Master Thesis

Extension of Partial Contact Representations

Peter Markfelder

Date of Submission: December 14, 2021[1]
Advisors: Prof. Dr. Alexander Wolff
Dr. Jonathan Klawitter

Julius-Maximilians-Universität Würzburg
Lehrstuhl für Informatik I
Algorithmen und Komplexität

[1]Final Version: February 2, 2022

Abstract

Contact representations of graphs are a well studied topic with practical applications in, for example, VLSI design and architecture. Recently, the *partial contact representation extension problem* emerged. It asks whether a given partial contact representation of a graph can be extended to a contact representation of the graph, that is, for a subset of the vertices, there are prescribed objects that need to appear in the extended contact representation. We study the partial contact representation extension problem for rectangle, square and triangle representations. For rectangles and maximal, triangle-free planar graphs, we show that the partial contact representation extension problem can be solved in linear time, given a corner edge labeling of the graph that describes the desired extension in a combinatorial way. In the affirmative, a rectangle contact representation can be obtained within the same time bound. For square representations, we propose a linear program that solves the partial square dual extension problem in polynomial time and that can be used to construct an extension for yes-instances. Regarding triangles, we define a set of necessary conditions that need to be satisfied for an extension to exist.

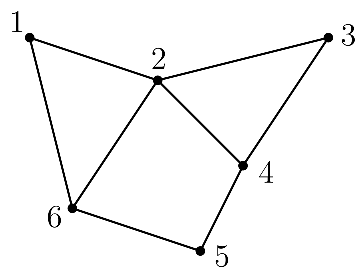
Zusammenfassung

Kontaktrepräsentationen von Graphen sind ein häufig untersuchtes Thema mit praktischen Anwendungen, zum Beispiel im Kontext von VLSI Design oder Architektur. Vor Kurzem wurde das *Kontaktrepräsentationserweiterungsproblem* vorgestellt, welches fragt, ob eine unvollständige Kontaktrepräsentation eines Graphen zu einer Kontaktrepräsentation erweitert werden kann. Dies bedeuted, dass für eine Teilmenge der Knoten geometrische Objekte vorgegeben sind, welche in der erweiterten Kontaktrepräsentation vorkommen müssen. Wir behandeln das Kontaktrepräsentationserweiterungsproblem für Repräsentationen mit Rechtecken, Quadraten und Dreiecken. Für Rechtecke und maximale, dreicksfreie planare Graphen zeigen wir, dass das Kontaktrepräsentationserweiterungsproblem in Linearzeit lösbar ist, gegeben ein Corner Edge Labeling des Graphen, welches die gesuchte Erweiterung kombinatorisch beschreibt. Sofern eine Erweiterung existiert, kann diese ebenfalls in Linearzeit konstruiert werden. Für Kontaktrepräsentationen mit Quadraten stellen wir ein lineares Programm vor, welches das Quadratdual-Erweiterungsproblem in Polynomialzeit löst und welches dazu verwendet werden kann, eine Erweiterung zu konstruieren, sofern eine solche existiert. Für Dreickskontaktrepräsentationen definieren wir eine Menge von notwendigen Bedingungen, welche erfüllt sein müssen, damit eine Erweiterung existiert.

Contents

1 Introduction

The visualization of data and information is an important tool to understand the relationships between entities. A computer or machine is able to process raw information, but for humans it is easier to identify patterns when the data is visually edited. In the context of the field of Graph Theory, a *graph* is an abstract data structure consisting of a set of objects called *vertices*, and a set of pairs of vertices, called *edges*. Graphs are common to model pairwise relationships between entities, such as people in social networks, where each vertex represents a person and the edges represent, for example, friendships or co-workers. Another application for graphs is the design of integrated circuits in computer micro controllers (VLSI), in which case the vertices represent electronic nodes and edges the connection between them. Graphs are common to model problems that have such pairwise relationships and there is a wide range of graph algorithms that are used to solve the underlying problems. To give an example, industry production lines can be modeled as graphs with the production steps being represented by vertices and the edges model the relative order in which those task must the completed for the production line to efficiently function. Graph algorithms can then be used to optimize that flow of partially assembled products. To give another example, graphs can be used to model maps and road networks with cities, landmarks and other points of interest as vertices and roads connecting them as edges. A common problem for such a model is finding the shortest path between two points.



	1	2	3	4	5	6
1	0	1	0	0	0	1
2		0	1	1	0	1
3			0	1	0	0
4				0	1	0
5					0	1
6						0

Fig. 1.1: The same graph represented as node-link diagram (left) and as adjacency matrix (right). The node-link diagram conveys the pairwise relationship for a human reader while the adjacency matrix is a straightforward representation for graph algorithms.

Graphs are however just mathematical concepts. To convey their information to the human reader, it is only natural to take a closer look at visualizations of graphs. When it comes to drawing graphs one common method is to visualize them as *node-link diagrams*. Those are drawings in the plane where the vertices are drawn as uniform objects, such as disks or boxes, with lines connecting the disks as edges. For comparison, another way to represent graphs is an *adjacency matrix*, which is a square matrix where the elements of the matrix indicate whether two vertices share an edge. Adjacency matrices are often used to encode graphs in graph algorithms, while node-link diagrams are common visual representations of graphs. See Figure 1.1 for a node-link diagram of a graph in comparison to its adjacency matrix.

Graph visualization is a well studied topic in computer science [BETT98, KW03, NR04]. Since asking whether a drawing of a graph is visually pleasing is subjective, it is difficult to develop metrics that define what makes a good drawing. However, one common metric to achieve good drawings in the context of node-link diagrams is minimizing the number of edge crossings. A graph that can be drawn without an edge crossing is called a *planar* graph. Other metrics include minimizing the drawing area, defined as minimizing the area of the bounding box containing all vertices, or putting constraints on the shape of the drawing, for example restricting the drawing to use only straight lines for their edges. Figure 1.2 shows an example of a graph being drawn in three different ways as node-link diagrams, making use of crossing minimization and straight lines to improve the drawing.

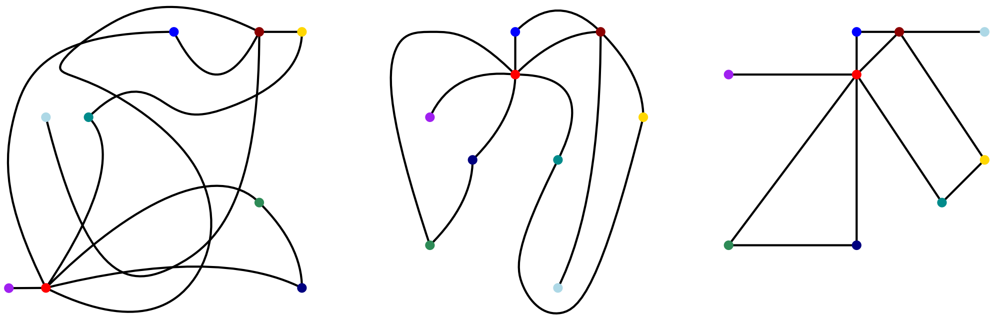


Fig. 1.2: Three drawings of the same graph. The drawings in the center and on the right are crossing free. Additionally the right drawing uses only straight lines for the edge segments.

