

Change Detection through Clustering and Spectral Analysis

Diane Donovan¹, Birgit Loch²,
H.B. Thompson¹ and Jayne Thompson³

¹ Department of Mathematics, University of Queensland, St Lucia 4072, Australia

² Department of Mathematics and Computing, University of Southern Queensland
Toowoomba 4350, Australia

³ School of Physics, Melbourne University, Parkville 3052, Australia

Abstract. This paper defines a metric on a sequence of graphs which can be used to measure the distance between consecutive graphs. The method is based on vertex clustering through spectral analysis of the Laplace matrix.

Keywords: Networks, Clustering, Spectral Graph Theory

1 Introduction

Our aim is to investigate intragraph clustering in large enterprise networks and to define a metric, for quantifying network change, based on the evolution of vertex clusters. The use of clustering as a technique for change detection is a relatively recent idea and one which requires further study. In this paper, a metric will be defined on a sequence of graphs G^1, \dots, G^s . The basis for the metric will be a clustering procedure which uses edge weights to determine a partition \mathcal{C}^h of the vertex set of graph G^h . Then a distance measure $d(\mathcal{C}^h, \mathcal{C}^{h+1})$, $1 \leq h \leq s-1$, will be computed and used to quantify the distance $d(G^h, G^{h+1})$. The clustering procedure will be based on spectral analysis. This combination of spectral analysis, intragraph clusters and change detection is a new field of study and one which will be explored in the current paper.

In spectral graph theory (see [11] Section 8.6, page 452, for a general discussion) the eigenvalues and eigenvectors of associated matrices are calculated and used to characterize a graph's global structure. The literature includes some papers studying spectral theory and quantifying change in networks. For instance, for each graph G^h selected from a sequence of graphs, G^1, \dots, G^s , Bunke, Dickinson, Kraetzl and Wallis (see [1], page 72) determine the k largest positive eigenvalues $\lambda_1^h, \dots, \lambda_k^h$ and quantify a graph distance measure by setting

$$d(G^h, G^{h+1}) = \frac{\sum_{j=1}^k (\lambda_j^h - \lambda_j^{h+1})^2}{\min\left(\sum_{j=1}^k (\lambda_j^h)^2, \sum_{j=1}^k (\lambda_j^{h+1})^2\right)}. \quad (1)$$

This measure is termed the *spectral distance* and it will be one of the techniques used to calibrate the results obtained in this paper (see Section 6). Using similar ideas, Robles-Kelly and Hancock [8] calculate the largest eigenvalue for the adjacency matrix of a graph. The associated eigenvector is then used to identify an ordered path traversing every vertex. This ordered path forms the basis for computing the distance between graphs within the sequence. Robles-Kelly and Hancock also go on to discuss clustering, however individual clusters are determined using the largest eigenvalue. In this paper, we take a different approach and adapt techniques which have been used extensively for image analysis and graphs embedded in Euclidean space, see [6]. Initially, we will focus on a rigorous theoretical discussion of the eigenvectors of the Laplace matrix and associated techniques for partitioning the vertex set. By carefully analysing the theory we are able to demonstrate that the second smallest eigenvalue provides a good measure for determining vertex cluster sets. As noted by Hagan and Kahng [4], the advantage of this technique is that the partitioning is based on global information extracted from the overall network. Once we have determined the vertex clustering for each graph in a sequence, the Rand Index (see Dickinson, Bunke, Dadej and Kraetzl [3] and also [1] page 118) is used to define a graph distance measure.

Section 3 provides the necessary background on the Rand index. The underlying theory is reviewed in Section 4, providing a rigorous justification for the techniques under investigation. Section 5 discusses the necessary heuristics and provides some justification for the methods used. The theory is then tested in Section 6 and a comparison is made with distances measures studied by Bunke, Dickinson, Kraetzl and Wallis [1].

