On Balanced ✚-Contact Representations

Stephane Durocher* and Debajyoti Mondal**

Department of Computer Science, University of Manitoba, Canada
{durocher,jyoti}@cs.umanitoba.ca

Abstract. In a ✚-contact representation of a planar graph G, each vertex is represented as an axis-aligned plus shape consisting of two intersecting line segments (or equivalently, four axis-aligned line segments that share a common endpoint), and two plus shapes touch if and only if their corresponding vertices are adjacent in G. Let the four line segments of a plus shape be its arms. In a c-balanced representation, $c \leq 1$, every arm can touch at most $\lceil c\Delta \rceil$ other arms, where Δ is the maximum degree of G. The widely studied T- and L-contact representations are c-balanced representations, where c could be as large as 1. In contrast, the goal in a c-balanced representation is to minimize c. Let c_k, where $k \in \{2, 3\}$, be the smallest c such that every planar k-tree has a c-balanced representation. In this paper we show that $1/4 \leq c_2 \leq 1/3 (= b_2)$ and $1/3 < c_3 \leq 1/2 (= b_3)$. Our result has several consequences. Firstly, planar k-trees admit 1-bend box-orthogonal drawings with boxes of size $\lceil b_k\Delta \rceil \times \lceil b_k\Delta \rceil$, which generalizes a result of Tayu, Nomura, and Ueno. Secondly, they admit 1-bend polyline drawings with $2\lceil b_k\Delta \rceil$ slopes, which is significantly smaller than the 2Δ upper bound established by Keszegh, Pach, and Pálvölgyi for arbitrary planar graphs.

1 Introduction

In a contact representation of a planar graph G, the vertices of G are represented using different non-overlapping geometric shapes (e.g., lines, triangles, or circles) and the adjacencies are represented by the contacts of the corresponding objects. Contact representations arise in many applied fields, such as cartography, VLSI floor-planning, and data visualization, which has motivated extensive research over the past several decades. In this paper we examine ✚-*contact representations* of planar graphs, i.e., each vertex in such a representation Γ corresponds to an axis-aligned plus shape, two plus shapes never cross, but touch if and only if their corresponding vertices are adjacent in the input planar graph. Let the four orthogonal parts associated with a plus symbol be its *left, right, up* and *down arms*. We call Γ a c-balanced representation, where $c \leq 1$, if every arm in Γ touches at most $\lceil c\Delta \rceil$ other arms, where Δ is the maximum degree of the

* Work of the author is supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

** Work of the author is supported in part by a University of Manitoba Graduate Fellowship.

S. Wismath and A. Wolff (Eds.): GD 2013, LNCS 8242, pp. 143–154, 2013.

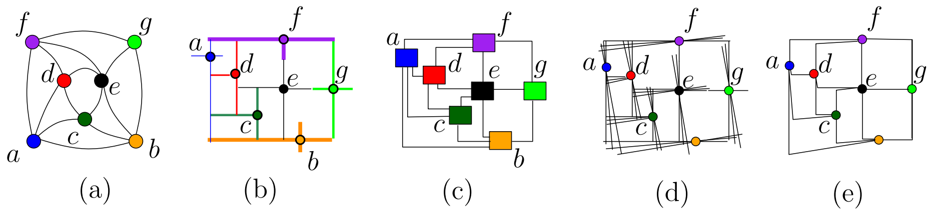

Fig. 1. (a) A graph G with $\Delta = 5$. (b) A (1/2)-balanced ✛-contact representation of G. (c) A box-orthogonal drawing of G. (d)–(e) A transformation into a polyline drawing.

underlying graph. The horizontal (or vertical) segments of two touching plus shapes in Γ may be collinear, e.g., the shapes representing the vertices e and g in Figures 1(a)–(b).

In 1994, de Fraysseix et al. [1] gave an algorithm to construct contact representations of planar graphs with axis-aligned T shapes. Many studies followed to characterize classes of planar graphs that admit contact representations with shapes simpler than T, such as axis-aligned segments [2]and L shapes [3]. L- and T-contact representations can be viewed as c-balanced ✛-contact representations, however, c may be required to be as large as 1. On the other hand, in a c-balanced representation, our goal is to minimize c.

Box-Orthogonal Drawings with Small Boxes of Constant Aspect Ratio. Balanced ✛-contact representations are useful in the study of box-orthogonal drawings in $\mathbb{R}^2$. A *k-bend box-orthogonal drawing* of a planar graph G is a planar drawing of G, where each vertex is represented as an axis-aligned box and each edge is drawn as an orthogonal polygonal chain with at most k bends. Every ✛-contact representation can be transformed into a box-orthogonal drawing [4], as shown in Figure 1(c). Some important aesthetics of a box-orthogonal drawing are the number of bends per edge, and the aspect ratio and size of the boxes. Biedl and Kaufmann [4] showed that every planar graph admits a 1-bend box-orthogonal drawing on an integer grid, but the width or height of a box in such a drawing could be as large as Δ. A c-balanced ✛-contact representation implies a 1-bend box-orthogonal drawing with boxes of size $\lceil c\Delta \rceil \times \lceil c\Delta \rceil$.

