

Experimentos con el algoritmo aproximado de contracción para el corte mínimo en python

Francisco Gerardo Meza Fierro

1. Introducción

En esta práctica se estudia el flujo máximo que existe en un grafo ponderado propuesto. El grafo es el mismo propuesto de la práctica anterior [1]. A diferencia de la práctica anterior, lo que se busca es una aproximación al flujo máximo mediante un corte mínimo.

Igual que en la previa práctica, las imágenes que presenta esta práctica fueron obtenidas a través de `gnuplot` y `R` mediante archivos generados a través de `python`.

2. Contracción del grafo

Para contraer el grafo, se realizaron los siguientes pasos:

1. Se toman dos nodos aleatorios del grafo que forman una arista en el mismo, evitando que ambos sean los nodos fuente y sumidero.
2. Se calcula el punto medio entre ellos donde se crea un nuevo nodo.
3. En ese nuevo nodo se conectan los vecinos que eran de los dos nodos aleatorios.
4. Se repite desde el paso 1 hasta que queden dos nodos.

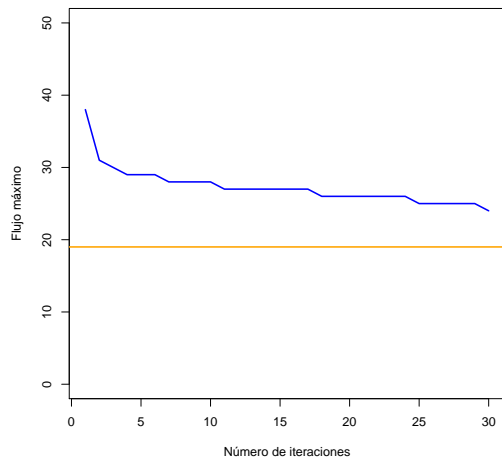
Para trabajar siempre sobre el mismo grafo y crear repeticiones que nos ayuden a comparar el flujo máximo óptimo del grafo con el flujo máximo aproximado sin alterar el primer grafo se usó la función `deepcopy()`.

3. Flujo máximo

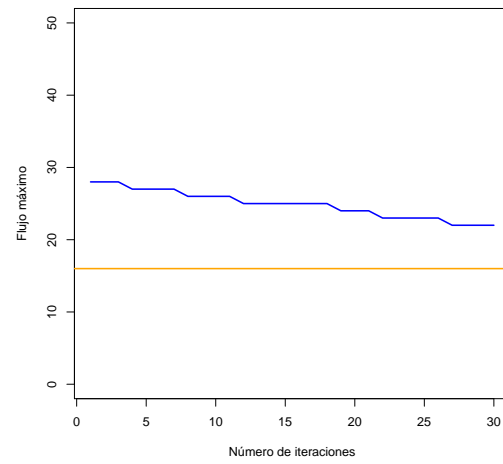
Para calcular el flujo máximo de los grafos, se implementó la función de Ford-Fulkerson.

Referencias

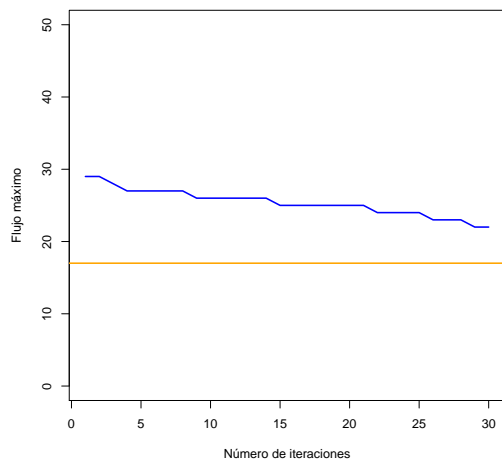
- [1] Francisco Meza. *FlujoEnRedes*
<https://github.com/Cicciofano/FlujoEnRedes>, 2018.



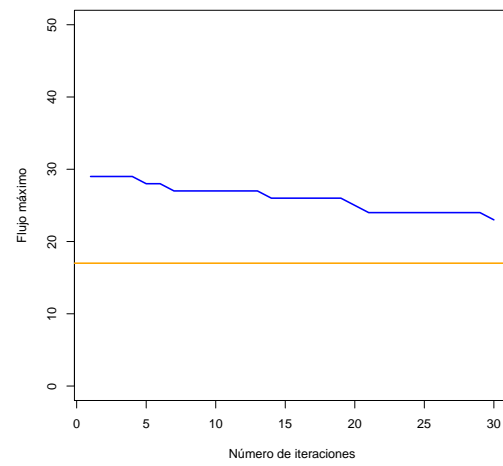
(a) Primer experimento.



(b) Segundo experimento.



(c) Tercer experimento.



(d) Cuarto experimento.

Figura 1: Comparación de los flujos máximos.