On Finding the Rectangular Duals of Planar Triangular Graphs

Xin He [1]

Department of Computer Science

State University of New York at Buffalo

Buffalo, NY 14260

E-mail: xinhe@cs.buffalo.edu

Abstract

We present a new linear time algorithm for finding rectangular duals of planar triangular graphs. The algorithm is conceptually simpler than the previous known algorithm. The coordinates of the rectangular dual constructed by our algorithm are integers and have pure combinatorial meaning. This allows us to discuss the heuristics for minimizing the size of the rectangular duals.

Key words: Algorithm, Planar graph, Rectangular dual.

AMS(MOS) subject classifications: 05C05, 05C38, 68Q25, 68R10.

1. Introduction

Let R be a rectangle. A *rectangular subdivision system* of R is a partition of R into a set $\Phi = \{R_1, R_2, \ldots, R_n\}$ of non-intersecting smaller rectangles such that no four rectangles in Φ meet at the same point. A *rectangular dual* of a graph $G = (V, E)$ is a rectangular subdivision system Φ and a one-to-one correspondence $f : V \to \Phi$ such that two vertices u and v are adjacent in G if and only if their corresponding rectangles $f(u)$ and $f(v)$ share a common boundary. Figures 1.1 and 1.2 show a graph G and its rectangular dual. If G has a rectangular dual, clearly G must be a planar graph.

The rectangular dual of a graph G finds applications in the floor planning of electronic chips and in architectural design [5, 9]. Each vertex of G represents a circuit module and the edges represent module adjacencies. A rectangular dual provides a placement of the circuit modules that preserves the required adjacencies.

The problem of finding rectangular duals has been studied in [2, 3, 6, 8, 12]. A linear time algorithm for this problem was given in [3]. This algorithm is rather complicated and requires real arithmetic for the coordinates of the rectangular dual. We present a new linear time algorithm for solving this problem. The coordinates of the rectangular dual R constructed by our algorithm are

[1] Research partially supported by NSF grant CCR-9011214.

integers and have pure combinatorial meaning. This allows us to discuss heuristics for reducing the size of R.

The present paper is organized as follows. Sections 2 introduces some definitions and lemmas needed by our algorithm. Section 3 presents the algorithm. Section 4 proves its correctness. In section 5, we discuss the heuristics for reducing the size of the rectangular dual. Section 6 concludes the paper.

2. Regular Edge Labeling of Planar Triangular Graphs

Let $G = (V, E)$ be a planar graph. Consider a fixed plane embedding of G. The embedding divides the plane into a number of regions. The unbounded region is called the *exterior face*. Other regions are called *interior faces*. The vertices and the edges on the exterior face are called *exterior vertices* and *exterior edges*, respectively. A cycle C of G divides the plane into its interior region and exterior region. If C contains at least one vertex in its interior region, C is called a *separating cycle* of G. For each vertex v, $N(v)$ denotes the set of neighbors of v and $Star(v)$ denotes the set of edges incident to v. Whenever these notations are used, it is understood that the members in each set are listed in counterclockwise order around v in the embedding.

Consider a planar graph $H = (V, E)$. Let v_0, v_1, v_2, v_4 be four vertices on the exterior face of H in counterclockwise order. Let P_i $(i = 0, 1, 2, 3)$ be the four paths on the exterior face of H consisting of the vertices between v_i and v_{i+1} (where the addition is mod 4). We seek a rectangular dual R_H of H such that the four vertices v_0, v_1, v_2, v_3 correspond to the four corner rectangles of R_H and the vertices on P_0 (P_1, P_2, P_3, respectively) correspond to the rectangles located along the north (west, south, east, respectively) boundary of R_H. Necessary and sufficient conditions for testing if H has a rectangular dual were discussed in [2, 3, 6]. These conditions, however, can be easily reduced to the following simpler form.

In order to simplify the problem, we modify H as follows: Add four new vertices v_N, v_W, v_S, v_E and connect v_N (v_W, v_S, v_E, respectively) to every vertex on P_0 (P_1, P_2, P_3, respectively). Then add four new edges $(v_N, v_W), (v_W, v_S), (v_S, v_E), (v_E, v_N)$. Let G be the resulting graph. It is easy to see that H has a rectangular dual R_H with v_0, v_1, v_2, v_3 corresponding to the four corner rectangles of R_H if and only if G has a rectangular dual R with exactly four rectangles on the boundary of R. Without loss of generality, we will only discuss planar graphs with exactly four vertices on its exterior face.

If G has a rectangular dual R, then every face of G, except the exterior face, must be a triangle (since no four rectangles of R meet at the same point). Moreover, since at least four rectangles are needed to fully enclose some non-empty area on the plane, any separating cycle of G must have length at least 4. The following theorem states that these two conditions are also sufficient for G to have a rectangular dual.

Theorem 1 [6]: A planar graph $G = (V, E)$ has a rectangular dual R with four rectangles on the boundary of R if and only if the following two conditions hold: (1) Every interior face of G is a triangle and the exterior face of G is a quadrangle; (2) G has no separating triangles.

A different form of Theorem 1 was given in [2, 3]. A graph satisfying the two conditions of Theorem 1 is called a *proper triangular planar* (PTP) graph. From now on, we only discuss such graphs.

Definition 1: A *regular edge labeling* (REL) of a PTP graph $G = (V, E)$ is a partition of the interior edges of G into two subsets $\{T_1, T_2\}$ of directed edges such that the following hold:

(1) For each interior vertex v, the edges in $Star(v)$ appear in counterclockwise order around v as follows: a set of edges in T_1 leaving v; a set of edges in T_2 entering v; a set of edges in T_1 entering v; a set of edges in T_2 leaving v.

(2) All interior edges incident to v_N are in T_1 and entering v_N. All interior edges incident to v_W are in T_2 and leaving v_W. All interior edges incident to v_S are in T_1 and leaving v_S. All interior edges incident to v_E are in T_2 and entering v_E.

From Theorem 1, we can easily prove the following:

Theorem 2: Every PTP graph $G = (V, E)$ has a REL.

