Sección de Matemáticas
Universidad de La Laguna

Helena Gil Rodríguez

Variantes del Método Simplex para Redes

Variants of the Network Simplex Method

Trabajo Fin de Grado
Grado en Matemáticas
La Laguna, Junio de 2018

Dirigido por
Carlos González Martín

Carlos González Martín
Departamento de Matemáticas, Estadística e Investigación Operativa
Universidad de La Laguna
38200 La Laguna, Tenerife

Agradecimientos

En agradecimiento a toda mi familia, amigos, compañeros de universidad y mi tutor. Sin ellos no sería posible llegar hasta este punto de la carrera: su fin.

Helena Gil Rodíguez
La Laguna, 8 de julio de 2019

Resumen · Abstract

Resumen

Este trabajo está dedicado al estudio y resolución de problemas importantes de Análisis de Redes: los problemas de flujo de costo mínimo sobre redes normalizadas. Concretamente, se afronta la resolución de dichos problemas usando algunas variantes del Método del Simplex.

El estudio se completa con la aplicación de herramientas computacionales básicas en la resolución de los ejemplos presentados previamente.

Palabras clave: *Programación lineal, Método Simplex, Dualidad, Problemas de Flujo de Costo Mínimo, Análisis de Redes.*

Abstract

This work is dedicated to the study and resolution of important problems of Network Analysis: the problems of minimum cost flow on standardized networks. Specifically, these problems are addressed by using some variants of the Simplex Method.

This study is completed with the application of basic computational tools in the resolution of the previously presented examples.

Keywords: *Lineal Programming, Simplex Method, Duality, Minimum Cost Flow Problems, Network Analysis.*

Contenido

Introducción

La Investigación Operativa (ver, por ejemplo, [1], [5], [6]) tiene como objeto fundamental " ayudar a tomar decisiones con aval científico ".

Habitualmente, se apoya en modelos matemáticos en los que aparecen problemas de Optimización.

El origen de la Investigación Operativa se sitúa al comienzo de la Segunda Guerra Mundial, en 1939, cuando a diversos grupos de científicos le encargaron diseñar formas de decidir sobre distintos problemas de índole militar.

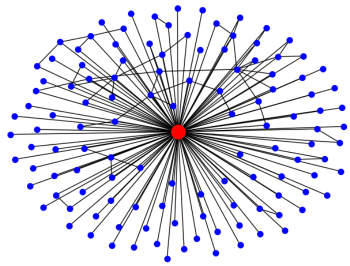
En 1947, G. Dantzing, matemático y estadístico, introdujo el Método del Simplex, y con ello comienza la Programación Lineal, una de las áreas más importantes de la I.O.

La Programación Lineal estudia y resuelve problemas de optimización lineal: se minimiza o maximiza una función lineal (función objetivo) en un contexto definido por ecuaciones o inecuaciones lineales (región factible).

Los problemas sobre redes son muy importantes en muchas de las áreas de actuación de la sociedad actual. Hay redes de comunicación, transporte, abastecimiento, distribución, etc, que son esenciales y que exigen funcionamientos con atributos de optimalidad.

Los problemas de redes de flujo constituyen un grupo importante en el que destacan los problemas de flujo de costo mínimo sobre una red dirigida normalizada.

De este modelo general, derivan casos importantes como el Problema de Transporte, el de Asignación, el de caminos mínimos, etc. Su estudio y resolución, desde la perspectiva de la Programación Lineal, constituye el interés del presente trabajo.



El capítulo 1 comienza introduciendo los fundamentos de la Programación Lineal, la dinámica del Método del Simplex primal, la dualidad, el Método Simplex Dual, ...

En el capítulo 2, veremos cómo resolver los problemas de flujo de costo mínimo mediante el Método Simplex Primal. Este consiste en determinar el patrón óptimo de flujo que circula por una red normalizada, respetando las disponibilidades de los vértices y las capacidades de los arcos.

En el capítulo 3, veremos cómo resolver los problemas de flujo de costo mínimo para otras variantes del Método Simplex Primal: el Método Simplex Dual y el Método Auto-Dual.

En el capítulo 4 resolveremos, mediante herramientas computacionales, los ejemplos propuestos en los capítulos 2 y 3.

1

Fundamentos de la Programación Lineal

Los problemas de Programación Lineal consisten en la optimización de una función lineal en un contexto determinado por un sistema de ecuaciones o inecuaciones lineales.

Un tipo particular de estos, los problemas de optimización sobre redes de flujo, centran el interés del presente trabajo desde la perspectiva de su resolución a través de la aplicación de la metodología de la Programación Lineal. Necesitaremos entonces, hacer un recorrido esquemático por los conceptos y propiedades básicos que permiten desarrollar el Método del Simplex y alguna de sus variantes, con el propósito de adaptarlos y aplicarlos posteriormente a problemas de flujos sobre redes.

1.1. Forma estándar

La forma estándar de un problema de P.L. es:

$$\begin{aligned} &min\ c^t x \\ &s.a.: Ax = b \\ &\quad x \geq 0 \end{aligned} \tag{1.1}$$

donde $c \in \mathbb{R}^n$ el vector de los costos, x el vector de variables de decisión, A la matriz de coeficientes tecnológicos de orden $m \times n$ y de rango igual a m, y $b \in \mathbb{R}^m$ el vector de los recursos.

La forma estándar de un problema de P.L. con variables acotadas es:

$$\begin{aligned} &min\ c^t x \\ &s.a.: Ax = b \\ &\quad l \leq x \leq u \end{aligned} \tag{1.2}$$

donde l y u son los vectores de cotas inferiores y superiores respectivamente.

1.2. Ejemplos

Ejemplo 1.1. **Problema de producción.**

Una frutería comercializa 3 tipos de manzanas: Fuji, Golden y Royal. Antes de comercializarlas deben pasar por dos revisiones de control de calidad. Los tiempos, en minutos por kilogramo, empleados en ambas revisiones, M1 y M2; los precios de comercialización de cada una de ellas y las horas diarias disponibles, vienen especificados en la siguiente tabla:

	Fuji	Golden	Royal	Minutos disponibles
M1	13	19	15	480
M2	5	3	4	180
Precio total(€/ Kg)	1,25	1,80	1,53	

Sean x_1 los kilogramos de manzanas tipo Fuji, x_2 los kg. de las Golden y x_3 los kg. de las Royal. Tenemos que:

$$\begin{aligned} &min\ 1,25x_1 + 1,80x_2 + 1,53x_3 \\ &s.a.: 13x_1 + 19x_2 + 15x_3 = 480 \\ &\quad 5x_1 + 3x_2 + 4x_3 = 180 \\ &\quad x_i \geq 0, i \in \{1,2,3\} \end{aligned} \tag{1.3}$$

Ejemplo 1.2. **Problema de transporte.**

Aliexpress y Amazon ofertan una cámara fotográfica a tres tiendas. El envío a cada una de estas va variando como aparece en la tabla. Mediamark solicita 250 cámaras a la semana, Worten 207 y El Corte Inglés 323. Por otro lado, Aliexpress oferta 380 a la semana y Amazon 400.

	Mediamarkt	Worten	Corte Inglés	Oferta
Aliexpress	5€	6€	8€	380
Amazon	3€	4€	5€	400
Demanda	250	207	323	

Queremos minimizar los costos globales, atiendiendo las demandas y las ofertas establecidas.

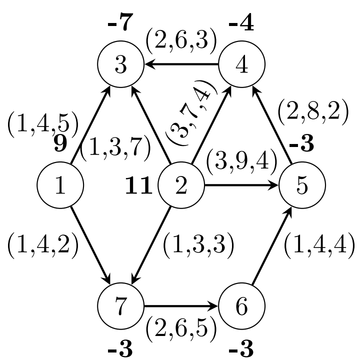
Sea x_{ij} el número de cámaras que se transportan de una empresa i a una tienda j , con $i \in \{$ Aliexpress, Amazon $\}$ y $j \in \{$ Mediamark,Worten,El Corte Inglés $\}$.

El problema de P.L. se nos quedaría de la forma:

$$
\begin{aligned}
&min\ 5x_{11} + 6x_{12} + 8x_{13} + 3x_{21} + 4x_{22} + 5x_{23} \\
&s.a.: x_{11} + x_{12} + x_{13} = 380 \\
&\quad x_{21} + x_{22} + x_{23} = 400 \\
&\quad x_{11} + x_{21} = 250 \\
&\quad x_{12} + x_{22} = 207 \\
&\quad x_{13} + x_{23} = 323 \\
&\quad x_{i,j} \geq 0, i \in \{1,2\}, j \in \{1,2,3\}
\end{aligned}
\tag{1.4}
$$

Ejemplo 1.3. **Problema de flujo de coste mínimo.** Dos almacenes de fruta reparten cada día a cinco mercados. Los camiones que van de los almacenes a los mercados, dependiendo de la ruta, tienen una capacidad mínima, una capacidad máxima de carga y un costo de transporte: (l_{ij}, u_{ij}, c_{ij}). Por otro lado, es obvio que los almacenes no pueden repartir más kilos de los que tienen y los mercados cómo máximo demandan una serie de kilos al día. El problema que se plantea, es qué ruta deben de llevar los camiones para que el coste de transporte sea el mínimo.

En el siguiente grafo vemos las rutas que pueden realizar y el número de cajas de frutas mínimo y máximo que pueden llevar por esa ruta y los costos que conlleva cada ruta (en euros):



El problema asociado al problema es:

$$
\begin{aligned}
&min\ 5x_{13}+2x_{17}+7x_{23}+4x_{24}+4x_{25}+3x_{27}+3x_{43}+2x_{54}+4x_{65}+5x_{76}\\
&s.a.: x_{13}+x_{17}=9\\
&\quad x_{23}+x_{24}+x_{25}+x_{27}=11\\
&\quad -x_{13}-x_{23}-x_{43}=-7\\
&\quad -x_{24}-x_{54}+x_{43}=-4\\
&\quad -x_{25}+x_{54}-x_{65}=-3\\
&\quad x_{65}-x_{76}=-3\\
&\quad x_{76}-x_{17}-x_{27}=-3\\
&\quad 1\leq x_{13}\leq 4, 1\leq x_{17}\leq 4, 1\leq x_{23}\leq 3, 3\leq x_{24}\leq 7\\
&\quad 3\leq x_{25}\leq 9, 1\leq x_{27}\leq 3, 2\leq x_{43}\leq 6, 2\leq x_{54}\leq 8\\
&\quad 1\leq x_{65}\leq 4, 2\leq x_{76}\leq 6
\end{aligned}
\tag{1.5}
$$

Los tres ejemplos anteriores son problemas de Programación Lineal. Conviene entonces, hacer un estudio general de su resolución.

1.3. Método Simplex Primal

Dado el problema estándar:

$$
\begin{aligned}
&min\ c^t x\\
&s.a.: Ax=b\\
&\quad x\geq 0
\end{aligned}
\tag{1.6}
$$

La **región factible** es $S=\{x\in\mathbb{R}^n/Ax=b, x\geq 0\}$.
Supongamos que la matriz A de coeficientes, de orden $m\times n$, se puede escribir como A=(B,N), donde B es una matriz cuadrada invertible de orden m y N matriz de orden $m\times(n-m)$.

1.3.1. Conceptos y propiedades

Definición 1.4. *$x=\binom{x_B}{x_N}\in S$ es* ***solución básica*** *si $x_B=B^{-1}b$ (variables básicas) y $x_N=0$ (variables no básicas). Una* ***solución básica factible (primal)*** *es una solución básica con $x_B\geq 0$.*

Propiedad 1.1 *El número de soluciones básicas factibles de un problema de P.L. es finito y está acotado por el número combinatorio $\binom{n}{m}$, donde n es el número de variables y m es el número de restricciones.*

Definición 1.5. *El vector de* ***costos relativos*** *es $\overline{c}^t=c^t-c_B^tB^{-1}A$, donde c_B^t es el vector de costos de las variables básicas. Es evidente que $\overline{c}_B=0$.*

Definición 1.6. *Sea $\overline{x}$ una solución básica factible. Como $\forall x \in S$, $c^t x = c^t\overline{x} + \overline{c}_N^t x_N$, la condición de optimalidad nos dice que si $\overline{c}_N \geq 0$, $\overline{x}$ es la **solución básica factible óptima**.*

Propiedad 1.2 *Si existe, el óptimo para un problema de P.L. es alcanzado en una solución básica factible. En otro caso, salvo que sea no factible, el problema planteado sería no acotado.*

1.3.2. Algoritmo del Simplex Primal

El método Simplex Primal parte de una solución básica factible inicial, y hasta que no se detecta la no acotación o la optimalidad, tenemos que encontrar una solución básica factible mejor.

Cuando la solución básica factible $\overline{x}$ no es óptima, $\exists s \in N$ tal que $\overline{c}_s < 0$. Para mejorar el valor objetivo de $\overline{x}$, se construye, en este caso, $\overline{x}+\lambda d^s$, con $\lambda \geq 0$ y $d^s = \binom{-B^{-1}a^s}{e^s}$ siendo a^s la columna de A asociada a x_s, y e^s el vector de $\mathbb{R}^{n-m}$ que tiene un uno en el lugar s y ceros en el resto. $c^t\overline{x} + \lambda d^s = c^t\overline{x} + \lambda\overline{c}_s$.

- Si $-B^{-1}a^s \leq 0$, $\overline{x} + \lambda d^s$ es solución factible $\forall\lambda \geq 0$ y, por tanto, λ puede crecer indefinidamene y se detecta la no acotación del problema.
- Si $y^s = -B^{-1}a^s \nleq 0$, λ puede variar en $[0,\overline{\lambda}]$, con $\overline{\lambda} = min\{\frac{\overline{b}_i}{y_{is}}/y_{is} > 0\}$, siendo $\overline{b} = B^{-1}b$.

