)*d(j))/card(p) ;
fobjetivo..               objetivo =E= sum((p,k), N(p,k) + S(p,k)) ;
```

```
model NIVELACION /all/

solve NIVELACION using MIP minimizing objetivo ;


cuadrados = sum((p,k), sum(j, car(j,k)*X.L(j,p))**2)
```

Al resolver el problema del ejemplo con este modelo se obtiene una planificación alternativa a la obtenida mediante el algoritmo de Burgess-Killebrew, con igual suma de cuadrados.

3.7.2 Asignación de recursos limitados

El problema de asignar recursos limitados en un proyecto es un problema muy habitual en la vida real. Este problema se distingue del anterior fundamentalmente porque aquí hay un límite o disponibilidad de los recursos en cada periodo de tiempo, de modo que no se pueden programar tareas en un periodo de tiempo de forma que la suma de las cantidades del recurso que necesitan esas actividades sea mayor que la disponibilidad del recurso. En este caso, habrá tareas que aun no existiendo una relación de precedencia entre ellas, haya que decidir en qué orden se hacen para no superar esa disponibilidad. En el ejemplo de la sección anterior podría ser un problema si la disponibilidad del recurso mano de obra es de 10 operarios, entonces la planificación prevista no se podría llevar a cabo. En particular, las tareas F y G, aun no teniendo una relación de precedencia se ve que no podrán realizarse a la vez y habrá que decidir cuál se hace antes.

En este planteamiento, el uso de los recursos en cada periodo de tiempo no es un objetivo que se quiere nivelar, sino una restricción rígida que se tiene que cumplir. Por otra parte, lo que ahora será un objetivo es terminar el proyecto cuánto antes.

Para este problema de minimizar la duración del proyecto asignando los recursos de forma factible, de nuevo la programación matemática es la herramienta que nos asegura el óptimo y más flexibilidad nos puede dar. Sin embargo, como plantea cierta dificultad y, en ocasiones, los modelos resultan muy grandes en tamaño (y por lo tanto su resolución puede ser demasiado lenta), se han desarrollado algunas heurísticas con este fin. En este apartado, se verá primero una heurística y después algunos modelos de programación matemática.

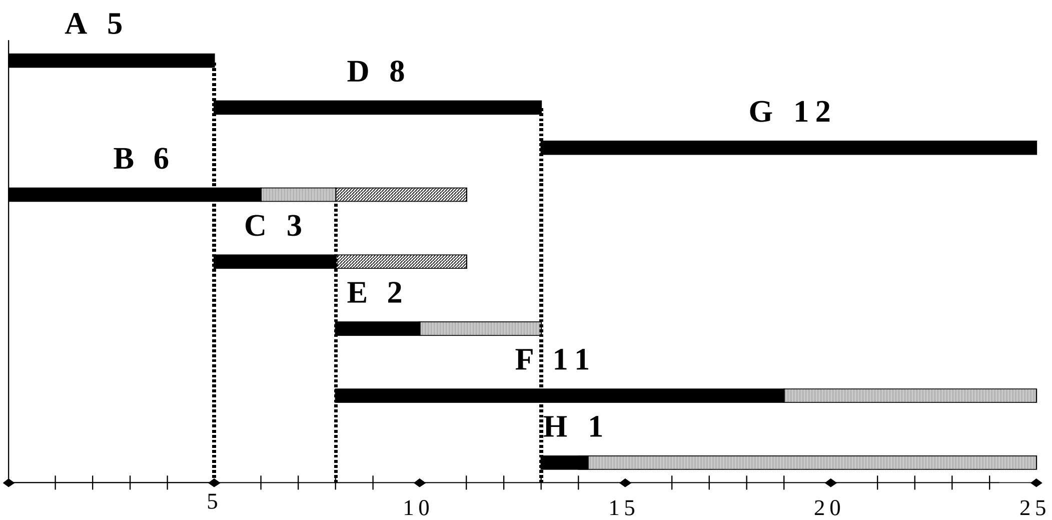
3.7.2.1 Algoritmo heurístico para el problema de asignación de recursos

El algoritmo que se va a presentar es un algoritmo sencillo, aunque hay otros más complejos que han sido implantados en diversos programas que son ampliamente utilizados (ALTAI, CORUA, RAMPS, RPSM, SPAR, etc.).

El algoritmo que se presenta, debido a Wiest y Levy, se basa en la idea de ir asignando los recursos periodo a periodo, empezando por el primero. La idea es programar las actividades que pueden realizarse en ese periodo, siempre que no se supere la disponibilidad de los recursos, dando prioridad a las que menor holgura tienen (las actividades críticas serán las de mayor prioridad).

Veamos el ejemplo de la sección anterior, suponiendo las mismas necesidades de mano de obra de cada actividad, pero con un límite de 10 operarios para realizar las tareas.

Partimos de la programación inicial resultante de aplicar el método del camino crítico, recogiendo la carga correspondiente a cada periodo.



Días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Carga	10	10	10	10	10	15	10	10	25	25	15	15	15	20	15	15	15	15	15	5	5	5	5	5	5

El algoritmo trabajaría así:

Día 1: Se asigna A y B pues no superan la disponibilidad.

Día 2: Se asigna A y B pues no superan la disponibilidad.

Día 3: Se asigna A y B pues no superan la disponibilidad.

Día 4: Se asigna A y B pues no superan la disponibilidad.

Día 5: Se asigna A y B pues no superan la disponibilidad.

Día 6: Sigue B (ya iniciada) y se asigna D (no tiene holgura), siendo la carga de 10. Se retrasa el inicio de C.

Día 7: Sigue D y se asigna el inicio de C. La carga es de 10.

Día 8: Siguen D y C.

Día 9: Siguen D y C, con carga 10, luego, no pueden empezar ni E ni F (que estaba así programado).

Día 10: Sigue D, pero no se pueden programar ni E ni F ya que consumen 10 unidades de recurso que junto a las 5 de D superan la disponibilidad (G y H ni se consideran por las precedencias).

Día 11: Sigue D.

Día 12: Sigue D.

Día 13: Sigue D.

Día 14: D ha acabado y hay que programar la que menor holgura tenga. Sería G (es crítica), pero no puede empezar hasta que no se acabe E, en este punto podemos asegurar que el proyecto no acabará en el tiempo mínimo que se había establecido. Siguiendo el algoritmo la que tiene menor holgura de las restantes es E, así pues se programa E que consume 10 unidades de recurso.

Día 15: Sigue E.

Día 16: Acabada E, se programa G ya que es la de menor holgura (crítica) consumiendo 5. F no puede programarse por consumir 10. H también se programa pues consume 5. Total: 10.

Día 17: Sigue sólo G, pues F consume 10 unidades que junto a las 5 de G superaría la disponibilidad.

Día 18: Sigue G.

Día 19: Sigue G.

Día 20: Sigue G.

Día 21: Sigue G.

Día 22: Sigue G.

Día 23: Sigue G.

Día 24: Sigue G.

Día 25: Sigue G.

Día 26: Sigue G.

Día 27: Sigue G y acaba.

Día 28: Empieza F, carga de 10.

Día 29: Sigue F.

Día 30: Sigue F.

Día 31: Sigue F.

Día 32: Sigue F.

Día 33: Sigue F.

Día 34: Sigue F.

Día 35: Sigue F.

Día 36: Sigue F.

Día 37: Sigue F.

Día 38: Acaba F y el proyecto.

Como se puede apreciar, el proyecto se ha alargado en 13 días al tener limitado el recurso de la mano de obra.

3.7.2.2 Modelos de programación matemática para el problema de asignación de recursos

Existen diversas formulaciones para este problema. Una de ellas sería plantear un modelo similar al de la sección de nivelación de recursos pero con variantes. En concreto, la duración del proyecto para a ser un objetivo en lugar de una restricción y el uso de los recursos una restricción en lugar de un objetivo.

La formulación podría ser la siguiente:

j : actividades del proyecto

k : recursos a nivelar

p : periodos de tiempo (de 1 a un valor que seguro no se vaya a superar; sin un análisis previo puede ser la suma de todas las duraciones, pero es mejor un pequeño análisis para reducir este índice; podría ser el resultado de la heurística de la sección anterior)

d_j : duración de la actividad j

car_{jk} : carga del recurso k que utiliza la actividad j por unidad de tiempo

$disp_k$: disponibilidad del recurso k

$G = (J, Q)$: grafo de precedencias, de modo que los nodos son las actividades y los arcos las relaciones de precedencia directas, es decir, existe un arco en Q si el nodo inicial corresponde a una actividad que ha de acabar antes que la correspondiente al nodo final.

Como variables vamos a considerar:

$$X_{jp} = \begin{cases} 1 & \text{si la actividad } j \text{ se realiza durante periodo } p \\ 0 & \text{en otro caso} \end{cases}$$

Las restricciones serán:

a') Todas las actividades han de hacerse dentro del tiempo permitido

$$\sum_p X_{jp} = d_j \quad \forall j$$

b') Respetar las relaciones de precedencia del grafo G .

$$\sum_{p'<p} X_{j'p'} \geq d_{j'} X_{jp} \quad \forall p, \forall (j', j) \in Q$$

c') Las actividades se han de hacer sin interrupción.

$$X_{jp} + X_{j\,p+d_j} \leq 1 \quad \forall j, p$$

d') No superar la disponibilidad de los recursos:

$$\sum_j car_{jk} X_{jp} \le disp_k \quad \forall p,k$$

Por último, hay que expresar la función objetivo en términos de nuestras variables. Una forma es ponderar los periodos de tiempo, de modo que se penalice la programación de una tarea en los periodos altos. Esto se puede hacer ponderando de menos a más las variables correspondientes a los periodos, es decir, el objetivo sería:

$$\min \sum_p \alpha_p (\sum_j X_{jp})$$

definiendo las ponderaciones de modo que cumplan $\alpha_p > \alpha_{p'}$ si $p > p'$. Esta formulación tiene un riesgo y es que como consideramos todas las tareas asignadas a un periodo, podemos penalizar más el tener 2 tareas en un periodo que alargar el proyecto. Para no correr este riesgo, se deberían incluir otras variables binarias que indiquen si en un periodo se programa alguna tarea o no, en lugar de contar el número de tareas. Las nuevas variables serían

$$Y_p = \begin{cases} 1 & \text{si se programa alguna actividad durante el periodo } p \\ 0 & \text{en otro caso} \end{cases}$$

Con estas nuevas variables habría que añadir una restricción que las ligue con las anteriores

$$\sum_j X_{jp} \le MY_p \quad \forall p$$

siendo M una cota que no se pueda superar y la nueva función objetivo sería

$$\min \sum_p \alpha_p Y_p$$

La formulación en GAMS sería la siguiente:

