Complexities of Using Graph Partitioning in Modern Scientific Problems and Application to Power System Islanding

Ali Peiravi, Rahim Ildarabadi

Ferdowsi University of Mashhad
Department of Electrical Engineering, School of Engineering, Mashhad IRAN
Telephone number: (0098) 511-881-5100
Ali_peiravi@yahoo.com, rahim.ildarabadi@yahoo.com

Abstract: The application of graph partitioning to modern scientific problems with various objectives has been attempted by many researchers in a variety of fields. Such applications are many and the following may be mentioned only to name a few examples. Applications such as vulnerability assessment of large power systems, power system islanding, design of VLSI circuits, dynamic system modeling and simulation, innovation graph state estimation, internet-like network partitioning, task mapping of parallel computation, database management, archaeological dating, power system transient studies, load balancing of parallel computing, molecular dynamics, DNA sequencing, categorizing amino acids, circuit netlists partitioning, etc. have been reported in the literature. In this paper, the complexities of the application of graph partitioning in modern scientific applications are thoroughly investigated in order to shed some light on this issue with such a diverse domain of applications. Fundamental graph theoretical and matrix algebraic concepts are discussed with sufficient examples. Application of these concepts to the problem of power system islanding is presented with suggestions to improve the speed and the objective function being used. [Journal of American Science 2009;5(5):1-12]. (ISSN: 1545-1003).

Key words: Graph partitioning, Fiedler vector, eigenvalues, Laplacian, Scientific applications, Power system islanding

1. Introduction

Since the development of graph theory, many applications have been sought by researchers in various fields. Perhaps one of the most important classes of applications of graph theoretic concepts to modern scientific problems is graph partitioning. This application is found under many disguises in the literature. Titles such as splitting, islanding, grouping, clustering, etc. are such names which have been adopted by the researchers in individual fields. The domain of application of these techniques is so vast that it encompasses almost all scientific fields. Therefore, an issue of such general application in science and engineering deserves a closer look.

Once the literature is reviewed, you find that graph partitioning has been applied to many problems such as power systems, electronics, communications, computers, genetics, image processing, etc. The applications are so many that one cannot list them all in one paper. However, a short survey of these applications is presented here in order to indicate the importance of taking a closer look at the complexities involved.

Graph partitioning can be applied in many modern day large scale system problems such as parallel processing, sparsity preserving orderings for sparse matrix factorizations, circuit placement, routing, system hierarchy, VLSI circuit testing, facility location, scattered network, hierarchical design of VLSI circuits, data mining, dynamic load balancing, parallel test pattern generation, power system islanding, annotation of protein sequence, and fault section estimation in large scale power systems.

To start, one may refer to the area of dynamic system modeling and simulation in the work of Rideout et al. (2009) who applied partitioning to detect weak coupling including one-way coupling or complete decoupling among elements of a dynamic system model. They attempted to partition the problem in order to reduce the models in which weak coupling is found so as to reduce the physical-domain model. This would enable one to perform parallel simulation of smaller individual submodels and reduce the computational time. They applied this method to the partitioning of the longitudinal and pitch dynamics of a medium-duty truck model. The intensity of dynamic coupling and the potential for model reduction are shown to depend on the magnitude of system parameters and the severity of inputs such as road roughness.

In applications to power systems, Bi et al. (2002) proposed a multiway graph partitioning method for partitioning a large scale power network into the desired number of connected subnetworks with the intention of fault section estimation. They proposed the balancing of work burden on each of the subnetworks formed. The proposed partitioning method minimizes the number of elements at the frontier of each subnetwork.

You et al. (2003) proposed a self-healing strategy to deal with catastrophic events when power system vulnerability analysis indicates that the system is approaching an extreme emergency state. They suggested the use of graph partitioning to adaptively divide the power system into smaller islands with consideration of quick restoration. In their approach, it was suggested that a load shedding scheme based on the rate of frequency decline be applied in the islands formed.

Kamwa et al. (2007) addressed the problem of dynamic vulnerability assessment of large power systems using a fuzzy clustering algorithm for partitioning the power system into a number of coherent electric areas. The objective function is imposed by selecting representative buses from the data set in such a way that the total fuzzy dissimilarity within each cluster is minimized. Zhang et al. (2008) used partitioning algorithms for innovation graph estimation to meet the management requirement of large-scale power grid to provide a simplified network in the upper-level control center to replace the original whole grid in two-level cooperative control centers on a provincial electrical network reducing the computational time requirements. Wang X. Z. et al. (2008) proposed an adaptive clustering algorithm based on power system network topology, initial power flow and given architecture to address power system transient stability studies. The sizes of the small cliques are derived using multi-constraint and multi-objective graph partitioning theory where the nodes represent units of computation, and the branches encode data dependencies. In a different work, Wang C. et al. (2008) presented a searching algorithm for islanding using a multilevel reduced graph partition algorithm. Peiravi and Ildarabadi (2009) proposed the use of multilevel kernel k-means partitioning for intentional islanding of power systems.

The applications in electronics are also outstanding. Hagen et al. (1992) addressed the problem of partitioning of circuit netlists in VLSI design. Using the well-known Fiedler vector, they presented a good approximation of the optimal ratio cut partition cost. Using Lanczos method for the sparse eigenvalue problem was found to be a robust basis for computing heuristic ratio cuts based on the Fiedler vector. They also considered the intersection graph representation of the circuit netlist as a basis for partitioning, and proposed a heuristic based on spectral ratio cut partitioning of the netlist intersection graph which was tested on industry benchmark suites.

Cherng et al. (1999) presented a two-level bipartitioning algorithm combining a hybrid clustering technique with an iterative partitioning process for VLSI circuits. Later on, Cherng and Chen (2003) presented a multi-level bipartitioning algorithm by integrating a clustering technique and an iterative improvement based partitioning process for VLSI circuit design in order to minimize the number of interconnects between the subsets of the circuit in order to reduce interconnect delays in deep submicron technology.

