Drawing Semi-bipartite Graphs in Anchor+Matrix Style

Kazuo Misue and Qi Zhou*
Department of Computer Science, University of Tsukuba
Tsukuba, Ibaraki, Japan
misue@cs.tsukuba.ac.jp, zhouqi727@gmail.com

***Abstract*—A bipartite graph consists of a set of nodes that can be divided into two partitions such that no edge has both endpoints in the same partition. A semi-bipartite graph is a bipartite graph with edges in one partition. Anchored map is a graph drawing technique for bipartite graphs and provides aesthetically pleasing layouts of graphs with high readability by restricting the positions of nodes in a partition. For this research, the objects of the anchored map technique were extended to semi-bipartite graphs. A hybrid layout style of anchored maps and matrix representations are proposed, and an automatic drawing technique is shown. The proposed technique arranges the nodes in one partition on a circumference like the anchored map of bipartite graphs. It also divides nodes in the other partition with edges into clusters and represents them in the matrix representations to make it easy to see connective subsets.**

***Keywords*-network visualization, graph drawing, semi-bipartite graph, anchored map, matrix representation**

I. Introduction

Anchored map is a graph drawing technique for bipartite graphs [8], [9]. A *bipartite graph* consists of a set of nodes that can be divided into two partitions such that no edge has both endpoints in the same partition. The anchored map technique is based on the unrestricted placement of nodes such as with the spring-embedder model [1]. However, it restricts the positions of nodes in a partition to provide aesthetically pleasing layout of graphs with high readability.

The purpose of this research is to extend the object of the anchor map technique to semi-bipartite graphs. A *semi-bipartite graph* is a bipartite graph with edges in one partition. We see many semi-bipartite graphs in the real world. For example, relations between items and consumers who bought the items make up a bipartite graph. Taking into account friendships among consumers, it becomes a semi-bipartite graph. Relations between Web pages and visitors to the pages also make up a bipartite graph, and links between Web pages creates a semi-bipartite graph. We believe visual representations of semi-bipartite graphs help with observation and analysis of such relationships with semi-bipartite graphs.

In an anchored map, nodes whose positions are restricted are called "anchors," and the other nodes are called "free nodes." In this study, we assume nodes in the partition with edges are free nodes. We can apply the anchored map technique to semi-bipartite graphs. However, it ignores edges between free nodes because there are no edges between free nodes in bipartite graphs. Therefore, edges among free nodes may bring about many problems in this situation. For example, adjacent free nodes may be placed apart from each other and edges connecting free nodes needlessly cross each other. These problems cause low readability of the layout.

We developed a new anchored map technique to solve such problems and draw semi-bipartite graphs as anchored maps. We took the following two measures.

- introduce aesthetic criteria considering edges between free nodes.
- introduce the matrix representation for the edges between free nodes.

We explain our new drawing technique we developed for semi-bipartite graphs as drawing objects. The combination of the anchored-map style and matrix representations is one of the most important features of this technique. We show the drawing standard, the aesthetic criteria, and a drawing procedure for semi-bipartite graphs.

II. Related Work

A. Bipartite Graph Drawing

Some studies on bipartite graphs have resulted in layout techniques as a building block of drawing a graph in the Sugiyama style [13]. For example, Newton et al. proposed a heuristic for two-sided bipartite graph drawing, where nodes in two partitions were laid out on two parallel lines [11]. Other studies tried to change the style for bipartite graphs. Zheng et al. described two layout models and proved theorems of edge crossing for these models [15]. Giacomo et al. proposed drawing bipartite graphs on two curves so that the edges do not cross [3]. These studies proposed techniques to minimize edge crossing in the two-sided style or its extended styles.

The anchored map technique [8] is also a drawing technique for bipartite graphs. We introduce research related to this in Section II-C because it is one of the most important precursors of this research.

