On the Computational Complexity of Edge Concentration

Xuemin Lin
School of Computer Science and Engineering
The University of New South Wales
Sydney, NSW 2052, Australia
email: lxue@cse.unsw.edu.au

June 30, 1999

Abstract

Suppose that $G = (U, L, E)$ is a bipartite graph with vertex set $U \cup L$ and edge set $E \subseteq U \times L$. A typical convention for drawing G is to put the vertices of U on a horizontal line and the vertices of L on another horizontal line, and then to represent edges by line segments between the vertices that determine them. "Edge concentration" is known as an effective method to draw dense bipartite graphs clearly. The key in the edge concentration method is to reduce the number of edges, while the graph structural information is retained. The problem of having a maximal reduction on the number of edges by the edge concentration method was left open. In this paper we show that this problem is NP-hard.

Keywords: Graph drawing, Bipartite graph, Edge cover, NP-complete.

1 Introduction

Graphs are commonly used in computer science to model relation structures such as programs, databases, and data structures. A good graph drawing gives a clear understanding of a structural model; a bad drawing is simply misleading. From last decade, a number of aesthetics for modelling goodness of graph drawings have been proposed from various applications, as well as many graph drawing algorithms [2, 3, 4, 6, 9, 10, 11] to achieve these aesthetics.

However, the applicability of most graph drawing algorithms was challenged in case a graph is *dense*, that is, a graph with a large number of edges. This is because a graph with too many edges physically cannot be drawn in a clear way within human vision. To resolve this, a technique of "edge concentration" has been proposed [2, 6, 7, 8, 11].

The edge concentration method was shown [7, 8, 11] to be very effective for drawing dense graphs clearly. In this paper, we shall investigate an open optimization problem in this method. Our investigation will be restricted to bipartite graphs. Note that a bipartite graph (U, L, E) is *complete* if $E = U \times L$. *Edge concentration* in a bipartite graph G consists of two steps [7, 8, 11]:

Step 1: Find a set $C = \{c_i : 1 \leq i \leq m\}$ of complete bipartite subgraphs of G, such that each edge is contained in at least one c_i. We call C a *bipartite cover* of G.

Step 2: Replace each c_i in C by a "star like" graph h_i (defined in Section 2). Then, replace G by $\cup_{i=1}^{m} h_i$.

For instance, the graph in Figure 1(a) can be transformed respectively into Figure 1(b) and 1(c) by applying the method of edge concentration with respect to two different bipartite covers.

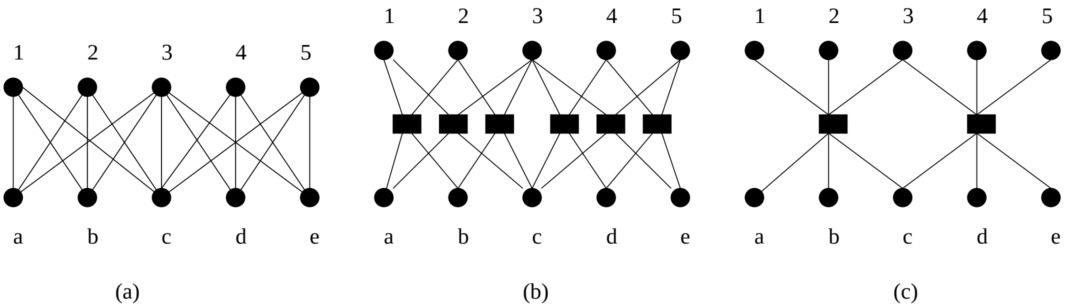

Figure 1: Edge concentration

The graph in Figure 1(c) is much simpler than that in Figure 1(b), because of fewer edges. Thus the following optimization problem has arisen while applying the edge concentration method:

> *Find a bipartite cover such that the number of edges in the resulting graph is minimum.*

We denote this problem by OPEC that stands for the Optimization Problem in Edge Concentration.

OPEC is important to effectively applying the method of edge concentration. It was first investigated in [7, 8, 11], but only heuristics were proposed. It has been conjectured [2, 6, 7, 8] that OPEC is NP-hard. In this paper, we will provide a proof to this conjecture.

The rest of the paper is organized as follows. In section 2, we will formally describe the edge concentration method, and then formally define OPEC. Section 3 will give the proof of NP-hardness of OPEC. This is followed by the conclusions.

