

Investigación Operativa: Informe

Práctica 1

Yimin Pan (G12)

Introducción

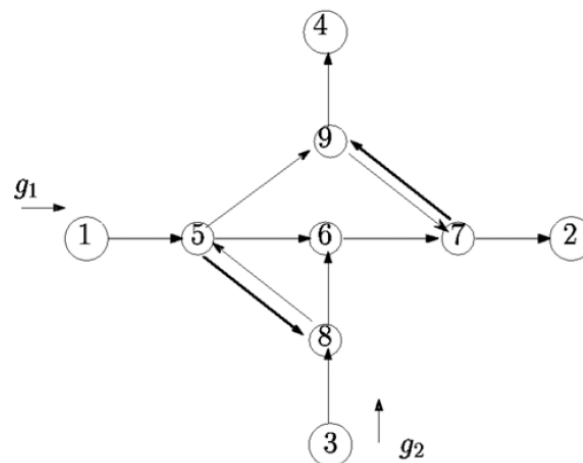
En esta primera práctica es una pequeña introducción a AMPL, un lenguaje de programación algebraica diseñada para solucionar problemas de optimización.

Destaca por su semejanza en cuanto a sintaxis a la notación matemática, por lo que es fácil traducir un problema de programación lineal a AMPL.

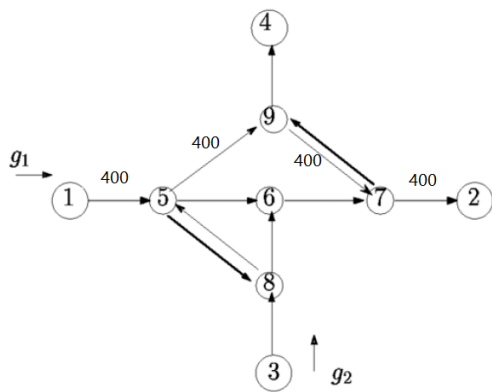
Problemas

- a) En este apartado se nos presenta el problema de flujos de redes como un problema de programación lineal (minCM). Pero queremos que los 400 unidades de flujo pase de 1 \rightarrow 2 y 400 de 3 \rightarrow 4, por lo que es una red multiartículo.

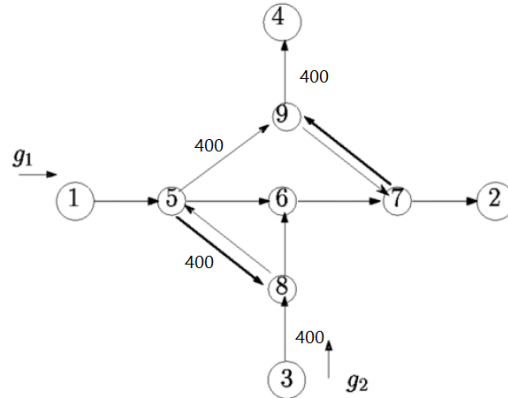
Link No.	(i, j)	t_0
1	(1, 5)	.1
2	(3, 8)	.1
3	(5, 6)	.2
4	(5, 8)	.7
5	(5, 9)	.5
6	(6, 7)	1.0
7	(7, 2)	.1
8	(7, 9)	.1
9	(8, 5)	1.0
10	(8, 6)	1.0
11	(9, 4)	.08
12	(9, 7)	.7



- b) Ya se nos proporcionó el modelo para este tipo de redes (minCM2), por lo que solo tenemos que introducir los datos de entrada en el fichero .dat para que un solver resuelva el problema. Y este es la red que obtuvimos:



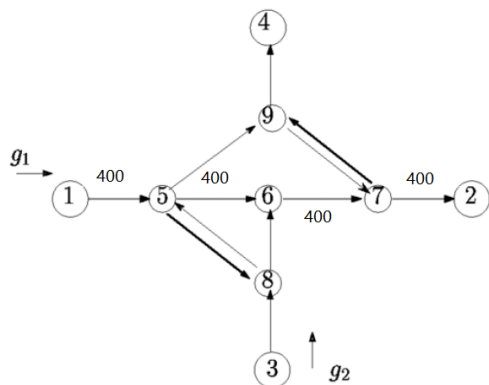
Flujo artículo 1



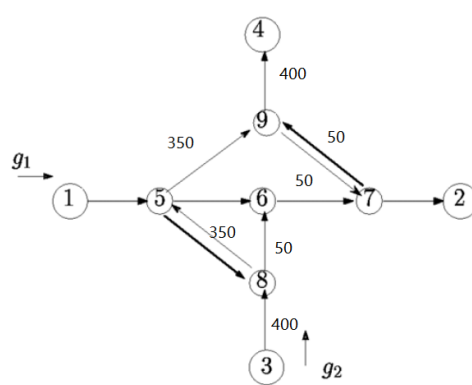
Flujo artículo 2

La función objetivo es minimizar el coste total para transportar el flujo de los nodos de entrada a los de salida, y cada arco tiene asignado un coste. Esta solución tiene una f.obj de 1232 que debería ser el óptimo dado que no hemos puesto ninguna restricción de capacidad en los arcos.

b.2) Ahora debemos restringir algún arco para que la solución encontrada por el solver sea diferente al del apartado anterior. Como hemos dicho antes, la solución anterior tiene una f.obj óptima, si forzamos que algún flujo no pase por la solución óptima (que en este caso muy probablemente solo haya una), el f.obj será mayor:



Flujo artículo 1

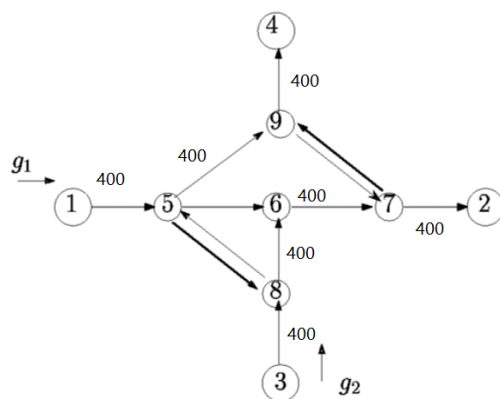


Flujo artículo 2

Forcé que pase como máximo 350 en (5,9) al observar que en la red de b.1) este

arco se transportaba 400. Dado que en cada artículo, nuestra entrada es 400, al restringir este nodo a 350 ya era suficiente para provocar un cambio, habrá 50 que se deberá de transportar por otra ruta. Y como habíamos dicho, este cambio hace que la nueva solución tenga un función objetivo mayor que el anterior, 1262 comprado con 1232. La razón es que el arco que restringimos probablemente tenga un coste bastante bajo comprado con otros.

- c) Este apartado nos pide resolver el problema de flujos multiartículo con un modelo de flujo con un único artículo (minCost).



El valor de f.obj ha cambiado, si nos fijamos en los flujo ahora, los 400 que van al nodo 4 proviene del nodo 1, y los 400 a 2 vienen del 3. Es decir 1->4 y 3->2, esto hace que tengamos ahora un f.obj de 1152, ya que tenemos más libertad en la selección de rutas, y el solver ha sido capaz de encontrar una mejor ruta que para los apartados anteriores.

Aparentemente puede parecer que ahora puede que tengamos un f.obj mayor que anteriormente, ya que tenemos 2 flujos de entrada compartiendo, pero esto no compensa el mayor grado de libertad.