2 Definitions

A *graph* $G = (V, E(V))$ is a *vertex set* V and a collection $E(V)$ of 2-element subsets, chosen from V . If $\{u, v\} \in E(V)$, then $\{u, v\}$ is called an *edge* and u and v the *endpoints* of the edge. In this paper all graphs will be simple, in that there are no repeated edges and all edges have two distinct endpoints. Each edge in the graph will be assigned a label or *weight*. Thus it will be assumed that there exists a function β , where

$$\beta : E(V) \longrightarrow \mathbb{Z}^+ \cup \{0\}.$$

It is implicit in the definition that the non-existence of an edge between two vertices is equivalent to an edge with weight zero. When we wish to emphasize the fact that the edges of the graph are labeled we will use the notation $G = (V, E(V), \beta)$.

It will be productive to define a number of matrices to summarize specific information about a graph. We begin by setting $m = |V|$ and define an ordering on the vertex set; that is, the vertices are given an arbitrary order v_1, \dots, v_m and this ordering is used to define a vector $\mathcal{V} = (v_1, \dots, v_m)^T$. An *adjacency matrix*

$A = (a_{ij})$, for a graph $G = (V, E(V), \beta)$, is defined to be a $|V| \times |V|$ matrix with entries

$$a_{ij} = \begin{cases} \beta(\{v_i, v_j\}), & \text{where } \{v_i, v_j\} \in E(V), \\ 0, & \text{otherwise.} \end{cases}$$

Given the adjacency matrix A we let $a_i = \sum_j a_{ij}$ be the sum of i th row (or column since A is symmetric) and define the *diagonal matrix* to be $D = (d_{ij})$, where $d_{ij} = a_i \delta_{ij}$. Finally $B = D - A$ is the *disconnection* or *Laplace matrix*. Note that since A is symmetric, B is self adjoint. Hence, by the Spectral Theorem, B has a full complement of orthogonal eigenvectors.

Example 1. These concepts are illustrated for the graph given in Figure 1.

- vertex set is $V = \{A, B, C, D, E, F\}$;
- vertex ordering is given by $\mathcal{V} = (A, B, C, D, E, F)^T$;
- edge set is $E(V) = \{\{A, D\}, \{B, D\}, \{B, C\}, \{C, E\}, \{D, E\}, \{D, F\}\}$;
- weights are $\beta(\{A, D\}) = 5$, $\beta(\{B, D\}) = 1$, $\beta(\{B, C\}) = 2$,
 $\beta(\{C, E\}) = 5$, $\beta(\{D, E\}) = 2$, $\beta(\{D, F\}) = 3$.
- matrices are

$$D - A = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 7 & 0 & 0 & 0 \\ 0 & 0 & 0 & 11 & 0 & 0 \\ 0 & 0 & 0 & 0 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 & 5 & 0 \\ 5 & 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 5 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 5 & 0 & 0 & -5 & 0 & 0 \\ 0 & 3 & -2 & -1 & 0 & 0 \\ 0 & -2 & 7 & 0 & -5 & 0 \\ -5 & -1 & 0 & 11 & -2 & -3 \\ 0 & 0 & -5 & -2 & 7 & 0 \\ 0 & 0 & 0 & -3 & 0 & 3 \end{bmatrix} = B.$$

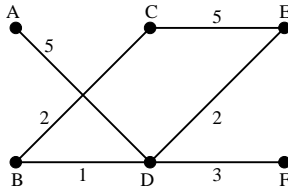


Figure 1: A weighted graph $G = (V, E(V), \beta)$

3 Distance Measure

The focus of this paper will be the partitioning of the vertex set V of a graph G into *cluster sets* C_1, \dots, C_t , such that $V = C_1 \cup \dots \cup C_t$ and $C_i \cap C_j = \emptyset$, for $1 \leq i < j \leq t$. The collection $\mathcal{C} = \{C_1, \dots, C_t\}$ is termed a *clustering* of V . This partition defines an equivalence relation $\rho(\mathcal{C})$ on the vertex set V ; that is, for any two vertices $x, y \in V$, $(x, y) \in \rho(\mathcal{C})$ if and only if there exists $C_i \in \mathcal{C}$ such that $x, y \in C_i$.