2 Edge Concentration

For notation simplification in the rest of the paper a complete bipartite graph (U, L, E) will be denoted only by its vertex set (U, L). The *edge concentrator* h is a one-one mapping which maps a complete bipartite graph $G = (U, L)$ into a 3-layered *hierarchical* graph [2, 6] (V_1, V_2, V_3, E) such that

- $V_1 = U$ and $V_3 = L$,
- V_2 consists of only one vertex v_G,
- $E = (V_1 \times \{v_G\}) \cup (\{v_G\} \times V_2)$.

By this, the two steps of the edge concentration method can be formally described below.

Step 1: Find a bipartite cover $C = \{c_i = (U_i, L_i) : 1 \leq i \leq k\}$ of G.

Step 2: Replace G by $H_C(G)$, where $H_C(G) = \cup_{i=1}^{k} h(c_i)$. .

Here, each c_i is called a *component* of the cover C; and $H_C(G)$ is called the *concentrated image* of G with respect to C.

Figures 1(a), 1(b) and 1(c) give an example. The original graph is depicted in Figure 1(a). Using the edge concentration method, the graph (concentrated image) in Figure 1(c) results from the bipartite cover $\{(\{1, 2, 3\}, \{a, b, c\}), (\{3, 4, 5\}, \{c, d, e\})\}$. The bipartite cover leading to the concentrated image in Figure 1(b) consists of six components:

$(\{1, 2\}, \{a, b\})$, $(\{1, 3\}, \{a, c\})$, $(\{2, 3\}, \{b, c\})$, $(\{3, 4\}, \{c, d\})$, $(\{3, 5\}, \{c, e\})$, and $(\{4, 5\}, \{d, e\})$.

Let $C = \{(U_i, L_i) : 1 \leq i \leq n\}$ is a set of complete bipartite graphs. $N(C)$ denotes the total sizes of the vertex sets in C, that is, $N(C) = \sum_{i=1}^{n}(|U_i| + |L_i|)$. For a complete bipartite graph $G = (U, L)$, the number of edges in $h(G)$ is $|U| + |L|$. This immediately implies:

Fact 1 *Given a bipartite cover C of G, the number of the edges in the concentrated image $H_C(G)$ is $N(C)$.* $\square$

By Fact 1, OPEC can be formally described as follows.

<u>OPEC:</u>

INSTANCE: A bipartite graph G.
QUESTION: Find a bipartite cover C of G such that $N(C)$ is minimized.

Before proving the NP-hardness of OPEC, we would like to distinguish the problem OPEC from the Cardinality Minimization problem [5] (CM), a well known NP-hard problem. The CM problem is to find a bipartite cover C such that the cardinality (i.e, the number of the components) of C is minimized.

At the first glance the CM problem seems similar to OPEC. However, these two problems are inherently different. For instance, with respect to the graph in Figure 2(a), there is no common solution to CM and OPEC. It can be immediately verified that the bipartite cover A below is the solution to CM but not the solution to OPEC , while the cover B is a solution to OPEC but not to CM.

A. $\{(\{a, b\}, \{1, 2, 3, 4, 5, 6\}),\ (\{b, c\}, \{3, 4, 5, 6, 7, 8\})\}$.

B. $\{(\{a, b\}, \{1, 2\}),\ (\{a, b, c\}, \{3, 4, 5, 6\}),\ (\{b, c\}, \{7, 8\})\}$.

The concentrated images with respect to A and B are respectively shown in Figures 2(b) and 2(c).

Furthermore, it does not seem easy to construct an explicit transformation from CM to a special case of OPEC. In fact, our proof is not based on a transformation from CM to OPEC.

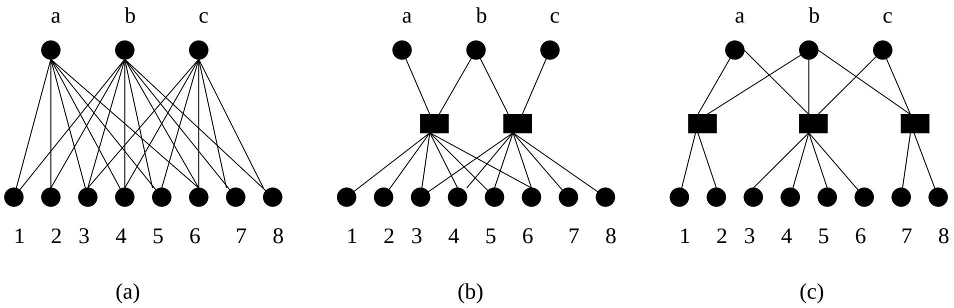

Figure 2: Difference between CM and OPEC

3 NP-hardness of OPEC

In this section, we shall prove the following Theorem.

