An Improved Graph Drawing Algorithm for Email Networks

Xiaoqiang Wang, Xiumei Zhou, Weiyao Lan and Shunxiang Wu

***Abstract*— This paper addresses the graph drawing problem for email networks. First, the graph drawing problem is formulated as a minimization problem. Then, a genetic algorithm based graph drawing method is proposed by solving the minimization problem. By taking into account the small-world property of the email-networks, the proposed method improved the force-directed graph drawing algorithm to draw the graph more intuitively and aesthetically. It also speeds up the graph drawing algorithm by ignoring repulsive force far away from the vertex, and prevents the nearly disconnected graph from being pushed to the border. Moreover, the key cliques of the email networks are placed in the central of the layout. Experiment tests show that the proposed method has better performance in satisfying the aesthetic criterions and time consumption.**

***Keywords*—Email-networks, small-world property, graph drawing.**

I. Introduction

Email has become the predominant means of communication in the information society. Pervading business, social and technical exchanges, email has been considered not only as an indicator of collaboration and knowledge exchange, but also as a tantalizing medium for research, since it provides plentiful data on personal communication in an electronic form [16]. Given its ubiquity, it is promising resource for tapping into the dynamics of information within organizations, and for extracting the hidden patterns of collaboration and leadership that are at the heart of communities in practice [16]. For example, analysis of email networks can be used to identify the informal communication structure within an organization or to discover the shared interests between people [6][15]. It can also be used to investigate the spread of computer viruses [14][5]. Thus, the modeling and analysis of massive, transient data streams raise new and challenging research problems. There are several analytical methods for the analysis of interaction data. Algorithms in the data stream and related models have been shown to be effective for statistical analysis and mining trends in large-scale data sets [13][16]. Alternatively, a graph or a network representation is a convenient and intuitive abstraction for analyzing data. Unique entities are represented as vertices, and the interactions between them are depicted as edges. The attributes of vertices and edges can be further typed, classified, or assigned based on relational information. Analyzing topological characteristics of the network, such as the vertex distribution degree, centrality and community structure, provides valuable insight into the structure and function of the interacting data entities. Common queries on these massive data sets also can be naturally encoded as variants of problems related to graph connection, flow, or partition. Therefore the problem of automatic network-diagram layout has received much attention[10]. Previous research on network-diagram layout focused on the problem of aesthetically optimal layout using the criteria such as the number of link crossings, the sum of all link lengths, and total diagram area. In this paper, we will present an improved force-directed graph placement algorithm by considering the characteristics of email network.

Visualization and analysis of email networks has been investigated by Fu *et. al.* [5][7][3]. In [5], by analyzing the data collected from the email server of National ICT Australia, Fu *et. al.* show the small-world characteristics of email networks, and point out that the email network is an "ultra-small-world" network with a small diameter and short graph distance between any pair of nodes. The clustering coefficient is low, which means that the network is relatively highly clustered [1][11]. The diameter of the network is small, usually 4-5, and the average path length reduces to 2-3. In this paper, we develop an improved force-directed graph drawing algorithm by inducting the above properties of email networks to the process and place the key cliques in the central of the layout. The proposed algorithm is speeded up by avoiding the calculation of the repulsive forces when vertices apart from each other at certain distance. Moreover, by using the grid square method, the problem of even placement of disconnected graphs within certain area is resolved. The experiment results show that the improved graph drawing algorithm has good performance in aesthetical satisfaction and time consumption.

The rest of this paper is organized as follows. Section II introduces the aesthetical criterions for graph drawing, and converts the graph drawing problem into a minimization problem by representing the aesthetical criterions as an evaluating function. The main results are presented in Section III, in which an improved graph drawing algorithm is developed by utilizing the small world property of the email networks. Section IV compares the performance in aesthetical satisfaction and time consumption between the proposed algorithm and the force-directed graph drawing algorithm. Finally, we conclude this paper with some remarks in Section V.

This work was partially supported by the National Nature Science Foundation of China (No.60704042), and the Natural Science Foundation of Fujian Province of China (No. 2008J0033).