Proof: By Theorem 1, G has a rectangular dual R. For each $v \in V$, let $R(v)$ denote the rectangle in R corresponding to v. For each interior vertex v, label each edge $(v, u) \in Star(v)$ as follows: If $R(u)$ is above $R(v)$, e is in T_1 and directed leaving v. If $R(u)$ is below $R(v)$, e is in T_1 and directed entering v. If $R(u)$ is to the left of $R(v)$, e is in T_2 and directed entering v. If $R(u)$ is to the right of $R(v)$, e is in T_2 and directed leaving v. This labeling satisfies the two conditions of Definition 1. □

Although Theorem 2 is proved from Theorem 1, our algorithm goes another way around: We find a REL of G first and construct a rectangular dual of G from the REL. We first prove some properties of the REL.

Let $G = (V, E)$ be a PTP graph and $\{T_1, T_2\}$ be a REL of G. Let G_1 be the directed subgraph of G induced by the edges in T_1 and the four exterior edges directed as $v_S \to v_W$; $v_W \to v_N$; $v_S \to v_E$; $v_E \to v_N$. Let E_1 denote the edge set of G_1. (E_1 is the union of T_1 and the four exterior edges.) Let G_2 be the directed subgraph of G induced by the edges in T_2 and the four exterior edges directed as $v_W \to v_S$; $v_S \to v_E$; $v_W \to v_N$; $v_N \to v_E$. Let E_2 denote the edge set of G_2. (E_2 is the union of

T_2 and the four exterior edges.) We will call G_1 the *S-N net* and G_2 the *W-E net* of G derived from the REL $\{T_1, T_2\}$.

Figure 1.1 shows a PTP graph G. An S-N net G_1 and the corresponding W-E net G_2 are shown in Figures 1.3 and 1.4. (Ignore the integers in the small boxes in Figures 1.3 and 1.4 for now.)

Figure 1

Lemma 3: (1) G_1 is acyclic with v_S as the only source and v_N as the only sink.

(2) G_2 is acyclic with v_W as the only source and v_E as the only sink.

Proof: By way of contradiction. Suppose either G_1 or G_2 contains a directed cycle. Let $C = \{v_1, \ldots v_l\}$ be such a cycle such that the total number of vertices that are on C or in the interior of C is minimized. Without loss of generality, suppose C is a cycle in G_1 and directed in clockwise direction. (The proof of other cases are similar.)

Case 1: C contains no vertices in its interior. If $l = 3$, then there is no edge in T_2 leaving v_2. This contradicts the condition (1) of Definition 1. Suppose $l > 3$. Then there is an edge $e = (v_i, v_j) \in E$ contained in the interior of C. e cannot be in T_1 since otherwise we would have a smaller cycle in G_1 which contradicts the choice of C. So e must be in T_2. But regardless of the direction of e in T_2, the condition (1) of Definition 1 is violated either at v_i or at v_j.

Case 2: C contains at least one vertex u_1 in its interior. Start at u_1, we can reach another vertex u_2 by using a T_2 edge. Similarly from u_2 we can reach another vertex u_3 by using a T_2 edge. Since every vertex u has an incident edge in T_2 leaving u, this process can be repeated again and again. Since C is the smallest cycle in both G_1 and G_2, we cannot have a cycle in G_2 completely contained in the interior of C. Thus we must reach a vertex $v_j \in C$. Then the condition (1) of Definition 1 is violated at v_j.

Since we get contradictions in all cases, both G_1 and G_2 are acyclic. Since every vertex v, other than v_S and v_N, has indegree and outdegree at least 1 in G_1, v_S is the only source and v_N is the only sink of G_1. Similarly, v_W is the only source and v_E is the only sink of G_2. □

Both G_1 and G_2 are the so-called *s-t planar graphs*. (An s-t planar graph is a directed acyclic planar graph with exactly one source s and exactly one sink t on its exterior face.) The properties of these graphs have been studied in [7, 10, 13]. Using these properties, the structure of G_1 can be summarized as follows:

(a) For each vertex v other than v_S and v_N, the edges entering v appear consecutively around v in G_1. The edges leaving v appear consecutively around v in G_1. Let e_1 and e_2 be the left-most and the right-most edges in G_1 entering v. Let e_3 and e_4 be the left-most and the right-most edges in

G_1 leaving v. The face of G_1 with e_1 and e_3 on its boundary is denoted by $left(v)$. The face of G_1 with e_2 and e_4 on its boundary is denoted by $right(v)$. We use f_W to denote $left(v_W)$ and f_E to denote $right(v_E)$. (In other words, the exterior face is divided into two faces f_W and f_E.) For the vertex v_S and v_N, define $left(v_S) = left(v_N) = f_W$ and $right(v_S) = right(v_N) = f_E$.

(b) For each interior face f of G_1, the boundary of f consists of two directed paths P_1 and P_2 starting at the same vertex and ending at the same vertex. (See Figures 1.3).

Similarly, the structure of G_2 can be summarized as follows.

(c) For each vertex v other than v_W and v_E, the edges entering v appear consecutively around v in G_2. The edges leaving v appear consecutively around v in G_2. Let e_1 and e_2 be the left-most and the right-most edges in G_2 entering v. Let e_3 and e_4 be the left-most and the right-most edges in G_2 leaving v. The face of G_2 with e_1 and e_3 on its boundary is denoted by $above(v)$. The face of G_2 with e_2 and e_4 on its boundary is denoted by $below(v)$. We use f_N to denote $above(v_N)$ and f_S to denote $below(v_S)$. (In other words, the exterior face is divided into two faces f_N and f_S.) For the vertex v_W and v_E, define $above(v_W) = above(v_E) = f_N$ and $below(v_W) = below(v_E) = f_S$.

(d) For each interior face g of G_2, the boundary of g consists of two directed paths P_1 and P_2 starting at the same vertex and ending at the same vertex. (See Figures 1.4).

