Information Processing Letters 31 (1989) 7–15
North-Holland

12 April 1989

AN ALGORITHM FOR DRAWING GENERAL UNDIRECTED GRAPHS

Tomihisa KAMADA and Satoru KAWAI

Department of Information Science, Faculty of Science, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113 Japan

Communicated by E.C.R. Hehner
Received 4 July 1988
Revised 13 November 1988

Keywords: Graph, network structure, layout, drawing algorithm

1. Introduction

Graphs (networks) are very common data structures which are handled in computers. Diagrams are widely used to represent the graph structures visually in many information systems. In order to automatically draw the diagrams which are, for example, state graphs, data-flow graphs, Petri nets, and entity-relationship diagrams, basic graph drawing algorithms are required.

The state of the art in automatic drawing is surveyed comprehensively in [7,19]. There have been only a few algorithms for general undirected graphs. This paper presents a simple but successful algorithm for drawing undirected graphs and weighted graphs. The basic idea of our algorithm is as follows. We regard the desirable "geometric" (Euclidean) distance between two vertices in the drawing as the "graph theoretic" distance between them in the corresponding graph. We introduce a virtual dynamic system in which every two vertices are connected by a "spring" of such desirable length. Then, we regard the optimal layout of vertices as the state in which the total spring energy of the system is minimal.

The "spring" idea for drawing general graphs was introduced in [6], and similar methods were used for drawing planar graphs with fixed boundary [2,20]. This paper brings a new significant result in graph drawing based on the spring model.

2. Graph drawing problem

We will clarify the graph drawing problem we treat in this paper. There are really a lot of ways to visualize graph structures. In some methods, the positions of vertices are restricted, e.g., they are placed on grid points [1,19], concentric circles [4], or parallel lines [14,18]. Edges can be drawn as straight lines, polygonal lines, or curves. We treat here the drawings in which the positions of vertices are not restricted and edges are drawn as straight lines. So our purpose is to determine the positions of vertices for a given graph G. (G is expressed by the set of vertices V and the set of edges E). And we assume that a given graph is connected. The picture of a disconnected graph is obtained by drawing its connected components separately. A graph can be decomposed into connected components in running time $O(|V|+|E|)$ [15].

First of all we must discuss the fundamental conditions of the general view of a graph. The graph structure encompasses so many kinds of structures, from trees to complete graphs, that it is difficult to find out the common criteria of nice drawings. However, the following two requirements in drawing graphs