Introducción a la Optimización

Álvaro Baíllo Moreno

Contenido

- Concepto de optimización.
- Áreas de estudio:
  - Programación lineal.
  - Programación lineal entera.
  - Programación no lineal.
  - Programación dinámica.

Punto de partida: el sistema

- Tenemos la responsabilidad de **gestionar un sistema**:
  - Economía de un país.
  - Proceso de fabricación.
  - Negocio de transporte.
  - Equipo de personas.
- La situación del sistema se caracteriza mediante **variables de estado**:
  - Describen totalmente la situación del sistema en un instante t.
  - Economía de un país: Inflación, PIB...
  - Proceso de fabricación: Unidades de producto terminado, unidades disponibles de materia prima...
  - Negocio de transporte: Unidades disponibles en los almacenes de origen, unidades demandadas en cada destino...
  - Equipo de personas: Personas que trabajan en cada turno...

Problema: tomar decisiones

- Las decisiones se denominan **variables de control**:
  - Economía de un país: Tipos de interés, política fiscal...
  - Proceso de fabricación: Unidades de producto a fabricar...
  - Negocio de transporte: Unidades a transportar desde cada almacén de origen a cada destino...
  - Equipo de personas: Personas que entran a trabajar en cada hora...
- El sistema debe verificar ciertas **restricciones**:
  - Economía de un país: Inflación y déficit público limitados...
  - Proceso de fabricación: Límite del almacén de producto terminado...
  - Negocio de transporte: Número máximo de unidades en cada trayecto...
  - Equipo de personas: Número máximo de horas de trabajo seguidas...

Objetivo: identificar las decisiones óptimas

- Normalmente **existen múltiples soluciones** para las que el sistema verifica las restricciones:
  - Se denominan **soluciones factibles**.
- Es necesario establecer **criterios** que permitan ordenar las soluciones de mejor a peor.
  - Se puede definir una **función objetivo** que valore cuantitativamente la bondad de una cierta solución.

Optimización: casos particulares

- Múltiples funciones objetivo: **Optimización multicriterio**
  - No hay coincidencia del óptimo para los diferentes objetivos.
- **No existe función objetivo**
  - Sistema de ecuaciones (no) lineales.
  - Encontrar una solución factible.
- No existen restricciones: **Optimización sin restricciones**
  - Determinar el mínimo de un función.

Optimización: áreas de estudio

- **Programación lineal (LP):**
  - Se utiliza en situaciones que pueden representarse utilizando:
    - **Expresiones lineales**: que no incluyan productos de variables.
    - **Variables continuas**.
  - Existen **algoritmos de resolución muy potentes** que permiten abordar problemas de gran tamaño:
    - > 10^6 variables, > 10^6 restricciones.
  - La **teoría de la dualidad** permite:
    - Interpretar económicamente los resultados obtenidos.
    - Analizar el impacto económico de las restricciones.
    - Obtener una nueva solución óptima de forma sencilla ante pequeñas variaciones de la situación.
  - Siempre que se pueda, **interesa** adoptar **este enfoque**.

Optimización: áreas de estudio

- **Programación lineal entera (MIP, MLIP):**
  - Se utiliza en situaciones que pueden representarse utilizando:
    - **Expresiones lineales**: que no incluyan productos de variables.
    - **Variables continuas**.
    - **Variables enteras** para representar niveles discretos de decisión.
    - **Variables binarias** para representar decisiones alternativas, etc.
  - Con este enfoque se puede plantear prácticamente cualquier problema, pero...
  - ...el **tamaño** de los problemas que se puede resolver es **menor**:
    - ~ 10^4 variables, ~ 10^4 restricciones.
  - Sólo debe utilizarse cuando no sea posible el enfoque LP.

Optimización: áreas de estudio

- **Programación no lineal (NLP):**
  - Se utiliza en situaciones que **deben** representarse utilizando:
    - **Expresiones no lineales**: como productos de variables.
    - **Variables continuas**.
  - En general es difícil garantizar que la solución obtenida es la óptima:
    - La programación cuadrática es una excepción.
  - La interpretación económica de la formulación suele ser interesante pero...
  - ...los algoritmos de resolución son poco potentes.
  - Es posible abordar sólo problemas de pequeño tamaño.
  - Normalmente es posible reformularlos como problemas MLIP.

Optimización: áreas de estudio

- **Programación dinámica (DP):**
  - Estudia problemas en los que las decisiones se toman en **varias etapas**.
    - El estado del sistema en cada etapa se define completamente con las variables de estado.
    - Las decisiones de cada etapa determinan el estado del sistema en etapas posteriores.
  - Para abordar estos problemas **se discretiza el abanico de estados** que el sistema puede tener **en cada etapa**.
  - La **estrategia de solución** se basa en el **principio de Bellman**:

> *Una política óptima se caracteriza por que, independientemente del estado inicial y de la decisión inicial, las decisiones restantes deben constituir una política óptima con respecto al estado resultante de la primera decisión*

Modelos y modelado

Álvaro Baíllo Moreno

Modelo

Esquema teórico, generalmente en forma **matemática**, de un sistema o de una realidad compleja (por ejemplo, la evolución económica de un país), que se elabora para **facilitar su comprensión y** el **estudio** de su comportamiento.

Diccionario de la lengua española. Real Academia Española.

Modelo

- Representación de una realidad:
  - **Compleja:** Puede involucrar equipo multidisciplinar.
  - **Precisa:** Formulada mediante expresiones matemáticas.
  - **Simplificada:**
    - Las simplificaciones introducidas son válidas en un cierto ámbito de utilización del modelo: no debe usarse fuera de ese ámbito.
    - Permiten mantener un equilibrio entre representación detallada y capacidad de obtener resultados numéricos.
- En el ámbito de la Investigación Operativa (IO):
  - Son herramientas de ayuda a la toma de **decisiones.**

Proceso de modelado

- Interacción entre dos agentes:

***Experto*:**
- Conoce el sistema de estudio.
- Quiere introducir mejoras.
- Desea asesoramiento para tomar decisiones.

***Modelador*:**
- Especifica y desarrolla el modelo que permite orientar esa toma de decisiones.
- Dos vertientes:
  - **Ciencia**
    - Análisis y detección de relaciones entre datos
    - Suposiciones y aproximaciones a los problemas
    - Algoritmos específicos de solución
  - **Arte**
    - Visión o interpretación de la realidad
    - Estilo en modelo y documentación
    - Elegancia y simplicidad en desarrollo
    - Uso de creativo de herramientas

Proceso de modelado

- **Diálogo** entre modelador y experto:
  - Entrevistas.
  - Conocimiento mutuo.
  - Manejo de un lenguaje común:
    - El modelador se familiariza con el sistema de estudio.
    - El experto se familiariza con las técnicas de modelado.
  - Especificación de objetivos comunes:
    - Acuerdo en la metodología y en los recursos.
- **Descripción** del comportamiento del sistema:
  - Identificación de variables de estado y de control.
  - Identificación de restricciones.
  - Identificación de posibles objetivos (económicos, de rendimiento, etc.)
  - Proporciona una primera aproximación al problema.

Proceso de modelado

- Formulación de **hipótesis** y **simplificaciones**
  - Ayudan a comprender mejor el problema.
  - Permiten que el problema sea tratable computacionalmente.
  - Es necesario valorar el impacto que pueden suponer en el resultado.
- Desarrollo de modelos reducidos o **prototipos**:
  - Permiten hacer **pruebas iniciales** y comprobar si el planteamiento es adecuado.
- Organización de la **información** disponible
  - Identificar la **información relevante** para el problema.
  - **Recopilación** de información: tarea costosa.
  - Creación de un **sistema de información**: e.g. base de datos.
  - Parámetros desconocidos: **estimaciones**.
  - Parámetros inciertos: representación de la **estocasticidad**.

Proceso de modelado

- **Primera versión del modelo:**
  - Definición de una **función objetivo**.
  - Especificación de las **variables** del problema, junto con sus límites.
  - Especificación de los datos de entrada o **parámetros**.
  - Incorporación de los **parámetros estocásticos**, mediante un equivalente determinista o mediante una representación de su distribución de probabilidad.
  - Formulación de las relaciones existentes entre variables y parámetros: **restricciones**.
  - Análisis de **tamaño y estructura** del problema
  - Identificación de **tipo de problema** (LP, MLIP, NLP, DP)
  - Énfasis en precisión y belleza en la formulación.

Proceso de modelado

- **Codificación:**
  - Selección de un lenguaje de programación o de modelado adecuado.
  - Desarrollo de un interfaz con el sistema de información.
- **Técnica de resolución:**
  - Selección de un algoritmo de obtención de solución óptima, cuasióptima o, al menos, satisfactoria.
  - Detección de soluciones cuasióptimas atractivas.
  - Evaluación de diferentes métodos de solución.
  - Diferentes implantaciones del algoritmo elegido.

Proceso de modelado

- **Verificación:**
  - Comparación entre el código implantado y el modelo.
  - Eliminación de errores en codificación.
  - Técnicas:
    - Probar con problemas de pequeño tamaño: prototipos.
    - Mantener algunas variables como constantes o eliminarlas.
    - Reforzar o relajar restricciones.
    - Eliminar la aleatoriedad de los elementos estocásticos.
- **Validación:**
  - Comparación de la salida del modelo con el sistema.
  - Comprobar validez de simplificaciones adoptadas.
  - Comprobación de adaptación a la realidad.
- **Refinamiento:**
  - Ampliación en el modelado por nuevas necesidades.

Proceso de modelado

- **Interpretación de los resultados:**
  - Análisis de sensibilidad en parámetros de entrada.
  - Robustez de la solución óptima.
- **Implantación, documentación y mantenimiento:**
  - Etapa fundamental para el éxito de un modelo.
  - Documentación clara, precisa y completa.
  - Manual de usuario con especificación técnica funcional, matemática e informática.
  - Formación de posibles usuarios.

Modelado en programación lineal

Álvaro Baíllo Moreno

Contenido

- Motivación con un ejemplo.
- Problema del transporte.
- Problema de asignación de tareas.
- Problema de transbordo.
- Problema del camino más corto.
- Resolución de problemas de la biblioteca.

Ejemplo LP1: Montaje de tarjetas

- La empresa LP monta dos tipos de **tarjetas electrónicas**, A y B, que se utilizan para el mismo tipo de aplicaciones:
  - La tarjeta A es de mejor calidad que la B, y por ello:
    - El **precio** que LP recibe por cada tarjeta tipo A, P_A, es mayor que el precio que paga por cada tarjeta tipo B, P_B.
    - El **tiempo de mano de obra** que requiere el montaje de A, T_A, es mayor que el tiempo de mano de obra que requiere B, T_B.
  - Las tarjetas que se pueden vender son limitadas:
    - Como mucho se pueden vender D tarjetas de cualquiera de los dos tipos.
  - La empresa LP cuenta con N **empleados** para el montaje de las tarjetas, cada uno de los cuales trabaja 8 h diarias.

