Global Alignment of Protein-Protein Interaction Networks for Analyzing Evolutionary Changes of Network Frameworks

A. Terada
Department of Computer Science
Tokyo Institute of Technology
2-12-1 Ookayama, Meguro-ku,
Tokyo, 152-8550, Japan

J. Sese
Department of Computer Science
Tokyo Institute of Technology
2-12-1 Ookayama, Meguro-ku,
Tokyo, 152-8550, Japan

Abstract

Recent technological advances have yielded protein-protein interaction (PPI) networks across multiple species. To interpret these networks, comparison methods to characterize interactions across species have gained importance. However, the methodologies of most such studies have been limited to the relationships of protein complexes or pathways between different species because methodologies for network comparisons are limited. In this study, we introduce a novel comparison problem of biological networks, focusing on the framework structure of the networks, and compare these structures to find changes and conservations in the smaller networks. For this purpose, we define an alignment score between two networks based on the validity of the framework structure in each species and on the conservation of homologous genes in the networks. We propose an algorithm to find a high-score alignment based on k-means clustering. Experiments using *D. melanogaster* and *C. elegans* PPI networks show that our algorithm identified the network conservation on genes belonging to cancer-related pathways.

1 Introduction

The recent availability of protein-protein interaction (PPI) networks and pathways for several species [1] makes it possible to compare the networks across different species. Some comparative studies [2, 3] have found relationships between network changes and functional gains. Most comparative studies have focused on the conservation of protein complexes or the paths in the pathways across species. The methods, however, can search only for local changes although changes that are more global are important to discover the changes in the usage of functional modules through evolution. To identify the large evolutionary rewiring events between networks, we developed a method to compare two different PPI networks by focusing on brief network structures.

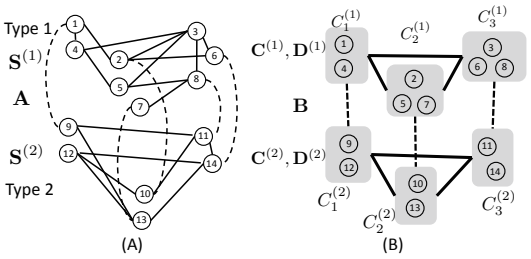


Figure 1: Example of global network alignment. (A) Given graph. (B) Aligned cluster networks of (A).

An example that motivates this study is shown in Figure 1(A). The upper and lower graphs both represent a PPI network observed from a single species. In the graphs, genes (vertices) and interactions (edges) are indicated by circles and solid lines, respectively. To simplify their mathematical description, we label the upper and lower graphs as type 1 and 2 graphs, respectively. A type 1 graph contains eight vertices and eleven edges, and a type 2 graph contains six vertices and eight edges.

We added the relationships of homolog genes between two different species to these graphs. The homolog genes can be represented by the edges across the two types of graphs. The dotted arc edges in Figure 1(A) indicate the edges. To discriminate between edges within a graph and edges across graphs, we refer to the edges within a graph as *interaction edges* and the edges across graphs as *homolog edges*. The homolog edges indicate genes with sequences that are almost identical between two species. These homolog edges are usually sparse because most genes have few homologs in other species. In this study, we compare the two graphs with sparse edges.

There are two major approaches to compare such networks. One approach is to extract the dense parts in each graph using protein complex extraction meth-

ods [4, 5] or clustering [6]and then compare the components, which allows us to identify similar complexes in the networks. This approach, however, may miss findings related to the important conservation of protein interactions because the first clustering step is independent of the subsequent associating step. The other approach focuses on alignments of the entire structure of networks [7]. In this study, we tackle the latter problem by using graph clustering.

To align an entire graph structure, we generate groups of vertices and focus on relationships across these groups. Figure 1(B) shows the relationship groups of Figure 1(A). In this figure, for example, a group $C_2^{(1)}$ consisting of vertices 2, 5, and 7 represents cluster ID 2 in the type 1 graph. Clusters in which most members are connected to each other are connected by solid (for a set of interaction edges) or dotted (for a set of homolog edges) lines in Figure 1(B).