Theorem 1 *OPEC is NP-hard.*

To prove Theorem 1, we will show the NP-completeness of the corresponding decision problem of OPEC, DOPEC, described as follows.

<u>DOPEC:</u>

INSTANCE: A bipartite graph G and an integer K.
QUESTION: Is there a bipartite cover C of G such that $N(C) \leq K$?

To prove the NP-completeness of DOPEC, we transform a well-known NP-complete problem, *3-Dimensional Matching (3DM)* problem [5], to a special case of DOPEC.

3DM:

INSTANCE: A set $M \subseteq W \times X \times Y$ where W, X, and Y are disjoint sets; and $|W| = |X| = |Y| = q$.
QUESTION: Does M contain a *matching* M', that is, $M' \subseteq M$ such that $|M'| = q$ and no two elements of M' agree in any coordinate?

An element $\zeta \in W \cup X \cup Y$ is *covered* by an element $(w, x, y) \in M \subseteq W \times X \times Y$ if either $\zeta = w$ or $\zeta = x$ or $\zeta = y$. A set $M'' \subseteq W \times X \times Y$ *covers* $W \cup X \cup Y$ if each element in $W \cup X \cup Y$ is covered by at least one element in M''. The following properties can be immediately verified.

Lemma 1 *Suppose that W, X, and Y are disjoint sets and $|W| = |X| = |Y| = q$. Then M'' is a matching if and only if M'' is the smallest subset of $W \times X \times Y$ that covers $W \cup X \cup Y$.*

Lemma 1 is a key for us to prove the NP-completeness of DOPEC by transforming 3DM to DOPEC. Note that an instance I_{3DM} of 3DM consists of

- an integer q and an integer n,
- three disjoint sets W, X, and Y such that $|W| = |X| = |Y| = q$, and
- a subset of $W \times X \times Y$: $M = \{(w_i, x_i, y_i) : 1 \leq i \leq n, w_i \in W, x_i \in X, y_i \in Y\}$.

Clearly, $n \leq q^3$. Without loss of generality, we assume that $n > q$.

Below, we construct an instance $T(I_{3DM})$ of DOPEC from a given instance I_{3DM} of 3DM. $T(I_{3DM})$ consists of an integer $K = n(3(n+1)^2 + 3(n+1)) + q$, and a bipartite graph $G = (U, L, E)$ that is constructed from I_{3DM} as follows.

- The vertex set U consists of $3q(n+1)$ vertices. U is divided into $3q$ equal sized disjoint subsets each of which is denoted by $u(\zeta)$ to correspond an element $\zeta \in W \cup X \cup Y$.
- The vertex set L consists of $3n(n+1)^2 + 1$ vertices. L is divided into $n+1$ disjoint subsets L_i for $0 \leq i \leq n$. For $1 \leq i \leq n$, L_i is used to construct a complete bipartite subgraph corresponding the element $(x_i, y_i, w_i) \in M$, and consists of $3(n+1)^2$ vertices. L_0 consists of only one vertex v_0, called the *full vertex* of G.
- The edge set of G is defined as follows to represent the elements in M.
  - Corresponding to each element (w_i, x_i, y_i) $(1 \leq i \leq n)$ in M, there is a complete bipartite subgraph $G_i = (U_i, L_i)$ of G, where $U_i = u(w_i) \cup u(x_i) \cup u(y_i)$. G_i is called an *eigen-subgraph* of G. Note that for $1 \leq i < j \leq n$, $L_i \cap L_j = \emptyset$ and $|U_i \cap U_j| = (n+1)|\{w_i, x_i, y_i\} \cap \{w_j, x_j, y_j\}|$.
  - For each vertex $v \in U$, $(v, v_0) \in E$.

The edges in all G_i and the edges from v_0 to U constitute the edge set E. □

For instance, let $q = 2$, $n = 3$, and $M = \{(w_1, x_1, y_1), (w_2, x_2, y_2), (w_3, x_3, y_3)\}$, where $w_3 = w_1$, $x_3 = x_2$ and $y_3 = y_2$. The corresponding bipartite graph after applying the above transformation is illustrated in Figure 3 using the corresponding concentrated image, where $|u(w_i)| = |u(x_i)| = |u(y_i)| = 4$ for $i = 1, 2$, and $|L_i| = 48$ for $i = 1, 2, 3$. Two eigen-subgraphs G_1 and G_2 are respectively depicted by closed curves, and the third eigen-subgraph G_3 is depicted by a thick-lined polygon.