Orthogonal drawings are box-orthogonal drawings with boxes of degenerate shapes, i.e., points. The graphs that admit orthogonal drawings are of maximum degree four. Hence a 0- and 1-bend orthogonal drawing gives a (1/4)-balanced ✛-contact representation. There have been several attempts in the literature to characterize the graphs that admit 0- and 1-bend orthogonal drawings [5,6]. Recently, Tayu, Nomura, and Ueno [7] showed that every 2-tree with maximum degree four admits a 1-bend orthogonal drawing. In this paper we show that 2-trees and planar 3-trees admit (1/3)- and (1/2)-balanced ✛-contact representations, respectively, and thus admit 1-bend box-orthogonal drawings with boxes of size $\lceil \Delta/3 \rceil \times \lceil \Delta/3 \rceil$ and $\lceil \Delta/2 \rceil \times \lceil \Delta/2 \rceil$, respectively.

Planar Slope Number with One Bend Per Edge. A *k-bend polyline drawing* of a planar graph G is a planar drawing Γ of G, where each vertex is represented as a point and each edge is drawn as a polygonal chain with at most k bends. Γ is a *t-slope drawing* of G if the number of distinct slopes used by the line segments in Γ is at most t. The *planar slope number* of G is the smallest number t such that G admits a t-slope 0-bend drawing. A rich body of literature examines planar slope number of different subclasses of planar graphs [8,9,10]. Keszegh et al. [11] proved a q^{Δ} upper bound on the planar slope number, where q is a constant. They also showed that every planar graph G admits a 1-bend polyline drawing with at most 2Δ slopes, by a transformation from T-contact representations into 1-bend polyline drawings, as follows. Replace each vertical (respectively, horizontal) arm with Δ closely spaced nearly vertical (respectively, horizontal) slopes, e.g., see Figure 1(d). Finally, choose the bend points from the intersection points of these slopes such that the resulting drawing remains planar, e.g., see Figure 1(e). In this paper we show that 2-trees and planar 3-trees admit (1/3)- and (1/2)-balanced ✢-contact representations, respectively, and thus admit 1-bend polyline drawings with at most $2\lceil \Delta/3 \rceil$ slopes, and $2\lceil \Delta/2 \rceil$ slopes, respectively.

2 Definitions and Preliminary Approach

In this section we introduce some definitions and construct (1/2)-balanced ✢-contact representations for 2-trees.

A *2-tree*, or *series-parallel graph* (SP graph) G is a two-terminal directed simple graph with $n \geq 2$ vertices, which is defined recursively as follows.

(a) If $n = 2$, then G has a single edge (u, v), where either u or v is the source and the other vertex is the sink.
(b) If $n > 2$, then G can be constructed from two SP graphs G_1 and G_2 from one of the following two operations, e.g., see Figure 2(a).
  - *Series Composition*: Identify the sink of G_1 with the source of G_2.
  - *Parallel Composition*: Identify the source and sink of G_1 with the source and sink of G_2, respectively. Finally, identify any parallel edges.

A c-balanced representation of a given SP graph G can be constructed as follows. Construct a rectangle R and place the source and sink of G at the top-left and bottom-right corners, respectively. Initially each edge of R can have $\lceil c\Delta \rceil$ contact points. If G is formed by a series composition of two SP graphs G_1 and G_2, then we split R into four rectangles, e.g., see Figure 2(b), and draw G_1 and G_2 into the top-left and bottom-right rectangles, respectively. If G is formed by a parallel composition of G_1 and G_2, then we take two copies R_i, $i \in \{1, 2\}$, of R and draw G_i inside R_i (later on we merge these two drawings inside R). In both series and parallel cases, we distribute the available contact points among the subproblems, i.e., we compute the recursive drawings with bounded number of contact points on the edges of their bounding rectangles. In our algorithms, we specify the distribution of contact points so that we can merge the recursively computed drawings maintaining planarity.

Let h be an arm of some vertex while constructing a ✢-contact representation. By the *number of free points* of h we refer to the number of other arms that can touch h, which we denote by $f(h)$. If $f(h) = 0$, then we say h is *saturated*, otherwise h is *unsaturated*. The *center* of a vertex is the point, at which all four of its arms meet. For a center m, we denote by m_l, m_r, m_u, m_d the left, right, up and down arms of m, respectively. *Distributing* an integer z among the arms of m in some order $\sigma = (m_d, m_r, m_u, m_l)$ is an operation that finds the first arm h such that $z \le \sum_{h' \le_\sigma h} f(h')$, then sets $f(h) = z - \sum_{h' <_\sigma h} f(h')$, and finally, for all arms h'' subsequent to h, sets $f(h'') = 0$. Such an operation is defined only when $z \le \sum_h f(h)$. By $d_i(v, G)$ and $d_o(v, G)$ we denote the in-degree and out-degree of vertex v in G. We omit the term G if it is clear from the context. We now present a construction of (1/2)-balanced representations of SP graphs.

Lemma 1. *Let G be a SP graph with source s and sink t, and let G' be the graph obtained from G by deleting the edge (s, t), if such an edge exists. Let $R = abcd$ be an axis-parallel rectangle such that s and t lie on the opposite corners a and c, respectively. Assume that $f(a_d), f(a_r), f(c_l), f(c_u)$ are prespecified. If $d_o(s, G') \le f(a_d)+f(a_r)$ and $d_i(t, G') \le f(c_l)+f(c_u)$, then G' admits a (1/2)-balanced ✢-contact representation Γ in R satisfying the following property.*

($\star$) *The number of contact points at each arm incident to s and t in Γ is at most the number of free points specified for that arm as input.*

Proof. We employ an induction on the number of vertices n of G. If $n = 2$, then G' consists of two vertices of degree zero, i.e., s and t, that lie on the two opposite corners a and c of R, respectively. It is now straightforward to verify Property ($\star$). Hence assume that $n > 2$, and the lemma holds for every G that has fewer than n vertices. We now consider the case when G has n vertices.