Given two graphs G^1 and G^2 , with associated clusterings \mathcal{C}^1 and \mathcal{C}^2 , we seek to quantify the distance $d(G^1, G^2)$ by specifying the distance between \mathcal{C}^1 and

\mathcal{C}^2 . We say a pair of vertices $x, y \in V$ are *consistent* if either $(x, y) \in \rho(\mathcal{C}^1)$ and $(x, y) \in \rho(\mathcal{C}^2)$, or $(x, y) \notin \rho(\mathcal{C}^1)$ and $(x, y) \notin \rho(\mathcal{C}^2)$. Otherwise the vertices x and y are said to be *inconsistent*. That is, two vertices x and y are consistent if they belong to the same cluster set in \mathcal{C}^1 and the same cluster set in \mathcal{C}^2 , or they are in different cluster sets in both \mathcal{C}^1 and \mathcal{C}^2 . Then $R^+ = |\{\{x, y\} \mid x, y \text{ are consistent vertices in } V\}|$ and $R^- = |\{\{x, y\} \mid x, y \text{ are inconsistent vertices in } V\}|$. Note that if $|V| = m$, then $R^+ + R^- = m(m - 1)/2$. Finally the *Rand Index* for a pair of clusterings \mathcal{C}^1 and \mathcal{C}^2 is defined to be

$$R(\mathcal{C}^1, \mathcal{C}^2) = 1 - \frac{R^+}{R^+ + R^-}.$$

We note that $R(\mathcal{C}^1, \mathcal{C}^2) \in [0, 1]$, with $R(\mathcal{C}^1, \mathcal{C}^2) = 0$ if and only if $|\mathcal{C}^1| = |\mathcal{C}^2|$ and all pairs of vertices are consistent, and $R(\mathcal{C}^1, \mathcal{C}^2) = 1$ if all pairs of vertices are inconsistent.

For a sequence of graphs the Rand Index will be used to measure the distance between consecutive graphs in the sequence. That is, for a given sequence of graphs G^1, \dots, G^s , we will determine a sequence of clusterings $\mathcal{C}^1, \dots, \mathcal{C}^s$, where \mathcal{C}^i is a clustering on the vertex set of graph G^i . Then

$$d(G^i, G^{i+1}) = R(\mathcal{C}^i, \mathcal{C}^{i+1}).$$

4 The clustering procedure

In this section we will follow the work of Hall [6], including ideas of Hagen and Kahng [4], and develop techniques needed to partition the vertex set of a general graph; see also [5]. Initially the partition will give two disjoint subsets, but repeated application will provide a hierarchical clustering tree with branches of degree 2, the root of the tree corresponding to V and the leaves corresponding to the cluster sets.

Thus initially we seek to take a graph $G = (V, E(V), \beta)$ and partition the vertex set V into two subsets U and W such that the sum of weights of edges connecting U and W , defined to be $w(U, W) = \sum_{x \in U, y \in W} \beta(\{x, y\})$, is minimized. More specifically, we seek to minimize the *cut ratio*

$$r = \frac{w(U, W)}{|U| \cdot |W|}.$$

We begin with the work of Hall [6], where $G = (V, E(V), \beta)$ is a graph embedded in \mathbb{R}^2 , and hence each vertex v_i corresponds to a point $(x_i, y_i) \in \mathbb{R}^2$. The vector $X^T = (x_1, \dots, x_m)$, where $x_i \leq x_j$, defines an ordering on the vertex set $V = (v_1, \dots, v_m)$. Hall seeks to reposition the vertices to minimise the sum of the weighted edges times the squared distances between the corresponding x -coordinates of X^T , minimizing

$$z = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (x_i - x_j)^2 a_{ij}.$$

Hall's method clusters together those vertices which are “strongly” connected.