B. Semi-bipartite Graph Drawing

To the best of our knowledge, there have been few studies on drawing semi-bipartite graphs. A semi-bipartite graph

* His current affiliation is with Nomura Research Institute.

1550-6037/11 $26.00 © 2011 IEEE
DOI 10.1109/IV.2011.24

model has been introduced by Xu et al. [14]. They gave a model of the gene ontology network as a semi-bipartite graph and proposed drawing methods.

C. Anchored Map

The anchored map technique restricts positions of nodes, called "anchors", in one partition, to provide aesthetically pleasing layout of graphs with high readability. Misue [8], [9] described a method for restricting the positions of anchors on a circumference.

Extensions were proposed for anchored maps. Ito et al. proposed an extension of the drawing space to three dimensions [6]. The positions of anchors are restricted on a spherical surface, so anchors have the freedom of two dimensions. Sato et al. proposed a method for showing clusters of free nodes [12]. Free nodes are divided into clusters by using similarities of sets of adjacent anchors and drawn with iso-similarity contour curves. We also use the idea of clustering free nodes. However, we divide them by using edges between free nodes and represent clusters as matrix representations. Ito et al. introduced another hierarchical structure into the partition of anchors and developed a technique using the hierarchical circular layout of bipartite graphs [7]. It is effective for visualization of large-scale bipartite graphs. The technique is related to our technique from the viewpoint that some structural elements are added to one partition. However, we add them to the partition of free nodes and propose a new representation style.

D. Matrix Representation

The techniques to express a network in the matrix representation are used for a long time. The matrix representation just displays an adjacency matrix, which is a mathematical expression of a graph. This enables quick reading of the adjacency of specific nodes, but it is unsuitable for tracing a path consisting of several nodes. Ghoniem compared matrix representation with node-link representation for easy reading of information from the representations [2]. MatrixExplorer is a tool that uses the matrix representation and a node-link representation to use the advantages of both representations [4]. In NodeTrix, matrix representations are locally combined with a node-link representation to provide advantages of the matrix representations [5]. The representation style used in this study was affected by NodeTrix. The matrix representation is combined with the anchored maps, which is a representation style of node-link diagrams.

III. Anchored Map + Matrix Representation

We propose a new representation style that combines an anchored map with matrix representation as a representation style of a semi-bipartite graph. We formalize the semi-bipartite graph and clarify the style by showing the drawing standard and the aesthetic criteria.

A. Semi-bipartite Graph

A *bipartite graph* is defined as $G_B = (A, B, E)$ with node partitions A and B, which are disjoint, i.e., $A \cap B = \emptyset$, and E is a finite set of edges, i.e., $E \subseteq A \times B$.

A *semi-bipartite graph* is defined as $G_{SB} = (A, B, E, F)$, where $G = (A, B, E)$ is a bipartite graph, and F is a finite set of edges connecting two nodes in B, i.e., $F \subseteq \{\{u, v\} | u, v \in B, u \neq v\}$[1]. We call the elements in set F "inner edges" of set B.

B. Combining anchored map and matrix representation

We devised a new drawing technique for semi-bipartite graphs. With this technique, nodes of set A are arranged on a circumference. Nodes of set B are divided into clusters by using inner edges. The clusters are arranged at appropriate positions expressing relations to other nodes and clusters well. Every cluster is represented in the matrix representation. Nodes in a matrix are arranged in an order that clearly expresses relations in the cluster.

C. Drawing Standards

The following standards are used for drawing anchored maps [9].

- C1 Nodes are represented as bullets (or small icons).
- C2 Nodes in partition A are arranged on a circumference.
- C3 Nodes in partition B have no limitation with regards to the coordinate system.
- C4 Edges of set E are represented as straight line segments.

We revised the drawing standards as follows to combine an anchored map with the matrix representation. We divided C1 into C1a and C1b and added C4b to C4.

- C1a Nodes in partition A are represented as bullets.
- C1b Nodes in partition B are represented as matrices or bullets.
- C4b Edges of set F are represented as matrices or straight line segments.

