Drawing Clustered Bipartite Graphs in Multi-Circular Style

Takao Ito, Kazuo Misue and Jiro Tanaka
Department of Computer Science
University of Tsukuba
ito@iplab.cs.tsukuba.ac.jp, misue@cs.tsukuba.ac.jp and jiro@cs.tsukuba.ac.jp

***Abstract*—Bipartite graphs are often used to illustrate relationships between two sets of data, such as web pages and visitors. At the same time, information is often organized hierarchically, for example, web pages are divided into directories by their contents. The hierarchical structures are useful for analyzing information. Graphs with both a bipartite structure and a hierarchical structure are called "clustered bipartite graphs." A new clustered bipartite graphs visualization technique was developed for representing both bipartite and hierarchical structures simultaneously. In this technique, nodes in one set of the bipartite graph, which are leaves of a tree, are arranged in hierarchical multi-circular style. Then, nodes in the other set of the bipartite graph are arranged by the force-directed method. The technique enables step-by-step exploration for large-scale bipartite graphs.**

***Keywords*-graph drawing, bipartite graph, hierarchical structure, anchored maps, circular layout;**

I. Introduction

Many kinds of information can be represented by graphs. We often look at the relationships between two sets of data to analyze information. For example, the relationships between customers and goods are studied to understand buying patterns, and relationships between web pages and visitors are studied to gain more understanding of website usages. Graphs representing such relationships are called bipartite graphs. Visualizing graph structures is one efficient analysis methods. We understand many-to-many relationships by visualizing bipartite graphs. However, although we can see overviews, understanding details of the information is difficult when large-scale bipartite graphs are visualized by conventional techniques [1], [2], [3].

Hierarchical structures are useful for analyzing information [4], [5], [6]. They are often used to organize large-scale information. Many kinds of goods have been divided into categories with a hierarchical structure. Web pages have been divided into directories by their contents.

We propose representing both bipartite and hierarchical structures simultaneously. Visualizing both structures makes it easier to get an overview and to explore details of the information at the same time. We call graphs with both a bipartite structure and a hierarchical structure "clustered bipartite graphs." Clustered bipartite graphs are graphs with recursive clustering structures over the nodes in one set of the bipartite graph (Figure 1).

Visualizing clustered bipartite graphs enables the user to explore the information in the following way. To analyze consumer buying information, we first analyze consumer preferences among categories. Then, we analyze preferences for goods in each category. Finally, differences among categories are analyzed by observing preferences for goods in each category and comparing them. To analyze website usages, we first analyze Web-browsing habits for the site contents. These data are analyzed by observing relationships between directories and visitors. Then, Web-browsing habits for each page and relationships between types of content are analyzed. Introducing the hierarchical structures into the bipartite structures enables step-by-step exploration of large-scale graphs.

We developed a visualization technique for drawing clustered bipartite graphs. In the technique, nodes in one set with the hierarchical structure are called "anchors" and are arranged in hierarchical multi-circular style. Nodes in the other set are called "free nodes" and are arranged using the force-directed method [7]. We call this visualization technique "hierarchical anchored maps."

In this paper, we describe a method of drawing hierarchical anchored maps, focusing to how arrange anchors in hierarchical multi-circular style. First, we explain the aesthetic criteria for hierarchical anchored maps, and then we explain a method to arrange the nodes to satisfy the criteria. Finally, we show drawing examples and explain the effectiveness of the visualization technique.

II. Related Work

A. Bipartite Graph Drawing

Zheng et al. [1] described two layout models for bipartite graphs and proved theorems of edge crossing for these models. Giacomo et al. [2] proposed drawing bipartite graphs on two curves so that the edges do not cross. Newton et al. [3] proposed new heuristics for two-sided bipartite graph drawing. These studies proposed algorithms to minimize edge crossing in the two-sided style or extended models but they did not consider large-scale bipartite graphs.

Misue[8] described anchored maps as a drawing technique for large-scale bipartite graphs. In anchored maps, nodes in one set of a bipartite graph are called *anchors* and nodes in the other set are called *free nodes*. Anchors are arranged on a circumference at equal intervals and free nodes are arranged using the force-directed method [7]. Misue proposed an algorithm to decide the order of anchors by hull-climb. Similar styles are used in systems proposed by Thiel et al. [9] and Donovan et al. [10]. However, it becomes more difficult to understand details as the number of anchors arranged on the circumference increases, because the anchors' arrangement approaches a straight line (Figure 6(a)). Our technique is am improvement over anchored maps because it represents overviews and details simultaneously.

