Herramientas para el diseño y análisis de redes de transporte urbano de pasajeros

Tema 4: Elementos básicos de programación matemática

Formulación general de programación matemática

$$
\begin{aligned}
\min \quad & f(x) & \\
\text{s.a.} \quad & h_i(x) = 0 & \forall\ i = 1, 2, \ldots, m, \\
& g_j(x) \leq 0 & \forall\ j = 1, 2, \ldots, p, \\
& x \in S. &
\end{aligned}
$$

donde x es un vector de n dimensiones, f, h y g son funciones reales de x y S es un conjunto en el espacio n-dimensional.

Programación lineal y entera

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.a.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

- La función objetivo y las restricciones son funciones lineales de las variables de decisión.
- Marco de modelado y soporte en métodos de resolución.
- En general es la metodología utilizada en este curso para el modelado y la resolución de problemas.

Terminología

Sigla	Nombre	Descripción
LP	Programación Lineal (Linear Programming)	Todas las variables son reales
IP	Programación Entera (Integer Programming)	Todas las variables son enteras
MIP	Programación Entera Mixta (Mixed Integer Programming)	Al menos una variable es entera
LR	Relajación Lineal (Linear Relaxation)	Resultado de eliminar la restricción de integralidad en un MIP

Métodos de resolución (más comunes)

LP

- Simplex y variantes (revisado, dual).
- Punto interior.

MIP

- Branch & Bound.
- Branch & Cut.

Tamaños de problemas

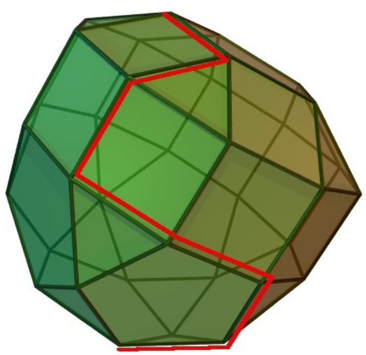
- El tamaño de un problema se mide en términos de la cantidad de variables y la cantidad de restricciones.

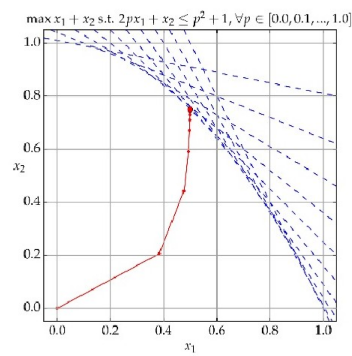
- Actualmente se consideran tres categorías según el tamaño:
  - Pequeña escala: cinco o menos variables, resoluble a mano o por una pequeña computadora.
  - Escala intermedia: cientos o miles de variables, resoluble en una computadora personal con un método de propósito general.
  - Gran escala: miles o millones de variables y restricciones, usualmente requiere software y hardware de propósito específico.

Dificultad computacional de resolución

- Para LP, el método simplex es eficiente en la práctica (obtiene el óptimo en una cantidad polinomial de pasos respecto al tamaño del problema), pero en teoría (peor caso) requiere una cantidad exponencial de pasos.

- Resultados más recientes muestran que en teoría LP puede resolverse en una cantidad polinomial de pasos (método del elipsoide), incluso de una forma computacionalmente eficiente en la práctica (método de punto interior).

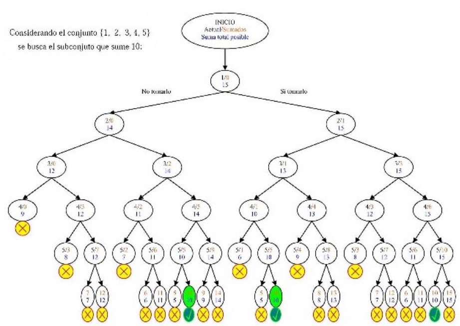


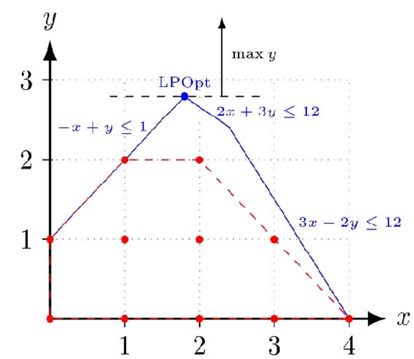


Dificultad computacional de resolución (cont.)

