



Resolución de problemas
con metaheurísticos

Formalización del problema de la partición del grafo

Laura Rodríguez Navas
rodrigueznava@posgrado.uimp.es

Mayo 2020

Resumen

In mathematics, a graph partition is the reduction of a graph to a smaller graph by partitioning its nodes into mutually exclusive groups. It is commonly used in scientific computing, VLSI circuit design, and task scheduling in multiprocessor computers, among others. Actually, finding partitions that simplifies graph analysis is a hard problem and the graph partitioning problem has gained importance due to its application for clustering and detection of cliques in social, pathological and biological networks.

Since graph partitioning is a hard problem, the practical solutions are based on heuristics, because typically graph partition problem fall under the category of NP-hard problems. For tackling this problem, I propose three algorithms based on heuristics, including local and global strategies. The two first algorithms presented are Kernighan–Lin and Fiduccia-Mattheyses, effective 2-way cuts by local search strategies. The third is the Spectral Bisection algorithm. Finally, I demonstrate with some examples my approach in an evaluation of these algorithms using a dataset.

Tabla de Contenidos

Resumen	1
1 Introducción a la Partición de Grafos	3
1.1 Introducción	3
1.2 Descripción el problema	3
2 Algoritmos basados en metaheurísticas	4
2.1 Kernighan-Lin	4
2.1.1 Descripción	5
2.1.2 Mejoras	5
2.1.3 Ejemplo	5
2.2 Spectral Bisection	5
2.2.1 Descripción	5
2.2.2 Ejemplo	5
2.3 Multilevel Spectral Bisection	5
2.3.1 Descripción	5
2.3.2 Ejemplo	5
2.4 Comparativa entre los diferentes algoritmos	5
3 Conclusiones	6
Bibliography	7

1. Introducción a la Partición de Grafos

1.1 Introducción

En los últimos años el problema de la partición de grafos ha sido ampliamente estudiado. El problema se deriva de situaciones del mundo real por lo que tiene aplicación en varias ramas de la ingeniería y de la investigación. La partición de grafos es una disciplina ubicada entre las ciencias computacionales y la matemática aplicada. El problema es un problema importante que tiene aplicaciones extensas en muchas áreas ya que se busca minimizar los costes o maximizar los beneficios a través de la correcta distribución de recursos.

La primera aplicación del problema fue en la partición de componentes de un circuito en un conjunto de tarjetas que juntas realizaban tareas en un dispositivo electrónico. Las tarjetas tenían un tamaño limitado, de tal manera que el dispositivo no llegara a ser muy grande, y el número de elementos de cada tarjeta estaba restringido. Si el circuito era demasiado grande podía ser dividido en varias tarjetas las cuales estaban interconectadas, sin embargo, el coste de interconexión era muy elevado por el que el número de interconexiones debía ser minimizado.

La aplicación descrita anteriormente fue presentada en [1], debido a Kernighan y Lin (1970), en la cual se define un algoritmo eficiente para encontrar buenas soluciones. En la aplicación, el circuito es asociado a un grafo y las tarjetas como subconjuntos de una partición. Los nodos del grafo son representados por los componentes electrónicos y las aristas forman las interconexiones entre los componentes y las tarjetas.

1.2 Descripción el problema

El problema de partición de grafos puede formularse como un problema de programación entera. Los problemas de programación entera generalmente están relacionados directamente a problemas combinatorios, esto genera que al momento de resolver los problemas de programación entera se encuentren restricciones dado el coste computacional de resolverlos; por ese motivo se han desarrollado algoritmos que buscan soluciones de una forma más directa, la restricción de los mismos es que no se puede garantizar que la solución encontrada sea la óptima.

Este problema ha sido demostrado como un problema NP-completo (<https://es.wikipedia.org/wiki/NP-hard>), lo que implica que las soluciones para él no pueden ser encontradas en tiempos razonables.

En el presente trabajo se pretende estudiar el algoritmo descrito en [1], [2] i

2. Algoritmos basados en metaheurísticas

2.1 Kernighan-Lin

Kernighan-Lin es un método para dividir un grafo que contiene nodos y vértices en subconjuntos separados que están conectados de manera óptima. Dado que se puede usar un grafo para representar una red eléctrica que contiene bloques, el algoritmo Kernighan-Lin se puede extender a la división de circuitos en subcircuitos.

Una aplicación importante del algoritmo es en los circuitos VLSI[1][3]. Concretamente se usa para encontrar un número mínimo de conexiones entre particiones para mejorar la velocidad o disminuir el consumo de energía.

Kernighan-Lin es un algoritmo:

- **Iterativo.** Esto significa que el grafo/circuito inicialmente ya está particionado, pero la aplicación de Kernighan-Lin intentará mejorar u optimizar la partición inicial. Kernighan-Lin es iterativo en lugar de constructivo.
- **Voraz.** Esto significa que el algoritmo hará cambios si hay un beneficio inmediato sin considerar otras formas posibles de obtener una solución óptima.
- **Determinista** porque se obtendrá el mismo resultado cada vez que se aplique el algoritmo. El mismo resultado será el mismo número de redes que cruzan la bisección, pero no necesariamente las mismas redes.

El algoritmo Kernighan-Lin[1], a menudo abreviado como K-L, es uno de los primeros algoritmos de partición de grafos y fue desarrollado originalmente para optimizar la colocación de circuitos electrónicos en tarjetas de circuito impreso para minimizar el número de conexiones entre las tarjetas.

El algoritmo K-L no crea particiones, sino que las mejora iterativamente. La idea original era tomar una partición aleatoria y aplicarle Kernighan-Lin. Esto se repetiría varias veces y se elegiría el mejor resultado. Mientras que para grafos pequeños esto ofrece resultados razonables, es bastante ineficiente para tamaños de problemas más grandes.

Hoy en día, el algoritmo se usa para mejorar las particiones encontradas por otros algoritmos. Como veremos, K-L tiene una visión algo "*local*" del problema, tratando de mejorar las particiones mediante el intercambio de nodos vecinos. Por lo tanto, complementa muy bien los algoritmos que tienen una visión más "*global*" del problema, pero tienden a ignorar las características locales. Un ejemplo de este tipo de algoritmos sería la partición espectral (ver sección 2.2).

C. Fiduccia y R. Mattheyses realizaron importantes avances prácticos en [2] quienes mejoraron el algoritmo de tal manera que una iteración se ejecuta en $O(n)$ en lugar de $O(n^2 \log n)$. A continuación, primero se describe el algoritmo original y después se discuten estas y otras mejoras.

2.1.1 Descripción

2.1.2 Mejoras

2.1.3 Ejemplo

2.2 Spectral Bisection

2.2.1 Descripción

2.2.2 Ejemplo

2.3 Multilevel Spectral Bisection

2.3.1 Descripción

2.3.2 Ejemplo

2.4 Comparativa entre los diferentes algoritmos

3. Conclusiones

Bibliography

- [1] B. W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *Bell System Technical Journal*, 49(2):291–307, 1970. 3, 4
- [2] C.M. Fiduccia and R.M. Mattheyses. A linear-time heuristic for improving network partitions. In *19th Design Automation Conference, Las Vegas, NV, USA, 14-16 June, 1982*. IEEE, 1982. 3, 4
- [3] C. P. Ravikumar and George W. Zobrist. *Parallel Methods for VLSI Layout Design*. Greenwood Publishing Group Inc., USA, 1995. 4