Naud et al. [11] developed the 3D-SE Viewer, in which bipartite graphs are arranged on concentric spheres in 3D space. Ito et al. [12] also developed a 3D visualization technique for bipartite graphs called sphere anchored maps, in which nodes in one set are arranged on a sphere. 3D visualization techniques increase the readability, although occlusion problems occur as a result of projecting. These techniques require interaction costs. To avoid these costs, we aim to develop a 2D visualization technique.

B. Visualization of Compound Graphs

There have benn several studies about visualizing two structures simultaneously. Sugiyama et al. [13] described a drawing technique for compund digraphs which represent adjacency relationships and inclusion relationships between nodes. Eades et al. [4] described visualization techniques for clustered graphs that have both a general graph structure and a tree structure. Ho et al. [5] developed 3D visualization techniques for clustered graphs. These techniques represent clusters using cone trees or free-style 3D tree layouts. Omote et al. [14] described intersecting clustered graphs that have a general graph structure and a tree structure except to allow sharing of leaves among clusters. Itoh et al. [15] described a hybrid space-filling and force-directed method for graphs that have a general graph structure and multiple categories.

We focus on a new class of graphs; the concept of clustered graphs is introduced into bipartite graphs. We have developed a visualization technique for such graphs.

III. Hierarchical Anchored Maps

A. Clustered Bipartite Graphs

A clustered bipartite graph $G = (A, F, C, E, T)$ is a graph that has both a bipartite structure and a tree structure. $G_B = (A \cup F, E)$ is a bipartite graph. A and F are finite sets of nodes, and A and F are disjoint. E is a finite set of edges, and E is a subset of $A \times F$. $G_T = (A \cup C, T)$ is a tree, where C is a set of non-leaf nodes (clusters). Leaves of G_T are exactly A. Figure 1 is a clustered bipartite graph.

B. Layout Style of Hierarchical Anchored Maps

We improved anchored maps to represent clustered bipartite graphs. Our basic idea is to arrange anchored maps recursively. In our technique, small anchored maps are formed by sub bipartite graphs. Figure 2(a) is a visualization of web pages and visitors using our technique. There are

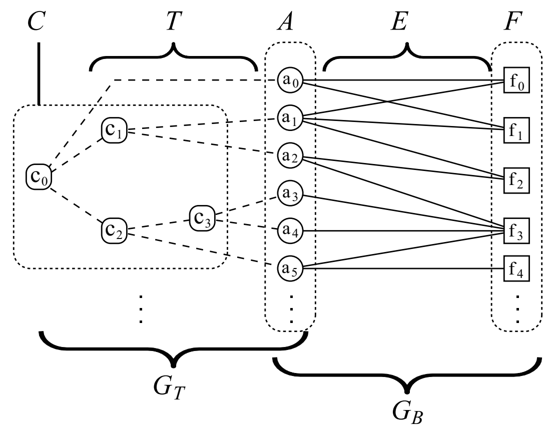


Figure 1. Clustered bipartite graph.

thirteen pages and three directories "/a", "/b," and "/b/c." In our technique, an anchored map is formed by web pages under each directory. The relationships of a bipartite graph composed of each cluster are obtained by forming small anchored maps. The hierarchical structure and relationships among the anchored maps are obtained by arranging small anchored maps while keeping the overviews.

C. Terms

We introduce the terms used in our hierarchical anchored maps. A map formed by anchors and child clusters included in each cluster is called *a cluster map*. A map formed by a child cluster is called *a child cluster map* of the cluster map formed by its parent cluster. In an opposite manner, a map formed by a parent cluster is called *a parent cluster map* of its child cluster maps. Moreover, maps formed by clusters with same parent cluster are called *brother cluster maps*. Additionally, anchors and child cluster maps are called *child elements* of their parent cluster map. In Figure 2(a), the red cluster map labeled "/a" is a child cluster map of the blue cluster map labeled "/" and the blue cluster map labeled "/" is a parent cluster map of the red cluster map labeled "/a." The red cluster map labeled "/a" is a brother cluster map of the green cluster map labeled "/b." The yellow cluster map labeled "/b/c" is called a grandchild cluster map of the blue cluster map labeled "/."