A node cluster is represented as a matrix. Therefore, edges of set F, whose endpoints are in the same cluster, are represented in a matrix. Other edges of set F are represented as line segments. The matrix representation of the clusters is the same as that widely used for general graphs. A row and column each corresponds to a node, and a symbol at the intersection of a row and column denotes that a node corresponding to the row and a node corresponding to the column are adjacent to each other.

[1] F is a set of undirected edges from this definition. If we want to express directed edges, we may use another definition such as $F \subseteq \{\langle u, v\rangle | u, v \in B, u \neq v\}$.

D. Aesthetic Criteria

The following rules are used for the aesthetic criteria of anchored maps [9].

- R1 Nodes are separated mutually more than the lowest distance.
- R2 Adjacent nodes are laid out as closely as possible (minimize the total length of edges.)
- R3 The number of edge crossings is as small as possible.
- R4 Anchors adjacent to common free nodes are laid out as closely as possible.
- R5 Free nodes adjacent to common anchors are laid out as closely as possible.

We expanded the aesthetic criteria to cover semi-bipartite graphs as drawing objects and combine an anchored map with the matrix representation. We replaced rule R2 with R2', rule R4 with R4', and added rule R6.

- R2' Minimize the total length of edges.
- R4' Anchors connected to each other are laid out as closely as possible.
- R6 Nodes adjacent to common nodes are laid out as closely as possible in a matrix.

Rule R2' is the same as R2 at a glance, but upon closer inspection they are different. The new representation style includes the matrix representation, and the edge lengths depend on the connecting points on a matrix. Therefore, we describe rule R2' with edge lengths rather than positions of adjacent nodes.

In addition, in a semi-bipartite graph, two anchors not sharing any free nodes may connect to each other via inner edges. We believe such anchors should be placed close to each other; therefore, we modified rule R4 to R4'.

We adopted rule R6 for the matrix representation to express clusters of free nodes. Rule R6 is formally defined by minimizing q in expression (1).

$$q = \sum_{f \in M} \sum_{u,v \in A(f)} |p(u) - p(v)|, \qquad (1)$$

where M is a set of free nodes represented in a matrix, and $A(f)$ is a set of nodes adjacent to the free node f in the subgraph[2] consisting of cluster M, that is, $A(f) = \{v \in M | \{v, f\} \in F\}$. $p(v)$ represents the position of node v in the matrix, i.e., $p : v \rightarrow \{1, 2, \cdots, |M|\}$.

We need to satisfy the aesthetic criteria as much as possible. We should give priority to rules because two or more rules may conflict. However, it is not easy to control the priority of rules using force-directed techniques. Therefore, we do not argue the priority here.

[2]the node-induced subgraph of G_{SB} by M

IV. Drawing Method

A. Outline of Layout Procedure

The drawing procedure is as follows.

1: Divide free nodes into clusters; creating a reduced graph
2: Determine the anchor order
3: Determine the positions of free nodes (i.e., free node clusters)
4: Determine the free node order in matrices
5: Link edges

Step 1: We divide nodes in set F into clusters by using inner edges. Each cluster includes nodes strongly connected to each other, and fewer edges connect different clusters. We used Newman's algorithm [10] to create such free-node clusters. Every cluster is replaced with a single node to create a reduced graph.

If we obtain a bipartite graph by this reduction, we can just use the current anchored map technique. In most cases, however, the reduced graph is also a semi-bipartite graph because it is generally impossible to make clusters so that there is no edge connecting the clusters. Therefore, a technique for laying out a semi-bipartite graph as an anchored map is necessary.

Step 2: Steps 2 and 3 are for obtaining the anchored map of a semi-bipartite graph. In step 2, anchor positions are determined using rules R2', R3, and R4'. The procedure is similar to that for bipartite graphs, but edges of set F should be considered.