Since G is a SP graph and $n > 2$, G' must be a SP graph, i.e., G' is obtained either by a series combination or a parallel composition of some SP graphs G_1 and G_2. Let s_i and t_i be the source and sink of G_i, respectively, where $i \in \{1, 2\}$. We now consider two cases depending on the composition of G_1 and G_2 in G'.

***Case 1 (Series Composition)*:** In this case $s = s_1$, $t_1 = s_2$ and $t_2 = t$. We first define two rectangles R_1 and R_2 inside R, where G_1 and G_2 will be drawn, respectively. To construct R_1 and R_2 we first add a vertex r inside R, which corresponds to the center of vertex $t_1 (= s_2)$. We then draw four orthogonal line segments re, rm, rg, rh such that $e \in ab$, $m \in bc$, $g \in cd$, $h \in ad$. Then $R_1 = aerh$ and $R_2 = rmcg$, as in Figure 2(b). We set $f(r_l) = f(r_r) = f(r_u) = f(r_d) = \lceil \Delta/2 \rceil$, and then assign the free points of s and t to s_1 and t_2, respectively.

If the edge (s_1, t_1) exists, then we draw (s_1, t_1) either along the polygonal chain ahr or aer, depending on whether $f(a_d) = 0$ or not. If the edge (s_2, t_2) exists, then we draw (s_2, t_2) either along the polygonal chain rgc or rmc, depending on whether $f(c_l) = 0$ or not. Here we consider the case when both (s_1, t_1) and (s_2, t_2) exist (the other cases can be treated similarly). Figure 2(c) shows such an example, where $f(a_d) \ne 0$ and $f(c_l) = 0$. Observe that while drawing (s_1, t_1) and (s_2, t_2), we use some free points of s_1 and t_2. Therefore, we decrease the free points by one for each arm that helps routing (s_1, t_1) and (s_2, t_2), e.g., see Figures 2(d)–(e). Since $d_o(s, G') \le f(a_d) + f(a_r)$ in R, we have $d_o(s_1, G_1 \setminus (s_1, t_1)) \le f(a'_d) + f(a'_r)$, where a' represents a in R_1. Since $d_i(t_1, G_1 \setminus (s_1, t_1)) \le \Delta - 1$, we have $d_i(t_1, G_1 \setminus (s_1, t_1)) \le f(r_u) + f(r_l)$ in R_1. Therefore, we can

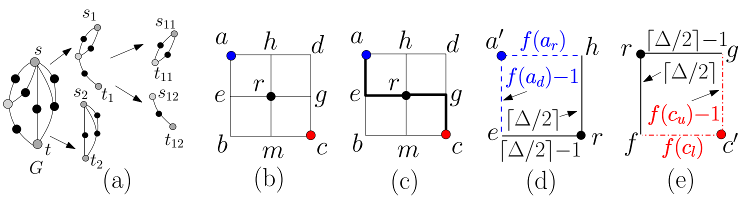

Fig. 2. (a) A few steps of series-parallel decomposition for some graph G, according to the definition. (b) Computation of R_1 and R_2. (c) Drawing (s_1, t_1) and (s_2, t_2). (d)–(e) Illustration for free points.

inductively draw G_i inside R_i, $i \in \{1, 2\}$. It is straightforward to merge these drawings by appropriate scaling. Since the drawings inside R_i maintain Property $(\star)$, the merged drawing also satisfies that property.

Case 2 (Parallel Composition): In this case $s = s_1 = s_2$ and $t = t_1 = t_2$. We first create two copies R_1 and R_2 of R, i.e., $R_1 = a'b'c'd'$ and $R_2 = a''b''c''d''$, where G_1 and G_2 will be drawn, respectively. Figure 3 illustrates an example.

We now define the free points of the arms of s_1, t_1 and s_2, t_2 that are inside R_1 and R_2, respectively. We distribute $d_o(s_1)$ among a'_d and a'_r in this order, i.e., we set $f(a'_d) = \min\{f(a_d), d_o(s_1)\}$, and $f(a'_r) = \max\{0, d_o(s_1) - f(a'_d)\}$. Similarly, distribute $d_i(t_1)$ among c'_l and c'_u in this order, i.e., set $f(c'_l) = \min\{f(c_l), d_i(t_1)\}$, and $f(c'_u) = \max\{0, d_i(t_1) - f(c'_l)\}$, e.g., Figure 3(b). The number of free points of s_2 and t_2 is the number of free points of s and t that remains after assigning free points to s_1 and t_1, as shown in Figure 3(c).

Since $d_o(s) \leq f(a_d) + f(a_r)$ and $d_o(s) = d_o(s_1) + d_o(s_2)$, according to our assignment of free points, $d_o(s_1) \leq f(a'_d) + f(a'_r)$. Similarly, since $d_i(t) \leq f(c_l) + f(c_u)$ and $d_i(t) = d_i(t_1) + d_i(t_2)$, we obtain $d_i(t_1) \leq f(c'_l) + f(c'_u)$. It is now straightforward to observe that $d_o(s_2) \leq f(a''_d) + f(a''_r)$ and $d_i(t_2) \leq f(c''_l) + f(c''_u)$. Therefore, by induction, can draw G_1 and G_2 inside R_1 and R_2, respectively.