El punto $\overline{x} + \overline{\lambda}d^s$ es una nueva solución básica factible.
Si $\overline{\lambda} = \frac{\overline{b}_r}{y_{rs}}$, quitando x_r de la base e incluyendo a x_s como variable básica, se determina la nueva solución básica factible.

Es conocido que el trabajo con el Método del Simplex se ve facilitado, a nivel docente, si se usa una tabla como la siguiente:

V.B.	x_B	x_N	-z	ctes
x_B	I	$B^{-1}N$	0	$B^{-1}b$
-z	0	$\overline{c}_N^t$	1	-z

que consiste en representar un problema de P.L. como un sistema de ecuaciones (ver, por ejemplo, [1].

Ejemplo 1.7. Resolver el problema:

$$
\begin{aligned}
& min\ x_1 + 2x_2 - 4x_3 \\
& s.a. : 2x_1 + x_2 \leq 3 \\
& \quad - 4x_1 + x_3 = 7 \\
& \quad x_i \geq 0, i \in \{1, 2, 3\}.
\end{aligned} \tag{1.7}
$$

Para comenzar, pasamos el problema de P.L. a la forma estándar añadiendo una variable de holgura x_4:

$$
\begin{aligned}
& min\ x_1 + 2x_2 - 4x_3 \\
& s.a. : 2x_1 + x_2 + x_4 = 3 \\
& \quad - 4x_1 + x_3 = 7 \\
& \quad x_i \geq 0, i \in \{1, 2, 3, 4, \}
\end{aligned} \tag{1.8}
$$

A continuación construimos la tabla con los datos:

V.B.	x_1	x_2	x_3	x_4	-z	Cte
x_4	2	1	0	1	0	3
x_3	-4	0	1	0	0	7
-z	1	2	-4	0	1	0

En la segunda y tercera fila tenemos los coeficientes de las dos ecuaciones de la región factible, y en la cuarta los coeficientes de la ecuación a minimizar.

Si en la tabla anterior, a la cuarta fila le sumamos la tercera multiplicada por 4, obtenemos:

V.B.	x_1	x_2	x_3	x_4	-z	Cte
x_4	2	1	0	1	0	3
x_3	-4	0	1	0	0	7
-z	-15	2	0	0	1	28

La variable no básica x_1, con costo relativo -15, es candidata a ser básica y esta sustituye a x_4 que pasa a ser no básica. Trasformamos la tabla anterior cambiando la variable básica por la nueva, y se queda de la forma:

Como todos los $c_N \geq 0$, tenemos que nuestra tabla es óptima con el valor objetivo óptimo z= $\frac{-101}{2}$ y con $\overline{x} = (\frac{3}{2}\ 0\ 13\ )$

1.3.3. Método de Penalización

Cuando inicialmente no podemos identificar una solución básica factible con $B = I$, forzamos su aparición añadiendo variables artificiales que, luego,

V.B.	x_1	x_2	x_3	x_4	-z	Cte
x_1	1	$\frac{1}{2}$	0	$\frac{1}{2}$	0	$\frac{3}{2}$
x_3	0	2	1	2	0	13
-z	0	$\frac{19}{2}$	0	$\frac{15}{2}$	1	$\frac{101}{2}$

hemos de eliminar. Para hacer este proceso, podemos utilizar herrramientas auxiliares como el Método de Penalización o el Método de las Dos Fases.
El Método de Penalización consiste en añadir tantas variables artificiales como sean necesarias, añadiéndole un costo M en caso de problema mínimo y -M en caso de problema de máximo, con M un número suficientemente grande.

Ejemplo 1.8. Veamos un ejemplo donde la solución básica factible inicial no se ve directamente.

$$\begin{aligned} &min\ 4x_1 + x_2 \\ &s.a. : 4x_1 + 3x_2 = 6 \\ &\quad x_1 + 2x_2 = 4 \\ &\quad x_i \geq 0, i \in \{1, 2\} \end{aligned} \tag{1.9}$$

Añadiendole las variables artificiales x_3, x_4, quedaría:

$$\begin{aligned} &min\ 4x_1 + x_2 + Mx_3 + Mx_4 \\ &s.a. : 4x_1 + 3x_2 + x_3 = 6 \\ &\quad x_1 + 2x_2 + x_4 = 4 \\ &\quad x_i \geq 0, i \in \{1, 2, 3, 4\} \end{aligned} \tag{1.10}$$

Y aplicamos el Método Simplex Primal con x_3, x_4 como las variables básicas factibles iniciales.

1.4. Dualidad

Dado el problema primal (P):

$$\begin{aligned} &min\ c^t x \\ &s.a. : Ax = b \\ &\quad x \geq 0, \end{aligned} \tag{1.11}$$

se define el dual (D) como:

$$\begin{aligned} &max\ b^t y \\ &s.a. : A^t y \leq c \end{aligned} \tag{1.12}$$

1.4.1. Conceptos y propiedades

Propiedad 1.3 *(Teorema débil de la dualidad)*
1. Cualquier solución factible de (D) define una cota inferior del valor óptimo de (P).
2. Cualquier solución factible de (P) define una cota superior del valor óptimo de (D).
3. Si (P) es no acotado, entonces (D) es no factible.
4. Si (D) es no acotado, entonces (P) es no factible.

Propiedad 1.4 *Si $\overline{x}$ es una solución factible primal e $\overline{y}$ es una solución factible dual tales que $c^t\overline{x} = b^t\overline{y}$, entonces $\overline{x}$ es solución óptima de (P) e $\overline{y}$ es solución óptima de (D).*

Propiedad 1.5 *(Teorema fuerte de dualidad)*
1. Si (P) tiene solución óptima, entonces (D) tiene solución óptima y los valores óptimos de ambos problemas coiciden.
2. Si (D) tiene solución óptima, entonces (P) tiene solución óptima y los valores óptimos de ambos problemas coiciden.

Propiedad 1.6 *1. Si (P) es factible y (D) es no factible, (P) es no acotado.*
2. Si (D) es factible y (P) es no factible, (D) es no acotado.

Propiedad 1.7 *(Condición de holguras complementarias)*
Sean $\overline{x}$ solución básica factible de (P) e $\overline{y}$ solución factible de (D), $\overline{x}$ es solución óptima de (D) si, y solo si, $(c - A^t\overline{y})^t\overline{x} = 0$.

Definición 1.9. *Para el problema (P), diremos que una solución básica es **factible dual** si, y solo si, $\overline{x}$ es una solución básica y $\overline{c}^t_N = c^t_N - c^t_B B^{-1}N \geq 0$*

Propiedad 1.8 *Si $\overline{x}=\binom{\overline{x}_B}{\overline{x}_N} = \binom{B^{-1}b}{0}$ es una solución básica, **los costos relativos** se pueden obtener a partir de $c - A^t\overline{y}$, con $\overline{y} = c^t_B B^{-1}$.*

Propiedad 1.9 *Una solución básica es óptima si, y sólo si, es solución básica factible primal y solución básica factible dual.*

1.4.2. Método Simplex Dual

El Método del Simplex presentado anteriormente trabaja, en cada iteración, con soluciones básicas factibles primales hasta detectar la no acotación o la optimalidad.

Una alternativa a esta forma de proceder podría consistir en trabajar, en cada iteración, con soluciones básicas factibles duales hasta detectar la no factiblidad o la optimalidad.
Esta es, precisamente, la estructura del Método Simplex Dual.

Ejemplo 1.10. Resolver con el Método Simplex Dual:

$$\begin{aligned} &min\ 4x_1 + 3x_2 + x_3 \\ &s.a.: -6x_1 - 2x_2 + x_3 + x_4 = -10 \\ &\quad -4x_1 + x_2 - 2x_3 + x_5 = -12 \\ &\quad x_i \geq 0, i \in \{1,2,3,4,5\} \end{aligned} \tag{1.13}$$

Una solución básica factible dual inicial sería x_B=(-10 -12). Por tanto, la tabla se nos queda de la forma:

V.B.	x_1	x_2	x_3	x_4	x_5	-z	Ctes
x_4	-6	-2	1	1	0	0	-10
x_5	-4	1	-2	0	1	0	-12
-z	4	3	1	0	0	1	0

Tenemos que $x_5 = -12$ debe salir de la base, pues toma el valor más negativo. Por otro lado, x_3 es la nueva variable básica, ya que $max\{\frac{\bar{c}_j}{y_{rj}}/y_{rj} < 0\}$ $= max\{\frac{4}{-4}, \frac{1}{-2}\} = \frac{1}{-2}$. Y su tabla asociada es:

V.B.	x_1	x_2	x_3	x_4	x_5	-z	Ctes
x_4	-8	-$\frac{3}{2}$	0	1	$\frac{1}{2}$	0	-16
x_3	2	-$\frac{1}{2}$	1	0	-$\frac{1}{2}$	0	6
-z	2	$\frac{7}{2}$	0	0	$\frac{1}{2}$	1	-6

Ahora vemos que $x_4 = -16$ debe salir de la base y x_1 es la nueva variable básica, ya que $max\{\frac{2}{-8}, \frac{7}{-3}\} = \frac{2}{-8}$.

Así tendríamos:

V.B.	x_1	x_2	x_3	x_4	x_5	-z	Ctes
x_1	1	$\frac{3}{26}$	0	-$\frac{1}{8}$	-$\frac{1}{16}$	0	2
x_3	0	-$\frac{7}{8}$	1	$\frac{1}{4}$	-$\frac{3}{8}$	0	2
-z	0	$\frac{25}{8}$	1	$\frac{1}{4}$	$\frac{5}{8}$	1	-10

Y esta solución es básica factible dual y primal, por tanto, es óptima.

1.5. Problemas de P.L. con variables acotadas

Los problemas con variables acotadas son de la forma:

$$\begin{aligned} &min\ c^t x \\ &s.a. : Ax = b \\ &\quad l \leq x \leq u \end{aligned} \tag{1.14}$$

donde la región factible es $S = \{x \in \mathbb{R}^n / Ax = b, l \leq x \leq u\}$.

Los distintos conceptos ya introducidos, presentan ahora definiciones generalizadas:

1.5.1. Conceptos y propiedades

Definición 1.11. *Sea $A = (B, N_1, N_2)$, $x \in S$ es **solución básica** si $x_{N_1} = l_{N_1}$, $x_{N_2} = l_{N_2}$ y $x_B = B^{-1}b - B^{-1}N_1 l_{N_1} - B^{-1}N_2 u_{N_2} = \widehat{b}$.*

Definición 1.12. *x es una solución básica **factible primal** si x es solución básica con $l_B \leq x_B \leq u_B$.*

Definición 1.13. *x es una solución básica **factible dual** si x es solución básica con $\overline{c}_{N_1} \geq$ y $\overline{c}_{N_2} \leq 0$.*

Definición 1.14. *x es una **solución óptima** si x es solución básica factible primal y factible dual.*

Propiedad 1.10 *El número de soluciones básicas es, como máximo, $2^{n-m}\binom{n}{m}$.*

Propiedad 1.11 *Si existe, el óptimo se alcanza en una solución básica factible primal y dual.*

1.5.2. Método Simplex para variables acotadas

Dada una solución factible x y una solución básica factible primal $\overline{x}$, se tiene en un problema de mínimo que $c^t x = c_B^t \overline{x} + \overline{c}_{N_1}^t y_{N_1} - \overline{c}_{N_2}^t y_{N_2}$, siendo $x_{N_1} = l_{N_1} + y_{N_1}$ y $x_{N_1} = u_{N_2} - y_{N_2}$. Así, la solución $\overline{x}$ será optima cuando $\overline{c}_{N_1} \geq 0$ y $\overline{c}_{N_1} \leq 0$.

Si una solución básica factible no es óptima sucede que $\exists s \in N_1 / \overline{c}_s < 0$ o $\exists s \in N_2 / \overline{c}_s > 0$. Es decir, se puede rebajar el valor objetivo incrementando una variable que está en su cota inferior o disminuyendo una que está en su cota superior.

Sea $s = argmin\{\{\overline{c}_j / j \in N_1, \overline{c}_j < 0\} \cup \{-\overline{c}_j / j \in N_2, \overline{c}_j > 0\}\}$, dependiendo de si $j \in N_1$ o $j \in N_2$, debemos hacer distintos pasos.

- Si $s \in N_1$, $l \leq \begin{pmatrix} \widehat{b} \\ l_{N_1} \\ u_{N_2} \end{pmatrix} + \lambda \begin{pmatrix} -y^s \\ e^s \\ 0 \end{pmatrix} \leq u \Rightarrow l_B \leq \widehat{b} - \lambda y^s \leq u_B$ y $l_s + \lambda \leq u_s$.

Entonces, definimos:
a) $\lambda_1 = min\{\frac{\widehat{b} - l_{B_i}}{y_{is}} / y_{is} > 0\}$ si $y^s \not\leq 0$. Si $y^s \leq 0$, $\lambda_1 = \infty$.
b) $\lambda_2 = min\{\frac{u_{B_i} - \widehat{b}}{-y_{is}} / y_{is} < 0\}$ si $y^s \not\geq 0$. Si $y^s \geq 0$, $\lambda_2 = \infty$.
c) $\lambda_3 = u_s - l_s$.

Sea $\overline{\lambda} = min\{\lambda_1, \lambda_2, \lambda_3\}$, veamos qué ocurre dependiendo del valor de $\overline{\lambda}$:
- Si $\overline{\lambda} = \infty$, entonces el problema es no acotado.
- Si $\overline{\lambda} = \lambda_1$, entonces x_s se hace básica y λ_1 nos da la variable que deja de ser básica y va a la cota inferior.
- Si $\overline{\lambda} = \lambda_2$, entonces x_s se hace básica y λ_2 nos da la variable que deja de ser básica y va a la cota superior.
- Si $\overline{\lambda} = \lambda_3$, entonces x_s pasa de su cota inferior a su cota superior.