3. Algorithm

Let $G = (V, E)$ be a PTP graph and $\{T_1, T_2\}$ be a REL of G. Consider the S-N net G_1 derived from $\{T_1, T_2\}$. For each edge $e \in E_1$, let $left(e)$ ($right(e)$, respectively) denote the face of G_1 on the left (right, respectively) of e. Define the *dual graph*, denoted by G_1^*, of G_1 as follows. The node set of G_1^* is the set of the interior faces of G_1 plus the two exterior faces f_W and f_E. For each edge $e \in E_1$, there is a corresponding arc e^* in G_1^* directed from the face $left(e)$ to the face $right(e)$. Since G_1 is an s-t graph, G_1^* is also an s-t graph [11]. Namely G_1^* is a directed acyclic planar graph with f_W as the only source and f_E as the only sink.

Similarly, define the dual graph G_2^* of G_2 as follows. For each edge $e \in E_2$, let $above(e)$ ($below(e)$, respectively) denote the face of G_2 on the left (right, respectively) of e. The nodes of G_2^* are the interior faces of G_2 plus the two exterior faces f_S and f_N. For each edge $e \in E_2$, there is a directed arc e^* in G_2^* from the face $below(e)$ to the face $above(e)$. G_2^* is a directed acyclic planar graph with f_S as the only source and f_N as the only sink.

Definition 2: A *consistent numbering of order* k_1 of G_1^* is a surjective mapping F_1 from the node set of G_1^* to the set of integers $\{0, 1 \ldots, k_1\}$ such that: (1) $F_1(f_W) = 0$ and $F_1(f_E) = k_1$; and (2) if there is an arc from the node f to the node g in G_1^* then $F_1(f) < F_1(g)$.

For an example, a topological ordering [1, 4] of G_1^* is a consistent numbering. As another example, if we define $F_1(f)$ to be the length of the longest path in G_1^* from f_W to f (with $F_1(f_W) = 0$), F_1 is also a consistent numbering. Define the *length* of G_1^* to be the the length of the longest path from f_W to f_E in G_1^*. Note that if the length of G_1^* is k, then any consistent numbering of G has order at least k by Definition 2. The consistent numbering of G_2^* can be defined similarly. We now can present our algorithm as follows.

Algorithm DUAL:

Input: A PTP graph $G = (V, E)$.

(1) Find a REL $\{T_1, T_2\}$ of G.

(2a) Construct the S-N net G_1 derived from $\{T_1, T_2\}$ and its dual graph G_1^*.

(2b) Compute a consistent numbering F_1 of G_1^*. Let $k_1 = F_1(f_E)$.

(2c) For each vertex $v \in V$ other than v_S and v_N, let $f_1 = left(v)$ and $f_2 = right(v)$ in G_1. Let $x_1(v) = F_1(f_1)$ and $x_2(v) = F_1(f_2)$. Define $x_1(v_N) = x_1(v_S) = 1$ and $x_2(v_N) = x_2(v_S) = k_1 - 1$.

(3a) Construct the W-E net G_2 derived from $\{T_1, T_2\}$ and its dual graph G_2^*.

(3b) Compute a consistent numbering F_2 of G_2^*. Let $k_2 = F_2(f_N)$.

(3c) For each vertex $v \in V$, let $g_1 = below(v)$ and $g_2 = above(v)$ in G_2. Let $y_1(v) = F_2(g_1)$ and $y_2(v) = F_2(g_2)$.

(4) For each vertex $v \in V$, assign v a rectangle $R(v)$ bounded by two vertical lines with x-coordinates $x_1(v)$, $x_2(v)$ and two horizontal lines with y-coordinates $y_1(v)$, $y_2(v)$.

End.

In section 4, we will prove the algorithm DUAL correctly computes a $k_1 \times k_2$ rectangular dual of G. For an example, the rectangular dual shown in Figure 1.2 is constructed from the information indicated in Figures 1.3 and 1.4. In this example, $F_1(f)$ is the length of the longest path from f_W to f in G_1^*. $F_2(g)$ is the length of the longest path from f_S to g in G_2^*. In Figure 1.3, the integers in the small boxes are the F_1-numbers of the faces of G_1. In Figure 1.4, the numbers in the small boxes are the F_2-numbers of the faces of G_2.

To implement the algorithm DUAL, we assume the embedding of G is given. (If not, it can be computed by using the well known linear time planarity algorithms.) Step 1 can be carried out by using the $O(n)$ algorithm in [3]. (The algorithm in [3] finds the set T_1 which is called the *path digraph*). For Step (2a), the graph G_1 and the dual graph G_1^* can be constructed from the embedding information of G. The implementation of Step (2b) depends on the choice of F_1. The most natural choice, the length of the longest path from f_S to f in G_1^*, can be calculated according to the topological ordering of G_1^* [1, 4]. For Step (2c), the left face and the right face of each vertex can be determined from the embedding information. All these steps take $O(n)$ time. Step (3) can be implemented similarly. Step (4) clearly takes $O(n)$ time. Thus the total running time of the algorithm is $O(n)$.

4. Correctness Proof

Before we prove the correctness of the algorithm DUAL, we need several definitions. Consider an S-N net G_1 of G. An *S-N path* is a directed path P in G_1 from v_S to v_N. Let P_1 and P_2 be two S-N paths of G_1. (P_1 and P_2 are not necessarily edge disjoint.) We say P_2 is *to the right of* P_1 if every edge $e \in P_2$ is either on P_1 or to the right of P_1.

Definition 3: A *path system* of G_1 is a collection $\{P_0, \ldots, P_{l-1}\}$ of S-N paths of G_1 such that:
(1) The union of the paths P_i $(0 \leq i \leq l-1)$ is the edge set E_1 of G_1.
(2) P_i is to the right of P_{i-1} for $1 \leq i \leq l-1$.