The authors are all with the Department of Automation, Xiamen University, Xiamen, Fujian, P. R. China 361005. Corresponding to W. Lan, email: wylan@xmu.edu.cn

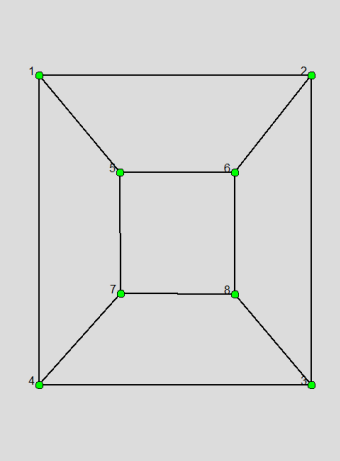


Fig. 1. Graph drawing without crossing edges.

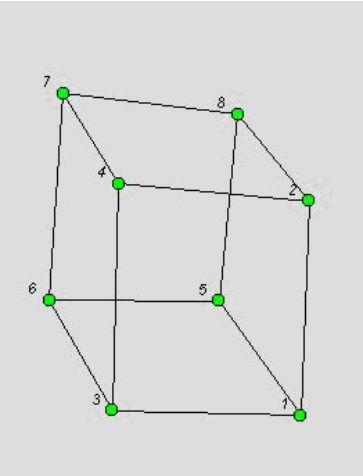


Fig. 2. Graph drawing with crossing edges.

II. Problem Formulation

The interacting data set of an email network can be represented as a graph abstraction $G(V, E)$. V is the set of vertices representing unique interacting entities, e.g., the senders and the receivers. And, E is the set of edges representing the interactions, e.g., the communication between the senders and the receivers. The number of vertices and edges are denoted by n and m respectively. We assume that each edge $e \in E$ has a positive integer weight $w(e)$, which denotes the amount of the communication. A path from vertex s to t can be defined as a sequence of edges $[u_i, u_{i+1}]$, $i = 0, 1, 2, \cdots, l-1$, where $u_0 = s$ and $u_l = t$. The length of a path is the sum of weights of the edges in the path. The distance between vertices s and t, which is denoted as $d(s, t)$, is the minimum length of any paths connecting s and t in G. The total number of shortest paths between vertices s and t is denoted as σ_{st}, and the number passing through vertex v is denoted as $\sigma_{st}(v)$.

Given a graph $G(V, E)$, graph drawing is to map each vertex $v \in V$ of the graph into a point $P(v)$ in a plane, and to map each edge $(u, v) \in E$ of the graph into a straight line with vertices $P(u)$ and $P(v)$. Graph drawing algorithm can be used to produce aesthetically-pleasing, two-dimensional pictures of graphs within a certain area, e.g., a screen. Graph drawing of an email network can be simplified to designate a pair of coordinates $(x(v), y(v))$ for every vertex, and then draw a directed line from the sender vertex to the receiver vertex. In the process of coordinates designation, the email network can be regarded as an undirected graphs by neglecting the direction of the communications. For drawing undirected graphs, the following aesthetic criterias are generally accepted [4][8].

1. Distribute the vertices evenly in the frame.
2. Minimize edge crossings.
3. Make edge lengths uniform.
4. Reflect inherent symmetry.

However, Criteria 2 is not always necessary. For example, Figure 1 shows a graph drawn without crossing edges, while Figure 2 shows the same graph drawn with crossing edges. It is clear that Figure 2 is better than Figure 1 in finding the inside topology of the graph. Thus, Criteria 2 is not considered in our algorithm. Moreover, to develop the graph drawing algorithm, the aesthetical criterias are simplified as follows:

C1 Each vertex should keep distant from others.
C2 Two vertices should get closer if there are links between them.