```
$TITLE Asignación de recursos limitados


OPTION OPTCR = 0


SETS J / A, B, C, D, E, F, G, H /

     K / MOBRA /
```

```
     P / p1*p38 /
     preced(j,j) precedencia de j1 con respecto a j2
       / A.C, A.D, B.E, B.F, C.E, C.F, D.G, D.H, E.G /

ALIAS (jj,j), (pp,p)

PARAMETERS
d(j)    duración  / A 5, B 6, C 3, D 8, E 2, F 11, G 12, H 1 /
disp(k) disponibilidad de recursos / MOBRA 10 /
alfa(p) ponderación de los periodos ;

alfa(p) = ord(p)

TABLE car(j,k) carga de recursos por actividad
   MOBRA
A    5
B    5
C    5
D    5
E   10
F   10
G    5
H    5

binary variables X(j,p), Y(p)
       variables objetivo

EQUATIONS
dura(j)              duración total de actividad j
precedencias(p,j,j) relaciones de precedencia
nointerrumpir(j,p)  no interrumpir las actividades
disponibilidad(k,p) disponibilidad del recurso k en periodo p
relacionvar(p)      relación uso de periodo
fobjetivo           función objetivo ;

dura(j)..                 sum(p, X(j,p)) =E= d(j) ;
```

```
precedencias(p,preced(jj,j))..
                          sum(pp $(ord(pp)<ord(p)), X(jj,pp)) =G= d(jj)*X(j,p) ;
nointerrumpir(j,p+d(j)).. X(j,p) + X(j,p+d(j)) =L= 1 ;
disponibilidad(k,p)..     sum(j, car(j,k)*X(j,p)) =L= disp(k) ;
relacionvar(p)..          sum(j, X(j,p)) =L= card(j)*Y(p) ;
fobjetivo..               objetivo =E= sum(p, alfa(p)*Y(p));