Ejemplo LP1: Montaje de tarjetas Formulación del problema

- La empresa LP quiere saber el número de tarjetas de tipo A y B que tiene que montar, X_A y X_B, para:
  - **maximizar sus ingresos,**
  - vendiendo como mucho D tarjetas,
  - y respetando la restricción de tiempo impuesta por la mano de obra.

$$
\begin{aligned}
&\max P_A X_A + P_B X_B \\
&\text{s.a:} \\
&\qquad X_A + X_B \le D, \\
&\qquad T_A X_A + T_B X_B \le 8N, \\
&\qquad X_A, X_B \ge 0.
\end{aligned}
$$

La programación lineal como método para asignar recursos escasos

- Ejemplo LP1:
  - Los recursos escasos son la mano de obra y la demanda.

$$\max \boxed{P_A X_A + P_B X_B}$$

Criterio para asignar los recursos escasos

s.a:

$$\boxed{X_A + X_B \leq D,}$$

Recurso escaso 1: demanda

$$\boxed{T_A X_A + T_B X_B \leq 8N,}$$

Recurso escaso 2: mano de obra

$$\boxed{X_A, X_B \geq 0.}$$

Restricciones adicionales que deben cumplirse: cotas

El problema del transporte

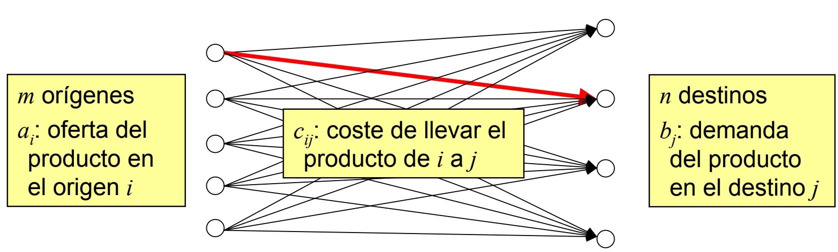
Función objetivo:

Minimizar el coste total de transporte de un cierto producto desde los orígenes a los destinos

Restricciones:

Satisfacer la demanda de cada destino

No superar la oferta disponible en cada origen



- ¿Qué **variable de control** nos interesa?

El problema del transporte

- Formulación del problema:

$$\min_{x_{ij}} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}\boxed{x_{ij}}$$

$$\text{s.a: } \boxed{\sum_{j=1}^{n} x_{ij} = a_i}\ \forall i,$$

$$\boxed{\sum_{i=1}^{m} x_{ij} = b_j}\ \forall j,$$

$$\boxed{x_{ij} \geq 0}\ \forall i, j.$$

Cantidad transportada desde i hasta j

Oferta disponible en cada origen i

Demanda en cada destino j

Cantidades positivas

El problema del transporte

- En la formulación anterior se ha supuesto que la oferta es igual a la demanda:

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

- Si la oferta es mayor que la demanda: $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$
  – Se añade un sumidero con coste de transporte nulo.
- Si la demanda es menor que la oferta: $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$
  – Se añade una fuente universal con coste muy elevado.

El problema del transporte

- Estructura del problema:

$$
\overbrace{x_{11}\quad x_{12}\quad \cdots\quad x_{1n}\quad x_{21}\quad x_{22}\quad \cdots\quad x_{2n}\quad \cdots\quad x_{m1}\quad x_{m2}\quad \cdots\quad x_{mn}}^{n\times m \text{ variables}}
$$

m restricciones de oferta

$$
\left\{
\begin{array}{cccccccccccc}
1 & 1 & \ldots & 1 & & & & & & & & \\
 & & & & 1 & 1 & \ldots & 1 & & & & \\
 & & & & & & & & \ddots & & & \\
 & & & & & & & & & 1 & 1 \;\ldots & 1
\end{array}
\right.
$$

n restricciones de demanda

$$
\left\{
\begin{array}{cccc|cccc|c|cccc}
1 & & & & 1 & & & & \ldots & 1 & & & \\
 & 1 & & & & 1 & & & \ldots & & 1 & & \\
 & & \ddots & & & & \ddots & & \ldots & & & \ddots & \\
 & & & 1 & & & & 1 & \ldots & & & & 1
\end{array}
\right.
$$

El problema del transporte

- La **matriz de restricciones** es unimodular:
  - Toda submatriz cuadrada tiene determinante 0, 1 ó -1.
- **Si la oferta** en todos los orígenes, a_i, **y la demanda** en todos los destinos, b_j, **toman valores enteros**:
  - **El valor de todas las variables** x_{ij} en la solución óptima **es entero**, sin necesidad de utilizar un algoritmo de programación lineal entera.

El problema de asignación de tareas

- Consideramos el problema de asignar n tareas a n personas (o máquinas) para realizarlas:
  - c_{ij} es el coste que supone la realización de la tarea i por la persona j.
  - El objetivo es minimizar el coste de realización de las tareas.
  - ¿Variable de control?
  - ¿Función objetivo?
  - Cada persona realiza una tarea. ¿Formulación?
  - Cada tarea es realizada por una persona. ¿Formulación?

El problema de asignación de tareas

$$\min_{x_{ij}} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}\boxed{x_{ij}}$$

$= 1$ si la persona i hace la tarea j
$= 0$ en otro caso

$$\text{s.a:}\ \boxed{\sum_{j=1}^{n} x_{ij} = 1,}\ \forall i,$$

Cada tarea i, una persona

$$\boxed{\sum_{i=1}^{m} x_{ij} = 1,}\ \forall j,$$

Cada persona, j una tarea

$$\boxed{x_{ij} \geq 0}\ \forall i, j.$$

Asignaciones positivas

El problema de transbordo

- Es una extensión del problema del transporte.
- En este caso **todos los nodos de la red admiten entrada y salida** de producto:
  - No se distingue entre nodos origen y destino: todos son iguales.
  - En lugar de distinguir entre oferta y demanda se define la **oferta neta** de cada nodo i, b_i (puede ser positiva, negativa o nula).
  - Suponemos que la **oferta** de producto es **igual a la demanda** de producto en el conjunto de la red.
  - Sigue habiendo un **coste de transporte** desde cada nodo i hasta cada nodo j de la red: c_{ij}

El problema de transbordo

- ¿Cómo se formularía este problema?

$$\min_{x_{ij}} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$$

$$\text{s.a:} \boxed{\sum_{j=1}^{n} x_{ij} - \sum_{k=1}^{n} x_{ki} = b_i} \forall i,$$

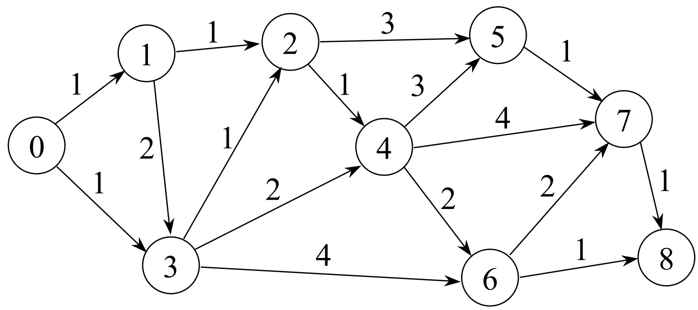
$$x_{ij} \geq 0, \forall i, j.$$

Balance del flujo en cada nodo i

Esta formulación es general y puede ser aplicada a los dos modelos vistos anteriormente.

El problema del camino más corto

- Se desea determinar el camino más corto para llegar del origen al destino en la red de la figura:



- ¿Cómo lo plantearíais?

El problema del camino más corto

$$
\begin{aligned}
&\min_{x_{ij}} \sum_{i=0}^{8}\sum_{j=0}^{8} c_{ij}x_{ij}\\
&\text{s.a: } \sum_{j=0}^{8} x_{ij} - \sum_{k=0}^{8} x_{ki} = b_i, \forall i,\\
&\quad x_{ij} \geq 0, \forall i, j,\\
&\quad b_0 = 1,\\
&\quad b_8 = -1,\\
&\quad b_i = 0, i = 1,\ldots,7
\end{aligned}
$$

Problemas de la biblioteca de problemas

- Ayuda en emergencias.
- Transporte de electrodomésticos.
- Logística.
- Producción e inventario.

Modelado en Programación Lineal
Formulación en GAMS

Álvaro Baíllo Moreno

Contenido

- Formulación del problema.
- Descripción del código en GAMS.
- Análisis de la salida de GAMS.

Ejemplo LP1: Montaje de tarjetas

- La empresa LP monta dos tipos de **tarjetas electrónicas**, A y B, que se utilizan para el mismo tipo de aplicaciones:
  - La tarjeta A es de mejor calidad que la B, y por ello:
    - El **precio** que LP recibe por cada tarjeta tipo A, P_A, es mayor que el precio que paga por cada tarjeta tipo B, P_B.
    - El **tiempo de mano de obra** que requiere el montaje de A, T_A, es mayor que el tiempo de mano de obra que requiere B, T_B.
  - Las tarjetas que se pueden vender son limitadas:
    - Como mucho se pueden vender D tarjetas de cualquiera de los dos tipos.
  - La empresa LP cuenta con N **empleados** para el montaje de las tarjetas, cada uno de los cuales trabaja 8 h diarias.

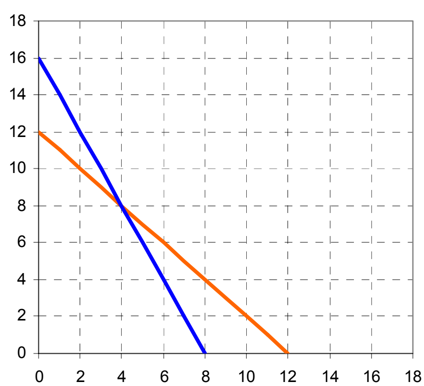
Ejemplo LP1: Montaje de tarjetas Formulación del problema

- La empresa LP quiere saber el número de tarjetas de tipo A y B que tiene que montar, X_A y X_B, para:
  - **maximizar sus ingresos,**
  - vendiendo como mucho D tarjetas,
  - y respetando la restricción de tiempo impuesta por la mano de obra.

$$
\begin{aligned}
&\max P_A X_A + P_B X_B \\
&\text{s.a:} \\
&\qquad X_A + X_B \le D, \\
&\qquad T_A X_A + T_B X_B \le 8N, \\
&\qquad X_A, X_B \ge 0.
\end{aligned}
$$

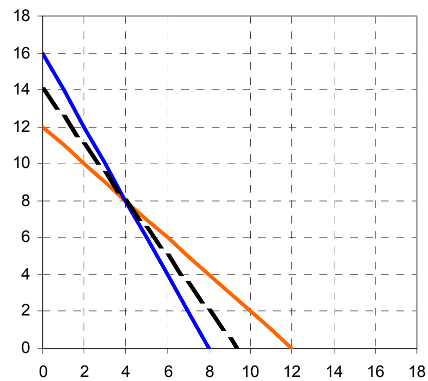
Ejemplo LP1: Montaje de tarjetas Caso numérico