The drawing of G_1 takes consecutive free points from the arms of s (respectively, t) in anticlockwise (respectively, clockwise) order. The drawing of G_2 takes the remaining consecutive free points in the same order. Therefore, one can merge the two drawings inside R_1 and R_2 avoiding edge crossings inside R. The details are omitted due to space constraints. □

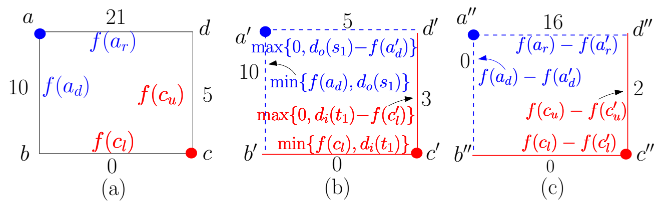

Fig. 3. (a) R. (b)–(c) Computation of R_1 and R_2. The numbers exterior to the three rectangles illustrate a concrete example, where $d_o(s_1) = 15$ and $d_i(t_1) = 3$.

Theorem 1. *Every SP graph G has a $(1/2)$-balanced ✛-contact representation.*

Proof. If the source s and sink t of G are not adjacent, then by Lemma 1, G admits the required representation. Otherwise, let G' be the graph $G \setminus (s,t)$. By Lemma 1, G' admits a (1/2)-balanced ✛-contact representation inside a rectangle $R = abcd$, where s and t lie on the opposite corners a and c, respectively, and the free points $f(a_d), f(a_r), f(c_l), f(c_u)$ are prespecified such that $d_o(s,G') \leq f(a_d) + f(a_r)$ and $d_i(t,G') \leq f(c_l) + f(c_u)$.

We define $f(a_d) = \lceil \Delta/2 \rceil - 1, f(a_r) = \lceil \Delta/2 \rceil, f(c_l) = \lceil \Delta/2 \rceil - 1$ and $f(c_u) = \lceil \Delta/2 \rceil$. Since $d_o(s,G') \leq \Delta - 1$ and $d_i(t,G') \leq \Delta - 1$, the conditions $d_o(s,G') \leq f(a_d) + f(a_r)$ and $d_i(t,G') \leq f(c_l) + f(c_u)$ hold. Hence by Lemma 1, we can compute a (1/2)-balanced ✛-contact representation Γ' of G' inside R. Finally, we draw the edge (s,t) along the polygonal chain abc. □

3 Balanced Representations for 2-Trees ($c = 1/3$)

The idea of the algorithm for computing (1/3)-balanced ✛-contact representations is similar to that of Section 2, however, here the construction is more involved. Let $\overline{uv}$ denote the line segment from u to v. We first prove the following lemma, which is similar to Lemma 1.

Lemma 2. *Let G be a SP graph with source s and sink t, and let G' be the graph obtained from G by deleting the edge (s,t), if such an edge exists. Let $R = k_1k_2k_3k_4$ be an axis-parallel rectangle such that s and t are centered at $a \in \overline{k_1k_2}$ and $c \in \overline{k_2k_3}$, respectively, but not at k_2. Assume that the free points of the arms of s and t that lie on R are prespecified. Let x (respectively, y) be the total number of free points of all arms of s (respectively, t) that lie inside R. If $d_o(s,G') \leq x$ and $d_i(t,G') \leq y$, then G' admits a $(1/3)$-balanced ✛-contact representation Γ in R satisfying the following property.*

(☆) *The number of contact points at each arm incident to s and t in Γ is at most the number of free points specified for that arm as input.*

Proof. We employ an induction on the number of vertices n of G. The case when $n = 2$ is straightforward, hence we now assume that $n > 2$, and the lemma holds for every G that has fewer than n vertices. We now consider the case when G has n vertices. Since G is a SP graph and $n > 2$, G' must be a SP graph, i.e., G' is obtained either by a series or a parallel composition of some SP graphs G_1 and G_2. Let s_j and t_j be the source and sink of G_j, respectively, where $j \in \{1,2\}$. We consider two cases depending on the composition of G_1 and G_2 in G'.

***Case 1 (Series Composition)*:** We first construct two rectangular regions R_1 and R_2 inside R, where G_1 and G_2 will be drawn, respectively, and then define the free points. In the following we construct R_1 and R_2 assuming that

$d_i(t_1) \geq 2\lceil\Delta/3\rceil$. Therefore, we ensure that three of the arms of t_1 lie in R_1 and one of the arms of s_2 lies in R_2. The case when $d_i(t_1) < \lceil\Delta/3\rceil$ (i.e., $d_o(s_2) \geq 2\lceil\Delta/3\rceil$) is symmetric. By slightly modifying the construction[1] we can deal with the case when $\lceil\Delta/3\rceil \leq d_i(t_1) < 2\lceil\Delta/3\rceil$. We omit the details due to space constraints.