Definition 4: Let F_1 be a consistent numbering of G_1^* of order k_1. For each $0 \leq i \leq k_1$, define:
(1) $FACE_i = \{f \mid f$ is a face of G_1 with $F_1(f) = i\}$.
(2) $LB_i = \{e \in E_1 \mid e$ is on the left boundary of a face $f \in FACE_i\}$.
(3) $RB_i = \{e \in E_1 \mid e$ is on the right boundary of a face $f \in FACE_i\}$.
(4) Define the *standard path system* $\{P_0, \ldots, P_{k_1-1}\}$ of G_1 as follows:

$P_0 = RB_0$; and $P_i = P_{i-1} - LB_i \cup RB_i$ for $1 \leq i \leq k_1 - 1$.

Note that: $FACE_0 = \{f_W\}$, $LB_0 = \emptyset$, $RB_0 = \{(v_S, v_W), (v_W, v_N)\}$. $FACE_{k_1} = \{f_E\}$, $LB_{k_1} = \{(v_S, v_E), (v_E, v_N)\}$, $RB_{k_1} = \emptyset$.

We make the following observations. Consider any edge $e \in E_1$. Let $g_1 = left(e)$, $g_2 = right(e)$, $p = F_1(g_1)$, $q = F_1(g_2)$. Since e is on the right boundary of g_1 and on the left boundary of g_2, $e \in RB_p$ and $e \in LB_q$. Since e's corresponding arc e^* is directed from g_1 to g_2 in G_1^*, we have $p < q$. So $LB_i \cap RB_i = \emptyset$ for all $0 \leq i \leq k_1$. Since each $e \in E_1$ is in exactly one RB_i $(0 \leq i \leq k_1 - 1)$, E_1 is the disjoint union of the sets RB_i $(0 \leq i \leq k_1 - 1)$. Similarly E_1 is the disjoint union of the sets LB_i $(1 \leq i \leq k_1)$.

Lemma 4: Let F_1 be a consistent numbering of G_1^* of order k_1. Then

(a) The standard path system $\{P_0, P_1, \ldots, P_{k_1-1}\}$ in Definition 4 is a path system of G_1.

(b) For each vertex $v \in V$, let $f_1 = left(v)$ and $f_2 = right(v)$ in G_1. Define $x_1(v) = F_1(f_1)$ and $x_2(v) = F_1(f_2)$. Then v is on the path P_i if and only if $x_1(v) \leq i \leq x_2(v) - 1$.

Proof: (a) We prove, by induction, the following hold for each i ($0 \leq i \leq k_1 - 1$): (1) P_i is an S-N path of G_1; and (2) $LB_{i+1} \subseteq P_i$.

Base step $i = 0$: (1) $P_0 = \{(v_S, v_W), (v_W, v_N)\}$ is an S-N path of G_1.

(2) Let e be an edge in LB_1. Then e is on the left boundary of a face $f \in FACE_1$. Let e^* be the arc in G_1^* corresponding to e. Since $F_1(f) = 1$, e^* must be directed from f_E to f in G_1^*. This implies $e \in RB_0 = P_0$. Since this is true for all $e \in LB_1$, we have $LB_1 \subseteq P_0$.

Induction step: Assume the claims (1) and (2) are true for $i - 1$, we show they are true for i.

(1) By induction hypothesis, P_{i-1} is an S-N path. Suppose $FACE_i = \{h_1, \ldots, h_l\}$ for some l. Let A_j and B_j be the left and the right boundary of h_j respectively ($1 \leq j \leq l$). Since (2) is true for P_{i-1}, the paths A_j ($1 \leq j \leq l$) are sub-paths of P_{i-1}. Since A_j and B_j ($1 \leq j \leq l$) start at the same vertex and end at the same vertex and P_i is obtained from P_{i-1} by replacing each A_j with B_j, P_i is an S-N path of G_1.

(2) Consider any edge $e \in LB_{i+1}$. Let $g_1 = left(e)$ and $g_2 = right(e)$. Since $e \in LB_{i+1}$, $F_1(g_2) = i + 1$. Suppose $F_1(g_1) = q$ for some q. Then $e \in RB_q$. Since e^* is directed from g_1 to g_2 in G_1^*, $q < i + 1$. By definition, e is added into P_q and deleted when P_{i+1} is constructed. So e is in P_r for all $q \leq r \leq i$. In particular $e \in P_i$. Thus $LB_{i+1} \subseteq P_i$. This completes the induction.

Each $e \in E_1$ is in some RB_i ($0 \leq i \leq k_1 - 1$) and hence in P_i. Therefore E_1 is the union of P_i's ($i = 0, \ldots k_1 - 1$). From the definition of P_i, it is easy to see P_i is to the right of P_{i-1} for all $1 \leq i \leq k_1 - 1$. Thus $\{P_0, \ldots, P_{k_1-1}\}$ is a path system of G_1.

(b) Since v is on the right boundary of f_1, it is added into the path $P_{x_1(v)}$. Since v is on the left boundary of f_2, it is removed when the path $P_{x_2(v)}$ is constructed. Hence v is on the paths P_i for exactly those indices i with $x_1(v) \leq i \leq x_2(v) - 1$. $\Box$

All above discussion can be repeated on the W-E net G_2 and its dual graph G_2^*. Let F_2 be a consistent numbering of G_2^* of order k_2. We can construct the standard path system $\{Q_0, \ldots, Q_{k_2-1}\}$ of G_2 from F_2 similar to Definition 4. For each vertex v of G, let $g_1 = below(v)$ and $g_2 = above(v)$ in G_2. Define $y_1(v) = F_2(g_1)$ and $y_2(v) = F_2(g_2)$. Similar to Lemma 4, it can be shown that v is on the path Q_j if and only if $y_1(v) \leq j \leq y_2(v) - 1$.

Lemma 5: Let G_1 and G_2 be the S-N net and the W-E net derived from a REL $\{T_1, T_2\}$ of G. Let F_1 and F_2 be two consistent numberings of G_1^* and G_2^*, respectively. Let u and v be two vertices of G.

(1) If $(u, v) \in T_2$ and is directed from u to v in G_2, then $x_2(u) = x_1(v)$.

(2) If there is a directed path from u to v in G_2 with length at least 2, then $x_2(u) < x_1(v)$.

