K-WAY GRAPH PARTITIONING

REGINA AYUNITA TARIGAN
INSTITUT TEKNOLOGI DEL
regina.tarigan@del.ac.id

ABSTRACT

A partition on a graph with small number of vertices and edges does not require any complex procedure. Technically, one can find such graph partition by using the eigenvalues of Laplacian matrices given by such graph. The eigenvector corresponding to the second smallest eigenvalue is then used to find the sign of our eigenvector. Later on, these eigenvector is called the Fiedler vector. Furthermore for k-partition graph, we enlarge the concept by finding the third, fourth or bigger eigenvector to obtain three, four or more partitions but in fact there is always used only the second eigenvector.

Keywords : Graph, Partition, Symmetric Matrix, Eigenvalue, Eigenvector, Fiedler Matrix

INTRODUCTION

In general, a graph partition is the reduction of a graph to a smaller graph by partitioning its set of nodes into mutually exclusive groups. Edges of the original graph that cross between the groups will be taken away in the partitioned graph.

Definition 1. Graph and Symmetric Matrices. A graph G consists of a finite set of vertices $V(G)$ and a set of edges $E(G)$ consisting of distinct, unordered pairs of vertices (Bapat, 2010). In proper notation, we can state that $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ is called a graph where $\mathrm{V} = \{\mathrm{v}_1, \cdots, \mathrm{v}_\mathrm{n}\}$ is a set of vertices and $\mathrm{E} = \{\mathrm{v}_\mathrm{i}, \mathrm{v}_\mathrm{j}\}$ is the set of edges.

Defintion 2. Degree of a graph. The degree of a vertex in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex. The degree of the vertex v is denoted by $deg(v)$ (Rosen, 2011).

Definition 3. Symmetric Matrix. A square matrix A is said to be symmetric if $A = A^T$ (Anton and Rorres, 2014) .

Definition 4. Laplacian Matrix. Let G be a graph with $V(G) = \{1, \dots, n\}$ and $E(G) = \{e_1, \dots, e_m\}$. The Laplacian matrix of G, denoted by $L(G)$, is the $n \times n$ matrix defined as follows. The rows and columns of $L(G)$ are indexed by $V(G)$. If $i \neq j$, then (i,j)-entry of $L(G) is\ 0$ if vertex i and j are not adjacent, and it is -1 if i and j are adjacent. The (i,i)-entry of $L(G)$ is d_i, the degree of the vertex i, $i = 1, 2, \dots, n$ (Bapat, 2010).

The proper notation of the previous definition as given by Hwang.

$$L_{(ij)}(G) = \begin{cases} -1\ if\ (v_i, v_j) \in E\ and\ i \neq j \\ \deg(i) \qquad\qquad if\ i = j \\ 0 \qquad\qquad\qquad otherwise \end{cases}$$

(Hwang, 2015)

Definition 5. Eigenvalue and Eigenvector. If *A* is an $n \times n$ matrix, then a nonzero vector **x** in R^n is called an ***eigenvector*** of *A* (or of the matrix operator T_A) if *A***x** is a scalar multiple of **x**; that is,

$$A\boldsymbol{x} = \lambda \boldsymbol{x}$$

for some scalar *λ*. The scalar *λ* is called an ***eigenvalue*** of *A* (or of T_A), and **x** is said to be an ***eigenvector corresponding to λ*** (Anton, 2014).

Since the Laplacian matrix is symmetric, all eigenvalues are real. Furthermore, let the eigenvalues of L(G) be ordered $\lambda_0 = 0 \leq \lambda_1 \leq \cdots \leq \lambda_{n-1}$. An eigenvector corresponding to λ_0 is vector of all ones. This eigenvector is also called the

Fiedler vector. The multiplicity of λ_0 is equal to the number of connected components of the graph. The second smallest eigenvalue λ_1 is greater then zero iff G is connected (Kabelíkoǘa, 2006).

The second smallest eigenvalue becomes our concern when it comes to the graph partition. This value is then used to find the proper eigenvector. The example of this particular idea is given by following.

Example 1.
Given a graph G

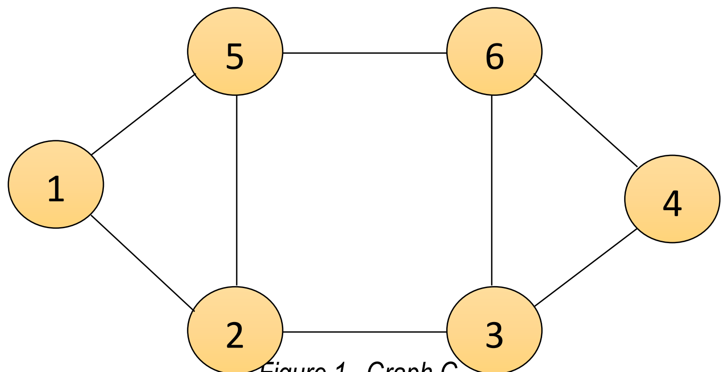

Figure 1. Graph G

The Fiedler matrices given by this graph is :

$$L_{ij}(G) = \begin{pmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 3 & -1 & 0 & -1 \\ 0 & 0 & -1 & 2 & 0 & -1 \\ -1 & -1 & 0 & 0 & 3 & -1 \\ 0 & 0 & -1 & -1 & -1 & 3 \end{pmatrix}$$

