A Heuristic for the Constrained One-Sided Two-Layered Crossing Reduction Problem for Dynamic Graph Layout

by

Dung Mai

A dissertation Idea Paper submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

Graduate School of Computer and Information Sciences
Nova Southeastern University

May 14, 2005

Problem Statement and Goal

Introduction

General hierarchical graph layout as shown in Figure 1 has been used effectively to display relationships between objects (North, 1995). Some examples include entity relationship models in databases, the Unified Modeling Language (UML) in software engineering, management organizational charts, and hierarchical layouts in computer networking to display Internet networks (Battista, Eades, Tamassia, & Tollis, 1999; Cohen, Battista, Tamassia, Tollis, & Bertolazzi, 1992; North & Woodhull, 2001; Raitner, 2004).

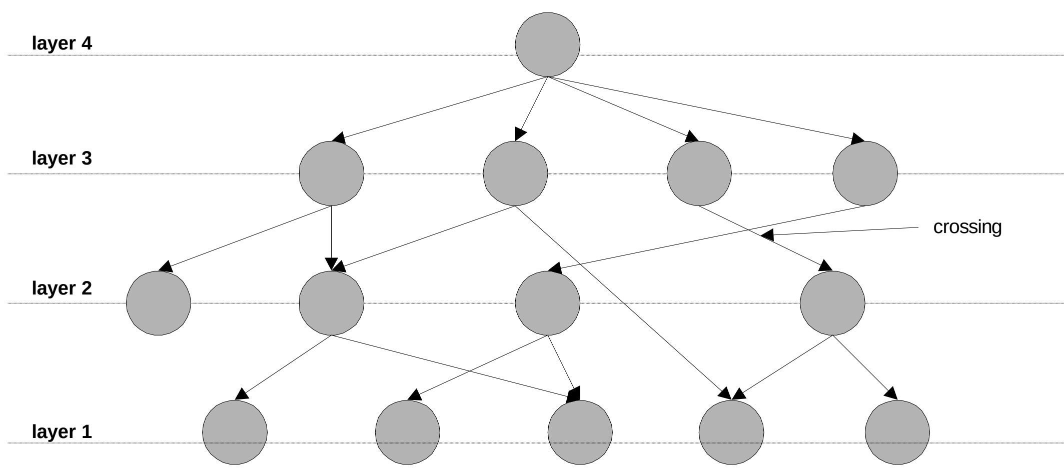

Figure 1. A general hierarchical graph layout

Although current hierarchical graph layout algorithms have been well studied (North & Woodhull, 2001) and have been effective for drawing static graphs of reasonable size, some graph visualization applications have ported to the Internet, whose environment is dynamic and demands fast rendering of large graphs. Other applications such as graph editing tools are required to support more sophisticated interactions such as inserting and deleting vertices from the graph or zooming in and zooming out in real-time mode (Cohen et al., 1992; North & Woodhull, 2001) while still fulfilling aesthetic criteria (North, 1995). Most standard graph layout algorithms are not incrementally stable. A simple change in the data may yield unpredictable changes between successive layouts (North, 1995), reducing the algorithms' scalability for real graph drawing applications. This problem motivated research in dynamic graph drawing algorithms. Cohen, Battista, Tamassia, Tollis, and Bertolazzi (1992) proposed generic algorithms for different types of graph drawing techniques. There has been significant progress in the hierarchical drawing area, such as an incremental graph layout called DynaDAG (North, 1995; North & Woodhull, 2001) and hierarchical graph views for dynamic graph layouts (Raitner, 2004).