P_A=3, P_B=2, T_A=4h, T_B=2h, D=12, N=4

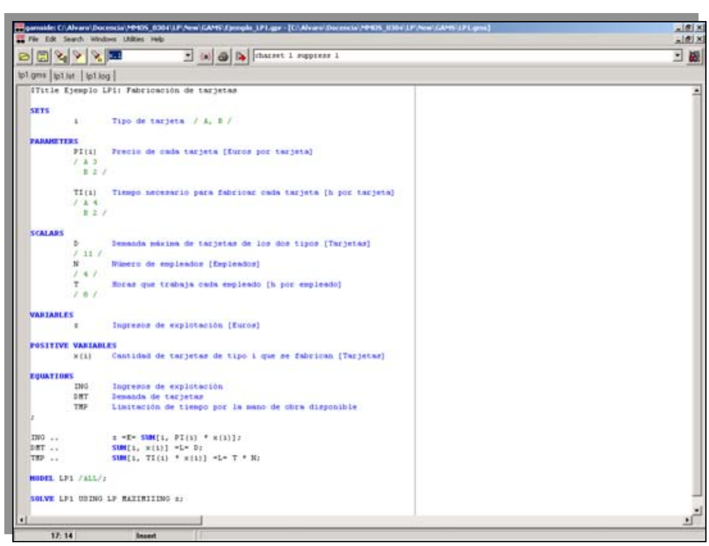


Ejemplo LP1: Montaje de tarjetas
Caso numérico

Recta isobeneficio correspondiente a $P_{T3} = 28$

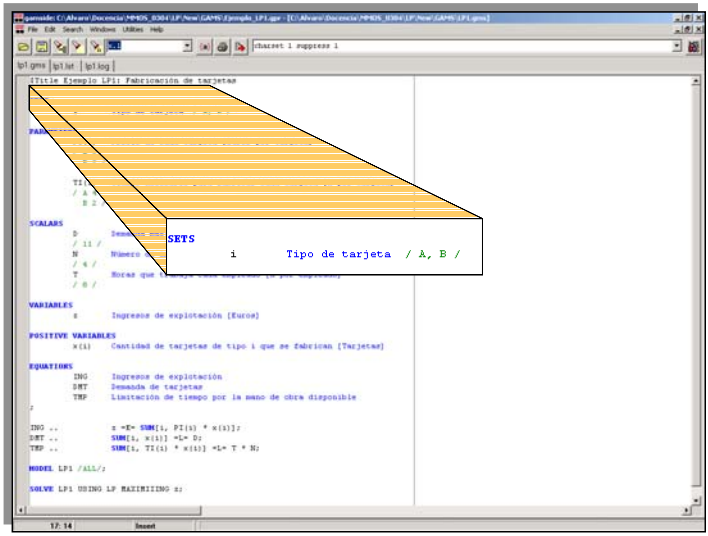


Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código

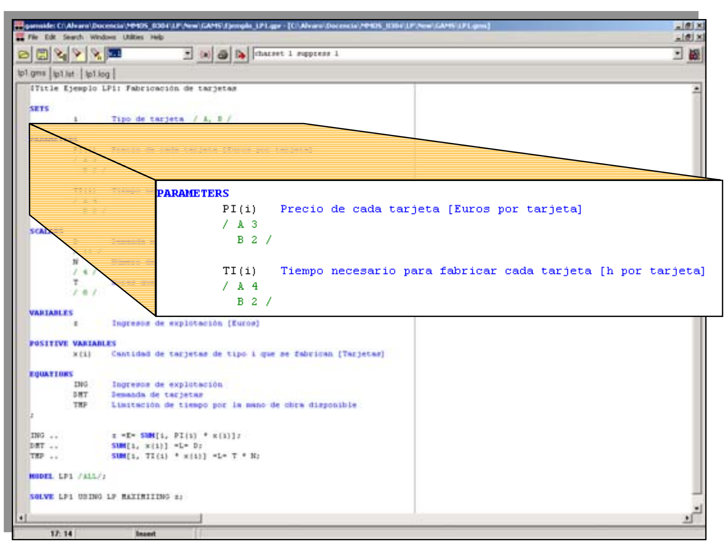


```
$Title Ejemplo LP1: Fabricación de tarjetas

SETS
        i       Tipo de tarjeta  / A, B /

PARAMETERS
        PI(i)   Precio de cada tarjeta [Euros por tarjeta]
        / A 3
          B 2 /

        TI(i)   Tiempo necesario para fabricar cada tarjeta [h por tarjeta]
        / A 4
          B 2 /

SCALARS
        D       Demanda máxima de tarjetas de los dos tipos [Tarjetas]
        / 11 /
        N       Número de empleados [Empleados]
        / 4 /
        T       Horas que trabaja cada empleado [h por empleado]
        / 8 /

VARIABLES
        z       Ingresos de explotación [Euros]

POSITIVE VARIABLES
        x(i)    Cantidad de tarjetas de tipo i que se fabrican [Tarjetas]

EQUATIONS
        ING     Ingresos de explotación
        DMT     Demanda de tarjetas
        TMP     Limitación de tiempo por la mano de obra disponible
;

ING ..          z =E= SUM[i, PI(i) * x(i)];
DMT ..          SUM[i, x(i)] =L= D;
TMP ..          SUM[i, TI(i) * x(i)] =L= T * N;

MODEL LP1 /ALL/;

SOLVE LP1 USING LP MAXIMIZING z;
```

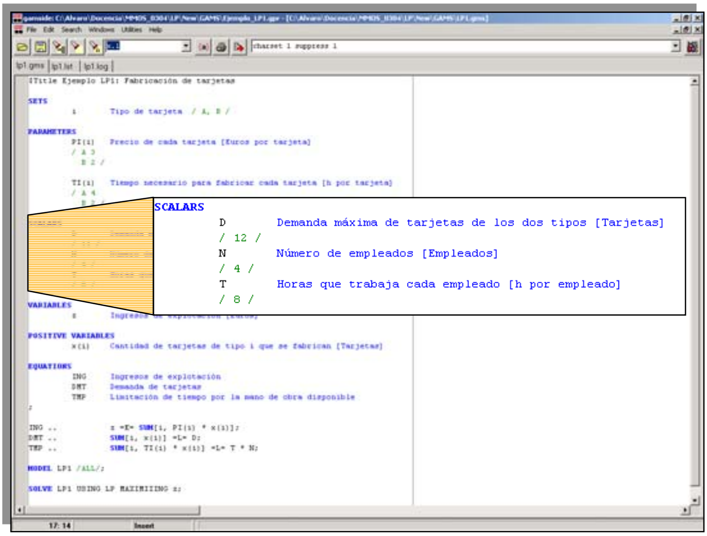
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código



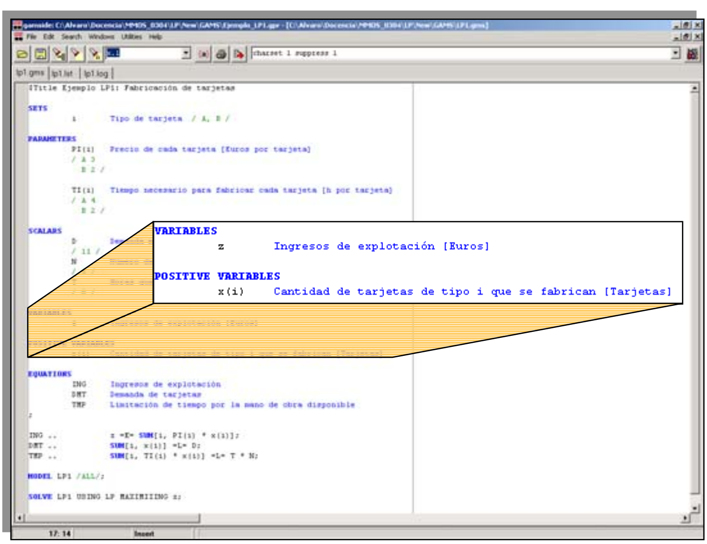
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código



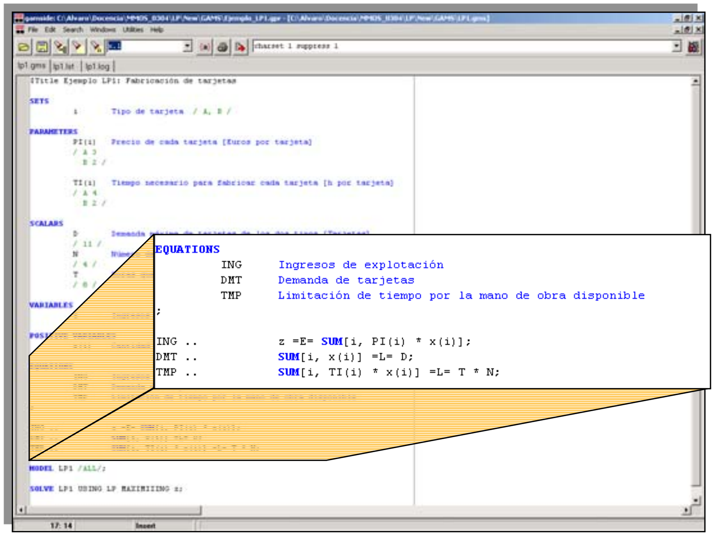
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código



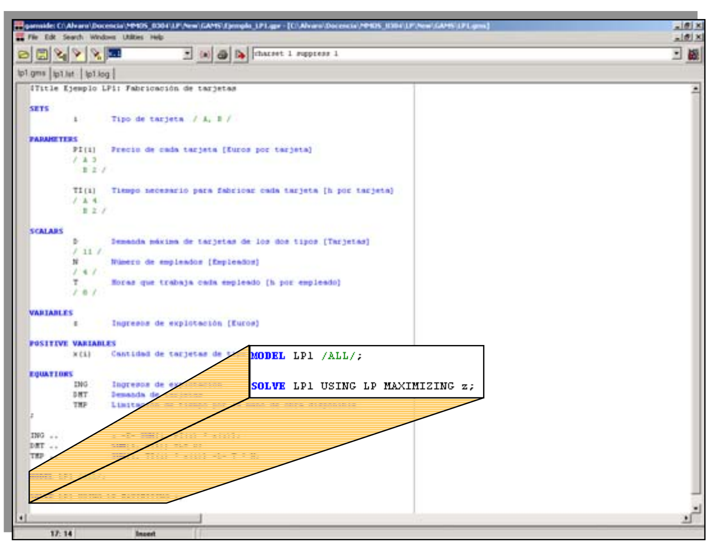
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código



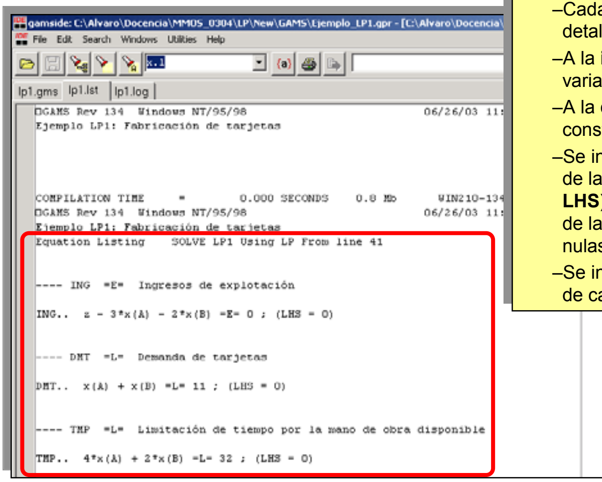
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código



Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: código

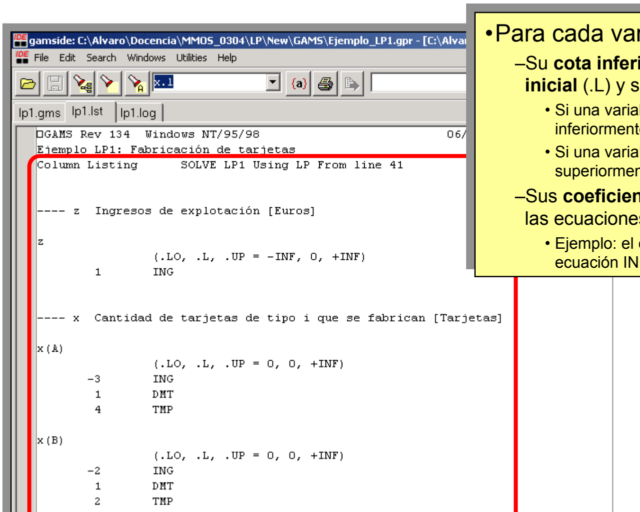


Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida



- Detalle de las ecuaciones:
  - Cada ecuación se desarrolla en detalle.
  - A la **izquierda** quedan todas las variables.
  - A la **derecha** están los términos constantes.
  - Se indica el **valor inicial** del lado de la izquierda (Left Hand Side, **LHS**) en función del valor inicial de las variables (por defecto son nulas).
  - Se indica la **infactibilidad inicial** de cada ecuación (INFES).

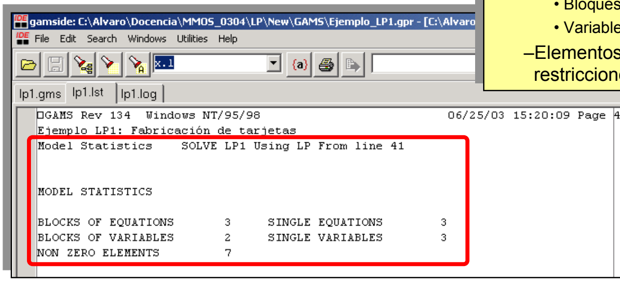
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida



- Para cada variable se indica:
  - Su **cota inferior** (.LO), su **valor inicial** (.L) y su **cota superior** (.UP)
    - Si una variable X no está acotada inferiormente: X.LO = -INF
    - Si una variable no está acotada superiormente: X.UP = +INF
  - Sus **coeficientes** en cada una de las ecuaciones del problema:
    - Ejemplo: el coeficiente de x(A) en la ecuación ING es -3.

