

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Technical report

GRAPH SPECTRAL PARTITIONING

by

Le Van Cuong

`cuong.lv212179M@sis.hust.edu.vn`

Supervised by

Dr. Nguyen Kiem Hieu

Spring 2023

1 Introduction

The spectral partitioning algorithm [1] was proposed in the context of computing a good ordering for the parallel factorization of sparse and symmetric matrices. The purpose is to distribute the work of matrix operations across multiple processors or computing nodes in a parallel computing system. Due to the computational efficiency, spectral partitioning has been used widely for graph partitioning. The algorithm is based on the eigenvalues (spectrum) and eigenvectors of the graph's Laplacian matrix to divide the graph into two or more disjoint subsets, or partitions. Generally, the Laplacian matrix of the graph is first constructed from the adjacency matrix and the degree matrix. The eigenvalues and eigenvectors of the Laplacian matrix are then computed, and based on these, the graph is divided into two initial partitions. Based on the edge separator (i.e., the set of all edges that connect the partitions), the vertex separator is calculated by a maximum matching.

The goal of this project is to understand how the spectral partitioning algorithm works and to re-implement it. According to the descriptions in the paper, the algorithm is implemented in C++ and the obtained results are identical to those reported by the authors. The source code is available at <https://github.com/cuonglvsoict/Graph-spectral-partitioning>, while the matrices used in the experiments can be found at <https://math.nist.gov/MatrixMarket/>.

2 Preliminaries

This section briefly introduces the graph partitioning problem and fundamentals of spectral graph theory.

Graph partitioning problem: Given an undirected graph $G = (V, E)$ on $|V| = n$ vertices indexed by $\{1, 2, \dots, n\}$. The goal is to find a vertex separator S that disconnects $G \setminus S$ into two parts (partitions) A and B such that $|S|$ and the cardinality difference between A and B are minimized.

Let A be the $n \times n$ adjacency matrix G . Then, A is given by

$$a_{uv} = \begin{cases} 1, & \text{if } (u, v) \in E \\ 0, & \text{otherwise.} \end{cases}$$

By convention, we let $a_{uu} = 0$ for all $u \in V$. Let D be the $n \times n$ degree matrix of G . Then, D is a diagonal matrix where d_{uu} is the degree of the vertex $u \in V$. Given A and D , the Laplacian matrix of the graph G is defined by $L = D - A$. Let $\lambda_1, \lambda_2, \dots, \lambda_n$ be the eigenvalues of L . The eigenvalues have some important characteristics as follows

- L has real eigenvalues $\lambda_1 = 0 \leq \lambda_2 \leq \dots \leq \lambda_n$.
- The number of zero eigenvalues of L is equal to the number of connected components of G . Thus, if G is connected graph then we have $\lambda_2 > 0$.

The second eigenvalue λ_2 is called algebraic connectivity, which related to the vertex and edge connectives of the graph G . Let $\mathbf{x} \in \mathbb{R}^n$ be the eigenvector corresponding to the eigenvalue λ_2 . The next section shows how \mathbf{x} is used to partition the graph G .

3 Spectral partitioning algorithm

Recall, our goal is to partition the vertices of G into two subsets A and B with nearly equal sizes, and the vertex separator S as small as possible. The spectral partitioning algorithm is as follows.

- Step 1.* Construct the Laplacian matrix L of graph G . Calculate the eigenvector \mathbf{x} corresponding to the second eigenvalue λ_2 . Calculate the median value x_m of elements of \mathbf{x} .
- Step 2.* Based on \mathbf{x} and x_m , divide V into two initial partitions, let say, A' and B' . The subset A' includes all vertices $i \in V$ with the corresponding component in the eigenvector $x_i \leq x_m$. The subset B' consists of all vertices $i \in V$ with $x_i > x_m$. If there are multiple vertices i with $x_i = x_m$, randomly assign them to A' and B' in such a way that $||A'| - |B'|| \leq 1$.