Step 3: Free node positions are determined using rules R1, R2', R3, and R5. The procedure is also similar to that for bipartite graphs, but edges of set F should be considered as well. If we want to obtain an anchored map without matrix representation, we draw line segments connecting adjacent nodes after determining positions of the free nodes in step 3.

Step 4: The clusters made in step 1 are represented in the matrix representation. The order of nodes in the matrix is then determined if it satisfies rule R6.

Step 5: For an anchored map of a bipartite graph, each node is expressed at a point (a bullet or a small icon), and each edge is drawn in a line segment. We do not need to worry about routing of edges if the node positions have been determined. Conditions for anchored maps of semi-bipartite graphs are basically the same for bipartite graphs. However, there are four positions to connect an edge to a free node when the node is represented in a matrix. We have to choose one of four positions for every edge to satisfy rule R2' and R3.

B. Drawing Anchored Map for Semi-bipartite Graph

As stated above, steps 2 and 3 are used to obtain the anchored map of a semi-bipartite graph. Therefore, we explain them as a drawing technique of semi-bipartite graphs. The technique we explain here is an extension of the technique suggested in a previous study [8] on bipartite graphs.

1) Determining Anchor Order: Because we determined that the anchors are arranged on a circumference at equal intervals in the drawing standard, what we should do in this step is to determine the anchor order on the circumference. The basic algorithm to determine the anchor order is the same as in the previous study. We used a technique that gradually improves the order while evaluating which order would be good, that is, satisfy the rules.

We have to determine the exact positions of the free nodes to evaluate the goodness of the order of the anchor order. The positions are determined using the spring-embedder model, which consumes a large amount of computation time. Therefore, the previous study used an index called "penalty", which can determine the goodness of free node placement.

Some penalties were defined based on rules R2, R3, and R4 [9]. Because we cannot evaluate rules R2 and R3 if the positions of the free nodes are not determined, we use the value of a penalty assuming that the free nodes are placed at their ideal positions.

For semi-bipartite graphs, we cannot determine the ideal positions of free nodes by using only the anchor positions because we should consider edges between the free nodes. Therefore, the use of penalty derived from rules R2' or R3 may be difficult.

We defined a penalty using rule R4'. This penalty is expressed with expression (2).

$$p = \sum_{u,v \in A, u \neq v} p(u,v) \tag{2}$$

$$p(u,v) = \frac{d_c(u,v)}{w_E \cdot p_E(u,v) + w_F \cdot p_F(u,v)} \tag{3}$$

Let p_E and p_F be the number of elements of set E and the number of the element of set F included in a path, respectively. That is, $p_E(u,v) = |P(u,v) \cap E|$, $p_F(u,v) = |P(u,v) \cap F|$, where $P(u,v)$ is the path between node u and v. Suppose that $d_c(u,v)$ is the distance between anchors u and v on the circumference. w_E and w_F are constant numbers to give weights to the elements of F and elements of E included in the path. In our implementation, $w_E = 1$ and $w_F = 2$. We found these values from experience of experiments.

When nodes u and v are not connected, i.e., there is no path between u and v, let $p(u,v) = 0$.

2) Determining Positions of Free Nodes: After we determined the anchor positions, we fix the anchors at the positions and arrange the positions of free nodes using the spring-embedder model [1].

C. Drawing Matrices for Node Clusters

In step 4, nodes in clusters are represented in the matrix representation.

1) Determining Free Nodes Order: We make the row and column orders the same. Therefore, what we should do in this step is to determine the node order.

We propose a variation of the barycenter (BC) method to satisfy rule R6, i.e., to place nodes related to each other close together. The BC method is a heuristic algorithm which has been proposed to reduce edge crossings in the hierarchical layout of directed graphs [13].