By using Matlab, we can find the eigenvalues and the eigenvector as following,

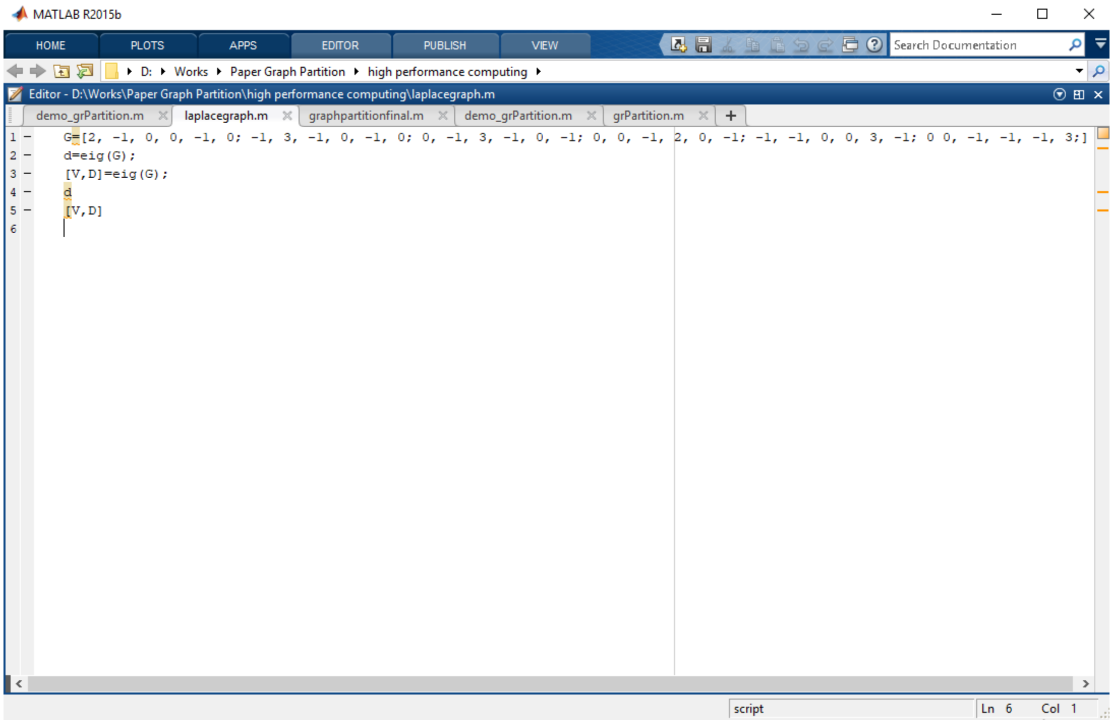

Figure 2. Matlab Code to find eigenvalue and eigenvector

hence the result is

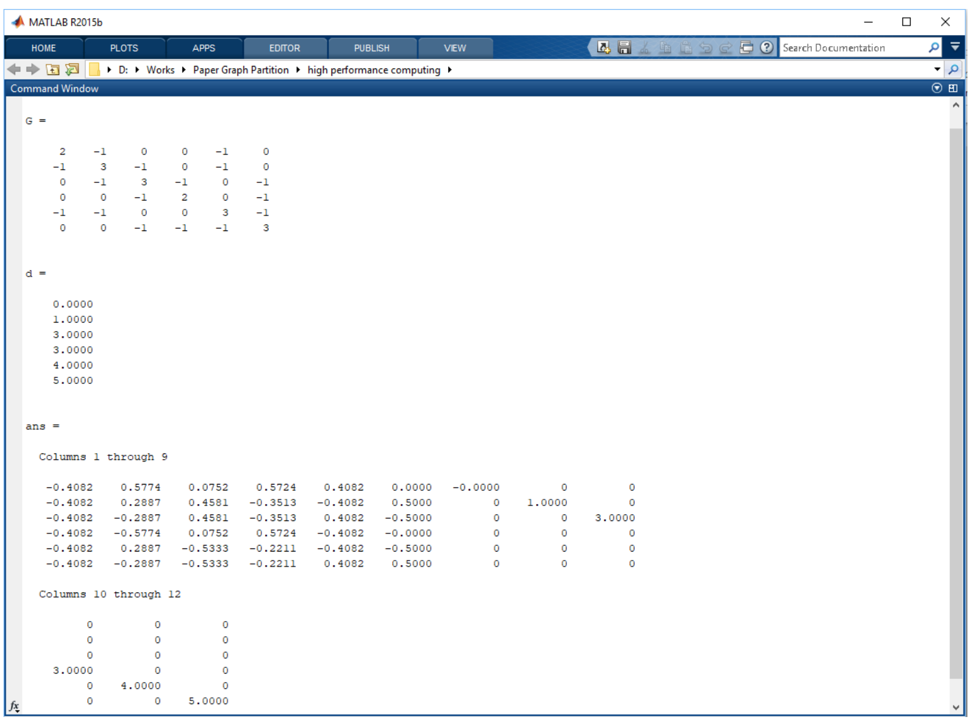

Figure 3. Matlab processing result

The eigenvector corresponding to the second smallest eigenvalue, which is 1.000 is given by :

$$(0.5774, 0.2887, -0.2887, -0.5774, 2.887, -0.2887)^{\mathrm{T}}$$

If we drop all the values and keep the signs of eigenvector corresponding to this second smallest eigenvalue, we obtain:

$$(+, +, -, -, +, -)$$

This is a perfect partition of the graph G into two subgraphs. One for "+" and one for "-". Using this ideam, we may partition G into G_1 and G_2 such that:

$G_1 = \{1, 2,5\}$ and $G_2 = \{3,4,6\}$

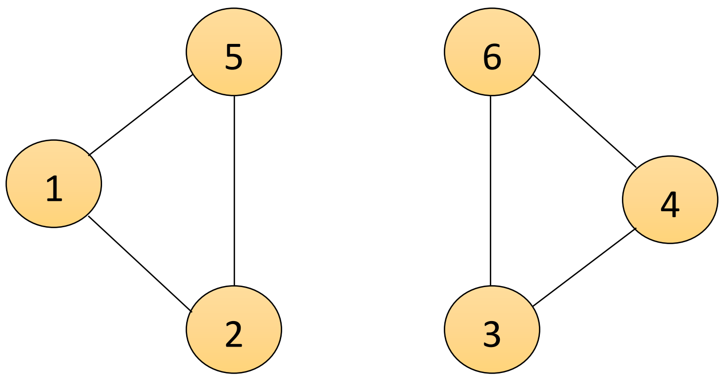

Figure 4. Graph G_1 and G_2 as the partition of graph G

Graph partitioning is actually a parallel processing that involving load balancing, and minimization of communication. In another word, it always finds as small as possible number of edges to take away in order to partition such graph into k parts.

Definition 6. Graph Partition. Furthermore, in mathematics, the graph partition problem is defined on data represented in the form of a graph

$$G = (V,E)$$

with *V* vertices and *E* edges, such that it is possible to partition *G* into smaller components with specific properties (Bapat, 2010). For instance, a *k*-way partition divides the vertex set into *k* smaller components.

A good partition is defined as one in which the number of edges running between separated components is small. Recently, the graph partition problem has gained importance due to its application for clustering and detection of cliques in social, pathological and biological networks.

A graphical partition is based on the eigenvalues and eigenvectors of the Laplacian matrix of a graph. The spectral partitioning algorithm is based on the intuition that the second lowest vibrational mode of a vibrating string naturally divides the string in half.

Applying that intuition to the eigenvectors of a graph we obtain the partitioning. The lowest eigenvalue of the Laplacian matrix is zero and corresponding eigenvector of all ones provides no information about the graph structure. More interesting is the second eigenvector (Fiedler vector). Dividing its elements according to the median we obtain two partitions (Example 1). We can also use the third, fourth or bigger eigenvector to obtain three, four or more partitions but in fact there is always used only the second eigenvector.