Contact Representations. While node-link diagrams are common to visualize graphs, other geometric representations of graphs have been studied. A *geometric intersection representation* of a graph G is a mapping that assigns each vertex a geometric object and two vertices of G are adjacent if and only if their assigned geometric objects intersect. The geometric objects are often restricted to be of the same shape, for example rectangles, triangles or straight line segments. For a *contact representation* it is further required that any two objects have disjoint interiors, that is their boundaries touch, but do not intersect. While every graph can be drawn as a node-link diagram, it is generally NP-hard to decide whether a graph admits a contact representation for a particular given shape. Figure 1.3 shows multiple contact representations of the same graph using different objects.

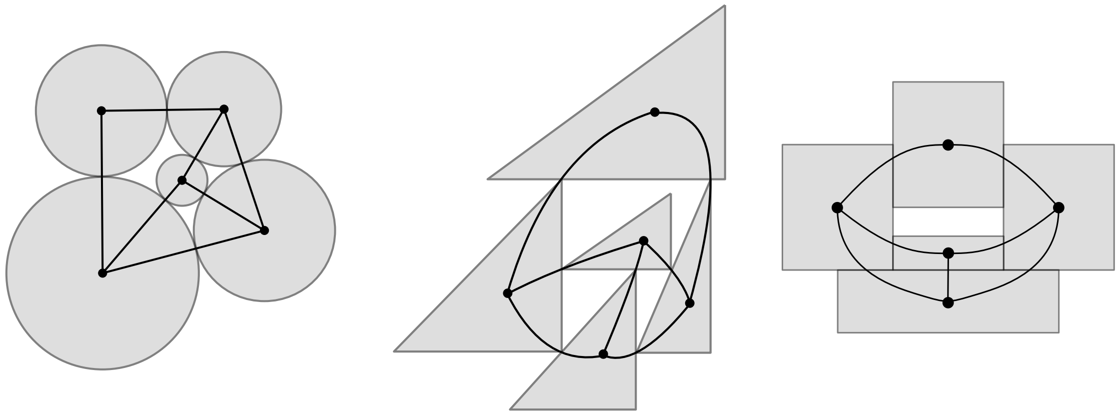


Fig. 1.3: Three different contact representations for the same graph using disks, triangles and rectangles, respectively.

A historical result regarding contact representations was found by Koebe [Koe36] with their Circle Packing Theorem. It states that the class of graphs that admit a contact representation using internally disjoint disks as objects is exactly the class of planar graphs. Schramm's Convex Packing Theorem [Sch07] is a strong generalization of Koebe's theorem. It states that if each vertex of a triangulated planar graph G has a convex prototype object, then there exists a contact representation of G where each vertex is represented by a (possibly degenerate) homothet of its prototype. If the prototypes have smooth boundaries, then there is no degenerate case.

Problems revolving around contact representations are usually geometric problems. To make them more accessible from an algorithmic point of view, it is common to use combinatorial descriptions of contact representations. Think of a contact representation of a graph as a structure described by coordinates of geometric objects, whereas a combinatorial description specifies the way in which two objects make contact, for example which corners or sides of objects make contact and in what relative order. Since Schramm's Convex Packing Theorem, contact representations of graphs with convex polygon prototypes have seen a lot of attention using combinatorial descriptions. Often, contact representations of *maximal planar* graphs, also called *triangulated* graphs, are considered, since a contact representation of a graph G induces a contact representation of any sub-graph of G.

Previous and Related Work. De Fraysseix et al. [dFPP90] studied triangle contact representation. They observed that *Schnyder woods* [Sch89, Sch90] can be seen as combinatorial descriptions of triangle contact representations of triangulated graphs and that any Schnyder wood can be used to construct a triangle contact representation. Angelini et al. [ACC^{+}19] have shown that any two right-angle axis-aligned triangle contact representation that have the same Schnyder wood can be morphed into each other. Following from Schramm's Convex Packing Theorem, Gonçalves et al. [GLP11] proved that every 4-connected triangulated graph admits a contact representation using homothetic triangles, that is, every vertex has the same triangle prototype. Felsner [Fel09] proposed a system of linear equations which are based on Schnyder woods, such that a solution to the system yields a triangle contact representation.

Historically, contact representations where all objects are axis-aligned rectangles (*rectangle contact representations*) have been studied due to their application in VLSI design and architectural floor planning [LL84, YS95, KS15, Ste73]. A combinatorial description for axis-aligned rectangle contact representations where no four rectangles make contact at their corners and the union of all rectangles is a rectangle (*rectangular dual*) was introduced by Kant and He [KH97]. The *regular edge labeling* of a rectangle contact representation is an orientation and 2-coloring of the contact graph that describes, for two touching rectangles, which of their four sides make the contact. Regular edge labelings are also known as *transversal structures* [Fus09]. Kant and He have shown that every 4-connected triangulated graph G with four vertices on the outer face admits a rectangular dual and that the regular edge labeling is a combinatorial description of the rectangular dual. Klawitter et al. [KNU15] introduced a combinatorial description for rectangle contact representations of *maximal, triangle-free, planar* (MTP) graphs called the *corner edge labeling* and proved an analogous result: Every MTP graph G admits an axis-aligned rectangle contact representation of G and the corner edge labeling is a combinatorial description of said contact representation.

Another historical work regarding *square contact representations* was made by Brooks et al. [BSST40], who studied the problem of dissecting rectangles into squares. Later Schramm [Sch93] showed that every 5-connected inner triangulation of a 4-gon admits a square contact representation. Following up from Schramm's result, Felsner [Fel13] proposed, similarly to their system of linear equations for triangle contact representations, a system of linear equations for square duals, which uses the regular edge labeling.

Felsner et al. [FSS18] considered contact representations of equiangular K-gons and introduced a combinatorial description called *K-contact-structure*. In the case of $K = 3$, the K-contact-structure is a Schnyder wood and in the case of $K = 4$, it is a regular edge labeling. Alam et al. [ABF$^+$12] studied proportional polygon contact representations of graphs with specified vertex weights that correspond to the area the polygon of a vertex covers in a contact representation. Specifically, they optimized such contact representations in regards to polygon complexity, cartographic error and unused area.

The discussion about contact representations is usually tied to the shape of the object representing each vertex and next to polygons, rectangles, squares and triangles, many other families of shapes have been studied. For example interval graphs [BL76], line segments [dFdM07, KUV13] or visualizations using cubes [FF11] and boxes [Tho86, CKU13] for three-dimensional representations.

A natural problem regarding contact representations is the *recognition problem*, which asks whether given a graph G, does G admit a contact representation using objects of a specific shape. Recently, a natural extension of the recognition problem has been studied. The *partial contact representation extension* problem ask, whether, for a given graph G and a partial contact representation C of G, that is, a contact representation where a subset of the vertices of G are already drawn, C can be extended to form a contact representation of the entire graph G. Since its introduction, this problem has been studied for segment contact graphs [CDK$^+$14] and bar-visibility representations

[CGG$^+$18]. Both problems are NP-complete, however, Chaplick et al. [CKK$^+$21] have recently studied the extension of partial rectangular duals and with the use of regular edge labelings, developed a linear time algorithm that decides whether an extension exists and, for yes-instances, constructs an extension in linear time as well. Related to partial contact representations, the extension problem for geometric intersection representations has also been studied for graphs such as interval graphs [KKO$^+$17], circle graphs [CFK13] and trapezoid graphs [KW17].

Contribution. Our contribution is a follows: We show the close relationship between regular edge labelings and corner edge labelings. In particular we show that an axis-aligned rectangle contact representation of an MTP graph and its corresponding corner edge labeling can be augmented to build a rectangular dual and that the augmented corner edge labeling is a combinatorial description of that rectangular dual. We use this relationship and the results of Chaplick et al. [CKK$^+$21] to solve the partial rectangle contact extension problem, given a corner edge labeling, for an MTP graph G in Chapter 3, that is, we find a rectangle contact extension that admits the given corner edge labeling.