In our example, the two large graphs are not of the same size, and it is clear that no exact mapping exists between them. Even if the two graphs are of the same size, this is a type of graph isomorphism problem that is NP-hard. Therefore, we defined a cost function to indicate the alignment level and determined the division of vertices that minimizes the cost function. To derive an alignment that minimizes the cost function, we introduced a new algorithm. To evaluate its performance, we compared the results of our algorithm to those of famous graph clustering algorithms. We conducted our experiments using real biological networks in two different species and identified the evolutionary conservation and changes between the networks.

2 Related Work

Graph clustering methods [6, 8] provide us with good partitions of large graphs; indeed, we can derive correspondences between two graphs based on these partitions. However, these methods cannot handle homolog edges. In addition, the overall structures of the clustering result do not summarize the original graph structure.

Graph summarization and approximation methods [9, 10] generate groups using relationships between clusters. However, existing methods can generate groups from a single graph, whereas our problem requires the simultaneous handling of two different graphs. Therefore, we cannot apply these techniques directly to our problem.

In addition, most of the methods for aligning two different graphs focus on dense graphs. Although Bayati *et al.* [7] have proposed a method to align two sparse graphs, their method maximizes the number of squares consisting of two interaction edges and two homolog edges. However, the graph shown in Figure 1(A) has no square structures, and hence, because of this assumption, this method cannot align the depicted graphs.

3 Preliminaries on network alignment

We first introduce the notation to describe the graphs. We then introduce our approach to compare two large graphs. In this study, we considered a graph that consists of two types of graphs. However, we can extend the algorithm to graphs that consist of more than three types of graphs.

Suppose that we are given two undirected and unweighted graphs $G^{(1)} = (V^{(1)}, E_{\mathrm{I}}^{(1)})$ and $G^{(2)} = (V^{(2)}, E_{\mathrm{I}}^{(2)})$. $V^{(i)}$ and $E_I^{(i)}$ ($i = 1$ or 2) include vertices and consist of interaction edges between vertices in $V^{(i)}$, respectively. Let E_{O} indicate homolog edges between two graphs representing relationships between the graphs. Let us call graph $(G^{(1)}, G^{(2)}, E_{\mathrm{O}})$ *relation graph*. The number in each circle in Figure 1(A) denotes the unique label of the vertex. $V^{(1)}$ and $V^{(2)}$ contain vertices 1 to 8 and 9 to 14, respectively.

Interaction and homolog edges connect vertices within a graph type and between different types of graphs, respectively. In Figure 1, interaction edges are solid lines, while homolog edges are denoted by dotted lines. Based on edge categorization, we generate two different types of related matrices of vertices.

Definition 1 Let n_i represent the number of vertices in a type i graph. We represent a graph as a set of related matrices $\vec{S}^{(i)} \in \mathbb{R}_{\geq 0}^{n_i \times n_i}$ and $\vec{A} \in \mathbb{R}_{\geq 0}^{n_1 \times n_2}$, where $\mathbb{R}_{\geq 0}$ denotes all the non-negative real numbers. $\vec{S}^{(i)}$ represents the edge weights for the interaction edges within $G^{(i)}$, and $\vec{A}$ represents the edge weights for the homolog edges between $G^{(1)}$ and $G^{(2)}$.

We assume 0-1 binary relations in Figure 1(A). Row and column orders of these matrices correspond to vertex numbers. The first row in $\vec{S}^{(2)}$ indicates the relationships between vertex 9, which is the first vertex in $V^{(2)}$, and the other vertices, while the leftmost column in $\vec{S}^{(2)}$ also indicates the relationships between vertex 9 and the other vertices. Because the graph consists of undirected edges, the matrix is symmetric. Each row in $\vec{A}$ denotes the relationships between the two types of graphs. The second row of $\vec{A}$ indicates connections between vertex 2 and vertices in the type 2 graph. Vertex 2 connects to vertex 10, and hence the second column is 1.

In this graph, we introduce a global network alignment problem, which involves finding an approximate large common structure between two types of

graphs. Because there are few interaction edges (as is often the case in biological networks and social networks), it is difficult to determine common subgraphs from the graphs. Therefore, we derive groups of vertices and focus on their relationships. To describe the relationships between groups, we define a cluster network.