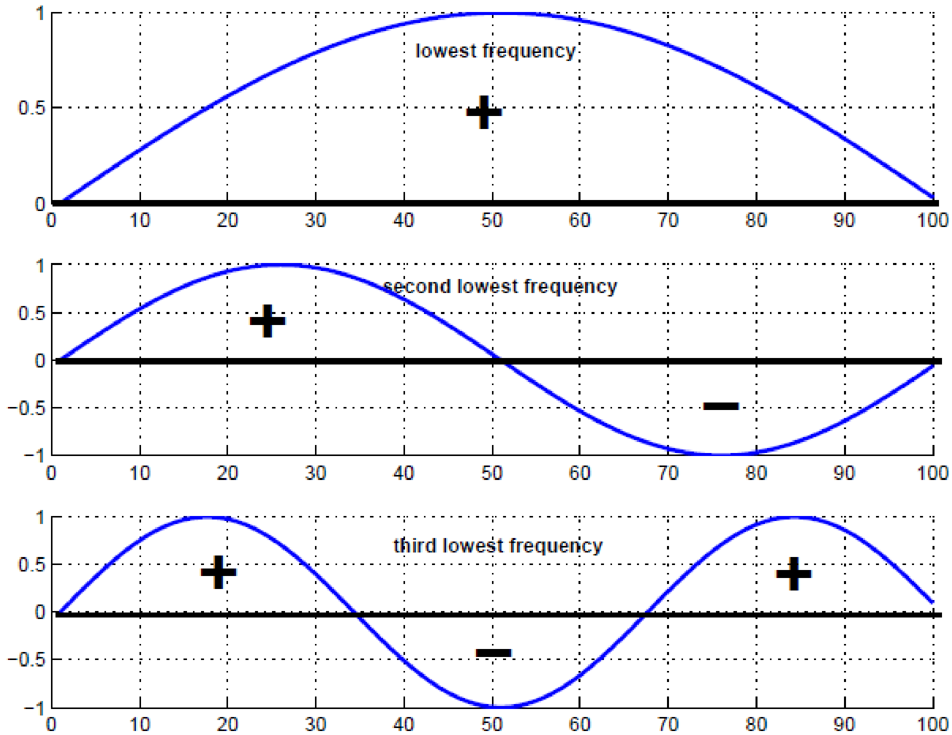

Figure 5. The frequences of vibrating string

RESEARCH METHOD

K-Partition Graph

We consider the *k*-partitioning problem, where the goal is to partition the vertices of an input graph G into k equally sized components, while minimizing the total weight of the edges connecting different components. We allow *k* to be part of the input and denote the cardinality of the vertex set by n. This problem is a natural and important generalization of well-known graph partitioning problems, including minimum bisection and minimum balanced cut.

Algorithm

Given a graph $G = (V, E)$ with $|V| = n$, partition V into k subsets, $V_1, V_2, \ldots, V_k$ such that $V_i \cap V_j = \emptyset$ for $i \neq j$, and $\bigcup_i V_i = V$, and the number of edges of E whose incident vertices belong to different subsets is minimized (Karypis, 1998).

A k-way partitioning of *V* is commonly represented by a partitioning vector P of length n, such that for every vertex $v \in V$, $P[v]$ is an integer between 1 and k, indicating the partition to which vertex v belongs. Given a partitioning P, the number of edges whose incident vertices belong to different partitions is callled the edge-cut of the partitioning.

The basic structure of a multilevel k-way partitioning algorithm is very simple. The graph $G = (V, E)$ is first coarsened down to a small number of vertices, a k-way partitioning of this much smaller graph is computed, and then this partitioning is projected back toward the original graph by successively refining the partitioning at each intermediate level.

Result

All the algorithms mentioned in the following work with second eigenvector and deal with graph bisection, that means they divide the vector set V into two disjoint sets: $V = V_1 \cup V_2$ that are equally large. To obtain more graph parts they could be applied recursively, until the parts are small and numerous enough. In fact, we try to find an edge separator E' - the smallest subset of E such that removing E' from E divides G into two disconnected subgraphs G_1 and G_2, with nodes V_1 and V_2 respectively.

Algorithm 1. Spectral bisection (Kabelíkoví, 2006)
Input : a graph $G = (V, E)$
Output : graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$
1. compute the Fiedler eigenvector v
2. search median of v
3. for each node i of G
   3.1. if $v(i) \leq$ median
        put node i in partition V_1
   3.2. else
        put node i in partition V_2
4. if $|V_1| - |V_2| > 1$ move some vertices with components equal to median from V_1 to V_2 to make this difference at most one
5. let V_1' 1 be the set of vertices in V_1 adjacent to some vertex in V_2
   let V_2' be the set of vertices in V_2 adjacent to some vertex in V_1
   set up the edge separator E'- the set of edges of G with one point in V_1' and the second in V_2'
6. let E_1 be the set of edges with both end vertices in V_1
   let E_2 be the set of edges with both end vertices in V_2
   set up the graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$
7. end