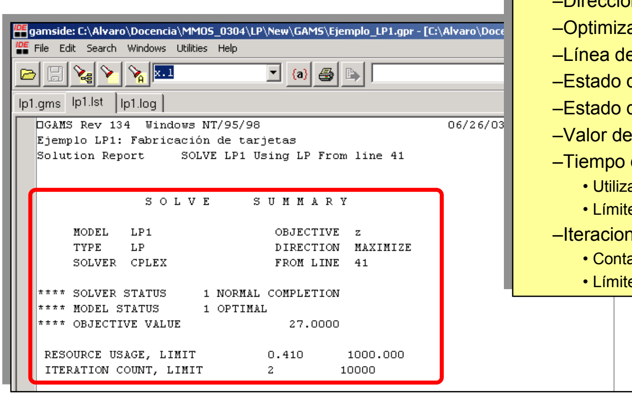
Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida

- Estadísticas del modelo:
  - Ecuaciones:
    - Bloques (3): ING, DMT, TMP.
    - Ecuaciones individuales (3).
  - Variables:
    - Bloques (2): z, x(i).
    - Variables individuales(3).
  - Elementos no nulos de la matriz de restricciones (7).



Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida

- Resumen de la solución:
  - Nombre del modelo: LP1
  - Variable objetivo: z
  - Tipo de modelo: LP
  - Dirección: maximización
  - Optimizador: CPLEX
  - Línea de la instrucción SOLVE: 41
  - Estado del optimizador: 1 (Normal)
  - Estado del modelo: 1 (Óptimo)
  - Valor de la función objetivo: 27.
  - Tiempo de CPU
    - Utilizado (Resource Usage): 0.410 s.
    - Límite: 1000 s.
  - Iteraciones:
    - Contadas (Iteration count): 2.
    - Límite: 10000 iteraciones.



Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida

- Detalle de la solución:
  - Valor de la función objetivo: 27.
  - Valor (LEVEL), cotas (LOWER, UPPER) y valor marginal (MARGINAL) de **cada ecuación**:
    - Ejemplo: DMT LOWER -INF
      LEVEL 11
      UPPER 11
      MARGINAL 1

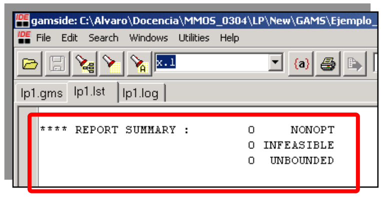
```
GAMS/Cplex    May 15, 2003 WIN.CP.CP 21.0 023.025.041.VIS For Cplex 8.1
Cplex 8.1.0, GAMS Link 23

Optimal solution found.
Objective :          27.000000


                           LOWER     LEVEL     UPPER    MARGINAL

---- EQU ING                 .         .         .        1.000
---- EQU DMT               -INF     11.000    11.000     1.000
---- EQU TMP               -INF     32.000    32.000     0.500

  ING  Ingresos de explotación
  DMT  Demanda de tarjetas
  TMP  Limitación de tiempo por la mano de obra disponible
```


Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida

- Detalle de la solución:
  - Valor (LEVEL), cotas (LOWER, UPPER) y valor marginal (MARGINAL) de **cada variable**:
    - Ejemplo: X(A): LOWER 0
      LEVEL 5
      UPPER +INF
      MARGINAL 0

```
                           LOWER     LEVEL     UPPER    MARGINAL

---- VAR z                 -INF     27.000     +INF       .

  z  Ingresos de explotación [Euros]

---- VAR x  Cantidad de tarjetas de tipo i que se fabrican [Tarjetas]

     LOWER     LEVEL     UPPER    MARGINAL

A      .       5.000     +INF       .
B      .       6.000     +INF       .
```

Ejemplo LP1: Montaje de tarjetas
Caso numérico con GAMS: salida



- Informe de la solución:
  - Informa acerca de problemas de infactibilidad o de no acotación.

UNIVERSIDAD PONTIFICIA ICAI ICADE COMILLAS MADRID
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA

Modelado en programación lineal entera

Álvaro Baíllo Moreno

Contenido

- Problemas MILP característicos:
  - Recubrimiento.
  - Empaquetado.
  - Partición.
  - Mochila.
  - Problema del viajante.
- Modelado con variables enteras y binarias:
  - Modelado de cantidades indivisibles.
  - Modelado de restricciones especiales.
  - Modelado de decisiones discretas.
  - Modelado de implicaciones.
  - Modelado de proposiciones lógicas.
  - Modelado de productos de variables binarias.

Modelos MILP característicos

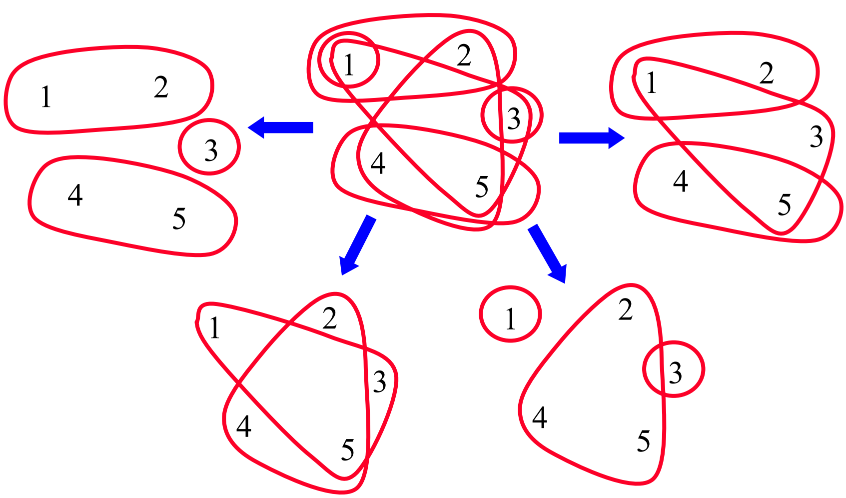
- Vamos a ver **algunos modelos MILP** que han sido **muy estudiados**.
- Con ello **se pretende**:
  - Hacer una **introducción** al modelado MILP.
  - Ilustrar la **importancia del modelado**:
    - En problemas sencillos para que su representación sea sencilla.
    - En problemas complejos para que su representación sea correcta.
  - Describir algunas **estructuras tipo** que se suelen presentar con más frecuencia de los que parece.

Problema de recubrimiento (Set covering)

- Consideremos un **conjunto S de personas**:
  - Por ejemplo: $S=\{1, 2, ..., 5\}$
  - Utilizamos el índice i para representar las personas: $i=1, ..., 5$.
- Supongamos que queremos organizar a estas personas en **varios equipos**:
  - El conjunto de equipos posibles es $\mathcal{S}$
  - Por ejemplo: $\mathcal{S} = \{\{1, 2\}, \{4, 5\}, \{1, 3, 5\}, \{2, 4, 5\}, \{1\}, \{3\}\}$
  - Utilizamos el índice j para representar los equipos: $j=1, ..., 6$.
  - Cada equipo j tiene un <u>coste</u> c_j:
    $c_1 = 1, c_2 = 1, c_3 = 1, c_4 = 1, c_5 = 1, c_6 = 1.$
- Queremos elegir los equipos que permiten tener ocupadas a **todas las personas con coste mínimo**.
  - No importa que una misma persona esté en dos equipos.

Problema de recubrimiento (Set covering)

- Hay múltiples soluciones factibles, pero sólo un óptimo:



Problema de recubrimiento (Set covering)

- Para formular el problema definimos una matriz cuyos elementos a_{ij} indican si la persona i está en el equipo j:
  $a_{ij} = 1$ si la persona i está en el equipo j.
  $a_{ij} = 0$ en caso contrario.
- La formulación es:

$$\min_{x_j} \sum_{j=1}^{m} \boxed{c_j x_j}$$

$$\text{s.a:} \boxed{\sum_{j=1}^{m} a_{ij} x_j \geq 1}, \forall i,$$

$$\boxed{x_j \in \{0,1\}}, \forall j.$$

Coste de los equipos formados

Cada persona i en al menos un equipo.

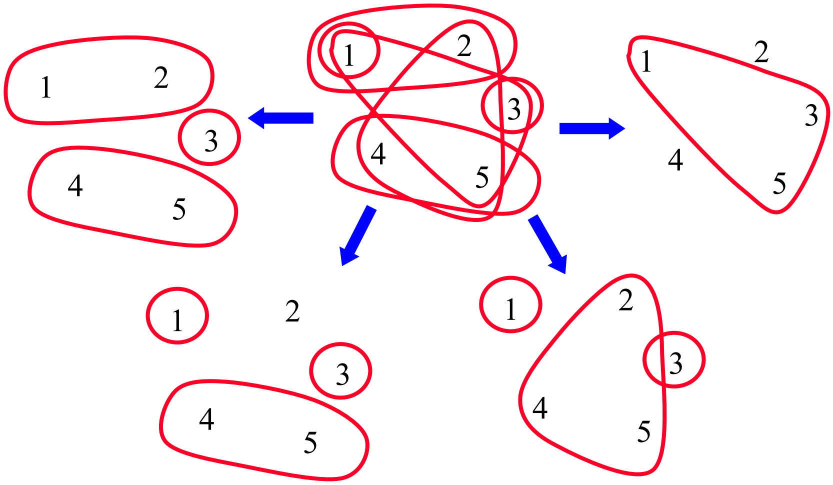
Variables binarias

Problema de empaquetado (Set packing)

- Consideremos un **conjunto S de personas**:
  - Por ejemplo: $S=\{1, 2, ..., 5\}$
  - Utilizamos el índice i para representar las personas: $i=1, ..., 5$.
- Supongamos que queremos organizar a estas personas en **varios equipos**:
  - El conjunto de equipos posibles es $\mathcal{S}$
  - Por ejemplo: $\mathcal{S} = \{\{1, 2\}, \{4, 5\}, \{1, 3, 5\}, \{2, 4, 5\}, \{1\}, \{3\}\}$
  - Utilizamos el índice j para representar los equipos: $j=1, ..., 6$.
  - Cada equipo j tiene un beneficio c_j:
    $c_1 = 1, c_2 = 1, c_3 = 1, c_4 = 1, c_5 = 1, c_6 = 1.$
- Queremos elegir los equipos que proporcionan **beneficio máximo sin que exista solape**.
  - No importa que una persona no esté en ningún equipo.

Problema de empaquetado (Set packing)

- Hay múltiples soluciones factibles, pero sólo un óptimo:



Problema de empaquetado (Set packing)

- La formulación es:

$$\max_{x_j} \sum_{j=1}^{m} \boxed{c_j x_j}$$

$$\text{s.a:} \boxed{\sum_{j=1}^{m} a_{ij} x_j \le 1}, \forall i,$$

$$\boxed{x_j \in \{0,1\}}, \forall j.$$

Beneficio de los equipos formados

Cada persona i en sólo un equipo.