Child elements in each cluster map are arranged on a circumference of a circle. The circle is called *the circle of the cluster map*. *The position of cluster map* is represented by the center of the circle and *the radius of the cluster map* is represented by the radius of the circle(Figure 2(a)).

The *Angles* of child elements define their locations in their parent cluster map. The *direction* of a cluster map is the origin of the angles in the cluster map. Figure 2(b) illustrates the angles and the direction.

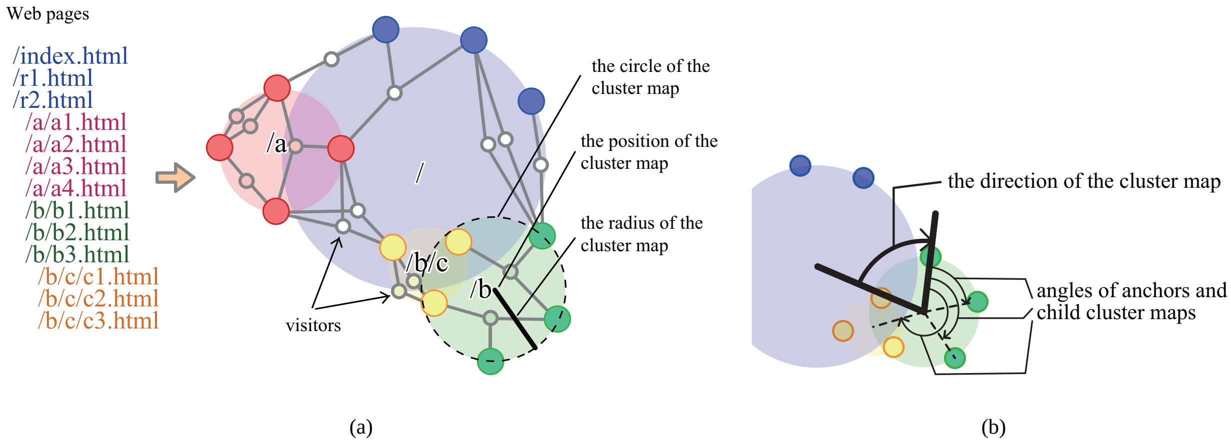


Figure 2. (a) Relationships between web pages and visitors drawn as a hierarchical anchored map. (b) *Angles* of child elements define their locations in their parent cluster map. *The direction* of the cluster map is the origin of the angles.

D. Aesthetic Criteria

We define the following aesthetic criteria to develop our method of drawing hierarchical anchored maps.

- E1 The number of free nodes arranged in unrelated cluster maps is minimized.
- E2 The length of the intersection between edges and cluster maps is minimized.
- E3 Anchors connected to common free nodes are laid out as closely as possible.
- E4 The area efficiency is as high as possible.

We use E1 and E2 because they are useful for splitting information into inside and outside areas of cluster maps. E3 is also used in Misue's anchored maps. It is useful for representing the information shown on bipartite graphs. E4 is used because it is useful for surveying the figure.

IV. Drawing Methods

A. Outline of the Layout Procedure

The algorithm proposed by Misue is a method for drawing a bipartite graph with a single circle. Therefore, it is enough only by deciding the order of the anchors to satisfy aesthetic criteria E3 in anchored maps.

To draw hierarchical anchored maps, we have to decide the angles that child elements occupy, the positions of cluster maps, and the order of the cluster maps. We focused on developing a method to arrange multiple circles with connectivity among cluster maps while maintaining overviews.

A hierarchical anchored map is arranged by the relative relations of each child cluster map to its parent cluster map. The relative relations are computed in three steps:

- Step1 Angles of child elements in each cluster map are computed.
- Step2 The radius and position of each child cluster map are computed.
- Step3 The direction of each child cluster map in its parent cluster map is computed.