Application in other areas such as genetic engineering and image processing should also be mentioned. Shepherd et al. (2007) used the Fiedler vector for partitioning or categorizing amino acids based on the Miyazawa-Jernigan matrix. Their proposed model splits the amino acid residues into two hydrophobic groups (LFI) and (MVWCY) and two polar groups (HATGP) and (RQSNEDK). Othman et al. (2006) proposed the application of partitioning to assign highly correlated Gene Ontology terms of annotated protein sequences to partially annotated or newly discovered protein sequences. Their proposed method is based on Gene Ontology data and annotations. The first problem considered by them relates to splitting the single monolithic Gene Ontology RDF/XML file into a set of smaller files that are easy to assess and process so that they may be enriched with protein sequences and inferred from Electronic Annotation evidence associations. The second problem involves searching for a set of semantically similar Gene Ontology terms to a given query. Dhillon et al. (2005) and Dhillon et al. (2007) presented a graph partitioning method based on a Multilevel Kernel k-Means approach with a high speed performance in partitioning graphs which they used on large-scale partitioning tasks such as image segmentation; social network analysis; and gene network analysis types of systems.

The support for load balancing simulations that are performed on heterogeneous parallel computing platforms is an important issue and it can only be effectively achieved if the graph is distributed so that it properly takes into account the available resources such as CPU speed, network bandwidth, etc. One such application is load balancing in parallel computing to minimize communications between the various processors such as the parallel simulation of power system dynamics by Xue and Qi (2007) who used a multilevel graph partition algorithm and introduced regional characteristics into the partition and improved the weights of the nodes and its scheme. Moulitsas and Karypis (2006) developed algorithms that can address the partitioning requirements of scientific computations and can correctly model the architectural characteristics of emerging hardware platforms given that heterogeneous technologies are becoming more widespread.

2. Objectives of Graph Partitioning Applications

In practical applications, nodes and branches of the graphs to be partitioned represent different objects and this must be taken into consideration when developing the graph partitioning algorithm. Moreover, the objectives in mind in various applications of graph partitioning to modern scientific problems are somewhat different based on the demands of the particular application. To cover these differences, it would be wise to consider graphs with weights and costs assigned to the elements as suggested by Aleksandrov et al. (2006) who presented an algorithm for computing cutsets in planar graphs with costs and weights on the nodes, where weights are used to estimate the sizes of the partitions and costs are used to estimate the size of the cutset. They measured the quality of the partitioning by the total cost of the elements in the cutset and the imbalance between the total weights of the parts. In such applications, the weights assigned to the nodes are usually estimates of computational time requirements of the corresponding tasks. Once the node weights are balanced, the total computational burden is balanced amongst the various processors in the system. On the other hand, minimizing the branch cut sets implies minimum communications between the various processors in the system.

As another example of such challenges and complexities, let's consider the problem of power system islanding. The usual weight assigned for the branches in power system islanding applications represents the power flow in the corresponding transmission line, whereas the nodes represent power system buses. The underlying relationship between the nodes and the branches is also widely different in different applications. These issues must also be taken into consideration in the design of graph partitioning algorithms for a given application. There are, however, subtle difficulties in each application. For example, in the scheme presented by Wang C. et al. (2008), the authors only attempt to make the generation load imbalance in each island as small as possible. This is a minimal requirement for restoration. Blackouts may be caused for various reasons, even though power system partitioning or islanding application is meant to prevent them. Pre-disturbance conditions such as maintenance outages, changes in generation pattern and unexpected events such as misoperation of relays or failure of circuit breakers may pre-expose the system towards blackout after a disturbance. After the system breaks into islands, the load/generation imbalance in the islands could result in blackout in the individual islands. Therefore, it is reasonable to attempt to direct the intentional islanding of the power system towards islands with minimal load/generation imbalance. However, other factors such as voltage collapse, cascading thermal overloads, and dynamic stability could also lead to power system blackout. Therefore, more complexities and more strict conditions must be imposed to achieve a better islanding solution. For example, inherent structural characteristics of the power system should be considered and the choice of the island should be independent of the disturbance as proposed by Rehtanz (2003).

3. Mathematics of Graph Partitioning

Graph partitioning is usually based on graph theoretic concepts. In order to understand the complexities involved in modern scientific applications, a detailed analysis of the current approaches and the underlying algorithms is necessary. Graph partitioning is a well known NP-complete problem in mathematics where a graph is divided into several pieces in such a way that the pieces are of about the same size and there are few connections between them. The unweighted graph partition problem is usually stated as follows:

Given a graph G(N,B) with N nodes and B branches, and given an integer k >1, partition N into k subsets N1, N2, ... Nk such that the subsets are disjoint and have equal size, and the number of branches which end in different parts is minimized. In its more general form, weighted graph partitioning problem where both nodes and branches may be weighted, the problem may be stated as follows:

Given a graph G(N,B) with N nodes and B branches, and given an integer k >1, partition the graph into k disjoint subsets of approximately equal weight and the size of the branch cuts is minimized. The size of a cut set is the sum of the weights of the branches contained in it, while the weight of a subset is the sum of the weights of the nodes in that subset. This partitioning problem may be solved by using graph-theoretic heuristics.

The graph's adjacency matrix and the degree matrix are needed to form the Laplacian matrix. The Laplacian matrix of an undirected, unweighted graph G(N,B) where the graph is without any self loops or multiple branches between any pair of nodes is an n by n symmetric matrix with one row and column for each node defined by

$$L = D - A \tag{1}$$

where the degree matrix $D = diag(d_1, d_2, \Lambda\ , d_3)$ and A is the well-known adjacency matrix.

The Laplacian matrix is symmetric and positive semidefinite. This may be extended to weighted graphs. If the eigenvalues of the Laplacian of a graph are sorted by increasing value, the eigenvector corresponding to the second (smallest) eigenvalue of the Laplacian matrix is called Fiedler vector, and it may be used in heuristics for various graph manipulations including spectral graph partitioning. The second (smallest) eigenvalue of the Laplacian matrix is greater than 0 if and only if G(N,B) is a connected graph and the number of times 0 appears as an eigenvalue in the Laplacian represents the number of connected components in the graph. The magnitude of this value that is also known as the algebraic connectivity reflects how well connected the graph is. This may be used in the analysis of network sysnchronizability which has applications in many fields. Given that the number of nodes of a connected graph is n and its diameter is D, the algebraic connectivity of the graph G(N,B) is bounded below by $\frac{1}{nD}$.