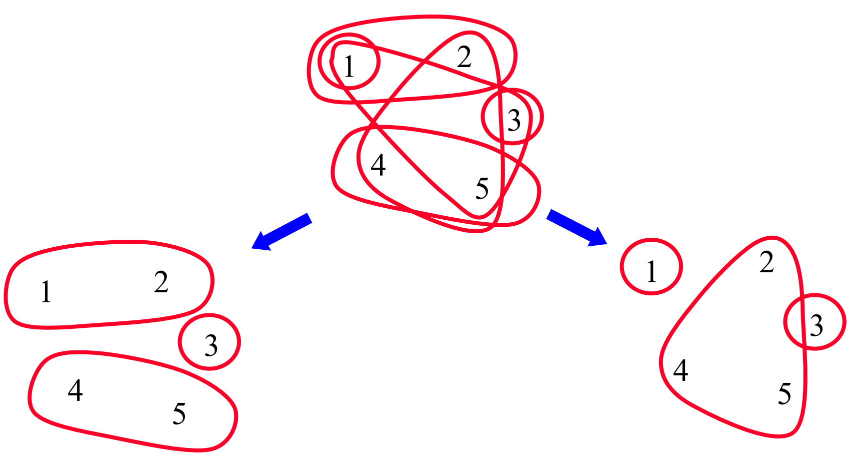
Variables binarias

Problema de partición (Set partitioning)

- Consideremos un **conjunto S de personas**:
  - Por ejemplo: $S=\{1, 2, ..., 5\}$
  - Utilizamos el índice i para representar las personas: $i=1, ..., 5$.
- Supongamos que queremos organizar a estas personas en **varios equipos**:
  - El conjunto de equipos posibles es $\mathcal{S}$
  - Por ejemplo: $\mathcal{S} = \{\{1, 2\}, \{4, 5\}, \{1, 3, 5\}, \{2, 4, 5\}, \{1\}, \{3\}\}$
  - Utilizamos el índice j para representar los equipos: $j=1, ..., 6$.
  - Cada equipo j tiene un beneficio c_j:
    $c_1 = 1, c_2 = 1, c_3 = 1, c_4 = 1, c_5 = 1, c_6 = 1.$
- Queremos elegir los equipos que proporcionan **beneficio máximo de forma que cada persona esté en un equipo y sólo un equipo**.

Problema de partición (Set partitioning)

- Hay múltiples soluciones factibles, pero sólo un óptimo:



Problema de partición (Set partitioning)

- La formulación es:

$$\max_{x_j} \sum_{j=1}^{m} \boxed{c_j x_j}$$

$$\text{s.a:} \boxed{\sum_{j=1}^{m} a_{ij} x_j = 1}, \forall i,$$

$$\boxed{x_j \in \{0,1\}} \; \forall j.$$

Beneficio de los equipos formados

Cada persona i en uno y sólo un equipo.

Variables binarias

Problema de la mochila (Knapsack)

- Supongamos una empresa que puede elegir entre n **proyectos**.
- Cada proyecto j ($j=1,\ldots,n$) tiene un **coste** c_j y supone un **beneficio** r_j.
- La empresa tiene un **presupuesto máximo** de b.
- El problema consiste en **elegir** aquellos **proyectos** que **maximizan el beneficio** respetando la restricción del **presupuesto**.

Problema de la mochila (Knapsack)

- Formulación:

$$\max_{x_j} \sum_{j=1}^{m} \boxed{r_j x_j}$$

$$\text{s.a:} \boxed{\sum_{j=1}^{m} c_j x_j \le b},$$

$$\boxed{x_j \in \{0,1\}}, \forall j.$$

Beneficio de los proyectos elegidos

Restricción de presupuesto

Variables binarias

- Aplicaciones:
  - Asignación de presupuestos.
  - Organización de un almacén.
  - ...

Problema del viajante (Traveling salesman)

- Objetivo:
  - Hacer un **recorrido** que pase por n ciudades sin repetir ninguna y volviendo a la ciudad de partida, de manera que la **distancia total recorrida sea mínima**.
- En este problema se tienen varias dificultades:
  - ¿Cuál es la **variable de control**?
  - ¿Cómo se garantiza que por cada ciudad se pasa sólo una vez?
  - ¿Cómo se impide que haya bucles incompletos?
- Vamos a ir respondiendo a cada una de estas preguntas.

Problema del viajante (Traveling salesman)

- Variable de control:
  - Vamos a utilizar los índices i e j para referirnos a las ciudades.
    - i: ciudad origen.
    - j: ciudad destino.
  - La variable de control es x_{ij}:

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

  - La distancia entre dos ciudades i e j viene dada por c_{ij}.

Problema del viajante (Traveling salesman)

- Formulación **incompleta** del problema:

$$\min_{x_{ij}} \sum_i \sum_{j \neq i} \boxed{c_{ij} x_{ij}}$$

Distancia recorrida

$$\boxed{\sum_{i \neq j} x_{ij} = 1}, \quad \forall j,$$

Cada ciudad j es destino de uno y sólo un tramo

$$\boxed{\sum_{j \neq i} x_{ij} = 1}, \quad \forall i,$$

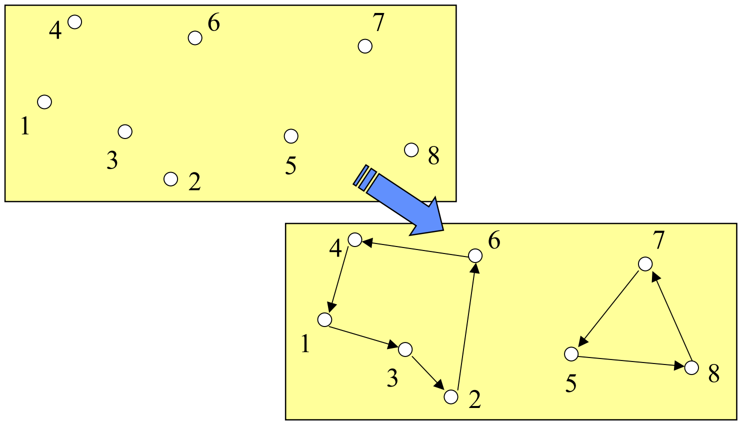
Cada ciudad j es origen de uno y sólo un tramo

$$\boxed{x_{ij} \in \{0,1\}}$$

Variables binarias

Problema del viajante (Traveling salesman)

- Es necesario añadir restricciones adicionales para evitar soluciones formadas por dos o más recorridos:



Problema del viajante (Traveling salesman)

- **Formulación 1:**
  - Consiste en añadir un conjunto de variables <u>enteras</u>:

$$u_i \in \mathbb{Z}, \quad i = 2, \ldots, n$$

  - Y se añaden las siguientes restricciones:

$$\boxed{u_i - u_j + (n-1) \cdot x_{ij} \le n-2, \quad \forall i > 1, \forall j \ne i, j > 1}$$

  - Vamos a particularizar estas restricciones para uno de los recorridos incompletos de la transparencia anterior (5 - 7 - 8):

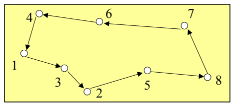
$$\left.\begin{aligned} u_5 - u_7 + 7 \cdot x_{57} &\le 6 \\ u_7 - u_8 + 7 \cdot x_{78} &\le 6 \\ u_8 - u_5 + 7 \cdot x_{85} &\le 6 \end{aligned}\right\} \Rightarrow 7 \cdot x_{57} + 7 \cdot x_{78} + 7 \cdot x_{85} \le 18 \Rightarrow 21 \le 18!$$

Sumamos las restricciones

  - Las restricciones no se cumplen en caso de recorridos incompletos, luego impiden la presencia de éstos.

Problema del viajante (Traveling salesman)

- **Formulación 1:**
  - Vamos a probar en el caso de un recorrido completo:



$$\boxed{\begin{aligned} u_3 - u_2 + 7 \cdot x_{13} &\le 6 \\ u_2 - u_5 + 7 \cdot x_{25} &\le 6 \\ u_5 - u_8 + 7 \cdot x_{58} &\le 6 \\ u_8 - u_7 + 7 \cdot x_{87} &\le 6 \\ u_7 - u_6 + 7 \cdot x_{76} &\le 6 \\ u_6 - u_4 + 7 \cdot x_{64} &\le 6 \end{aligned}}$$

Consideramos sólo las restricciones correspondientes a recorridos $i \to j$ que se hacen $(x_{ij} = 1)$

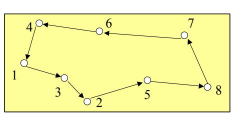
Sumamos las restricciones

$$\boxed{u_3 - u_4 + 7 \cdot x_{13} + 7 \cdot x_{25} + 7 \cdot x_{58} + 7 \cdot x_{87} + 7 \cdot x_{76} + 7 \cdot x_{64} \le 36}$$

$$\boxed{u_4 \ge u_3 + 6}$$

Problema del viajante (Traveling salesman)

- **Formulación 1:**
  - Si en lugar de sumar seis restricciones sumamos sólo cinco:



$$\boxed{\begin{aligned} u_3 - u_2 + 7 \cdot x_{13} &\le 6 \\ u_2 - u_5 + 7 \cdot x_{25} &\le 6 \\ u_5 - u_8 + 7 \cdot x_{58} &\le 6 \\ u_8 - u_7 + 7 \cdot x_{87} &\le 6 \\ u_7 - u_6 + 7 \cdot x_{76} &\le 6 \end{aligned}}$$

Sumamos las restricciones

$$\boxed{u_6 \ge u_3 + 5}$$

En definitiva, la variable u_i indica el orden en que se ha visitado la ciudad i.

  - Este procedimiento añade casi tantas restricciones como parejas de ciudades.

Problema del viajante (Traveling salesman)

- **Formulación 2:**
  - Consideramos todos los subconjuntos de más de 2 y de menos de $n-2$ ciudades que no contienen a la ciudad 1.
  - Añadimos una restricción para cada uno de estos subconjuntos U que impida que en él se forme un bucle:

$$\boxed{\sum_{i,j\in U} x_{ij} \le \mathrm{Card}(U)-1 \quad \forall U \subset \{1,\ldots,n\} \,\big|\, 2 \le \mathrm{Card}(U) \le n-2}$$

    donde $\mathrm{Card}(U)$ es el cardinal del subconjunto U (número de elementos de U).
  - Con este método estamos añadiendo del orden de $2n$ restricciones.
  - Por ejemplo, para $n=8$ tendríamos que añadir $\binom{7}{2}+\binom{7}{3}+\binom{7}{4}$ restricciones.
  - Al eliminar los bucles de 2 y 3 ciudades estamos eliminando también los de 5 y 6 ciudades.

Problema del viajante (Traveling salesman)

- **Formulación 3:**
  - Cambiamos la elección de la variable de control:
    - i: ciudad origen.
    - j: ciudad destino.
    - k: tramo de recorrido.
  - La variable de control es x_{ijk}:

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

Problema del viajante (Traveling salesman)

- **Formulación 3:**

$$\min_{x_{ijk}} \sum_i \sum_{j\neq i} \sum_k \boxed{c_{ij} x_{ijk}}$$ ← Distancia recorrida

$$\boxed{\sum_{i\neq j} \sum_k x_{ijk} = 1,} \quad \forall j,$$ ← Cada ciudad j es destino de uno y sólo un tramo

$$\boxed{\sum_{j\neq i} \sum_k x_{ijk} = 1,} \quad \forall i,$$ ← Cada ciudad j es origen de uno y sólo un tramo

$$\boxed{\sum_{i} \sum_{j\neq i} x_{ijk} = 1,} \quad \forall k,$$ ← En cada tramo k sólo un recorrido

$$\boxed{\sum_{i\neq j} x_{ijk} = \sum_{r\neq j} x_{jrk+1}} \quad \forall j,k,$$ ← Si j es destino del tramo k entonces es origen del $k{+}1$.

$$\boxed{x_{ijk} \in \{0,1\}}$$ ← Variables binarias

Modelado con variables enteras o binarias

- El uso de **variables enteras o binarias** aumenta considerablemente las posibilidades de modelado respecto a la programación lineal:
  - Modelado de **cantidades discretas** (e.g. número de personas que se asignan a una tarea).
  - Modelado de decisiones que implican un **coste fijo** o de **arranque**:
    - Adquirir o no un activo (un edificio, una máquina, etc.)
    - Poner en marcha un proceso.
  - Modelado de **decisiones que posibilitan** la toma de **otras decisiones**:
    - La compra de un determinado aparato (variable binaria) permite después tomar decisiones relativas a su operación.
  - Modelado de **restricciones no lineales y no convexas**.
  - Modelado de **implicaciones** y de **condiciones lógicas**.