Hereby is the result of the matlab processing

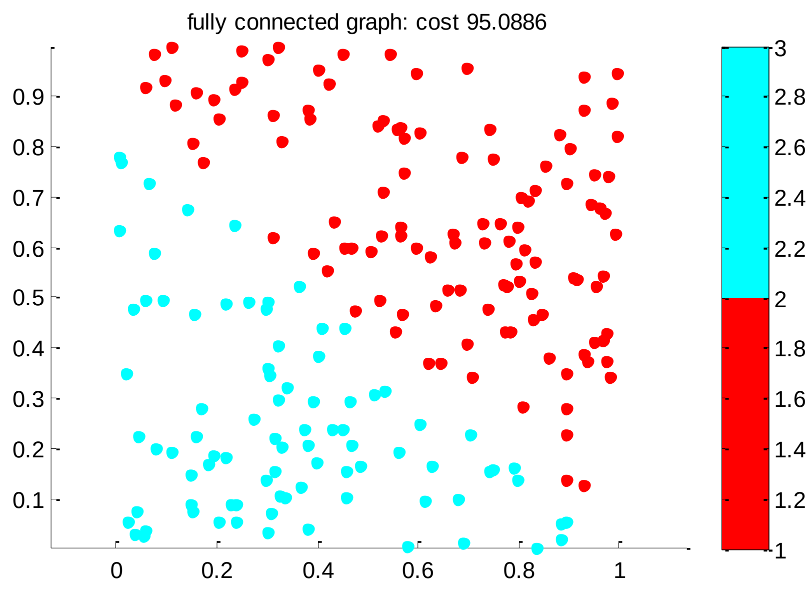

Figure 6. Partitioning of a fully connected graph with edge weights as a decreasing function of distance.

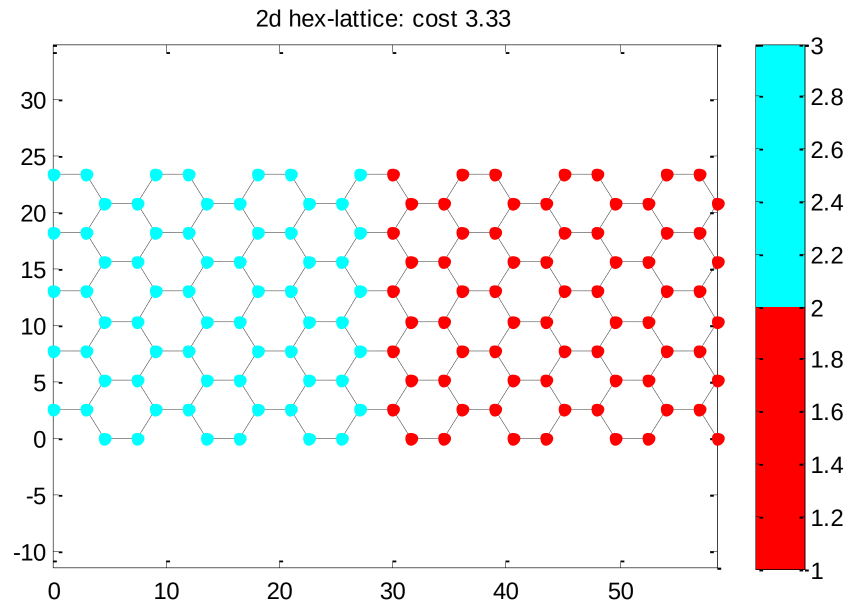

Figure 7. Partitioning of a 2-dimensional hexagonal lattice with edge weights as a decreasing function of distance.

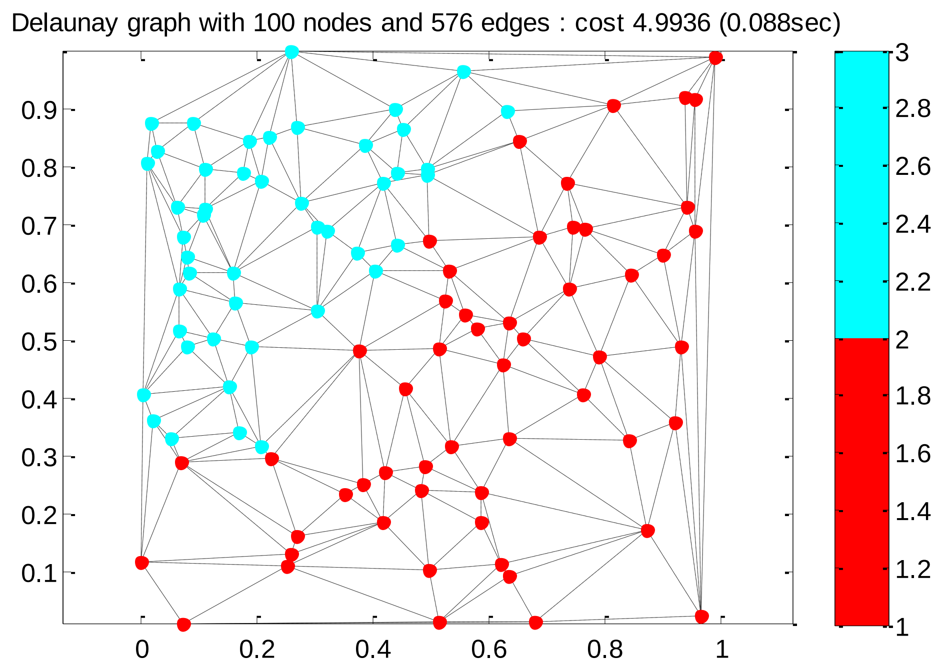

Figure 8. Partitioning of a Delaunay graph with edge weights as a decreasing function of distance.

