

Visualizing Algorithms for the Design and Analysis of Survivable Networks

Ala Eddine Barouni¹, Ali Jaoua², and Nejib Zaguia³

¹ University of Tunis, department of computer science, Tunisia
ala.barouni@fst.rnu.tn

² King Fahd University of Petroleum and Minerals, department of computer science,
Dhahran, Arabie Saoudia
ajaoua@ccse.kfupm.edu.sa

³ School of Information Technology and Engineering, Ottawa, Ontario, Canada
zaguia@site.uottawa.ca

Abstract. We present algorithms for the drawing of survivable telecommunication networks. The visualization of telecommunication networks is a very important problem. For some specific rings in a network, we may have a high traffic. The network designers may decide to add more equipment to the nodes (sites) of these rings in order to increase the performance of the network. Therefore, one of the most important properties of the survivable telecommunication network, is that rings should be easily recognizable. Given a ring cover of survivable telecommunication networks, we provide three techniques for drawing a ring cover. We should mention that all these drawings should respect many criteria in order to preserve the readability of the drawing. As in most of the graph drawing algorithms, the area used for the drawing is very important, our proposed algorithms produce drawings that require $O(n^2)$ area, where n is the number of nodes in the ring cover. These drawings are clear and easy to understand by the network designers.

1 Introduction

The problem of drawing a graph in the plane has received increasing attention recently due to the large number of applications stated in [1]. For example automatic layout of pert diagrams [2], layout algorithm for data flow diagrams [3] and layout algorithm of entity-relationships diagrams [4]. The design and analysis of telecommunication network is a very important area (for more details see [5], [6] and [7]). In this paper, we study techniques for visualizing telecommunication networks. We are motivated to design a network which (1) satisfies the traffic requirements, (2) can survives failures, and (c) the cost of the network is minimum. A network is *survivable* if it can survive the failure of a link, that is, the removal of the link e does not disconnect the network, and the traffic that was originally supported by the link e can be accommodated by another path. The “*Multi-Ring Architecture*” is considered

to be a cost-effective survivable network because of its simplicity and improved survivability (see [8] and [9]).

Consider a network N represented by a graph $G = (V, E)$, where V is the set of nodes representing the sites and E is a set of links representing the electrical links between nodes. A ring R in G is defined to be a cycle consisting of nodes $n_1, n_2, \dots, n_p \in V$. For some specific rings in a network, we may have a high traffic. The network designers may decide to add more equipment to the nodes (sites) of these rings in order to increase the performance of the network. Therefore, one of the most important properties of Multi-Ring Architecture, is that rings should be easily recognizable. We focus on drawing networks on the computer screen so that properties of the Multi-Ring Architecture can easily be viewed. Similar to graph drawing, many criteria should be considered in order to enhance the readability. Usually, a general optimization method are used, i.e., minimize the crossings, the area, the number of bends (in orthogonal drawings), and the number of slopes (in polyline drawings). Readability is also our concern, because we have noticed, so far, that unnecessary crossings may create rings that do not exist in the original network. In this paper, we will address the issue of visualizing survivable telecommunication networks by presenting multi-ring architecture drawing techniques.

We define a *Ring Cover* C as a set of rings that covers all links in a network [10]. For a network N represented by a graph G , given a ring cover C a *contact node* c is a node of G which belongs to the intersection of at least two rings of C . The *resolution* is a constant defined by the user which is considered to be the minimal distance between any two nodes in the drawing of the ring cover. The resolution rule is that the nodes must be kept far enough from each other, so that, the human eye can tell them apart. The resolution rule prevents the drawing algorithm from arbitrary scaling down the picture. The problem is defined as follows: let N be a telecommunication network represented by a graph $G = (V, E)$, where V is the set of nodes representing the sites of switches and E is a set of links representing the electrical wires or optical fiber links between nodes. Corresponding to the graph G , a $|V| \times |V|$ matrix T is defined so that: the entries T_{ij} denote the amount of traffic between the node i and the node j , where $1 \leq i, j \leq |V|$. In [10], some techniques are provided to find a ring cover in survivable networks by using the matrix T as input. Thus, for a network N , we assume that the ring cover C is given and therefore our task is to deal with the ring cover drawing. We shall present three different techniques of ring cover drawing. They take into account many criteria in order to produce layouts that are clear and easy to understand. Since the nodes of the network correspond to sites, they have geographic coordinates. Hence, the network can be drawn naturally with little effort. However, for some complicated structure of networks, the important properties that designers are interested in, such as rings, are not displayed.