Taking a hint from the BC method, we developed the algorithm shown in Figure 1 to determine the node order in a matrix. q in the algorithm is expressed by expression (1). For the number of repeat times m, we used the number of nodes in cluster M in our implementation.

```
give an initial order to every node in M
report m times {
    for each node,
        compute the barycenter of its adjacent nodes.
    sort all nodes by their barycenters.
    calculate q and record the order with the value.
}
select the order with the minimum value of q
```

Figure 1. Algorithm to determine node order in matrix

V. Evaluation

A. Evaluation of Penalty

A good penalty precisely predicts how the final layout satisfies the aesthetic criteria. Because the penalty we propose is derived from rule R4', we can expect a certain effect for R4'. We conducted an experimental evaluation on how well the penalty can predict for rules R2' and R3.

We examined correlations between penalty values and total edge lengths (rule R2') and between penalty values and the number of edge crossings (rule R3). We generated 100 random semi-bipartite graphs with 10 and 15 anchors. The numbers of free nodes and edges were varied. For each graph, we randomly generated 1000 anchor orders to calculate correlation coefficients to reduce experiment time.

For the graphs with 10 anchors, 99% had a correlation coefficient over 0.6, and 73% had one over 0.8 for rule R2'. For rule R3, 84% of the graphs had a correlation coefficient over 0.6. The results for the graphs with 15 anchors are almost the same as with the 10 anchors. For rule R2', 98% of the graphs had a correlation coefficient over 0.6 and 70% had one over 0.8. For rule R3, 95% of the graphs had a correlation coefficient over 0.6. From these results, we believe the penalty we propose is effective in predicting the aesthetic criteria. We can expect it is especially effective in the predicting edge length (rule R2').

B. Evaluation of BC Method Variation

We conducted an experiment to examine how rule R6 is satisfied using the variation of the BC method.

For each randomly generated undirected graph, we calculated a value of rule R6 (q) for every node-order pattern. We also calculated q for the order found with the variation of the BC method, and examined the ranking of value of q in all patterns.

We generated 1000 random graphs with five, six, and seven nodes to examine the ranking of values of q. For the graphs with five nodes, 70% had a value in the top position. In other words, the method found the optimal orders of about 700 graphs. For the graphs of the other two sizes, 60% – 65% had optimal orders. From these results, we believe that the variation of the BC method is effective for rule R6.

VI. Examples and Discussion

We show examples drawn using the technique we developed. The semi-bipartite graph shown in this section was extracted from a social networking service. We expressed relations between users and communities using a bipartite graph and expressed friendships using inner edges. All examples represent the same semi-bipartite graph, and communities are represented as anchors.

Figure 2(a) shows an example drawn using the technique of a previous study [9]. The positions of all nodes were determined without regard to the inner edges. The edges were then drawn as straight line segments. In other words, the inner edges do not affect the placement of nodes.

Figure 2(b) shows an example drawn also using the technique of the precedent study. The positions of anchors were determined without regard to the inner edges. We used the inner edges when determining the position of free nodes by using the spring-embedder model.

Figure 2(c) shows an example of the same graph drawn with only steps 2 and 3 in the developed procedure. Free nodes were not divided into any clusters, and every free node was drawn as a bullet. We can see that the number of crossings of the inner edges in Figure 2(c) are less than those in Figure 2(b).

Figure 2(d) shows an example of the same graph drawn using the developed technique. Free nodes were divided into eight clusters, and each was drawn in the matrix representation. We can clearly see the clusters and roughly understand the connection patterns inside the clusters.

VII. Conclusions

We developed a drawing technique for semi-bipartite graphs. We extended the anchored map drawing technique of bipartite graphs and combined it with the matrix representation into a hybrid representation. In the new representation, clusters of free nodes are represented in the matrix representation, so we can easily see the connect components of the nodes consisting of inner edges, which is a feature of semi-bipartite graphs. We defined a penalty for arranging anchors in semi-bipartite graphs and developed a variation of the BC method for determining the order of free nodes in a matrix.

There is room for improving the penalty we defined and the variation of the BC method we developed. We plan to develop more effective techniques through experimental evaluation.