Here, a simple example is used to illustrate the effort required in partitioning a graph using the Laplacian. Consider the simple weighted graph shown in Figure 1.

The Adjacency matrix for this graph is

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 2 \\ 1 & 2 & 0 \end{bmatrix}$$

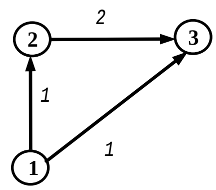

Figure 1 The graph with 3 nodes and 3 branches

The Degree matrix D for this graph is

$$D = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

The Laplacian matrix Q is

$$Q = D - A = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 3 & -2 \\ -1 & -2 & 3 \end{bmatrix}$$

Then the eigenvalues should be computed as shown below:

$$\det(Q - I\lambda) = \left\| \begin{bmatrix} 2-\lambda & -1 & -1 \\ -1 & 3-\lambda & -2 \\ -1 & -2 & 3-\lambda \end{bmatrix} \right\| = 0$$

$$\Rightarrow (2-\lambda)((3-\lambda)(3-\lambda)-4) + (-1)(-1)((-1)(3-\lambda) - (-1)(-2)) + (-1)((-1)(-2) - (-1)(3-\lambda)) = 0$$

$$\Rightarrow (-1)\lambda^3 + (2+3+3)\lambda^2 + (-3-12-18+4+2)\lambda + 36 - 8 - 3 - 2 - 5 \Rightarrow -\lambda^3 + 8\lambda^2 - 15\lambda = 0$$

$$\Rightarrow \lambda = 4 \pm \sqrt{16-15} = 4 \pm 1$$

$$\Rightarrow \lambda_1 = 0\ ,\ \lambda_2 = 3\ ,\ \ \lambda_3 = 5$$

Next the eigenvectors are computed as follows:

$$\Rightarrow AV_i = \lambda_i V_i \quad i = 1,2,3\ ,\ V_i = \begin{bmatrix} V_{1i} \\ V_{2i} \\ V_{3i} \end{bmatrix}$$

The first eigenvector V_1 : will be:

$$\Rightarrow \begin{bmatrix}2 & -1 & -1\\ -1 & 3 & -2\\ -1 & -2 & 3\end{bmatrix}\begin{bmatrix}V_{11}\\ V_{21}\\ V_{31}\end{bmatrix} = 0\begin{bmatrix}V_{11}\\ V_{21}\\ V_{23}\end{bmatrix}$$

$$\Rightarrow \begin{cases}2V_{11} - V_{21} - V_{31} = 0\\ -V_{11} + 3V_{21} - 2V_{31} = 0\\ -V_{11} - 2V_{21} + 3V_{31} = 0\end{cases}$$

$$\Rightarrow V_{21} = 2V_{11} - V_{31} \Rightarrow \begin{cases}-V_{11} + 6V_{11} - 3V_{31} - 2V_{31} = 0\\ -V_{11} - 4V_{11} + 2V_{31} + 3V_{31} = 0\end{cases}$$

$$\Rightarrow \begin{cases}5V_{11} - 5V_{31} = 0\\ -5V_{11} + 5V_{31} = 0\end{cases}$$

$$\Rightarrow V_{11} = V_{13} \Rightarrow V_{21} = V_{11}$$

Suppose that $V_{11} = 1$

$$\Rightarrow V_{21} = V_{31} = 1 \Rightarrow V_1 = \begin{bmatrix}1\\ 1\\ 1\end{bmatrix}$$

$$\hat{V}_1 = \begin{bmatrix}\frac{1}{\sqrt{3}}\\ \frac{1}{\sqrt{3}}\\ \frac{1}{\sqrt{3}}\end{bmatrix} = \begin{bmatrix}0.5774\\ 0.5774\\ 0.5774\end{bmatrix}$$

Normalized

The second eigenvector V_2 or the Fiedler vector is

$$\begin{bmatrix}2 & -1 & -1\\ -1 & 3 & -2\\ -1 & -2 & 3\end{bmatrix}\begin{bmatrix}V_{11}\\ V_{22}\\ V_{32}\end{bmatrix} = 3\begin{bmatrix}V_{12}\\ V_{22}\\ V_{32}\end{bmatrix}$$

$$\Rightarrow \begin{cases}2V_{12} - V_{22} - V_{32} = 3V_{12}\\ -V_{12} + 3V_{22} - 2V_{32} = 3V_{22}\\ -V_{12} - 2V_{22} + 3V_{32} = 3V_{32}\end{cases}$$

$$\Rightarrow \begin{cases}V_{12} = -V_{22} - V_{32}\\ -V_{12} = 2V_{32}\\ -V_{12} = 2V_{22}\end{cases}$$

$$\Rightarrow V_{12} = 2 \Rightarrow V_{23} = -1 \text{ , } V_{22} = -1$$

$$\Rightarrow V_2 = \begin{bmatrix}2\\ -1\\ -1\end{bmatrix} \Rightarrow \hat{V}_2 = \begin{bmatrix}\frac{2}{\sqrt{5}}\\ -\frac{1}{\sqrt{5}}\\ -\frac{1}{\sqrt{5}}\end{bmatrix} = \begin{bmatrix}0.8165\\ -0.4082\\ -0.4082\end{bmatrix}$$

And the third eigenvector V_3 is:

$$\begin{bmatrix}2 & -1 & -1\\ -1 & 3 & -2\\ -1 & -2 & 3\end{bmatrix}\begin{bmatrix}V_{13}\\ V_{23}\\ V_{33}\end{bmatrix} = 5\begin{bmatrix}V_{13}\\ V_{23}\\ V_{33}\end{bmatrix} \Rightarrow \begin{cases}2V_{13} - V_{23} - V_{33} = 5V_{13}\\ -V_{13} + 3V_{23} - 2V_{33} = 5V_{23}\\ -V_{13} - 2V_{23} + 3V_{33} = 5V_{33}\end{cases}$$

$$\Rightarrow \begin{cases}3V_{13} = -V_{23} - V_{33}\\ \frac{V_{13}}{3} = -V_{23} - V_{33}\\ \frac{V_{13}}{3} = V_{23} + V_{33}\end{cases} \Rightarrow V_{13} = 0 \text{ , } V_{23} = -V_{33} \Rightarrow V_{23} = -V_{33}$$

$$\Rightarrow V_{23} = 1 = -V_{33}$$

$$\Rightarrow V_3 = \begin{bmatrix}0\\ 1\\ -1\end{bmatrix} \Rightarrow \hat{V}_3 = \begin{bmatrix}0\\ \frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{bmatrix} = \begin{bmatrix}0\\ 0.7071\\ -0.7071\end{bmatrix}$$