- Si $s \in N_2$, $l \leq \begin{pmatrix} \widehat{b} \\ l_{N_1} \\ u_{N_2} \end{pmatrix}$ - $\lambda \begin{pmatrix} -y^s \\ 0 \\ e^s \end{pmatrix} \leq u \Rightarrow l_B \leq \widehat{b} + \lambda y^s \leq u_B$ y $l_s \leq u_s - \lambda$.

Entonces, tenemos que:
a) $\lambda_1 = min\{\frac{\widehat{b} - l_{B_i}}{-y_{is}} / y_{is} < 0\}$ si $y^s \not\geq 0$. Si $y^s \geq 0$, $\lambda_1 = \infty$.
b) $\lambda_2 = min\{\frac{u_{B_i} - \widehat{b}}{y_{is}} / y_{is} > 0\}$ si $y^s \not\leq 0$, $\lambda_2 = \infty$.
c) $\lambda_3 = u_s - l_s$

Sea $\overline{\lambda} = min\{\lambda_1, \lambda_2, \lambda_3\}$, veamos qué ocurre dependiendo del valor de $\overline{\lambda}$:
- Si $\overline{\lambda} = \infty$, entonces el problema es no acotado.
- Si $\overline{\lambda} = \lambda_1$, entonces x_s se hace básica y λ_1 nos da la variable que deja de ser básica y va a la cota inferior.
- Si $\overline{\lambda} = \lambda_2$, entonces x_s se hace básica y λ_2 nos da la variable que deja de ser básica y va a la cota superior.
- Si $\overline{\lambda} = \lambda_3$, entonces x_s pasa de su cota superior a su cota inferior.

Ejemplo 1.15.

$$\begin{aligned} & min\ 3x_1 + 2x_2 - 4x_3 \\ & s.a.: 2x_1 - x_2 + 2x_3 + x_4 = 12 \\ & \quad - x_1 + 2x_2 - x_3 + x_5 = 10 \\ & \quad 1 \leq x_1 \leq 5, 0 \leq x_2 \leq 6, 1 \leq x_3 \leq 6 \\ & \quad 0 \leq x_4, x_5 \end{aligned} \tag{1.15}$$

Tomemos por ejemplo $B = \{4,5\}$, $N_1 = \{2,3\}$ y $N_2 = \{1\}$. La solución básica factible inicial es: $x_1 = 5$, $x_2 = 0$, $x_3 = 1$, $x_4 = 0$, $x_5 = 16$, así la tabla queda:

	5	0	1	-	-			
V.B	x_1	x_2	x_3	x_4	x_5	-z	C.T	Ctes
x_4	2	-1	2	1	0	0	12	0
x_5	-1	2	-1	0	1	0	10	16
-z	3	2	-4	0	0	1	0	-11

Tenemos que $s = 3 \in N_1$, y $min\{\lambda_1, \lambda_2, \lambda_3\} = \lambda_1 = 0$. Luego x_3 pasa a ser básica y x_4 sale de la base con su cota inferior, por lo que tenemos que $B = \{3,5\}$, $N_1 = \{2,4\}$ y $N_2 = \{1\}$:

	5	0	-	0	-			
V.B	x_1	x_2	x_3	x_4	x_5	-z	C.T	Ctes
x_3	1	$-\frac{1}{2}$	1	1	0	0	6	1
x_5	0	$\frac{3}{2}$	0	1	1	0	16	16
-z	7	0	0	2	0	1	24	-11

Ahora, $s = 1 \in N_2$, y $min\{\lambda_1, \lambda_2, \lambda_3\} = \lambda_3 = 4$. Luego x_1 pasa a la cota inferior, y entonces: $B = \{3,5\}$, $N_1 = \{1,2,4\}$. Ajustando la tabla, se nos queda que la solución es óptima.

	1	0	-	0	-			
V.B	x_1	x_2	x_3	x_4	x_5	-z	C.T	Ctes
x_3	1	$-\frac{1}{2}$	1	1	0	0	6	5
x_5	0	$\frac{3}{2}$	0	1	1	0	16	16
-z	7	0	0	2	0	1	24	17

2

Problemas de flujos de costo mínimo. El Método Simplex Primal para redes

2.1. Problemas de flujos de costo mínimo

Consideramos la red dirigida conexa $R = (V, A)$, donde V es el conjunto de n vértices V=$\{1, 2, ..., n\}$ y $A \subset V \times V$ es el conjunto de m arcos.
Supongamos que $\forall i \in V$, b_i es la disponiblidad de flujo en i. Si $b_i > 0$, i es un vértice de oferta; si $b_i < 0$, i es un vértice de demanda; y si $b_i = 0$, i es un vértice de transbordo.
$\forall (i, j) \in A$, c_{ij} es el costo por unidad de flujo que circula por el arco (i, j).
$\forall (i, j) \in A$, l_{ij} y u_{ij} son las capacidades mínima y máxima, respectivamente, de flujo que pueden circular a través de cada arco.
Si $\forall (i, j) \in A$, x_{ij}, es el flujo que circula de i a j, entonces, el problema de flujos de costo mínimo se puede plantear como:

$$\begin{aligned} & min \sum_{i \in V} \sum_{j \in Suc(i)} c_{ij} x_{ij} \\ & s.a.: \sum_{j \in Suc(i)} x_{ij} - \sum_{j \in Pred(i)} x_{ji} = b_i, \forall i \in V \\ & \quad l_{ij} \le x_{ij} \le u_{ij}, \forall (i, j) \in A \end{aligned} \tag{2.1}$$

Con $Suc(i) = \{j \in V/(i, j) \in A\}$, $Pred(i) = \{j \in V/(j, i) \in A\}$.
Este problema se puede poner en forma matricial como:

$$\begin{aligned} & min\ c^t x \\ & s.a.: Mx = b \\ & \quad l \le x \le u \end{aligned} \tag{2.2}$$

siendo M una matriz de orden $n \times m$, $b \in \mathbb{R}^n$, l, x, $u \in \mathbb{R}^m$.

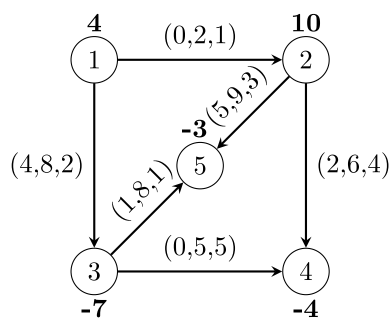
Usemos la representación gráfica:

Al lado de cada vértice escribimos su disponibilidad y, al lado de cada arco, las capacidades mínima y máxima y el costo por unidad de flujo, respectivamente. En el caso sin capacidades, $l_{ij} = 0$ y $u_{ij} = \infty$.

Ejemplo 2.1. Un ejemplo de problema de flujo de costo mínimo con capacidades sería:

$$\begin{aligned} &min\ x_{12} + 2x_{13} + 3x_{25} + x_{35} + 5x_{34} + 4x_{24} \\ &s.a. : x_{12} + x_{13} = 4 \\ &\quad - x_{12} + x_{25} + x_{24} = 10 \\ &\quad - x_{34} - x_{24} = -4 \\ &\quad - x_{13} + x_{34} + x_{35} = -7 \\ &\quad - x_{35} - x_{25} = -3 \\ &\quad 0 \leq x_{12} \leq 2, 4 \leq x_{13} \leq 8, 5 \leq x_{25} \leq 9 \\ &\quad 1 \leq x_{35} \leq 8, 0 \leq x_{34} \leq 5, 2 \leq x_{24} \leq 6 \end{aligned} \tag{2.3}$$

La red correspondiente es:



Para un estudio más detallado de estos problemas, ver, por ejemplo, [2] y [4].

2.2. Conceptos y propiedades.

Definición 2.2. *Una* ***cadena*** *es una sucesión de arcos que conecta un vértice con otro.*

Definición 2.3. *Un* ***camino*** *es una sucesión de arcos que va del vértice i al j siguiendo las direcciónes de los arcos. En nuestro ejemplo 2.3, un camino sería ir del vértice 1 al 4 de la siguiente forma:* $1 \to 3 \to 4$.

Definición 2.4. *Un* ***ciclo*** *es una cadena que comienza y termina en el mismo vértice. Un ciclo del ejemplo 2.3 sería:* $1 \to 3 \to 5 \leftarrow 2 \leftarrow 1$.

Nota. En este trabajo vamos a considerar que las cadenas, los caminos y los ciclos son simples, es decir, los vértices utilizados son todos distintos.

Propiedad 2.1 *Para que el problema de flujo de costo mínimo tenga solución factible, debe cumplirse que* $\sum_{i=1}^{n} b_i = 0$.

Definición 2.5. *Supongamos que* $M = (B, N_1, N_2)$*, con* B *base, que es una matriz no singular de orden* $n-1$. x *es una* ***solución básica*** *si* $x_{N_1} = l_{N_1}$, $x_{N_2} = u_{N_2}$. *Si, además,* $l_B \leq x_B \leq u_B$, x *es una* ***solución básica factible****.*

Definición 2.6. *Un* ***árbol*** T *es una red conexa* $R = (V, A)$ *que no contiene ciclos. Si T=(V,A') con* $A' \subset A$, T *se denomina* ***árbol generador*** *de R.*

Proposición 2.7. *i) Si T es un árbol generador, entonces de las n-1 columnas asociadas en la matriz M se extrae una submatriz cuadrada de orden n-1 no singular.*

ii) Dada cualquier submatriz cuadrada no singular de orden n-1 de M, sus columnas se corresponden con los arcos de un árbol generador.

Por lo tanto, el rango de la matriz M es igual a n-1.

Propiedad 2.2 *Las posibles soluciones básicas del problema de flujo de costo mínimo se corresponden con árboles generadores de la red asociada y tienen* $n-1$ *variables básicas.*

2.3. El Método Simplex Primal para redes

2.3.1. El Método Simplex Primal para redes sin capacidades

Sea problema de flujo de costo mínimo sin capacidades:

$$\begin{aligned} &min \sum_{i \in V} \sum_{j \in Suc(i)} c_{ij} x_{ij} \\ &s.a.: \sum_{j \in Suc(i)} x_{ij} - \sum_{j \in Pred(i)} x_{ji} = b_i, \forall i \in V \\ &\quad x_{ij} \geq 0, \forall (i,j) \in A \end{aligned} \tag{2.4}$$

El dual es:

$$
\begin{aligned}
& max \sum_{i \in V} b_i y_i \\
& s.a.: y_i - y_j \leq c_{ij}, \forall (i,j) \in A
\end{aligned}
\tag{2.5}
$$

El Método Simplex Primal para redes parte de un árbol generador T, es decir, una solución básica factible inicial x_B, y hasta que no se detecte la optimalidad o la no acotación, debemos de encontrar una solución básica factible mejor.

La condición de optimalidad nos dice que una solución es óptima si los costos relativos de las variables no básicas son mayores o iguales a cero.

A continuación veremos cómo se obtienen los costos relativos y cómo calcular otras soluciones básicas factibles en caso de que la actual no sea la óptima.

El problema 2.4 lo podemos también representar como:

$$
\begin{aligned}
& min\ c_B^t x_B + c_B^t x_B \\
& s.a.: Bx_B + Nx_N = b \\
& x_B \geq 0, x_N \geq 0,
\end{aligned}
\tag{2.6}
$$

y su dual es:

$$
\begin{aligned}
& max\ b^t y \\
& s.a.: B^t y \leq c_B \\
& N^t y \leq c_N
\end{aligned}
\tag{2.7}
$$

Forzando la igualdad de la primera inecuación matricial de 2.7, obtenemos que $c_B^t = y^t B$ donde a los elementos de y los denominamos potenciales; y despejando la y^t y hallando las holguras en la segunda inecuación, llegamos a $c_N^t - c_B^t B^{-1} N = \overline{c}_N^t$. los $\overline{c}_N^t = c_N^t - c_B^t B^{-1} N$ los denominamos costos relativos.

Si ahora nos fijamos en 2.5, es fácil ver que los potenciales salen de: $c_{ij} = y_i - y_j$, $\forall (i,j) \in A_T$ y los costos relativos son: $\overline{c}_{ij} = c_{ij} - (y_i - y_j)$, $\forall (i,j) \in A - A_T$, donde A son los arcos de la red y A_T los arcos del árbol generador.

Una vez calculados los costos relativos, vemos si cumple la condición de optimalidad; si se cumple paramos con el proceso y la solución factible básica es óptima; en caso contrario tenemos que ver cómo se mejora la solución básica factible actual adaptando el Método del Simplex.

La variable que se va a convertir en básica es la que tiene el $(r, s) \in A - A_T$ con costo relativo más negativo.

Para ver qué variable sale de la base, hacemos $x_{rs} = \lambda$, y añadimos el arco (r, s) a T formando en este un único ciclo.

Los arcos del ciclo deben modificar su flujo sumando o restando flujo anterior un λ (sumando en el caso de que el arco tenga mismo sentido, en el ciclo, que el arco (i, j) y restando en el caso de que tenga el sentido contrario).

λ será el mínimo valor que cumpla la desigualdad: $x_{ij} \pm \lambda \geq 0$, con $\lambda \geq 0$.

Una vez obtenido el λ, observamos que al menos un x_{ij} ha pasado a ser igual a cero. Una de estas variables dejará de ser básica.

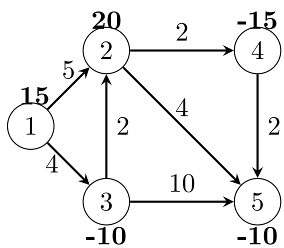
Si se quiere ampliar conocimientos, consultar, por ejemplo, [2] y [4].

Para determinar una solución básica factible inicial, podemos utilizar métodos auxiliares, como por ejemplo, el Método de Penalización o el Método de las Dos Fases. Usaremos, en el siguiente ejemplo, el Método de Penalización.