(3) If $(u, v) \in T_1$ and is directed from u to v in G_1, then $y_2(u) = y_1(v)$.

(4) If there is a directed path from u to v in G_1 with length at least 2, then $y_2(u) < y_1(v)$.

Proof: We only prove (1) and (2). The proof of (3) and (4) is similar.

(1) Suppose $(u, v) \in T_2$ and is directed from u to v. Let e_1 (e_2, respectively) be the rightmost outgoing (incoming, respectively) edge of u in G_1. Let e_3 (e_4, respectively) be the leftmost outgoing (incoming, respectively) edge of v in G_1. Let f be the face of G_1 with e_1, e_2, e_3, and e_4 on its boundary. Then $f = right(u) = left(v)$ and $x_2(u) = x_1(v) = F_1(f)$.

(2) Let $u = u_0, u_1, \ldots, u_p = v$ ($p \geq 2$) be a directed path in G_2 from u to v. By (1), $x_2(u_{l-1}) = x_1(u_l)$ for all $1 \leq l \leq p$. Since $x_1(u_l) < x_2(u_l)$ for all $0 \leq l \leq p$ and $p \geq 2$, we have $x_2(u) = x_2(u_0) < x_1(u_p) = x_1(v)$. □

From above two lemmas, we can prove the following:

Theorem 6: The algorithm DUAL correctly constructs a rectangular dual of G in $O(n)$ time.

Proof: We have shown the algorithm can be implemented in linear time. We next prove the correctness of the algorithm. Let $\{P_0, \ldots, P_{k_1-1}\}$ be the standard path system of G_1 derived from F_1 and let $\{Q_0, \ldots, Q_{k_2-1}\}$ be the standard path system of G_2 derived from F_2. In the rectangular dual R constructed by the algorithm DUAL, each S-N path P_i ($0 \leq i \leq k_1 - 1$) corresponds to a vertical strip bounded by the two vertical lines with x-coordinates i and $i + 1$. Each W-E path Q_j ($0 \leq j \leq k_2 - 1$) corresponds to a horizontal strip bounded by the two horizontal lines with y-coordinates j and $j + 1$. Let $R(v)$ be the rectangle with coordinates $x_1(v), x_2(v), y_1(v), y_2(v)$. To show the set $\{R(v) \mid v \in V\}$ forms a rectangular dual of G, we need to prove the following claims.

(1) We show that each unit square R_{ij} ($0 \leq i \leq k_1 - 1$ and $0 \leq j \leq k_2 - 1$) with x-coordinates i, $i+1$ and y-coordinates $j, j+1$ is occupied by a rectangle $R(v)$ for a unique $v \in V$. Consider the S-N path P_i and the W-E path Q_j. Except the four special cases (a) $i = 0, j = 0$ (b) $i = k_1 - 1, j = 0$ (c) $i = 0, j = k_2 - 1$ (d) $i = k_1 - 1, j = k_2 - 1$, P_i and Q_j intersect at a unique vertex $v \in V$. By Lemma 4 (b), v is the unique vertex satisfying all of the following inequalities: $x_1(v) \leq i$, $i + 1 \leq x_2(v)$, $y_1(v) \leq j$, $j + 1 \leq y_2(v)$. Hence $R(v)$ is the unique rectangle occupying R_{ij}. For the four special cases, this claim is not true. (For example, both v_S and v_W belong to the intersection of P_0 and Q_0.) The four special cases correspond to the four corner unit squares of R. However, the special definition $x_1(v_S) = x_1(v_N) = 1$ and $x_2(v_S) = x_2(v_N) = k_1 - 1$ at the Step (2b) of the algorithm DUAL ensures that each of the four unit corner squares of R is occupied by one of $R(v_W), R(v_E)$.

(2) We show if $e = (u, v)$ is an edge in G, then the corresponding rectangles $R(u)$ and $R(v)$ share a common boundary. If e is an exterior edge, this is ensured by the definition of $R(v_N)$, $R(v_W)$, $R(v_S)$, $R(v_E)$. So assume e is an interior edge. Suppose $e \in T_1$ and is directed from u to v. (Other cases are similar). Let P_i be an S-N path containing e. By Lemma 4 (b), $x_1(u) \leq i \leq x_2(u) - 1$ and $x_1(v) \leq i \leq x_2(v) - 1$. By Lemma 5 (3), $y_2(u) = y_1(v) = j$ for some j. Thus $R(u)$ and $R(v)$ have the line segment connecting two points (i, j) and $(i + 1, j)$ as their common boundary.

(3) We show if two rectangles $R(u)$ and $R(v)$ share a common boundary, then (u, v) is an edge in G. Assume the common boundary of $R(u)$ and $R(v)$ contains a horizontal line segment I connecting two points (i, j) and $(i + 1, j)$. (Other cases are similar.) Since $x_1(u) \leq i$, $i + 1 \leq x_2(u)$ and $x_1(v) \leq i$, $i + 1 \leq x_2(v)$, both u and v are on the S-N path P_i. We need to show (u, v) is an edge on P_i. If not, there exists a directed path from u to v in G_1 of length at least 2. By Lemma 5 (4), we have $y_2(u) < y_1(v)$. This contradicts the assumption that $R(u)$ and $R(v)$ share I as their common boundary.

Thus $e = (u, v)$ is an edge of G if and only if $R(u)$ and $R(v)$ share a common boundary. Hence $\{R(v) \mid v \in V\}$ form a rectangular dual of G. □

5. Heuristics for Reducing the Size of the Rectangular Dual

The rectangular dual produced by the algorithm DUAL has size $k_1 \times k_2$, where k_i $(i = 1, 2)$ is the order of the consistent numbering of F_i of G_i^*. As mentioned earlier, if k_1 is the length of the longest path from f_W to f_E in G_1^* and k_2 is the length of the longest path from f_S to f_N in G_2^*, then any consistent numbering of G_1^* has order at least k_1 and any consistent numbering of G_2^* has order at least k_2. Thus the size of the smallest rectangular dual that can be produced from a given REL $\{T_1, T_2\}$ is exactly $k_1 \times k_2$. So in order to reduce the size of the rectangular dual of G, we must try to find a good REL. In this section, we present two such heuristics: (1) Delete certain edges from G_1 to obtain another S-N net so that the corresponding rectangular dual R' has smaller width (at the cost of possible increase in the height of R'). (2) Add certain edges into G_1 to obtain another S-N net so that the corresponding rectangular dual R'' has the same width and possibly smaller height.