Definition 2 (Cluster network) Let $C_j^{(i)} \subseteq V^{(i)}$ be a group of vertices. Let $\mathcal{C}^{(i)}$ and $L_{\mathrm{I}}^{(i)}$ denote a set of clusters in type 1 and links between clusters in $\mathcal{C}^{(i)}$. Let L_{B} contain links between clusters $C \in \mathcal{C}^{(1)}$ and $C' \in \mathcal{C}^{(2)}$.

Figure 1(B) shows a cluster network derived from Figure 1(A). In the figure, six nodes represented by gray squares containing circle vertices indicate clusters. $\mathcal{C}^{(1)}$ contains $C_1^{(1)}$, $C_2^{(1)}$ and $C_3^{(1)}$, and $\mathcal{C}^{(2)}$ contains $C_1^{(2)}$, $C_2^{(2)}$ and $C_3^{(2)}$. The solid and dotted lines represent links within a type and between types, respectively. For instance, $C_1^{(1)}$ has two links to clusters within the same type $C_2^{(1)}$ and $C_3^{(1)}$ as well as a link to $C_1^{(2)}$ which is a cluster in the other type. The structures of the two cluster networks are identical, and there are relationships between corresponding vertices; hence, the two types of graphs can be perfectly aligned. In most real datasets, it is difficult to find such perfectly aligned networks. Instead of finding the alignment, we introduce an index describing the quality of the alignment, and try to minimize the index.

4 Method

We now introduce a novel method for deriving the global network alignment, which is to generate clusters using a newly introduced distance function based on the distribution of the number of edges connecting to clusters. Using the experiments with a synthetic dataset and a real dataset, we compare the characteristics of these methods.

4.1 Alignment based on similarities among edge connections

We here introduce a feature vector that represents connections between vertices.

Let $\vec{C}^{(i)} \in \{0,1\}^{n_i \times k}$ denote an indicator matrix for graph $G^{(i)}$, where n_i and k are the number of vertices in $G^{(i)}$ and a user-defined number of groups, respectively. When vertex v belongs to cluster $C_j^{(i)}$, $\vec{C}_{vj}^{(i)}$ is 1; however, when v does not belong to $C_j^{(i)}$, $\vec{C}_{cj}^{(i)}$ is 0.

Let $\vec{M}_{\mathrm{I}}^{(i)}$ denote a matrix including the number of interaction edges between clusters in a type i graph. The pqth element of the matrix represents the number of edges between clusters $C_p^{(i)}$ and $C_q^{(i)}$. Given indicator functions $\vec{C}^{(i)}$ and edge information $\vec{S}^{(i)}$, $\vec{M}_{\mathrm{I}}^{(i)} = \frac{1}{2}\vec{C}^{(i)\mathrm{T}}\vec{S}^{(i)}\vec{C}^{(i)}$.

We also define a matrix $\vec{F}^{(i)}$, which includes feature vectors for each vertex. The pqth element of the matrix indicates the number of edges connecting to $C_q^{(i)}$ from vertex p. $\vec{F}^{(i)}$ can be calculated as $\vec{S}^{(i)}\vec{C}^{(i)}$.

Using these matrices, we can define a cost function describing the edge relationships. Let $\vec{C}$, $\vec{M}_{\mathrm{I}}$ and $\vec{F}$ be $\begin{pmatrix} \vec{C}^{(1)} & 0 \\ 0 & \vec{C}^{(2)} \end{pmatrix}$, $\begin{pmatrix} \vec{M}_{\mathrm{I}}^{(1)} & \vec{0} \\ \vec{0} & \vec{M}_{\mathrm{I}}^{(2)} \end{pmatrix}$ and $\begin{pmatrix} \vec{F}^{(1)} & \vec{0} \\ \vec{0} & \vec{F}^{(2)} \end{pmatrix}$, respectively. We can define a cost function $J_1(\vec{C}) = Dist(\vec{F}, \vec{C}\vec{M}_{\mathrm{I}})$, where $\vec{C}\vec{M}_{\mathrm{I}}$ represents the cluster centers specified in the indicator matrix $\vec{C}$. We use the cosine distance as the $Dist$ function; that is, we calculate the cosine distance between $\vec{F}$ and $\vec{C}\vec{M}_{\mathrm{I}}$ for each row and calculate the average (or summation) of the values. The cosine distance is defined as $1 - \frac{\vec{x}\cdot\vec{y}}{|\vec{x}||\vec{y}|}$ for vectors $\vec{x}$ and $\vec{y}$.