For square dual contact representations, we follow up on Felsner's [Fel13] work and model, given a regular edge labeling, the partial square dual extension problem as a system of linear (in)equations. A solution to that system yields a square dual extension that admits the given regular edge labeling, should such an extension exist. Otherwise there is no solution to the system of linear (in)equalities (Chapter 4).

For partial right-angle axis-aligned triangle contact representations we define a set of necessary conditions that must be met for an extension to exist (Chapter 5).

In Chapter 2 we properly define contact representations, combinatorial descriptions and the partial contact representation extension problem. Chapter 6 gives an overview for the results of this work and open problems for possible future work.

2 Preliminaries

In this chapter we introduce some precise terminology. A *polygon contact system* C is a finite set of polygons in the plane such that no two polygons intersect. The contact system has an *exceptional touching* if two polygons meet at exactly one of their corners each. Throughout this work we only consider polygon contact systems without exceptional touchings.

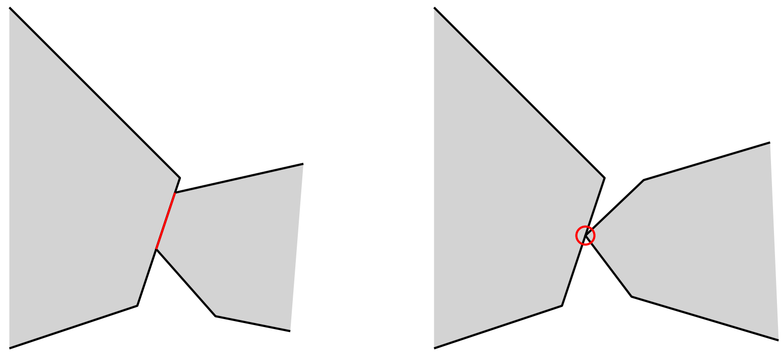


Fig. 2.1: The contact of two polygons. The left figure depicts a contact forming a line segment and the right figure shows a contact consisting of a single point.

Types of Contact. For a polygon contact system without an exceptional touching, every contact is either a line segment or a *single-point contact* as seen in Figure 2.1. Note that a single-point contact is different from an exceptional touching as a single-point contact between two polygons p_1 and p_2 involves a corner of p_1 and a line segment of p_2, whereas an exceptional touching consists of a corner of p_1 and p_2 each. For contacts forming line segments we make a further distinction. We call a contact a *side contact* if the line segment describing the contact is equivalent to one side of the two touching polygons. We say a contact forming a line segment is a *corner contact* otherwise. Side and corner contacts can be seen in Figure 2.2.

Observation 1. *In a polygon contact system without exceptional touchings, every contact between two polygons p_1 and p_2 is either a side contact, corner contact or single-point contact.*

We call an element of the set {side, corner, single-point} the *type of contact* between two polygons.

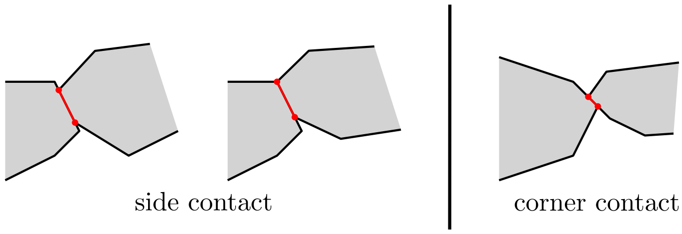


Fig. 2.2: Side contacts (left) and corner contact (right). Note that the case where two polygons make contact at their corners is a side contact.

Combinatorial Descriptions. Two polygon contact systems C_1 and C_2 that have no exceptional touching are called *combinatorial equivalent* if C_1 can be continuously deformed into C_2, such that each intermediate state is a polygon contact system without exceptional touching and the type of contact for all touching polygons is preserved. This gives rise to an equivalence class $\mathcal{C}$ of combinatorial equivalent contact systems and allows for the following observation:

Observation 2. *Let C_1 and C_2 be two combinatorial equivalent polygon contact systems and p_1 and p_2 be two touching polygons in C_1 and C_2. Then the contact between p_1 and p_2 involves the same two sides of p_1 and p_2 in C_1 and C_2 for side and corner contact and the same side and corner for single-point contact.*

When C_1 and C_2 are combinatorial equivalent, every intermediate state when deforming C_1 into C_2 has to be a polygon contact system without exceptional touchings. It is therefore not possible to break the contact between p_1 and p_2, that is, the only deformations in regards to the contact of p_1 and p_2 that do not alter the type of contact are sliding them along their touching sides and stretching the touching sides. However, when trying to change the sides of p_1 and p_2 that make contact, sliding and stretching will always create either the case of an intermediate state having an exceptional touching, or the case of the type of contact between p_1 and p_2 changing.

Combinatorial equivalency also implies that the relative order of the contacts along a side of a polygon does not change as seen in Figure 2.3. We call a data structure that describes a class $\mathcal{C}$ of combinatorial equivalent contact systems a *combinatorial description* of $\mathcal{C}$. In simple terms, while a polygon contact system is defined as coordinates of the points of a set of polygons, the combinatorial description shows which sides s_1 and s_2 of two polygons make contact, the type of contact and the relative order of the contacts along s_1 and s_2.

Contact Representations. For a polygon contact system C, the *contact graph* $G^*(C)$ of C has, for each polygon $p \in C$ a vertex $v(p)$ and, for each pair of polygons $p_1, p_2 \in C$ an edge $(v(p_1), v(p_2))$ if and only if p_1 and p_2 touch. If C has no exceptional touching, then $G^*(C)$ is planar and it inherits a planar embedding from C. Note that for any contact system C the contact graph $G^*(C)$ is unique, that is there is exactly one contact graph for C.

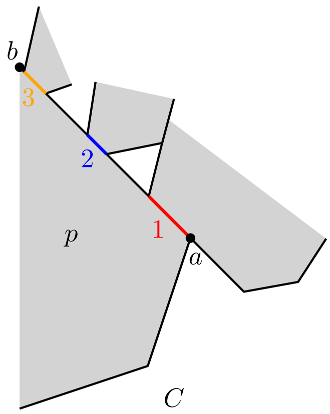


Fig. 2.3: Three contacts of a polygon contact system C. Following the side of polygon p from point a to point b, then the relative order in which the contacts appear is $\{1, 2, 3\}$ for all combinatorial equivalent polygon contact system of C.

Definition 3 (Polygon Contact Representation)**.** *For a planar graph G, a polygon contact system C is called a polygon contact representation $C(G)$ of G, if $G^*(C) = G$.*

For a polygon contact representation $C(G)$ and a vertex $v \in V(G)$ we define $P_{C(G)}(v)$ as the polygon that represents v in $C(G)$. We say G *admits* a polygon contact representation if G is the contact graph of a polygon contact system. Throughout this work we only consider polygon contact representations. For this reason, when we refer to contact representations we always mean polygon contact representations. We may use the two terms interchangeably.

Partial Contact Representation Extension Problem. For a graph G and a subset of vertices $U \subset V(G)$, let $G[U]$ be the sub-graph of G induced by U. Then a *partial polygon contact representation* $A(G[U])$ is a polygon contact representation of $G[U]$. For each $u \in U$, we call $P_{A(G[U])}(u)$ a *fixed* polygon.