A. Determine the leftmost arm h in the sequence a_d, a_r, a_u that is not saturated.
B. Determine the leftmost arm h' in the sequence c_l, c_u, c_r that is not saturated.
C. If h and h' lie on the boundary of R, then we compute R_1 and R_2 according to the cases (C_1)–(C_3). Figure 4 shows that the case analysis is exhaustive by examining all possible positions of a and c in R. In Figure 4, the point r corresponds to the center of $t_1 (= s_2)$.
   - (C_1) If h is parallel to h' and $h = a_r$ (i.e., Column 3 of Row 1 in Figure 4), then we draw a straight line pq such that p, q are two points on h, h', respectively. Let r and r' be two distinct points on pq such that dist$(p, r) <$dist(p, r'). We then draw a line segment $r'z \perp pq$, such that $z \in c_u$. R_1 and R_2 are the rectangles that contain the unsaturated arms, i.e., in this case R_1 (respectively, R_2) is the rectangle with diagonal $r'k_4$ (respectively, $r'c$). Sometimes R_i, $i \in \{1, 2\}$, may not contain the center of the corresponding source and sink. In such a case, we add a dummy copy of the source or sink, e.g., see the gray diamond shapes in Figure 4. Note that while computing the drawing of G_1 inside R_1 inductively, we rotate R_1 by 90° anticlockwise such that the preconditions of the induction hold. Furthermore, we define $f(s_1z) = 0$ such that no unnecessary adjacencies are created in the inductive drawing. Since the addition of dummy copy of a source or sink is straightforward, we do not explicitly describe them in the subsequent cases.
   - (C_2) If h is parallel to h' and $h \in \{a_d, a_u\}$ (i.e., Column 2 of Row 1 and Columns 1–2 of Row 2 in Figure 4), then we draw a straight line pq such that p, q are two points on h, h', respectively. Let r be a point on pq. We then draw a line segment $rz \perp pq$, such that $z \in c_l$.
   - (C_3) Otherwise, $h \perp h'$ (i.e., Columns 1 and 4 of Row 1, Columns 3–4 of Row 2, and Row 3 in Figure 4). Here we draw a polygonal chain p, r, q such that p, q are two points on h, h', respectively, $pr \perp rq$. We then draw a line segment $rz \perp rq$, such that either $z \in k_2k_3$ (when rq is horizontal), or $z \in k_3k_4$ (when rq is vertical).

     An interesting case is shown in Column 2 of Row 3 in Figure 4, where the dummy vertex is placed in the proper interior of the segment qk_3 instead of placing it on q. The reason is to respect the precondition of the induction that s_2 and t_2 should not lie on q. Here we set the free points of the left and up arms of t_2 to 0 to avoid any unnecessary adjacencies in the recursive construction.

[1] Here we ensure that at least two arms of t_1 (respectively, s_2) lie in R_1 (respectively, R_2). At most one arm of t_1 on the boundary of R_1 may coincide with an arm of s_2 on the boundary of R_2, where we assign the free points depending on $d_i(t_1)$.

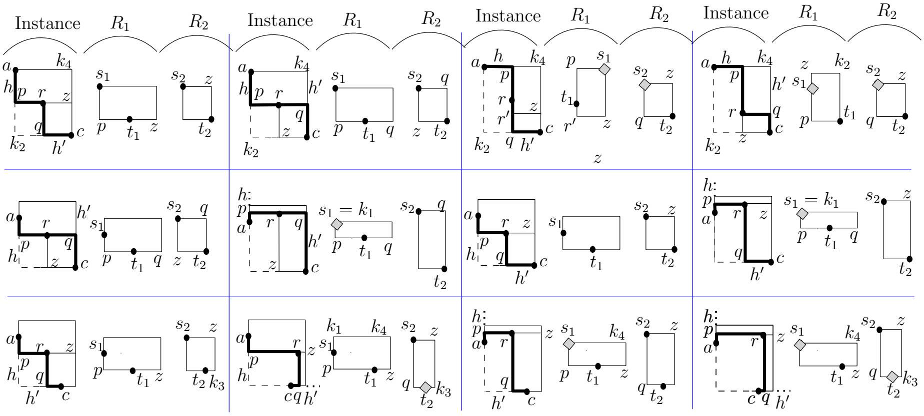

Fig. 4. Computation of R_1 and R_2 when h and h' lie on the boundary of R, and $d_i(t_1) \geq \lceil \Delta/3 \rceil$. **(Case** $a = k_1$**,** $c = k_3$**)**: Row 1. **(Case** $a \neq k_1$**,** $c = k_3$**)**: Row 2. **(Case** $a = k_1$**,** $c \neq k_3$**)**: Symmetric to Row 2. **(Case** $a \neq k_1$**,** $c \neq k_3$**)**: Row 3.

D. Otherwise, at least one of h, h' is in the proper interior of R. In this scenario we consider the following cases depending on the positions of a and c in R.
   (D_1) If $a \neq k_1$ and $c = k_3$ (i.e., Row 1 of Figure 5), then we follow (C_2) or (C_3), depending on whether $h||h'$ or $h \perp h'$, setting $p = r$.
   (D_2) If $a = k_1$ and $c \neq k_3$, then the computation is symmetric to (D_1).
   (D_3) Otherwise, both h and h' may lie in the proper interior of R. In this case, if h' lies on the boundary of R, then the computation of R_1 and R_2 is shown in Row 2 of Figure 5. Otherwise, h' lies in the proper interior of R, and the computation of R_1 and R_2 depends on whether $f(c_r) \neq 0$ (i.e., see Row 3 of Figure 5) and $f(c_r) = 0$ (i.e., Row 4 in Figure 5). The details are omitted due to space constraints.