Although $J_1(\vec{C})$ provides us with a division of vertices, the cost function only uses interaction edges and does not consider homolog edges. We here modified $J_1(\vec{C})$ to add the effect of homolog edges to the cost function. Similarly, to interaction edges, we can calculate the number of homolog edges between clusters. Let $\vec{M}_{\mathrm{B}}$ denote a matrix including the number of homolog edges between different types of clusters. The pqth element of the matrix represents the number of edges between cluster $C_p^{(1)}$ and cluster $C_q^{(2)}$. The matrix $\vec{M}_{\mathrm{B}}$ can be described as $\frac{1}{2}\vec{C}^{(1)\mathrm{T}}\vec{A}\vec{C}^{(2)}$. Adding this matrix to the central vectors of $\vec{M}$ provides a new measure for counting the homolog edges. Let $\bar{\vec{M}}_{\mathrm{I}}$ indicate a matrix with a pqth element that satisfies $\vec{M}_{\mathrm{I}pq} / \sum_{q=1}^{k} \vec{M}_{\mathrm{I}pq}$. Similarly, let $\bar{\vec{M}}_{\mathrm{B}}$ indicate a matrix with a pqth element such that $\vec{M}_{\mathrm{B}}$ indicates a matrix with a pqth element that satisfies $\vec{M}_{Bpq} / \sum_{q=1}^{k} \vec{M}_{Bpq}$. By combining these two matrices, we define a new cost function as

$$J_2(\vec{C}) = Dist\left(\vec{F}, \vec{C}\left(\bar{\vec{M}}_{\mathrm{I}} + \alpha \begin{pmatrix} 0 & \bar{\vec{M}}_{\mathrm{B}} \\ \bar{\vec{M}}_{\mathrm{B}}^{\mathrm{T}} & 0 \end{pmatrix}\right)\right),$$

where α is a user-specified weight. A large value of α indicates that homolog edges are important.

Even if homolog edges are considered in the cost function, cluster networks from two different types of graphs might form completely different structures. To calculate the cost function, a consideration of the correspondence cluster's feature vector on the opposite

Algorithm 1 ANGIE: Aligning Networks Globally with Interconnect Edges

Require: A relation graph represented as a set of matrices $\vec{S}^{(1)}$, $\vec{S}^{(2)}$, and $\vec{A}$ and the number of clusters k.

1: Initialize $\vec{C}^{(1)} \in \{0,1\}^{n_1 \times k}$ and $\vec{C}^{(2)} \in \{0,1\}^{n_2 \times k}$ randomly where $\vec{C}^{(i)}\vec{1} = \vec{1}$.
2: **repeat**
3: // Update cluster center vector $\vec{M}$ and feature vectors.
4: Compute normalized cluster center matrices $\bar{\vec{M}}_{\mathrm{I}}^{(i)}$ and $\bar{\vec{M}}_{\mathrm{B}}$
5: Compute feature vectors $\vec{F}$ using current cluster assignment $\vec{C}^{(1)}$ and $\vec{C}^{(2)}$.
6: // Assign vertices to the cluster with the closest central vector.
7: Update $\vec{C}^{(1)}$ and $\vec{C}^{(2)}$ so as to minimize $J(\vec{C})$.
8: **until** Cluster assignment of vertices in $\vec{C}^{(1)}$ and $\vec{C}^{(2)}$ does not change

graph might suggest whether similar structures exist in the two cluster networks. Therefore, we merge the central vectors of corresponding clusters in the other types of graphs. Here, we suppose that the correspondence cluster of C_j^i is $C_j^{i'}$ where $i' = 1$ when $i = 2$ and $i' = 2$ when $i = 1$. Note that the cluster represented by the first row of $\vec{M}_{\mathrm{I}}^{(1)}$ corresponds to the first row of $\vec{M}_{\mathrm{I}}^{(2)}$ and the first row of $\vec{M}_{\mathrm{B}}$ representing homolog edges corresponds to the first column of $\vec{M}_{\mathrm{B}}$. With this observation, we introduce a new cost function:

$$J(\vec{C}) = Dist\left(\vec{F}, \vec{C}\left(\bar{\vec{M}}_{\mathrm{I}} + \alpha\begin{pmatrix} 0 & \bar{\vec{M}}_{\mathrm{B}} \\ \bar{\vec{M}}_{\mathrm{B}}^{\mathrm{T}} & 0 \end{pmatrix} + \beta\begin{pmatrix} \bar{\vec{M}}_{\mathrm{I}}^{(2)} & \bar{\vec{M}}_{\mathrm{B}}{}^{\mathrm{T}} \\ \bar{\vec{M}}_{\mathrm{B}} & \bar{\vec{M}}_{\mathrm{I}}^{(1)} \end{pmatrix}\right)\right), \quad (1)$$

where α and β are user-specified weights. Large values β indicate a high correspondence with the other type of network.

We now introduce an algorithm to identify clusters that minimize the cost function $J(\vec{C})$. The function contains two parameters. In this section, we suppose that the parameters are given, but we will evaluate the effect of the parameters and discuss how to select values automatically in our discussion of our experimental results.

Algorithm 1 represents the overall procedure of the proposed algorithm, which is called Aligning Networks Globally with Interconnect Edges (ANGIE). ANGIE is based on a k-means algorithm. Although this simple algorithm generates a relation between two graphs, there is no guarantee that this result is optimal. We show that this algorithm reaches better performance than the famous graph clustering methods even if the dataset has noise.

The assignment of vertices to clusters may not converge. Avoiding the infinite loop, we previously determine the maximum number of repeats in practice. In our experiments, we set the maximum number as 100 because the cost value is almost stable or periodic after the number of repeats.

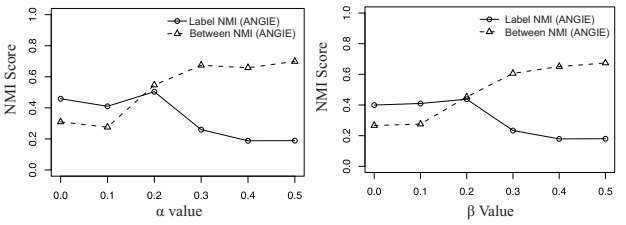
5 Experiments

In this section, we evaluate our method ANGIE using a synthetic dataset and a PPI network dataset. The experiments using the synthetic dataset demonstrate that ANGIE shows a better performance than existing methods. Experiments using a PPI network dataset demonstrate that our algorithm can identify evolutionary changes.

5.1 Results for a synthetic network

We generate a synthetic network dataset to evaluate the performance of our algorithm. The synthetic dataset includes 2,000 vertices, about 11,000 interaction edges and 500 homolog edges. (Owing to space limitations, the details are omitted.)

All experiments were performed using a 3.2 GHz AMD® Opteron™ machine with 1GB of memory running on Linux kernel 2.6. We implemented our algorithms in Java™ 5. All run times in these experiments were less than 1 minute.



(A) α: Homolog edge weights.

(B) β: Weights of corresponding cluster vectors.

Figure 2: NMI scores.

To evaluate our alignment results, we used true cluster labels. We used normalized mutual information (NMI) [11] which has been used in many applications to measure the performance of clustering methods [12]. A larger NMI value indicates a better clustering result. However, even if the result has a high NMI value, structures of two cluster networks may be totally different. Therefore, we also check the NMI score between clusters over the two types of graphs. It is noteworthy that a relatively large NMI value indicates that groups from two different graphs are related to each other. To distinguish between these two different NMI values, we call the NMI value with true cluster labels *label-NMI*, while we call the NMI value with the corresponding group members *between-NMI*.

ANGIE has two parameters. We now change them independently and check the effect of these changes on the alignment result. Figures 2(A) and (B) show the effect of parameters α and β, respectively. α and β appear in Equation 1.

In Figure 2(A), the larger value on the X-axis is due to a greater consideration of the homolog edges during the assignment step in line 7 of Algorithm 1. Owing to the large α, the value of label-NMI value increases when α is less than 0.2, but in decreases when α is greater than 0.2. When α is zero, the homolog connections do not generate clusters and the cluster assignment is determined only by the interaction edges. Since this synthetic graph is very sparse, considering only homolog edges seems appropriate based on this result. However, when α is high, vertices are divided according to homolog edges. However, homolog edges have no information on cluster divisions with respect to a specific graph type, and therefore cluster division fails with an increase in α. Nevertheless, when α is large, cluster division between two different types of graphs is similar. Therefore, between-NMI values increase with an increase in α.