Under above simplified criterias, the graph drawing problem can be formulated as a minimization problem:

$$\min f(G(V, E)) \tag{1}$$

with

$$f(G) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{L^2}{|p_i - p_j|} + \sum_{(v_i, v_j) \in E} \frac{|p_i - p_j|^2}{L} \tag{2}$$

where p_i is the position of vertex $v_i \in V$, $|p_i - p_j|$ is the distance between vertices v_i and v_j,

$$L = k\sqrt{\frac{s}{n}} \tag{3}$$

is the optimal distance between vertices v_i and v_j. n is the number of vertices, s is the area of screen, k is a tuning parameter which is found experimentally. From (2), we can see that, breaking criteria C1 will increase the first part of $f(G)$, and breaking criteria C2 will increase the second part of $f(G)$. Thus, the graph can be drawn in an aesthetically-pleasing way under criteria C1 and C2 by minimizing $f(G)$.

III. An Improved Graph Drawing Algorithm

The graph drawing problem is converted into a minimization problem as described by (1). In recent years, genetic algorithm has been developed rapidly as a random search algorithm, which plays an important role in function optimization. The genetic algorithm based graph drawing method has been investigated by [2][12] et. al. In this section, we will develop an improved graph drawing algorithm for

email networks by utilizing the small-world property of the email networks.

A. Genetic Algorithm Based Graph Drawing

Denoting x_i, y_i as the x, y coordinates of $p(v_i)$ respectively, where $p(v_i)$ is the projection of vertex $v_i \in V$ in the screen. To draw a graph $G(V, E)$, all the vertexes are projected to a defined area:

$$S = \{(x, y) \mid a \le x \le b, \ \ c \le y \le d\} \tag{4}$$

Thus, the minimization problem (1) can be rewritten as

$$\min_{(x,y)\in S} f(x, y) \tag{5}$$

with

$$f(x,y) = \sum_{i=1}^{n}\sum_{j=i+1}^{n} \frac{L^2}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2)}} + \sum_{(v_i,v_j)\in E} \frac{(x_i - x_j)^2 + (y_i - y_j)^2)}{L} \tag{6}$$

The process to solve the minimization problem (5) using genetic algorithm can be briefly described as follows. In order to place the cliques in the central of the layout, we will first compute the degree of each vertex and fix the vertex with highest degree in the center of the layout. If the graph is disconnected, we put all the cliques around a circle in the middle of the layout. Denoting all the vertices which have the highest degree in their own cliques as v_{r+1}, v_{r+2}, $\cdots$,v_n, we will not move the position of these vertices. Assign the following coordinates to these vertices:

$$\begin{aligned} (x_n, y_n) &= \left(\frac{1}{2}l, \frac{1}{2}w\right) \\ x_{r+i} &= \frac{1}{2}l + L\cos\left(\theta_{r+i-1} + \frac{n_{r+i}\pi}{n}\right) \\ y_{r+k} &= \frac{1}{2}w + L\sin\left(\theta_{r+i-1} + \frac{n_{r+i}\pi}{n}\right) \end{aligned}$$

for $i = 1, 2, \cdots, n - r$. L is the optimal distance between vertices which is defined by (3), it can be adjusted by tuning the parameter k to arrange the layout more aesthetically-pleasing. n_{r+i} is the count of vertices of $(r + k)$th cliques.

$$\theta_{r+i} = \theta_{r+i-1} + \frac{2n_{r+i}\pi}{n}, \ \text{ with } \ \theta_{r+1} = 0$$

Then, encode the candidate solution of (5) $((x_1, y_1), (x_2, y_2), \cdots, (x_r, y_r))$ as chromosome with a simple data structure. Then, the fitness of every individual in the population is evaluated, the individuals having lower fitness are modified (recombined and possibly randomly mutated) to form a new candidate solution. The new solution is then used in the next iteration of the algorithm. Finally, the algorithm terminates when a minimum solution of (5) (or an approximate solution) has been produced. Specifically, we need to consider the following issues.

1. Coding. In order to avoid the process of decoding and to reduce the time consumption, we set the string $(x_1, y_1, x_2, y_2, \cdots, x_r, y_r)$ as genetic representation, instead of setting every candidate solution (x_1, y_1), (x_2, y_2), $\cdots$, (x_r, y_r) as the chromosome.

