Computing β-Drawings of 2-Outerplane Graphs in Linear Time
(Extended Abstract)

Md. Abul Hassan Samee, Mohammad Tanvir Irfan, and Md. Saidur Rahman

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology (BUET)
Dhaka - 1000, Bangladesh
{samee,mtirfan,saidurrahman}@cse.buet.ac.bd

Abstract. A straight-line drawing of a plane graph G is a drawing of G where each vertex is drawn as a point and each edge is drawn as a straight-line segment without edge crossings. A proximity drawing Γ of a plane graph G is a straight-line drawing of G with the additional geometric constraint that two vertices of G are adjacent if and only if no other vertex of G is drawn in Γ within a "proximity region" of these two vertices in Γ. Depending upon how the proximity region is defined, a given plane graph G may or may not admit a proximity drawing. In one class of proximity drawings, known as β-drawings, the proximity region is defined in terms of a parameter β, where $\beta \in [0, \infty)$. A plane graph G is β-drawable if G admits a β-drawing. A sufficient condition for a biconnected 2-outerplane graph G to have a β-drawing is known. However, the known algorithm for testing the sufficient condition takes time $O(n^2)$. In this paper, we give a linear-time algorithm to test whether a biconnected 2-outerplane graph G satisfies the known sufficient condition or not. This consequently leads to a linear algorithm for β-drawing of a wide subclass of biconnected 2-outerplane graphs.

Keywords: Graph Drawing, Proximity Drawing, β-Drawing, Proximity Graph, 2-Outerplane graph, Slicing Path, Good Slicing Path.

1 Introduction

Let Γ be a straight-line drawing of a plane graph G. Let $\Gamma(u)$ be the point on the plane to which the vertex u of G is mapped in Γ. Then Γ is a *proximity drawing* of the plane graph G if Γ satisfies the following proximity constraint: two vertices u and v of G are adjacent if and only if a well-defined "proximity region" corresponding to the points $\Gamma(u)$ and $\Gamma(v)$ is empty, i.e. the region does not contain $\Gamma(w)$ for any other vertex w of G. The exact definition of proximity region is problem-specific. As a matter of fact, there is an infinite number of different types of proximity regions. For example, an infinite family of parameterized proximity regions has been introduced in [5]. This family of parameterized proximity regions gives rise to an important class of proximity drawings, known as *β-drawings*, where β stands for a parameter that can take any real number value in $[0, \infty)$.

A plane graph G is β-drawable if G admits a β-drawing. Not all graphs are β-drawable for all values of β. For example, the graph G_1 illustrated in Fig. 1(a) is not

S.-i. Nakano and Md. S. Rahman (Eds.): WALCOM 2008, LNCS 4921, pp. 81–87, 2008.

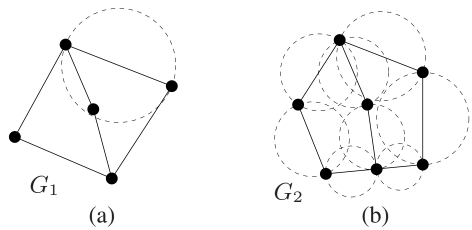

Fig. 1. (a) A graph G_1 which is not β-drawable for $\beta \in (1, 2)$, and (b) a graph G_2 which is β-drawable under the same constraints

β-drawable for $\beta = 1$. This fact can be explained as follows. Suppose we want to achieve a β-drawing of G_1 with $\beta = 1$. The β-region of two vertices u and v for $\beta = 1$ is a circle with $\Gamma(u)$ and $\Gamma(v)$ as its two antipodal points. For the graph G_1, wherever we place the four external vertices, the internal vertex will be inside the β-region of at least one of the four pairs of neighboring external vertices (as shown with the dotted circle in Fig. 1(a)). Therefore the graph G_1 is not β-drawable for $\beta = 1$. Following this same line of reasoning, one can easily work out that the graph G_2 shown in Fig. 1(b) is β-drawable for $\beta = 1$.

The *proximity drawability problem*, i.e. the problem whether a given graph admits a particular proximity drawing or not, has originated from the well-known "proximity graphs." Proximity graphs have wide applications in computer graphics, computational geometry, pattern recognition, computational morphology, numerical analysis, computational biology, GIS, instance-based learning and data-mining [3].

Several research outcomes regarding the proximity drawability of trees and outerplanar graphs are known [1, 2, 6]. One of the problems left open in [6] is to extend the problem of β-drawability of graphs to other nontrivial classes of graphs apart from trees and outerplanar graphs. In [4], the authors gave a sufficient condition for β-drawability of biconnected 2-outerplane graphs, where $\beta \in (1, 2)$. Although their sufficient condition induces a large and non-trivial class of biconnected 2-outerplane graphs, their algorithm for testing whether a given biconnected 2-outerplane graph satisfies those conditions or not, takes time $O(n^2)$.

In this paper, we give a linear-time algorithm for testing whether a biconnected 2-outerplane graph G satisfies the sufficient condition presented in [4]. Our algorithm essentially relies on the sufficient condition presented in [4], but works on a new set of conditions devised by us on "slicing paths" of G.

The rest of this paper is organized as follows. In Section 2, we present some definitions and preliminary results. In Section 3, we give a linear-time algorithm to test whether G satisfies the sufficient condition presented in [4] or not, and in the positive case, to compute a β-drawing of G. Finally, Section 4 is a conclusion.

2 Preliminaries

In this section we give some definitions and present our preliminary results.