The main DynaDAG algorithm is based on a well-known algorithm for drawing a hierarchical graph layout proposed by Sugiyama, Tagawa, and Toda (1981). The Sugiyama algorithm has three basic steps: (1) layer assignment, (2) crossing reduction, and (3) horizontal coordinate assignment. If the input graph is cyclic, a preprocessing step is used first to eliminate cycles. The crossing reduction problem for directed acyclic graphs is an NP-complete problem (Eades, McKay, & Wormald, 1986; Garey & Johnson, 1983). Thus most proposed heuristics have focused on one-sided two-layered crossing reduction, which computes the minimum number of crossings between two layers one at a time. The algorithm holds all the nodes of one side fixed and permutes the nodes of the other side until the number of crossings is minimal. Figures 2-1 and 2-2 show the arrangement of the nodes in the two layers of a graph before and after the algorithm is performed.

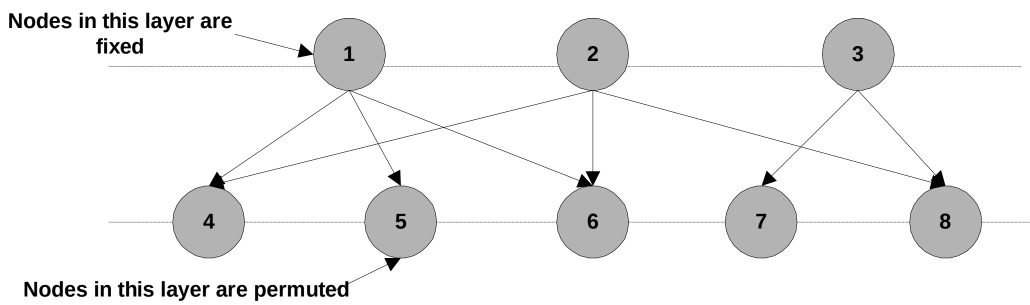

Figure 2-1. Before the one-sided two-layer crossing reduction problem algorithm is performed

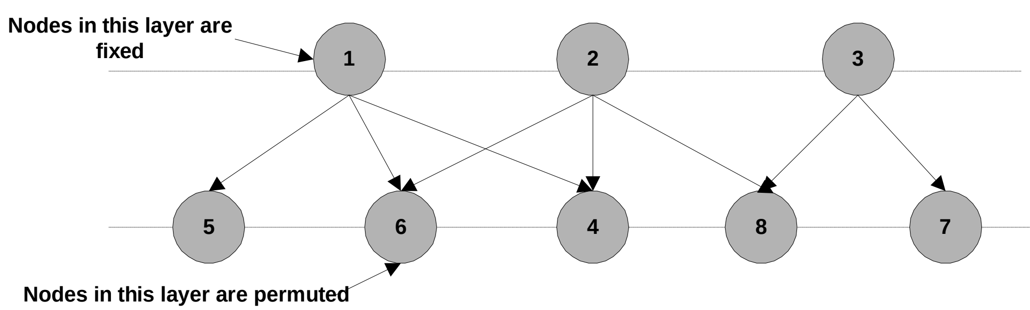

Figure 2-2. After one-sided two-layer crossing reduction problem algorithm is performed

Problem

According to North (1995) a dynamic graph layout is a sequence of successive graph layouts (L_0, L_1, L_2… L_n), and a good dynamic graph layout should have minimal changes in layout between two successive layouts L_i and $L_{(i+1)}$. To measure the effectiveness of a dynamic graph layout, North (1995) proposed three aesthetic properties: *consistency*, *stability*, and *readability*. *Consistency* means that the layout should adhere to the predefined business rules for a domain, *stability* requires minimal changes between successive layouts, and *readability* helps to make the layout easier to comprehend (North, 1995). The DynaDAG system uses these three parameters as rules for generating successive layouts. Thus the aesthetic criteria influence the Sugiyama algorithm, which the DynaDAG system uses, in constructing hierarchical graph layouts; especially the second step, which minimizes the number of crossings. Aesthetic criteria in fact impose constraints on the layout, and these constraints impact the computation of the crossing reduction problem.

The current implementation of the DynaDAG system has not taken into account the influence of the aesthetic criteria when computing the minimal number of crossings. Thus balancing the aesthetic criteria with the reduction in the number of crossings presents an interesting and challenging problem for one-sided two-layered crossing reduction for dynamic graph layout. Further research is needed to address this problem.