B. Step1 : How the Angles of Child Elements in Each Cluster Map are Decided

First, the order of anchors and child cluster maps in their parent cluster map is obtained using Misue's algorithm. Each child cluster map in the parent cluster map is assumed as a single anchor (Figure 3). Edges connecting to anchors included in the parent cluster map are only considered to obtain the order because we want to represent the local relationships organized by the sub bipartite graph. In the case of the graph shown in Figure 1, when the order in the cluster map formed by c_2 is computed, $G'_B(c_2) = (\{c_3, a_5\} \cup \{f_3, f_4\}, \{(c_3, f_3), (a_5, f_3), (a_5, f_4)\})$ is a bipartite graph input to Misue's algorithm.

Then, child cluster maps are assigned their spaces in proportion to the number of anchors they include. This method gives the relative positions of anchors and child cluster maps from the position of their parent cluster map. The positions are on the circumference of the parent cluster map and they are used to decide the angles of child elements.

C. Step2 : How the Radius and Position of Each Child Cluster Map are Decided

Radii and positions of cluster maps affect E1 and E4. We tried three styles to find one that meets our aesthetic criteria.

[a] First, we tried a style in which the lengths of segments between adjacent anchors in the parent cluster map are used as the lengths of segments between adjacent anchors in each child cluster map to keep overviews and child cluster maps are arranged to be adjacent to their parent cluster map. However, if circles of child cluster maps are arranged to be adjacent to the circle of their parent cluster map, grandchild cluster maps may intersect with the parent cluster map. This

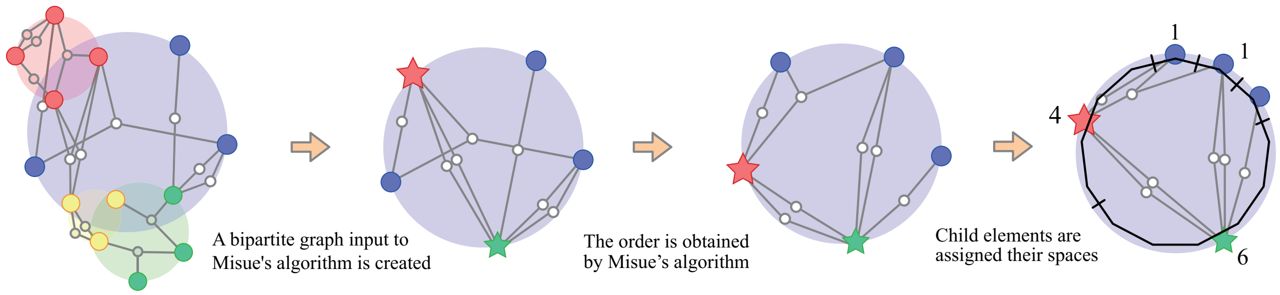


Figure 3. Procedure for obtainning the angles of child elements in the root cluster map in the graph in Figure 2(a).

has the potential to infringe E1. Consequently, the position of a child cluster map is arranged at a distance of the sum of the radius of its parent cluster map, the radius of the child cluster map, and the sum of the diameters of the cluster maps below the child cluster map (Figure 4(a)). In this style, if the graphs have deep hierarchy, the area efficiency is low.

[b] Second, we arranged child cluster maps inside their parent cluster map. This style looks like the balloon layout [16]. However, if radii are obtained as in **[a]**, there is a possibility that unrelated free nodes will be laid out in cluster maps. This infringes E1. Therefore, the distance from the circumference of the parent cluster map to the chord is used as the diameter of each child cluster map(Figure 4(b)). Child cluster maps are arranged to be inscribed in their parent cluster map. In this style, cluster maps at deep levels are smaller than **[a]**. Therefore, the area efficiency is low.

[c] Third, we used the radii in **[a]** and arranged the child cluster maps on the circumference of the parent cluster map. However, if child cluster maps are arranged on the circumference simply, there is a possibility that unrelated free nodes could be laid out in cluster maps. Therefore, we restrict child cluster maps to being arranged over chords. Child cluster maps intersecting with chords are arranged to be adjacent to chords(Figure 4(c)). We use this style because it gives relatively good area efficiency.