Computation of free points: If R_2 contains an arm of r that does not lie on the boundary of R_1, then we set $f(r_l) = f(r_r) = f(r_u) = f(r_d) = \lceil \Delta/3 \rceil$. Otherwise, R_2 contains only one arm of r and it is shared with R_1, i.e., Columns 3–4 of Row 1 in Figure 4, and Row 4 of Figure 5. In such a case, we assign $\lceil \Delta/3 \rceil$ free points to the arms of r that are not shared, and for the shared arm, we assign $d_o(s_2, G_2)$ free points in R_2 and $\lceil \Delta/3 \rceil - d_o(s_2, G_2)$ free points in R_1.

We now assign the free points of s and t to s_1 and t_2, respectively, and place $t_1 (= s_2)$ on r. If the edge (s_i, t_i) exists, $i \in \{1, 2\}$, then we draw (s_i, t_i) along h and h', as shown in bold in Figures 4 and 5. Observe that while drawing (s_i, t_i), we use some free points of s_1 and t_2. Therefore, we decrease the free points by one for each arm that helps routing (s_i, t_i). Since R_1 includes all unsaturated arms of s that lie in R, the number of free points of a in R_1 is at least $d_o(s_1, G_1 \setminus (s_1, t_1))$. According to our assignment of free points, the number of free points of r in R_1 is at least $d_i(t_1, G_1 \setminus (s_1, t_1))$. Therefore, we can inductively draw G_1 inside R_1, and similarly G_2 inside R_2.

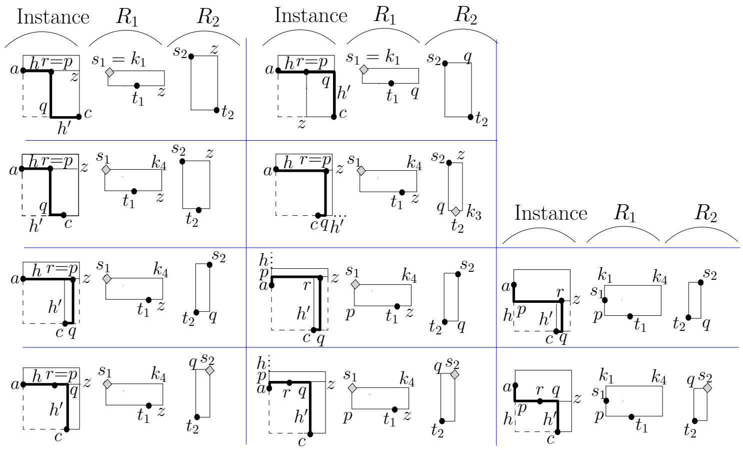

Fig. 5. Computation of R_1 and R_2 when at least one of h, h' lie in the proper interior of R, and $d_i(t_1) \geq \lceil \Delta/3 \rceil$. **(Case** $a \neq k_1$ **and** $c = k_3$**)**: Row 1. **(Case** $a = k_1$ **and** $c \neq k_3$**)**: Symmetric to Row 1. **(Case** $a \neq k_1$ **and** $c \neq k_3$**)**: Rows 2–4.

While computing the drawings of G_1 and G_2 inductively, sometimes we rotated R_1 and R_2 anticlockwise. Therefore, before merging such a drawing, we rotate it clockwise by the same amount. Furthermore, while computing the drawings of G_1 and G_2, sometimes we added some dummy source and sink. For any arm h of the dummy vertex, that is not a part of the arm of its real copy (e.g., see the illustration in Case (C_1)), we set $f(h) = 0$. Therefore, the merged drawing correctly realizes all adjacencies. Since the drawings inside R_1 and R_2 maintains Property (☆), the merged drawing also satisfies that property.

***Case 2 (Parallel Composition)*:** In this case $s = s_1 = s_2$ and $t = t_1 = t_2$. We first create two copies R_1 and R_2 of R, i.e., $R_1 = a'b'c'd'$ and $R_2 = a''b''c''d''$, where G_1 and G_2 will be drawn, respectively. We now define the free points. Recall that in Case 2 of Lemma 1, we distributed $d_o(s_1)$ among $f(a'_d), f(a'_r)$, and $d_i(t_1)$ among $f(c'_l), f(c'_u)$. Since here we may have at most three arms of s and t inside R, we distribute $d_o(s_1)$ among a'_d, a'_r and a'_u in this order, and similarly, distribute $d_i(t_1)$ among c'_l, c'_u and c'_r in this order. The number of free points in the arms of s_2 and t_2 is determined by the free points of s and t that remains after assigning free points to s_1 and t_1.

According to our assignment of free points, $d_o(s_1) = f(a'_d) + f(a'_r) + f(a'_u)$. Since $d_o(s) \leq f(a_d) + f(a_r) + f(a_u)$ and $d_o(s) = d_o(s_1) + d_o(s_2)$, the inequality $d_o(s_2) \leq f(a''_d) + f(a''_r) + f(a''_u)$ holds. Similarly, $d_i(t_1) = f(c'_l) + f(c'_u) + f(c'_r)$ and $d_i(t_2) \leq f(c''_l) + f(c''_u) + f(c''_r)$. Therefore, by induction, we can draw G_1 and G_2 inside R_1 and R_2, respectively.

