Information Processing Letters 113 (2013) 55–59

Contents lists available at SciVerse ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

An optimal algorithm for computing the non-trivial circuits of a union of iso-oriented rectangles

Panagiotis D. Alevizos

Department of Mathematics, University of Patras, GR-265 00 Patras, Greece

ARTICLE INFO

Article history:
Received 17 May 2012
Received in revised form 14 September 2012
Accepted 11 November 2012
Available online 23 November 2012
Communicated by M. Yamashita

Keywords:
Computational geometry
Circuit
Contour
Rectilinear polygons
Optimal algorithm
Sweep plane

ABSTRACT

Given n rectangles $R_1, \ldots, R_n$ on the plane with sides parallel to the coordinate axes, Lipski and Preparata (1981) [1] have presented a $\Theta(n \log n)$ time and $O(n \log n)$ space algorithm for computing the non-trivial circuits of the union $\mathcal{U} = R_1 \cup \cdots \cup R_n$. In this paper, we are presenting a simple algorithm, which computes the non-trivial circuits of $\mathcal{U}$ in $\Theta(n \log n)$ time and $\Theta(n)$ space.

1. Introduction

Rectangles and orthogonal segments are the fundamental ingredients of a number of applications, some of which have acquired extreme importance in the wake of current technological breakthroughs. Suffice it to mention its relevance to Very-Large-Scale-Integration (VLSI) [2–4], to geography and related areas [5], to computer graphics and animation (hidden-line problems, shadows, etc.) [6,7], and to the organization of data in two-dimensional magnetic bubble memories [8]. In [9–11], an interesting model, which is based on the geometry of rectangles, has been developed, in order to control concurrent operations (access and update) to a database by several users (this method is described also in [12]). There is also considerable theoretical interest in problems related to the one discussed in this paper. Algorithms and data structures have been devised for quite a number of related problems, such as testing a set of rectangles for disjointness and reporting all intersecting pairs of rectangles [13], determining the measure of their union, i.e., the size of the covered area of the plane [14] (this algorithm is described in [15]), answering intersection queries (given a query rectangle, which rectangles of the set intersect it) [16], to name only a few. Another geometric problem in this class, which we consider in this paper, is determination of the non-trivial circuits of a union $\mathcal{U}$ of iso-oriented rectangles. This is the following problem.

2. Computing the non-trivial circuits of a union $\mathcal{U}$

First of all, we need some notation. Let $R_1, \ldots, R_n$ be rectangles on the plane with sides parallel to the coordinate axes, where E be the set of their segments ($|E| = 4n$) and V the set of their vertices ($|V| = 4n$). We consider each edge of E as being lined by two *arcs* with opposite directions and lying on either side of the edge, as illustrated in Fig. 1. This means that each rectangle R_i is stored in a cyclic doubly-linked list $L(R_i)$, which represents the two orientations of R_i: a *clockwise* orientation, when marching along R_i in the *external* region of R_i (i.e., when using

E-mail address: alevizos@math.upatras.gr.

0020-0190/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ipl.2012.11.002

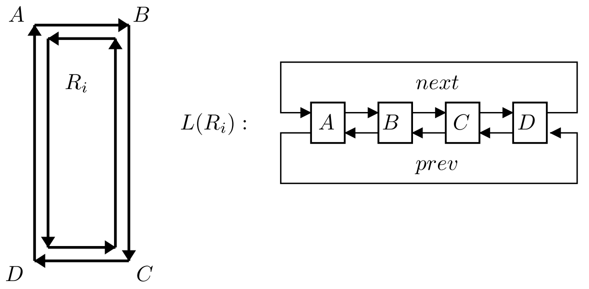

Fig. 1. The doubly-linked cyclic list $L(R_i)$ associated to a rectangle R_i. Each rectangle is supplied with four external and four internal arcs.

