

Tarea 6: Generalizaciones

5171

4 de junio de 2019

Objetivo: Se describe un problema y su relevancia en el reporte, igual como el estado de arte de algoritmos para ello y sí o no se basa en la resolución del problema de flujo máximo. Se selecciona un generador de instancias, realizando modificaciones, se varía el orden de las instancias de tal forma que pueda estimar experimentalmente la complejidad asintótica del algoritmo creado. Se compara la complejidad observada experimentalmente con un análisis teórico de la complejidad asintótica que su algoritmo tiene, utilizando visualizaciones, cuadros y métodos estadísticos según necesidad para establecer la precisión y confiabilidad de los hallazgos

1. Problema

Supongamos que nos dan un conjunto de n proyectos que podríamos realizar, identificamos cada proyecto por un número entero entre 1 y n . Algunos proyectos no pueden ser iniciados hasta que otros proyectos se completen. Este conjunto de dependencias es descrito por un gráfico acíclico dirigido, donde una arista (i, j) indica que el proyecto i depende del proyecto j . Finalmente, cada proyecto i tiene un beneficio asociado p_i que se nos otorga si el proyecto se completa. Sin embargo, algunos proyectos tienen beneficios negativos, que interpretamos como costos. Podemos optar por terminar cualquier subconjunto X de los proyectos que incluya a todos sus dependientes; Es decir, para cada proyecto $x \in X$, cada proyecto del que depende x también está en X . Nuestro objetivo es encontrar un subconjunto válido de los proyectos cuyo beneficio total es lo más grande posible. En particular, si todos los trabajos tienen un beneficio negativo, la respuesta correcta es no hacer nada. Entonces definimos un grafo G un nodo origen s y nodo destino t en G , donde s tiene un costo o beneficio cero.

2. Estado del arte

La aplicación principal de este tipo de problemas es programación de la producción de open-pit mines, donde se han aplicado diferentes métodos de optimización para resolverlos. Entre ellos se encuentran:

- Dagdelen and Johnson (1986)
- Caccetta and Hill (2003)
- Ramazan (2007)

- Gershon, (1987)
- Tolwinski and Underwood, 1996

por mencionar algunos.

3. Instancias generadas

Se utilizo el algoritmo de acomodo spectral, para mejor visualización de los grafos, en la figura 1 se muestra un ejemplo de una instancia, con nodo fuente en verde, con nodo sumidero en rojo, ademas se les agregaron pesos.

Referencias

- [1] ELISA S. *Optimización de flujo en redes*, 2019.
<https://elisa.dyndns-web.com/teaching/opt/flow/>
- [2] ARIC A. HAGBERG, DANIEL A. SCHULT AND PIETER J. SWART *Exploring network structure, dynamics, and function using NetworkX*, 2008.
<https://networkx.github.io/documentation/stable/index.html>