In [11] and [12], three techniques of ring cover drawing are provided:

1. Inside Drawing : Each ring is drawn inside other rings.
2. Outside Drawing : Each ring is drawn outside of other rings.
3. Mixed Drawing : Each ring can be drawn outside or inside other rings.

These three techniques are based on the following assumptions:

1. Any two adjacent rings share only one contact node.
2. The *ring-contact node graph* [11] is a tree and therefore it contains no cycles.

They proposed two open problems:

1. Drawing a ring cover when any two adjacent rings share one contact node and the ring-contact node graph is not a tree.
2. Drawing a ring cover when rings share more than one contact node.

In [13] and [14], we presented two techniques (Inside and outside) of ring covers drawing in order to deal with the open problems presented in [11].

In this paper we focus on expanding the conditions of the ring cover drawing. Our aim is to present a new technique to draw the ring cover, called mixed drawing. Before presenting the mixed drawing algorithm, we should present the inside and the outside drawing, since the mixed drawing uses both the inside and the outside drawing. Our research includes the case treated in [11], [13] and [14], augmented by the following contributions:

1. The ring cover contains rings that share more than one contact node.
2. The ring-contact node graph, which is considered to be the underlying structure of the ring cover is not a tree.
3. A combination of both cases: the ring cover contains rings which share more than one contact node and the ring-contact node graph has cycles.
4. The design and the implementation of a new system for drawing telecommunication networks based on the Multi-Ring Architecture. The system invites the user to enter the ring cover through its underlying structure and then generates the picture of the network.

2 Inside Drawing

In this section, we will describe the inside drawing technique which consists of drawing each ring inside the other one. We will present our approach which consists of extending the drawing of the ring cover to other types of networks. We treat the case where any two adjacent rings in the ring cover can share more than one contact node, and we consider the possibility of having *ring-cycles* in the ring cover. Our approach consists of extending the drawing of the ring cover to other types of networks in which we consider the case where the rings could share more than one contact node and the ring cover contains *ring-cycles*. Before going into details, let us define some concepts:

A path P which connect two contact nodes c_i and c_j is called a *free* path if and only if c_i and c_j are the only contact nodes in this path.

A *ring-cycle* $RC = \{R_1, cn_1, R_2, cn_2, \dots, R_i, cn_i, R_{i+1}, cn_{i+1}\}$, where $R_i = R_{i+1}$ and $cn_i = cn_{i+1}$, is a set of rings R 's and special contact nodes cn 's, so that rings form a cycle in the ring cover, that is:

- Any two adjacent rings R_j and R_{j+1} share exactly one contact node cn_j , $j = 1, \dots, i$.
- And in each ring R_j , $j = 2, \dots, i+1$, there exists at least one free path between two distinct special contact nodes cn_{j-1} and cn_j belonging to the ring R_j .

Let a ring $R = \{n_1, n_2, \dots, n_p\}$ be an ordered sequence of its nodes, we call two nodes n_i and n_k ($i \neq k$) adjacent in R , if there exists a direct link e between n_i and n_2 within the

ring R . Let $R_x = \{n_1, n_2, \dots, n_p\}$ and $R_y = \{m_1, m_2, \dots, m_q\}$ be two rings, such that R_x and R_y are pairwise compatible [11], and we suppose that they share contact nodes $\{c_1, c_2, \dots, c_k\}$. A multi-contact node mc is the set $\{c_i, c_2, \dots, c_k\}$ of contact nodes, such that c_i, c_{i+1} are adjacent nodes in R_x and in R_y , where $1 < i < k-1$.

Let $MC = \{mc_1, \dots, mc_n\}$ be the set of all multi-contact nodes in the ring cover \mathcal{C} and $mc_i \in MC$ be the multi-contact node between two rings R_x and R_y . We say R_x and R_y are *strictly compatible* if $mc_i \cap mc_j = \emptyset$ for $j = 1, 2, \dots, n$ and $j \neq i$.

Our approach consists of specifying a new underlying structure of the ring cover which will be considered later on as the basis of our drawing. We assume first, that any two adjacent rings in the ring cover are strictly compatible, and we shall introduce the *generalized-ring-contact node graph* considered to be the new underlying structure of the ring cover. We define the *generalized-ring-contact node graph* $G_I(V_I, E_I)$ as follows:

V_I contains five types of vertices :

1. Rings in the ring cover, denoted by R_i .
2. Special contact nodes that belong to the ring-cycle, denoted by cn_i .
3. Contact nodes that do not belong to ring-cycle, denoted by cn .
4. Ring-cycles in the ring cover, denoted by RC_i .
5. Multi-contact nodes, denoted by mc_i .