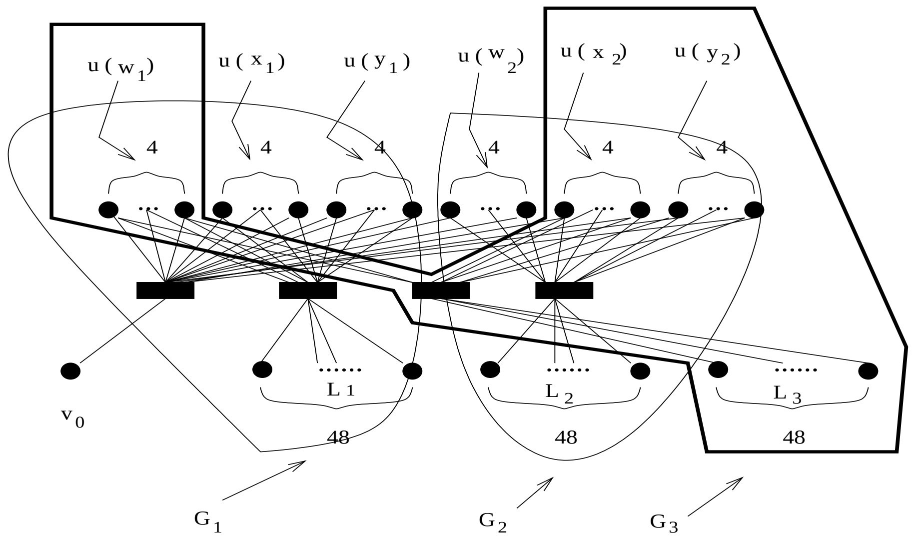

Figure 3: The transformation from 3DM to DOPEC

Based on the above transformation from 3DM to DOPEC, the following lemma is immediate.

Lemma 2 *Suppose that W, X and Y are the three disjoint sets in an instance I_{3DM} of 3DM, and G is the bipartite graph in $T(I_{3DM})$. Then, any complete bipartite subgraph (U', L') of G, which contains a non-full vertex of G in L', cannot contain two vertices $v_1 \in u(\zeta)$ and $v_2 \in u(\mu)$ $(\zeta \neq \mu)$ in U' such that ζ and μ are either both in W, or both in X, or both in Y.* □

Note that to prove the NP-completeness of DOPEC, we need only to prove the two facts below.

F1: The transformation of each instance I_{3DM} of 3DM to $T(I_{3DM})$ of DOPEC takes polynomial time with respect to q.

F2: For each instance I_{3DM} of 3DM, $T(I_{3DM})$ of DOPEC has a solution if and only if I_{3DM} has a solution.

From the construction of $T(I_{3DM})$ from I_{3DM}, F1 immediately follows. The rest of this section is devoted to the proof of F2. Specifically, the "if part" is proven as the following Lemma.

Lemma 3 *If an instance I_{3DM} of $3DM$ has a solution s_{3DM}, then the instance $T(I_{3DM})$ of DOPEC has a solution.*

Proof: Let I denote the set of sub-indices used to represent a solution s_{3DM} to I_{3DM}; that is, s_{3DM} can be represented as $\{(w_j, x_j, y_j) : j \in I\}$. Suppose that $\{(U_i, L_i) : 1 \leq i \leq n\}$ is the set of eigen-subgraphs of the bipartite graph G in $T(I_{3DM})$. A set $C = \{c_l : 1 \leq l \leq n\}$ of complete bipartite subgraphs may be constructed as follows:

- $c_l = (U_l, L_l)$ if $l \notin I$, and
- $c_l = (U_l, L_l \cup \{v_0\})$ if $l \in I$ where v_0 is the full vertex of G.

One can immediately verify that C is a bipartite cover of G. Note $|I| = q$. Thus $N(C) = n(3(n+1) + 3(n+1)^2) + q$. Therefore C is a solution to the instance $T(I_{3DM})$ of DOPEC. $\square$

To prove the "only if" part of F2, we first prove that a solution (a bipartite cover) C to $T(I_{3DM})$ where $K = n(3(n+1) + 3(n+1)^2) + q$ has the following two properties:

1. $N(C) = n(3(n+1) + 3(n+1)^2) + q$, and
2. C takes the unique form as shown in the proof of Lemma 3.