We note that since A is symmetric

$$\begin{aligned}
 z &= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (x_i - x_j)^2 a_{ij} \\
 &= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (x_i^2 - 2x_i x_j + x_j^2) a_{ij} \\
 &= \frac{1}{2} \left(\sum_{i=1}^m x_i^2 a_i - 2 \sum_{i=1}^m \sum_{j=1}^m x_i x_j a_{ij} + \sum_{j=1}^m x_j^2 a_j \right) \\
 &= \sum_{i=1}^m x_i^2 a_i - \sum_{i=1}^m \sum_{j=1}^m x_i x_j a_{ij} \\
 &= X^T D X - X^T A X \\
 &= X^T B X.
 \end{aligned}$$

Hence to minimize the square of the distance between the x -coordinates of the vertices we seek to minimize the expression $X^T B X$, where B is symmetric, positive semidefinite.

Hagen and Kahng [4] extend these ideas to study general graphs, not just those which are embedded in the plane.

So let $G = (V, E(V), \beta)$ be any graph and $\{U, W\}$ be any partition of the vertex set $V = \{v_1, \dots, v_m\}$ into two subsets; that is, $V = U \cup W$ and $U \cap W = \emptyset$. Set $p = |U|/m$ and $q = |W|/m$ and let $X^T = (x_1, x_2, \dots, x_m)$ be a vector of length m with coordinates defined by

$$x_i = \begin{cases} q, & \text{if } v_i \in U, \text{ and} \\ -p, & \text{if } v_i \in W. \end{cases} \quad (2)$$

Note that $p+q = |U|/m + |W|/m = (|U| + |W|)/m = 1$, $X \cdot \mathbf{1} = qpm + (-p)qm = 0$, and

$$|x_i - x_j| = \begin{cases} 0, & \text{if } v_i, v_j \in U \\ 1, & \text{if } v_i \in U, v_j \in W \\ 1, & \text{if } v_i \in W, v_j \in U \\ 0, & \text{if } v_i, v_j \in W. \end{cases}$$

Also note

$$w(U, W) = \frac{1}{2} \sum_{\{v_i, v_j\} \in E(V)} (x_i - x_j)^2 a_{ij}.$$

Moreover

$$\begin{aligned}
 |X|^2 &= \sum_{i=1}^m x_i^2 = (q)^2 pm + (-p)^2 qm = pqm(q + p) \\
 &= \frac{pqm^2}{m} = \frac{|U||W|}{m}.
 \end{aligned}$$

So returning to Hall's analysis, where $\frac{1}{2} \sum_{\{v_i, v_j\} \in E(V)} (x_i - x_j)^2 a_{ij} = X^T B X$,

$$r = \frac{w(U, W)}{|U||W|} = \frac{\sum_{\{v_i, v_j\} \in E(V)} (x_i - x_j)^2 a_{ij}}{2m|X|^2} = \frac{X^T B X}{m|X|^2}. \quad (3)$$

Hence to minimize r we need to minimize $X^T B X / |X|^2$, where X satisfies the conditions given in (2) and $X \cdot \mathbf{1} = 0$.

This may be achieved by using Lagrange multipliers to minimize $X^T B X$ amongst X with $X^T X = 1$ and $X^T \mathbf{1} = 0$. Thus given that B is symmetric and $X^T B X = \sum_{ij} x_i b_{ij} x_j$, we have

$$\begin{aligned} \frac{\partial}{\partial x_s} (X^T B X) &= \sum_j B_{sj} x_j + \sum_i x_i B_{is}, \text{ and} \\ \nabla X^T B X &= B X + B^T X = 2B X. \end{aligned}$$