Normalized

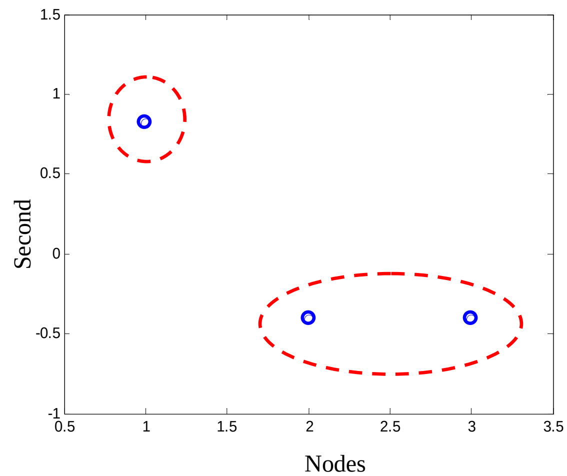

Figure 2 A sketch showing how the graph may be partitioned using the Fiedler vector

$$\Rightarrow V = \begin{bmatrix}0.5774 & 0.8165 & 0\\ 0.5774 & -0.4082 & 0.7071\\ 0.5774 & 0.4082 & -0.7071\end{bmatrix}$$

The second column of this matrix is the Fiedler vector and it may be used to bipartition the graph as shown in Figure 2.

The bipartitioned form of the graph in Figure 1 is shown in Figure 3 based on the Fiedler vector which indicates that the graph should be partitioned into two partitions with node 1 in one partition and nodes 2 and

3 in the second partition. In this simple illustrative example, it can be seen that the minimum cut set found {(n1,n2), (n1,n3)} has a total flow of 2 which is clearly the minimum of all possible cut sets.

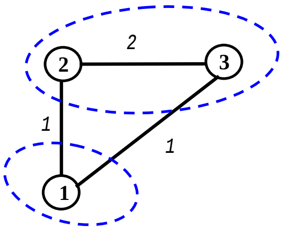

Figure 3 The bipartitioned graph with three nodes and three branches

One complexity involved in this basic procedure is the amount of mathematical effort required to compute the eigenvalues and the eigenvectors as illustrated by the example. In large scale systems where thousands of nodes are involved, the time complexity becomes burdensome and makes a solution practically impossible to obtain. The computational bottleneck of this partitioning procedure lies in the eigenvector calculation. Notice that since only the sign of each component of the Feidler vector is needed in order to partition the graph, an exact answer is not really required. This could be potentially useful in finding a fast solution approach. In applications where the Laplacian matrix is dense, there exist routines such as eig in Matlab that require (4/3)*|N|^3 time. However, in applications in which the graph has relatively few connections compared to a complete graph, this would not be computationally wise. In such cases, it is more suitable to resort to the Lanczos algorithm which is an iterative algorithm. For an n-by-n sparse symmetric matrix L, the Lanczos algorithm computes a k-by-k symmetric tridiagonal matrix T, whose eigenvalues are good approximations of the eigenvalues of T. The eigenvectors of T may be used to get approximate eigenvectors of A. The most computationally expensive part of this algorithm is in building T that requires k matrix-vector multiplications with L. Since the largest and the smallest eigenvalues of L including λ_2 converge first, a good approximation can be obtained given k much smaller than n.

The Lanczos algorithm applies to Hermitian matrices and transforms the original matrix into a tridiagonal matrix that is real and symmetric, $T_{mm} = V_m^* L V_m$ whose diagonal elements are denoted by $\alpha_j = t_{jj}$, and the off-diagonal elements are denoted by $\beta_j = t_{j-1,j}$ Moreover, the terms $t_{j-1,j} = t_{j,j-1}$ due to symmetry.

The following iterations should be performed:

Assume that $v_0 = 0$, v_1 is a random vector with norm 1, and $\beta_1 = 0$.

For j=1 to m

$$w_j = Lv_j - \beta_j v_{j-1}$$

$$\alpha_j = (w_j, v_j)$$

$$w_j = w_j - \alpha_j v_j$$

$$\beta_{j+1} = \| w_j \|$$

$$v_{j+1} = \frac{w_j}{\beta_{j+1}}$$

Return

The next step would be to find the eigenvalues of the matrix T as follows:

$$T = \begin{bmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_2 & & & \\ & \beta_3 & . & . & & \\ & & . & . & \beta_{m-1} & \\ & & & \beta_{m-1} & \alpha_{m-1} & \beta_m \\ & & & & & \alpha_m \end{bmatrix} \qquad (2)$$

The eigenvalues of T are $\lambda_1^{(m)}$ and the corresponding eigenvectors $u_1^{(m)}$ can now be easily computed. The Lanczos vectors $V = (v_1, v_2, v_3, \Lambda, v_m)$ generated through the above iterative approach may be used to compute them. The eigenvalues thus computed are approximations of the eigenvalues of L.

$$y_i = V_m u_i^{(m)}$$

where V_m is the transformation matrix whose column vectors are $v_1, v_2, v_3, \Lambda, v_m$. The set of vectors $v_1, v_2, v_3, \Lambda, v_m$ forms an orthogonal basis which would yield good approximate eigenvalues and eigenvectors for the original matrix if precise arithmetic is used. However, due to round off errors introduced during the computations, the Lanczos algorithm suffers from lack of numerical stability, and measures to prevent the loss of orthogonality must be adopted. One may periodically reorthogonalize the vector v against all previous ones. Since this would take a lot of time to do, one must estimate the degree of nonorthogonality and reorthogonalize only when needed.