Then, from C we explicitly construct a solution to the corresponding instance I_{3DM} of 3DM. The proof starts from characterizing a solution to $T(I_{3DM})$; and it consists of three steps - Lemmas 4, 5, and 6.

Lemma 4 *Let $C = \{c_i : 1 \leq i \leq m\}$ be a solution to the instance $T(I_{3DM})$ of DOPEC. Then each eigen-subgraph (U_i, L_i) of G must have a vertex $v_i^* \in L_i$ that is contained in only one component c_j of C.*

Proof: For each $c_i \in C$, we use $U(c_i)$ and $L(c_i)$ to denote the two sets of vertices in c_i; that is, $c_i = (U(c_i), L(c_i))$. If the Lemma does not hold, then there is an eigen-subgraph (U_j, L_j) of G such that each vertex in L_j is contained by at least two components of C. Therefore, $\sum_{i=1}^{m} |L(c_i)| \geq 3n(n+1)^2 + 3(n+1)^2$. Consequently, $N(C) > \sum_{i=1}^{m} |L(c_i)| > n(3(n+1)^2 + 3(n+1)) + q$. On the other hand, since C is a solution to $T(I_{3DM})$,

$$N(C) \leq K = n(3(n+1)^2 + 3(n+1)) + q. \quad (1)$$

This leads to a contradiction. $\square$

Lemma 5 *Let C be a solution to the instance $T(I_{3DM})$ of DOPEC. Each eigen-subgraph of G in $T(I_{3DM})$ must be contained by one component of C.*

Proof: By Lemma 4, every eigen-subgraph $G_i = (U_i, L_i)$ of G has a vertex $v_i^* \in L_i$ that appears in only one component, denoted by c_{G_i}, of C. Clearly, U_i must be contained by c_{G_i} because c_{G_i} has to contain every edge from v_i^* to U_i. Lemma 2 implies that $c_{G_i} \neq c_{G_j}$ if $G_i \neq G_j$.

Let Π_1 denote the set of those eigen-subgraphs which violate the Lemma, and let Π_2 denote the complementary of Π_1. We will assume that $\Pi_1 \neq \emptyset$ and derive a contradiction. Let C_{Π_1} denote $\{c_{G_i} : G_i \in \Pi_1\}$ and let C_{Π_2} denote $\{c_{G_i} : G_i \in \Pi_2\}$.

Note that $\Pi_1 \neq \emptyset$. This means that Π_1 contains at least one eigen-subgraph $G_l = (U_l, L_l)$ of G. Since G_l violates the Lemma, there must be one vertex $v^0 \in L_l$ which is not contained in c_{G_l}. Consider that:

1. $\forall G_i \in \Pi_1 \cup \Pi_2$, c_{G_i} is a complete bipartite subgraph of G,
2. $\forall G_i \in \Pi_1 \cup \Pi_2$, c_{G_i} contains U_i, and
3. some vertices in U_i for $i \neq l$ are not connected to v^0 by an edge in G.

These imply that v^0 is also not contained in any other c_{G_j} where $G_j \neq G_l$. Since v^0 and all its edges to U_l must be contained by the components of C, we have

$$\cup_{c \in C-(C_{\Pi_1} \cup C_{\Pi_2})} U(c) \supseteq U_l. \tag{2}$$

From (2), we can deduce that:

$$\begin{aligned} N(C) &= \sum_{c \in C_{\Pi_1} \cup C_{\Pi_2}} |U(c)| + \sum_{c \in C-(C_{\Pi_1} \cup C_{\Pi_2})} |U(c)| + \sum_{c \in C} |L(c)| \\ &\geq 3n(n+1) \ + \ 3(n+1) + \ 3n(n+1)^2 \\ &> n(3(n+1)^2 + 3(n+1)) + q \end{aligned}$$

Note that C is a solution to $T(I_{3DM})$. Thus (1) should hold. The contradiction is obtained. $\square$

Lemma 6 *Let $\{G_i = (U_i, L_i) : 1 \leq i \leq n\}$ be the set of eigen-subgraphs of $G = (U, L)$ in $T(I_{3DM})$, and let C be a solution to $T(I_{3DM})$. Then*

- *$N(C) = K = n(3(n+1)^2 + 3(n+1)) + q$, and*
- *3DM also has a solution to the corresponding instance I_{3DM}.*

Proof: By the proof of Lemma 5, we know that there are n components in C, denoted by c_{G_i} for $1 \leq i \leq n$, each of which contains an eigen-subgraph $G_i = (U_i, L_i)$ of G. Let c_{G_j} be represented by $(U(c_{G_j}), L(c_{G_j}))$. Applying Lemma 2, we have that for $1 \leq j \leq n$, $U(c_{G_j}) = U_j$ and either $L(c_{G_j}) = L_j$ or $L(c_{G_j}) = L_j \cup \{v_0\}$ (v_0 is the full vertex of G).