model ASIGNACION /all/
solve ASIGNACION using MIP minimizing objetivo
```

Al resolver este ejemplo, se observa que no se puede obtener una duración inferior a los 38 días obtenidos con el método heurístico. Por otra parte, siendo un proyecto tan sencillo, el tiempo de resolución del modelo fue de 18 segundos con el optimizador CPLEX 9.0 en un PC a 1.1 GHz, lo que hace suponer que el tiempo para proyectos mayores puede ser inviable. Una reformulación del problema puede ser más eficiente.

A su vez, hay formulaciones alternativas cuando los recursos se consideran individualmente (por ejemplo, una máquina concreta) que resultan más eficientes que aplicar esta formulación.

3.8 Biblioteca de problemas

PROBLEMA 1

Considérese la lista (simplificada) de actividades que intervienen en la construcción de una casa.

Actividad	Descripción	Antecesores	Duración
A	Cimientos	–	5
B	Muros y Techos	A	8
C	Tejado	B	10
D	Cables Eléctricos	B	5
E	Ventanas	B	4
F	Revestimiento	E	6

G	Pintar	C, F	3

1. Dibujar la red del proyecto, determinar la ruta crítica, obtener la holgura total de cada actividad y la holgura libre de cada una.
2. Supóngase que se puede reducir la duración de cada actividad contratando más trabajadores. En la tabla siguiente se dan los costes diarios de la reducción de la duración de las actividades. Establecer el problema de programación lineal que hay que resolver para minimizar el coste total de la terminación del proyecto en 20 días.

Actividad	Coste diario por reducción (€/día)	Máxima reducción (días)
Cimientos	3000	2
Muros y Techos	1500	3
Tejado	2000	1
Cables Eléctricos	4000	2
Ventanas	2000	2
Revestimiento	3000	3
Pintar	4000	1

PROBLEMA 2

El promotor de un concierto de rock en una ciudad debe realizar las tareas que se dan a continuación antes de poder realizar el concierto.

Activ.	Descripción	Antec.	a (mín)	b (máx)	m (más prob.)
A	Encontrar lugar	–	2	4	3
B	Encontrar ingenieros	A	1	3	2
C	Contratar acto inic.	A	2	10	6
D	Anunciar radio y TV	C	1	3	2

E	Instalar venta billetes	A	1	5	3
F	Instal. eléctricas	B	2	4	3
G	Imprimir publicidad	C	3	7	5
H	Arreglar transporte	C	0.5	1.5	1
I	Ensayos	F, H	1	2	1.5
J	Detall. último minuto	I	1	3	2

Considérese que todas las duraciones tienen distribución beta.

1. Dibujar la red de proyecto.
2. Determinar la ruta crítica.
3. Si el promotor quiere tener una probabilidad de 0.99 de terminar la preparación para el 30 de junio, ¿cuándo tendrá que empezar con el trabajo para encontrar un lugar para el concierto?

PROBLEMA 3

Las actividades que configuran un proyecto, sus precedencias, así como sus duraciones más optimistas, pesimistas y más probables vienen recogidas en la siguiente tabla:

Actividad	Predecesores	Dur. optim.	Dur. más prob.	Dur. pesimista
A	–	2	2.5	6
B	–	1	1.5	5
C	–	2	5.5	6
D	A	3	4	5
E	A, B	3	5	7
F	A, B	7	9	11
G	C	5	6	7

H	D, E	10	10	10
I	F	9	9	9
J	F, G	7	8	9

1. Suponiendo que las distribuciones de las duraciones son todas de tipo beta, ¿para qué valor la probabilidad de que el proyecto esté terminado antes de esa fecha es de 0.8? (Nota: El valor de una N(0,1) que deja 0.8 de probabilidad a su izquierda es 0.84).
2. Dar las holguras libres y totales de todas las actividades, suponiendo duraciones medias.

PROBLEMA 4

Un proyecto consiste en las siguientes actividades, siendo posible realizar cada una de las mismas de tres formas diferentes con una duración y un coste distinto para cada una. Las actividades, sus predecesores y sus duraciones (en semanas) y costes se reflejan en la siguiente tabla:

Tareas	Antec.	D. normal	C. normal	D. acel.	C. acel.	D. urg.	C. urg.
A	–	9	3	7	4	5	6
B	A	8	3	6	4	4	5
C	A	7	2	5	3	3	5
D	C	10	3	8	4	6	6
E	B, D	12	4	10	5	7	7
F	D	10	4	8	5	6	6

1. Dar la duración mínima del proyecto y las holguras libre y total de las actividades con una duración normal de éstas.
2. Plantear un problema de programación lineal para resolver el problema de elegir cómo llevar a cabo cada actividad de entre las tres formas posibles y en qué instante,

de modo que se minimice el coste, se respeten las precedencias, el proyecto no dure más de 25 semanas y teniendo en cuenta que si la actividad A se lleva a cabo de forma urgente la F también ha de llevarse a cabo de forma urgente.

PROBLEMA 5

Las estimaciones de la duración mínima, más probable y máxima de las distintas actividades que configuran un proyecto se dan en la siguiente tabla:

Actividad	A, B	C	D, G, I	E, H, J, K	F, L	M	N, O	P	Q	R, S	T	U
Mínima	2	1	3	1	2	1	4	5	7	4	5	1
Más probable	5	4	6	3	3	2	6	6	9	5	8	5
Máxima	7	5	9	5	4	3	10	7	10	6	9	6

Estas actividades están ligadas por las prelaciones siguientes:

A	Precede a D, C, F
B, C	Precede a E, G, H
D	Precede a I
E, F, I	Precede a L, J, K, M
G	Precede a J, K, M
H	Precede a K, M

J, L	Precede a N
K	Precede a O, P, Q, R
M	Precede a O
N, P	Precede a S
Q, O	Precede a U
R, S	Precede a T

1. Se pide dibujar un gráfico que represente las actividades y sus prelaciones, el tiempo medio mínimo en que puede ser acabado el proyecto y los eventos y actividades críticas, suponiendo distribuciones tipo beta.
2. Caracterizar la distribución de probabilidad del tiempo de realización del proyecto.

PROBLEMA 6

Un proyecto consta de las siguientes actividades cuyas duraciones mínimas, más probables y máximas se dan en la siguiente tabla:

Actividad	A	B	C	D	E	F	G	H	I
Mínima	3	2	1	2	2	3	3	8	2
Más probable	6	6	3	3	3	5	5	9	3
Máxima	15	10	5	10	4	7	7	10	4

Las relaciones de precedencias entre tareas son las siguientes:

A, B	Preceden a C, D, E
C	Precede a F, G, H
D	Precede a G, H
E	Precede a H
F, G	Precede a I

a) Dibujar la red del proyecto, determinar la duración media mínima del proyecto, el camino crítico y las holguras de las actividades suponiendo duraciones medias y distribuciones beta.

b) ¿Qué duración debería decirse para tener una probabilidad de 0.8 de acabar el proyecto a tiempo?

PROBLEMA 7

Un proyecto de investigación consta de una serie de actividades, que se recogen en la tabla siguiente, junto con sus relaciones y sus duraciones en meses.

Actividad	Descripción	Antecesoras	Duración
A	Asignación de tareas	–	1

B	Búsqueda bibliográfica	A	6
C	Diseño de experimentos	A	5
D	Programación de software	A	5
E	Análisis de resultados anteriores	B	3
F	Realización de experimentos	B, C	1
G	Análisis de resultados	F	1
H	Simulación informática	B, C, D	4
I	Validación de resultados	E, G	1
J	Redacción del informe final	H, I	2

¿Cuál será la fecha más temprana en que se podrá entregar el informe final? Si en el momento de comenzar alguna de las tareas, el investigador que la debe realizar causa baja, ¿cuánto tiempo se podrá retrasar cada actividad, para buscar otro investigador, sin que esto afecte a la fecha de entrega del proyecto? ¿Sería necesario modificar en algún caso la planificación del resto de las actividades?

PROBLEMA 8

Una red de actividades tiene las siguientes características

Actividad	Precede directamente a	Duración
A	D, E, F	5
B	E, F	7
C	E, F, G	9
D	H, K	10
E	H, K	4
F	I, L	5
G	I, L	N(7,2)
H	J	6
I	J	2
J		4
K		N(11,3)

L		8

Como se ve la actividad G tiene una duración aleatoria normal de media 7 y desviación típica 2, en tanto que la de K, también normal, tiene media 11 y desviación típica 3.

a. Proponer un modelo de programación lineal entera que minimice el tiempo medio de ejecución del proyecto, suponiendo que las actividades F y G utilizan un mismo recurso y, por lo tanto, no pueden ejecutarse simultáneamente.
b. Suponiendo ahora que no existe la limitación del recurso único y utilizando el método PERT, determinar el camino crítico para la duración media.
c. Bajo las hipótesis del método PERT, ¿qué duración habría que dar para tener una probabilidad del 97.5 % de acabar a tiempo?
d. Suponiendo independencia de las duraciones, pero nada más ¿qué duración habría que dar para tener esa probabilidad del 97’5% de acabar a tiempo? Compara también los resultados del PERT y el caso exacto si K tuviera desviación típica 1, y en el caso en que fuera determinista (duración de K 11).

PROBLEMA 9

En la siguiente tabla se presentan cinco actividades, sus precedencias y duraciones.

Actividad	Precede a	Duración [días]
A	D	10
B	C y E	Uniforme [4,8]
C	D	5
D	–	Triangular [4,6,8]
E	–	5

Los tres parámetros de la distribución triangular son el mínimo, la moda y el máximo.

Se pide:

1. Dibujar el grafo de la red con las actividades y sus duraciones.

2. Suponiendo duraciones medias de las actividades determinar la duración mínima del proyecto, los márgenes (holguras) libres y totales.
3. Describir un procedimiento que permita simular la duración del proyecto teniendo en cuenta las duraciones aleatorias de algunas actividades. Realizar la simulación utilizando estas series de números aleatorios:

   Serie 1: 0.34, 0.56, 0.18, 0.86, 0.05

   Serie 2: 0.53, 0.07, 0.62, 0.59, 0.32
4. Determinar la probabilidad de que el camino crítico obtenido para las duraciones medias sea el camino crítico en la ejecución del proyecto.
5. Suponer ahora que D tiene distribución Uniforme en [4,8]. Dar la distribución exacta de la duración del proyecto, y comparar los resultados de la probabilidad de que el proyecto se extienda más de 17, 18, 19, 20 y 21 días según el método PERT y según la distribución exacta.
6. Dada la siguiente formulación de programación lineal para el problema con duraciones medias, y la correspondiente tabla óptima, responder a las siguientes cuestiones.

$$
\begin{aligned}
&\min TOT \\
&T_D - T_A \geq 10 \\
&T_C - T_B \geq 6 \\
&T_E - T_B \geq 6 \\
&T_D - T_C \geq 5 \\
&TOT - T_D \geq 6 \\
&TOT - T_E \geq 5 \\
&T_i, TOT \geq 0
\end{aligned}
$$

	T_A	T_B	T_C	T_D	T_E	TOT	S_1	S_2	S_3	S_4	S_5	S_6	