The *partial polygon contact representation extension problem* asks whether, given a graph G, a subset $U \subset V(G)$ and a partial polygon contact representation $A(G[U])$, $A(G[U])$ can be extended to a polygon contact representation $C(G)$. In particular we require, for each vertex $u \in U$, that $P_{A(G[U])}(u) = P_{C(G)}(u)$. As a natural extension of the problem, for yes-instances, we further study how the extension $C(G)$ can be constructed. See Figure 2.4 for an example instance of this problem where all polygons are axis-aligned rectangles.

In this work, we study the variant of the partial polygon contact representation extension problem in which we are not only given G, U and $A(G[U])$, but also a combinatorial description of $C(G)$.

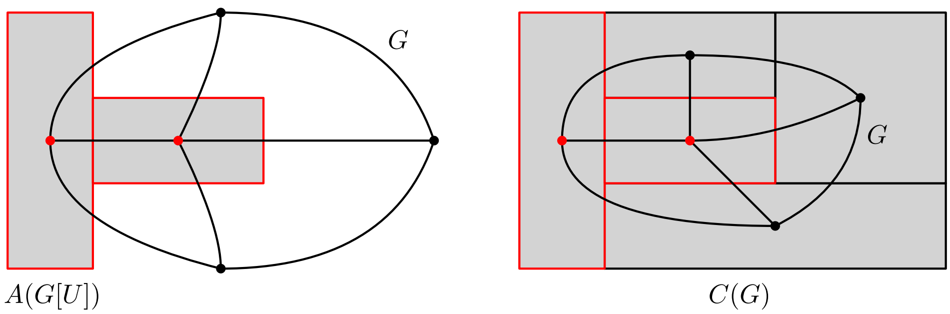


Fig. 2.4: Instance of the partial contact representation extension problem using rectangles. The left figure shows the Graph G, the red vertices the subset $U \subset V(G)$ and the red rectangles the partial contact representation $A(G[U])$. The right figure shows an extension $C(G)$.

3 Rectangle Contact Representations

In this chapter we take a closer look at contact representations of graphs using rectangles. An *axis-aligned rectangle* is a rectangle in the plane, given by a pair of points (x_1, y_1) and (x_2, y_2) in the plane that define the rectangle as the cross product $[x_1, x_2] \times [y_1, y_2]$ of two bounded closed intervals. For sake of simplicity, when we refer to a *rectangle*, we always mean an axis-aligned rectangle. Then a *rectangle contact system* C is a finite set of rectangles, such that no two rectangles intersect. In other words, a rectangle contact system is a polygon contact system where all polygons are rectangles. Since a rectangle is axis-aligned, it is implied that every contact in a rectangle contact system without an exceptional touching forms a line segment. Therefore, there are no single-point contacts between two rectangles. Analogously to polygon contact representations, for a graph G, a *rectangle contact representation* $C(G)$ is a rectangle contact system C with its contact graph $G^*(C)$, such that $G^*(C) = G$.

Following the definition of rectangle contact representations, we can define the partial rectangle contact representation extension problem as a special case of the partial polygon contact representation extension problem.

Definition 4 (Partial Rectangle Contact Representation Extension Problem)**.** *Given a graph G, a subset $U \subset V$, a partial rectangle contact representation $A(G[U])$ and a combinatorial description of a rectangle contact representation $C(G)$, can $A(G[U])$ be extended to $C(G)$.*

Throughout this chapter, we denote the rectangle R of a vertex $v \in V(G)$ that represents v in a rectangle contact representation C as $R(v)$.

3.1 Rectangle Contact Representations and Rectangular Duals

A graph G is called *triangle-free* if it contains no cycle of length 3 and a triangle-free graph G is considered *maximal* if G is not a sub-graph of a triangle-free graph. We consider rectangle contact representations of *maximal, triangle-free, planar* (MTP) graphs and extensions of partial rectangle contact representations of such graphs. As it is common, we assume that the input graph G has exactly four vertices on the outer face which act as a rectangular frame for the rectangle contact representation $C(G)$. We call these vertices *outer* vertices and all other vertices *inner* vertices and name them after their geographic position v_{N},v_{E},v_{S} and v_{W} respectively. This implies that $R(v_{\mathrm{N}})$ is the topmost rectangle, $R(v_{\mathrm{E}})$ the rightmost, $R(v_{\mathrm{S}})$ the bottom most and $R(v_{\mathrm{W}})$ the leftmost. Let G be an MTP graph with exactly four vertices on the outer face. Then each inner face has at least degree 4, since G is triangle-free, and at most degree 5, since

G is maximal. For a rectangle contact representation $C(G)$ that means that each inner face of G is either represented by a rectangle shaped face in $C(G)$ for inner faces with degree 4, or by an L-shaped face in $C(G)$ for inner faces with degree 5. Note that inner faces with degree 5 could also be represented by a rectangle shaped face in $C(G)$, if two rectangles make contact at their corner. However, we ignore this as a degenerate case since every rectangle contact representation of an MTP graph with that case present can be augmented by slightly offsetting the rectangles that make contact at their corners to again form an L-shaped face in $C(G)$. See Figure 3.5 (left and center) for an example of a rectangle contact representation of an MTP graph and note the shapes of the faces of $C(G)$ and their relationship to the faces of G.

Rectangle contact representations of MTP graphs are similar to another rectangle contact representation called rectangular duals. A *rectangular dual* of a graph G is a rectangle contact representation $C(G)$ for which,

(a) no four rectangles share a point, that is $C(G)$ has no exceptional touchings, and

(b) the union of all rectangles is a rectangle.

Figure 3.2 (left and center) shows an example. Note that G only admits a rectangular dual if it is internally triangulated. Otherwise there would be an inner rectangle that does not touch another rectangle on at least one of its sides, which violates property (b) of rectangular duals. It is known that a plane internally triangulated graph admits a rectangular dual if and only if its outer face has degree 4 and it contains no separating triangle, that is there exists no triangle whose removal disconnects the graphs [KK85]. In other words, the graph is 4-connected. Such a graph is called a *properly triangulated planar* (PTP) graph. Every rectangle contact representation can be augmented to form a rectangular dual by adding vertices and edges to the faces of its contact graph to triangulate said contact graph.

3.2 Combinatorial Descriptions of Rectangle Contact Representations

We consider combinatorial descriptions of rectangular duals first. A combinatorial description contains the information which sides of a rectangle make contact, whether the contact is a corner or side contact and the relative order of contacts along each side of each rectangle.

Regular Edge Labeling. Let G be a PTP graph and $C(G)$ a rectangular dual. Since a rectangle $R \in C(G)$ has exactly four sides, the information which sides of two rectangles make contact can be encoded in an orientation and 2-coloring of the edges $E(G)$. Two colors and a binary orientation yields four possible combinations, one for each side of R. For two vertices $v, u \in V(G)$ that share an edge in G, we color the edge $\{u, v\}$ blue if the contact between $R(u)$ and $R(v)$ is a horizontal line segment and we color it red otherwise. We further orient a blue edge $\{u, v\}$ as (u, v) if $R(u)$ and $R(v)$ make contact at the top side of $R(u)$ and the bottom side of $R(v)$. Analogously we orient a red edge $\{u, v\}$ as (u, v) if $R(u)$ and $R(v)$ make contact at the right side of $R(u)$ and the left side of $R(v)$. The relative clockwise order of the contacts is then depicted as the clockwise order of the edges around a vertex as seen in Figure 3.1. In such a representation all edges of a vertex v must abide by the following clockwise local coloring order around a vertex v:

- Outgoing blue edges,
- Outgoing red edges,
- Incoming blue edges,
- Incoming red edges.