- Para IP y MIP no existe un método general de resolución que encuentre el óptimo en una cantidad polinomial de pasos (sin embargo, parte de las enumeraciones en que se basan los métodos de resolución pueden evitarse con podas y cortes).

- La solución óptima de algunos problemas IP y MIP es la misma que la de sus respectivos LR, por lo tanto son problemas fáciles de resolver computacionalmente.





Algunos problemas seleccionados

- Problema de la mochila.
- Problema de transporte (básico y con costos fijos).
- Ubicación de instalaciones.
- Flujo máximo y camino más corto en redes.

Problema de la mochila

Seleccionar un subconjunto de elementos (a partir de un conjunto de n objetos) maximizando la ganancia y respetando una restricción de peso de la mochila.

$$\begin{aligned}
\max \quad & \sum_i c_i x_i & \\
\text{s.a.} \quad & \sum_i a_i x_i \leq b & \forall\ i = 1, 2, \ldots, n, \\
& x_i \in \{0, 1\} & \forall\ i = 1, 2, \ldots, n.
\end{aligned}$$

donde c_i y a_i son la ganancia y peso respectivamente del objeto i, b es la capacidad de la mochila y x_i es la variable que indica si se incluye o no el objeto i.

El problema es polinomial si los objetos son divisibles (x real).

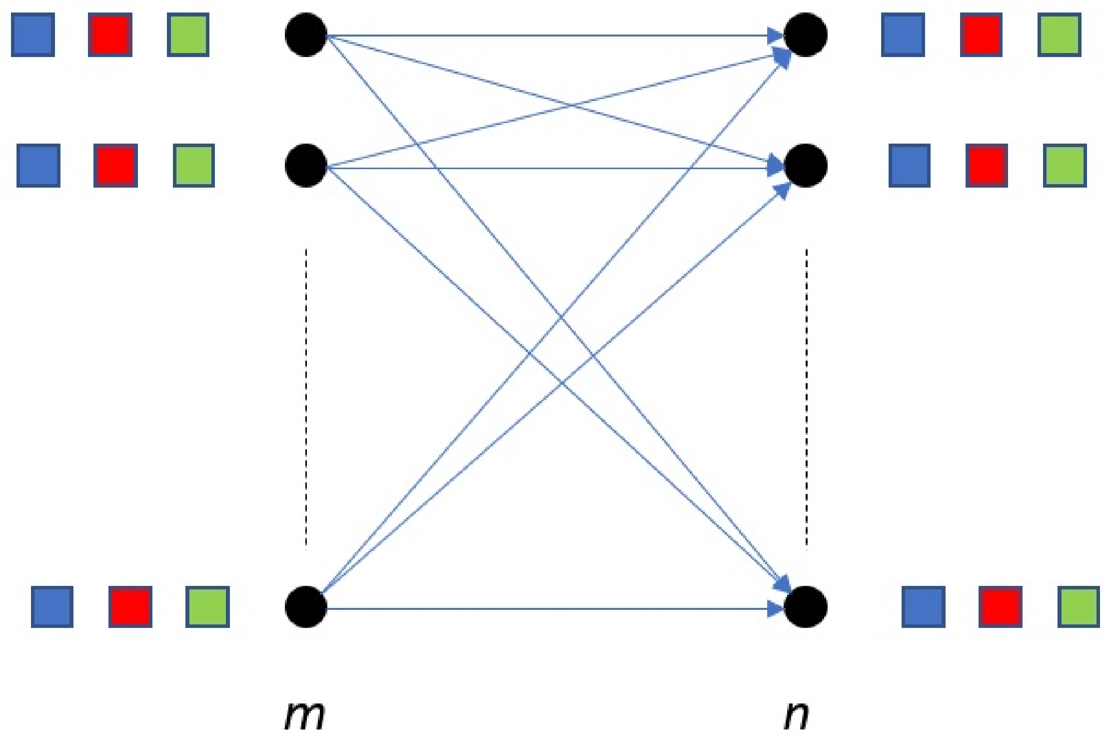
Problema de transporte

Se debe enviar mercadería desde m puntos, la cual debe recibirse en n puntos, minimizando los costos de transporte entre origen y destino.

$$
\begin{aligned}
\min \quad & \sum_{ij} c_{ij} x_{ij} & \\
\text{s.a.} \quad & \sum_{j} x_{ij} = a_i & \forall\ i = 1, 2, \ldots, m, \\
& \sum_{i} x_{ij} = b_j & \forall\ j = 1, 2, \ldots, n, \\
& x_{ij} \geq 0 & \forall\ i = 1, 2, \ldots, m; j = 1, 2, \ldots, n.
\end{aligned}
$$

donde a_i es lo disponible en i, b_j es lo que debe llegar a j, c_{ij} es el costo de transporte desde i hasta j y x_{ij} es la variable que indica cuanta mercadería se envía desde i hasta j.