It is interesting to note that although the real eigenvectors are not really needed in partitioning applications, one may compute them from the eigenvectors of T as follows

$$q_i = [v_0 \quad v_1 \quad . \quad . \quad . \quad v_m] w_i \tag{3}$$

where the v's are the Lanczos vectors. Many researchers have based their partitioning algorithms upon the Lanczos technique which uses the approximate Fiedler vector.

4. Pioneering Graph Partitioning Heuristics

The problem of partitioning nodes of a graph with costs associated with the branches into subsets no larger than a given maximum size with the objective of minimizing the total cost of the branch cutest was first considered by the pioneering work of Kernighan-Lin (1970). A typical application of this heuristic would be the placement of the components of electronic circuits onto various printed circuit boards with the objective of minimizing the number of inter-board connections. The limitation of each board in terms of the maximum number of components which can be placed on it should also be considered. They proposed an iterative, 2 way, and balanced minimum cutset partitioning heuristic. In this procedure,

a) The node pairs that yield the largest decrease or the smallest increase in the size of the cutest are exchanged.

b) The nodes are then locked so that they may not participate in any further exchanges.

c) The above procedure is repeated until all the nodes are locked.

d) The set with the largest partial sum is found for swapping.

e) All nodes are unlocked.

The details of this procedure are outlined in Figure 4.

There are several complexities involved in any graph partitioning problem. Kernighan and Lin (1970) introduced the probability that a heuristic procedure finds an optimal solution in a single trial as one such consideration. They concluded that this probability is around 0.5 for matrices of size 30x30, 0.2 to 0.3 for matrices of size 60x60 and from 0.05 to 0.1 for matrices of size 120x120. They assumed 50% dense matrices in these estimates. Although many modern day scientific applications of graph partitioning involve matrices that are much less dense, the size of the matrices involved is much higher than the ones considered in these estimates. This indicates that the challenge still remains a viable one since finding an optimal solution in a short time is a vital issue especially in real time applications. Another

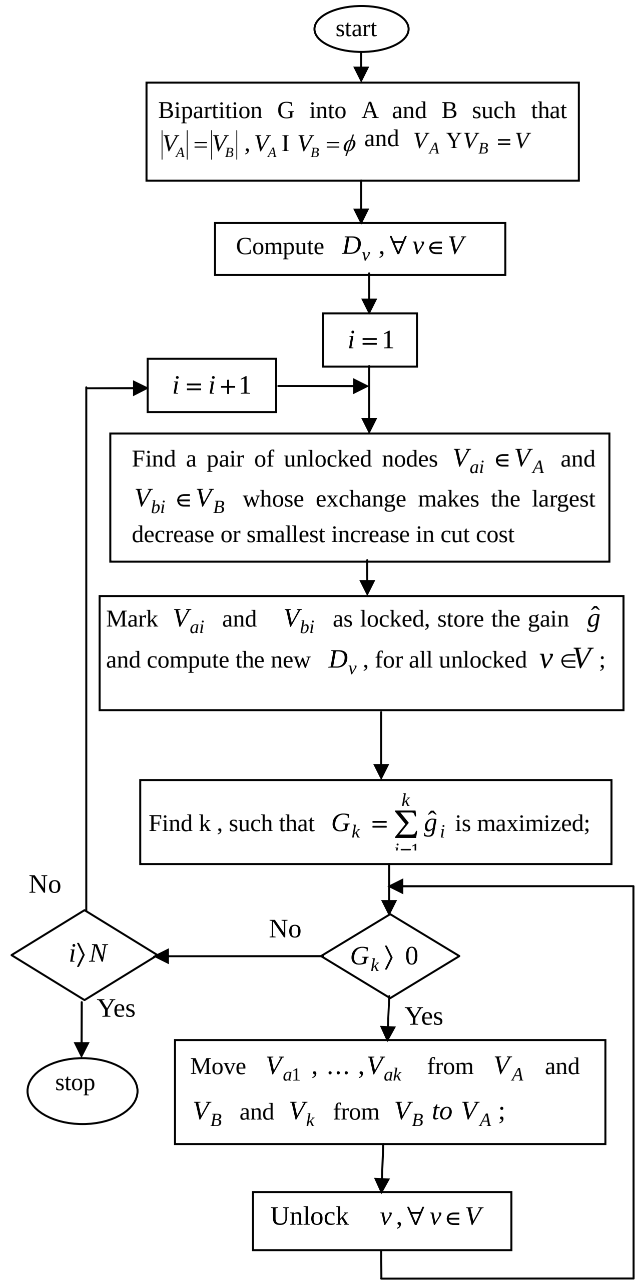

Figure 4 The flowchart of the first phase of Kernighan-Lin partitioning heuristic

complexity is the running time of the procedure in the Kernighan-Lin partitioning procedure. The total time for a single pass involves computation of the D values initially which is in the order of n^2, updating the D values that is proportional to n^2, and sorting D values is in the order of $n^2 \log n$. Since the number of passes required is estimated to be a few before a phase 1 optimal partition is found, the total running time for phase 1 is in the order of $n^{2.4}$. Another complexity of the Kernighan-Lin partitioning heuristic is the restriction imposed in the size of the partitions being exactly equal. Although this may be useful in some applications, it is certainly not a requirement of all modern day scientific applications of partitioning. Other factors exist which are more important than the exact size of the partition. For example, in power system islanding applications, it is more desirable to have stable islands after partitioning with minimal load/generation imbalance in each island formed than to force them to be exactly of equal size. This restriction may not even be practically possible to enforce in all power networks. Kernighan and Lin (1970) addressed this issue themselves in their original work and proposed partitioning into unequal subsets. Yet a further complexity is the restriction of having two partitions, which should also be relaxed to several partitions in some applications. This was also briefly noted in their work.