For the outer vertices, v_{N} is adjacent to only incoming blue edges, v_{E} to only incoming red edges, v_{S} to only outgoing blue edges and v_{W} to only outgoing red edges. This local coloring rule can be seen in Figure 3.6 (left). Figure 3.2 shows an example of a complete regular edge labeling for a rectangular dual.

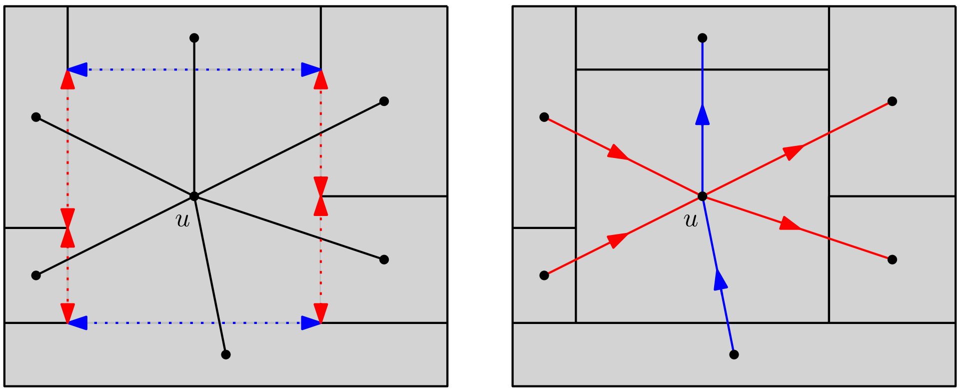


Fig. 3.1: The regular edge labeling as combinatorial description of a rectangular dual for a single vertex u. The coloring and orientation of the edges adjacent to u shows which sides of the rectangles make contact. The clockwise order of the edges around u shows the relative order of the contacts.

This combinatorial description is called a *regular edge labeling* (REL) and was introduced by Kant and He [KH97]. It is also known as *transversal structure*. For a graph G, the type of contact is not encoded in the edges of a regular edge labeling REL(G), but can be obtained by looking at the inner triangles of REL(G). For example, let $v \in V(G)$ be a vertex of G and $u_t \in V(G)$ the rightmost top neighbor of v, that is, (v, u_t) is a blue edge and going around v clockwise and starting from (v, u_t), the next edge encountered is an outgoing red edge. Analogously let u_b be the rightmost bottom neighbor of v, that is, (u_b, v) is a blue edge and the next edge encountered counter clockwise is an outgoing red edge. Then to determine the type of contact between $R(v)$ and, for a vertex $v' \in V(G)$ with (v, v') is a red edge, another rectangle $R(v')$, we look at the edges (u_t, v')

and (u_b, v'). If either both $(u_t, v'), (u_b, v') \notin E(G)$ or both $(u_t, v'), (u_b, v') \in E(G)$, then $R(v)$ and $R(v')$ make side contact. Next we consider the case where only one of (u_t, v') and (u_b, v') is an edge of G. Without loss of generality, we assume that $(u_t, v') \in E(G)$ and $(u_b, v') \notin E(G)$. Should (u_t, v') be a blue edge, then $R(v)$ and $R(v')$ make side contact and corner contact otherwise. For a PTP graph G, the regular edge labeling $\mathrm{REL}(G)$ therefore is a combinatorial description of a rectangular dual $C(G)$.

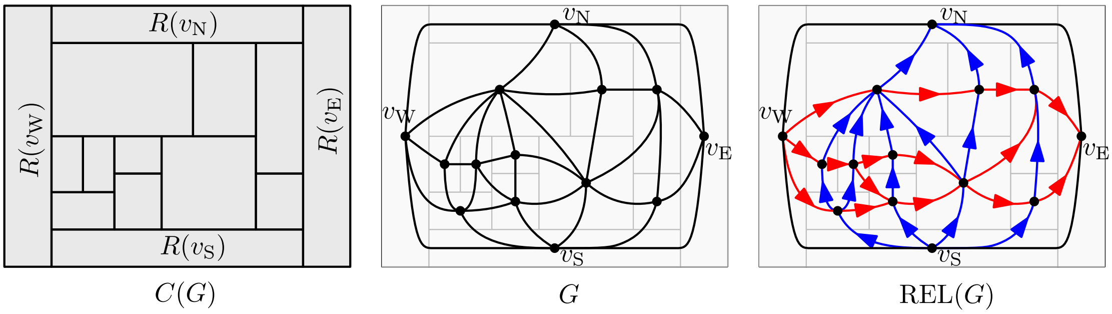


Fig. 3.2: A rectangular dual $C(G)$ (left) for the graph G (center) and the corresponding $\mathrm{REL}(G)$ (right).

At first it seems like regular edge labelings can be adapted to build a combinatorial description of rectangle contact systems of MTP graphs as well by re-utilizing the same coloring and orientation rules of regular edge labelings for rectangular duals. Since a combinatorial description preserves the type of contact of two rectangles however, there exist rectangle contact representations of MTP graphs that have the same regular edge labeling but are not combinatorial equivalent as seen in Figure 3.3.

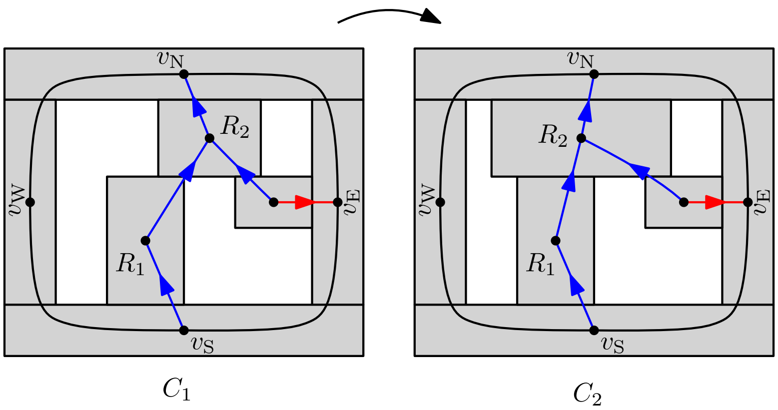


Fig. 3.3: The transformation of a rectangle contact system C_1 into C_2. Note that the regular edge labeling does not change, however C_1 and C_2 are not combinatorial equivalent since the type of contact between the rectangles R_1 and R_2 changes from corner to side contact.

We therefore use a different combinatorial description of rectangle contact representations of MTP graphs. Instead of using the sides that make contact to describe the rectangle contact representation, it is possible to use the pair of corners that mark the end points of a contact line segment as seen in Figure 3.4. This idea was used by

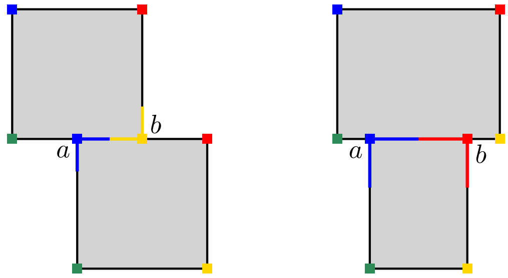


Fig. 3.4: Two examples of contacts described by rectangle corners a and b respectively.

Klawitter et al. [KNU15] to define a combinatorial description of rectangle contact representations of MTP graphs called *corner edge labeling* in their work regarding rectangle contact representations of MTP graphs.