Recall that each eigen-subgraph G_i of G corresponds to the element $(x_i, y_i, z_i) \in M$ in the instance I_{3DM}. Below, we prove that (a): the set $M' = \{(x_j, y_j, z_j) :$ the corresponding c_{G_j} contains $v_0\}$ covers $X \cup Y \cup Z$; and

(b): $|M'| = q$ and $C - \{c_{G_j} : 1 \leq j \leq n\} = \emptyset$.

If M' does not cover $X \cup Y \cup Z$, then there must be an element $\zeta \in X \cup Y \cup Z$ that is not covered by M'. According to the structure of G and the above property of c_{G_j} for every j, we can conclude that any edge from the vertex set $u(\zeta)$ to v_0 (in G) is not contained in $\{c_{G_j} : 1 \leq j \leq n\}$. Therefore, $u(\zeta)$ must be contained in $C - \{c_{G_j} : 1 \leq j \leq n\}$. Note that $|u(\zeta)| = (n+1)$. Consequently, $N(C) \geq n(3(n+1)^2 + 3(n+1)) + (n+1) > K$. This contradicts the assumption that C is a solution to $T(I_{3DM})$. Hence, (a) holds.

From Lemma 1 and the fact that C is a solution to $T(I_{3DM})$, (b) immediately follows. Consequently, $N(C) = K$ (because (a) and (b) imply that C takes the same form as that in proof of Lemma 3). Immediately, (a) and (b) also imply that M' is a solution to I_{3DM}. Thus the Lemma holds. $\square$

Now we can claim that DOPEC is NP-complete. This implies Theorem 1.

4 Conclusion

The edge concentration method has been demonstrated as a very effective tool to visualize dense graphs clearly by reducing graph density. Apart from the existence of several heuristics, the problem of having a greatest reduction on density for bipartite graphs was left open. In this paper, we prove that this problem is NP-hard. However, our proof does not exclude the possibility of developing a good heuristic to produce an approximation solution within a constant factor to the optimal solution. This will be our future study.

Acknowledgment

The author is very grateful to the anonymous referees for their comments and suggestions regarding an improvement of the presentation throughout the paper.

References

[1] J. A. Bondy and U. S. R. Murty, *Graph Theory with Application*, The Macmilan Press LTD, 1977.

[2] G. Di Battista, P. D. Eades, R. Tamassia, and I. Tollis, Algorithms for Automatic Graph Drawing: An Annotated Bibliography, *Computational Geometry: Theory and Application*, 4, 235-282, 1994.

[3] P. Eades, Q. Feng, X. Lin, and H. Nagamochi, Straight-Line Drawing Algorithms for Hierarchical Graphs and Clustered Graphs, to appear in *Algorithmica.*

[4] P. Eades, X. Lin and R. Tamassia, An Algorithm for Drawing a Hierarchical Graph, *International Journal of Computational Geometry and Applications*, 6(2), 145-155, 1996.

[5] M. R. Garey and D. S. Johnson, *Computers and Intractability: a guide to the theory of the NP-Completeness*, Freeman, 1979.

[6] X. Lin, *Analysis of Algorithms for Drawing Graphs*, PHD thesis, University of Queensland, 1992.

[7] F. Newbery, Edge Concentration: A Method for Clustering Directed Graphs, Graph Drawing by Force-Directed Placement, *International Workshop on Software Configuration Management*, Princeton, 1989.

[8] F. N. Paulish and W. F. Tichy, Edge: An Extendible Graph Editor, *Software-Practice and Experience*, 20(s1), 63-88, 1990.

[9] K. Sugiyama and K. Misue, Graph drawing by the magnetic spring model, *Journal of Visual Languages and Computing* 6 (3), 217 – 232, 1995.

[10] D. Tunkelang, A Practical Approach to Drawing Undirected Graphs, *TR* No. 2, Carnegie Mellon University, 1995.

[11] J. Warfield, Crossing Theory and Hierarchy Mapping, *IEEE Transaction on Systems, Man, and Cybernetics*, SMC-7(7), 502-523, 1977.