$c_j - z_j$	0	1	0	0	0	0	0	1	0	1	1	0	−17
T_D	0	−1	0	1	0	0	0	−1	0	−1	0	0	11
T_C	0	−1	1	0	0	0	0	−1	0	0	0	0	6
T_E	0	−1	0	0	1	0	0	0	−1	0	0	0	6
S_1	1	−1	0	0	0	0	1	−1	0	−1	0	0	1
TOT	0	−1	0	0	0	1	0	−1	0	−1	−1	0	17
S_6	0	0	0	0	0	0	0	−1	1	−1	−1	1	6

- Interpretar los resultados, incluyendo el significado de todas las variables de la tabla, el valor y significado de las duales, y los costes reducidos de las originales
- ¿Existen soluciones alternativas? Justificar, y, en su caso, dar esas soluciones

3.9 Resultados de la biblioteca de problemas

RESULTADO DEL PROBLEMA 1

1. Rutas críticas hay dos: A-B-E-F-G y A-B-C-G. Duración = 26. La única actividad que no es crítica es D, cuyas holguras total y libre coinciden, siendo 8.

2.

$$
\begin{aligned}
&\min 3000r_A + 1500r_B + 2000r_C + 4000r_D + 2000r_E + 3000r_F + 4000r_G \\
&x_B \geq x_A + 5 - r_A \\
&x_C \geq x_B + 8 - r_B \\
&x_D \geq x_B + 8 - r_B \\
&x_E \geq x_B + 8 - r_B \\
&x_F \geq x_E + 4 - r_E \\
&x_G \geq x_C + 10 - r_C \\
&x_G \geq x_F + 6 - r_F \\
&x_G + 3 - r_G = x_A + 20 \\
&0 \leq r_A \leq 2; 0 \leq r_B \leq 3; 0 \leq r_C \leq 1; 0 \leq r_D \leq 2; 0 \leq r_E \leq 2; 0 \leq r_F \leq 3; 0 \leq r_G \leq 1 \\
&x_i \geq 0 \quad \forall i
\end{aligned}
$$

Solución: reducir la duración de A en 2, la de B en 3, la de C en 1 y la de E en 1. Siendo la programación: A empieza en 0, B en 3, C en 8, D en 8, E en 8, F en 11 y G en 17. Se logra con un coste de 14.500.

RESULTADO DEL PROBLEMA 2

1.

2. Ruta crítica: A-C-G, con duración esperada de 14 días. Su varianza es de 2.3333.

3. Sabiendo que Z(0.01)=2.32, es decir, el valor de una distribución normal de media 0 y desviación 1 que deja a la derecha una probabilidad de 0.01 es 2.32, se deduce que debe empezar 17.54 días antes, redondeando, debe comenzar 18 días antes.

RESULTADO DEL PROBLEMA 3

2. Para 21.8.

3. A-FF=TF=0; B-FF=TF=1; C-FF=0, TF=2; D-FF=1, TF=4; E-FF=0, TF=3; F-FF=TF=0; G-FF=1, TF=2; H-FF=TF=3; I-FF=TF=0; J-FF=TF=1.

RESULTADO DEL PROBLEMA 4

1. Duración = 38. Holguras: A-FF=TF=0; B-FF=TF=9; C-FF=TF=0;

   D-FF=TF=0; E-FF=TF=0; F-FF=TF=2.

2.

$$
\begin{aligned}
\min \quad & 3X_{A1}+3X_{B1}+2X_{C1}+3X_{D1}+4X_{E1}+4X_{F1}+ \\
& +4X_{A2}+4X_{B2}+3X_{C2}+4X_{D2}+5X_{E2}+5X_{F2}+ \\
& +6X_{A3}+5X_{B3}+5X_{C3}+6X_{D3}+7X_{E3}+6X_{F3} \\
& \sum_{i=1}^{3} X_{ji}=1 \quad \forall j\in\{A,B,C,D,E,F\} \\
& T_B \geq T_A+9X_{A1}+7X_{A2}+5X_{A3} \\
& T_C \geq T_A+9X_{A1}+7X_{A2}+5X_{A3} \\
& T_D \geq T_C+7X_{C1}+5X_{C2}+3X_{C3} \\
& T_E \geq T_B+8X_{B1}+6X_{B2}+4X_{B3} \\
& T_E \geq T_D+10X_{D1}+8X_{D2}+6X_{D3} \\
& T_F \geq T_D+10X_{D1}+8X_{D2}+6X_{D3} \\
& T_E+12X_{E1}+10X_{E2}+7X_{E3} \leq 25 \\
& T_F+10X_{F1}+8X_{F2}+6X_{F3} \leq 25 \\
& X_{A3} \leq X_{F3} \\
& X_{ji}\in\{0,1\}, T_j \geq 0 \quad \forall i=1,2,3 \quad \forall j\in\{A,B,C,D,E,F\}
\end{aligned}
$$

RESULTADO DEL PROBLEMA 5

Hay dos caminos críticos: A-D-I-L-N-S-T y A-D-I-J-N-S-T. La esperanza de la duración con ambos es 38.8333 y la varianza en un caso es 4 + 25/36 y en el otro 4 + 13/36.

RESULTADO DEL PROBLEMA 6

1. El camino crítico es A-D-H, con una duración media de 20 y una varianza de 5.888. Las holguras son: B(FF=0 ó 1, TF= 1), C(0,1), E(1,1), F(1,2), G(0,1), I(1,1).
2. La duración sería: 20 + 0.84 $\sqrt{5.8888}$ = 22.04.

RESULTADO DEL PROBLEMA 8

b. Variables:

CP : duración media del proyecto

T_i : instante de inicio de la actividad i

$$X_{FG} = \begin{cases} 1 & \text{si F se hace antes que G} \\ 0 & \text{en otro caso (G antes que F)} \end{cases}$$

Datos:

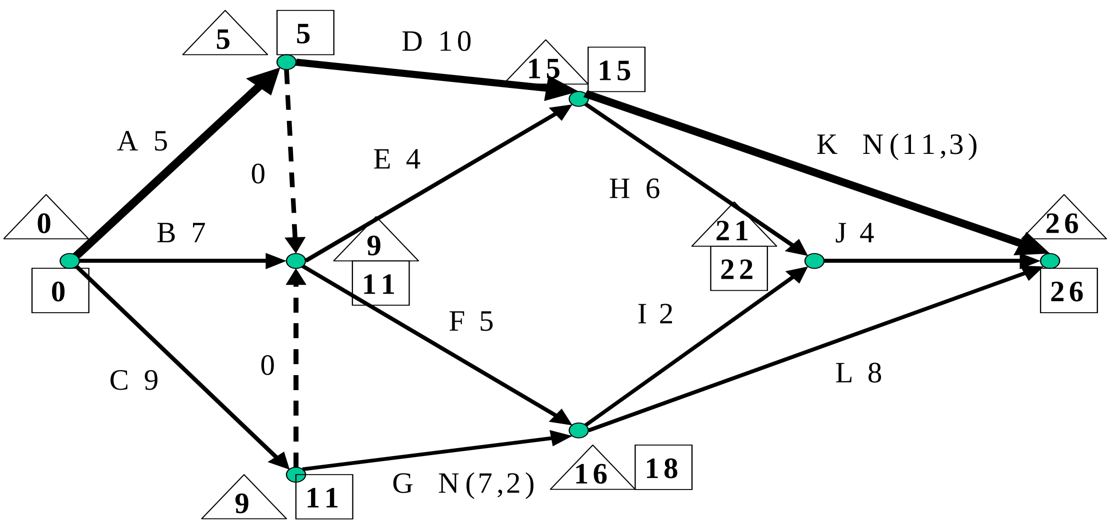
d_i : duración de la actividad i

Precedencias: $(i,i') \in P$

Modelo:

$$\min CP$$
$$T_{i'} \geq T_i + d_i \quad \forall (i,i') \in P$$
$$CP \geq T_i + d_i \quad \forall i \quad o \quad i = J, K, L$$
$$T_F \geq T_G + d_G - mX_{FG}$$
$$T_G \geq T_F + d_F - m(1 - X_{FG})$$
$$CP, T_i \geq 0, \; X_{FG} \in \{0,1\}$$

b.



c. El camino crítico está formado por las actividades *A*, *D* y *K*.

$$E[CP] = E[D_A] + E[D_D] + E[D_K] = 5 + 10 + 11 = 26$$

$$V[CP] = \sigma_A^2 + \sigma_D^2 + \sigma_K^2 = 0 + 0 + 9 = 9$$

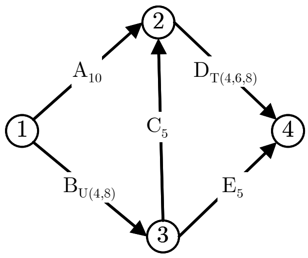
$$CP \overset{d}{=} N(26,3) \quad 0.025 \geq P(CP \geq f) = P(Z \geq \frac{f-26}{3})$$

$$\frac{f-26}{3} = 1.96 \quad f = 26 + 3 \cdot 1.96 \approx 32$$

d.

RESULTADO DEL PROBLEMA 9

1.



2. Las duraciones medias son

Actividad	Duración media [días]
A	10
B	6
C	5
D	6
E	5

Nodo	Instante más temprano t_i	Instante más tardío T_i
1	$t_1 = 0$	$T_1 = \min\{T_2 - d_A, T_3 - d_B\} =$ $\max\{11-10, 6-6\} = 0$

3	$t_3 = t_1 + d_B = 0 + 6 = 6$	$T_3 = \min\{T_4 - d_E, T_2 - d_C\} = \max\{17-5, 11-5\} = 6$
2	$t_2 = \max\{t_1 + d_A, t_3 + d_C\} = \max\{0+10, 6+5\} = 11$	$T_2 = T_4 - d_D = 17 - 6 = 11$
4	$t_2 = \max\{t_3 + d_E, t_2 + d_D\} = \max\{6+5, 11+6\} = 17$	$T_4 = 17$

Actividad	Holguras libre y total [días]
A	1
B	0
C	0
D	0
E	6

La duración mínima de proyecto es de 17 días. El camino crítico es B-C-D.

3. Para hacer la simulación de la duración real del proyecto hay que tomar muestras de las duraciones aleatorias. Las muestras de la distribución uniforme se pueden obtener mediante la transformada inversa. Las muestras de la triangular se pueden obtener mediante la transformada inversa o mediante aceptación-rechazo simple.

   Con la primera serie se calculan las muestras de la duración de B $d_B = (8-4)u + 4$. Con la segunda las de la duración de D. Las funciones de densidad y distribución son

$$f(d_D) = \begin{cases} 0.25d_D - 1 & 4 \le d_D \le 6 \\ -0.25d_D + 2 & 6 \le d_D \le 8 \end{cases}$$

$$F(d_D) = \begin{cases} 0.125d_D^2 - d_D + 2 & 4 \le d_D \le 6 \\ -0.125d_D^2 + 2d_D - 7 & 6 \le d_D \le 8 \end{cases}$$

   Por el método de la transformada inversa (TI)

$$\begin{cases} 0.125d_D^2 - d_D + 2 = u & u \le 0.5 \\ -0.125d_D^2 + 2d_D - 7 = u & u \ge 0.5 \end{cases}$$

$$\begin{cases} d_D = \dfrac{1 \pm \sqrt{1 - 4 \cdot 0.125(2-u)}}{0.25} & u \le 0.5 \\ d_D = \dfrac{-2 \pm \sqrt{4 - 4 \cdot 0.125(7+u)}}{-0.25} & u \ge 0.5 \end{cases}$$

Por el método de aceptación rechazo simple (AR). Con el primer número aleatorio obtengo una muestra de la duración $d_D = (8-4)u_1 + 4 = 4 \cdot 0.53 + 4 = 6.12$. Con el segundo una muestra de la altura $y = 0.5 \cdot 0.07 = 0.035$. Como $0.035 = y \le f(d_D) = -0.25 \cdot 6.12 + 2 = 0.47$ acepto la muestra. Otra muestra $d_D = (8-4)u_1 + 4 = 4 \cdot 0.62 + 4 = 6.48$, valor de la ordenada $y = 0.5 \cdot 0.59 = 0.295$, Como $0.295 = y \le f(d_D) = -0.25 \cdot 6.58 + 2 = 0.38$ acepto la muestra.

Núms.	Duración B [días]	Núms.	Duración D [días] (TI)	Duración D [días] (AR)
0.34	5.36	0.53	6.06	6.12
0.56	6.24	0.07	4.75	-
0.18	4.72	0.62	6.26	6.48
0.86	7.44	0.59	6.19	-
0.05	4.2	0.32	5.6	

4. Para que el camino crítico calculado para duraciones medias siga siendo crítico es necesario que la duración total de *B* más *C* sea superior a la de *A*. Esta probabilidad viene dada por la condición de que

$$P(d_B + d_C > d_A) = P(d_B > d_A - d_C) = P(d_B > 5) =$$
$$= 1 - P(d_B \le 5) = 1 - F(5) = 1 - 0.25 \cdot 5 + 1 = 0.75$$

5.

6. Valores variables originales

- T_i : A y B empiezan en 0, C en 6, D en 11 y E en 6.
- TOT : La duración del proyecto es 17 días.

Función objetivo: 17 días de proyecto

Variables de holgura:

- S1=1 D se demora 1 día en empezar desde que acaba A (puesto que A no precede a nadie más, se puede ver como holgura de A)
- S6= 6 desde que acaba E hasta el final del proyecto hay 6 días (holgura de E que no precede a nadie)
- Resto holguras es cero, todo empieza cuando acaba lo precedente y D determina el final del proyecto.

Variables duales:

- Y1= 0: Si A aumenta su duración no afecta al proyecto (no crítica)
- Y2= 1: Si B aumenta su duración en 1 se retrasa el proyecto en 1, al retrasar a C (B crítica)
- Y3= 0: Si B aumenta su duración en 1 no afecta al proyecto por retrasar a E (sí por C)
- Y4= 1: Si C aumenta su duración en 1 se retrasa el proyecto (C crítica) y D también lo será.
- Y5= 1: Si D aumenta su duración en 1 se retrasa el proyecto en 1 (crítica)
- Y6= 0: Si E aumenta su duración no afecta al proyecto

CoStes reducidos de no básicas:

- De A: 0, aumentar el instante de inicio de A no afecta al proyecto (no crítica)
- De B: 1, aumentar en 1 el instante de inicio de B retrasa el proyecto en 1 (crítica)

Otras soluciones básicas:

	T_A	T_B	T_C	T_D	T_E	TOT	S_1	S_2	S_3	S_4	S_5	S_6	
$c_j - z_j$	0	1	0	0	0	0	0	1	0	1	1	0	-17
T_D	0	-1	0	1	0	0	0	-1	0	-1	0	0	11
T_C	0	-1	1	0	0	0	0	-1	0	0	0	0	6
T_E	0	-1	0	0	1	0	0	0	-1	0	0	0	6
T_A	1	-1	0	0	0	0	1	-1	0	-1	0	0	1
TOT	0	-1	0	0	0	1	0	-1	0	-1	-1	0	17
S_6	0	0	0	0	0	0	0	-1	1	-1	-1	1	6

	T_A	T_B	T_C	T_D	T_E	TOT	S_1	S_2	S_3	S_4	S_5	S_6	
$c_j - z_j$	0	1	0	0	0	0	0	1	0	1	1	0	-17
T_D	0	-1	0	1	0	0	0	-1	0	-1	0	0	11
T_C	0	-1	1	0	0	0	0	-1	0	0	0	0	6
T_E	0	-1	0	0	1	0	0	-1	0	0	0	0	12
S_1	1	-1	0	0	0	0	1	-1	0	-1	0	0	1
TOT	0	-1	0	0	0	1	0	-1	0	-1	-1	0	17
S_3	0	0	0	0	0	0	0	-1	1	-1	-1	1	6

	T_A	T_B	T_C	T_D	T_E	TOT	S_1	S_2	S_3	S_4	S_5	S_6	
$c_j - z_j$	0	1	0	0	0	0	0	1	0	1	1	0	-17
T_D	0	-1	0	1	0	0	0	-1	0	-1	0	0	11
T_C	0	-1	1	0	0	0	0	-1	0	0	0	0	6
T_E	0	-1	0	0	1	0	0	-1	0	0	0	0	12
T_A	1	-1	0	0	0	0	1	-1	0	-1	0	0	1
TOT	0	-1	0	0	0	1	0	-1	0	-1	-1	0	17
S_3	0	0	0	0	0	0	0	-1	1	-1	-1	1	6

Todas las posibles soluciones son:

$$\begin{pmatrix} T_A \\ T_B \\ T_C \\ T_D \\ T_E \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \\ 6 \\ 11 \\ 6 \end{pmatrix} + \lambda_2 \begin{pmatrix} 1 \\ 0 \\ 6 \\ 11 \\ 6 \end{pmatrix} + \lambda_3 \begin{pmatrix} 0 \\ 0 \\ 6 \\ 11 \\ 12 \end{pmatrix} + \lambda_4 \begin{pmatrix} 1 \\ 0 \\ 6 \\ 11 \\ 12 \end{pmatrix} = \begin{pmatrix} \lambda_2 + \lambda_4 \\ 0 \\ 6 \\ 11 \\ 6 + 6(\lambda_3 + \lambda_4) \end{pmatrix} = \begin{pmatrix} \mu \\ 0 \\ 6 \\ 11 \\ 6 + \theta \end{pmatrix}$$