E_I contains five types of edges defined as follow:

1. $\{(R, c) : R \in C, \text{ and } c \in R\}$.
2. $\{(RC, cn) : cn \in RC\}$.
3. $\{(cn, R) : R \in C, cn \in R, cn \in RC \text{ and } R \notin RC\}$.
4. $\{(RC, R) : R \in C \text{ and } R \in RC\}$.
5. $\{(R, mc) : R \in C, \text{ and nodes of } mc \text{ are in } R\}$.

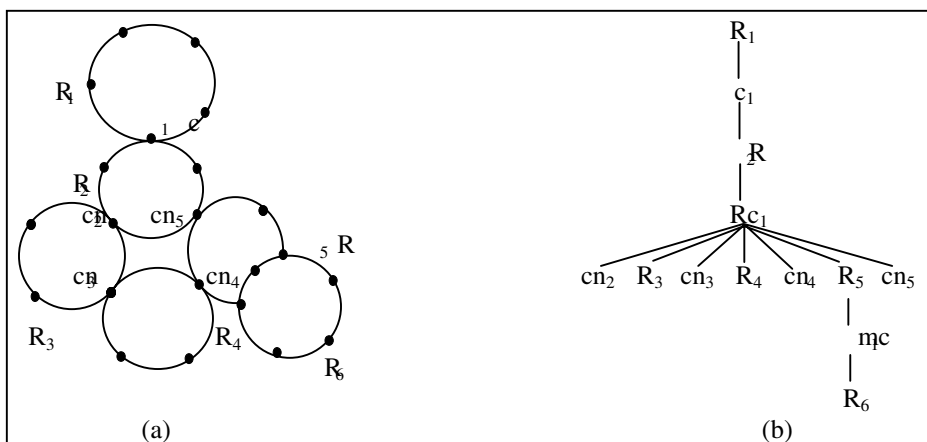


Fig. 1. (a) Ring cover; (b) Generalized-ring-contact node graph

In our drawing we focus only on the case where the generalized-ring-contact node graph G_I is a tree (see figure 1(b)), we call \mathcal{IT}_I . Our algorithm traverses the tree \mathcal{T}_I two times : The first traversal of the tree \mathcal{T}_I is in postorder fashion. It consists of computing the radius of the circles corresponding to the rings and the ring-cycles in the ring cover. The second traversal of the tree \mathcal{T}_I is in preorder fashion. It consists of placing the parent node and its children in the right location.

Using the above procedure, we conclude the following theorem:

Theorem 1:

Given a ring cover C and its generalized-ring-contact node tree, the inside drawing algorithm *InsideDraw* [15] produces a ring cover drawing such that :

1. there are no crossings;
2. the area required by the drawing is $O(n^2)$, where n is the number of nodes in C ;
3. the time complexity is $O(n + m^2)$, where n is the number of nodes in the ring cover and m is the number of the rings in the ring cover.

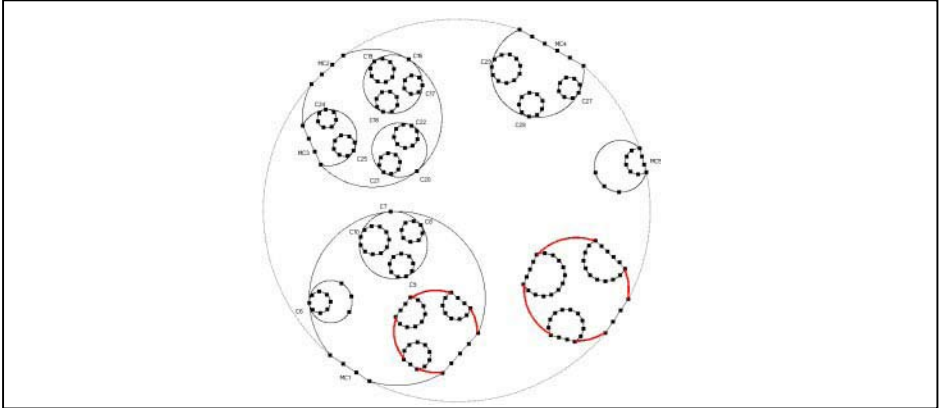


Fig. 2. The Ring cover contains ring-cycles and rings that share more than one contact node with other rings.