Corner Edge Labeling. For an MTP graph G, the corner edge labeling $\mathrm{CEL}(G)$ for a rectangle contact representation $C(G)$ focuses on the contact at the four corners of each rectangle. Let $V(G)$ denote the vertices, $E(G)$ the edges and $F(G)$ the inner faces of G. A corner edge labeling $\mathrm{CEL}(G)$ is constructed from G in the following steps:

1. Each edge in $E(G)$ is replaced by a pair of parallel edges, called an *edge pair*. For each of the four outer vertices, two half edges are added.
2. For each inner face $f \in F(G)$, let $A(f)$ be the set of vertices that are adjacent to f. Then for each $f \in F(G)$ a vertex $v(f)$ is added with single edges $(v(f), u)$ for each $u \in A(f)$. Let $W(G)$ be the set of vertices added in this step.
3. The edges are oriented, such that every vertex $v \in V(G) \cup W$ has out degree exactly 4.
4. Each edge in $E(G)$ is colored $\{0, 1, 2, 3\}$, such that
    - around each vertex $v \in V(G)$ the four outgoing edges of v are colored 0, 1, 2, 3 in that clockwise order and
    - all (incoming) edges between two outgoing edges colored c and $c+1$ clockwise around v are colored $c+2$ or $c+3$, $c \in \{0, 1, 2, 3\}$ and all indices modulo 4.

This induces an inner triangulation of G where every inner face is a triangle or 2-gon for the edge pairs. Figure 3.5 shows an example for a corner edge labeling and Figure 3.6 (right) shows the local coloring rules for each inner vertices and the four outer vertices. Klawitter et al. [KNU15] have shown that, for any given MTP graph G, there exists a corner edge labeling $\mathrm{CEL}(G)$ and that it is a combinatorial description of a rectangle contact representation $C(G)$.

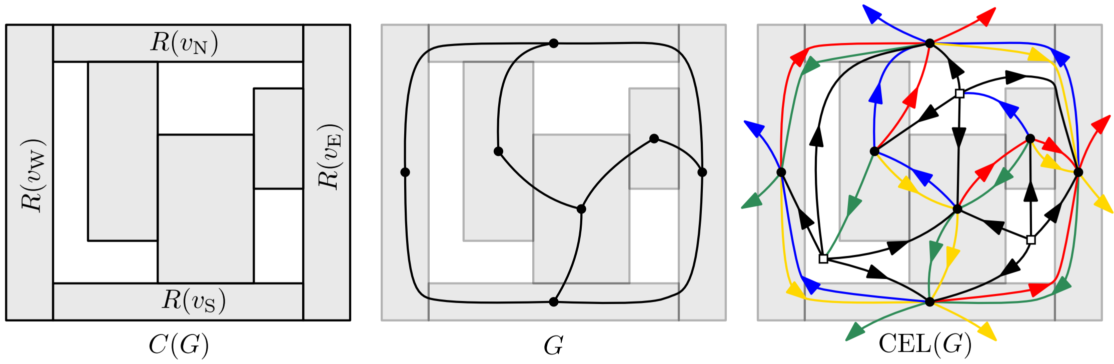


Fig. 3.5: A rectangle contact representation $C(G)$ (left) for an MTP graph G (center) and the corner edge labeling CEL(G) (right).

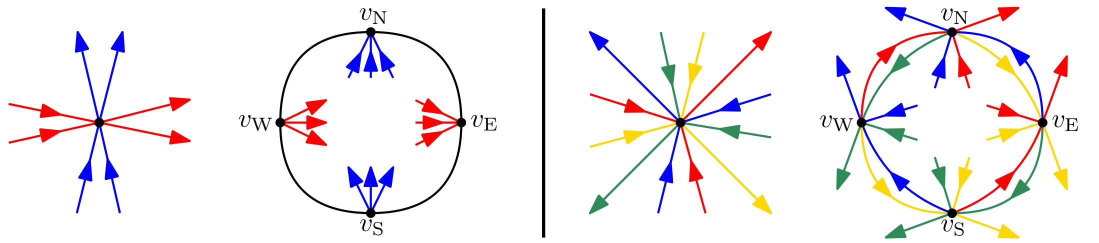


Fig. 3.6: Local coloring rules for regular edge labelings (left) and corner edge labelings (right)

3.3 From Rectangle Contact Representation to Rectangular Dual

In this section we show the close relationship between rectangle contact representations of MTP graphs and rectangular duals. Every rectangle contact system C can be augmented to form a rectangular dual by slicing the inner faces of C and adding those slices as rectangles. Since every inner face of an MTP graph G has either 4 or 5 vertices adjacent to it, only at most one slice is necessary to cut the faces of $C(G)$ into rectangles. Precisely, faces with only 4 adjacent vertices require no slicing since they will always appear as rectangles in a rectangle contact representation while faces with 5 adjacent vertices required exactly one slice.

For an MTP graph G and a rectangle contact representation $C(G)$ the, rectangular dual D_G is obtained by augmenting $C(G)$ in the following way: Each inner face f of $C(G)$ with 4 adjacent rectangles is replaced by a single rectangle that makes contact with all four neighbors of f. Each inner face g of $C(G)$ with 5 adjacent rectangles is replaced by two rectangles R_1 and R_2, such that R_1 and R_2 have a bottom-to-top contact, i.e. g is segregated by a horizontal slice.

The shape of an inner face of $C(G)$ is encoded in the corner edge labeling CEL(G), so the corner edge labeling can be augmented as well to form a combinatorial description of D_G. Let f be a face of $C(G)$ with 5 adjacent vertices and vertex $w \in W$ the vertex representing f in CEL(G). Then w is replaced by two vertices u_1 and u_2 that

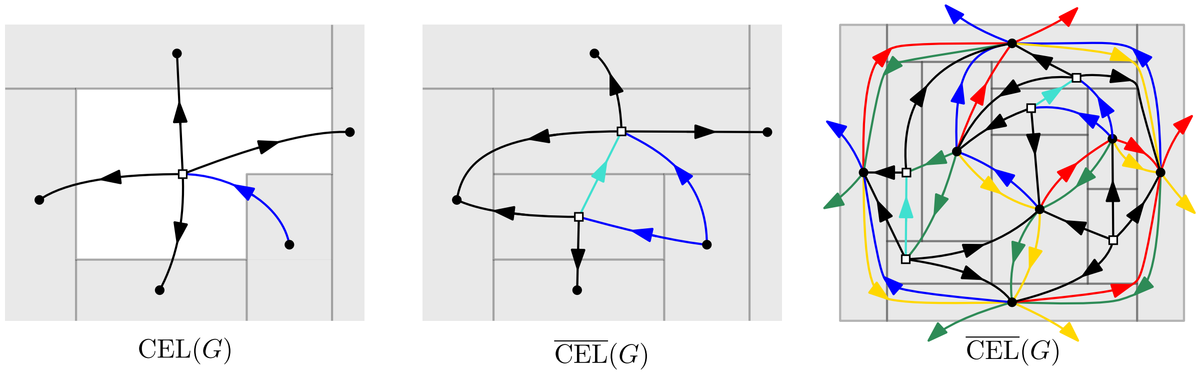


Fig. 3.7: The augmentation of a rectangle contact representation of a MTP graph G to a rectangular dual and the corresponding augmentations of the corner edge labeling CEL(G) to the augmented corner edge labeling $\overline{\text{CEL}}(G)$

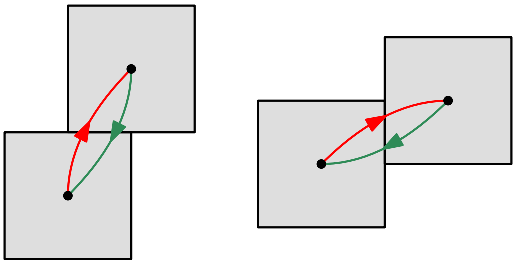


Fig. 3.8: Top-to-bottom and left-to-right contact of two rectangles. Note the clockwise order of the edges around the vertices swapping.