Goal

The objective of this research is to develop a method to solve the crossing reduction problem for dynamic graph layout. The proposed heuristic should achieve an optimal solution that reduces the numbers of crossings but also satisfies North's (1995) aesthetic criteria. The proposed dissertation will extend the North and Woodhull (2001) research and develop a variant version of the online hierarchical graph drawing system, while finding a heuristic for solving the problem of minimizing the number of crossings. The specific goals of the this research are as follows:

(1) Extend the work of North and Woodhull (2001) by developing an incremental graph drawing framework that supports four operations as follows:

a. Inserting a node or pseudo-node n and a given set of edges connecting n to existing nodes.

b. Deleting a node n and all its incident nodes.

c. Adding an edge.

d. Deleting an edge.

(2) Formulate a mathematical equation that defines the relationship between aesthetic constraints and the minimization of the number of crossings for the one-sided two-layered crossing reduction problem.

(3) Develop a heuristic to solve this specialized constrained one-sided two-layered crossing problem.

Relevance and Significance

Relevance

Graph visualization and graph drawing play key roles in many applications across disciplines; for instance, relational database modeling, objected oriented modeling or UML, business modeling, organizational diagrams, molecular layout, and DNA layout (Battista et al., 1999). With advances in computer hardware and the exponential growth of data, user experience with computer visualization is also getting more sophisticated. Static graph layouts do not scale up well for displaying very large data sets or interactive graph layouts where users need to interact with the graph in real time. Cohen et al. (1992) proposed a dynamic algorithm for drawing planar graphs for a variety of standard drawings. The algorithm defined an important property for dynamic graph drawing called "smooth update." This property enables consistence and stability between successive layouts. North (1995) proposed a specialized incremental graph layout system called DynaDAG, which supports hierarchical graph drawing. This system extends the Sugiyama heuristic and includes several dynamic operations such as "insert node or edge” and “delete node or edge.” North (1995) also formalized the notion of “smooth update” and associated it with the consistency, stability, and readability criteria. Raitner (2004) proposed a solution based on a clustering graph layout that overcomes the physical limitation of computer screen size by allowing users to focus on certain parts of the graph using zooming in and zooming out mechanisms. Buchsbaum and Westbrook (2000) also developed similar algorithms for maintaining large hierarchical graph layouts such as depictions of telephone call data structures. However, most current research concentrates on dynamically updating the layout or maintaining the layout; few present a way to minimize the number of crossings for dynamic graph layouts. The original contribution of the proposed dissertation is to propose a heuristic that reduces the number of crossings but also satisfies the aesthetic criteria defined by North (1995).

Significance

The proposed research will contribute to the dynamic graph layout field by developing a method for solving the crossing reduction problem for dynamic graph layout.

Barriers and Issues

There is an inherent tradeoff between satisfying the criteria for a good hierarchical dynamic graph and minimizing the number of crossings (Forster, 2004). Satisfying the criteria may increase the number of crossings, but reducing crossings compromises the criteria. Thus minimizing the number of crossings can be defined as a constraint in the proposed mathematical equation that is influenced by aesthetic criteria. Many researchers have focused on solving the constrained one-sided two-layered crossing problem, but none of those researchers has studied constraints that are specialized for dynamic hierarchical graph drawing. Finocchi (2001) asserted that minimizing the number of crossings for a constrained dynamic graph layout is NP-complete. Thus, exploring the trade-offs between the minimizing the number of crossings and satisfying the aesthetic criteria, this dissertation seeks a solution that balances the optimization and the aesthetic criteria.

Approach

This research involves two main tasks. The first task is to develop a modified version of the on-line graph drawing system that was proposed by North et al. The operations supported by the proposed dynamic graph layout system and the aesthetic criteria proposed by North et al. will influence the formulation of a mathematical relation that defines the relationship between aesthetic criteria and minimizing the number of crossings for the two-layered crossing problem. The second task is to develop a method for solving the crossing reduction problem.