The idea of merging the drawings of G_1 and G_2 into R is similar to the Case 2 of Lemma 1. Observe that the drawing of G_1 takes consecutive free points from the arms of s (respectively, t) in anticlockwise (respectively, clockwise) order. On the other hand, the drawing of G_2 takes the remaining consecutive free points from the arms of s (respectively, t) in anticlockwise (respectively, clockwise) order. Therefore, one can merge the two drawings inside R_1 and R_2 avoiding edge crossings inside R. Since the drawings inside R_1 and R_2 maintains Property (☆), the combined drawing also satisfy that property. □

Theorem 2. *Every SP graph G has a $(1/3)$-balanced ✢-contact representation, but not necessarily a $(1/4-\epsilon)$-balanced representation, for any $\epsilon > 0$.*

Proof. The proof for the upper bound is analogous to the proof of Theorem 1. The only difference is that here we use Lemma 2 instead of Lemma 1. The proof for the lower bound is implied by SP graphs with $\Delta \geq 4$ and Δ mod $4 = 0$. □

4 Balanced Representations of Planar 3-Trees ($c = 1/2$)

In this section we show that planar 3-trees admit (1/2)-balanced representations. A *planar* 3*-tree* G with $n \geq 3$ vertices is a triangulated planar graph such that if $n > 3$, then G contains a vertex whose deletion yields a planar 3-tree with $n-1$ vertices. Let x, y, z be a cycle in G. By G_{xyz} we denote the subgraph induced by x, y, z and the vertices that lie interior to the cycle. Every planar 3-tree G with $n > 3$ vertices contains a vertex that is the common neighbor of all three outer vertices of G. We call this vertex the *representative vertex* of G. Let p be the representative vertex of G and let a, b, c be the three outer vertices of G, as in Figure 6(a). The subgraphs G_{abp}, G_{bcp} and G_{cap} are planar 3-trees. Let G'_{abp}, G'_{bcp} and G'_{cap} be the subgraphs obtained by deleting the outer edges of G_{abp}, G_{bcp} and G_{cap}, respectively. These subgraphs the three *nested components* of G. By $d(u, G)$, we denote the degree of vertex u in G. Given a planar 3-tree G and a rectangle R, we recursively divide R into three sub-rectangles where the nested components of G will be drawn. We first prove the following lemma.

Lemma 3. *Let G be a planar* 3*-tree with outer vertices a, b, c and representative vertex p, and let G' be the graph obtained from G by deleting the outer edges of G. Let $R = k_1k_2k_3k_4$ be an axis-parallel rectangle such that a, b, c lie on k_1k_2, k_2k_3 and k_4, respectively. Assume that the number of free points of each arm of a, b, c is prespecified. If the inequalities $d(a, G') \leq f(a_d) + f(a_u)$, $d(b, G') \leq f(b_l) + f(b_r)$, and $d(c, G') \leq f(c_l) + f(c_d)$ hold, then G' admits a $(1/2)$-balanced ✢-contact representation Γ in R satisfying the following property.*

(✻) *The number of contact points at each arm incident to a, b and c in Γ is at most the number of free points specified for that arm as input.*

Proof. We employ an induction on the number of vertices of G. If $n = 3$, then G' consists of only three isolated vertices a, b and c that lie on k_1k_2, k_2k_3 and k_4, respectively. It is now straightforward to verify Property (✻). Hence we assume that $n \geq 4$ and the lemma holds for all G with smaller than n vertices. We now consider the case when G has n vertices.

We first compute three sub-rectangles R_1, R_2 and R_3, where G'_{abp}, G'_{bcp} and G'_{cap} will be drawn, respectively. Define h to be either a_u or a_d depending on whether $d(a, G'_{abp}) \geq f(a_d)$ or not. Similarly, define h' to be either b_r or b_l

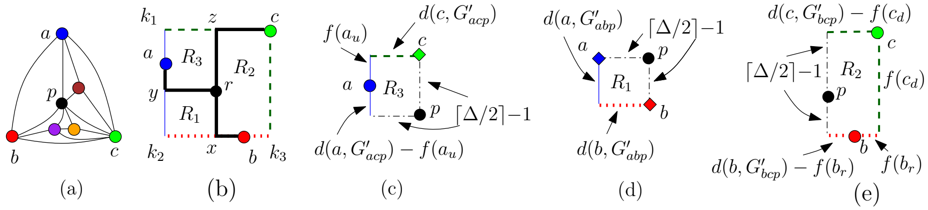

Fig. 6. (a) A plane 3-tree. (b) Computation of R_1, R_2 and R_3, where $d(a, G'_{abp}) < f(a_d)$, $d(b, G'_{abp}) < f(b_l)$, and $d(c, G'_{bcp}) \geq f(c_d)$. (c)–(e) Assignment of free points.

depending on whether $d(b, G'_{abp}) \geq f(b_l)$ or not. Since the inequalities $f(a_d) + f(a_u) \geq d(a, G') \geq 1$ and $f(b_l) + f(b_r) \geq d(b, G') \geq 1$ hold, h and h' must be unsaturated. Let x and y be two points on h' and h, respectively, as shown in Figure 6(b). Draw two line segments $xr \perp h'$ and $yr \perp h$ such that they meet at point r. Define h'' to be the arm c_l or c_d depending on whether $d(c, G'_{bcp}) \geq f(c_d)$ or not. Since $f(c_l) + f(c_d) \geq d(c, G') \geq 1$, h'' must be unsaturated. We draw an orthogonal line segment rz such that $z \in h''$. Observe that rx, ry and rz divides R into three sub-rectangles R_1, R_2 and R_3, i.e., the sub-rectangles that contain corners k_2, k_3 and k_1, respectively.