Problema de transporte



Problema de transporte con costos fijos

Al problema anterior se agrega un costo fijo f_{ij} por habilitar la conexión desde i hacia j, lo que se indica mediante la variable binaria y_{ij}.

$$
\begin{array}{lll}
\min & \sum_{ij}(c_{ij}x_{ij} + f_{ij}y_{ij}) & \\
\text{s.a.} & \sum_{j} x_{ij} = a_i & \forall\ i = 1, 2, \ldots, m, \\
& \sum_{i} x_{ij} = b_j & \forall\ j = 1, 2, \ldots, n, \\
& x_{ij} \leq My_{ij} & \forall\ i = 1, 2, \ldots, m; j = 1, 2, \ldots, n, \\
& x_{ij} \geq 0 & \forall\ i = 1, 2, \ldots, m; j = 1, 2, \ldots, n, \\
& y_{ij} \in \{0, 1\} & \forall\ i = 1, 2, \ldots, m; j = 1, 2, \ldots, n.
\end{array}
$$

donde M es un valor suficientemente grande (puede ser $\sum_j b_j$). El problema es difícil de resolver debido al costo fijo introducido y a la correspondiente variable binaria.

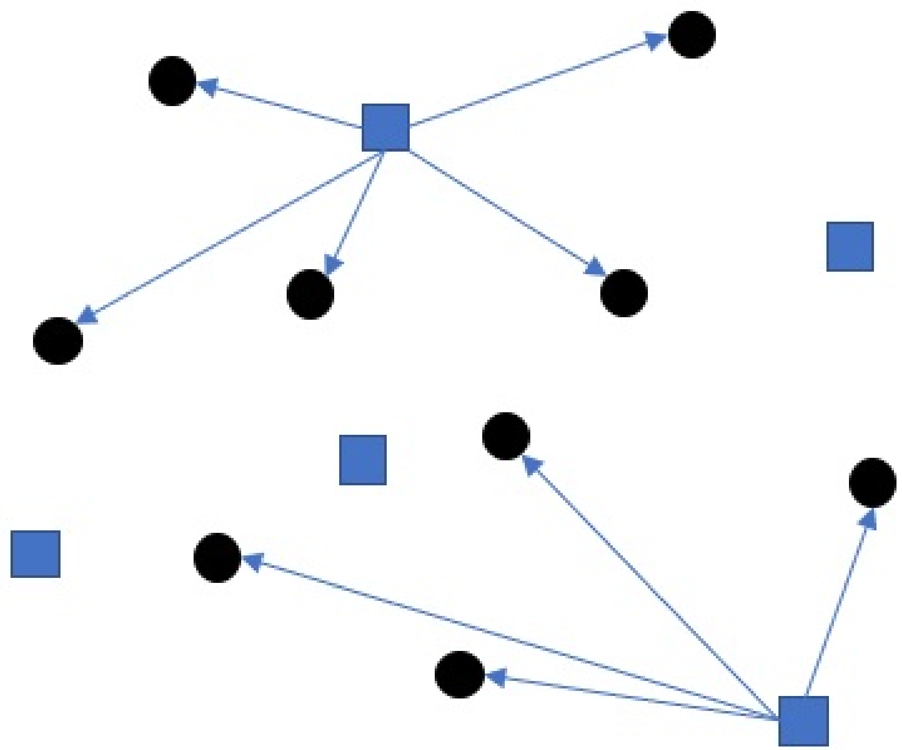
Ubicación de instalaciones

Establecer centros de distribución (a partir de m candidatos) para distribuir mercaderías a un conjunto de n clientes.

$$\begin{aligned}
\min \quad & \sum_{ij} c_{ij} x_{ij} + \sum_i f_i y_i \\
\text{s.a.} \quad & \sum_i x_{ij} = 1 && \forall\, j = 1, 2, \ldots, n, \\
& x_{ij} \leq y_i && \forall\, i = 1, 2, \ldots, m; j = 1, 2, \ldots, n, \\
& x_{ij} \geq 0 && \forall\, i = 1, 2, \ldots, m; j = 1, 2, \ldots, n, \\
& y_i \in \{0, 1\} && \forall\, i = 1, 2, \ldots, m.
\end{aligned}$$

donde f_i es el costo de establecer la instalación candidata i, y_i es una variable binaria que indica si se establece o no dicha instalación y x_{ij} es la cantidad de mercadería distribuida desde la instalación i al cliente j. El problema es difícil por razones similares al anterior.