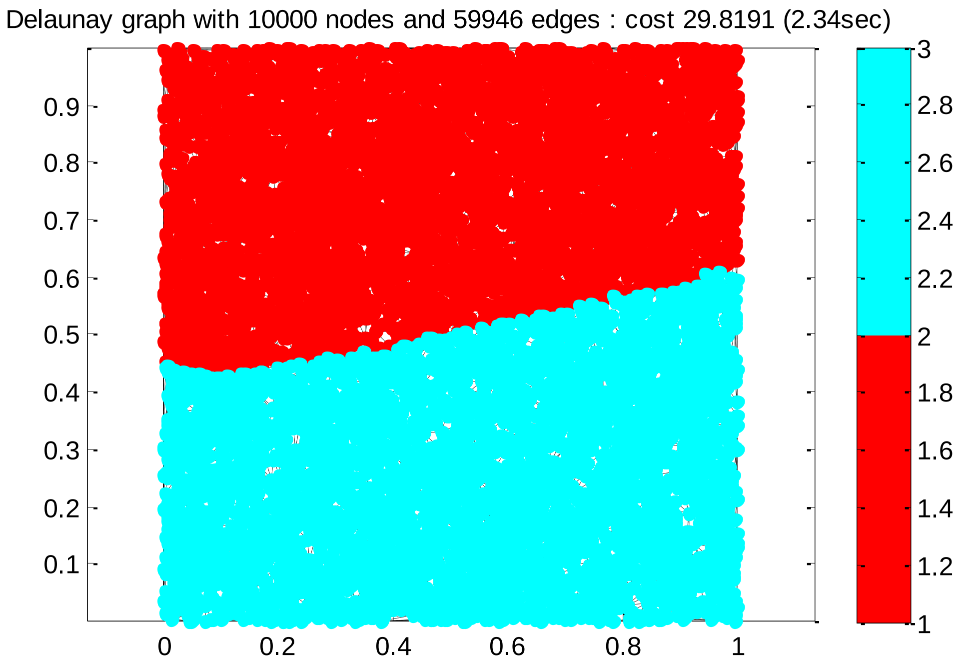

Figure 9. Same as Figure 8, but with a very large graph.

Let us consider a connected graph $G = (V, E), n = |V|$. K-way partitioning of G is a division of its vertices into k disjoint subset of size nearly equal to n/k. The main idea is to apply the spectral bisection as much as necessary, exactly until we have desired count of parts (say k).

There are two posibilities. When k is power of two, we use recursive bisection in its simple form. When k is not power of two, we have to use a modified recursive bisection.

Algorithm 2. Recursive bisection (Kabelíková, 2006)
Input: a graph $G = (V, E)$; an integer k (count of desired partitions)
Output: graphs $G_1 = (V_1, E_1), \dots, G_k = (V_k, E_k)$

1. apply Spectral bisection(G) (Agorithm 1) to find G_1, G_2
2. if $(k/2 > 1)$
   2.1. Recursive bisection $(G_1, k/2)$
   2.2. Recursive bisection $(G_2, k/2)$
3. return partitions $G_1, \dots G_k$
4. end

By using the algorithm concept, we can build such code in matlab to find a graph partition as following :