Design

Overall system: A modified version of the on-line graph drawing system will support four operations: *insert a node with a given set of edges connecting the node to existing nodes*, *delete a node and all its incident edges*, *insert an edge connecting existing nodes*, and *delete an edge*. These operations and the aesthetic criteria described earlier will be used as a foundation for designing and implementing a specialized constraint for dynamic graph drawing. Thus they will influence the design of a new technique for solving the constrained two-layered crossing problem that is specialized for dynamic graph drawing.

Data structure: The proposed dynamic graph drawing system shall have a data structure that is used to capture the vertices and edges attributes and support the aforementioned dynamic operations. A concrete data structure will be constructed in the implementation phase.

Criteria for the constrained one-sided two-layered crossing problem: Based on North (1995), a good dynamic graph layout system should produce a stable successive sequence of graph layouts. Bohringer and Newbery (1990) suggested defining specialized constraints on the dynamic graph layout that will maintain the stability of successive layouts. According to North (1995), while creating a constraint for static graph layout is easy, creating a constraint for a dynamic graph layout is dependent on a set of the operations that the system supports. The four operations of the proposed system described earlier shall influence the design of the constraints function.

Algorithms: The crossing reduction problem will be defined as a multivariate function, of which the optimal value is a function that depends on three aforementioned aesthetic criteria and minimizing the number of crossings. Each variable shall have a weight based on its importance. Based on the work of Forster (2004), a modified version of the barycenter heuristic will be developed. This modified version will need to be adapted for this specialized constrained one-sided two-layer crossing problem based on the design and implementation of the dynamic graph system.

Resources

A prototype of the proposed dynamic graph layout system shall use the One-line Hierarchical Graph Drawing proposed by North & et al. as a foundation.

Software

The system will use an open source relational database management system for storing graph data, and will be built using the Java language.

Hardware

Typical personal computers will be used in this research; no special hardware has been identified at this time.

References list

Battista, G. D., Eades, P., Tamassia, R., & Tollis, I. (1999). *Graph drawing algorithms for the visualization of graphs*. New Jersey: Prentice Hall.

Bohringer, K., & Newbery, P. (1990). Using constraints to achieve stability in automatic graph layout algorithms. *Proceedings of ACM CH 90*, 43-51.

Buchsbaum, A. L, & Westbrook, J. R. (2000). Maintaining hierarchical graph views. Proceedings of the Eleventh Annual ACM-Siam Symposium on Discrete Algorithms. 566 – 575.

Cohen, R. F., Battista, G. D., Tamassia, R., Tollis, I. G, & Bertolazzi, P. (1992). *A framework for dynamic graph drawing*. Annual Computational Geometry. Berlin, Germany.

Eades, P., McKay, B. D., and Wormald, N. C. (1986). On an edge crossing program. Proceedings of the *Australian Computer Science Conference*, 327-334.

Finocchi, I. (2001). Layered drawings of graphs with crossing constraints. *Proc. COCOON'01, V. 2108 of LNCS*, 357–368. Springer.

Forster, M. (2004). *A fast and simple heuristic for constrained two-layered crossing reduction*. Graph Drawing 2004.

Garey, M. R., & Johnson, D. S. (1983). Crossing number is NP-complete. *SIAM Journal on Algebraic and Discrete Methods*, 312-316.

North, S. (1995). *Incremental layout in DynaDAG*. Software and Systems Research Center. AT & T Bell Laboratories.

North, S., Woodhull, G. (2001). On-line hierarchical graph drawing. *9th International Symposium on Graph Drawing*, 232 – 246.

Raitner, M. (2004). Maintaining hierarchical graph views for dynamic graphs. *Technical Report, MIP-0403*, University of Passau.

Sugiyama, K., Tagawa, S., & Toda, M. (1981). Methods for visual understanding of hierarchical systems. *IEEE Trans. On System, Man, and Cybernatics, no(2):*109-125.