2. Fitness function. In order to minimize the objective function, we can find an arbitrary positive number

$$C_{\max} \ge f(x_1, y_1, \cdots, x_r, y_r)$$

and set the fitness function as

$$F(x_1, y_1, \cdots, x_r, y_r) = C_{\max} - f(x_1, y1, \cdots, x_r, y_r)$$

3. Selecting strategy. In order to prevent precocious [2], we use Sigma ratio of transformation technology to transform individual's fitness $f(i)$ of individual i to $ExpVal(i)$ which is defined as

$$ExpVal(i) = \begin{cases} 1 + (f(i) - f(t)/2\sigma(t)), & \text{if } \ \sigma(t) > 0 \\ 1, & \text{if } \ \sigma(t) < 0 \end{cases}$$

where $f(t)$ and $\sigma(t)$ respectively represent fitness and standard variance of tth generation group. Then for $ExpVal(i)$,we adopt the choosing strategy based on adaptation value proportion but retain the chromosome with the maximum fitness.

4. Assign parameters. We set P_c as probability of hybridization, P_m as mutation probability. These two parameters influence the time consumption of the algorithm and the possibility of termination. We should set different value according to different graphs.

5. Design of genetic operator. For hybrid operator, two new individual

$$A' = (a_1, a_2, a_3, b_4, \cdots, b_{2r})$$

and

$$B' = (b_1, b_2, b_3, a_4, \cdots, a_{2r})$$

can be obtained by selecting two parent bodies

$$A = (a_1, a_2, \cdots, a_{2r})$$

and

$$B = (b_1, b_2, \cdots, b_{2r})$$

and hybrid position 3 by using the way of simple point hybrid. For Variation operator ,we adopt non-uniform mutation as follows: Assume the parent body

$$A = (a_1, a_2, \cdots, a_k, \cdots, a_{2r})$$

of which the kth component is selected for variation. Assume the kth component is the y coordinates, which ranges from $[c, d]$. Then, a new individual can be obtained after the variation,

$$A' = (a_1, a_2, \cdots, a'_k, \cdots, a_{2r})$$

where

$$a'_k = \begin{cases} a_k + \Delta(t, d - a_k), & \text{if } \ Random(2) = 0 \\ a_k + \Delta(t, a_k - c), & \text{if } \ Random(2) = 1 \end{cases}$$

Here, $Random(2)$ generates a random integer 0 or 1, while

$$\Delta(t, y) = y(1 - r(1 - t/T)5)$$

where t and T are respectively the current and largest evolved algebra. Obviously, $\Delta(t, y)$ ranges from zero to y and tends

to zero as t tends to T. This shows that mutation search in a larger scope at the early stage of evolution, but in the latter part it plays a role of local fine-tuning.
6. Termination criterion. Algorithm terminates after running for several generation .

Thus, the drawing algorithm framework of the undirected graph based on genetic algorithm is described as follows:

```
Begin:
t: =0, initialize(p(t)); Evaluate(P(t));
while: t<T Do
    begin
    P1: = Select(P(t)); P2: = Crossover(P1);
    P(t+1): =Mutate(P2); Evaluate(P(t+1));
    t: t + 1;
    end
Draw_Graph(G, X, Y);
End .
```

In the algorithm, the vertexes with coordinates $X = (x_1, x_2, \cdots, x_n)$ and $Y = (y_1, y_2, \cdots, y_n)$ of graph $G(X, Y)$ are drawn during the $DrawGraph(GXY)$ procedure.

B. Speeding up the Algorithm

As indicated by Kamada and Kawai [9], the connected components fly apart and flat themselves against the walls, as shown in Figure 3. One solution suggested by [9] is to partition the graph into some components and give each component a region of area proportional to its size [9], with each component laid out independently. Without finding the close connected components, we achieve this 'regional' effect by using the grid-square method, as shown in Figure 4. We will benefit when drawing nearly disconnected graphs such as the twin copies of pentacle connected by a single strand. In this method, the repulsive forces are computed only between it and the vertices within certain distances, and attractive forces will be computed as usual. This is nearly equivalent to compute $f(G)$ as

$$f(G) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} u \frac{L^2}{|p_i - p_j|} + \sum_{(v_i, v_j) \in E} \frac{|p_i - p_j|^2}{L} \qquad (7)$$

with

$$u = \begin{cases} 1, & |p_i - p_j| < d \\ 0, & |p_i - p_j| \geq d \end{cases}$$

where $d = \frac{1}{2}\sqrt{s/n}$. This method will also speed up the iteration because we need not calculate the repulsive forces when vertices apart from each other at certain distance.