Modelado de cantidades discretas

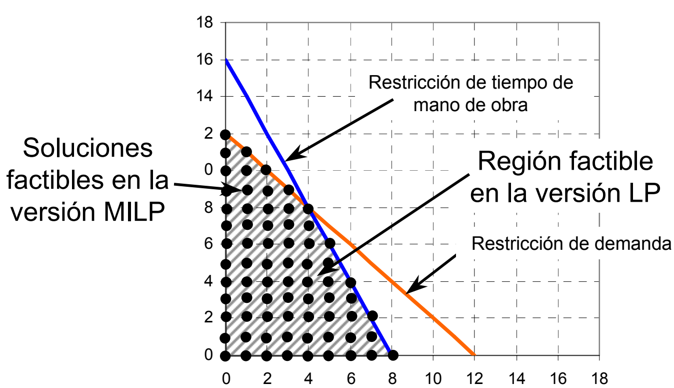
- Ejemplo LP1:
  – Supongamos que queremos que el número de tarjetas montadas sea siempre entero (no se pueden dejar tarjetas a medio montar para el día siguiente):

$$
\begin{aligned}
&\max \boxed{P_A X_A + P_B X_B} && \leftarrow \text{Criterio para asignar los recursos escasos}\\
&\text{s.a:} &&\\
&\quad \boxed{X_A + X_B \le D,} && \leftarrow \text{Recurso escaso 1: demanda}\\
&\quad \boxed{T_A X_A + T_B X_B \le 8N,} && \leftarrow \text{Recurso escaso 2: mano de obra}\\
&\quad \boxed{X_A, X_B \ge 0.} && \leftarrow \text{Restricciones adicionales que deben cumplirse: cotas}
\end{aligned}
$$

Modelado de cantidades discretas

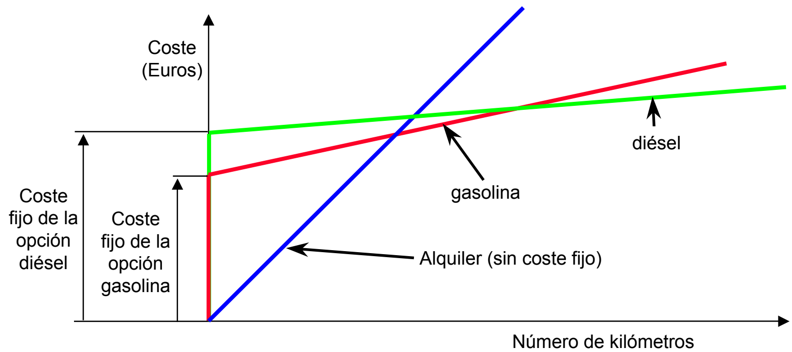
- Ejemplo LP1:

P_A=3, P_B=2, T_A=4h, T_B=2h, D=12, N=4



Modelado de decisiones que implican un coste fijo o de arranque

- **Coste fijo:** Ejemplo CF
  - Comparación entre la compra de un coche de gasolina, un coche diésel y la alternativa de utilizar coches de alquiler:



Modelado de decisiones que posibilitan la toma de otras decisiones:

- Ejemplo CF
  - Cada decisión i implica dos costes:
    - Coste fijo de adquisición: f_i
    - Coste variable por kilómetro recorrido: v_i
  - Utilizamos dos variables:
    - Elección de la alternativa i: $x_i \in \{0,1\}$
    - Distancia recorrida si se elige la alternativa i: y_i

$$
\begin{aligned}
&\min \sum_i \boxed{f_i x_i} + \boxed{v_i y_i} \\
&\text{s.a:} \\
&\quad \sum_i y_i \geq \boxed{d,} \\
&\quad \boxed{y_i \leq x_i d}, \forall i, \\
&\quad y_i \in \mathbb{R}, \forall i \\
&\quad x_i \in \{0,1\}.
\end{aligned}
$$

Coste variable asociado a la alternativa i

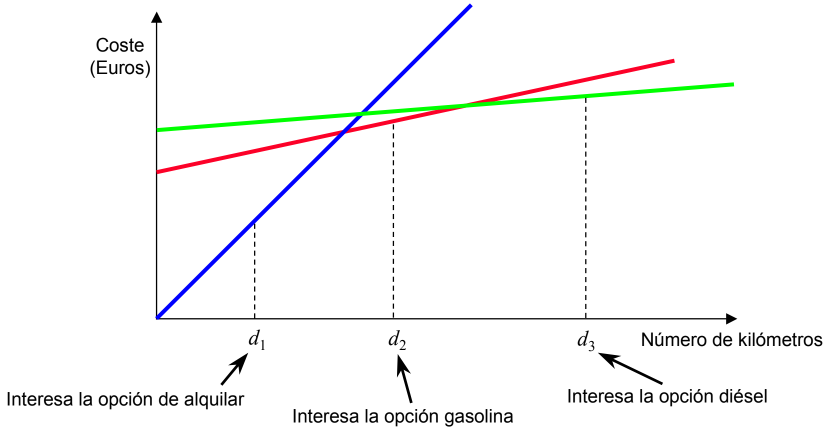
Coste fijo asociado a la alternativa i

Distancia mínima que se espera recorrer

Relación entre la distancia recorrida y la elección de la alternativa

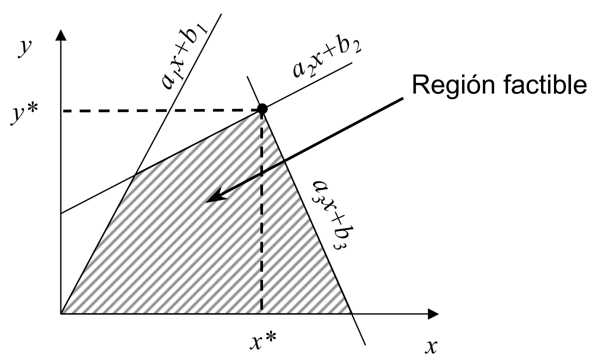
Modelado de decisiones que posibilitan la toma de otras decisiones:

- Ejemplo CF
  - La decisión de inversión (x_i) viene condicionada por el uso que se vaya a dar a esa inversión (y_i)



Modelado de restricciones no lineales y no convexas

- Consideremos el problema de maximizar y sabiendo que el valor de y tiene que encontrarse en la región factible de la figura:



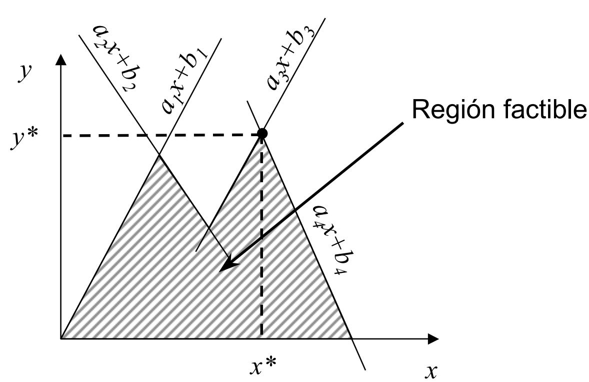
Modelado de restricciones no lineales y no convexas

- Este problema se puede formular en forma LP por ser la región factible **convexa**:

$$
\begin{aligned}
&\max\ y\\
&\text{s.a:}\\
&\quad y \le a_1 x + b_1,\\
&\quad y \le a_2 x + b_2,\\
&\quad y \le a_3 x + b_3,\\
&\quad x, y \in \mathbb{R},\\
&\quad x \ge 0.
\end{aligned}
$$

Modelado de restricciones no lineales y no convexas

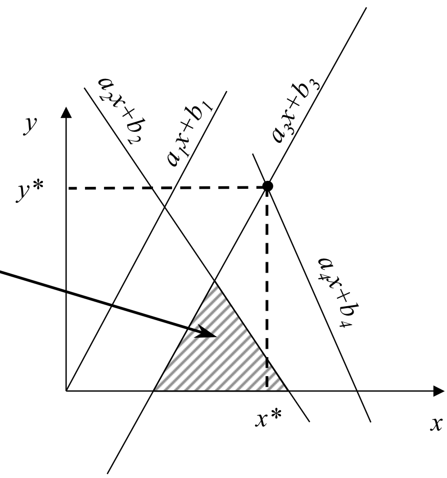
- Ahora consideremos el mismo problema pero con una región **no convexa**:



Modelado de restricciones no lineales y no convexas

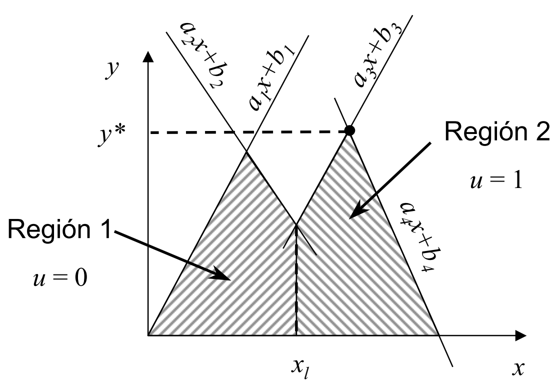
- No es posible dar una formulación LP para este problema:

$$
\begin{aligned}
&\max\ y\\
&\text{s.a:}\\
&\qquad \boxed{\begin{aligned} y &\le a_1 x + b_1,\\ y &\le a_2 x + b_2,\\ y &\le a_3 x + b_3,\\ y &\le a_4 x + b_4,\end{aligned}}\\
&\qquad x, y \in \mathbb{R},\\
&\qquad x \ge 0.
\end{aligned}
$$



Modelado de restricciones no lineales y no convexas

- En esta situación hay que dividir la región no convexa en regiones convexas y utilizar variables binarias para seleccionar la región convexa.



Modelado de restricciones no lineales y no convexas

- Formulamos el problema de la siguiente manera:

$$
\begin{aligned}
&\max y \\
&\text{s.a:} \\
&\qquad y \le \overbrace{a_1 x_1 + b_1(1-u)}^{\text{Región 1}} + \overbrace{a_3 x_2 + b_3 u}^{\text{Región 2}}, \\
&\qquad y \le a_2 x_1 + b_2(1-u) + a_4 x_2 + b_4 u, \\
&\qquad \boxed{0 \le x_1 \le x_l(1-u),} \longleftarrow \text{Región 1} \\
&\qquad \boxed{x_l u \le x_2 \le xu,} \longleftarrow \text{Región 2} \\
&\qquad x_1, x_2, y \in \mathbb{R}, \\
&\qquad u \in \{0,1\}.
\end{aligned}
$$

Modelado de restricciones no lineales y no convexas

- Si u=0:

$$
\begin{aligned}
&\max y \\
&\text{s.a:} \\
&\qquad y \le a_1 x_1 + b_1 + \cancel{a_3 x_2}, \\
&\qquad y \le a_2 x_1 + b_2 + \cancel{a_4 x_2}, \\
&\qquad 0 \le x_1 \le x_l, \\
&\qquad \boxed{0 \le x_2 \le 0,} \longleftarrow \text{Forzamos que } x_2\text{=0} \\
&\qquad x_1, x_2, y \in \mathbb{R}.
\end{aligned}
$$

Estamos
en la
Región 1

Modelado de restricciones no lineales y no convexas

- Si u=1:

$$
\begin{aligned}
&\max y\\
&\text{s.a:}\\
&\quad y \le \cancel{a_1 x_1} + a_3 x_2 + b_3,\\
&\quad y \le \cancel{a_2 x_1} + a_4 x_2 + b_4,\\
&\quad \boxed{0 \le x_1 \le 0,}\\
&\quad x_l \le x_2 \le \bar{x},\\
&\quad x_1, x_2, y \in \mathbb{R}.
\end{aligned}
$$

Forzamos que x_1=0

Estamos en la Región 2

Modelado de implicaciones

- Las variables binarias se pueden utilizar cuando se quiere modelar que el cumplimiento de una condición implica el cumplimiento de otra:

A se cumple $\Rightarrow$ B se cumple

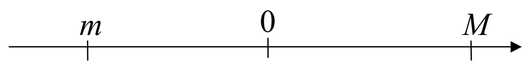
- Ejemplos:
  - Si una variable supera un cierto valor entonces una restricción debe cumplirse.
  - Si una restricción no se verifica entonces otra debe verificarse.
- Las variables binarias se utilizan para:
  - **Detectar** el cumplimento de restricciones.
  - **Forzar** el cumplimiento de restricciones.