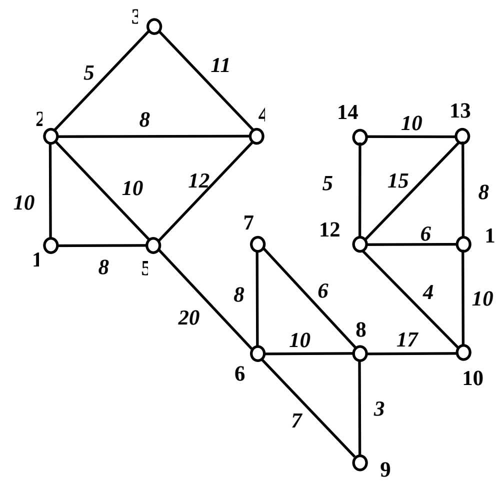

Figure 5 The graph of a 14-bus power system with 21 branches

5. An example of power system partitioning using the Fielder vector

To gain a better feel for the complexity of the work, an example is presented here to demonstrate the application of the spectral method for partitioning a power system graph into 3 partitions, using the smallest 3 eigenvalues. The weighted graph of a 14-bus power system is shown in Figure 5.

The Laplacian matrix fro the above graph is as follows where e_{ij} is weight of the branches that connect the two nodes v_i and v_j (note that $e_{ii} = 0$). Therefore, the Laplacian matrix is as shown below:

$$Q_{i,j} = \begin{cases} \sum_{h=1}^{14} e_{ih} & ; i = j \\ -e_{ij} & ; i \neq j \end{cases}$$

$$Q = \begin{bmatrix} 18 & -10 & 0 & 0 & -8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -10 & 33 & -5 & -8 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -5 & 16 & -11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -8 & -11 & 31 & -12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -8 & -10 & 0 & -12 & 50 & -20 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -20 & 55 & -8 & -20 & -7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -8 & 14 & -6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -20 & -6 & 46 & -3 & -17 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -7 & 0 & -3 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -17 & 0 & 31 & -10 & -4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -10 & 24 & -6 & -8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -4 & -6 & 30 & -15 & -5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -8 & -15 & 33 & -10 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -5 & -10 & 15 \end{bmatrix}$$

and the eigenvalues are:

[λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_9	λ_{10}	λ_{11}	λ_{12}	λ_{13}	λ_{14}]=
[0	1.62	5.31	11.27	13.67	14.80	24.46	28.94	33.02	40.47	42.95	47.88	58.76	82.83]

The second and third eigenvectors of the Laplacian of the graph are shown in Figure 6 which shows that this graph can be divided into three partitions as follows: [Partition 1: {1, 2, 3, 4, 5}, Partition 2: {6, 7, 8, 9}, Partition 3: {10, 11, 12, 13, 14}]

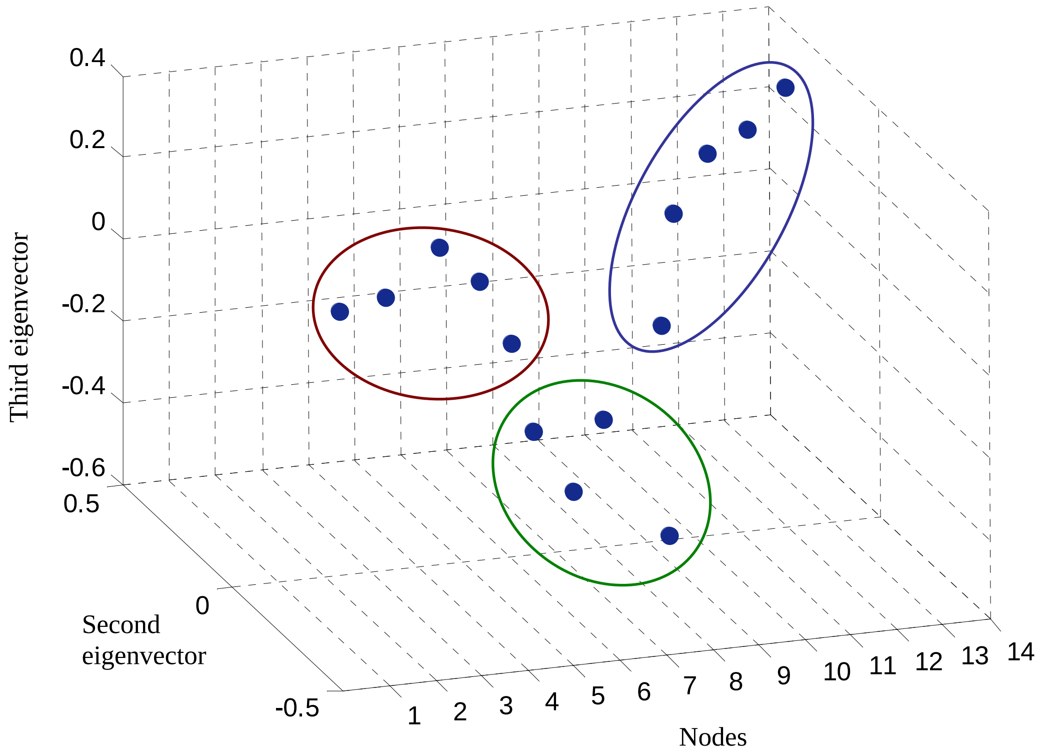

Figure 6 The second and third eigenvectors with respect to the nodes and the three partitions of the power system based on the eigenvectors of the Laplacian of the graph

Various techniques have also been proposed to reduce the complexities of graph partitioning and the computational burden even more. One may mention techniques such as multilevel clustering by Hendrickson et al. (1995), recursive spectral bisection method by Xu et al. (1998) or local clustering by Orponen et al. (2005) as examples of such efforts.

Yang et al. (1994) noted that the complexity of large problems can be efficiently reduced using the concept of divide and conquer. They proposed the use of ratio cut objective function in logic partitioning of VLSI design since it automatically coordinates the two traditional goals of logic partitioning, mincut and equipartition. The complexity in this application is how to reduce the needed number of trials while maintaining the quality of solutions. A preprocessing circuit-clustering procedure to improve the performance is proposed.

Hendrickson et al. (1995) proposed a multilevel algorithm for graph partitioning where the graph is approximated by a sequence of increasingly smaller graphs. The smallest graph is then partitioned using a spectral method. The partition is then propagated back through the hierarchy of graphs. They periodically applied a variant of the Kernighan-Lin algorithm to refine the partition. They claimed that the entire algorithm could be implemented to execute in time proportional to the size of the original graph. The proposed algorithm used branch and node weights to preserve in the coarse graphs as much structure of the original graph as possible to allow its applicability in physical problems such as the terminal propagation technique used in VLSI layout.