are connected by a single, directed edge $\{u_1, u_2\}$, such that the orientation of $\{u_1, u_2\}$ shows the top-to-bottom relationship of $R(u_1)$ and $R(u_2)$ in D_G. This relationship is determined as follows: The vertex w has exactly four outgoing edges and one incoming colored edge (v, w) in CEL(G). We consider the case where (v, w) has color 0 and that the color 0 represents the contact of the top left corner of $R(v)$. Then $\{u_1, u_2\}$ is oriented (u_1, u_2) and, starting from (v, w) and going around w clockwise, u_1 inherits the first two encountered outgoing edges of w. Starting from (v, w) and going around w counter clockwise, u_2 inherits the first three encountered outgoing edges of w. The edge (v, w) is replaced by the two edges (v, u_1) and (v, u_2), which are colored 0. Note that one outgoing edge of w is inherited twice. We call the vertices $\{u_1, u_2\}$ a *face-vertex pair* and the edges $\{(v, u_1), (v, u_2)\}$ the *uni-colored edges* of $\{u_1, u_2\}$. For the cases of (v, w) being colored 1, 2 or 3 CEL(G) can be augmented analogously with similar inheritance rules. See Figure 3.7 for an example of the operation described in this paragraph. We call the resulting structure the *augmented corner edge labeling* $\overline{\text{CEL}}(G)$ of G. If we imagine a contact representation $C(G)$ of an MTP graph G and the vertices added for the corner edge labeling and augmented corner edge labeling as rectangles to extend $C(G)$ to a rectangular dual, we see that the augmented corner edge labeling $\overline{\text{CEL}}(G)$ induces a PTP graph.

Observation 5. *Let G be an MTP graph, $C(G)$ a rectangle contact representation, D_G the rectangular dual obtained by horizontally slicing the faces of $C(G)$ and $\overline{\mathrm{CEL}}(G)$ an augmented corner edge labeling. Then $\overline{\mathrm{CEL}}(G)$ induces the contact graph of $G^*(D_G)$.*

3.4 Augmented Corner Edge Labeling and Regular Edge Labeling

Following up from the definition of augmented corner edge labelings and the relationship between rectangle contact representations of MTP graphs and rectangular duals, we show the close relationship between augmented corner edge labelings and regular edge labelings. Let G be a MTP graph and $\mathrm{CEL}(G)$ a corner edge labeling of G. Then the augmented corner edge labeling $\overline{\mathrm{CEL}}(G)$ can be transformed into the regular edge labeling $\mathrm{REL}\,(G^*(D_G))$ and vice versa. We describe the transformation processes in the following paragraphs.

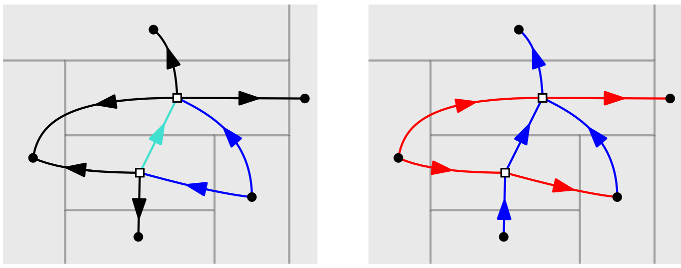
From Augmented Corner Edge Labeling to Regular Edge Labeling.

Proposition 6. For an MTP graph G, every augmented corner edge labeling $\overline{\mathrm{CEL}}(G)$ induces a regular edge labeling $\mathrm{REL}\,(G^*(D_G))$.

Proof. We construct a regular edge labeling $\mathrm{REL}\,(G^*(D_G))$ from $\overline{\mathrm{CEL}}(G)$ as follows: For two vertices $u, v \in V(G)$, let $((u,v)_i, (u,v)_j)$ be an edge pair in $\overline{\mathrm{CEL}}(G)$. The coloring and orientation of $((u,v)_i, (u,v)_j)$ directly translates to a bottom-to-top or left-to-right relationship between the rectangles $R(u)$ and $R(v)$ as seen in Figure 3.8. We replace $((u,v)_i, (u,v)_j)$ with a single, directed and 2-colored edge that represents this relationship as in the definition of regular edge labelings. The resulting structure is similar to the partial regular edge labeling seen in Figure 3.3.

By definition of face-vertex pairs, it is known that, for a face-vertex pair $\{u_1, u_2\}$, the rectangles $R(u_1)$ and $R(u_2)$ have a top-to-bottom relationship that is encoded in the orientation of the edge (u_1, u_2) in $\overline{\mathrm{CEL}}(G)$. We therefore color (u_1, u_2) blue. One vertex of $\{u_1, u_2\}$ has, disregarding the orientation of the edges, four neighbors in $\overline{\mathrm{CEL}}(G)$ and the other has five. Without loss of generality, let u_1 be the vertex with four neighbors and u_2 the one with five. Since the color and orientation of the directed edge (u_1, u_2) is already known, we can deduce the orientation and coloring of the other three edges adjacent to u_1 by the local coloring rules of regular edge labelings seen in Figure 3.6. For the uni-colored edges $\{(v, u_1), (v, u_2)\}$ this means, that the edge $\{v, u_1\}$ gets colored and oriented. The other edge $\{v, u_2\}$ is then colored blue and inherits the orientation from $\{u_1, u_2\}$. If $\{u_1, u_2\}$ is oriented (u_1, u_2), then $\{v, u_2\}$ is oriented as (v, u_2) and as (u_2, v) otherwise. This leaves u_2 with either two incoming or two outgoing blue edges and the other three edges adjacent to u_2 can be colored according to the local coloring rules of regular edge labelings, see Figure 3.9.

For a vertex $w \in W$ with four adjacent vertices, that is a vertex representing a rectangular face in $\mathrm{CEL}(G)$, the coloring and orientation of its four outgoing edges can be deduced from the edge pairs of the adjacent vertices as seen in Figure 3.11. Since the local coloring rules are satisfied for each vertex, the resulting structure is a regular edge labeling REL $(G^*(D_G))$. $\square$



$\overline{\text{CEL}}(G)$ REL $(G^*(D_G))$

Fig. 3.9: A face-vertex pair in the augmented corner edge labeling $\overline{\text{CEL}}(G)$ and the orientation and coloring of the adjacent edges in the regular edge labeling REL $(G^*(D_G))$.

From Regular Edge Labeling to Augmented Corner Edge Labeling.

Proposition 7. For an MTP graph G, every regular edge labeling REL $(G^*(D_G))$ induces an augmented corner edge labeling $\overline{\text{CEL}}(G)$.

Proof. For any rectangular dual all inner rectangle contacts form a T-shape. Therefore, there are exactly four possible contacts between three adjacent rectangles as seen in Figure 3.10. Let v, u and w be vertices that form a 3-cycle in the regular edge labeling of REL $(G^*(D_G))$. Since G is triangle-free, at least one of v, u and w represents a face of the rectangle contact representation of G. Let v and u be vertices that represent a rectangle in the contact representation of G, that is, let $v, u \in V(G)$ be vertices of G and $w \in W$ be a vertex that represents either a vertex of a face-vertex pair or a rectangular face in the rectangle contact representation. Then the coloring and orientation of the edges $\{v, u\}$, $\{v, w\}$ and $\{u, w\}$ in REL $(G^*(D_G))$ induces the contact of one corner of either v or u. The union of all such induced corner contacts for all inner 3-cycles describes the corner contact between all rectangles of the contact representation of G, and thus, the corner edge labeling CEL(G).