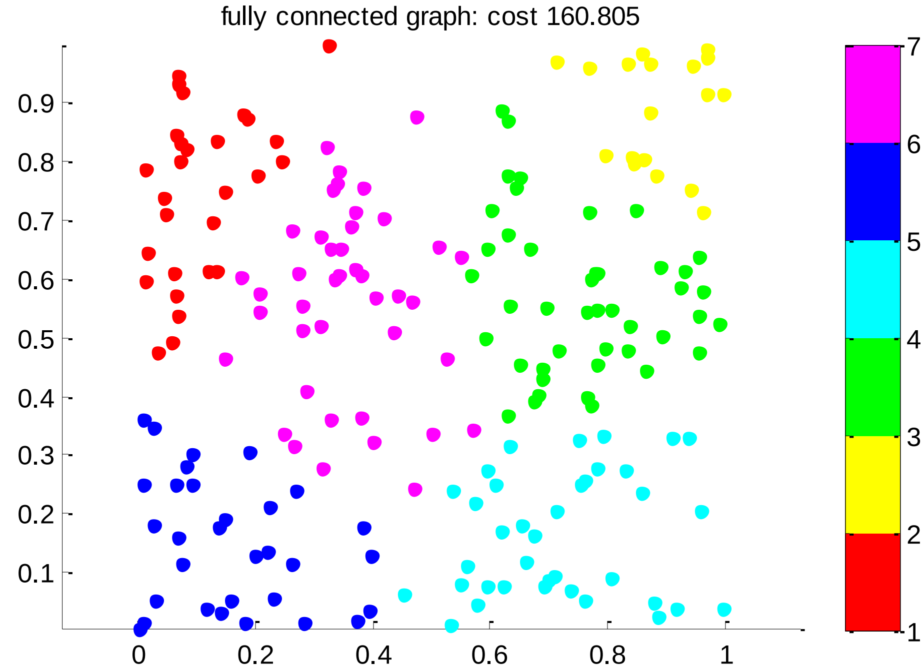

Figure 10. Partitioning of a fully connected graph with edge weights as a decreasing function of distance

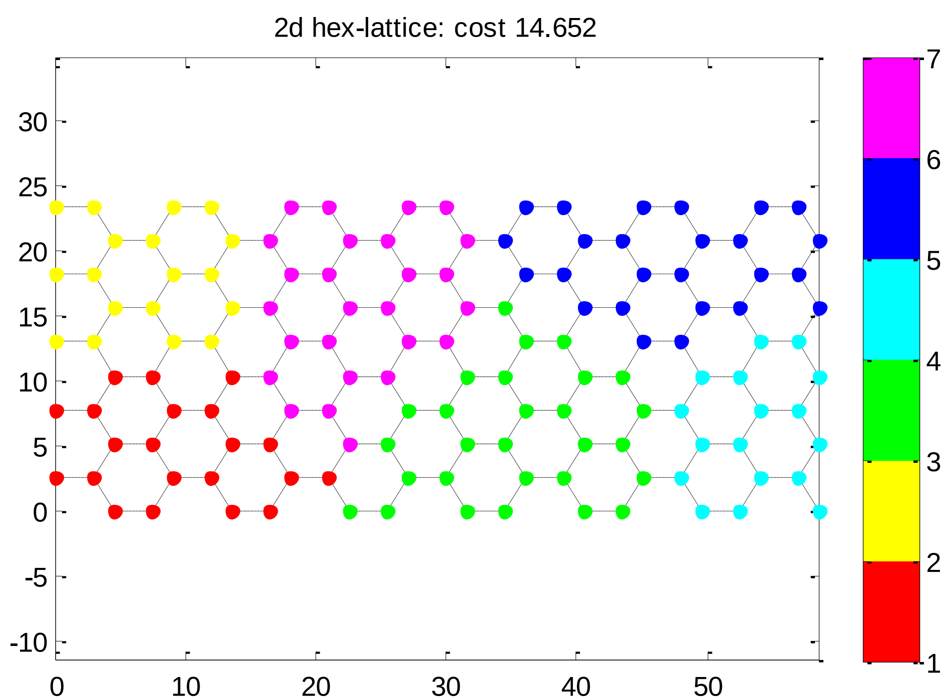

Figure 11. Partitioning of a 2-dimensional hexagonal lattice with edge weights as a decreasing function of distance.