IV. Experiments and Performance Evaluation

We evaluate the performance of the proposed algorithm according to the following two factors: time consumption and degree of aesthetical satisfaction.

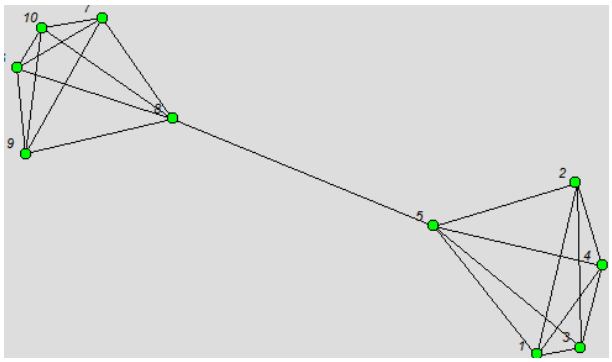


Fig. 3. The connected components fly apart.

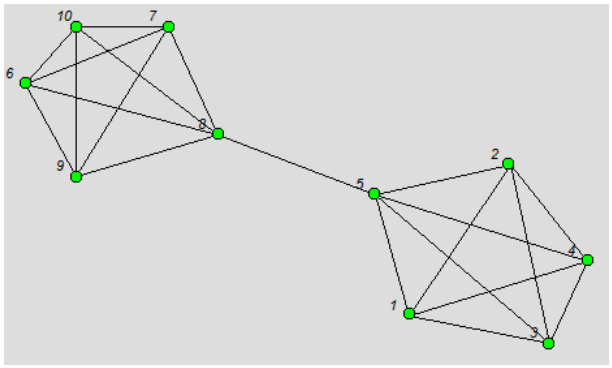


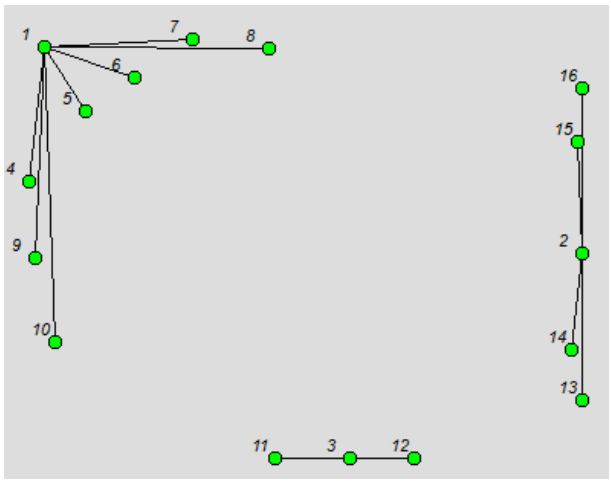
Fig. 4. The rearranged connected components.

A. Degree of Aesthetical Satisfaction

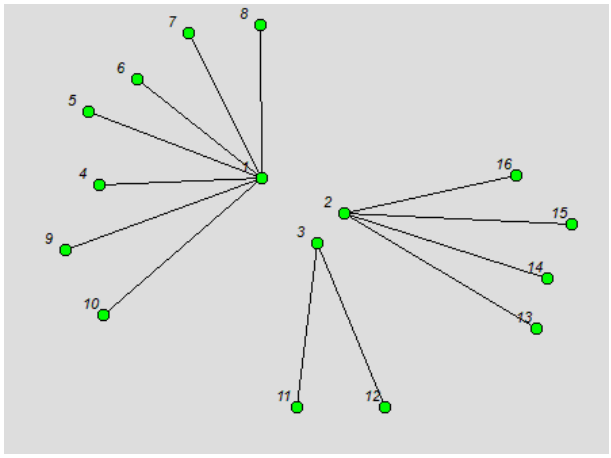
Based on the commonly aesthetic criteria, we will focus on whether the vertices have been distributed evenly in the frame, which will helpful for reflecting inherent symmetry of the network. The experiment result are shown in Figure 5. Figure 5.(a) is the layout produced by the force-directed algorithm, all the vertices have been repelled to the frame, since there are only repulsive force between them to repel every cliques far from each other and no enough attractive forces to prevent the forward phenomenon. Figure 5.(b) is the layout made by our improved algorithm. The vertices around the center cycle are the key cliques. Thus, we can easily find the key cliques from the graph and this is one of the major purpose of social network analysis.