Acknowledgements

This work was supported in part by the Okawa Foundation for Information and Telecommunications.

References

[1] Peter Eades, A Heuristic for Graph Drawing, Congressus Numerantium 42, pp. 149–160, 1984.

[2] Mohammad Ghoniem, Jean-Daniel Fekete, and Philippe Castagliola, A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations, IEEE Symposium on Information Visualization 2004, pp. 17–24, 2004.

[3] Emilio Di Giacomo, Luca Grilli, and Giuseppe Liotta, Drawing Bipartite Graphs on Two Curves, Graph Drawing: 14th International Symposium, GD 2006, Lecture Notes in Computer Science, Vol. 4372, pp. 380–385, Springer Berlin Heidelberg, 2007.

[4] Nathalie Henry and Jean-Daniel Fekete, MatrixExplorer: a Dual-Representation System to Explore Social Networks, IEEE Transactions on Visualization and Computer Graphics, Vol. 12, No. 5, pp. 677–684, 2006.

[5] Nathalie Henry, Jean-Daniel Fekete, and Michael J. McGuffin, NodeTrix: A Hybrid Visualization of Social Networks, IEEE Transactions on Visualization and Computer Graphics, Vol. 13, No. 6, pp. 1302–1309, 2007.

[6] Takao Ito, Kazuo Misue, Jiro Tanaka, Sphere Anchored Map: A Visualization Technique for Bipartite Graphs in 3D, Proceedings of 13th International Conference on Human-Computer Interaction (HCI International 2009), Human Interface, Part II, HCII 2009, LNCS 5611, pp.811-820, 2009.

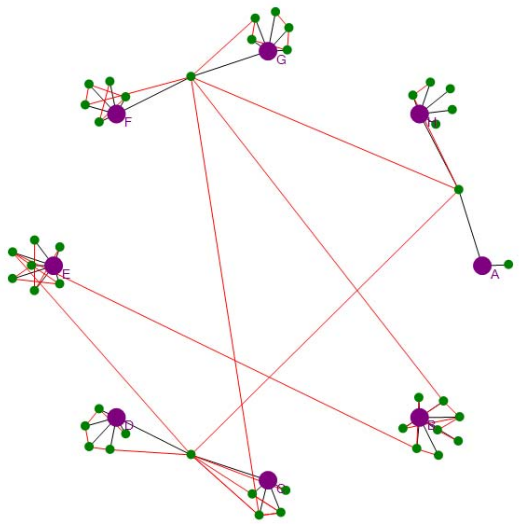
[7] Takao Ito, Kazuo Misue, Jiro Tanaka, Drawing Clustered Bipartite Graphs in Multi-Circular Style, Proceedings of 14th International Conference Information Visualization (IV10), pp.23-28, 2010.

[8] Kazuo Misue: Drawing Bipartite Graphs as Anchored Maps, Proceedings of Asia-Pacific Symposium on Information Visualization (APVIS 2006), CRIPT, vol. 60, pp. 169–177, 2006.

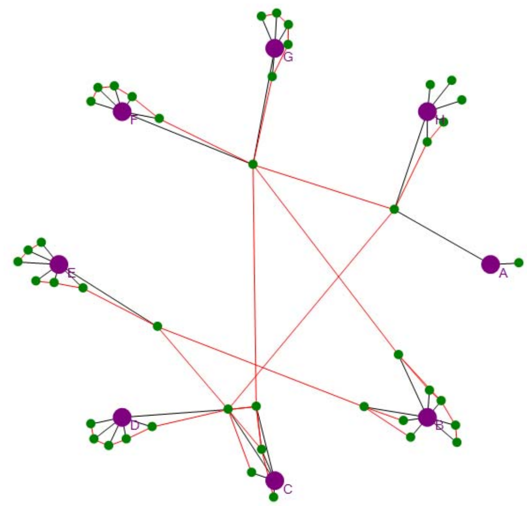
[9] Kazuo Misue, Anchored Map: Graph Drawing Technique to Support Network Mining, IEICE Trans. Inf. & Syst., Vol. E91-D, No. 11, pp. 2599–2606, 2008.