$$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1 \quad \lambda_i \geq 0 \quad o \quad \mu, \theta \in [0,1]$$

4 Modelos de secuenciación en máquinas

En un negocio o industria es habitual que se establezcan tareas que hay que desarrollar existiendo un determinado orden total o parcial para llevarlas a cabo. En esta sección se van a tratar problemas en los que hay unas tareas que llevar a cabo con determinados recursos limitados. La terminología utilizada proviene del ámbito de la producción denominándose trabajos (*jobs*) a las tareas y máquinas a los recursos. Aunque el contexto en que se utilicen no sea productivo, en general, es fácil hacer la traslación a este lenguaje.

El planteamiento del problema que se va a ver recibe el nombre de problema *general job-shop*, no se hace la traducción de este término al castellano por no existir un claro convenio al respecto. En este planteamiento se supone que existen n *trabajos* $\{J_1,...,J_n\}$ que han de ser procesados por m *máquinas* $\{M_1,...,M_m\}$ [10]. Se supone que cada trabajo ha de pasar por cada máquina una y sólo una vez, denominándose *operación* al proceso de un trabajo en una máquina y denotándose por o_{ij} la operación de procesar el trabajo i en la máquina j.

Se denominan *restricciones tecnológicas* a condiciones que puedan ser puestas en el orden en que los trabajos deben ser procesados en las máquinas. En el caso general, se supone que cada trabajo tiene su propio orden de proceso sin que exista relación con el orden de cualquier otro trabajo. Sin embargo, existe un caso particular de gran importancia en el que el orden es el mismo para todos los trabajos, denominándose a este problema *flow-shop* ya que es como si los trabajos fluyeran entre las máquinas en el mismo orden.

Cada operación o_{ij} requiere un cierto tiempo p_{ij} para ser desarrollada, denominado *tiempo de proceso.* Por convenio, en este tiempo se incluye cualquier tiempo requerido para ajustar la máquina para este trabajo o tiempo de transporte hasta la máquina. Este tiempo se supone fijo y conocido con antelación. Igualmente, todos los datos se suponen deterministas y conocidos por el planificador.

---

[10] Algunos autores se refieren a las máquinas como procesadores, especialmente cuando se trata este tipo de problemas en planificación de trabajos informáticos.

También se supone que las máquinas están siempre disponibles, aunque no se supondrá lo mismo para los trabajos, es decir, se considerará que los trabajos pueden no estar disponibles para ser procesados al principio del periodo de planificación. Se denominará *release date* o *ready time* y se denotará por r_i al instante en el que el trabajo J_i puede empezar a ser procesado o el pedido puede empezar a ser preparado.

El problema general *job-shop* consiste en encontrar una secuencia en la que los trabajos pasan por las máquinas para ser procesados, tal que

a) Asumiendo ciertas hipótesis, sea una planificación compatible con las restricciones tecnológicas, es decir, sea factible, y

b) Sea óptima respecto a algún criterio de desarrollo.

Las hipótesis, que supone el problema *job-shop* y que se verán a continuación, hacen que no todos los problemas de programación y secuenciación puedan ser modelados como un *job-shop* ni resueltos con las técnicas desarrolladas para este tipo de problemas. Sin embargo, son una buena introducción a los conceptos manejados en la teoría de secuenciación y a las técnicas utilizadas con un desarrollo coherente, de modo que además de su valor en sí mismo, permiten seguir con facilidad las teorías desarrolladas en otros contextos.

Empezaremos comentando cuáles son las hipótesis del *job-shop* y después se comentarán las distintas medidas de desarrollo y los posibles objetivos asociados a éstas que se pueden plantear en un problema de estas características, así como las relaciones entre ellas.

4.1 Hipótesis del *job-shop*

1. Cada trabajo es una entidad y, por lo tanto, no pueden procesarse dos operaciones de un mismo trabajo simultáneamente.
2. No existe interrupción, es decir, cada operación una vez empezada debe ser completada antes de que otra operación pueda empezar en esa máquina.
3. Cada trabajo incluye una y sólo una operación en cada máquina, por lo que todos los trabajos incluyen exactamente m operaciones.

4. No se permite cancelación, es decir, no puede cancelarse una operación ya iniciada.
5. Los tiempos de proceso son independientes de la secuencia seguida, lo que excluye tiempos de ajuste en las máquinas diferentes según la secuencia de trabajos considerada o tiempos de transporte entre máquinas.
6. Se permite inventario intraproceso, es decir, los trabajos pueden esperar hasta que la máquina siguiente esté libre. No se considera por tanto que los trabajos deban ser continuos de operación en operación.
7. Hay sólo una máquina de cada tipo, no pudiendo elegir entre varias máquinas para desarrollar una operación
8. Las máquinas pueden estar inactivas.
9. Las máquinas no pueden procesar más de una operación a la vez.
10. Las máquinas están disponibles durante todo el periodo de planificación.
11. Las restricciones tecnológicas son conocidas e inmutables.
12. No existe aleatoriedad, es decir, son conocidos y fijos todos los datos que intervienen: número de trabajos, número de máquinas, tiempos de proceso, etc.

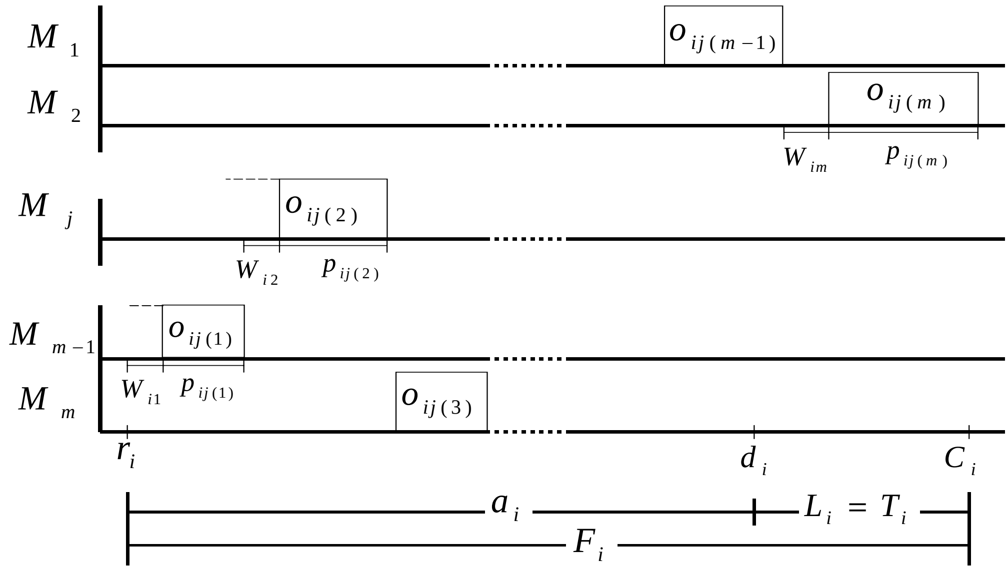
4.2 Medidas de desarrollo y objetivos

En general, no es fácil definir el objetivo en un problema de secuenciación y planificación ya que suelen ser numerosos y a menudo conflictivos. Ya en 1966 Mellor enumeró 27 metas u objetivos que pueden plantearse, a lo que se añade la dificultad de modelar matemáticamente estos objetivos para poder encontrar una solución que los optimice. A continuación se muestran algunas definiciones y notaciones sobre los datos (en minúsculas) y medidas (denotadas por mayúsculas) que se pueden hacer sobre la planificación antes de hablar de los objetivos que se pueden plantear sobre ellas:

- p_{ij} tiempo de proceso de la operación o_{ij}
- r_i instante en que el trabajo J_i puede iniciar su proceso o que el pedido puede empezar a ser preparado (*release date* o *ready time*)
- d_i instante en que el trabajo J_i debería ser terminado (*due date*) o fecha de entrega

- a_i amplitud del periodo de planificación del trabajo J_i o plazo de entrega; $a_i = d_i - r_i$
- W_{ik} tiempo de espera del trabajo J_i antes de realizarse su k-ésima operación (que no implica que sea en la máquina k)
- W_i tiempo total de espera del trabajo J_i; claramente, $W_i = \sum_{k=1}^{m} W_{ik}$
- C_i instante de finalización o cumplimentación de J_i; se verifica $C_i = r_i + \sum_{k=1}^{m} (W_{ik} + p_{ij(k)})$, donde $j(k)$ denota la máquina j donde se realiza la k-ésima operación en la que el trabajo J_i debe procesarse
- F_i tiempo de proceso o tiempo de cumplimentación o tiempo de suministro (*flow time*), es decir, el tiempo que pasa el trabajo J_i desde que empieza a ser procesado hasta que es acabado, luego, se cumple $F_i = C_i - r_i$
- L_i desviación de J_i respecto a su instante límite de acabado, es decir, $L_i = C_i - d_i$. Si el trabajo acaba antes de la fecha límite este valor será negativo, mientras que si se retrasa, será positivo. Es habitual que se planteen variables diferentes y no negativas para el retraso y para el adelanto, lo que se muestra a continuación.
- T_i demora en la entrega del trabajo J_i: $T_i = \max\{L_i, 0\}$
- E_i adelanto en la entrega del trabajo J_i: $E_i = \max\{-L_i, 0\}$

En el siguiente diagrama de Gantt se representan estas cantidades para un único trabajo y m máquinas donde el orden determinado por las restricciones tecnológicas es $(M_{m-1}, M_j, M_m, ..., M_1, M_2)$ y para una determinada programación:



Respecto a las medidas, sólo se ha hablado de las medidas de un trabajo, pero cuando se lleva a cabo una planificación hay que referirse a medidas incluyendo todos los trabajos. Habitualmente para agregar las medidas de todos los trabajos se recurre a utilizar la media de las medidas consideradas (por ejemplo, la media de los tiempos de proceso) o el máximo de éstas (por ejemplo, el máximo de los instantes de los trabajos). Se denotará por $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ a la media, así por ejemplo la media de los tiempos de proceso será $\overline{F} = \frac{1}{n}\sum_{i=1}^{n} F_i$ y por $X_{\max} = \max\{X_1, ..., X_n\}$ al máximo, de modo que el máximo de los instantes de finalización se denotará y calculará como $C_{\max} = \max\{C_1, ..., C_n\}$.

Por último, se define el tiempo inactivo (*idle*) de la máquina j como $I_j = C_{\max} - \sum_{i=1}^{n} p_{ij}$.

Veamos ahora los criterios más utilizados para dirigir la búsqueda de una planificación. Estos criterios los vamos a agrupar según a qué tipo de medidas afecten.

4.2.1 Criterios basados en los instantes de finalización

Los principales criterios en esta categoría son $F_{\max}$, $C_{\max}$, $\overline{F}$ y $\overline{C}$. Minimizar el primero supone minimizar el máximo tiempo de proceso, que esencialmente supone que el coste de la programación está relacionado directamente con su trabajo más largo. Minimizar $C_{\max}$, el máximo instante de finalización, supone que el coste depende del tiempo total dedicado a la producción completa de todos los artículos; a este valor se le llama *tiempo total de producción*, utilizándose también la palabra anglosajona *make-span* para referirse a este valor. Obsérvese que ambos valores son iguales si los instantes en que pueden empezarse los trabajos (*release dates*) son cero. Minimizar $\overline{F}$ implica que el coste se considera directamente relacionado con el tiempo medio de procesar un trabajo. Se puede comprobar que es equivalente minimizar $\overline{F}$ a minimizar $\overline{C}$, lo cual no es cierto para $F_{\max}$ y $C_{\max}$, debido a que las propiedades del máximo y de la media son diferentes.