5.1 Reducing the width of the rectangular dual

Let G_1^* be the dual of G_1 and k_1 be the length of G_1^*. In order to reduce the width of the rectangular dual R, we must reduce k_1.

Let $e = (u, v)$ be an interior edge of G_1 directed from u to v. We say e is *redundant* if u has at least two outgoing edges and v has at least two incoming edges in G_1. It is easy to show that the directed graph $G_1 - \{e\}$, obtained from G_1 by deleting a redundant edge e, is still an S-N net of G. Let G^* denote the dual graph of $G_1 - \{e\}$. G^* can be obtained from G_1^* by removing e's corresponding arc e^* and merging the two faces of G_1 with e on their boundary into a single face. Our strategy for reducing the width of R is to keep deleting the redundant edges from G_1 until no redundant edges remain. We say the resulting S-N net is *unreducible.*

The removal of some redundant edges does reduce the length of G_1^*, while the removal of other redundant edges does not. So the order in which the redundant edges are removed is important. Let e be an redundant edge of G_1 and let $e^* = (f_1, f_2)$ be its corresponding arc in G_1^*. Suppose $p_1 = F_1(f_1)$ and $p_2 = F_1(f_2)$. We say e is *critical* if e^* is the only arc in G_1^* from a face with F_1-number p_1 to a face with F_1-number p_2. It is easy to show that the removal of a critical redundant edge e reduces the length of G_1^* by 1. Thus when deleting redundant edges from G_1, we always delete critical edges first. The following is a heuristic algorithm for this strategy. It is to be inserted into the algorithm DUAL following the step (1).

Algorithm Reduce Width:

Input: An S-N net G_1 of a PTP graph $G = (V, E)$.

Repeat:

If there is a critical redundant edge $e \in G_1$, then $G_1 \leftarrow G_1 - \{e\}$.

Else find an arbitrary redundant edge $e \in G_1$ and $G_1 \leftarrow G_1 - \{e\}$.

Until G_1 is unreducible.

This algorithm can be easily implemented in $O(n^2)$ time. Figure 2 shows an example. When performing this algorithm on the S-N net shown in Figure 1.3, two critical redundant edges (l, i) and (n, h) are deleted. At this point, the edges (j, i) and (p, j) are redundant, but none is critical. So we arbitrarily delete the edge (j, i). After this is done, the edge (p, j) becomes critical and is deleted. The resulting unreducible S-N net and the corresponding rectangular dual R' is shown in Figure 2. The width of R' is reduced from 10 to 7. The height of R' is increased from 11 to 12.

Figure 2

5.2 Reducing the height of the rectangular dual

Given an S-N net G_1, it is possible to add certain edges into G_1 without increasing the length of the dual graph G_1^*. By doing so, we hope to reduce the length of the corresponding dual graph G_2^*, and thus reduce the height of the rectangular dual R.

Two edges incident to a vertex v are *adjacent* if they are consecutive around v in the embedding. An interior edge $e = (u, v) \in G_2$ is called an *l-candidate edge* if it satisfies the following two conditions:

(a) There are at least two edges in G_2 incident to u and at least two edges in G_2 incident to v.

(b.l) e is adjacent to the rightmost outgoing edge of u in G_1 and is adjacent to the leftmost incoming edge of v in G_1.

Similarly, e is called a *r-candidate edge* if it satisfies (a) and the following condition:

(b.r) e is adjacent to the rightmost incoming edge of u in G_1 and is adjacent to the leftmost outgoing edge of v in G_1.

If we add an l-candidate edge $e = (u, v)$ into G_1 and direct it from u to v (see Figure 3.1. where the shaded lines denote the edges in G_2 and the solid lines denote the edges in G_1); or if we add a r-candidate edge $e = (u, v)$ into G_1 and direct it from v to u (Figure 3.2), it can be shown that the resulting graph G_1' is an S-N net of G.

Figure 3

We next investigate the condition under which a candidate edge can be added into G_1 without increasing the length of G_1^*. We only discuss the r-candidate edges. The condition for the l-candidate edges is similar. Let $e = (u, v)$ be a r-candidate edge and f be the face of G_1 containing e in its interior. Let P be the left boundary and Q the right boundary of f. The vertex u divides P into two paths: P_1 ends at u and P_2 starts at u. Similarly, v divides Q into two paths: Q_1 ends at v and Q_2 starts at v (Figure 3.3). Let P_1^* and P_2^* be the sets of the arcs in G_1^* corresponding to the edges in P_1 and P_2, respectively. Let Q_1^* and Q_2^* be the sets of the arcs in G_1^* corresponding to the edges in Q_1 and Q_2, respectively. An arc $e^* = (g_1, g_2)$ in G_1^* is called a *jump arc* if $F_1(g_1) < F_1(g_2) - 1$. e^* is called an *essential arc* if $F_1(g_1) = F_1(g_2) - 1$.

Let G_1' be the graph obtained by adding a r-candidate edge e into G_1. The dual graph $G_1'^*$ of G_1' can be obtained from G_1^* as follows: The face f is divided into two faces f_1 and f_2 with e as their common boundary. A new arc (f_1, f_2) from f_1 to f_2 is introduced. Each arc $(g_1, f) \in P_1^*$ is replaced by a new arc (g_1, f_1). Each arc $(g_2, f) \in P_2^*$ is replaced by a new arc (g_2, f_2). Each arc $(f, h_1) \in Q_1^*$ is replaced by a new arc (f_1, h_1). Each arc $(f, h_2) \in Q_2^*$ is replaced by a new arc (f_2, h_2) (Figure 3.3.)