Ejemplo 2.8. Resolver:

$$\begin{aligned} &min\ 5x_{12} + 4x_{13} + 2x_{32} + 2x_{24} + 4x_{25} + 10x_{35} + 2x_{45} \\ &s.a.: x_{12} + x_{13} = 15 \\ &\quad - x_{12} - x_{32} + x_{25} + x_{24} = 20 \\ &\quad - x_{13} + x_{32} + x_{35} = -10 \\ &\quad - x_{24} + x_{45} = -15 \\ &\quad - x_{35} - x_{25} - x_{45} = -10 \\ &\quad x_{ij} \geq 0 \end{aligned} \tag{2.8}$$

La red correspondiente es:



donde al lado de los arcos hemos puesto el costo por unidad de flujo.

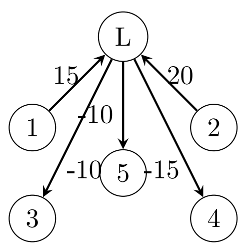
Como no disponemos de una solución básica factible inicial, añadimos un vértice ficticio, L, con $b_L = 0$, y construimos un árbol artificial con arcos que entran desde los vértices de oferta y salen desde L hasta los vértices de demanda.

La posible eliminación posterior de esos arcos artificiales se realiza penalizándolos con un costo $M > 0$ suficientemente grande dentro del proceso conocido como Método de Penalización.

Por tanto, se nos quedaría el problema de la siguiente manera:

$$\begin{aligned}
min\ & 5x_{12} + 4x_{13} + 2x_{32} + 2x_{24} + 4x_{25} + 10x_{35} + 2x_{45} + Mx_{1L} + Mx_{2L} + Mx_{L3} + Mx_{L4} \\
& + Mx_{L5} \\
s.a.:\ & x_{12} + x_{13} + x_{1L} = 15 \\
& - x_{12} - x_{32} + x_{25} + x_{24} + x_{2L} = 20 \\
& - x_{13} + x_{32} + x_{35} - x_{L3} = -10 \\
& - x_{24} + x_{45} - x_{L4} = -15 \\
& - x_{35} - x_{25} - x_{45} - x_{L5} = -10 \\
& x_{ij} \geq 0
\end{aligned} \tag{2.9}$$

Las variables básicas y sus valores son: $x_{1L} = 15$, $x_{2L} = 20$, $x_{L3} = -10$, $x_{L4} = -15$, $x_{L5} = -10$. Y $A_T = \{(1, L), (2, L), (L, 3), (L, 4), (L, 5)\}$. El árbol generador es:



Siguiendo el proceso del Método Simplex para redes, ahora debemos ver si la solución es óptima o no. En caso de que no sea óptima, tenemos que volver a buscar otra solución básica factible, y así hasta encontrar una que cumpla la condición de optimalidad.

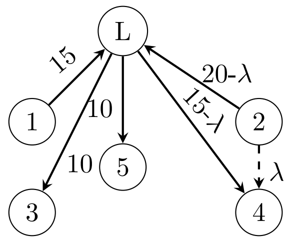
Veamos cuáles son los potenciales:

$y_1 = 0.$
$c_{1L} = y_1 - y_L$, $M = 0 - y_L \Rightarrow y_L = -M.$
$c_{2L} = y_2 - y_L$, $M = y_2 + M \Rightarrow y_2 = 0.$
$c_{L3} = y_L - y_3$, $M = -M - y_3 \Rightarrow y_3 = -2M.$
$c_{L4} = y_L - y_4$, $M = -M - y_4 \Rightarrow y_4 = -2M.$
$c_{L5} = y_L - y_5$, $M = -M - y_5 \Rightarrow y_5 = -2M.$

Ahora, para las variables no básicas calculamos los costos relativos:
$\overline{c}_{12} = c_{12} - (y_1 - y_2)$, $\overline{c}_{12} = 5 - (0 - 0) = 5$.
$\overline{c}_{13} = c_{13} - (y_1 - y_3)$, $\overline{c}_{13} = 4 - (0 + 2M) = 4 - 2M$.
$\overline{c}_{24} = c_{24} - (y_2 - y_4)$, $\overline{c}_{24} = 2 - (0 + 2M) = 2 - 2M$.
$\overline{c}_{25} = c_{25} - (y_2 - y_5)$, $\overline{c}_{25} = 4 - (0 + 2M) = 4 - 2M$.
$\overline{c}_{32} = c_{32} - (y_3 - y_2)$, $\overline{c}_{32} = 2 - (-2M - 0) = 2 + 2M$.
$\overline{c}_{35} = c_{35} - (y_3 - y_5)$, $\overline{c}_{35} = 10 - (-2M + 2M) = 10$.
$\overline{c}_{45} = c_{45} - (y_4 - y_5)$, $\overline{c}_{45} = 2 - (-2M + 2M) = 2$.

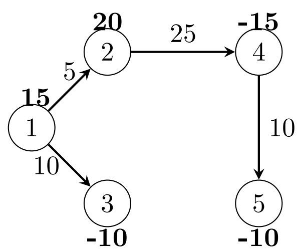
El costo relativo más negativo es $\overline{c}_{24}$, por tanto, x_{24} es la variable que entra, y $x_{24} = \lambda$.

Ajustando los valores de los x_{ij} se nos quedaría de la forma:



Y observamos que el mínimo valor de λ que satisface que: $x_{ij} \pm \lambda \geq 0$ es $\lambda = 15$. Esto hará que $(L, 4)$ salga del árbol.

En este ejemplo, repitiendo cuatro veces más el método, llegamos a que los costos relativos son todos positivos, por tanto, la solución básica factible es óptima con $x_{12} = 5, x_{13} = 10, x_{24} = 25, x_{45} = 10$.



2.3.2. El Método Simplex Primal para redes con capacidades

Partimos del problema de flujo de costo mínimo:

$$\begin{aligned}
&min \sum_{i\in V}\sum_{j\in Suc(i)} c_{ij}x_{ij} \\
&s.a.: \sum_{j\in Suc(i)} x_{ij} - \sum_{j\in Pred(i)} x_{ji} = b_i, \forall i \in V \\
&\quad l_{ij} \leq x_{ij} \leq u_{ij}, \forall (i,j) \in A
\end{aligned} \tag{2.10}$$

El proceso es análogo al caso de sin capacidades modificando adecuadamente la condición de optimalidad y la manera de seleccionar las variables que entran y salen de la base.
Por tanto, hacemos el mismo proceso, y una vez hayamos llegado al cálculo de los costos relativos $\overline{c}_{ij}$ $\forall (i,j) \in A - A_T$, la condición de optimalidad nos dice que la solución es óptima si:

$$\overline{c}_{ij} \geq 0, \qquad \forall (i,j) \in N_1$$

$$\overline{c}_{ij} \leq 0, \qquad \forall (i,j) \in N_2$$

Si la condición de optimalidad no se cumple, debemos mejorar buscando otras soluciones básicas factibles o hasta determinar la optimalidad o detectar la no acotación. Para ello, debemos buscar qué variable entra a ser básica y cual sale de la base.
La variable que entra en la base debe cumplir:

$$min\{\{\overline{c}_{ij}/\overline{c}_{ij} < 0, \forall (i,j) \in N_1\} \cup \{-\overline{c}_{ij}/\overline{c}_{ij} > 0, \forall (i,j) \in N_2\}\}.$$

Supongamos que el arco que cumple con lo anterior es el (r,s). Este formará parte de un único ciclo. Debemos ver qué variable sale de la base.
Se procede de la manera siguiente:
-Si $x_{rs} = l_{rs}$, con $(r,s) \in N_1$, entonces, el nuevo valor será $x_{rs} = l_{rs} + \lambda$.
-Si $x_{rs} = u_{rs}$, con $(r,s) \in N_2$, entonces, el nuevo valor será $x_{rs} = u_{rs} - \lambda$.
A los otros arcos del ciclo también tenemos que sumarle o restarle un λ para que la red quede compensada.
El valor de λ será el máximo valor de λ que cumpla las siguientes desigualdades.

Cuando $x_{rs} = l_{rs} + \lambda$, $(r,s) \in N_1$:

a) $l_{ij} \leq x_{ij} + \lambda \leq u_{ij}$, si $(i,j) \in A_T$ y tiene sentido coincidente a (r,s) en el ciclo.

b) $l_{ij} \leq x_{ij} - \lambda \leq u_{ij}$, si $(i,j) \in A_T$ y tiene sentido contrario a (r,s) en el ciclo.

c)$l_{rs} \leq l_{rs} + \lambda \leq u_{rs}$

Cuando $x_{rs} = u_{rs} - \lambda$, $(r,s) \in N_2$:

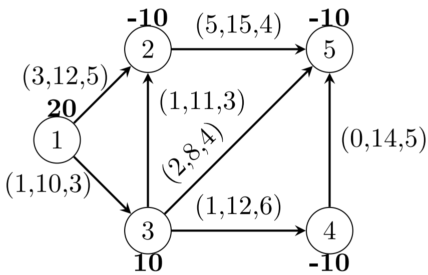
a) $l_{ij} \leq x_{ij} - \lambda \leq u_{ij}$, si $(i,j) \in A_T$ y tiene sentido coincidente a (r,s) en el ciclo.

b) $l_{ij} \leq x_{ij} + \lambda \leq u_{ij}$, si $(i,j) \in A_T$ y tiene sentido contrario a (r,s) en el ciclo.

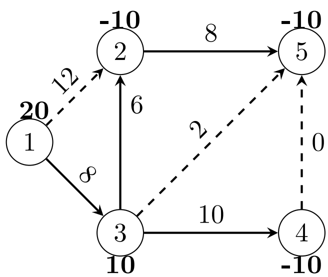
c)$l_{rs} \leq u_{rs} - \lambda \leq u_{rs}$.

Una vez obtenido el valor de λ, observamos que al menos un x_{ij} se anula, por lo que el arco (i,j) correspondiente saldrá de la base.

Ejemplo 2.9. Minimizar:



Partimos de la solución básica factible primal inicial:



donde $B = \{(1,3),(3,2),(2,5),(3,4)\}$, $N_1 = \{(3,5),(4,5)\}$, $N_2 = \{(1,2)\}$.

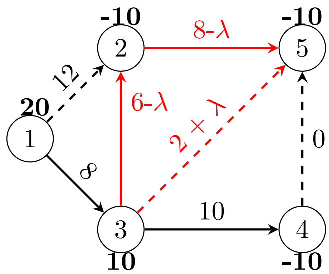
Siguiendo los pasos del Método Simplex Primal para redes, veamos cuáles son sus potenciales y cuáles son sus costos relativos para ver si la solución es óptima o debemos seguir buscando otra.

Veamos sus potenciales:
$y_1 = 0$.
$c_{13} = y_1 - y_3$, $3 = 0 - y_3 \Rightarrow y_3 = -3$.
$c_{32} = y_3 - y_2$, $3 = -3 - y_2 \Rightarrow y_2 = -6$.
$c_{25} = y_2 - y_5$, $4 = -6 - y_5 \Rightarrow y_5 = -10$.
$c_{34} = y_3 - y_4$, $6 = -3 - y_4 \Rightarrow y_4 = -9$.
Los potenciales para la solución básica factible son: $y_1 = 0$, $y_2 = -6$, $y_3 = -3$, $y_4 = -9$, $y_5 = -10$.

Ahora, para las variables no básicas calculamos los costos relativos:
$\bar{c}_{12} = c_{12} - (y_1 - y_2)$, $\bar{c}_{12} = 5 - (0+6) = -1$. Como $(1,2) \in N_2$ y $\bar{c}_{12} \leq 0$, se cumple la condición de optimalidad.
$\bar{c}_{45} = c_{45} - (y_4 - y_5)$, $\bar{c}_{45} = 5 - (-9+10) = 4$. Como $(4,5) \in N_1$ y $\bar{c}_{45} \geq 0$, se

cumple la condición de optimalidad.
$\bar{c}_{35} = c_{35} - (y_3 - y_5)$, $\bar{c}_{35} = 4 - (-3 + 10) = -3$. Tenemos que $(3,5) \in N_1$ y $\bar{c}_{35} \leq 0$, por tanto no se cumple la condición de optimalidad. x_{35} debe entrar en la base.

Siguiendo con las pautas: $x_{35} = l_{35} = 2 \Rightarrow x_{35} = 2 + \lambda$ y se crea un ciclo:



Y ahora veamos qué variable sale de la base. Para ello, veamos quién es λ buscando el máximo valor de λ que cumpla las siguientes desigualdades:

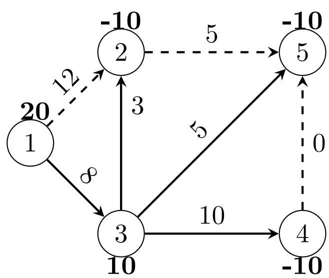
$$5 \leq 8 - \lambda \leq 15$$

$$1 \leq 6 - \lambda \leq 11$$

$$2 \leq 2 + \lambda \leq 8$$

Y se observa que $\lambda = 3$. De esta manera, x_{25} sale de la base.

La nueva solución básica factible es el árbol generador:



donde $B = \{(1,3),(3,2),(3,5),(3,4)\}$, $N_1 = \{(2,5),(4,5)\}$, $N_2 = \{(1,2)\}$. Repitiendo el proceso anterior, tenemos que los potenciales son: $y_1 = 0$, $y_2 = -6$, $y_3 = -3$, $y_4 = -9$, $y_5 = -7$ y los costos relativos son:
$\bar{c}_{12} = -1$. Como $(1,2) \in N_2$ y $\bar{c}_{12} \leq 0$, se cumple la condición de optimalidad.
$\bar{c}_{45} = 7$. Como $(4,5) \in N_1$ y $\bar{c}_{45} \geq 0$, se cumple la condición de optimalidad.
$\bar{c}_{25} = 3$. Como $(2,5) \in N_1$ y $\bar{c}_{25} \geq 0$, se cumple la condición de optimalidad.