3 Outside Drawing

We present in this section the outside drawing technique which consists of drawing each ring outside the other rings. Given a ring cover C and its generalized-ring-contact node tree \mathcal{T}_I (presented in section 2), our algorithm draws rings outside each other, such that the resolution rule is respected, no crossings are generated, and a picture of the ring cover is produced in an optimal area. Our algorithm traverses the generalized-ring-contact node tree twice: The first traversal of the tree \mathcal{T}_I is in postorder fashion. It consists of computing the radius of the circles corresponding to the rings and the ring-cycles in the ring cover. The second traversal of the tree \mathcal{T}_I is in preorder fashion. It consists of placing the parent node and its children in the right position.

Using the above procedure, we conclude the following theorem:

Theorem 2:

Given a ring cover C and its generalized-ring-contact node tree, the outside drawing algorithm *OutsideDraw* [15] produces a ring cover drawing such that:

1. there are no crossings;
2. the area required by the drawing is $O(n)^2$, where n is the number of nodes in the ring cover.
3. the time complexity is $O(n + m^2)$, where n is the number of nodes in the ring cover and m is the number of the rings in the ring cover.

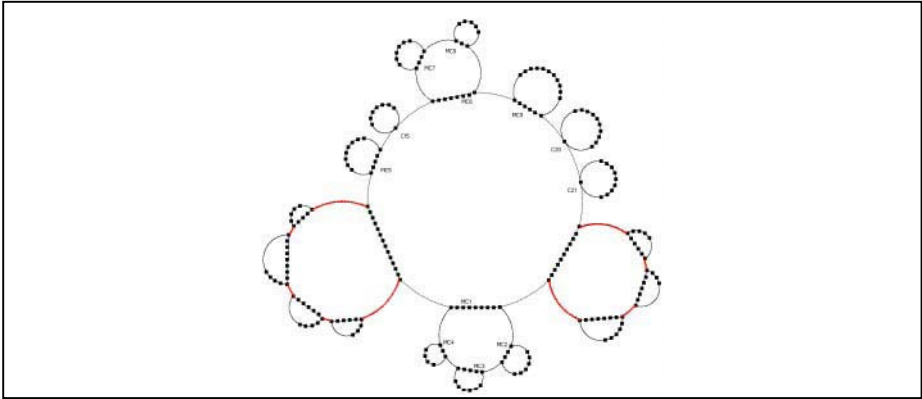


Fig. 3: Ring cover contains rings which share more than one contact node and contains two ring-cycles.

4 Mixed Drawing

In the first part of this section, we will present the mixed drawing technique which uses both inside and outside drawing algorithm for the ring cover drawing. We introduce the mixed drawing for the following reason. First, because it is impossible to draw rings outside each other, when more than two rings share the same multi-contact node. The second reason, is that, the mixed drawing algorithm uses less area than the outside drawing algorithm.

4.1 Basic idea

In the inside drawing, we are able to draw rings when more than two rings share a multi-contact node. Our idea consists of drawing some subtrees of T_r by the inside algorithm, and other subtrees by the outside algorithm. First, we traverse the tree in preorder fashion, if we encounter a ring node α , such that it contains a child which is a multi-contact node or a ring-cycle, then the ring corresponding to α and its subtree will be drawn by the inside drawing algorithm, else if α does not have any child

which is either a multi-contact node or a ring-cycle, then v and its children will be drawn one outside the other.

4.2 Algorithm

The algorithm *MixedDraw_LabelNode* will traverse the tree T_I in preorder fashion. If we encounter a node u which has at least one child which is either a multi-contact node or a ring-cycle, we assign a value true to *INSIDE* associated to the node u and all the nodes in the subtree of u . This means that, all the rings and the ring-cycles of the subtree of u will be drawn inside each other. If all the children of u are different than a ring-cycle and a multi-contact node, we assign the value false to *INSIDE* associated to the node u .

Algorithm *MixedDraw_LabelNode*(v);

Input: Ring cover C , and its generalized-ring-contact node tree T_I with the root v .

Output: An update of the label *INSIDE* associated to the rings and the ring-cycles.