Let p be the F_1-number of the face f in G_1^*. Let p_1 and p_2 be the F_1-numbers of the faces f_1 and f_2 in $G_1'^*$, respectively. If all arcs in P_1^* are jump arcs in G_1^*, then $p_1 < p$ and $p_2 = p$. The F_1-number of any other face $g \neq f$ in G_1^* remains unchanged in $G_1'^*$. Thus the length of $G_1'^*$ is the same as the length of G_1^*. If there is an essential arc in P_1^*, then $p_1 = p$ and $p_2 = p + 1$. However, if all arcs in Q_2^* are jump arcs in G_1^*, then the F_1-number of any other face $g \neq f$ in G_1^* remains unchanged in $G_1'^*$ and the length of $G_1'^*$ is the same as the length of G_1^*. If there is an essential arc in P_1^* and an essential arc in Q_2^*, then the length of $G_1'^*$ equals the length of G_1^* plus 1.

This observation motivates the following definition: A r-candidate edge is *includable* if either all arcs in P_1^* are jump arcs or all arcs in Q_2^* are jump arcs in G_1^*. Similarly, an l-candidate edge is *includable* if either all arcs in P_2^* are jump arcs, or all arcs in Q_1^* are jump arcs in G_1^*. As discussed above, if we add an includable r- or l-candidate edge into G_1, the resulting directed graph G_1' is an S-N net of G, and the length of the corresponding dual graph $G_1'^*$ remains the same. Our strategy for reducing the height of the rectangular dual R is to keep adding includable candidate edges into G_1 until none can be found. We say the resulting S-N net is *saturated*. The following is a heuristic algorithm for this strategy. It is to be inserted into the algorithm DUAL following the step (1).

Algorithm Reduce Height:

Input: An S-N net G_1 of a PTP graph $G = (V, E)$.

Repeat:

If there is an includable l- or r-candidate edge e, then $E_1 \leftarrow E_1 \cup \{e\}$.

Until G_1 is saturated.

It is easy to implement this algorithm in $O(n^2)$ time. Figure 4 shows an example. When performing this algorithm on the S-N net shown in Figure 1.3, the includable candidate edges (s, p), (p, l), (h, f), (f, b), and (q, r) are added into G_1. (Note that the r-candidate edge (f, b) becomes includable only after the edge (h, f) has been added into G_1.) The resulting saturated S-N net and the corresponding rectangular dual R'' are shown in Figure 4. The width of R'' remains 10. The height of R'' is reduced from 11 to 10.

Figure 4

6. Conclusions

A new linear time algorithm for finding a rectangular dual R of a proper triangular planar graph G is presented. The algorithm is conceptually simple and the rectangular dual constructed has integer coordinates. The algorithm is based upon new understanding of the structure of PTP graphs, which is of independent interests. We also presented heuristics for reducing the width and the height of R. Several related optimization problems are interesting and deserve further studies. Let $w(R)$ and $h(R)$ denote the width and the height of R. How to find a rectangular dual R of G so that $w(R)$ is minimized? $w(R) + h(R)$ is minimized? or $w(R)h(R)$ is minimized?

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, The Design and Analysis of Computer Algorithms,

Addison-Wesley, Reading MA, 1974.

[2] J. Bhasker and S. Sahni, A Linear Time Algorithm to check for the Existence of a Rectangular Dual of a Planar Triangulated Graph, Networks 17, 1987, pp. 307-317.

[3] J. Bhasker and S. Sahni, A Linear Algorithm to Find a Rectangular Dual of a Planar Triangulated Graph, Algorithmica 3, 1988, pp. 247-278.

[4] E. Horowitz and S. Sahni, Fundamentals of Computer Algorithms, Computer Science Press, Potomac MD, 1988.

[5] W. R. Heller, G. Sorkin, and K. Mailing, The Planar Package Planner for System Designers, Proc. of the 19th Design Automation Conference, Las Vegas, 1982, pp. 253-260.

[6] K. Koźmiński and E. Kinnen, Rectangular Duals of Planar Graphs, Networks 15, 1985, pp. 145-157.

[7] D. Kelly and I. Rival, Planar Lattices, Canadian J. Math. 27, 1975, pp. 636-665.

[8] Y-T Lai and S. M. Leinwand, A Theory of Rectangular Dual Graphs, Algorithmica 5, 1990, pp. 467-483.

[9] K. Mailing, S. H. Mueller, and W. R. Heller, On Finding Most Optimal Rectangular Package Plans, Proc. of the 19th Design Automation Conference, Las Vegas, 1982, pp. 663-670.

[10] F. P. Preparata and R. Tamassia, Fully Dynamic Techniques for Point Location and Transitive Closure in Planar Structures, Proceedings of the 29th IEEE Symposium on Foundations of Computer Science, 1988, pp. 558-567.

[11] P. Rosenstiehl and R. E. Tarjan, Rectilinear Planar Layouts and Bipolar Orientations of Planar Graphs, Disc. Comp. Goem. 1, 1985, pp. 343-353.

[12] C. Thomassen, Interval Representations of Planar Graphs, J. of Combinatorial Theory, Series B 40, 1986, pp. 9-20.

[13] R. Tamassia and J. S. Vitter, Optimal Parallel Algorithms for Transitive Closure and Point Location in Planar Structures, Proceedings of The 1989 ACM Symposium on Parallel Algorithms and Architectures, 1989, pp. 399-407.