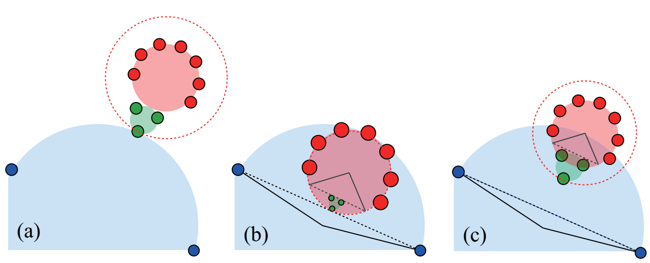


Figure 4. Three anchor layout styles: (a) circumscribed, (b) inscribed, and (c) on-circumference.

D. Step3 : How the Direction of Each Child Cluster Map is Decided

In the third step, the direction of each child cluster map on its parent cluster map is computed. The direction of each cluster map affects E2.

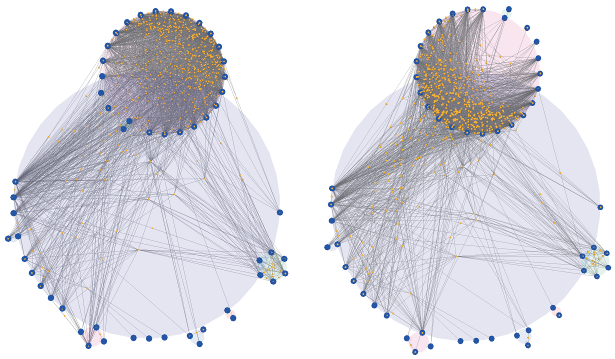
The directions of the cluster map are decided from the route cluster map to the next cluster map below it in the sequence. In exploring, anchors in child and brother cluster maps are arranged on the positions of their parent cluster maps.

We describe the method to decide the direction of each cluster map. First, we choose a certain direction d_0. Next, the length of the intersection is computed in d_0 and $d_0 + \pi$, and the direction with the shorter intersection length is chosen. Let the direction giving the shorter intersection be d1. Then, the length of the intersection is computed in $d_1 - \pi/2$ and $d_1 + \pi/2$, and the direction with the shorter intersection length is chosen from d_1, $d_1 - \pi/2$ and $d_1 + \pi/2$. The same operation is repeated, halving the width until the width becomes smaller than the threshold. Finally, the direction with the shortest intersection length is used.

When E2 is assumed to be an index of the direction, the directions of other cluster maps affect the best direction of a certain cluster map. Therefore, to minimize the length of the total interaction, we should consider this effect. Then, if the number of cluster maps is u and the direction is explored in each cluster map with n° accuracy, we need to explore u^a patterns, where $a = 360/n$. In contrast, our method needs to explore $u \log a$ patterns by ignoring the effects of brother cluster maps.

We performed an experiment in which the exploration of u^a patterns and our method are compared using E2. In the experiment, we ranked u^a patterns using E2 and investigated the ranking of the direction obtained by our method. Our method was able to obtain the directions ranked in the top 1%.

Figure 5(a) shows a layout in which the directions of cluster maps are not considered and (b) shows a layout in which the directions are considered. If the directions are not considered, edges between two clusters often intersect with cluster maps. It involves visual clutter. By contrast, we can see that consideration of the directions reduces visual cluster. Additionally, relationships between anchors and other clusters are clearer.



(a) (b)

Figure 5. Effectiveness of consideration of directions for child cluster maps. Our technique reduced the visual cluster in (b).

V. Examples and Discussion

We show a visualization of web access logs for the first author's website. Figure 6(a) shows an example of anchored maps. Visitors are drawn as free nodes and web pages are drawn as anchors. The visualization of relationships of the bipartite graph between visitors and web packages enables us to understand relationships among multi-pages.

The same data is drawn as a hierarchical anchored map in Figure 6(b). Directory information is used as the hierarchical structure for anchors. We focused on the directory "/vse" drawn in the lower right. Web pages in this directory are accessed by visitors, who make up a group of free nodes in the lower right. There is a freeware website developed by first the author in this directory. We found that there are three types of visitors: those who accessed "/vse/" and "/vse/download.html", "/vse" and "/vse/manual.html" and all pages in "/vse."

The graph shown in Figure 6 has a tendency that many free nodes are arrange around the circumference in anchored maps. For example, free nodes are arranged on anchors in the zoomed part of Figure 6(a). In the single-circular layout, it is difficult to understand relationships in a part of the graph. By contrast, our technique improved this problem by forming cluster maps. Our technique can draw such graphs while maintaining readability.