We place the vertex p on r, draw the edges $(a, p), (b, p)$ and (c, p) along ry, rx and rz, respectively, and then assign $\lceil \Delta/2 \rceil - 1$ free points at each arm of r. To define the free points of the other arms of R_i, we distribute the free points of a, b and c as follows. We distribute $d(a, G'_{abp})$ among a_d and a_u (in R_1), and $d(a, G'_{acp})$ among a_u and a_d (in R_3). We then distribute $d(b, G'_{abp})$ among b_l and b_r (in R_1), and $d(b, G'_{bcp})$ among b_r and b_l (in R_2). Finally, we distribute $d(c, G'_{bcp})$ among c_d and c_l (in R_2), and $d(c, G'_{acp})$ among c_l and c_d (in R_3).

Let G'_i be the nested component of G that corresponds to R_i, $i \in \{1, 2, 3\}$. Observe that some outer vertices of G'_i may not lie on R_i. Hence we cannot directly apply the induction hypothesis. Hence for each vertex a, b or c that does not lie on the boundary of R_i but belongs to G'_i, we add a dummy copy of that vertex at x, y or z, respectively. Furthermore, for each arm h of the dummy copy that is not a part of any arm of its real copy, we set $f(h) = 0$, e.g., $f(c_d) = 0$ in Figure 6(c). Consequently, the recursively computed drawings do not create any unnecessary adjacencies. Observe that each R_i now meets the preconditions of the induction, as shown in Figures 6(c)–(e), and hence we inductively draw G'_i inside R_i. To apply the induction, we need to be careful of the vertex that play the role of k_4, i.e., the corner having exactly two arms inside the rectangle that are perpendicular to each other, e.g., the position of p in Figures 6(c)–(d), and the position of c in Figure 6(e). Each R_i contains exactly one of r and c at one of its four corners, which plays the role of k_4 in R_i. Since the smaller drawings satisfy Property (✼), the final drawing satisfies Property (✼). □

Theorem 3. *Every planar 3-tree G admits a $(1/2)$-balanced ✢-contact representation, but not necessarily a $(1/3)$-balanced representation.*

Proof. Let a, b, c be the outer vertices of G, and let $R = k_1k_2k_3k_4$ be an axis-parallel rectangle. Place a, b, c on k_1k_2, k_2k_3 and k_4, respectively, draw the outer edges of G along the boundary of R, and finally, assign $\lceil \Delta/2 \rceil - 1$ free points at each arm of a, b and c. Let G' be the graph obtained by removing the outer edges of G. By Lemma 3, G' has a (1/2)-balanced ✛-contact representation in R. The lower bound that $c > 1/3$ is implied by the graph K_4. □

5 Conclusion

We have proved that 2-trees (respectively, planar 3-trees) admit c-balanced ✛-contact representations, where $1/4 \leq c \leq 1/3$ (respectively, $1/3 < c \leq 1/2$). A natural open question is to find tight bounds on c. Although our representations for planar 3-trees preserve input embedding, the representations for 2-trees do not have this property. Thus it would be interesting to examine whether there exist algorithms for (1/3)-balanced representations of 2-trees that preserve input embedding. Another intriguing open question is to characterize planar graphs that admit c-balanced ✛-contact representations, for small fixed values c.

References

1. de Fraysseix, H., de Mendez, P.O., Rosenstiehl, P.: On triangle contact graphs. Combinatorics, Probability and Computing 3(2), 233–246 (1994)
2. Czyzowicz, J., Kranakis, E., Urrutia, J.: A simple proof of the representation of bipartite planar graphs as the contact graphs of orthogonal straight line segments. Information Processing Letters 66(3), 125–126 (1998)
3. Kobourov, S.G., Ueckerdt, T., Verbeek, K.: Combinatorial and geometric properties of planar Laman graphs. In: Khanna, S. (ed.) Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1668–1678. SIAM (2013)
4. Biedl, T.C., Kaufmann, M.: Area-efficient static and incremental graph drawings. In: Burkard, R.E., Woeginger, G.J. (eds.) ESA 1997. LNCS, vol. 1284, pp. 37–52. Springer, Heidelberg (1997)
5. Kant, G.: Drawing planar graphs using the canonical ordering. Algorithmica 16(1), 4–32 (1996)
6. Zhou, X., Nishizeki, T.: Orthogonal drawings of series-parallel graphs with minimum bends. SIAM Journal on Discrete Mathematics 22(4), 1570–1604 (2008)
7. Tayu, S., Nomura, K., Ueno, S.: On the two-dimensional orthogonal drawing of series-parallel graphs. Discrete Applied Mathematics 157(8), 1885–1895 (2009)
8. Dujmović, V., Suderman, M., Wood, D.R.: Graph drawings with few slopes. Computational Geometry 38(3), 181–193 (2007)
9. Lenhart, W., Liotta, G., Mondal, D., Nishat, R.I.: Planar and plane slope number of partial 2-trees. In: Wismath, S., Wolff, A. (eds.) GD 2013. LNCS, vol. 8242, pp. 412–423. Springer, Heidelberg (2013)
10. Jelínek, V., Jelínková, E., Kratochvíl, J., Lidický, B., Tesar, M., Vyskocil, T.: The planar slope number of planar partial 3-trees of bounded degree. Graphs and Combinatorics 29(4), 981–1005 (2013)
11. Keszegh, B., Pach, J., Pálvölgyi, D.: Drawing planar graphs of bounded degree with few slopes. SIAM Journal on Discrete Mathematics 27(2), 1171–1183 (2013)