Interactive Graph Drawing Algorithms

Ashim Garg
Department of Computer Science and Engineering
State University of New York at Buffalo
Buffalo, NY 14260
agarg@cs.buffalo.edu

Abstract

Traditional graph drawing algorithms can also be called *one-shot* algorithms because given a graph as input, they typically construct a drawing only once. However, in several applications, such as software engineering and database design, users interact extensively with a displayed graph, continuously adding or deleting its vertices and edges. In such a scenario, a graph drawing system should update a drawing each time the displayed graph is updated by the user. Unfortunately, traditional drawing algorithms may not be suitable in such situations. Since they typically construct a drawing from scratch, they may fail to update a drawing quickly after the user updates the displayed graph. Also, the new drawing constructed after the update may be significantly different from the previous drawing even if only a small change was made in the displayed graph, making it difficult for the user to correlate the previous drawing and the new drawing (making only a small change in a drawing when a small change is made in the displayed graph is also called *preserving the user's mental map* [4] of the displayed graph). Hence, it is important to devise techniques especially suited for interactive settings. In this paper, we survey some of the techniques, such as those of [6, 5, 4, 4, 1, 3, 2, 8, 7], developed for interactive graph drawing. Notice that the term *interactive graph drawing* has a broad meaning, covering several aspects of graph visualization, including geometric positioning of graph objects, animation, and support for user-interaction. In this paper, we focus

only on the problem of constructing geometric representations of the vertices and edges of a graph in a setting that allows addition or deletion of its vertices and edges.

References

[1] T. Biedl and M. kaufmann. Area-efficient static and incremental graph drawing. In *Proc. 5th Annu. European Sympos. Algorithms*, volume 1284 of *Lecture Notes Comput. Sci.*, pages 37–52. Springer-Verlag, 1997.

[2] U. Brandes and D. Wagner. A bayesian paradigm for dynamic graph layout. In G. Di Battista, editor, *Graph Drawing (Proc. GD '97)*, volume 1353 of *Lecture Notes Comput. Sci.*, pages 236–247. Springer-Verlag, 1997.

[3] S. Bridgeman, J. Fanto, A. Garg, R. Tamassia, and L. Vismara. Interactivegiotto: An algorithm for interactive orthogonal graph drawing. In G. Di Battista, editor, *Graph Drawing (Proc. GD '97)*, volume 1353 of *Lecture Notes Comput. Sci.*, pages 303–308. Springer-Verlag, 1997.

[4] P. Eades, W. Lai, K. Misue, and K. Sugiyama. Preserving the mental map of a diagram. In *Proceedings of Compugraphics 91*, pages 24–33, 1991.

[5] K. Miriyala, S. W. Hornick, and R. Tamassia. An incremental approach to aesthetic graph layout. In *Proc. Internat. Workshop on Computer-Aided Software Engineering*, 1993.

[6] S. Moen. Drawing dynamic trees. *IEEE Software*, 7:21–8, 1990.

[7] S.C. North. Incremental layout in dynaDAG. In F.J. Brandenburg, editor, *Graph Drawing (Proc. GD '95)*, Lecture Notes Comput. Sci., pages 409–418. Springer-Verlag, 1996.

[8] A. Papakostas and I.G. Tollis. Issues in interactive orthogonal graph drawing. In F.J. Brandenburg, editor, *Graph Drawing (Proc. GD '95)*, Lecture Notes Comput. Sci., pages 419–430. Springer-Verlag, 1996.