the pointers *next* of $L(R_i)$), and a *counterclockwise* orientation, when marching along R_i in the *internal* region of R_i (i.e., when using the pointers *prev* of $L(R_i)$). Following the above convention, we say that each rectangle R_i is supplied with four *external* and four *internal arcs*. The union $\mathcal{U} = R_1 \cup \cdots \cup R_n$ determines a subdivision of the plane into regions. Each region is bounded from a directed *circuit* (cycle) composed of (alternating) vertical and horizontal (external and/or internal) arcs. An arc is called *non-trivial* if it contains an endpoint of the segment to which it belongs. A circuit is called *non-trivial* if it contains a non-trivial arc (Fig. 2). A circuit is said to be *contour circuit* if it consists of external arcs only. The set of all – trivial and non-trivial – contour circuits is called the *contour* of $\mathcal{U}$ (Fig. 3). The contour of $\mathcal{U}$ has $O(n^2)$ arcs. In [17], Widmayer and Wood have presented an optimal algorithm, which determines the contour of a union $\mathcal{U}$ in $O(n \log n + p)$ time and $O(n)$ space, where p is the number of arcs in the contour. The *non-trivial contour* is the collection of all non-trivial contour circuits in $\mathcal{U}$. The *external contour* of $\mathcal{U}$ is the boundary between $\mathcal{U}$ and the unbounded region of the plane. The external contour is the unique clockwise oriented non-trivial circuit. Any other non-trivial circuit is counterclockwise oriented. It is straightforward to show that

$$\begin{aligned}\text{External contour} &\subseteq \text{Nontrivial contour}\\ &\subseteq \text{Nontrivial circuits}.\end{aligned}$$

The number of arcs in the non-trivial circuits is $O(n)$, while the set of all circuits of $\mathcal{U}$ has $O(n^2)$ arcs, the increase being due to trivial circuits [12]. Lipski and Preparata [1] have presented a time-optimal algorithm for computing the non-trivial circuits of a union $\mathcal{U}$ in $\Theta(n \log n)$ time and $O(n \log n)$ space. In this paper, we will describe a simple algorithm, which determines the non-trivial circuits, the non-trivial contour and the external contour of $\mathcal{U}$ in $\Theta(n \log n)$ time and $\Theta(n)$ space.

2.1. The algorithm

Our algorithm is based on the following simple idea proved in Theorem 1: "each (trivial or non-trivial) arc of a non-trivial circuit C_i is (horizontally or vertically) visible from an endpoint $p \in V$ of a non-trivial arc of C_i":

Step 1: For each point $p \in V$, we find at most four visibility points (Fig. 4(a)): From p we trace two horizontal

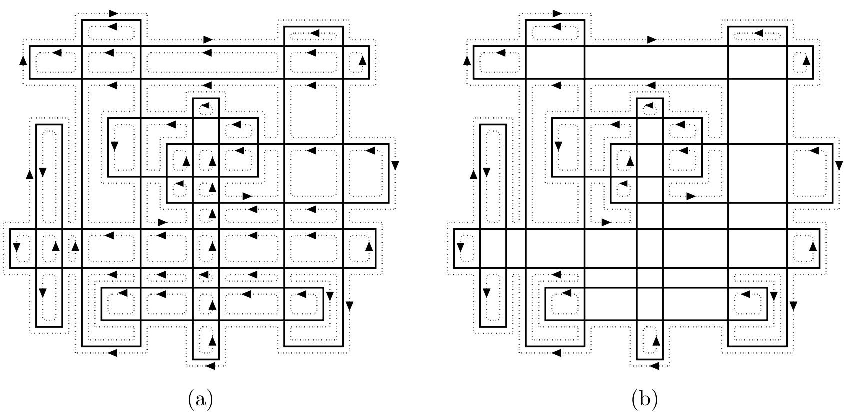

Fig. 2. (a) The set of all circuits and (b) the set of non-trivial circuits, for a union of rectangles.

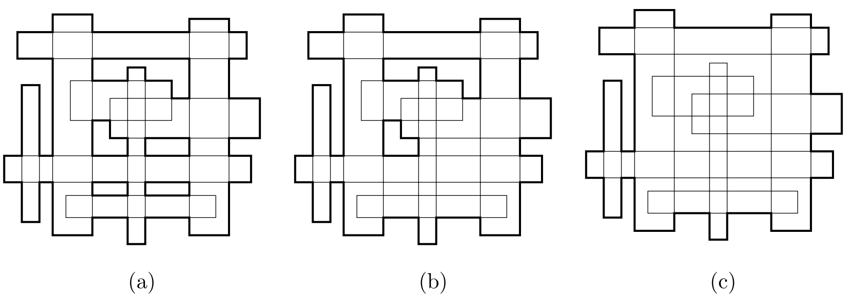

Fig. 3. (a) The contour, (b) the non-trivial contour and (c) the external contour of a union of rectangles.