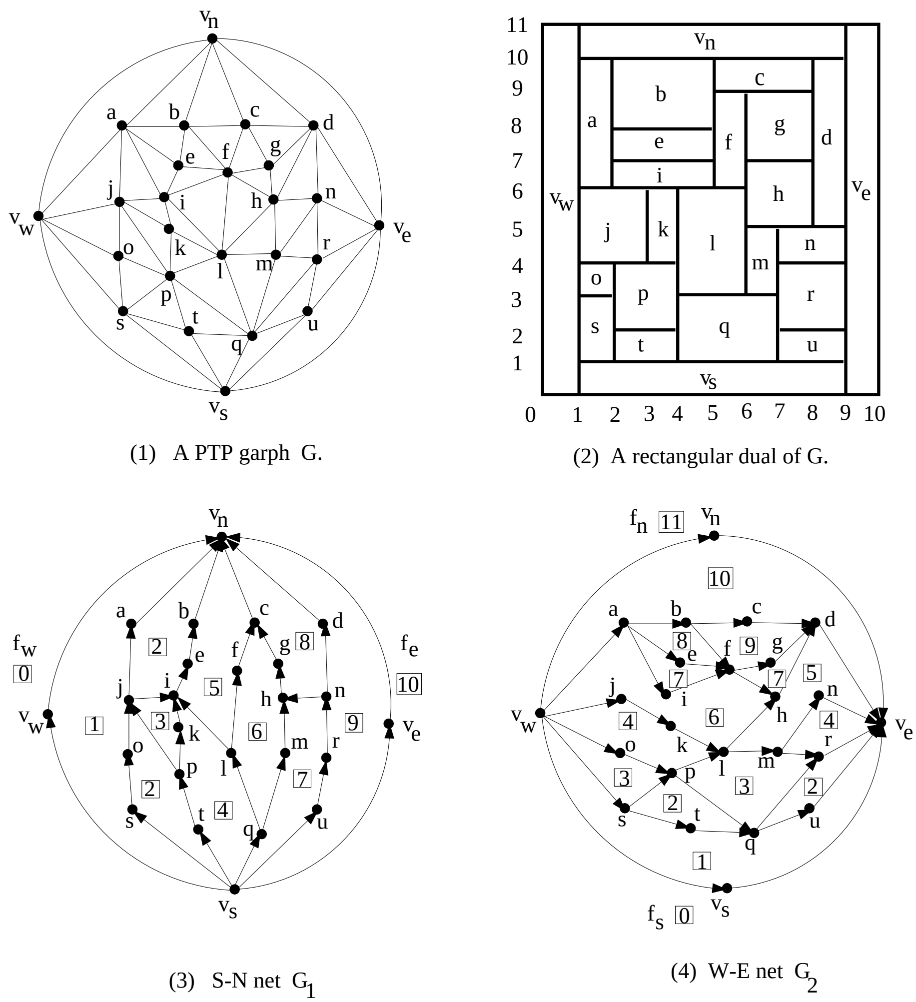


Figure 1. A PTP garph G and a rectangular dual of G.

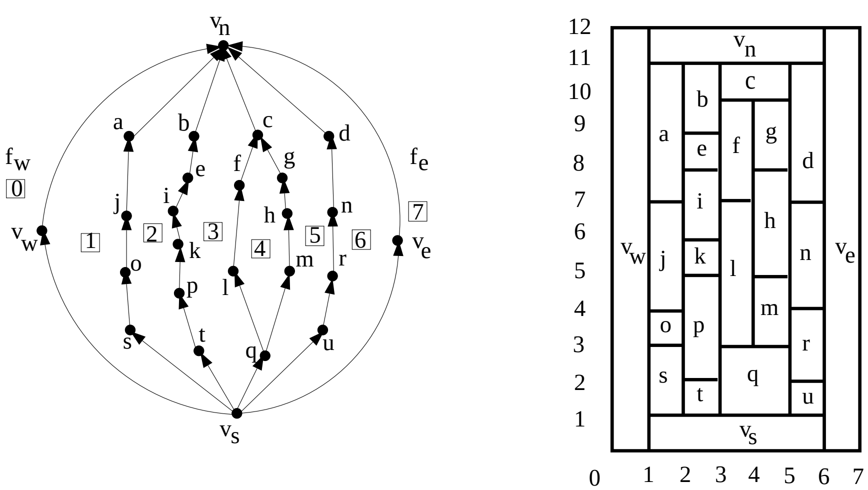
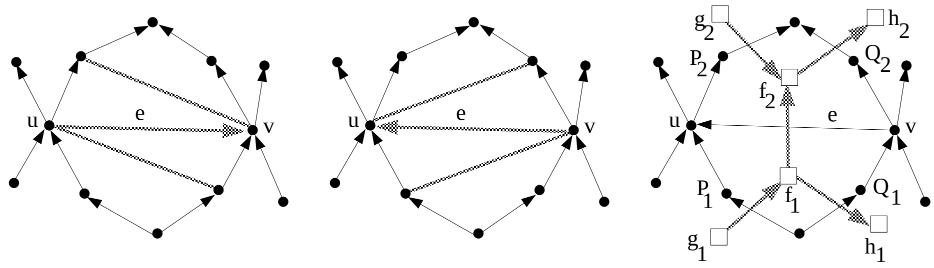


Figure 2. An unreducible S-N net and the corresponding rectangular dual.



(1) l-candidate edge (2) r-candidate edge (3) $G_1'^*$

Figure 3. Adding a candidate edge into an S-N net.

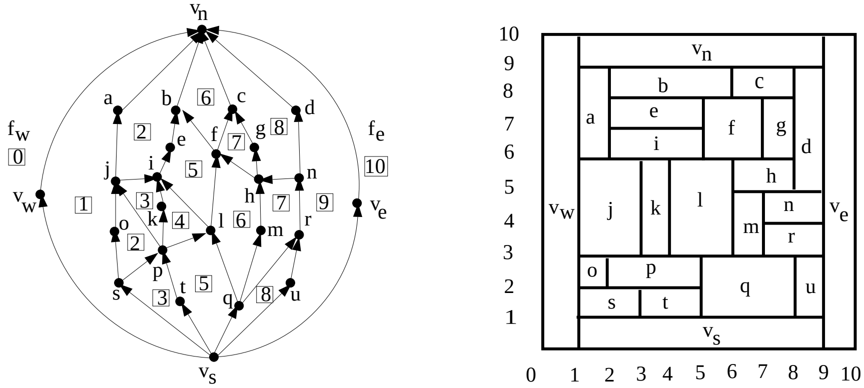


Figure 4. A saturated S-N net and the corresponding rectangular dual.