B. Time Consumption

We evaluate the algorithm's time consumption performance by the count of iterations rather than time cost exactly in the calculating process. Two controlled trails are conducted in the experiment. The first one is a triangulated triangle as shown in Figure 6.(a), the other one is a three-dimensional layout of a mesh as shown in Figure 6.(b). Table I is the experiment result which compares the performance on time consumption between the force directed algorithm and the improved algorithm. For both cases, the improved algorithm need less iteration in the experiments.



(a) Graph drawing with force directed algorithm.

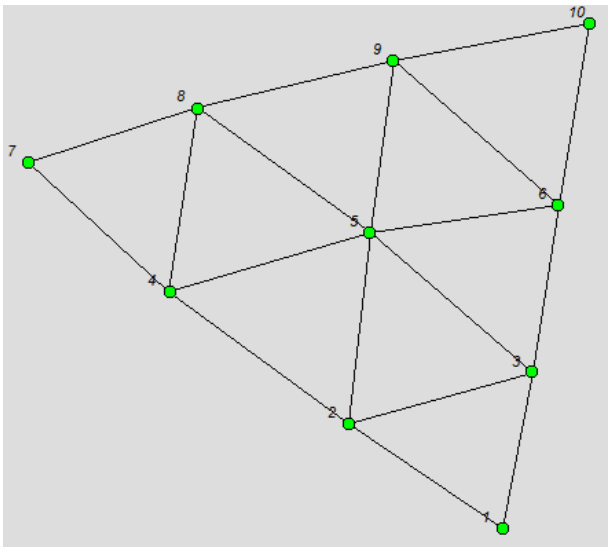


(b) Graph drawing with the improved algorithm.

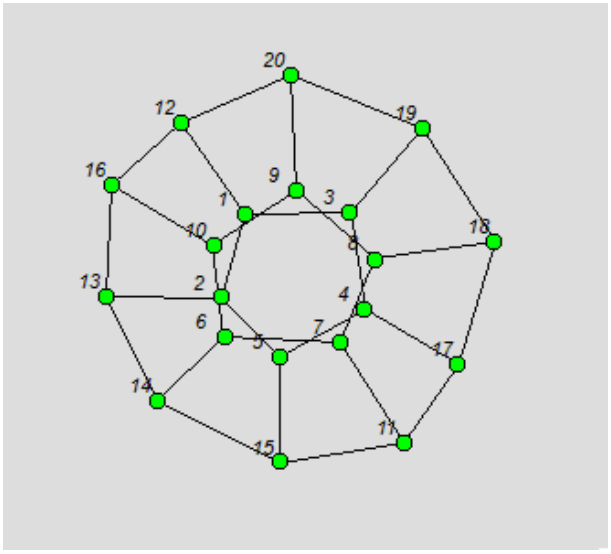
Fig. 5. Aesthetical satisfaction of graph drawing algorithm.

TABLE I
EXPERIMENT RESULT.

controlled Trials	Triangle	Mesh
Vertices	10	20
Edges	18	30
Iterations of Old Algorithm	57	85
Iterations of Improved Algorithm	54	78
Fitness	1.0	1.0



(a) Triangulated triangle.



(b) 3D layout of a mesh.

Fig. 6. The graph for time consumption test.