Consequently, for the Lagrangian $L = X^T B X - \lambda(X^T X - 1) - \mu X^T \mathbf{1}$, $\nabla L = 2B X - 2\lambda X - \mu \mathbf{1}$. After noting that $B \cdot \mathbf{1} = 0$ and so $\mathbf{1}$ is an eigenvector with eigenvalue 0, the optimal solution occurs when $0 = (B - \lambda I)X$ and non-trivial solutions for X can be obtained by calculating the eigenvalues λ_i of B and associated eigenvectors. Premultiplying by X^T gives $0 = X^T B X - \lambda X^T X$ and since it is assumed that $X^T X = 1$,

$$\lambda = X^T B X = z.$$

According to the Courant-Fischer Minimax principle ([2], page 106) the second eigenvalue

$$\lambda = \min_{X \perp \mathbf{1}, X \neq 0} \frac{X^T B X}{|X|^2}$$

and so Equation (3) implies

$$r = \frac{w(U, W)}{|U||W|} \geq \frac{\lambda}{m}.$$

This suggests r can be minimized by using the *Fiedler eigenvector* corresponding to the second smallest eigenvalue λ . Further, this eigenvalue can be used to determine the coordinates of the “position” vector X which satisfies the conditions given in (2). That is, $z = \frac{1}{2} \sum_{\{v_i, v_j\} \in E(V)} (x_i - x_j)^2 a_{ij} = X^T B X$ is minimized by the Fiedler eigenvector, with norm 1, and so intuitively the best approximation to this minimum will be obtained by assigning values to X which reflect the differences in the coordinates of the eigenvector and satisfy the conditions given in (2). More precisely, if the difference for two coordinates of the eigenvector is small then the difference between the corresponding coordinates given in (2) should be small. Conversely, if the difference for two coordinates of the eigenvector is large then the difference between the corresponding coordinates given in (2) should be large.

So to determine a partition of the vertices of V into two subsets, the components of the Fiedler vector are ordered in ascending numerical value giving $\mathcal{F} = (f_1, \dots, f_m)$. The same ordering is applied to the vector \mathcal{V} to obtain a new ordering of the vertex set, or equivalently a new vector \mathcal{V}^+ . This vector can now be split into subvectors reflecting the partitioning of V into two subsets U and W such that the sum of the edges joining vertices from different subsets is minimized. The value $|U|$ is termed the *splitting index*. A heuristic for determining the splitting index is discussed in the next section.

Once U and W are determined using the splitting index, the process is repeated for the induced graphs $G(U) = (U, E(U), \beta)$ and $G(W) = (W, E(W), \beta)$.

5 Splitting Index

Hagen and Kahng [4] propose four heuristics for determining the splitting index:

- (i) partition $\mathcal{V}^+ = (v_1^+, \dots, v_m^+)$ based on the sign of the corresponding component in the Fiedler vector; that is, $U = \{v_i^+ \mid f_i \in \mathcal{F}, f_i < 0\}$ and $W = \{v_i^+ \mid f_i \in \mathcal{F}, f_i \geq 0\}$;
- (ii) partition \mathcal{V}^+ around the median value of \mathcal{F} ;
- (iii) partition $\mathcal{V}^+ = (v_1^+, \dots, v_m^+)$ by determining the maximum difference between consecutive components of \mathcal{F} ; that is, $a_i = |f_i - f_{i+1}|$, $1 \leq i \leq m-1$, $U = \{v_1^+, \dots, v_t^+ \mid a_t = \max\{a_i\}\}$ and $W = \{v_{t+1}^+, \dots, v_m^+\}$;
- (iv) partition \mathcal{V}^+ to obtain the least cut ratio; that is, for $1 \leq i \leq m-1$, calculate $r_i = w(U_i, W_i)/|U_i||W_i|$, where $U_i = \{v_1^+, \dots, v_i^+\}$ and $W_i = \{v_{i+1}^+, \dots, v_m^+\}$, then set $U = \{v_1^+, \dots, v_t^+ \mid r_t = \min\{r_i\}\}$ and $W = \{v_{t+1}^+, \dots, v_m^+\}$.