Por tanto, nuestra solución es óptima.

3

El Método Simplex Dual para Redes. El Método Auto-Dual

3.1. El Método Simplex Dual para Redes

Trabajaremos en este capítulo con problemas de flujo de costo mínimo sin capacidades.

El Método Simplex Dual para redes trabaja, en cada iteración, con soluciones básicas factibles duales hasta detectar la no factiblidad del problema o una solución básica que sea primal y, en consecuencia, óptima. Si la solución básica actual no es factible primal, es porque al menos una x_{ij} $\forall (i,j) \in A_T$, es negativa.

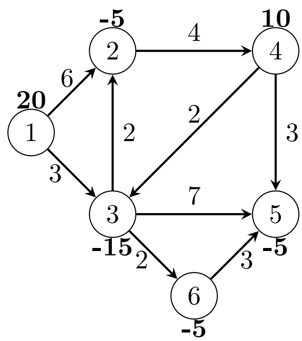
Así, el método consiste en eliminar el arco con el flujo $x_{ij} < 0$ más negativo de T, haciendo que por el circule un flujo igual a cero. De esta manera, se nos queda divido en dos componentes conexas T' y T", por lo que debemos buscar un arco puente entre T' y T" que tenga sentido contrario a (i,j), para que entre en el nuevo árbol generador con flujo igual a $a - x_{ij} > 0$. Cuando tenemos varios arcos candidatos, escogemos el que tenga el menor costo relativo.

Si añadimos el arco (r,s) al árbol T, con $x_{rs} = \lambda$, se forma un ciclo. Por tanto, habrá que modificar los demás flujos que forman el ciclo, añadiendo un $\pm\lambda$, para que el grafo quede compensado. Es obvio que λ es el número de unidades que necesita el arco que vamos a eliminar, para ser 0 . Es decir, si $x_{ij} = a < 0$, entonces $\lambda = -a > 0$.

Para una información más detallada consultar, por ejemplo, [4].

Ejemplo 3.1. Resolver:

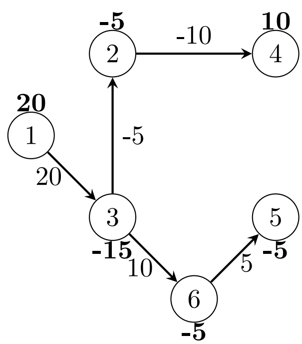
$$\begin{aligned} &min\ 6x_{12} + 3x_{31} + 2x_{32} + 4x_{24} + 2x_{43} + 7x_{35} + 3x_{45} + 2x_{46} + 3x_{65} \\ &s.a.: x_{12} + x_{13} = 20 \\ &\quad - x_{12} - x_{32} + x_{24} = -5 \\ &\quad - x_{13} + x_{32} - x_{43} + x_{35} + x_{36} = -15 \\ &\quad - x_{24} + x_{43} + x_{45} = 10 \\ &\quad - x_{35} - x_{45} - x_{65} = -5 \\ &\quad - x_{36} + x_{56} = -5, x_{ij} \geq 0 \end{aligned} \tag{3.1}$$



para la solución básica factible dual inicial: $x_{13} = 20$, $x_{32} =$ -5, x_{24}= -10, x_{36}=10, x_{65}=5.

Tenemos que la solución básica factible dual no es óptima, así que veamos si cumple la condición de optimalidad.

Para ello, usando el árbol generador correspondiente, tenemos que hallar los costos relativos para ver si se da la condición de factibilidad dual:



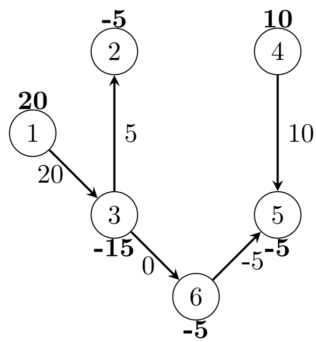
Los correspondientes potenciales son: $y_1 = 0$, $y_2 = -5$, $y_3 = -3$, $y_4 = -9$, $y_5 = -8$, $y_6 = -5$.

Así, los valores de los costos relativos son: $\overline{c}_{12} = 6 - (0+5) = 1$, $\overline{c}_{43} = 2 - (-9+3) = 8$, $\overline{c}_{35} = 7 - (-3+8) = 2$ y $\overline{c}_{45} = 3 - (-9+8) = 4$.

Por tanto, se cumple la condición de optimalidad y la solución básica no es factible primal ya que los arcos $(3,2)$ y $(2,4)$ poseen flujos negativos. Es por ello que debemos aplicar el Método Simplex Dual para Redes.

Tenemos que x_{24} = -10 es el flujo más negativo, por tanto, debe salir el arco (2,4) del árbol generador, y para ver qué arco va entrar en la base, tenemos que ver qué arcos cumplen las condiciones citadas anteriormente.

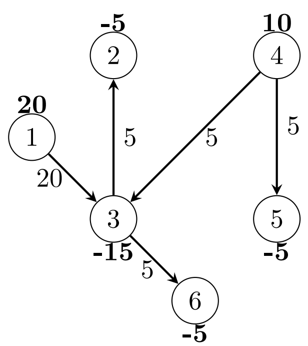
En este ejemplo, son (4,3) y (4,5) los posibles candidatos y $\overline{c}_{43} = 8 > \overline{c}_{45} = 4$. Por tanto $(4,5)$ debe entra en la base para formar un nuevo T, con $x_{45} = 10$. Modificando los demás flujos llegamos a que el nuevo árbol generador es:



La nueva solución básica tiene que $x_{65} = -5$, por tanto, la solución no es óptima. Tenemos que $y_1 = 0$, $y_2 = -5$, $y_3 = -3$, $y_4 = -4$, $y_5 = -8$, $y_6 = -5$. Los costos relativos son: $\overline{c}_{12} = 1, \overline{c}_{43} = 4, \overline{c}_{24} = 4, \overline{c}_{35} = 2$.

Así que $c_{ij} \geq 0 \forall (i,j) \in A_T$, por lo que se cumple la condición de optimalidad y debemos aplicar el MSD para Redes.

El arco (6,5) sale del árbol, y el arco (4,3) es el único que cumple las condiciones citadas, por lo que, repitiendo el mismo proceso que en el paso anterior, llegamos a que el nuevo árbol generador es:



Tenemos que todas las variables básicas y todos los $\overline{c}_{ij}$ son no negativos, por lo que la solución es óptima con $x_{13} = 20$, $x_{32} = 5$, x_{43}=5, x_{36}=5, x_{65}=5.

3.2. El Método Auto-Dual

Como acabamos de ver, el Método Simplex Dual para Redes se utiliza cuando la solución básica no es factible primal pero sí se cumple la condición de optimalidad.

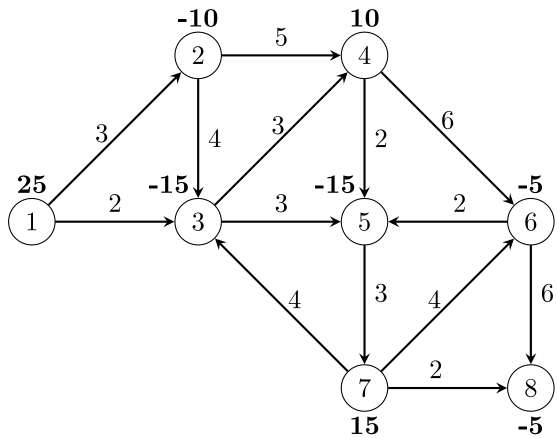
El Método Auto-Dual (ver, por ejemplo [3]) parte de una solución básica que no es ni factible primal ni factible dual, por lo que, para conseguir llegar a una solución óptima, debemos perturbar tanto los flujos como los costos relativos.

Sea μ el parámetro de perturbación, la optimalidad del problema se conseguirá para un intervalo $\underline{\mu} \leq \mu \leq \overline{\mu}$, y el proceso finalizará cuando encontremos un intervalo donde esté contenido el cero, ya que así se llega a la optimalidad del problema sin perturbación.

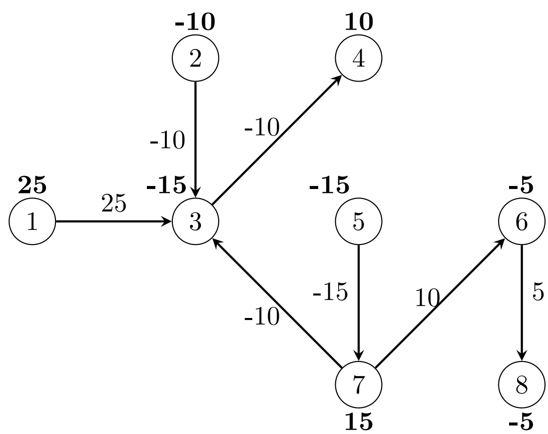
Así, si partimos de un problema de costo mínimo donde la solución básica tiene flujos y costos relativos negativos, le añadiremos la perturbación μ a estos, y realizamos las operaciones de pivotajes (primal o dual, según convenga) explicadas anteriormente. Veámoslo más claro con el siguiente ejemplo:

Ejemplo 3.2. Resolver

$$\begin{aligned}
min\ & 3x_{12} + 2x_{13} + 4x_{23} + 5x_{24} + 3x_{34} + 3x_{35} + 2x_{45} + 6x_{46} + 3x_{57} + 2x_{65} + 6x_{68} + 4x_{76} \\
& + 2x_{78} + 4x_{73} \\
s.a. :\ & x_{12} + x_{13} = 25 \\
& - x_{12} + x_{23} + x_{24} = -10 \\
& - x_{13} - x_{23} + x_{34} + x_{35} - x_{36} = -15 \\
& - x_{24} + x_{43} + x_{45} = 10 \\
& - x_{35} - x_{45} - x_{65} = -5 \\
& - x_{36} + x_{56} = -5, x_{ij} \geq 0
\end{aligned} \tag{3.2}$$



siendo la solución básica inicial: $x_{13} = 25$, $x_{23} = -10$, $x_{34} = -10$, $x_{73} = -10$, $x_{57} = -15$,$x_{76} = 10$, $x_{68} = 5$, es decir, el árbol generador:



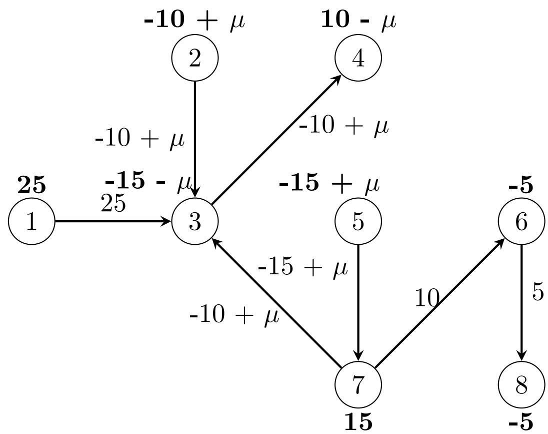
Tenemos que la solución básica no es factible primal. Veamos si se cumple la condición de optimalidad.

Para ello veamos cuáles son sus potenciales y sus costos relativos: $y_1 = 0$, $y_2 = 2$, $y_3 = -2$, $y_4 = -5$, $y_5 = 5$, $y_6 = -2$, $y_7 = 2$, $y_8 = -8$.

Así, los valores de los costos relativos son: $\overline{c}_{12} = 3 - (0 - 2) = 5$, $\overline{c}_{24} = 5 - (2 + 5) = -2$, $\overline{c}_{35} = 3 - (-2 - 5) = 10$, $\overline{c}_{45} = 2 - (-5 - 5) = 12$, $\overline{c}_{46} = 6 - (-5 + 2) = 9$, $\overline{c}_{65} = 2 - (-2 - 5) = 9$ y $\overline{c}_{78} = 2 - (2 + 8) = -8$.

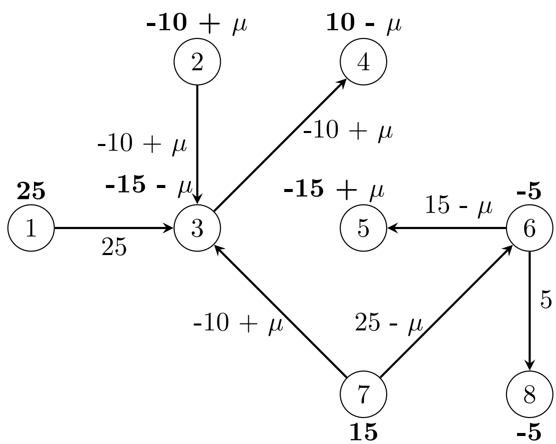
Por tanto, el problema no tiene solución básica factible primal ni cumple la condición de optimalidad.

Así que perturbamos el problema de costo mínimo para que sea óptimo. Para ello, hacemos $x_{23} = -10+\mu$, $x_{34} = -10+\mu$, $x_{73} = -10+\mu$, $x_{57} = -15+\mu$, $\overline{c}_{24} = -2+\mu$ y $\overline{c}_{78} = -8+\mu$. Por tanto, ahora $c_{24} = 5+\mu$ y $c_{78} = 2+\mu$.



Este problema es óptimo si $\mu \geq 15$. Si $\mu \in [8, 15]$, tenemos que la solución no es factible primal pero sí cumple la condición de optimalidad, por tanto, debemos seguir el mismo proceso que en el MSD. Es decir, (5,7) debe salir de la base y (6,5) entra.