V. CONCLUSIONS

The growing popularity of computer network based social networks and the ability to collect gigabytes of unbiased social information provides a unique opportunity for computer scientists to develop new computational techniques for mining social network patterns. The contribution of the paper includes (i) Graph drawing algorithm is improved by using graph property to guide the graph placement. (ii) Efficiency of calculating is improved by using genetic algorithm to maximize the fitness function of the total placement. (iii) Indicate a method to hold the disconnected graph together; (iv) Speed up the calculating process by ignoring the repulsive forces between two vertices far away from each other. The proposed method is a handy computational tool for email network analyze and will enable advancing understanding of such social networks. Our work can be further improved by incorporating semantic information about the contents of email. Also, it is possible to find other methods to represent the graph more intuitively and more easily for researcher to find the inner information.

REFERENCES

[1] U. Brandes and T. Erlebach. Network Analysis: Methodological Foundations. In: *Lecture Notes in Computer Science*, vol. 3418, Springer, 2005

[2] N. Chaiyaratana, and A. M. S. Zalzala. Recent developments in evolutionary and genetic algorithms: theory and applications. In: *Genetic Algorithms in Engineering Systems: Innovations and Applications*, pp. 270-277, 1997.

[3] J. Ellson, E. R. Gansner, L. Koutsofios, S. C. North, and G. Woodhull. Graphviz and dynagraph - static and dynamicgraph drawing tools. *Graph Drawing Software*, 2003.

[4] T. Fruchterman and E. Reingold. Graph drawing by force-directed placement. *Soft-ware Practice and Experience*, vol. 21, no. 11, pp.1129–1164, 1991.

[5] X. Y. Fu, S. H. Hong, and N. S. Nikolov. Visualization and Analysis of Email Networks, pp.1–8, 2007. DOI: 10.1109/APVIS.2007.329302.

[6] R. Guimer, L. Danon, A. Daz-Guilera, and F. G. Y. A. Arenas. The real communication network behind the formal chart: Community structure in organizations. In: *The 7th Granada Seminar on Computational and Statistical Physics*, Granada, Spain, 2002.

[7] S. Girdzijauskas, A. Datta, and K. Aberer. On small world graphs in non-uniformly distributed key spaces. In: *Proceedings of the 21st International Conference on Data Engineering*, 2005.

[8] J. W. Huang, L. S. Kang, and Y. P. Chen. A new graph drawing algorithm for undirected graphs, *Software Journal*, vol. 11, no. 1, pp.138-142, 2000.

[9] T. Kamada, and S. Kawai. An algorithm for drawing general undirected graph. *Information Letters*, vol. 31, no. 1, pp.7–15, 1989.

[10] C. Kosak, J. Marks, and S. Shieber. Automating the layout of network diagrams with specified visual organization. *IEEE Transaction on System , Man and Cybernetics*, vol. 24, no. 3, pp. 440-454, 1994.

[11] S. Manfredi, M. di Bernardo, and F. Garofalo. Small-world effects in networks: an engineering interpretation. In: *Proceedings of the 2004 International Symposium on Circuits and Systems* vol.4, pp. 820–823, May 2004.

[12] Q. C. Meng, T. J. Feng, Z. Chen,C. J. Zhou, and J. H. Bo. Genetic algorithms encoding study and a sufficient convergence condition of GAs. , 1999. In: *The Proceedings of the 1999 IEEE International Conference on Systems, Man, and Cybernetics*, vol.1, pp. 649-652, Oct. 1999. DOI:10.1109/ICSMC.1999.814168.

[13] M. Newman. The structure and function of complex networks. *SIAM Review*, vol.45, no.2, pp.167–256, 2003.

[14] M. E. J. Newman, S. Forrest, and J. Balthrop. Email networks and the spread of computer viruses. *Physical Review*, vol.66, pp.1–4, 2002.

[15] M. F. Schwartz and D. C. M. Wood. Discovering shared interests using graph analysis. *Communications of the ACM*, vol.36, pp.78–89, 1993.

[16] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman. Email as spectroscopy: automated discovery of community structure within organizations, communities and technologies. In: *Proceedings of the First International Conference on Communities and Technologies*, pp.81–96, 2003.