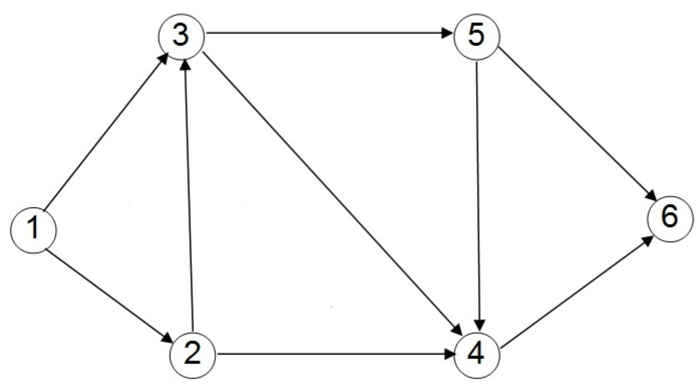
Ubicación de instalaciones



Problemas en redes

Grafo dirigido $G = (N, A)$ con:

- Costos en los arcos, por ejemplo, tiempo de viaje estimado en base a distancia y velocidad promedio observada.
- Capacidades en los arcos, por ejemplo, cantidad de vehículos por unidad de tiempo que pueden atravesar el arco, estimada en base a cantidad de carriles y velocidad máxima permitida.
- Nodos origen y destino.



Flujo máximo en una red

Encontrar el máximo flujo posible desde un nodo origen o a uno destino d a través de una red $G = (N, A)$ con capacidades en los arcos.

$$
\begin{aligned}
\max \quad & v \\
\text{s.a.} \quad & \sum_{(i,j)\in A} x_{ij} - \sum_{(j,i)\in A} x_{ji} = \delta_i & \forall\ i \in N, \\
& 0 \le x_{ij} \le u_{ij} & \forall\ (i,j) \in A.
\end{aligned}
$$

donde (i, j) denota el arco en A que va desde el nodo i al j (ambos $\in N$), u_{ij} es la capacidad del arco (i, j) y δ_i es igual a v si $i = o$, igual a $-v$ si $i = d$ y es 0 en otro caso. Las variables v y x_{ij} denotan el flujo total en la red y en el arco (i, j) respectivamente.

Bajo ciertas hipótesis, el problema se resuelve en tiempo polinomial mediante algoritmos específicos (por ejemplo, Ford & Fulkerson).

Camino de costo mínimo en una red

Encontrar el camino de costo mínimo desde un nodo origen o a todos los demás en una red $G = (N, A)$ con costos en los arcos.

$$
\begin{aligned}
\min \quad & \sum_{ij} c_{ij} x_{ij} & \\
\text{s.a.} \quad & \sum_{(i,j)\in A} x_{ij} - \sum_{(j,i)\in A} x_{ji} = \theta_i & \forall\ i \in N, \\
& x_{ij} \geq 0 & \forall\ (i,j) \in A.
\end{aligned}
$$

donde c_{ij} es el costo del arco (i, j), $n = |N|$, y θ_i es igual a $n - 1$ si $i = o$ y es -1 en otro caso. La variable x_{ij} denota el flujo en el arco (i, j).

Bajo ciertas hipótesis, el problema se resuelve en tiempo polinomial mediante algoritmos específicos (por ejemplo, Dijkstra).

Bibliografía

- Ahuja, RK; Magnanti, TL; Orlin JB (1993) Network Flows. Prentice-Hall.
- Chen, D-S; Batson, RG; Dang, Y (2010) Applied Integer Programming. Wiley.
- Luenberger, DG; Ye, Y (2016) Linear and Nonlinear Programming. International Series in Operations Research & Management Science 228. Springer International Publishing. DOI 10.1007/978-3-319-18842-3.
- Wikipedia, imágenes tomadas de:

  https://en.wikipedia.org/wiki/Simplex_algorithm
  https://en.wikipedia.org/wiki/Interior-point_method
  https://en.wikipedia.org/wiki/Integer_programming
  https://es.wikipedia.org/wiki/Ramificaci%C3%B3n_y_poda

- Notas del docente del curso.