Ahora tenemos que el nuevo árbol generador es:

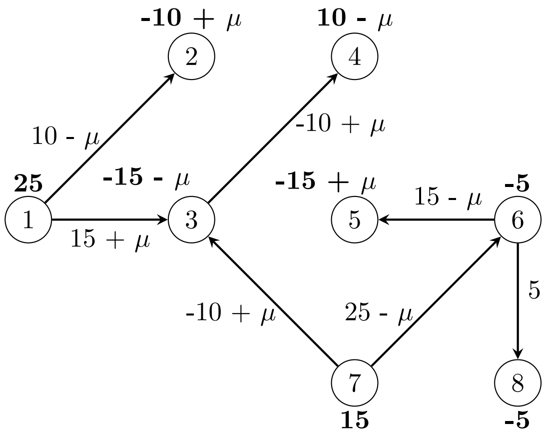


La solución básica es: $x_{13} = 25$, $x_{23} = -10+\mu$, $x_{34} = -10+\mu$, $x_{73} = -10+\mu$, $x_{76} = 25-\mu$, $x_{65} = 15-\mu$, $x_{68} = 5$.

Así, las potencias son: $y_1 = 0$, $y_2 = 2$, $y_3 = -2$, $y_4 = -5$, $y_5 = -4$, $y_6 = -2$, $y_7 = 2$, $y_8 = -8$. Y los costos relativos son: $\overline{c}_{12} = 5$, $\overline{c}_{24} = 5+\mu-(2+5) = -2+\mu$, $\overline{c}_{35} = 1$, $\overline{c}_{45} = 3$, $\overline{c}_{46} = 9$, $\overline{c}_{57} = 9$ y $\overline{c}_{78} = 2+\mu-(2+8) = -8+\mu$.

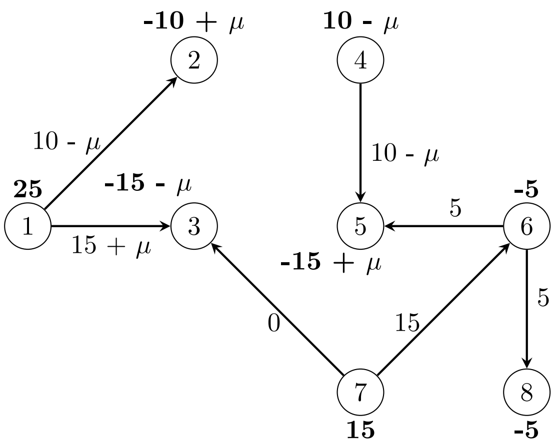
Y este problema es óptimo para $\mu = 10$. Si $\mu \in [8, 10]$, tenemos que (2,3) debe salir y (1,2) entra.

Ahora tenemos que nuestro árbol generador es:



Siendo las potencias: $y_1 = 0$, $y_2 = -3$, $y_3 = -2$, $y_4 = -5$, $y_5 = -4$, $y_6 = -2$, $y_7 = 2$, $y_8 = -8$ y los costos relativos: $\overline{c}_{23} = 5$, $\overline{c}_{24} = 3 + \mu$, $\overline{c}_{35} = 1$, $\overline{c}_{45} = 3$, $\overline{c}_{46} = 9$, $\overline{c}_{57} = 9$ y $\overline{c}_{78} = -8 + \mu$.

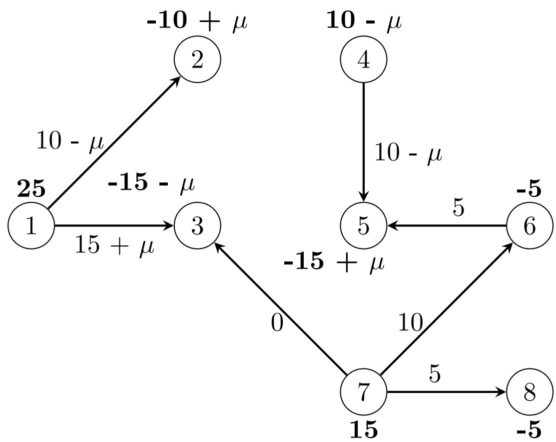
Esta solución es óptima para $\mu = 10$. Si $\mu \in [8, 10]$, tenemos que (3,4) sale y (4,5) entra. De esta forma, tenemos que T es:



Y tenemos que las potencias son: $y_1 = 0$, $y_2 = -3$, $y_3 = -2$, $y_4 = -2$, $y_5 = -4$, $y_6 = -2$, $y_7 = 2$, $y_8 = -8$ y los costos relativos: $\overline{c}_{23} = 5$, $\overline{c}_{24} = 6 + \mu$, $\overline{c}_{35} = 1$, $\overline{c}_{34} = 3$, $\overline{c}_{46} = 6$, $\overline{c}_{57} = 9$ y $\overline{c}_{78} = -8 + \mu$.

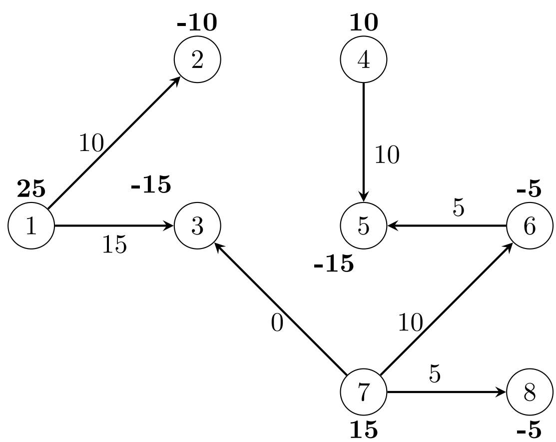
Por lo que esta solución básica es óptima para $\mu = [8, 10]$, y ya tenemos que la solución es básica factible primal, pero no es óptima ya que $\overline{c}_{78} < 0$. Por tanto, debemos de hacer el pivotaje primal explicado en el capítulo 2.

El arco (7,8) debe entrar y (6,8) salir. De esta manera tenemos que la solución básica factible primal es:



Las potencias son: $y_1 = 0$, $y_2 = -3$, $y_3 = -2$, $y_4 = -2$, $y_5 = -4$, $y_6 = -2$, $y_7 = 2$, $y_8 = -\mu$ y los costos relativos: $\overline{c}_{23} = 5$, $\overline{c}_{24} = 6 + \mu$, $\overline{c}_{35} = 1$, $\overline{c}_{34} = 3$, $\overline{c}_{46} = 6$, $\overline{c}_{57} = 9$ y $\overline{c}_{68} = 8 - \mu$.

Y vemos que esta solución es óptima para $\mu \in [-6, 8]$. Como $0 \in [-6, 8]$. Tenemos que esta solución es óptima para el problema inicial, quedándose entonces de la forma:



4

Resolución computacional de problemas de flujo de coste mínimo

4.1. Introducción

En el capítulo 2, hemos visto cómo se resuelven los problemas de flujo de coste mínimo mediante el Método Simplex Primal para Redes sin capacidades y con capacidades. Para ello, hemos resuelto un ejercicio de cada tipo.

En el capítulo 3, hemos visto cómo se resuelven los problemas de flujo de coste mínimo mediante el Método Simplex Dual para Redes y el Método Auto-dual y, junto a ellos, un ejemplo resuelto de cada tipo.

En este capítulo resolveremos los ejemplos planteados en los capítulos anteriores, usando dos complementos del Excel (Solver y Microsoft Solver Foundation) y algunas librerías de R.

4.2. Resolución de problemas en Microsoft Excel

Microsoft Excel es una aplicación de hojas de cálculo. Estas hojas contienen celdas que forman una matriz de filas y columnas. En ellas, podemos añadir datos de tipo texto, numéricos e incluso fórmulas. Con el Microsoft Excel, de una manera muy sencilla y rápida, podemos resolver diversas operaciones matemáticas aunque se compongan de un número elevado de datos. Es por ello, que es muy útil en el mundo de la contabilidad y tareas financieras. Para más información sobre el excel, podemos acudir a [9].

En nuestro caso, como hemos citado, vamos a emplear el Microsoft Excel para resolver problemas de flujo de coste mínimo (usando los correspondientes complementos para resolver problemas de Programación Lineal).

4.2.1. Microsoft Solver Foundation

El MSF es un complemento que no viene instalado previamente en Excel. Por ello, debemos acudir a [10] para instalarlo.

Este complemento facilita la creación y resolución de modelos de optimización. Podemos modelar los problemas programando en lenguaje OML.

Una vez instalado este complemento, en el Microsoft Excel nos aparece una pestaña que pone “Solver Foundation”. Si pulsamos esta pestaña, nos aparece la opción “Model”, que será donde programemos lo que explicaremos a continuación.

Ejemplos de flujo de costo mínimo sin capacidades resueltos con MSF

Como ya vimos en el capítulo dos, el problema de flujo de coste mínimo sin capacidades, parte de n vértices y m arcos. Cada vértice tiene una disponibilidad, y para cada arco tenemos el flujo que circula por él. Con estos datos, podemos modelar los problema que se resuelven mediante el Método Simplex Primal para Redes.

Necesitamos programar el problema 2.4 en el MSF. Para ello, una vez que estemos en el apartado “Model”, comenzamos definiendo en “Sets”: n y m, para la indexación de los n vértices y los m arcos. A continuación, debemos añadir los parámetros nver[n], disp[n], donde “nver” son cada vértice enumerado y “disp” su disponibilidad. Los parámetros desde[m] y hasta[m] son los arcos del grafo, donde “desde” es de dónde parte el arco y “hasta” es a dónde llega. El parámetro costo[m] se refiere al costo que tiene cada arco.

A cada uno de estos parámetros debemos asignarle a qué conjunto de números pertenece. Es obvio que nver[n], desde[m], hasta[m] $\in \mathbb{N}$, por lo que debemos señalar la opción de “Nonnegative Integer”. Disp[n],costo[m] $\in \mathbb{R}$, así que les seleccionamos la opción ”Real”.

Una vez tengamos en la hoja de cálculo el problema planteado, para cada parámetro debemos señalar en qué posiciones están cada uno de sus respectivos elementos.

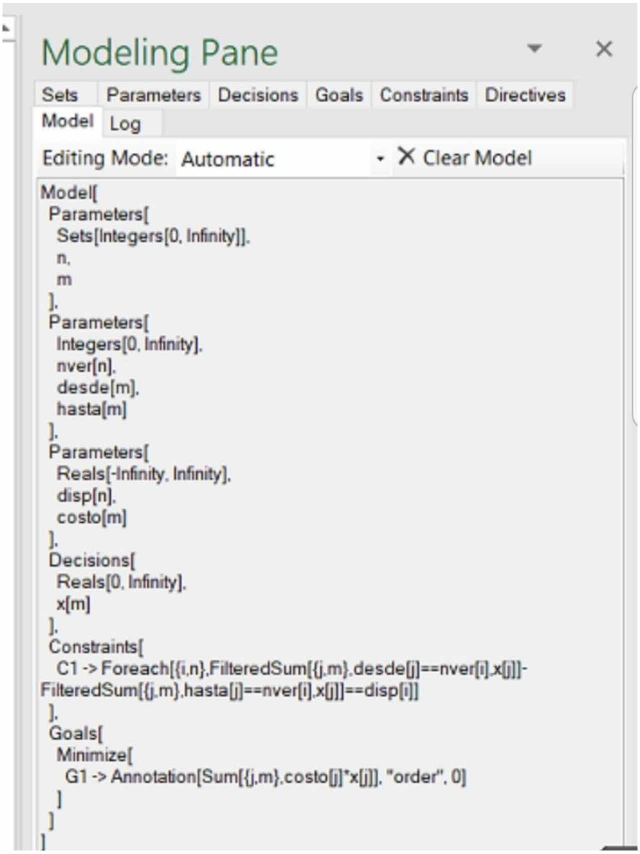
En el apartado “Decisions”, debemos introducir el vector x, que como ya sabemos, son los m flujos que circulan por la red, por tanto, debemos añadirle que dependen de m.

Tenemos que los flujos $x_{ij} \in \mathbb{R}_0^+$, por tanto, debemos seleccionar la opción de “Nonnegative Real”.

En ”Constraints” debemos programar, en lenguaje OML, las restricciones de 2.4, es decir, debemos programar : $\sum_{j\in Suc(i)} x_{ij} - \sum_{j\in Pred(i)} x_{ji} = b_i, \forall i \in V$. Esto se hace de la siguiente manera : “ Foreach[{i,n},FilteredSum[{j,m},desde[j]–nver[i],x[j]]-FilteredSum[{j,m},hasta[j]–nver[i],x[j]]–disp[i]]”.

Y finalmente, en el apartado de "Goals", como tenemos un problema de mínimo, añadimos la opción "minimize" y programamos $min \sum_{i \in V} \sum_{j \in Suc(i)} c_{ij} x_{ij}$ de la siguiente manera: "Sum[{ j,m},costo[j]*x[j]]".

De esta manera, nos sale en pantalla el modelo:



En nuestra hoja de cálculo debemos añadir todos estos datos, poniendo en columnas los vértices, las disponibilidades de cada vértice, el origen del arco, el fin del arco y su costo.