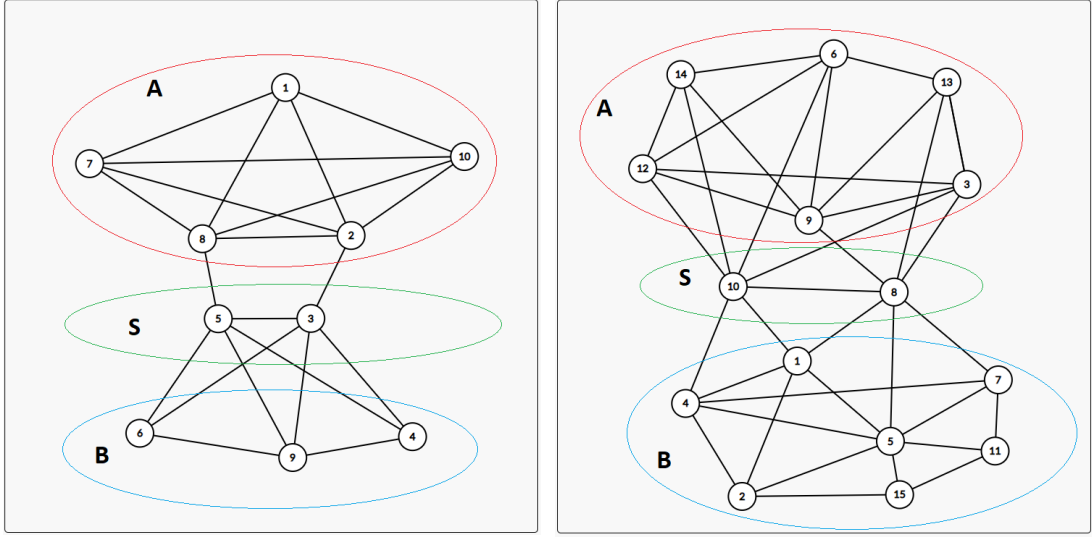


Figure 1: Results on small test instances, where A and B are two graph partitions and S is the vertex separator.

Step 3. Extract the edge separator E' including all the edges with one endpoint in A' and the other in B' . Let A_1 be the set of endpoints belonging to A' and B_1 be the set of endpoints belonging to B' . Then we have a bipartite $H = (A_1, B_1, E')$.

Step 4. Calculate the vertex separator S of G by finding the minimum vertex cover of the bipartite H . The minimum cover of H can be found through finding a maximum matching of H , which can be done in polynomial time since H is a bipartite.

Step 5. Calculate the graph' partitions as $A = A' \setminus S$ and $B = B' \setminus S$.

4 Numerical results

This section presents the numerical results to validate the implementation.

4.1 Results on small test instances

Two small graphs are constructed for testing during the implementation and visualization. The results obtained from the implemented algorithm are plotted in Figure 1. The figure shows that the algorithm successfully partitions the graphs into subsets such that the size of the separators S are minimized.

Table 1: Results on datasets used in the original paper

Dataset	Implemented by authors			Re-implemented		
	$ S $	$ A $	$ B $	$ S $	$ A $	$ B $
BCSPWR09	20	857	846	20	857	846
BCSPWR10	31	2623	2646	31	2623	2646
BCSSTK13	236	862	905	236	862	905
CAN 1072	33	525	514	33	525	514
JAGMESH	26	442	468	26	442	468

4.2 Results on datasets used in the original paper

This subsection reports the experimental results on several datasets that have been used by the authors in [1]. Noting that the stochastic factor of the spectral partitioning algorithm arises in *Step 3* (see, section 3), where there may exist multiple vertices $i \in V$ with their corresponding components in the second eigenvector equal to the median value x_m , and these vertices are randomly assigned into two initial partitions A' and B' . However, this is unlikely to happen in reality due to the nature of the eigendecomposition in *Step 1*. Therefore, the algorithm could be considered as a deterministic algorithm, and if the implementation is correct, the obtained results should be identical to the results reported in [1].

Five test instances are used in this experiments, i.e., five matrix graph downloaded from <https://math.nist.gov/MatrixMarket/>. The results are tabulated in Table 1, where the obtained partitions and vertex separators have the same size as in the original paper, i.e., Table 1 in [1].

References

- [1] Alex Pothén, Horst D Simon, and Kang-Pu Liou. “Partitioning sparse matrices with eigenvectors of graphs”. In: *SIAM journal on matrix analysis and applications* 11.3 (1990), pp. 430–452.