Xu et al. (1998) proposed a fast implementation of the recursive spectral bisection method for k-way partitioning since recursive bisections for k-way partitioning using optimal strategies at each step may not lead to a good overall solution. The relaxed implementation accelerates the partitioning process by relaxing the accuracy requirement of spectral bisection method. Since the quality of the solution of a spectral bisection of a graph primarily highly depends on the accuracy of its Fiedler vector, they proposed a tight iteration number bound and a loose residual tolerance for Lanczos algorithms to compute the Fiedler vector.

Multilevel versions of the Kernighan-Lin algorithm have been used for partitioning large graphs. In these algorithms, the graph is coarsened until it becomes so small that the processes for the problem at hand may be applied fast. Then the partitions are aggregated. Multilevel versions of the spectral method which are based on applying the spectral method at various levels have also been successfully used. In these methods, it is required to compute the Fiedler vector. Holzrich et al. (1999) proposed a purely spectral approach in which the calculation of the Fiedler vector is done using the Davidson algorithm. The problem at hand is to be set up in the form of a graphical preconditioner to the Davidson algorithm.

Spectral algorithms are usually based on the Fiedler vector of the Laplacian. Determining the Fiedler vector of the Laplacian or adjacency matrices of graphs is the most computationally expensive part of graph partitioning as well as other applications such as graph coloring, envelope reduction, and seriation. In many applications, an approximation of the Fiedler vector is used to speed up the solution.

Barnard (1995) proposed a parallel multilevel recursive spectral bisection algorithm for distributed parallel processing to balance the loads between the processors as well as minimize the interprocessor communication. Noting that the problem of finding a partition that balances the work of all processors and minimizes interprocessor communication is an NP-complete problem and heuristic approaches should be relied upon, recursive spectral bisectioning was chosen since it provides good partitions. In this approach, the eigenvector corresponding to the smallest non-trivial eigenvalue of the Laplacian of the graph is computed first. Then the graph is bisected into two partitions by finding the median of the components of the eigenvector. One partition includes nodes corresponding with the elements of the eigenvector that are less than or equal to the median, while the second partition includes nodes corresponding with the elements of the eigenvector that are greater than the median.

Several researchers have proposed graph-partitioning algorithms based on heuristics like the Kernighan-Lin method [Kernighan and Lin 1970], spectral methods [Pothen et al. 1990], genetic algorithms [Maini et al. 1994], or combinations of different methods. Although they are reported to work well for the chosen application, they can not guarantee asymptotically optimal bounds on the size of the obtained separators in the worst case. Most of them work on general graphs and are based on relatively simple and easy to code routines.

There are also other approaches for estimating the eigenvectors of the Laplacian of a graph. Srinivasan et al. (2002) proposed the use of Monte Carlo techniques which use an iterative scheme to converge to the correct eigenvalues and eigenvectors for this purpose. The complexities in this approach lie in the fact that Monte Carlo technique requires many iterations to converge; there are no generally accepted acceleration techniques, and it is very difficult to determine when convergence has been achieved.

The authors believe that since the exact values of the eigenvectors are not really necessary in portioning graphs, better approaches should be sought. The use of multilevel kernel k-means to speed up the partitioning and the modification of objective functions used in order to better reflect the complexities of the application in hand are proposed. Peiravi and Ildarabadi (2009) reported a paper in this respect in the area of power systems intentional islanding to illustrate this perspective. More work is being carried out in order to improve the application to make it more realistic and faster so that it becomes suitable for real time application to controlled power system islanding. It is suggested that the use of directed graphs be followed in order to properly count for the direction of power flow in the transmission lines. It is also suggested that the objective function be changed from just minimum cutest flows to a different measure considering the generation/load imbalance that will exist in the islands being formed a well as the instabilities such as frequency or voltage instability that may pursue the formation of islands

6. Conclusions

In this paper the various approaches for graph partitioning and their application to modern day scientific problems were presented. The mathematical basis of graph partitioning was discussed along with an example to show the amount of work required to carry out the partitioning even in simple cases.

Since the graph partitioning problem is NP-complete, it may not result in an optimum solution of practical problems. There exist many variations of this approach aimed at improving its performance. However, one should not be very optimistic about these techniques and think that they present a universal solution to partitioning problems since there are certain graphs for which each version of these methods performs poorly. Guattery and Miller (1998) showed that some of the existing spectral algorithms for graph partitioning perform poorly against the usual claim that they work well in practice. They present a generalized definition of spectral methods to include the use of a specified number of the eigenvectors corresponding to the smallest eigenvalue of the Laplacian matrix of the graph and show that even if these algorithms use a constant number of eigenvectors, there are graphs for which these algorithms do no better than they would using just the Fiedler vector. They also show that the use of the Fiedler vector would produce poor partitions.

Heuristics proposed for partitioning tasks were shown along with the complexities involved. Improvements in application to various problems were discussed with an example applied to power system islanding. It is concluded that the challenge still remains until careful attention is paid to the various complexities which exist in the application of graph-theoretic partitioning heuristics to modern day scienctific problems.

References

[1] Aleksandrov, L., Djidev, H., Guo H., Maheshwari, A., Partitioning planar graphs with costs and weights, Journal of Experimental Algorithmics, Vol. 11, 2006, pp.1-28.

[2] Barnard, S. T., PMRSB: Parallel Multilevel Recursive Spectral Bisection, Proceedings of the 1995 ACM/IEEE conference on Supercomputing, December 04 - 08, 1995, San Diego, California, United States, 1995.

[3] Bi, T., Ni, Y. , Shen, C. M. , Wu, F. F. , Efficient multiway graph partitioning method for fault section estimation in large-scale power networks, IEE Proceedings, Generation, transmission and distribution, May 2002, Vol. 149, No. 3, pp. 289-294.

[4] Cherng, J., Chen, S., Tsai, C., Ho, J., An efficient two-level partitioning algorithm for VLSI circuits, Proceedings of the 1999 Design Automation Conference, ASP-DAC '99, Asia and Pacific, Vol. 1, pp. 69-72, Wanchai , Hong Kong, 18-21 Jan. 1999.