En el ejemplo 2.8, la hoja de cálculo con los datos es la que se muestra a continuación. Y si pulsamos el botón de Solver, el MSF nos resuelve el problema y nos devuelve la tabla "Solver Foundation Results", donde se incluye el valor objetivo y los valores de la solución óptima:

nver	disp	narc	desde	hasta	costo
1	15	1	1	2	5
2	20	2	1	3	4
3	-10	3	3	2	2
4	-15	4	2	4	2
5	-10	5	2	5	4
		6	3	5	10
		7	4	5	2

Solver Foundation Results

Name	Value
Solution Type	Optimal
G1	135
x[0]	5
x[1]	10
x[2]	0
x[3]	15
x[4]	10
x[5]	0
x[6]	0

Haciendo lo mismo con el ejemplo 3.1, tenemos que:

nver	disp	narc	desde	hasta	costo
1	20	1	1	2	6
2	-5	2	1	3	3
3	-15	3	3	2	2
4	10	4	2	4	4
5	-5	5	4	3	2
6	-5	6	3	5	7
		7	4	5	3
		8	3	6	2
		9	6	5	3

Solver Foundation Results

Name	Value
Solution Type	Optimal
G1	105
x[0]	0
x[1]	20
x[2]	5
x[3]	0
x[4]	5
x[5]	0
x[6]	5
x[7]	5
x[8]	0

Y para el ejemplo 3.2:

nver	disp	narc	desde	hasta	costo
1	25	1	1	2	3
2	-10	2	1	3	2
3	-15	3	2	3	4
4	10	4	2	4	5
5	-15	5	3	4	3
6	-5	6	3	5	3
7	15	7	4	5	2
8	-5	8	4	6	6
		9	5	7	3
		10	6	5	2
		11	6	8	6
		12	7	6	4
		13	7	8	2
		14	7	3	4

Solver Foundation Results

Name	Value
Solution Type	Optimal
G1	140
x[0]	10
x[1]	15
x[2]	0
x[3]	0
x[4]	0
x[5]	0
x[6]	10
x[7]	0
x[8]	0
x[9]	5
x[10]	0
x[11]	10
x[12]	5
x[13]	0

Ejemplo de flujo de costo mínimo con capacidades resuelto con MSF

Para el problema con capacidades, es análogo al del caso anterior, añadiéndole a la hoja de cálculo, las columnas de coste mínimo y máximo. Por lo que tenemos que programar el problema 2.10 en el MSF con las mismas órdenes que el anterior pero incluyendo los parámetros cmin[m],cmax[m], señalando las columnas que correspondan con los datos y seleccionando "Nonnegative Integer". También, debemos añadir en "Constrains" las cotas de los flujos de la siguiente manera: "Foreach[$\{j, m\}$,cmin[j] <= x[j] <=cmax[j]]".

El modelo se queda entonces de esta forma:

```
Model[
 Parameters[
  Sets[Integers[0, Infinity]],
  n,
  m
 ],
 Parameters[
  Integers[0, Infinity],
  nver[n],
  desde[m],
  hasta[m],
  cmin[m],
  cmax[m]
 ],
 Parameters[
  Integers[-Infinity, Infinity],
  disp[n]
 ],
 Parameters[
  Reals[-Infinity, Infinity],
  costo[m]
 ],
 Decisions[
  Reals[-Infinity, Infinity],
  x[m]
 ],
 Constraints[
  C1 -> Foreach[{i,n},FilteredSum[{j,m},desde[j]==nver[i],x[j]]-
FilteredSum[{j,m},hasta[j]==nver[i],x[j]]==disp[i]],
  C2 -> Foreach[{j,m},cmin[j]<=x[j]<=cmax[j]]
 ],
 Goals[
  Minimize[
   G1 -> Annotation[Sum[{j,m},costo[j]*x[j]], "order", 0]
  ]
 ]
]
```

En el ejemplo 2.9, la hoja de cálculo con los datos es la que se muestra a continuación, donde también se ha añadido la tabla que nos devuelve el MSF con los valores de las variables y el valor objetivo:

nver	disp	narc	desde	hasta	cmin	cmax	costo
1	20	1	1	2	3	12	5
2	-10	2	1	3	1	10	3
3	10	3	2	5	5	15	4
4	-10	4	3	2	1	11	3
5	-10	5	3	4	1	12	6
		6	3	5	2	8	4
		7	4	5	0	14	5

Solver Foundation Results

Name	Value
Solution Type	Optimal
G1	193
x[0]	12
x[1]	8
x[2]	5
x[3]	3
x[4]	10
x[5]	5
x[6]	0

4.2.2. Solver

El Solver es un complemento del Microsoft Excel que ya viene preinstalado. Este se sitúa en el apartado de “Datos”.

Con este complemento podemos resolver los problemas de Programación Lineal de una manera muy intuitiva como veremos a continuación.

Consiste en una pantalla donde nos pregunta dónde queremos establecer el valor objetivo, el tipo de problema que queremos resolver, dónde están situadas las variables y restricciones, y qué método queremos usar para resolverlo.

Ejemplos de flujo de costo mínimo sin capacidades resueltos con Solver

Para comenzar a resolver un problema de flujo de costo mínimo, debemos crear una hoja de excel de la misma manera que explicamos para el MSF. En este caso, vamos a dejar una columna libre entre la columna de disponibilidad y la de el número de arcos, y otra casilla libre entre “hasta” y “costos”.

Usaremos la segunda columna que hemos dejado libre para los valores de las variables de la solución óptima que nos dará el Solver.

En la primera columna que hemos dejado vacía entre “disp” y “narc”, tenemos que añadir a cada casilla una fórmula. Para ello, nos fijamos qué vértice le corresponde a esa casilla. Una vez sabido esto, la fórmula que debemos añadir es la suma de las casillas de la columna “Xij” cuando “desde” coincide con el número del vértice junto a la resta de las casillas de la columna “Xij” cuando “hasta” coincide con el número del vértice. Por ejemplo ,en nuestro ejemplo, en el primer recuadro azul, hemos añadido la fórmula “=H4+H5”, ya que del vértice 1, salen los arcos (1,2) y (1,3). Como ningún arco entra en el vértice 1, no tenemos que restarle nada.

Debemos elegir una casilla para que el Solver nos devuelva en esta el valor objetivo óptimo. Para ello, debemos añadir en dicha casilla, la fórmula de suma de productos (producto escalar) de la columna de las variables básicas con la de los costos. En el ejemplo que mostramos a continuación, hemos pintado de verde la casilla del valor objetivo, y esta tiene la fórmula “=SUMAPRODUCTO(I4:I10;H4:H10)”.

Una vez tengamos todo esto formulado, debemos de abrir el Solver, y nos aparece una pantalla como la que se muestra a continuación:

Como vemos, hemos añadido la localización del valor objetivo, la localización de los flujos y hemos añadido la restricción que teníamos, es decir, las disponibilidades han de ser iguales a los elementos de la columna pintada de azul.

En el ejemplo 2.8, hemos añadido a la hoja de cálculo, los datos escritos en las casillas de color blanco, mientras que el Solver nos ha devuelto los datos escrito en las casillas de colores:

nver	disp		narc	desde	hasta	Xij	costo
1	15	15	1	1	2	5	5
2	20	20	2	1	3	10	4
3	-10	-10	3	2	4	25	2
4	-15	-15	4	2	5	0	4
5	-10	-10	5	3	2	0	2
			6	3	5	0	10
			7	4	5	10	2
							135

Si nos fijamos en este ejemplo en particular, si lo comparamos con el que hemos resuelto con el MSF, tenemos que el valor óptimo es el mismo, ya que es único, pero la solución óptima difiere (hay óptimas alternativas).
Resolviendo el ejemplo 3.1 aplicando lo anterior, tenemos que:

nver	disp		narc	desde	hasta	Xij	costo
1	20	20	1	1	2	0	6
2	-5	-5	2	1	3	20	3
3	-15	-15	3	3	2	5	2
4	10	10	4	2	4	0	4
5	-5	-5	5	3	5	0	7
6	-5	-5	6	4	3	5	2
			7	4	5	5	3
			8	3	6	5	2
			9	6	5	0	3
						105	

En el ejemplo 3.2, tenemos que:

nver	disp		narc	desde	hasta	Xij	costo
1	25	25	1	1	2	10	3
2	-10	-10	2	1	3	15	2
3	-15	-15	3	2	3	0	4
4	10	10	4	2	4	0	5
5	-15	-15	5	3	4	0	3
6	-5	-5	6	3	5	0	3
7	15	15	7	4	5	10	2
8	-5	-5	8	4	6	0	6
			9	5	7	0	3
			10	6	5	5	2
			11	6	8	0	6
			12	7	6	10	4
			13	7	8	5	2
			14	7	3	0	4
							140

Ejemplo de flujo de costo mínimo con capacidades resuelto con Solver

La diferencia de estos problemas es que tenemos que añadir las columnas de “cmin” y “cmax”, y a la hora de implementar las restricciones del Solver, tenemos que poner que la columna de “ cmin” $\leq$ “Xij” y que la columna de “cmax” $\geq$ “Xij”. Así, los parámetros de Solver son:

En el ejemplo 2.9, la hoja de cálculo con los datos es:

nver	disp		narc	desde	hasta	Xij	cmin	cman	costo
1	20	20	1	1	2	12	3	12	5
2	-10	-10	2	1	3	8	1	10	3
3	10	10	3	2	5	5	5	15	4
4	-10	-10	4	3	2	3	1	11	3
5	-10	-10	5	3	4	10	1	12	6
			6	3	5	5	2	8	4
			7	4	5	0	0	14	5
									193

En [5] podemos ver cómo resolver todo este tipo de ejercicios con Solver.

4.3. R

R es un paquete con librerías de programas que, fundamentalmente, se pueden usar para análisis estadístico. Podemos instalarlo en [7]. También contiene librerías que se pueden usar para resolver problemas de optimización [11]

La herramienta R-Studio, que se puede descargar en [8] sirve para aplicar el R con mayor facilidad. En nuestro caso, vamos a utilizarlo para resolver problemas de flujo de costo mínimo.

Ejemplos de flujo de costo mínimo sin capacidades resueltos con R

Veamos con el siguiente ejemplo, el 2.8, cómo se formula el problema de flujo de coste mínimo.

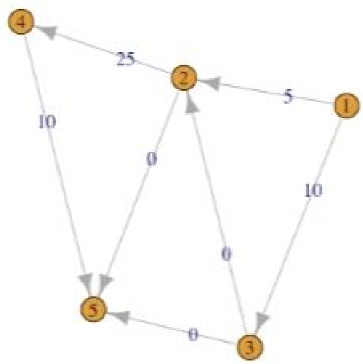
```
library(lpSolveAPI)
pfcm <- make.lp(0,7)
set.objfn(pfcm, c(5,4,2,2,4,10,2))
#desde 1,1,3,2,2,3,4
#hasta 2,3,2,4,5,5,5
add.constraint(pfcm, c(1,1,0,0,0,0,0), "=", 15)
add.constraint(pfcm, c(-1,0,-1,1,1,0,0), "=", 20)
add.constraint(pfcm, c(0,-1,1,0,0,1,0), "=", -10)
add.constraint(pfcm, c(0,0,0,-1,0,0,1), "=", -15)
add.constraint(pfcm, c(0,0,0,0,-1,-1,-1), "=", -10)
RowNames <- c("c1", "c2","c3","c4","c5")
ColNames <- c("x12", "x13", "x32","x24", "x25", "x35","x45" )
dimnames(pfcm) <- list(RowNames, ColNames)
solve(pfcm)
get.variables(pfcm)
get.objective(pfcm)
17:1   (Top Level) ÷

Console ~/
> get.variables(pfcm)
[1]  5 10  0 25  0  0 10
> get.objective(pfcm)
[1] 135
```

Como observamos, lo primero que debemos hacer es cargar la libreria "lpSolveAPI". Una vez cargada, el comando "make.lp(0,7)" crea un objeto de modelo de programación lineal, donde 0 es el número de restricciones y 7 es el número de arcos (variables) que tiene nuestro ejemplo. A continuación, se muestran los comandos para definir los costos, los arcos y las disponibilidades. Finalmente se muestra cómo obtener el valor objetivo y la solución óptima. En la consola, debajo de donde hemos programado, se muestra la solución junto al valor óptimo.

Si queremos ahora dibujar nuestra solución, tenemos que cargar las librerías "tidyverse" e "igraph". Luego, si escribimos lo siguiente:

```
#dibujar solución
library(tidyverse)
library(igraph)
edgelist <- data.frame(
  from = c(1,1,3,2,2,3,4),
  to = c(2,3,2,4,5,5,5),
  sol = c(5,10,0,25,0,0,10))
g <- graph_from_edgelist(as.matrix(edgelist[,c('from','to')]))
E(g)$sol <- edgelist$sol
plot(g, edge.label = E(g)$sol)
```

donde en la línea 22 hemos añadido de dónde salen los arcos, en la línea 23 a dónde llegan, y en la línea 24 hemos puesto la solución óptima. Entonces, R dibuja la solución óptima:



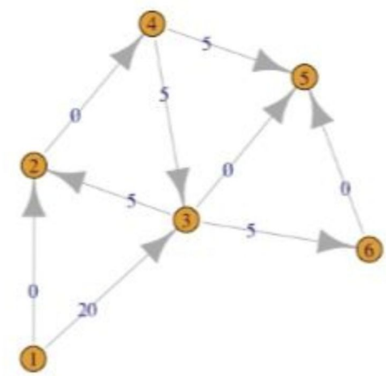
El ejemplo 3.1 sería :

```
library(lpSolveAPI)
pfcm <- make.lp(0,9)
set.objfn(pfcm, c(6,3,2,4,7,2,3,2,3))
#desde 1,1,3,2,3,4,4,3,6
#hasta 2,3,2,4,5,3,5,6,5
add.constraint(pfcm, c(1,1,0,0,0,0,0,0,0), "=", 20)
add.constraint(pfcm, c(-1,0,-1,1,0,0,0,0,0), "=", -5)
add.constraint(pfcm, c(0,-1,1,0,1,-1,0,1,0), "=", -15)
add.constraint(pfcm, c(0,0,0,-1,0,1,1,0,0), "=", 10)
add.constraint(pfcm, c(0,0,0,0,-1,0,-1,0,-1), "=", -5)
add.constraint(pfcm, c(0,0,0,0,0,0,0,-1,1), "=", -5)
RowNames <- c("c1", "c2","c3","c4","c5","c6")
ColNames <- c("x12", "x13", "x32", "x24", "x35","x43","x45","x36","x65" )
dimnames(pfcm) <- list(RowNames, ColNames)
solve(pfcm)
get.variables(pfcm)
get.objective(pfcm)

18:1   (Top Level) ÷

Console ~/
> get.variables(pfcm)
[1]  0 20  5  0  0  5  5  5  0
> get.objective(pfcm)
[1] 105
```