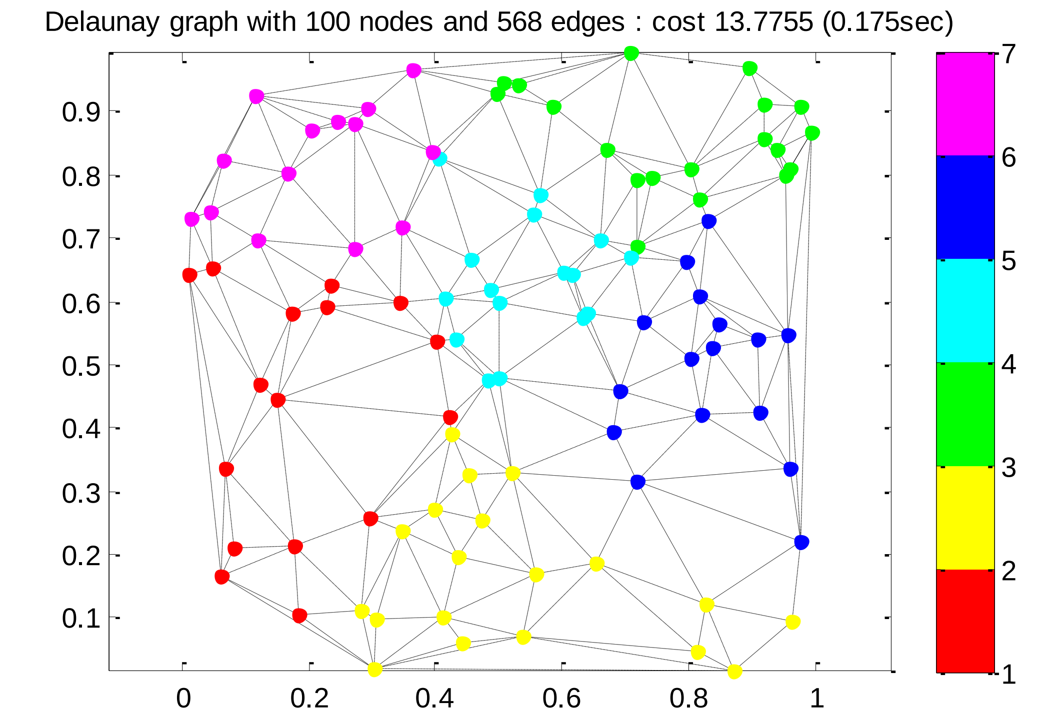

Figure 12. Partitioning of a Delaunay graph with edge weights as a decreasing function of distance.

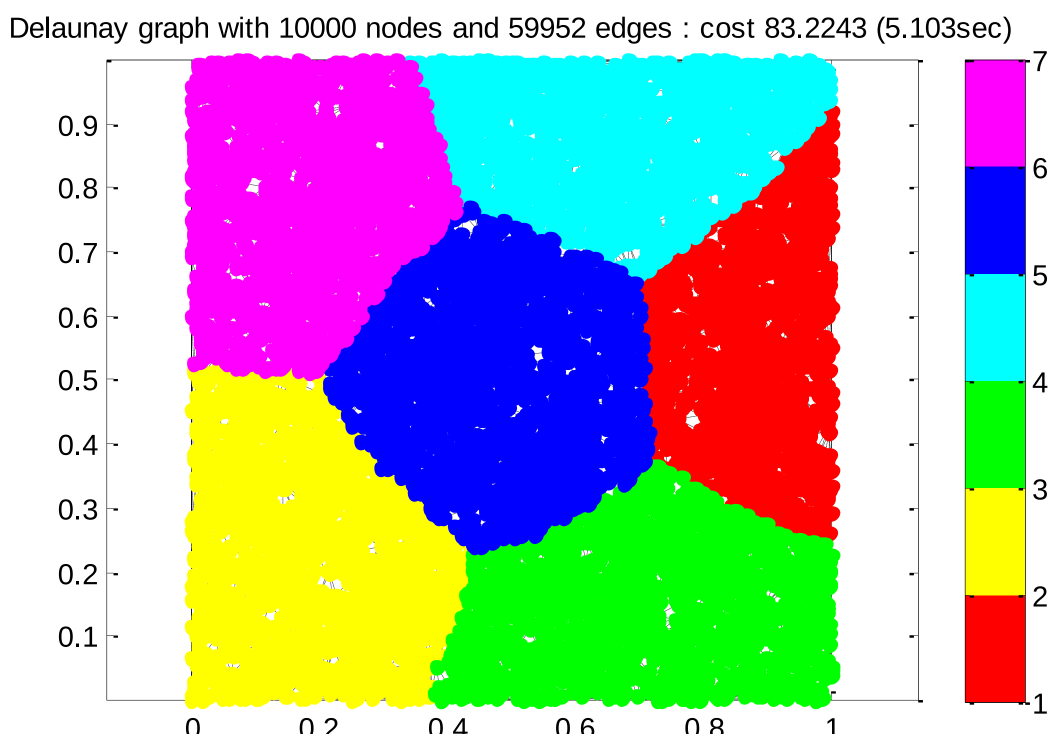

Figure 13. Same as Figure 12, but with a very large graph.

CONCLUSION

a. Only need to find the largest two eigenvectors approximately.
b. The method can be used again to re-partition the subgraphs.
c. The method often produce a partition with a short interface (minimizing the communication cost) and commonly equal weighted subgraph.

BIBLIOGRAPHY

Anton, Howard and Rorres, Chris. 2014. Elementary Linear Algebra Applications Version, Eleventh Edition. Canada: Wiley.
Bapat, R.B. 2010. Graph and Matrices. New Delhi: Springer with co-publication with Hindustan Book Agency. Hwang, Feng-Nan. 2015. Class Module: General Sparse Matrices: Storage, Partitioning and Preconditioning. Taiwan: Department of Mathematics, National Central University.Karypis, George and Kumar, Vipin. 1998. Multilevel k-way Partitioning Scheme for Irregular Graphs. Journal of Parallel and Distributed Computing. 48, PC971404, 96–129. Kabelíkoύa, Pavla. 2006. Thesis: Graph Partitioning Using Spectral Methods. Czech Republic: VŠB - Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science Department of Computer Science.
Rosen, Kenneth H. 2011. Discrete Mathematics and Its Applications, Seventh Edition. New York: McGrawHill.