Our technique supports explorative information analysis because it enables us to watch local relationships after understanding overviews. Local relationships may be understood even by anchored maps. Unfortunately, there are 14 directories in the data shown in Figure 6. If local relationships are visualized by each directory, 14 visualizations are needed. Viewers of such visualizations may lose their way because many visualizations are independent from each other. Our technique guides where the user should look, because data are visualized in a single view, overviews are given and the hierarchical structure is represented. It enables step-by-step exploration for large-scale bipartite graphs.

A limitation of our technique is that the readability depends on the hierarchical structures. In Figure 6, web pages in the upper part of the figure can not be divided into clusters by directory information alone. To show more details, we need other hierarchical structures.

VI. Conclusions

We developed a new visualization technique for clustered bipartite graphs. Clustered bipartite graphs are graphs with both a bipartite structure and a hierarchical structure. We call the technique "hierarchical anchored maps." One of the advantages of the technique is that it represents both bipartite and hierarchical structures simultaneously. It enables step-by-step exploration of large-scale bipartite graphs in which details are explored after understanding overviews.

Our contributions are introducing the concept of clustered graphs into bipartite graphs and developing a visualization technique for such graphs. Understanding relationships between two sets of data, such as web pages and visitors, is important in various fields. We present a visualization technique that effectively use hierarchical structures. It shows that the concept of clustered graphs may increase the scalability of visualization for bipartite graphs. Additionally, this technique extends the application areas of bipartite graphs.

References

[1] L. Zheng, L. Song, and P. Eades, "Crossing minimization problems of drawing bipartite graphs in two clusters," in *Proc. of APVIS 2005*, 2005, pp. 33–38.

[2] E. Giacomo, L. Grilli, and G. Liotta, "Drawing bipartite graphs on two curves," in *Graph Drawing 2006, LNCS 4372*, 2007, pp. 380–385.

[3] M. Newton, O. Sykora, and I. Vrt'o, "Two new heuristics for two-sided bipartite graph drawing," in *Graph Drawing 2002, LNCS 2528*, 2002, pp. 312–319.

[4] P. Eades and Q. Feng, "Multilevel visualization of clustered graphs," in *Graph Drawing 1996, LNCS 1190*, 1997, pp. 101–112.

[5] J. Ho and S. Hong, "Drawing clustered graphs in three dimensions," in *Graph Drawing 2005, LNCS 3843*, 2006, pp. 492–502.

[6] J. Abello, F. Ham, and N. Krishnan, "Ask-graphview: A large scale graph visualization system," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 669–676, 2006.

[7] P. Eades, "A heuristic for graph drawing," *Congressus Numeranitium*, vol. 42, pp. 149–160, 1984.