Y si añadimos los comandos:

```
#dibujar solución
library(tidyverse)
library(igraph)
edgelist <- data.frame(
  from = c(1,1,3,2,3,4,4,3,6),
  to = c(2,3,2,4,5,3,5,6,5),
  sol = c(0,20,5,0,0,5,5,5,0))
g <- graph_from_edgelist(as.matrix(edgelist[,c('from','to')]))
E(g)$sol <- edgelist$sol
plot(g, edge.label = E(g)$sol)

```

tenemos el grafo con la solución óptima:



El ejemplo 3.2 es:

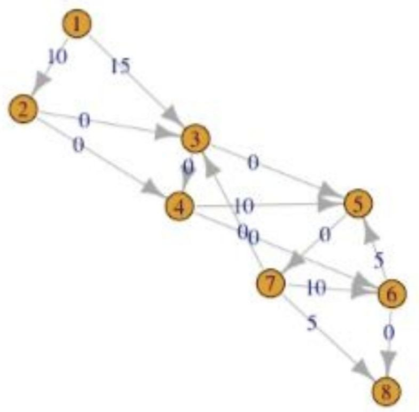
```r
library(lpSolveAPI)
pfcm <- make.lp(0,14)
set.objfn(pfcm, c(3,2,4,5,3,3,2,6,3,2,6,4,2,4))
#desde 1,1,2,2,3,3,4,4,5,6,6,7,7,7
#hasta 2,3,3,4,4,5,5,6,7,5,8,6,8,3
add.constraint(pfcm, c(1,1,0,0,0,0,0,0,0,0,0,0,0,0), "=", 25)
add.constraint(pfcm, c(-1,0,1,1,0,0,0,0,0,0,0,0,0,0), "=", -10)
add.constraint(pfcm, c(0,-1,-1,0,1,1,0,0,0,0,0,0,0,-1), "=", -15)
add.constraint(pfcm, c(0,0,0,-1,-1,0,1,1,0,0,0,0,0,0), "=", 10)
add.constraint(pfcm, c(0,0,0,0,0,-1,-1,0,1,-1,0,0,0,0), "=", -15)
add.constraint(pfcm, c(0,0,0,0,0,0,0,-1,0,1,1,-1,0,0), "=", -5)
add.constraint(pfcm, c(0,0,0,0,0,0,0,0,-1,0,0,1,1,1), "=", 15)
add.constraint(pfcm, c(0,0,0,0,0,0,0,0,0,0,-1,0,-1,0), "=", -5)
RowNames <- c("c1", "c2","c3","c4","c5","c6","c7","c8")
ColNames <- c("x12", "x13", "x23", "x24", "x34","x35","x45","x46",
              "x57", "x65", "x68", "x76","x78","x73" )
dimnames(pfcm) <- list(RowNames, ColNames)
solve(pfcm)
get.variables(pfcm)
get.objective(pfcm)
```

```
21:1   (Top Level) ÷

Console ~/
> get.variables(pfcm)
 [1] 10 15  0  0  0  0 10  0  0  5  0 10  5  0
> get.objective(pfcm)
[1] 140
```

Ahora, para dibujarla programamos:

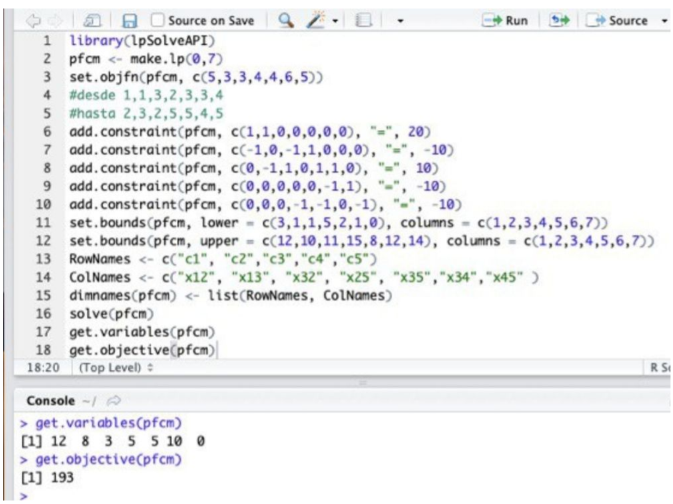
```r
#dibujar solución
library(tidyverse)
library(igraph)
edgelist <- data.frame(
  from = c(1,1,2,2,3,3,4,4,5,6,6,7,7,7),
  to = c(2,3,3,4,4,5,5,6,7,5,8,6,8,3),
  sol = c(10,15,0,0,0,0,10,0,0,5,0,10,5,0))
g <- graph_from_edgelist(as.matrix(edgelist[,c('from','to')]))
E(g)$sol <- edgelist$sol
plot(g, edge.label = E(g)$sol)
```

y la solución grafica es:



Ejemplo de flujo de costo mínimo con capacidades resuelto con R

El caso de problemas de flujo de costo mínimo con capacidades es análogo al de sin capacidades, pero en este caso, debemos de añadir las restricciones que poseen las variables, es decir, el acotamiento que tienen.



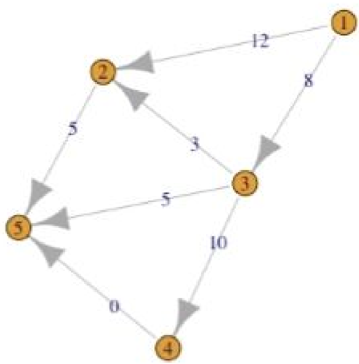
```
library(lpSolveAPI)
pfcm <- make.lp(0,7)
set.objfn(pfcm, c(5,3,3,4,4,6,5))
#desde 1,1,3,2,3,3,4
#hasta 2,3,2,5,5,4,5
add.constraint(pfcm, c(1,1,0,0,0,0,0), "=", 20)
add.constraint(pfcm, c(-1,0,-1,1,0,0,0), "=", -10)
add.constraint(pfcm, c(0,-1,1,0,1,1,0), "=", 10)
add.constraint(pfcm, c(0,0,0,0,0,-1,1), "=", -10)
add.constraint(pfcm, c(0,0,0,-1,-1,0,-1), "=", -10)
set.bounds(pfcm, lower = c(3,1,1,5,2,1,0), columns = c(1,2,3,4,5,6,7))
set.bounds(pfcm, upper = c(12,10,11,15,8,12,14), columns = c(1,2,3,4,5,6,7))
RowNames <- c("c1", "c2","c3","c4","c5")
ColNames <- c("x12", "x13", "x32", "x25", "x35","x34","x45" )
dimnames(pfcm) <- list(RowNames, ColNames)
solve(pfcm)
get.variables(pfcm)
get.objective(pfcm)
```

```
> get.variables(pfcm)
[1] 12  8  3  5  5 10  0
> get.objective(pfcm)
[1] 193
>
```

Y para verla gráficamente, escribimos:

```
#dibujar solución
library(tidyverse)
library(igraph)
edgelist <- data.frame(
  from = c(1,1,3,2,3,3,4),
  to = c(2,3,2,5,5,4,5),
  sol = c(12,8,3,5,5,10,0))
g <- graph_from_edgelist(as.matrix(edgelist[,c('from','to')]))
E(g)$sol <- edgelist$sol
plot(g, edge.label = E(g)$sol)
```

Y R nos devuelve:



A

Conclusiones

Este trabajo está dedicado al estudio de aplicaciones de las distintas variantes del Método del Simplex para resolver problemas de flujo de costo mínimo sobre redes normalizadas.

El uso de la Programación Lineal en la resolución de estos problemas está justificado por las propiedades que tiene la matriz de restricciones del modelo general que garantiza la integridad de las soluciones básicas (árboles generadores de la red).

Por esto, hemos estudiado el uso del Método del Simplex y sus variantes en la resolución de problemas de flujo de coste mínimo.

Las herramientas computacionales se han empleado para resolver casos sencillos. Este estudio se puede prolongar, desde el punto de vista computacional, en la resolución de problemas reales de mayor entidad.

Bibliografía

[1] LESLIE WINSTON, WAYNE. *Investigación de Operaciones.* Grupo Editorial Iberoamérica, 1994.

[2] MOKHTAR S., BAZARAA. *Programación Lineal y Flujo en Redes.* Editorial Limusa, 1998.

[3] J. VANDERVEI , ROBERT. *Linear Programming. Foundations and Extensions.* Editorial Springer, 2014.

[4] RAVINDRA K.,AHUJA.THOMAS L.,MAGNANTI.JAMES B., ORLIN. *Network Flows.* Pretince-Hall, 1993.

[5] F., HILLIER.G., LIEBERMAN.*Introducción a la Introducción de Operaciones.*Mc Graw Hill, 2010.

[6] LEÓN GONZÁLEZ,ÁNGEL *Manual de Investigación de Operaciones.*Ediciones Uninorte, 2003.

[7] *https://www.r-project.org/*

[8] *https://www.rstudio.com/*

[9] *https://products.office.com/es-es/excel*

[10] *https://msdn.microsoft.com/en-us/devlabs/hh145003*

[11] *https://https://cran.r-project.org/*

Variants of the Network Simplex Method

Sección de Matemáticas
Universidad de La Laguna

Helena Gil Rodríguez
Facultad de Ciencias · Sección de Matemáticas
Universidad de La Laguna
alu0100989915@ull.edu.es

Abstract

This work is dedicated to the study and resolution of important problems of Network Analysis: the problems of minimum cost on standardized networks flow. Specifically, these problems are addressed by using some variants of the Simplex Method.

This study is completed with the application of basic computational tools in the resolution of the previously presented examples.

1. Introduction

Linear Programming problems consist of the optimization of a linear function in a context defined by a system of linear equations or inequations.

A particular type of these, the problems of optimization on networks flow, are focused on the interest of this work from the perspective of its resolution through the application of the methodology of Linear Programming.

Then, we will need to make a schematic tour of the basic concepts and properties that allow to develop the Simplex Method and some of its variants, for the purpose of adapting them and subsequently applying them to problems of network flows.

2. Foundations of Linear Programming

The standard form of a P.L. problem is:

$$\begin{aligned} &min\, c^t x \\ &s.a.: Ax = b \\ &x \geq 0 \end{aligned} \qquad (1)$$

where $c \in \mathbb{R}^n$ is the cost vector, x is the decision variable vector, $A = (B, N)$ is the matrix of technology coefficients and $b \in \mathbb{R}^m$ is the resource vector.
The feasible region is $S = \{x \in \mathbb{R}^n / Ax = b, x \geq 0\}$.

So x is a basic solution if $x_B = B^{-1}b$ and $x_N = 0$, where x_B are basic variables and x_N are non basic variables.
A feasible basic solution is a basic solution with $x_B \geq 0$.

3. The Minimum-Cost Network Flow

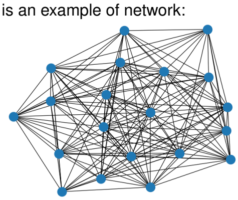
A problem of minimum cost flows can be raised as:

$$\begin{aligned} &min \sum_{i \in V} \sum_{j \in Suc(i)} c_{ij} x_{ij} \\ &s.a.: \sum_{j \in Suc(i)} x_{ij} - \sum_{j \in Pred(i)} x_{ji} = b_i, \forall i \in V \\ &l_{ij} \leq x_{ij} \leq u_{ij}, \forall (i,j) \in A \end{aligned} \qquad (2)$$

with $Suc(i) = \{j \in V / (i,j) \in A\}$, $Pred(i) = \{j \in V / (j,i) \in A\}$, where b_i is the flow availability, c_{ij} is the cost per unit of flow circulating on the arc, x_{ij} is the flow circulating from i to j, l_{ij} and u_{ij} are the minimum and maximum flow capacities, respectively, that can circulate through each arc.

x is a basic solution if $x_{N_1} = l_{N_1}$, $x_{N_2} = u_{N_2}$. If, in addition, $l_B \leq x_B \leq u_B$, x is a feasible basic solution, and the possible basic solutions of the problem of minimum cost flow correspond to the generators of the associated network.

The next graph is an example of network:



The Primal Simplex Method for Network starts from a T generator tree, that is, an initial feasible basic solution x_B, and until the optimal condition is detected, We must repeat the process with other feasible basic solutions until we find the optimal one.

4. Variants of the Simplex Method for Network

The Dual Simplex Method for Network works, in each iteration, with dual feasible basic solutions until detecting the Factability of the problem or a solution that is Primal and, consequently, optimal. If the current basic solution is not feasible Primal, it is because at least one x_{ij} $\forall (i,j) \in A_T$, is negative.

The Self-Dual Method starts from a basic solution that is neither feasible Primal nor dual feasible, so to achieve an optimal solution, we must disrupt the flows and the relative costs.

5. Computational resolution of minimum cost flow problems

Microsoft Excel has add-ons for resolving network flow problems: Solver and Microsoft Solver Foundation.

R is a program package that contains libraries capable of solving flow problems.

The examples raised throughout the work we have solved with these three tools.

References

[1] LESLIE WINSTON, WAYNE. *Investigación de Operaciones.* Grupo Editorial Iberoamérica, 1994.

[2] MOKHTAR S., BAZARAA. *Programación Lineal y Flujo en Redes.* Editorial Limusa, 1998.

[3] J. VANDERVEI , ROBERT. *Linear Programming. Foundations and Extensions.* Editorial Springer, 2014.

[4] RAVINDRA K.,AHUJA.THOMAS L.,MAGNANTI.JAMES B., ORLIN. *Network Flows.* Pretince-Hall, 1993.

[5] F., HILLIER.G., LIEBERMAN.*Introducción a la Introducción de Operaciones.*Mc Graw Hill, 2010.

[6] LEÓN GONZÁLEZ,ÁNGEL *Manual de Investigación de Operaciones.*Ediciones Uninorte, 2003.

[7] *https://www.r-project.org/*

[8] *https://www.rstudio.com/*

[9] *https://products.office.com/es-es/excel*

[10] *https://msdn.microsoft.com/en-us/devlabs/hh145003*

[11] *https://https://cran.r-project.org/*