[5] Cherng, J., Chen, S., An efficient multi-level partitioning algorithm for VLSI circuits, Proceedings of the 16th International Conference on VLSI Design (VLSI'03), 4-8 January 2003, New Delhi, India , pp.70-75, 2003.

[6] Dhillon, Inderjit S., Guan, Yuqiang, Kulis, Brian, “A Fast Kernel Based Multilevel Algorithm for Graph Partitioning,” In the Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery Data Mining (KDD), pp. 629-634, August 2005.

[7] Dhillon, Inderjit S., Guan, Yuqiang, Kulis, Brian, “Weighted Graph Cuts Without Eigenvectors: a Multilevel Approach,” IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 29, No. 11, pp. 1944-1957, November 2007.

[8] Guattery, S., Miller, G. L., On the quality of spectral separators, SIAM Journal of Matrix Analysis and Applications, Vol. 19, No. 3, July 1998, pp.701-719.

[9] Hagen L., Kahng, A. B., “New Spectral Methods for Ratio Cut Partitioning and Clustering,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 11, No. 9, pp.1074-1085, Sept. 1992.

[10] Hendrickson, B., Leland, R., A Multilevel Algorithm for Partitioning Graphs, Proceedings of the 1995 ACM/IEEE conference on Supercomputing, December 04 - 08, 1995, San Diego, California, United States, 1995.

[11] Holzrich, M., Oliveira, S., A graph based method for generating the Fiedler vector of irregular problems, Proceedings of the 11 IPPS/SPDP'99 Workshops Held in Conjunction with the 13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Processing, April 12-16, 1999, San Juan, PR, pp.978-985.

[12] Kamwa, I., Pradhan, A. K., Joos, G., Automatic segmentation of large power systems into fuzzy coherent areas for dynamic vulnerability assessment, IEEE Transactions on Power Systems, Vol. 22, No.4, pp. 1974-1985, 2007.

[13] Kernighan, B. W. Lin, S., “An efficient heuristic procedure for partitioning graphs,” The Bell System Technical Journal, Vol. 49, No. 2, Feb. 1970.

[14] Maini, H., Mehrotra, K., Mohan, C., Ranka, S., Genetic algorithms for graph partitioning and incremental graph partitioning, Proceedings of supercomputing '94, Nov. 1994.

[15] Moulitsas, I., Karypis, G., Architecture aware partitioning algorithms, Proceedings of the 8th international conference on Algorithms and Architectures for Parallel Processing, Agia, Cyprus, June 9-11, 2008, pp.42-53, 2008.

[16] Orponen, P., Schaeffer, S. E., Local clustering of large graphs by approximate Fiedler vectors, WEA: International Workshop on Experimental and Efficient Algorithms No..4 , GRECE (10/05/2005) , 2005 ,Vol.3503, pp.524-533.

[17] Othman, R. M., Derris, S. , Illias, R. M., Computational method for annotation of protein sequence according to gene ontology terms, Int'l Journal Biomedical Sciences, Vol. 1, No.3, 2006, pp.186-199.

[18] Peiravi, A., Ildarabadi, R., A Fast Algorithm for Intentional Islanding of Power Systems Using the Multilevel Kernel k-Means Approach, Journal of Applied Sciences (In Press).

[19] Pothen, H. Simon, and K. Liou, Partitioning Sparse Matrices with Eigenvectors of Graphs, SIAM J. Matrix Anaysis and Applications, Vol. 11, No. 3, pp. 430-452, 1990.

[20] Rideout, D. G., Stein, J. L., Louca, L. S., Extension and application of an algorithm for systematic identification of weak coupling and partitions in dynamic system models, Simulation Modelling Practice and Theory, Vol. 17, No. 1, pp.271-292, 2009.

[21] Rehtanz, C., Autonomous systems and intelligent agents in power system control and operation, Springer, 2003.

[22] Shepherd, S. J., Beggs, C. J. , Jones, S., Amino acid partitioning using a Fiedler vector model, European Biophysics Journal, Vol. 37, No. 1, December 2007, pp.105-109.

[23] Srinivasan, M., Mascagni, M., Monte Carlo techniques for estimating the Fiedler vector in graph applications, ICCS 2002: International Conference on Computational Science - Part II, 21-24 April 2002 , Amsterdam, 2002 , Vol.2331, pp.635-645.

[24] Wang, X. Z. , Yan, Z. , Xue, W. , An adaptive clustering algorithm with high performance computing application to power system transient stability simulation, 3rd International Conference on Deregulation and Restructuring and Power Technologies, DRPT 2008, art. no. 4523578, pp. 1137-1140, 2008.

[25] Wang C., Zhang B., Li Peng, Shu Jin, Cheng Lin-yan, Hao Zhi-guo, Bo Zhi-qian, Klimek, Andrew, “Power System Islanding Based on Multilevel Reduced Graph Partitioning Algorithm”, In the Proceedings of the 43rd International Universities Power Engineering Conference, 2008. UPEC 2008. , pp. 1-6, 1-4 Sept. 2008.

[26] You, H., Vittal, V., Yang, Z., Self-healing in power systems: An approach using islanding and rate of frequency decline-based load shedding, IEEE Transactions on Power Systems 18 (1), pp. 174-181, 2003.

[27] Xu, C., Nie, Y., Relaxed implementation of spectral methods for graph partitioning, Lecture notes in computer science, International sympsoium on solving irregularly structured problems in parallel, No. 5, Berkeley CA , 1998, Vol.1457, pp.366-375.

[28] Xue, W., Qi, S., Multilevel Task Partition Algorithm for Parallel Simulation of Power System Dynamics, 7th International Conference on Computational Science ICCS 2007, Part I, LNCS 4487, pp.529-537, , Beijing, China, May 27-30, 2007.

[29] Yang, W. P. Ker J. S., Kuo, Y. H. Ho Y. K., Two-level partitioning algorithm with stable performance, IEE proceedings. Circuits, devices and systems, Vol. 141, No. 3,.pp.197-202, 1994.

[30] Zhang, Y. J., Shi, Y. B. , Zhou, S. Q., Partitioning algorithm for innovation graph state estimation, Journal of Harbin Engineering University Vol. 29, No. 11, pp. 1166-1171, 2008.