Figure 2(B) shows the effect of the mixture ratio of the different vectors according to the graph vector. When β is zero, vertices from the two different types of graphs form clusters independently. With an increase in β, the algorithm simultaneously attempts to create clusters based on the two different types of graphs. Figure 2(B) shows characteristics similar to those of Figure 2(A). The value of label-NMI decreases when $\beta \geq 0.2$, and the value of between-NMI increases according to an increase in β. The larger β also indicates a higher importance of the corresponding cluster structure. Therefore, when β is large, between-NMI is high, and label-NMI is low.

Because there are no algorithms whose problem is similar to ours, we cannot compare our method with other methods exactly. However, by using homologous edges, the scores may become larger than the results generated by running a clustering method on each species network. We compared the results of a well-known group-clustering method called METIS [6] and a model-based clustering method called HRGC introduced by Long *et al.* [10]. METIS's label- and between-NMI values are 0.406 and 0.387, respectively. HRGC's scores are less than 0.2. Both of these values are lower than the results generated by ANGIE, and hence, for this dataset, ANGIE generates a better alignment than METIS.

5.2 Results for a biological network

We applied our algorithm to a real PPI dataset observed from two different species [1], *D. melanogaster* (fruit fly) and *C. elegans* (nematode worm). The fruit fly dataset contains 897 genes (that is, vertices) and 1,855 interactions (edges), and the nematode worm dataset contains 478 genes and 732 interactions. As homolog edges, we used information in the Kyoto Encyclopedia of Genes and Genomes (KEGG) [13]. The number of homolog edges is 379. Note that these graphs are very sparse. We divide the graph into 40 groups because the dataset contains 57 functional groups based on pathway annotations in KEGG.

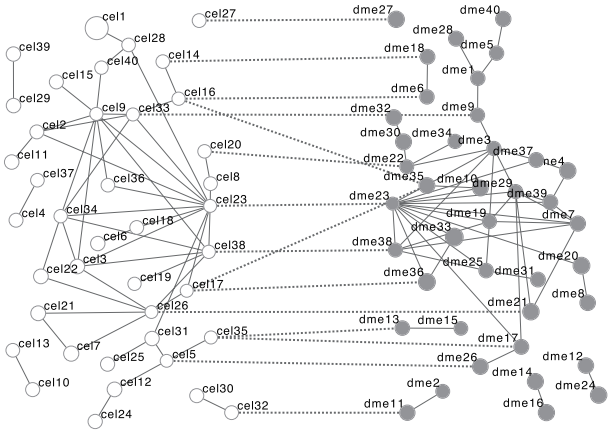


Figure 3: Biological result. Alignment between nematode worm (cel) and fruit fly (dme).

To obtain the best alignment, we calculated the NMI values over all combinations of the two parameters ranging from 0 to 1 and selected the results with the largest average NMI value according to the gene functional groups. As the result, both α and β were set at 0.1, and the NMI values for the fruit fly and nematode worm were 0.274 and 0.323, respectively. We also calculated the best NMI value using HRGC, resulting in 0.220 and 0.295 for the fruit fly and nematode worm, respectively. With this framework, we can observe whether ANGIE finds better alignment than NMI using a real dataset. As for the synthetic dataset, we independently applied METIS to two graphs to compare its scores with ANGIE's scores; the resulting NMI values were 0.32 for the fruit fly and 0.35 for the nematode worm. Although these NMI values are higher than the ANGIE result, the framework structures of the cluster networks are totally different between two species because the method did not consider homolog genes, and hence it is difficult to compare the brief structures of the networks.

Figure 3 shows the cluster graph identified by ANGIE. In the figure, nodes contain sets of vertices (genes), and the number of the nodes represents the cluster number. In addition, "dme" and "cel" indicate the fruit fly and nematode worm, respectively.