[10] M. E. J. Newman. Fast algorithm for detecting community structure in networks. Physical Review E 69, 066133, 2004.

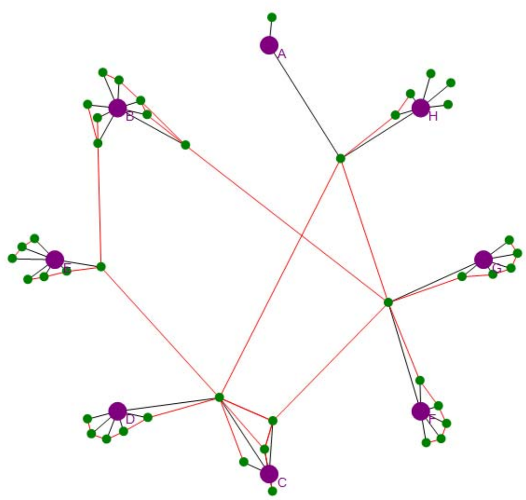
[11] Matthew Newton, Ondrej Sýkora, and Imrich Vrt'o, Two New Heuristics for Two-Sided Bipartite Graph Drawing, Graph Drawing: 10th International Symposium, GD 2002, Lecture Notes in Computer Science, Vol. 2528, pp. 312–319, Springer Berlin Heidelberg, 2002.



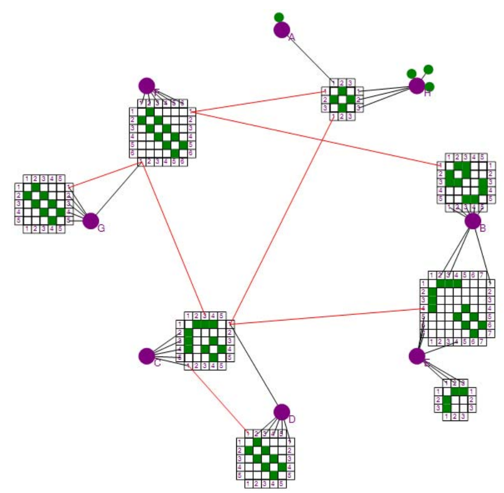
(a) Example drawn using technique of previous study. Nodes were arranged without regard to inner edges.



(b) Example drawn using technique of previous study. Anchors were arranged without regard to inner edges.



(c) Example drawn using part of proposed technique. Positions of nodes were determined with regard to inner edges.



(d) Example drawn using proposed technique. Clusters of free nodes were drawn in matrix representation.

Figure 2. Example of visualized semi-bipartite graph

[12] Shuji Sato, Kazuo Misue and Jiro Tanaka: Readable Representation for Large-Scale Bipartite Graphs, 12th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2008), LNAI 5178, pp. 831–838, 2008.

[13] Kozo Sugiyama, Shojiro Tagawa, and Mitsuhiko Toda, Method for Visual Understanding of Heirarchical System Structures, IEEE Transactions on Systems, Man and Cybernetics, Vol. 11, Issue 2, pp. 109–125, 1981.

[14] Kai Xu, Rohan Williams, Seok-Hee Hong, Qing Liu, and Ji Zhang, Semi-bipartite Graph Visualization for Gene Ontology Networks, Graph Drawing: 17th International Symposium, GD 2009, Lecture Notes in Computer Science, Vol. 5849, pp.244–255, Springer Berlin Heidelberg, 2010.

[15] Lanbo Zheng, Le Song, and Peter Eades, Crossing Minimization Problems of Drawing Bipartite Graphs in Two Clusters, Proceedings of Asia-Pacific Symposium on Information Visualization (APVIS 2005), CRPIT, Vol. 45, pp. 33–38, 2005.