Begin

1. **If** ((v is a ring that contains at least one child which is a multi-contact node or ring-cycle) or v is a ring-cycle) **then**

- Assign the value true to *INSIDE* associated to the node v and to all the nodes in the subtree of v **return**;

EndIf;

2. **If** (v is a ring **and** (all the children of v are contact nodes) **then**

Begin

- Assign the value false to *INSIDE* associated to the node v ;
- **For** all the children u_i of v **do** *MixedDraw_LabelNode* (u_i);

EndIf;

3. **If** (v is a contact node) **then**

- **For** all the children u_i of v **do** *MixedDraw_LabelNode* (u_i);

EndIf;

End.

The time complexity of *MixedDraw_LabelNode* algorithm is of order $O(n)$, where n is the number of nodes in the ring cover. We process (by means of updating the *INSIDE* label) each node in T_I only one time.

In the second traversal of T_I , the algorithm starts with the root of T_I . If v is a ring or a ring-cycle which *INSIDE* value is true, then we call the *InsideDraw* algorithm to compute the radius of the circles corresponding to the rings and the ring-cycles \tilde{r}_i . If the value of *INSIDE* is equal to false, then the node v and all its children will be drawn each one outside the other. By doing so we mix both drawings: the inside drawing and the outside drawing to obtain a new drawing called the mixed drawing.

Algorithm *MixedDraw*(v)

Input: Ring cover C , and its generalized-ring-contact node tree T_l with the root v .

Output : Radius of the circles corresponding to the rings and the ring-cycles in C .

Begin

1. **If** ((v is a ring or v is a ring-cycle) and ($INSIDE(v)$ is true))

then

- *InsideDraw*(v); **Return**;

EndIf;

2. **If** ((v is a ring) and ($INSIDE(v)$ is false)) **then**

Begin

- **For** all the children u_i of v **do** *MixedDraw*(u_i);
- *OutsideDraw*(v) { we place v and its children one outside the other }

EndIf;

3. **If** (v is a node representing a contact node) **then**

- **For** all the children u_i of v **do** *MixedDraw*(u_i);

EndIf;

End.

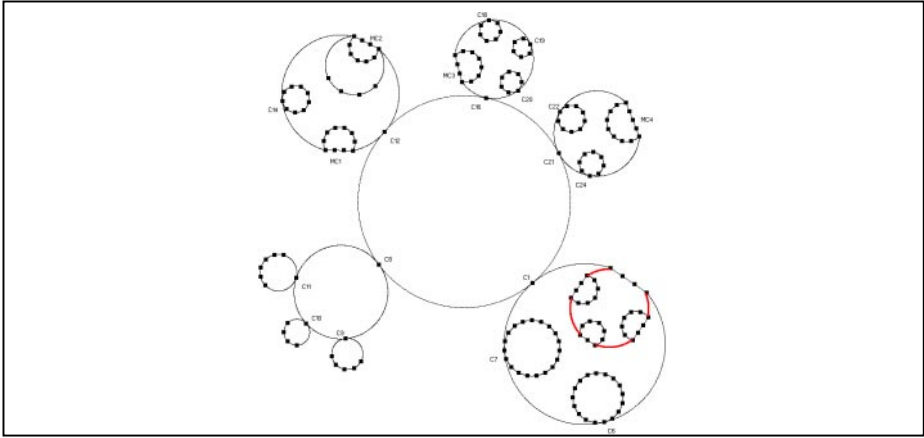


Fig. 4. A picture of a ring cover generated by the system. We notice that if a ring u has at least one child which shares more than one contact node with other rings or a ring-cycle, then u and all the nodes of its subtree will be drawn one inside the other.

Using the above procedure, we conclude the following theorem:

Theorem 3

Given a ring cover C and its generalized-ring-contact node tree T_l , the algorithm *MixedDraw* produces a ring cover drawing such that:

1. There are no crossings;
2. the area required by the drawing is $O(n^2)$, where n is the number of nodes in C ;
3. the time complexity is $O(n + m^2)$, where n is the number of nodes in the ring cover and m is the number of the rings in the ring cover.

5 Conclusion and Open Problems

In this paper, we presented three techniques for drawing a ring cover of survivable telecommunication networks: the inside drawing, the outside drawing and the mixed drawing. Our contribution to this subject was to expand the drawing of the ring cover to more complicated structures of survivable telecommunication networks. In fact, we presented an improved version for the outside and the inside drawing in order to deal with the case where the rings share more than one contact node. We proposed the ring-multi-contact node graph as a new underlying structure for the ring cover. Then we have considered the case where the ring cover contains ring-cycles in which we have presented the extended-ring-contact node graph as the new underlying structure for the ring cover. Finally we considered the case where the ring cover contains ring-cycles and rings that share more than one contact node. We also introduced the generalized-ring-contact node graph as the final underlying structure for the ring cover. While dealing with the outside drawing technique, we encountered the problem that occurs when at least three rings share the same multi-contact node, that is why we proposed a new approach for the mixed drawing algorithm.