Node size is associated with the number of vertices in the cluster. Each edge indicates that the density of edges between two connected clusters is high; that is, members in two clusters are highly connected to each other. We note 50 links between clusters within the same species that consist of the largest number of interaction edges represented by solid lines and 15 links between two species that consist of the largest number of homolog edges represented by dotted lines. Both large components in the two different types of graphs form complex structures in the center that and have long edges. Central nodes have many connections to the other nodes (so-called hubs). Although we connected parallel links between the two species, we observe few crossing links in the networks, and hence the cluster networks are topologically similar. Cluster Nos. 23 and 38 in both graphs are connected by homolog edges. Checking the gene functions in clusters cel38 and dme38 connected by the homolog edges, we note that both clusters have many MAPK pathway-related genes; imperfections in these genes cause cancers. Moreover, these genes act as hubs in a biological network [14]. Thus, the ANGIE result successfully aligned the biological network from both the network structure and the biological function points of view.

6 Concluding Remarks

In this paper, we introduced a global network alignment problem that involves finding an approximate large common structure between more than two types of large graphs. The problem can be applied to the comparison of PPI networks between species. One difficulty in applying these problems is that PPI networks have sparse relationships between them. To align graphs, we introduced ANGIE, which clusters vertices by using the distribution of edge connections between clusters. We use synthetic data to show that ANGIE can produce better alignment than both HRGC as well as METIS, the latter of which is an often-utilized graph clustering algorithm. We also applied our method to PPI network data observed from fruit flies and nematode worms, and have successfully aligned the corresponding networks to identify common network structures.

In this study, we have focused on two types of graphs. Our method can be easily extended to three or more graphs. Thus, an interesting avenue for future work is to apply the proposed method to understanding time-series network changes in analyzing cellular network mechanisms for the differentiation of cells. It would also be interesting to develop the method to describe a near-optimal alignment solution.

Acknowledgements

This work was partially supported by KAKENHI (23128504) and Kayamori foundation of informational science advancement.

References

[1] S. Razick *et al.*, "irefindex: A consolidated protein interaction database with provenance," *BMC Bioinformatics*, vol. 9, no. 1, p. 405, Sep 2008.

[2] B. Kelley, B. Yuan, F. Lewitter, R. Sharan, B. Stockwell, and T. Ideker, "Pathblast: a tool for alignment of protein interaction networks," *Nucleic Acids Research*, vol. 32, p. W83, 2004.

[3] R. Sharan *et al.*, "Conserved patterns of protein interaction in multiple species," *Proc. Natl. Acad. Sci.*, vol. 102, pp. 1974–1979, 2005.

[4] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, p. 167, 2003.

[5] F. Moser, R. Colak, A. Rafiey, and M. Ester, "Mining cohesive patterns from graphs with feature vectors," in *SDM '09*, 2009, pp. 593–604.

[6] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 359–392, 1998.

[7] Bayati *et al.*, "Algorithms for large, sparse network alignment problems," in *ICDM '09*, Washington, DC, USA, 2009, pp. 705–710.

[8] C. H. Q. Ding *et al.*, "A min-max cut algorithm for graph partitioning and data clustering," in *ICDM '01*. IEEE Computer Society, 2001, pp. 107–114.

[9] Y. Tian, R. A. Hankins, and J. M. Patel, "Efficient aggregation for graph summarization," in *SIGMOD '08*, New York, NY, USA, 2008, pp. 567–580.

[10] B. Long, Z. Zhang, and P. S. Yu, *Knowledge and Information Systems*, vol. 24, pp. 393–413, 2009.

[11] A. Strehl and J. Ghosh, "Relationship-based clustering and visualization for high-dimensional data mining," *INFORMS J. on Computing*, vol. 15, no. 2, pp. 208–230, 2003.

[12] S. Zhong and J. Ghosh, "A unified framework for model-based clustering," *J. Mach. Learn. Res.*, vol. 4, pp. 1001–1037, 2003.

[13] M. Kanehisa and S. Goto, "KEGG: Kyoto Encyclopedia of Genes and Genomes," *Nucleic Acids Research*, vol. 28, no. 1, pp. 27–30, 2000.

[14] R. Sompallae, V. Stavropoulou, M. Houde, and M. G. Masucci, "The mapk signaling cascade is a central hub in the regulation of cell cycle, apoptosis and cytoskeleton remodeling by tripeptidyl-peptidase ii," *Gene Regul Syst Bio*, vol. 2, pp. 253–265, 2008.