1 Introduction

Finding sparse cuts in graphs is an important problem in graph theory. A sparse cut is a subset $S$ of the vertices that has minimal conductance $\phi(S)$, defined as:

$$\phi(S) = \frac{w(S, \bar{S})}{\min\{w(S), w(T)\}}$$

where $w(S)$ is the total weight of all edges between vertices in $S$ and $w(S, T)$ is the weight of the edges between $S$ and $T$. By definition, relatively few edges leave a sparse cut compared to the edges that are contained within the vertices of the cut. As such, sparse cuts can be used to effectively divide a graph into multiple parts. In this way, finding multiple sparse cuts is related to finding clusters in a set of data points. Since the classic sparse cut problem is NP-hard and has applications in data analysis and image processing, approximating the sparsest cut has been an area of active research.

Spectral methods are theoretically appealing for generating graph cuts due to Cheeger’s inequality, which bounds the conductance of a graph using the second smallest eigenvalue of the normalized Laplacian of a graph.

**Cheeger’s Inequality:** For any graph $G = (V, E)$,

$$\frac{\lambda_2}{2} \leq \phi_G \leq 2\sqrt{\lambda_2}$$

where $\lambda_2$ is the second smallest eigenvalue of the normalized Laplacian, and $\phi_G$ is the conductance of the graph, defined as:

$$\phi_G = \min_{S: |S| \leq |V|/2} \phi(S)$$

Recent work has proven generalizations of Cheeger’s inequality using higher eigenvalues [1, 2]. These papers have provided algorithms to generate multiple cuts from higher eigenvalues. However, these algorithms have not been extensively tested against other methods of generating cuts, and it is likely that improvements can be made.
2 Project Overview

The project will be to implement the algorithm presented in [2], to test several improvements, and finally to test the final algorithm’s performance against other well-known algorithms for graph partitioning, such as METIS and GRACLUS. There are a number of important implementation details that are not explicitly described in [2], as well as several potential areas of improvement. Moreover, different spectral-based algorithms [1, 3] may offer some insights into how the algorithm in [2] might be improved.

One way to compare different cuts of a graph \( G = (V, E) \) is by using the normalized cut, defined as:

\[
NC(S_1, ..., S_k) = \sum_{i=1}^{k} \frac{w(S_i, \bar{S_i})}{w(S_i)} + \frac{w(T, \bar{T})}{w(T)}
\]

where the \( S_i \) are disjoint subsets of the vertices of the graph and \( T \) is the subset of the vertices not in any \( S_i \). Essentially, the normalized cut is the sum of the conductances of all of the subsets of \( V \) generated by the cut, where the remaining vertices are counted as an additional subset. This project will use the normalized cut to compare the quality of the cuts made by different algorithms. While the normalized cut is useful for comparing \( k \)-partitions of a graph, it does not work well when comparing partitions of different \( k \). Determining the best \( k \) for a graph is a highly nontrivial problem. One reach goal for this project would be to generate experimental data that may inform a solution to this problem, and perhaps to find the proper \( k \) in clear cases.

3 Deliverables

- Matlab implementation of a multiple sparse cut algorithm using higher eigenvalues
- Improvements on the algorithm
- Experimental results and comparisons with other standard algorithms such as METIS and GRACLUS
- Algorithm to determine the number of clusters in a graph in easy cases
• Written report on the above with a discussion of implementation details and experimental results

References