In order to guarantee the readability of the drawing, we have to respect the following criteria:

1. No crossing is allowed, because if it is the case, we may visualize rings that do not exist in the original network.
2. Rings should be easily recognizable in the picture.
3. The resolution rule should be respected (for visibility purposes).
4. Use of minimal area.

Our proposed drawing algorithms produce drawings that require $O(n^2)$ area, where n is the number of nodes in the ring cover. Besides our theoretical research, we have done an experimental work which consist of implementing the previously designed algorithm. For efficiency purposes, we used the object-oriented language called "Borland Delphi".

There are still some issues which have not been addressed yet and that we consider them as being open problems, two of them are described here:

1. Are there other reasonable ways to draw ring covers ?
2. How can we draw ring cover when the generalized-ring-contact node graph is not a tree ?

6 Acknowledgement

We would like to thank Professor Ioannis G. Tollis for his help and the interesting discussion we had about this subject.

References

1. G. Di Battista, P. Eades, R. Tamassia and I.G. Tollis. "Algorithms for Automatic Graph Drawing: An Annotated Bibliography," Department of computer Science, Brown University, Technical Report, 1993.
2. G. Di Battista, E. Pietrosanti, R. Tamassia, and I.G. Tollis, "Automatic Layout of PERT Diagrams with XPERT," Proc. IEEE Workshop on Visual Languages (VL'89).
3. C. Batini, E. Nardelli, and R. Tamassia, "A Layout Algorithm for Data Flow Diagrams". IEEE Trans Software Eng., vol. 12 1986.
4. C. Batini, M. Talamo, and R. Tamassia, "Computer Aided Layout of Entity-Relationships Diagrams" The Journal of Systems and Software, vol. 4, 1984.
5. J.C.Shah , "Restoration Network Planning Tool", Proc. 8th Annual Fiber Optic Engineers Conf. April 21, 1992.
6. W.D.Grover, B.D.Venables, J.H. Sandham, and A.F. Milne, "Performance Studies of a Self-Healing Network Protocol in Telecom Canada Long Haul Networks " Proceedings of IEEE GLOBECOM 1990.
7. H.Sakauchi, Y.Nishimura, and S.Hasegawa, " A Self-Healing Network with an economical Spare-Channel Assignment " IEEE GLOBECOM 1990.
8. Tsong-Ho Wu and R. C. Lau , "A Class of Self-Healing Ring Architectures for SONET Network Application" IEEE Trans. on Communication, Vol. 40, No. 11 Nov. 1992.
9. T.H. Wu, D.J. Collar, and R.H. Cardwell, "Survivable Network Architectures for Broad-band Fiber Optic Networks: Model and Performance Comparison" IEEE journal of Lightwave. Technology, Vol. 6, No. 11, November 1988.
10. L.M. Gardner, M. Heydari, J. Shah, I.H. Sudborough, I.G. Tollis, and C. Xia, "Techniques for Finding Ring Covers in Survivable Networks" Proceedings of IEEE GLOBECOM 1994.
11. Ioannis G. Tollis and Chunliang Xia. "Drawing Telecommunication Networks". DIMACS International Workshop, Graph Drawing'94. Princeton, New Jersey, USA, October 1994. Proceedings.
12. Ioannis G. Tollis and Chunliang Xia. "Graph drawing algorithms for the design and analysis of telecommunication networks". Graph Drawing'93, Proceedings of the ALCOM International workshop on graph drawing Sèvres France.
13. Ala Eddine Barouni, Ali Jaoua and Nejib Zaguia. "Drawing Algorithms for telecommunication networks". In 3rd CGC workshop on Computational Geometry. Brown University, Providence, RI, USA. October 1998.
14. Ala Eddine Barouni, Ali Jaoua and Nejib Zaguia. "Planar Drawing Algorithms of survivable telecommunication networks". In Proceedings of the Japan Conference on Discrete and Computational Geometry. Tokai University, Japan. December 1998.
15. Ala Eddine. Barouni "Drawing Algorithm for survivable telecommunication networks", Master Theses at the university of Ottawa Canada. April 1997.