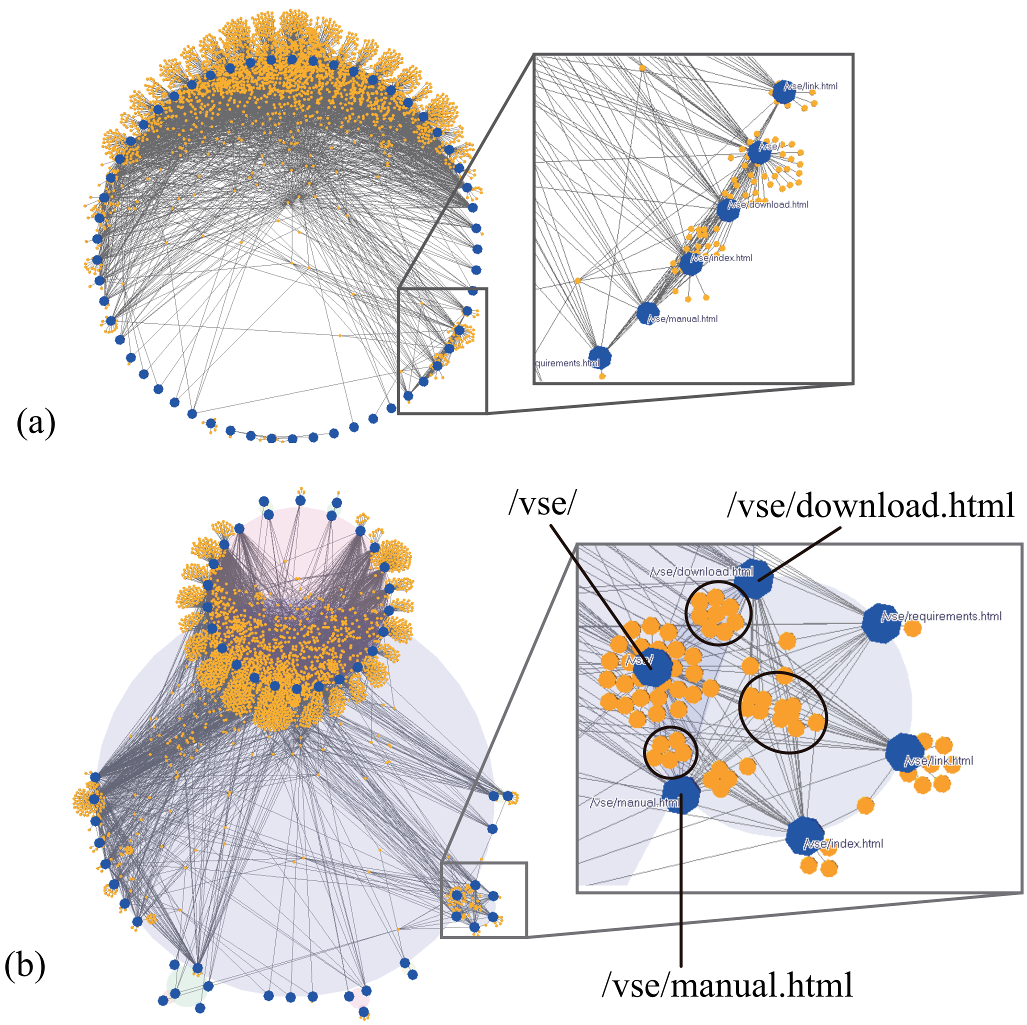


Figure 6. The relationships between web pages and visitors are drawn (a) as an anchored map and (b) as a hierarchical anchored map. There are 57 web pages, 14 directions, 2465 visitors, and 6993 accesses. Web pages are drawn as anchors and visitors are drawn as free nodes. Edges mean accesses. (b) Web pages divided into directories.

[8] K. Misue, "Anchored map: Graph drawing technique to support network mining," *IEICE Trans. Inf. & Syst.*, vol. E91-D, no. 11, pp. 2599–2606, 2008.

[9] K. Thiel, F. Dill, T. Kotter, and M. Berthold, "Towards visual exploration of topic shifts," in *IEEE International Conference on Systems, Man and Cybernetics 2007*, 2007, pp. 522–527.

[10] J. O'Donovan, B. Smyth, B. Gretarsson, S. Bostandjiev, and T. Hollerer, "Peerchooser: visual interactive recommendation," in *Proc. of CHI '08*, 2008, pp. 1085–1088.

[11] A. Naud, S.Usui, N. Ueda, and T. Taniguchi, "Visualization of documents and concepts in neuroinformatics with the 3d-se viewer," *Frontiers in neuroinformatics*, vol. 1, no. 1, Nov 2007.

[12] T. Ito, K. Misue, and J. Tanaka, "Sphere anchored map: A visualization technique for bipartite graphs in 3d," in *Proc. of HCI International 2009, LNCS 5618*, 2009, pp. 811–820.

[13] K. Sugiyama and K. Misue, "Visualization of structural information: Automatic drawing of compound digraphs," *IEEE Trans. SMC*, vol. 21, no. 4, pp. 876–892, 1991.

[14] H. Omote and K. Sugiyama, "Method for drawing intersecting clustered graphs and its application to web ontology language," in *Proc. of APVIS 2006*, 2006, pp. 89–92.

[15] T. Itoh, C. Muelder, K. Ma, and J. Sese, "A hybrid space-filling and force-directed layout method for visualizing multiple-category graphs," in *Proc. of PacificVis 2009*, 2009, pp. 121–128.

[16] I. Herman, G. Melancon, and M. S. Marshall, "Graph visualization and navigation in information visualization: A survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 1, pp. 24–43, 2000.