It is suggested in [4] that method (iv) be used. However, our research indicates that the above methods are susceptible to singularities and thus we will follow the work of Hopcroft, Khan, Kulis and Selman, [7], by investigating instabilities in the data and thus determining the most appropriate heuristic.

Full details of the data set are given in Section 6, however for a random selection of graphs the following observations were made. The graphs contained a large number of pendant vertices (vertices incident with at most one edge). For these graphs heuristics (i), (iii) and (iv) were equivalent. Thus heuristics (ii) and (iv) were tested on a number of randomly selected graphs, then the pendant vertices were removed. When heuristic (iv) was applied the pendant vertices dominated the majority of cluster sets and their removal reduced the clustering to a small number of cluster sets containing a large number of vertices. When heuristic (ii) was applied and the pendant vertices were removed the number of cluster sets did not change. At times some of the pendant vertices were collected together into the same cluster but at times they were fairly evenly distributed across the cluster sets. Hence given the data set, it was decided that heuristic (ii) should be applied to determine the splitting index.

6 Experimentation

In this section we apply the above procedure (using heuristic (ii)) to data obtained from an enterprise communication network⁴. This data set was obtained by placing probes on physical links and recording the volume of information in daily communications. Three types of statistics were gathered: sender and receiver identifications and a count of the TCP/IP traffic. The sender and receiver identifications were clustered into 328 business domains or vertices in the resultant network. Data was collected over a period of 102 days and this information gives rise to a series of graphs G^1, \dots, G^{102} .

In [1], Bunke, Dickinson, Kraetzl and Wallis conducted a number of tests on the distance between consecutive graphs in the above sequence. We have selected two techniques proposed in [1], the Spectral Distance (see (1), Section 1) and the Edit Distance, and compared these techniques with the procedure presented in this paper. The results of these tests are given below, but first we provide a brief discussion of edit distance.

Let $G^h = (V, E^h(V), \beta^h)$ and $G^{h+1} = (V, E^{h+1}(V), \beta^{h+1})$, where $|V| = m$ and let $B^h = [b_{ij}^h]$ and $B^{h+1} = [b_{ij}^{h+1}]$ represent the corresponding $m \times m$ disconnection matrices. Then the *edit distance* is given by

$$d(G^h, G^{h+1}) = \sum_{1 \leq i < j \leq m} |b_{ij}^h - b_{ij}^{h+1}| + \sum_{b_{ii}^h = 0, b_{ii}^{h+1} \neq 0} 1 + \sum_{b_{ii}^h \neq 0, b_{ii}^{h+1} = 0} 1.$$

That is, the sum of the differences in the weights of edges in G^h and G^{h+1} plus the number of vertices of non-zero degree which only occur in one of G^h or G^{h+1} . This is a robust measure as it accurately records the overall change in communication traffic across the network.

The values of $d(G^h, G^{h+1})$, using each of the three techniques (spectral distance, edit distance and clustering techniques), has been calculated for $1 \leq h \leq 101$ and the results are displayed in the following graphs.

In the first of these graphs, we note that both the edit distance and the clustering procedure detect major changes in the network between days 20 and 25, 60 and 65, and 85 and 90. The spectral distance detects the first of these changes, but not the other two major perturbations. A more interesting aspect highlighted in the first graph is the differences between days 20 and 90. Other than the major changes mentioned above, the edit distance shows relatively little change from one day to the next. However the clustering technique tends to imply that during the same period there are changes in intergroup communications, indicating that the dynamics of the network may be changing through this period. These results suggest that the clustering technique presented here is useful in detecting major changes in the overall communication traffic, but may also be useful in detecting changes in group dynamics.