Algunos autores utilizan las medidas medias pero ponderadas, de modo que los trabajos llevan distinto peso según una importancia relativa dentro del proceso completo.

4.2.2 Criterios basados en las fechas de entrega

Es muy habitual que el coste de una programación de trabajos esté relacionado con lo que se desvía del objetivo de entregar en plazo los trabajos y, por lo tanto, las medidas que se utilizan son $\overline{L}$, $L_{\max}$, $\overline{T}$ y T_{max}. Minimizar los dos primeros es apropiado cuando existe una recompensa por entregar un trabajo antes de tiempo, mientras que si ésta no existe sino que sólo existe penalización en caso de demora, lo apropiado es utilizar alguna de las últimas.

Hay casos en los que la penalización en que se incurre por terminar un trabajo fuera de plazo no es proporcional a la cantidad, de modo que es lo mismo entregarlo un minuto tarde que un siglo. Una situación de este tipo puede darse por ejemplo si un vuelo es programado para aterrizar después del tiempo máximo para el que dispone de combustible, las consecuencias son igual de catastróficas tanto si es un minuto o una hora lo que se pasa de ese instante. Por lo tanto, en ocasiones más que plantear un objetivo que sea función de la cantidad en que se sobrepasa la fecha de entrega, se busca minimizar el *número de trabajos fuera de plazo*, siendo entonces preferible, por ejemplo, retrasar un trabajo mucho y mantener los demás en plazo que retrasar todos un poquito y que se salgan muchos de plazo.

4.2.3 Criterios basados en el nivel de inventario y el coste de utilización

En este caso lo que habitualmente se desea minimizar es el número medio de trabajos en espera o el número medio de trabajos sin terminar. Sin embargo, cuando se habla de número medio es la media en el tiempo, es decir, en el periodo de planificación, valores que se calculan a partir de las variables $N_W(t)$ número de trabajos esperando entre máquinas en el instante t y $N_U(t)$ número de trabajos no acabados o en proceso en el instante t. El cálculo de las medias en tal caso será $\overline{N}_W(t) = \frac{1}{C_{\max}} \int_0^{C_{\max}} N_W(t)\, dt$ y $\overline{N}_U(t) = \frac{1}{C_{\max}} \int_0^{C_{\max}} N_U(t)\, dt$, respectivamente. Ambas medidas están directamente relacionadas con el coste de inventario inherente al proceso.

Otras medidas que se utilizan en este contexto son minimizar el número medio de trabajos acabados con el fin de reducir el coste de inventario de productos acabados o maximizar el número medio de trabajos que realmente están siendo procesados con el fin de maximizar el uso eficiente de las máquinas. En este último sentido y mirando más desde el punto de vista de las máquinas, es posible plantear como objetivos minimizar la media o el máximo tiempo inactivo de las máquinas.

4.2.4 Relaciones entre las medidas de desarrollo

Entre todas las medidas que se han nombrado anteriormente existen relaciones por las cuales unas son equivalentes a otras, es decir, que las secuencias óptimas son las mismas si se plantea un objetivo o se plantea otro equivalente. Estas equivalencias permiten reformular un problema pudiendo cambiar un objetivo por otro más sencillo o cuya formulación matemática tenga mejores propiedades a la hora de resolverlo. A continuación, se recogen algunas relaciones entre las medidas consideradas:

a) Son equivalentes las medidas medias $\overline{C}$, $\overline{F}$, $\overline{W}$ y $\overline{L}$.

b) No son equivalentes las medidas análogas a las anteriores pero para el máximo, excepto en casos particulares. Dos de estos casos son:

   b.1) Si los instantes de posible inicio (*release dates*) son 0 para todos los trabajos, son equivalentes $C_{\max}$ y $F_{\max}$.

b.2) Si todos los trabajos tienen una misma fecha de entrega son equivalentes $C_{\max}$ y $L_{\max}$.

c) Una programación que es óptima según el criterio $L_{\max}$ también lo es para $T_{\max}$, sin embargo al revés no tiene por qué ser cierto. En cualquier caso, nos permite resolver el problema con el criterio $L_{\max}$, cuya formulación y resolución son más sencillas que las del otro criterio.

d) Son equivalentes $C_{\max}$, $\overline{N}_p$ (el número medio de trabajos siendo realmente procesados, no en espera) e $\overline{I}$ (desocupación media de las máquinas, obtenida según la siguiente expresión $\overline{I} = \frac{1}{m}\sum_{j=1}^{m} I_j = \frac{1}{m}\sum_{j=1}^{m}(C_{\max} - \sum_{i=1}^{n} p_{ij})$).

e) Son equivalentes las medidas $\overline{N}_U$ y $\overline{C}/C_{\max}$.

f) Son equivalentes las medidas $\overline{N}_W$ y $\overline{W}/C_{\max}$.

g) Por último, para problemas de secuenciación en una sola máquina son equivalentes $\overline{C}$, $\overline{F}$, $\overline{W}$, $\overline{L}$, $\overline{N}_U$ y $\overline{N}_W$.

4.3 Problemas con una máquina

Los problemas más sencillos que se pueden plantear son los problemas que involucran a una única máquina. Estos problemas son más habituales de lo que se pueda pensar en un primer momento: desde la secuenciación de programas a ser procesados por un ordenador con un único procesador, a operaciones en una sección de un hospital en la que se dispone de un único quirófano o equipo quirúrgico o secciones dentro de un proceso productivo en que hay una única máquina para desarrollar una determinada tarea.

En lo que se refiere a estos problemas vamos a suponer que todas las *release dates* son cero, es decir, se supone para este tipo de problemas que $r_i = 0\ \forall i$ y una sola máquina, luego $m = 1$.

El primer resultado que se tiene para este tipo de problemas y fácil de inferir, es que para cualquiera de los objetivos que se plantee que sea regular, es decir, que sea no decreciente con los instantes de cumplimentación (la mayoría de los considerados), existe una planificación óptima en la que la máquina no está inactiva en ningún momento.

Otro resultado que se tiene para problemas con una máquina es que aunque se ha supuesto que no hay interrupción, esto no supone ninguna restricción ya que en estos problemas permitir la interrupción no puede mejorar ninguna planificación que sea óptima respecto a ninguno de los criterios regulares vistos.

Estos dos resultados unidos tienen como consecuencia que la solución buscada sea una *programación permutación* de los trabajos, es decir, el problema se reduce a buscar una permutación de los trabajos que minimice el objetivo planteado. Dado que la solución es una permutación, es evidente que para cualquier permutación $C_{\max}$ y $F_{\max}$ son los mismos, por lo que estos objetivos no serán considerados en problemas de una máquina con *release dates* cero en todos los trabajos.

A continuación, se presentan las permutaciones que minimizan algunos de los criterios vistos:

a) *Minimización del tiempo medio de proceso* $\overline{F}$. La permutación que resuelve este problema es la denominada *programación de tiempo de proceso creciente*, es decir, la que ordena los trabajos por su tiempo de proceso de menor a mayor. Dada la equivalencia entre objetivos, esta programación es también la óptima para los objetivos $\overline{C}, \overline{W}, \overline{L}, \overline{N}_U, \overline{N}_W$.

b) *Minimización de la máxima desviación respecto a sus fechas de entrega* $L_{\max}$. En este caso la permutación que proporciona el óptimo es la denominada *programación de fecha de entrega creciente*, es decir, aquélla que ordena los trabajos por su fecha de entrega en forma creciente. También ésta será la solución para el objetivo $T_{\max}$.

c) *Minimización del número de trabajos demorados*: para ello se plantea el siguiente algoritmo debido a Moore y Hodgson, del que se puede asegurar que encuentra la secuencia óptima:

1. Obtener la programación de fecha de entrega creciente.
2. Encontrar el primer trabajo que está demorado respecto a su fecha de entrega en la secuencia actual $J_{i(l)}$. Si no existe, ir al paso 4.

3. Encontrar el trabajo de mayor tiempo de proceso secuenciado antes de $J_{i(l)}$, incluído $J_{i(l)}$, y quitarlo de la secuencia. Volver al paso 2 con la secuencia actual (que tiene un trabajo menos que la anterior).
4. Formar una secuencia óptima con la secuencia que se tiene actual y añadiendo al final todos los trabajos que han sido rechazados en las iteraciones anteriores en cualquier orden.

   Esos trabajos rechazados serán los únicos demorados (se entregarán fuera de plazo).

Otro elemento que suele aparecer en problemas con una máquina es la existencia de *condiciones de precedencia* entre los trabajos, algunas veces debidas a la importancia relativa que puedan tener éstos. Es decir, es posible que se planteen condiciones de orden, de modo que un trabajo no pueda empezar mientras otro no ha sido completado. No deben ser confundidas estas condiciones con las restricciones tecnológicas; las restricciones tecnológicas imponen condiciones de orden a las operaciones de un trabajo, mientras que las condiciones de precedencia restringen la secuencia de operaciones de distintos trabajos.