Variables binarias como indicadores

Modelado de implicaciones

- Ejemplo 1:

$$\boxed{x \le M\delta}$$



  - M es un valor positivo *grande* y m un valor negativo *pequeño*.
  - x puede tomar cualquier valor entre m y M
  - δ es una variable binaria.
  - Veamos qué ocurre para cada uno de los posibles valores de δ:
    - $\delta = 0$ obliga a que $x \le 0$: $\boxed{\delta = 0 \Rightarrow x \le 0}$
    - $\delta = 1$ no obliga a nada.
  - Por otro lado, el valor de x también influye sobre el valor de δ:
    - $x > 0$ obliga a que $\delta = 1$: $\boxed{x > 0 \Rightarrow \delta = 1}$
    - $x \le 0$ no obliga a nada.

Modelado de implicaciones

- Es interesante comprobar que con la expresión $x \le M\delta$ hemos conseguido representar dos implicaciones:

$$\boxed{\text{IF } \delta = 0 \text{ THEN } x \le 0}$$

$$\boxed{\text{IF } x > 0 \text{ THEN } \delta = 1}$$

- En realidad estas dos implicaciones son la misma, si se tiene en cuenta que:

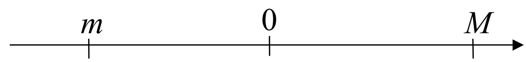
$$A \Rightarrow B \text{ equivale a } \overline{B} \Rightarrow \overline{A}$$

- La posibilidad de representar implicaciones utilizando expresiones matemáticas es importante, porque en programación matemática no es posible utilizar bucles IF ... THEN ...

Modelado de implicaciones

- Ejemplo 2:

$$\boxed{x \geq m\delta}$$



  - M es un valor positivo *grande* y m un valor negativo *pequeño*.
  - x puede tomar cualquier valor entre m y M
  - δ es una variable binaria.
  - Veamos qué ocurre para cada uno de los posibles valores de δ:
    - $\delta = 0$ obliga a que $x \geq 0$: $\boxed{\delta = 0 \Rightarrow x \geq 0}$
    - $\delta = 1$ no obliga a nada.
  - Por otro lado, el valor de x también influye sobre el valor de δ:
    - $x < 0$ obliga a que $\delta = 1$: $\boxed{x < 0 \Rightarrow \delta = 1}$
    - $x \geq 0$ no obliga a nada.

Modelado de implicaciones

- Con la expresión $x \leq m\delta$ hemos conseguido representar dos implicaciones:

$$\boxed{\text{IF } \delta = 0 \text{ THEN } x \geq 0}$$

$$\boxed{\text{IF } x < 0 \text{ THEN } \delta = 1}$$

- Nuevamente, estas dos implicaciones son la misma, si se tiene en cuenta que:

$$A \Rightarrow B \text{ equivale a } \overline{B} \Rightarrow \overline{A}$$

Modelado de implicaciones

- Resumen de lo que sabemos hacer hasta ahora:

$$\left.\begin{array}{l}\boxed{\text{IF } \delta = 0 \text{ THEN } x \le 0} \\ \boxed{\text{IF } x > 0 \text{ THEN } \delta = 1}\end{array}\right\} \quad \boxed{x \le M\delta}$$

$$\left.\begin{array}{l}\boxed{\text{IF } \delta = 0 \text{ THEN } x \ge 0} \\ \boxed{\text{IF } x < 0 \text{ THEN } \delta = 1}\end{array}\right\} \quad \boxed{x \ge m\delta}$$

Modelado de implicaciones

- Vamos a estudiar cuatro casos distintos a partir de la restricción:

$$\sum_j a_j x_j \le b$$

- Supondremos que siempre se verifican las siguientes condiciones:

$$\sum_j a_j x_j - b \le \boxed{M}$$

Suponemos que M es un valor positivo “muy grande”.

$$\sum_j a_j x_j - b \ge \boxed{m}$$

Suponemos que m es un valor negativo “muy pequeño”.

Modelado de implicaciones

- **Caso 1** (≤) $\text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j \le b$
  - Dicho de otra manera: $\delta = 1 \Rightarrow \sum_j a_j x_j \le b$
  - La formulación es: $\sum_j a_j x_j - b \le (1-\delta) M$

    de modo que hay dos situaciones posibles:

$$\delta = 0 \Rightarrow \boxed{\sum_j a_j x_j \le b + M}$$

No pasa nada: se cumple siempre.

$$\delta = 1 \Rightarrow \boxed{\sum_j a_j x_j \le b}$$

Forzamos el cumplimiento de la restricción.

Modelado de implicaciones

- Como en los ejemplos anteriores, la formulación

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

además de modelar la implicación

$$\text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j \le b$$

también modela la implicación:

$$\text{IF } \sum_j a_j x_j > b \text{ THEN } \delta = 0$$

Modelado de implicaciones

– Veamos una particularización:

$$\boxed{\text{IF } \delta = 1 \text{ THEN } x \le 0}$$

– La formulación correspondiente es:

$$\boxed{x \le (1-\delta)M}$$

– Obsérvese la similitud con el Ejemplo 1 visto anteriormente.

Modelado de implicaciones

- **Caso 2** (≥) $\boxed{\text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j \ge b}$

– Dicho de otra manera: $\boxed{\delta = 1 \Rightarrow \sum_j a_j x_j \ge b}$

– La formulación es: $\boxed{\sum_j a_j x_j - b \ge (1-\delta)m}$

de modo que hay dos situaciones posibles:

$\delta = 0 \Rightarrow \boxed{\sum_j a_j x_j \ge b + m}$ ← No pasa nada: se cumple siempre.

$\delta = 1 \Rightarrow \boxed{\sum_j a_j x_j \ge b}$ ← Forzamos el cumplimiento de la restricción.

Modelado de implicaciones

– Como en los ejemplos anteriores, la formulación

$$\boxed{\sum_j a_j x_j - b \geq (1-\delta) m}$$

además de modelar la implicación

$$\boxed{\text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j \geq b}$$

también modela la implicación:

$$\boxed{\text{IF } \sum_j a_j x_j < b \text{ THEN } \delta = 0}$$

Modelado de implicaciones

– Veamos una particularización:

$$\boxed{\text{IF } \delta = 1 \text{ THEN } x \geq 0}$$

– La formulación correspondiente es:

$$\boxed{x \geq (1-\delta) m}$$

– Obsérvese la similitud con el Ejemplo 2 visto anteriormente.

Modelado de implicaciones

- **Caso 3** $(\geq)$ $\boxed{\text{IF } \sum_j a_j x_j \geq b \text{ THEN } \delta = 1}$

  dicho de otra manera: $\boxed{\sum_j a_j x_j \geq b \Rightarrow \delta = 1}$

  – Sabiendo que las dos expresiones siguientes son equivalentes:

  $A \Rightarrow B$ equivale a $\overline{B} \Rightarrow \overline{A}$

  convertimos la implicación en otra:

  $$\sum_j a_j x_j \geq b \Rightarrow \delta = 1 \text{ equivale a } \delta = 0 \Rightarrow \sum_j a_j x_j < b$$

  que a su vez equivale a:

  $$\boxed{\delta = 0 \Rightarrow \sum_j a_j x_j \leq b - \varepsilon}$$

Modelado de implicaciones

– Hemos llegado a una implicación similar a la del Caso 1.

– La diferencia es que en lugar de $\sum_j a_j x_j - b$ estamos manejando la expresión $\sum_j a_j x_j - b + \varepsilon$

– Por este motivo, en lugar de M hay que manejar $M + \varepsilon$.

– La formulación de esta implicación es:

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

de modo que hay dos situaciones posibles:

$\delta = 0 \Rightarrow \boxed{\sum_j a_j x_j \leq b - \varepsilon}$ ← Forzamos el cumplimiento de la restricción.

$\delta = 1 \Rightarrow \boxed{\sum_j a_j x_j \leq b + M}$ ← No pasa nada: se cumple siempre.

Modelado de implicaciones

– Como en los ejemplos anteriores, la formulación

$$\boxed{\sum_j a_j x_j - b + \varepsilon \le \delta\left(M + \varepsilon\right)}$$

además de modelar la implicación

$$\boxed{\text{IF } \sum_j a_j x_j \ge b \text{ THEN } \delta = 1}$$

también modela la implicación:

$$\boxed{\text{IF } \delta = 0 \text{ THEN } \sum_j a_j x_j < b}$$

Modelado de implicaciones

– Veamos una particularización:

$$\boxed{\text{IF } x \ge 0 \text{ THEN } \delta = 1}$$

– La formulación correspondiente es:

$$\boxed{x + \varepsilon \le \delta\left(M + \varepsilon\right)}$$

Modelado de implicaciones

- **Caso 4** ($\leq$) $\boxed{\text{IF } \sum_j a_j x_j \leq b \text{ THEN } \delta = 1}$

  dicho de otra manera: $\boxed{\sum_j a_j x_j \leq b \Rightarrow \delta = 1}$

  – Sabiendo que las dos expresiones siguientes son equivalentes:

  $A \Rightarrow B$ equivale a $\overline{B} \Rightarrow \overline{A}$

  convertimos la implicación en otra:

  $\sum_j a_j x_j \leq b \Rightarrow \delta = 1$ equivale a $\delta = 0 \Rightarrow \sum_j a_j x_j > b$

  que a su vez equivale a:

$$\boxed{\delta = 0 \Rightarrow \sum_j a_j x_j \geq b + \varepsilon}$$

Modelado de implicaciones

– Hemos llegado a una implicación similar a la del Caso 1.

– La diferencia es que en lugar de $\sum_j a_j x_j - b$ estamos manejando la expresión $\sum_j a_j x_j - b - \varepsilon$

– Por este motivo, en lugar de m hay que manejar $m - \varepsilon$.

– La formulación de esta implicación es:

$$\boxed{\sum_j a_j x_j - b - \varepsilon \geq \delta(m - \varepsilon)}$$

de modo que hay dos situaciones posibles:

$\delta = 0 \Rightarrow \boxed{\sum_j a_j x_j \geq b + \varepsilon} \longleftarrow$ Forzamos el cumplimiento de la restricción.

$\delta = 1 \Rightarrow \boxed{\sum_j a_j x_j \geq b + m} \longleftarrow$ No pasa nada: se cumple siempre.

Modelado de implicaciones

– Como en los ejemplos anteriores, la formulación

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

además de modelar la implicación

$$\text{IF } \sum_j a_j x_j \leq b \text{ THEN } \delta = 1$$

también modela la implicación:

$$\text{IF } \delta = 0 \text{ THEN } \sum_j a_j x_j > b$$

Modelado de implicaciones

– Veamos una particularización:

$$\text{IF } x \leq 0 \text{ THEN } \delta = 1$$

– La formulación correspondiente es:

$$x - \varepsilon \geq \delta(m - \varepsilon)$$

Modelado de implicaciones

- **Caso 5** $(=)$ $\boxed{\text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j = b}$

  dicho de otra manera: $\boxed{\delta = 1 \Rightarrow \sum_j a_j x_j = b}$

  - Este caso es una combinación de los casos 1 y 2.
  - La formulación es:

$$\boxed{\begin{array}{l} \sum_j a_j x_j - b \le (1-\delta) M \\ \sum_j a_j x_j - b \ge (1-\delta) m \end{array}}$$

  - $\delta = 1$ **fuerza** el cumplimiento de las dos restricciones simultáneamente.