⁴ We wish to acknowledge the generous support of Miro Kraetzl, who among other things has given us access to the data set.

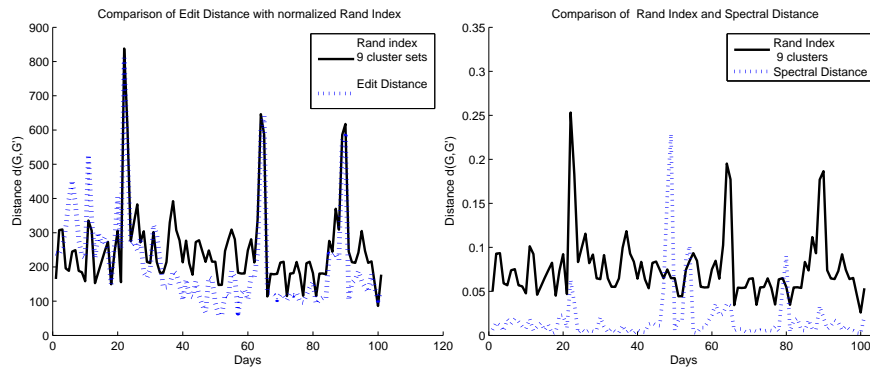


Figure 2: Time Series: Network Evolution

7 Conclusions and further research

The above results show that the Fiedler vector can be used to determine an intra-graph clustering. Further that this clustering technique may tentatively provide a method comparable to the edit distance for detecting change in networks. For the data tested here, there are some instances where the clustering technique detects changes which were not so obvious in the edit distance. These results are positive and indicate that further investigation of the technique is warranted, and may lead to more refined methods providing a robust tool for the detection of change within large enterprise networks. Possible areas for further investigation are extended analysis of the heuristics used to determine the splitting index; an investigation of the use of the M-cut ratio as proposed by Shi and Malik; an investigation of related algorithms which overcome the problem of a heirarchical clustering process, for instance some of the techniques proposed by Tolliver and Miller [10].

References

1. Horst Bunke, Peter J Dickinson, Miro Kraetzl and Walter D Wallis: A Graph-Theoretic Approach to Enterprise Network Dynamics, Birkhäuser. Boston (2007)
2. John W Dettman: Mathematical methods in Physics and Engineering. 2nd edition McGraw-Hill New York (1969)
3. Peter J Dickinson, Horst Bunke, Arek Dadej and Miro Kraetzl: A novel graph distance measure and its application to monitoring change in computer networks. In the Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics. volume 3 Orlando, FL (2003) 33–338
4. Lars Hagen and Andrew B Kahng: New spectral methods for ratio cut partitioning and clustering. IEEE Transactions Computer-Aided Design Santa Clara CA (1992) 422–427
5. Lars W Hagen and Andrew B Kahng: Combining problem reduction and adaptive multistart: A new technique for superior iterative partitioning. IEEE Transactions

- on Computer-aided Design of Intergrated Circuits and Systems **16** No. 7 (1997) 709–717
6. Kenneth M Hall: An r -dimensional quadratic placement algorithm. Management Science **17** No. 3 (1970) 219–229
 7. John Hopcroft, Omar Khan, Brian Kulis and Bart Selman: Tracking evolving communities in large linked networks. Proceedings of the National Academy of Sciences of the United States of America (2004) 1–5 doi:10.1073/pnas.0307750100
 8. Antonio Robles-Kelly and Edwin R Hancock: Graph Edit Distance from Spectral Seriation. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** No. 3 (2005) 365–378
 9. Jianbo Shi and Jitendra Malik: Normalized Cut and Image Segementation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** No. 8 (2000) 888–905
 10. David A Tolliver and Gary L Miller: Graph partitioning by spectral rounding: application in image segmentation and clustering. Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06) (2006) 1–8.
 11. Douglas B West: Introduction to Graph Theory. Prentice-Hall Inc. Upper Saddle River (1996).