Añadir estas condiciones, aunque reduce las permutaciones que son factibles, en general, supone una complicación en la formulación más que una ventaja, por ello se suele evitar, si es posible, reformulando las fechas de entrega de los trabajos más que añadiendo restricciones. Sin embargo no siempre es posible modelarlo haciendo tal reformulación, por lo que se han desarrollado algoritmos específicos para algunos de estos problemas. En esta introducción sólo vamos a ver un caso particular, para otros casos es posible plantear un problema de programación matemática formulando las correspondientes restricciones o buscar algoritmos específicos[11].

El único caso particular que vamos a considerar es el de las cadenas de productos. En este caso se supone que los trabajos están divididos formando una partición en K cadenas de $n_1,...,n_K$ trabajos respectivamente. Estas cadenas son secuencias de trabajos con un orden determinado y que han de ser procesados todos inmediatamente uno detrás

---

[11] Pueden verse la mayoría de ellos en French, S. (1982) *Sequencing and Scheduling. An Introduction to the Mathematics of the Job-Shop.* Ellis-Horwood.

de otro. Así pues, el problema es cómo secuenciar las cadenas de trabajos, quedando entonces determinada la secuencia completa de trabajos. En este caso se define el tiempo de proceso de una cadena como la suma de los trabajos que la forman (si p_{ij} es el tiempo de proceso del j-ésimo trabajo de la cadena i-ésima sería $p'_i = \sum_{j=1}^{n_i} p_{ij}$). La secuencia óptima de cadenas de trabajos para minimizar el tiempo medio de proceso será aquella en la que se secuencian las cadenas ordenándolas en forma creciente por el tiempo medio de proceso por trabajo, es decir, por el cociente del tiempo de proceso de la cadena y su tamaño (número de trabajos que incluye). Matemáticamente sería la secuencia tal que $\frac{p'_{i(1)}}{n_{i(1)}} \leq \frac{p'_{i(2)}}{n_{i(2)}} \leq \cdots \leq \frac{p'_{i(K)}}{n_{i(K)}}$.

Por último, comentar que para otros objetivos o precedencias, la programación entera y la programación dinámica se presentan como las herramientas más adecuadas. En particular la programación dinámica puede obtener programaciones óptimas con cálculos muy inferiores a lo que sería una enumeración exhaustiva. A continuación se muestra un ejemplo resuelto mediante programación dinámica con recursividad hacia atrás, aunque estos problemas también admiten de forma inmediata la recursividad hacia delante.

En este ejemplo se supone que hay cuatro trabajos cuyos tiempos de proceso y fechas de entrega se recogen a continuación. El objetivo es minimizar el retraso medio, es decir, minimizar $\frac{1}{4}\sum_{i=1}^{4} \max\left\{C_i - d_i, 0\right\}$, que es una función no lineal del tiempo de cumplimentación, lo que hace recomendable usar la programación dinámica más que la programación no lineal entera.

Trabajos	J_1	J_2	J_3	J_4
Tiempos proceso	8	6	10	7
Fecha entrega	14	9	16	16

Vamos a considerar como etapas las fases de agregar un trabajo más a la secuencia, así la etapa 4 será programar el cuarto trabajo, la etapa 3 será programar el tercer trabajo y así sucesivamente. En cuanto a las decisiones, en cada etapa hay que decidir qué trabajo hay que programar en esa posición y para saber cuáles son posibles y evaluar la función objetivo con el coste de esa etapa y el coste futuro, la información que es necesaria de etapas anteriores, los estados, son los trabajos que ya han sido programados. Obsérvese que no es necesario saber en qué orden, pero sí qué trabajos. A continuación se muestran las tablas de las diferentes etapas con recursividad hacia atrás, siendo el objetivo $\sum_{i=1}^{4} \max\{C_i - d_i, 0\}$, ya que se dividirá por 4 al final para facilitar los cálculos.

Etapa 4

Decisiones / Estados	J_1^4	J_2^4	J_3^4	J_4^4	Óptimo	Objetivo
J_1, J_2, J_3	–	–	–	31–16	J_4^4	15
J_1, J_2, J_4	–	–	31–16	–	J_3^4	15
J_1, J_3, J_4	–	31–9	–	–	J_2^4	22
J_2, J_3, J_4	31–14	–	–	–	J_1^4	17

Etapa 3

Decisiones / Estados	J_1^3	J_2^3	J_3^3	J_4^3	Óptimo	Objetivo
J_1, J_2	–	–	24–16+15	21–16+15	J_4^3	20
J_1, J_3	–	24–9+15	–	25–16+22	J_2^3	30
J_1, J_4	–	21–9+15	25–16+22	–	J_2^3	27
J_2, J_3	24–14+15	–	–	23–16+17	J_4^3	24
J_2, J_4	21–14+15	–	23–16+17	–	J_1^3	22
J_3, J_4	25–14+22	23–9+17	–	–	J_2^3	31

Etapa 2

Decisiones / Estados	J_1^2	J_2^2	J_3^2	J_4^2	Óptimo	Objetivo
J_1	–	14–9+20	18–16+30	0+27	J_2^2	25
J_2	14–14+20	–	16–16+24	0+22	J_1^2	20
J_3	18–14+30	16–9+24	–	17–16+31	J_2^2	31
J_4	15–14+27	13–9+22	17–16+31	–	J_2^2	26

Etapa 1

Decisiones	J_1^1	J_2^1	J_3^1	J_4^1	Óptimo	Objetivo
	0+25	0+20	0+31	0+26	J_2^1	20

Deshaciendo el camino hacia atrás, vemos que la secuencia óptima es $(J_2^1, J_1^2, J_4^3, J_3^4)$, con un retraso medio de 20/4=5 unidades de tiempo.

4.4 Problemas con varias máquinas

En esta sección vamos a ver primero algunos algoritmos constructivos para ciertos problemas con varias máquinas. Después veremos técnicas más generales aplicadas a otros problemas con estas características.

Para los algoritmos constructivos supondremos que todas las *release dates* son iguales, es decir, $r_i = 0,\ i = 1, ..., n$.

El primer caso que vamos a considerar es el problema *flow-shop* con dos máquinas y minimizando el máximo tiempo de cumplimentación, es decir, un problema con n trabajos, 2 máquinas, en el que todos los trabajos han de pasar primero por la máquina 1 y luego por la 2 y en que se desea minimizar el máximo tiempo de cumplimentación $C_{\max}$. Para este tipo de problemas, tanto para dos máquinas como para tres, se puede demostrar que podemos limitarnos a buscar una secuencia óptima entre las programaciones permutación, es decir, que existe una programación óptima en la que el orden en el que cada máquina procesa los trabajos es el mismo para todas las máquinas.

Para obtener la secuencia óptima se definen $a_i = p_{i1}$ (tiempo de proceso del trabajo J_i en la máquina 1) y $b_i = p_{i2}$ (tiempo de proceso del trabajo J_i en la máquina 2). El algoritmo, conocido como algoritmo de Johnson de tipo constructivo y que asegura que se alcanza el óptimo, es el siguiente:

1. Sea $k = 1$ y $l = n$.
2. Sea la lista actual de trabajos no programados = $\{J_1, J_2, ..., J_n\}$.
3. Encontrar el menor valor entre los tiempos a_i y b_i de los trabajos no programados.
4. Si ese menor valor se alcanza en un a_i, es decir, es el tiempo de proceso del trabajo J_i en la primera máquina, entonces hacer:

   4.1. Programar J_i en la k-ésima posición de la secuencia de procesos

   4.2. Borrar J_i de la lista actual de trabajos no procesados

   4.3. Incrementar k a $k+1$

   4.4. Ir al paso 6
5. Si ese menor valor se alcanza para un b_i, es decir, es el tiempo de proceso del trabajo J_i en la segunda máquina, entonces hacer:

   5.1. Programar J_i en la l-ésima posición de la secuencia de procesos

   5.2. Borrar J_i de la lista actual de trabajos no procesados

   5.3. Reducir l a $l-1$

   5.4. Ir al paso 6
6. Si hay trabajos todavía sin programar volver al paso 3. En otro caso, parar.

Este algoritmo puede ser extendido a un caso más general, en el que no sólo se relaje la condición de que todos los trabajos pasen en el mismo orden por las dos máquinas, sino, además se puede aplicar si hay trabajos que no tienen que pasar por ambas máquinas sino sólo por una de ellas. El objetivo es el mismo, minimizar el tiempo máximo de cumplimentación. Así el algoritmo de Johnson para dos máquinas extendido supone que existen cuatro tipos de trabajos: los trabajos de tipo A que son los que tienen que ser procesados sólo por la máquina M_1; los trabajos de tipo B que son los que únicamente han de pasar por la máquina M_2; los trabajos tipo C que han de pasar primero por la máquina M_1 y luego por la M_2; y los trabajos tipo D que han de pasar por ambas máquinas pero en el orden inverso a los anteriores.

En estas condiciones para construir una programación óptima de los trabajos se debe seguir el siguiente procedimiento:

1. Secuenciar los trabajos tipo A en cualquier orden para dar la secuencia S_A.
2. Secuenciar los trabajos tipo B en cualquier orden para dar la secuencia S_B.
3. Secuenciar los trabajos tipo C según el algoritmo de Johnson anterior, visto para el problema *flow-shop*, para dar la secuencia S_C.
4. Secuenciar los trabajos tipo D según el algoritmo de Johnson anterior, visto para el problema *flow-shop*, intercambiando los papeles de las máquinas, para dar la secuencia S_D.
5. Obtener la programación óptima:

Máquina	Orden de proceso
M_1	(S_C, S_A, S_D)
M_2	(S_D, S_B, S_C)

No vamos a entrar a ver más algoritmos constructivos, aunque existen algunos para problemas similares a los anteriores. Realmente para problemas que difieran sustancialmente de lo que se ha visto, la programación dinámica en algunos casos y la programación entera en la mayoría de ellos se presentan como la alternativa más razonable. A continuación se presenta un ejemplo sencillo por su tamaño, aunque no por la dificultad para modelarlo y una formulación para él mediante programación lineal entera mixta de tipo general que es fácilmente aplicable para casos más complejos.

Sea un problema con dos trabajos y 3 máquinas, cuyo objetivo es minimizar la máxima demora. Los tiempos de proceso, las fechas de entrega y las condiciones tecnológicas se muestran en la tabla siguiente.

	M_1	M_2	M_3	*Due dates*	Orden
J_1	10	15	18	50	M_1, M_2, M_3
J_2	20	12	15	55	M_3, M_2, M_1

Para este problema definimos inicialmente las siguientes variables de decisión:

T_{ij}: instante en que se inicia el proceso del trabajo J_i en la máquina M_j

A partir de ellas modelaremos el problema añadiendo las variables auxiliares que sean necesarias. Veamos las condiciones:

- Restricciones tecnológicas: se establece el orden en cada trabajo. La forma general sería: Si el trabajo J_i ha de ser procesado antes en la máquina M_j que en la $M_{j'}$, $T_{ij} + p_{ij} \leq T_{ij'}$. Así se tiene,

$$T_{11} + 10 \leq T_{12} \quad T_{12} + 15 \leq T_{13}$$
$$T_{23} + 15 \leq T_{22} \quad T_{22} + 12 \leq T_{21}$$

(Obsérvese que por transitividad sólo es necesario poner los órdenes inmediatos)

- No simultaneidad en una máquina: dados dos trabajos y una máquina o se procesa primero uno y luego el otro o viceversa. Es decir, en forma general, dados los trabajos J_i y J_k y la máquina M_j, se tiene la disyunción: $T_{ij} + p_{ij} \leq T_{kj}$ o $T_{kj} + p_{kj} \leq T_{ij}$. Esta disyunción se puede formular como conjunción mediante variables binarias auxiliares que representen

$$\delta_{ikj} = \begin{cases} 1 & \text{si } J_i \text{ se procesa después de } J_k \text{ en la máquina } M_j \\ 0 & \text{en caso contrario} \end{cases}$$

como

$$T_{kj} + p_{kj} - T_{ij} \le M(1-\delta_{ikj})$$
$$T_{ij} + p_{ij} - T_{kj} \le M\delta_{ikj}$$

siendo M una cota para el valor del lado izquierdo de las restricciones.

Así para nuestro caso particular las restricciones serán:

M_1	M_2	M_3
$T_{11}+10-T_{21} \le M(1-\delta_{211})$ $T_{21}+20-T_{11} \le M\delta_{211}$	$T_{12}+15-T_{22} \le M(1-\delta_{212})$ $T_{22}+12-T_{12} \le M\delta_{212}$	$T_{13}+18-T_{23} \le M(1-\delta_{213})$ $T_{23}+15-T_{13} \le M\delta_{213}$

- Función objetivo: minimizar la máxima demora. Para ello se introducen unas variables auxiliares que representen la demora en cada trabajo y otra que acote todas ellas y que será minimizada. Sea T_i la demora del trabajo J_i. Como se desean obtener estos valores y minimizar su máximo, para cada trabajo se toma la última operación de su proceso completo (determinada por las restricciones tecnológicas) y que denotaremos por $m(i)$ y entonces podemos formular las demoras como $T_{im(i)} + p_{im(i)} - T_i \le d_i$ y para acotar todas ellas, $T_i \le T_{\max}$, minimizando este último valor. Así para nuestro caso particular, tendríamos:

$$\min T_{\max}$$
$$T_1 \le T_{\max}$$
$$T_2 \le T_{\max}$$
$$T_{13}+18-T_1 \le 50$$
$$T_{21}+20-T_2 \le 55$$

- Carácter de las variables. Hay que definir el carácter no negativo de las variables y el binario de las auxiliares definidas:

$$T_{ij} \ge 0, \quad T_i \ge 0, \quad \delta_{ikj} \in \{0,1\}$$

Por último, comentar que todos los algoritmos y técnicas planteadas son de optimización, es decir, se garantiza que alcanzan la programación óptima según el criterio considerado. Sin embargo, dado que la formulación mediante programación entera o no lineal entera es la única alternativa en algunos de los casos y dada la dificultad para resolver ésta en cuanto el tamaño de un problema es elevado (lo cual puede ocurrir incluso con no muchas máquinas y trabajos) también existen algoritmos de tipo heurístico para resolverlos, aunque no se incluirán en este capítulo introductorio.

4.5 Biblioteca de problemas

PROBLEMA 1

Encontrar la forma óptima de secuenciar las siguientes 8 tareas en una máquina para minimizar el tiempo medio de cumplimentación.

Tarea	1	2	3	4	5	6	7	8
Tiempo de proceso	10	2	4	7	3	1	3	2

PROBLEMA 2

Encontrar la forma óptima de secuenciar las siguientes 7 tareas en una máquina para minimizar la desviación máxima respecto a la fecha de entrega y dar para esta secuencia la máxima desviación en que se incurre.

Tarea	1	2	3	4	5	6	7
Tiempo de proceso	29	13	31	20	7	3	9
Fecha de entrega	80	20	67	48	100	30	50

PROBLEMA 3

Encontrar la forma óptima de secuenciar las siguientes 10 tareas en una máquina para minimizar el número de trabajos demorados.

Tarea	1	2	3	4	5	6	7	8	9	10
Tiempo de proceso	5	3	1	2	4	4	2	1	1	4
Fecha de entrega	19	16	25	3	8	14	31	23	2	15

PROBLEMA 4

Supóngase que la ESA tiene una única lanzadera con la que situar en órbita estaciones espaciales. Se desea situar 8 estaciones, cada una de ellas específicamente diseñada para desarrollar ciertas observaciones astronómicas. Cada estación debe ser situada en posición y construida antes de cierta fecha o si no, será inútil. Dados los datos siguientes, ¿en qué orden deberían ser situadas las estaciones? Suponer que se empieza el 1 de enero de 2002.

Estación	Tiempo de montaje en lanzadera, lanzarlmiento y puesta en funcionamiento	Debe estar en órbita el 1º de
1	1 año y 2 meses	Abril 2006
2	7 meses	Enero 2003
3	11 meses	Agosto 2003
4	3 meses	Marzo 2006
5	1 año y 8 meses	Septiembre 2005
6	4 meses	Agosto 2002
7	7 meses	Diciembre 2002
8	1 año y 2 meses	Junio 2004

PROBLEMA 5

Minimizar el tiempo medio de cumplimentación de los 12 trabajos siguientes si hay dos cadenas de trabajos que hay que respetar (3, 6, 9, 12) y (1, 2, 4, 8, 10).

Trabajo	1	2	3	4	5	6	7	8	9	10	11	12
Tiempo de proceso	2	1	5	6	3	4	1	2	5	6	1	6

PROBLEMA 6

Minimizar el *make-span* para el siguiente problema *flow-shop* con 8 trabajos y 2 máquinas.

	Máquinas	
Trabajos	1	2
1	2	10
2	3	7
3	8	2
4	7	5
5	9	3
6	4	4
7	1	9
8	6	8

PROBLEMA 7

Encontrar una secuencia que minimice el tiempo máximo de cumplimentación para los siguientes 14 trabajos y dos máquinas. Representarlo en un diagrama de Gantt.

	Orden de proceso y tiempo			
Trabajo	Primera máquina		Segunda máquina	
1	1	8	2	7
2	2	12	1	14
3	1	13	2	9
4	1	6	–	–
5	1	7	2	13
6	2	9	1	8
7	2	12	–	–

8	1	4	2	10
9	1	10	2	6
10	2	10	1	11
11	2	6	–	–
12	2	8	–	–
13	1	5	2	13
14	1	7	–	–

PROBLEMA 8

Encontrar la secuencia que minimiza el retraso o demora media con que los trabajos son entregados, con los siguientes datos:

Tarea	1	2	3	4
Tiempo de proceso	9	12	7	14
Fecha de entrega	15	19	23	21

PROBLEMA 9

1. Formular un problema de programación matemática para encontrar la programación que minimiza el tiempo máximo de cumplimentación de los siguientes tres trabajos, si no hay restricciones tecnológicas, es decir, no hay un orden establecido entre las operaciones de cada trabajo. Los tiempos de proceso se dan a continuación

Trabajos	Máquina 1	Máquina 2	Máquina 3
1	6	8	3
2	9	5	2
3	4	8	17

2. Hacer lo mismo pero si además se supone que las máquinas han de procesar todos los trabajos en el mismo orden

PROBLEMA 10

Seis alcaldes de pueblo acuden juntos a la capital de la provincia para visitar las delegaciones de Industria y Hacienda, según la siguiente tabla.

Alcalde	1ª Delegación	2ª Delegación
Villamochuelos	Hacienda (60 min)	Industria (25 min)
Valdealpargatas	Industria (20 min)	Hacienda (40 min)
Cañaveras	Hacienda (70 min)	
Montaraces	Industria (25 min)	Hacienda (60 min)
Conejera de Arriba	Industria (30 min)	
Belloteros de Abajo	Hacienda (52 min)	Industria (30 min)

¿Cómo se organizan para volver cuanto antes a sus pueblos, sabiendo que los alcaldes no pueden ser atendidos simultáneamente en la misma Delegación y que han de volver juntos? ¿Cuánto tiempo pasarán en la capital? ¿Cuándo podrán los funcionarios de las respectivas delegaciones irse a comer por haber acabado de atenderles? Representar en un diagrama la secuencia sobre cada una de las delegaciones.

4.6 Resultados de la biblioteca de problemas

RESULTADO DEL PROBLEMA 1

La secuencia óptima es (6, 8, 2, 5, 7, 3, 4, 1) con un tiempo medio de cumplimentación de 97/8.

RESULTADO DEL PROBLEMA 2

La secuencia es (2, 6, 4, 7, 3, 1, 5), siendo la máxima desviación 25.

RESULTADO DEL PROBLEMA 3

Una secuencia óptima es (9, 4, 5, 6, 2, 1, 8, 3, 7, 10) en la que sólo hay un trabajo demorado.

RESULTADO DEL PROBLEMA 4

Una secuencia óptima es (6, 7, 8, 4, 1) con los trabajos 2, 3 y 5 cancelados.

RESULTADO DEL PROBLEMA 5

Una secuencia óptima es (7, 11, 5, 1, 2, 4, 8, 10, 3, 6, 9, 12)

RESULTADO DEL PROBLEMA 6

La secuencia óptima es (7, 1, 2, 6, 8, 4, 5, 3) con *make-span* 49.

RESULTADO DEL PROBLEMA 7

Una programación óptima es: en la máquina 1 (8, 13, 5, 3, 1, 9, 4, 14, 10, 2, 6) y en la máquina 2 (10, 2, 6, 7, 11, 12, 8, 13, 5, 3, 1, 9) con máximo tiempo de cumplimentación 115.

RESULTADO DEL PROBLEMA 8

Una secuencia es (1, 2, 3, 4) con retraso medio de 7 (Programación dinámica).

RESULTADO DEL PROBLEMA 9

El tiempo mínimo de cumplimentación es 29.

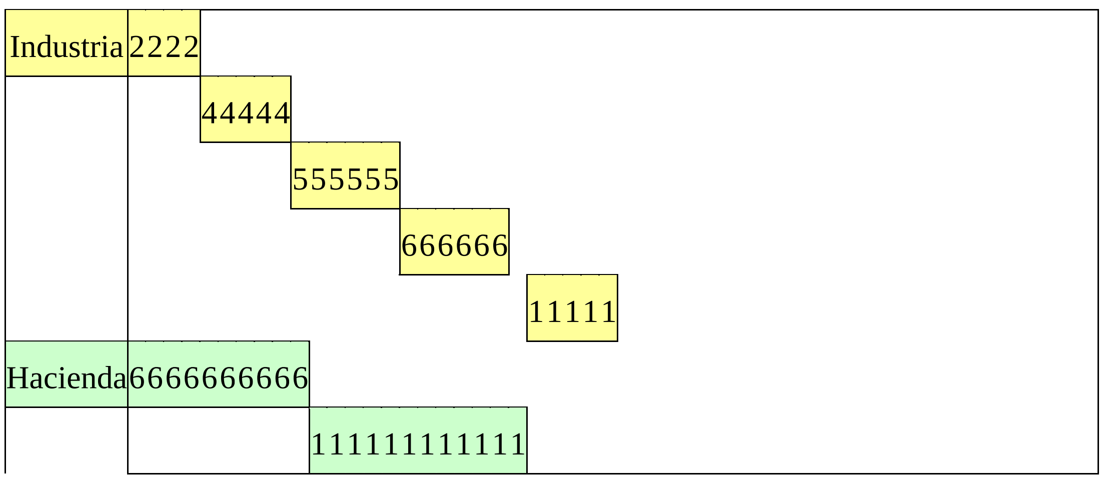
RESULTADO DEL PROBLEMA 10

Se trata de un problema de secuenciación de tareas en dos máquinas. Suponemos que la máquina 1 es la Delegación de Industria y la 2 la de Hacienda. Luego los trabajos se pueden clasificar en estos tipos

	Alcalde	Tipo	Máquina 1	Máquina 2
			Industria	Hacienda
1	Villamochuelos	D	25	60
2	Valdealpargatas	C	20	40
3	Cañaveras	B		70
4	Montaraces	C	25	60
5	Conejera de Arriba	A	30	
6	Belloteros de Abajo	D	30	52

Se deben secuenciar los trabajos de cada uno de los tipos independientemente. Sólo los tipos C y D tienen más de un trabajo luego requieren secuenciación. Los de tipo C se ordenan según el algoritmo de Johnson y los de tipo D por el mismo algoritmo pero intercambiando la posición de las máquinas.

Máquina 1	**Industria**	2, 4, 5, 6, 1	Valdealpargatas, Montaraces, Conejera, Belloteros, Villamochuelos
Máquina 2	**Hacienda**	6, 1, 3, 2, 4	Belloteros, Villamochuelos, Cañaveras, Valdealpargatas, Montaraces



		33333333333333		
			22222222	
				444444444444

El alcalde de Villamochuelos no puede ir a la Delegación de Industria hasta no haber acabado en la de Hacienda, a los 112 minutos. Los alcaldes pasan un total de 52+60+70+40+60=282 minutos en la capital. El funcionario de Hacienda está ocupado durante ese tiempo y el de Industria durante 52+60+25=137 minutos.

Hay soluciones que evitar el tiempo de espera de la delegación de Industria. Por ejemplo, ésta.

Máquina 1	**Industria**	2, 4, 5, 1, 6	Valdealpargatas, Montaraces, Conejera, Villamochuelos, Belloteros
Máquina 2	**Hacienda**	6, 3, 1, 2, 4	Belloteros, Cañaveras, Villamochuelos, Valdealpargatas, Montaraces