Modelado de implicaciones

- **Caso 6** $(=)$ $\boxed{\text{IF } \sum_j a_j x_j = b \text{ THEN } \delta = 1}$

  dicho de otra manera: $\boxed{\sum_j a_j x_j = b \Rightarrow \delta = 1}$

  - Este caso es una combinación de los casos 3 y 4.
  - Con una variable δ_1 detectamos el cumplimiento del caso 3:

$$\sum_j a_j x_j - b + \varepsilon \le \delta_1 (M + \varepsilon)$$

  - Con otra variable δ_2 detectamos el cumplimiento del caso 4:

$$\sum_j a_j x_j - b - \varepsilon \ge \delta_2 (m - \varepsilon)$$

Modelado de implicaciones

- Hacemos que $\delta = 1$ **detecte** que $\delta_1 = 1$ y que $\delta_2 = 1$:

$$\boxed{\delta \geq \delta_1 + \delta_2 - 1}$$

Estamos modelando una proposición lógica:

Si $\delta_1 = 1$ y $\delta_2 = 1$ entonces $\delta = 1$

- La formulación completa es:

$$\sum_j a_j x_j - b + \varepsilon \leq \delta_1 (M + \varepsilon)$$
$$\sum_j a_j x_j - b - \varepsilon \geq \delta_2 (m - \varepsilon)$$
$$\delta \geq \delta_1 + \delta_2 - 1$$

Modelado de implicaciones: Resumen

IF $\delta = 1$ THEN $\sum_j a_j x_j \leq b$	$\sum_j a_j x_j - b \leq (1-\delta) M$
IF $\delta = 1$ THEN $\sum_j a_j x_j \geq b$	$\sum_j a_j x_j - b \geq (1-\delta) m$
IF $\delta = 1$ THEN $\sum_j a_j x_j = b$	$\sum_j a_j x_j - b \leq (1-\delta) M$ $\sum_j a_j x_j - b \geq (1-\delta) m$
IF $\sum_j a_j x_j \geq b$ THEN $\delta = 1$	$\sum_j a_j x_j - b + \varepsilon \leq \delta (M + \varepsilon)$
IF $\sum_j a_j x_j \leq b$ THEN $\delta = 1$	$\sum_j a_j x_j - b - \varepsilon \geq \delta (m - \varepsilon)$
IF $\sum_j a_j x_j = b$ THEN $\delta = 1$	$\sum_j a_j x_j - b + \varepsilon \leq \delta_1 (M + \varepsilon)$ $\sum_j a_j x_j - b - \varepsilon \geq \delta_2 (m - \varepsilon)$ $\delta \geq \delta_1 + \delta_2 - 1$

Modelado de dobles implicaciones

- **Caso 1:** $\boxed{\delta = 1 \Leftrightarrow \sum_j a_j x_j \le b}$

$$\left.\begin{array}{l} \text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j \le b \\ \text{IF } \sum_j a_j x_j \le b \text{ THEN } \delta = 1 \end{array}\right\} \quad \delta = 1 \Leftrightarrow \sum_j a_j x_j \le b \quad \left\{\begin{array}{l} \sum_j a_j x_j - b \le (1-\delta) M \\ \sum_j a_j x_j - b - \varepsilon \ge \delta (m - \varepsilon) \end{array}\right.$$

$$\delta = 0 \Rightarrow \left\{\begin{array}{l} \sum_j a_j x_j - b \le M \\ \sum_j a_j x_j - b - \varepsilon \ge 0 \end{array}\right.$$

$\sum_j a_j x_j - b \le M$ ← No pasa nada: se cumple siempre.

$\sum_j a_j x_j - b - \varepsilon \ge 0$ ← Equivale a lo que queríamos → $\sum_j a_j x_j \le b \Rightarrow \delta = 1$

$$\delta = 1 \Rightarrow \left\{\begin{array}{l} \sum_j a_j x_j - b \le 0 \\ \sum_j a_j x_j - b \ge m \end{array}\right.$$

$\sum_j a_j x_j - b \le 0$ ← Lo que queríamos.

$\sum_j a_j x_j - b \ge m$ ← No pasa nada: se cumple siempre.

Modelado de dobles implicaciones

- **Caso 2:** $\boxed{\delta = 1 \Leftrightarrow \sum_j a_j x_j \ge b}$

$$\left.\begin{array}{l} \text{IF } \delta = 1 \text{ THEN } \sum_j a_j x_j \ge b \\ \text{IF } \sum_j a_j x_j \ge b \text{ THEN } \delta = 1 \end{array}\right\} \quad \delta = 1 \Leftrightarrow \sum_j a_j x_j \ge b \quad \left\{\begin{array}{l} \sum_j a_j x_j - b \ge (1-\delta) m \\ \sum_j a_j x_j - b + \varepsilon \le \delta (M + \varepsilon) \end{array}\right.$$

$$\delta = 0 \Rightarrow \left\{\begin{array}{l} \sum_j a_j x_j - b \ge m \\ \sum_j a_j x_j - b + \varepsilon \le 0 \end{array}\right.$$

$\sum_j a_j x_j - b \ge m$ ← No pasa nada: se cumple siempre.

$\sum_j a_j x_j - b + \varepsilon \le 0$ ← Equivale a lo que queríamos → $\sum_j a_j x_j \ge b \Rightarrow \delta = 1$

$$\delta = 1 \Rightarrow \left\{\begin{array}{l} \sum_j a_j x_j - b \ge 0 \\ \sum_j a_j x_j - b \le M \end{array}\right.$$

$\sum_j a_j x_j - b \ge 0$ ← Lo que queríamos.

$\sum_j a_j x_j - b \le M$ ← No pasa nada: se cumple siempre.

Modelado de proposiciones lógicas

- Hemos visto cómo expresar implicaciones entre restricciones y variables binarias:
  - Ya somos capaces de formular matemáticamente las expresiones:

    "si se cumple esta restricción entonces δ debe valer 1".

    "si δ vale 1 entonces debe cumplirse esta restricción ".

    "esta restricción se cumple si y sólo si δ vale 1".
  - En general, somos capaces de detectar el cumplimiento de una restricción con variables binarias.
- Ahora nos planteamos la posibilidad de establecer **relaciones lógicas entre dos o más restricciones**.
- Al cumplimiento de una restricción lo llamaremos **proposición**:
  - Ejemplo de proposición: "se verifica que $\sum_j a_j x_j \geq b$"

Modelado de proposiciones lógicas sencillas

- Consideremos que tenemos dos proposiciones, X_1 y X_2.
  - Supongamos que queremos modelar dos situaciones:
    - Al menos una de las dos proposiciones debe ser cierta.
    - Las dos proposiciones deben ser ciertas.
  - Para modelar estas situaciones, a cada proposición le asignamos una variable binaria de manera que:

    $\delta_1 = 1 \Rightarrow X_1$ es cierta $\quad \delta_2 = 1 \Rightarrow X_2$ es cierta
  - Si queremos que se cumpla al menos una de las dos restricciones:

    $\boxed{X_1 \text{ o } X_2 \text{ equivale a } \delta_1 + \delta_2 \geq 1}$
  - Si queremos que se cumplan las dos:

    $\boxed{X_1 \text{ y } X_2 \text{ equivale a } \delta_1 = 1, \delta_2 = 1}$

Modelado de proposiciones lógicas sencillas

- Consideremos que tenemos dos proposiciones, X_1 y X_2.
  - Supongamos que queremos modelar la situación:
    - Si X_1 es cierta entonces X_2 debe serlo.
  - A cada proposición le asignamos una variable binaria de manera que:

$$X_1 \text{ es cierta} \Rightarrow \delta_1 = 1 \qquad \delta_2 = 1 \Rightarrow X_2 \text{ es cierta}$$

  - La situación se modela entonces:

$$\boxed{X_1 \Rightarrow X_2 \text{ equivale a } \delta_1 - \delta_2 \leq 0}$$

Modelado de proposiciones lógicas sencillas

- Consideremos que tenemos dos proposiciones, X_1 y X_2.
  - Supongamos que queremos modelar la situación:
    - X_1 es cierta si y sólo si X_2 es cierta.
  - A cada proposición le asignamos una variable binaria de manera que:

$$X_1 \text{ es cierta} \Leftrightarrow \delta_1 = 1 \qquad \delta_2 = 1 \Leftrightarrow X_2 \text{ es cierta}$$

  - La situación se modela entonces:

$$\boxed{X_1 \Leftrightarrow X_2 \text{ equivale a } \delta_1 - \delta_2 = 0}$$

Modelado de proposiciones lógicas complejas

- **Ejemplo:**
  - Si se fabrica el producto A o B (o ambos) entonces debe fabricarse también al menos uno de los productos C, D o E.

    X_i proposición: se fabrica el producto i

    $\delta_i=1$ variable binaria asociada al cumplimiento de la proposición i

    $$X_i \Leftrightarrow \delta_i = 1$$
  - Esta situación se modela como sigue:

    $\delta_A + \delta_B \geq 1 \Rightarrow \delta_C + \delta_D + \delta_E \geq 1$
  - Añadimos una variable binaria intermedia δ para obtener una doble implicación:

    $\delta_A + \delta_B \geq 1 \Rightarrow \delta = 1 \Rightarrow \delta_C + \delta_D + \delta_E \geq 1$
  - Veamos cómo se modela cada una de estas implicaciones

    $\delta_A + \delta_B \geq 1 \Rightarrow \delta = 1 \equiv \delta_A + \delta_B - 2\delta \leq 0$

    $\delta = 1 \Rightarrow \delta_C + \delta_D + \delta_E \geq 1 \equiv - \delta_C - \delta_D - \delta_E + \delta \leq 0$

Modelado de proposiciones lógicas complejas

- **Formulación alternativa**:

  $(X_A \text{ o } X_B) \rightarrow (X_C \text{ o } X_D \text{ o } X_E)$

  equivale a

  $\delta_A \geq 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$

  $\delta_B \geq 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$

  $\delta_A \geq 1 \rightarrow \delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$

  $\delta_B \geq 1 \rightarrow \delta = 1 \rightarrow \delta_C + \delta_D + \delta_E \geq 1$

- Formulación:

  $X_i \leq \delta_i$

  $\delta_A - \delta \leq 0$

  $\delta_B - \delta \leq 0$

  $- \delta_C - \delta_D - \delta_E + \delta \leq 0$

  $\delta_i \in \{0,1\}$

  $\delta \in \{0,1\}$

Productos con variables binarias

$\delta_1 \ \delta_2 = 0$	$\delta_1 = 0$ ó $\delta_2 = 0$	$\delta_1' + \delta_2' \geq 1$ $\delta_1 + \delta_1' = 1$ $\delta_2 + \delta_2' = 1$ $\delta_i, \delta_i' \in \{0,1\}$
$\delta_1 \ \delta_2 = \delta_3$	$\delta_3 = 1 \leftrightarrow \delta_1 = 1$ y $\delta_2 = 1$	$-\delta_1 \quad +\delta_3 \leq 0$ $-\delta_2 \quad +\delta_3 \leq 0$ $\delta_1 \quad +\delta_2 \quad -\delta_3 \leq 1$ $\delta_i \in \{0,1\}$
$x\delta$ $x \geq 0$ $\delta \in \{0,1\}$	Reemplazar $x\delta$ por y $\delta = 0 \to y = 0$ $\delta = 1 \to y = x$	$y \geq 0$ $y \leq M\delta$ $-x + y \quad \leq 0$ $x - y \quad +M\delta \quad \leq M$ $x \leq M$

Problemas de la biblioteca de problemas

- Construcción de almacenes.
- Gestión de autobuses.
- Adquisición de camiones.
- Planificación del Metro.
- Oficina de correos.
- Abastecimiento.
- Adquisición de máquinas troqueladoras.
- Secuenciación de trabajos en una máquina.
- Producción II.