For vertices $w \in W$ that are not part of a face-vertex pair, it is easily observable that the operation depicted in Figure 3.11 is reversible, that is, the four edges adjacent to w are replaced by the four uncolored and outgoing edges that are in $\overline{\text{CEL}}(G)$.

For a face-vertex pair $\{u_1, u_2\}$, the coloring of the uni-colored edges $\{(v, u_1), (v, u_2)\}$ and the orientation of $\{u_1, u_2\}$ can be obtained again by corner contact induction. All other edges adjacent to u_1 and u_2 are oriented as outgoing. $\square$

As a short excursion, note that since corner edge labelings and 4-*gon contact structures* introduced by Felsner et al. [FSS18] are similarly defined to corner edge labelings, Proposition 6 and Proposition 7 emphasize the equivalence of 4-gon contact structures and regular edge labelings.

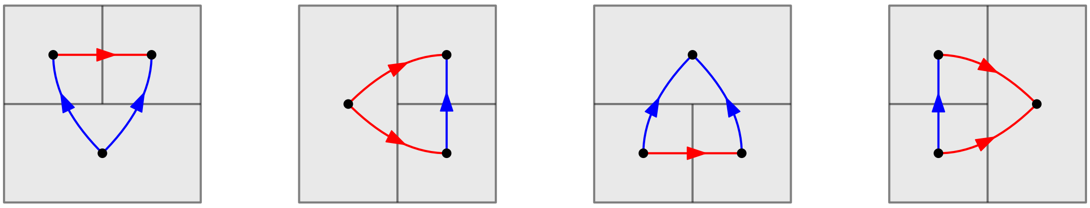


Fig. 3.10: The four contact types in a regular edge labeling inducing the corner contact of the rectangles.

Chaplick et al. have shown that the partial rectangle contact representation extension problem for a PTP graph, and given a regular edge labeling $\mathrm{REL}(G)$, can be solved in linear time.

Theorem 8 (Chaplick et al. [CKK+21])**.** *Given a PTP graph G and a regular edge labeling $\mathrm{REL}(G)$, the partial rectangular dual extension problem can be solved in linear time. For yes-instances, an explicit rectangular dual can be constructed in linear time.*

Note that following from Euler's formula for planar graphs, that is, the number of faces and edges is linear in the number of vertices, building an augmented corner edge labeling and transforming an augmented corner edge labeling into a regular edge labeling can be done in linear time. This allows us to solve the partial rectangle contact representation problem for MTP graphs in linear time in the following way: For an MTP graph G and given a corner edge labeling $\mathrm{CEL}(G)$, we transform $\mathrm{CEL}(G)$ into the augmented corner edge labeling $\overline{\mathrm{CEL}}(G)$ and $\overline{\mathrm{CEL}}(G)$ into the regular edge labeling $\mathrm{REL}\left(G^*(D_G)\right)$. The linear time algorithm of Chaplick et al. can then be used to solve the partial rectangular dual problem for the graph $G^*(D_G)$ and its constructed regular edge labeling $\mathrm{REL}\left(G^*(D_G)\right)$. Since G is a sub-graph of $G^*(D_G)$, this also yields a solution to the partial rectangle contact representation problem for MTP graphs.

Theorem 9. *Let G be an MTP graph and $\mathrm{CEL}(G)$ a corner edge labeling of G, then the partial rectangle contact representation problem can be solved in linear time. For yes-instances, an explicit rectangle contact representation of G can be constructed within the same time bound.*

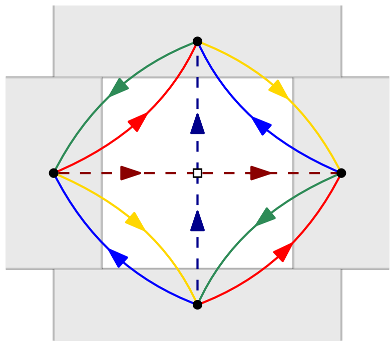


Fig. 3.11: A rectangular face in a rectangle contact representation of an MTP graph G with an underlying corner edge labeling $\mathrm{CEL}(G)$ (normal lines). The dashed lines show the orientation and coloring of the edges in the corresponding regular edge labeling $\mathrm{REL}\,(G^*(D_G))$.

4 Extending Partial Square Duals

Following up from the topic of extending partial rectangle contact representations, we consider drawing square duals of a graph in this chapter. A *square dual* is a rectangular dual where all rectangles are squares and the union of all squares is a square. Throughout this chapter we denote a square representing a vertex u in a contact representation as $S(u)$. Similarly to rectangle contact representations and rectangular duals, we assume that a graph G has exactly four vertices on its outer face and we name them $v_\mathrm{N}, v_\mathrm{E}, v_\mathrm{S}, v_\mathrm{W}$ by the geographic position of their squares respectively. Unlike those *outer* vertices for rectangular duals, we however further assume that all squares of outer vertices $S(v_\mathrm{N}), S(v_\mathrm{E}), S(v_\mathrm{S}), S(v_\mathrm{W})$ have the same side length as seen in Figure 4.1. This implies that there is no square dual contact representation of G which is why we apply the conditions for square duals only to inner vertices and their squares. That is, we consider the sub graph G' induced by the vertices $V(G) \setminus \{v_\mathrm{N}, v_\mathrm{E}, v_\mathrm{S}, v_\mathrm{W}\}$. In particular, we give a solution to the following problem in this chapter.

Definition 10 (Partial Square Dual Extension Problem)**.** *Given a PTP graph G, a subset $U \subset V$, a partial square dual $A(G[U])$ and a regular edge labeling* $\mathrm{REL}(G)$*, can $A(G[U])$ be extended to a square dual $C(G)$, such that* $\mathrm{REL}(G)$ *is a combinatorial description of $C(G)$.*

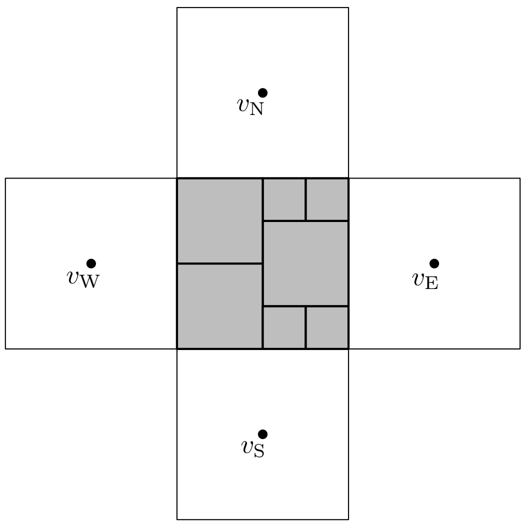


Fig. 4.1: The four outer squares that encapsulate the square dual in the middle.

Given a PTP graph G and a regular edge labeling REL(G), Felsner [Fel13] proposed a system of linear equalities, such that a solution to the system yields a square dual $C(G)$ that admits REL(G). Such linear programs are a well known tool to solve difficult problems. A *linear program* consists of a linear objective function and a set of linear (in)equalities, called *linear constraints*, for a set of variables. A solution to a linear program is a non-negative assignment of the variables, such that all linear constraints are satisfied and the objective function is optimized. Note that due to their objective function, linear programs inherently model optimization problems. However, they can also be used for decision problems, that is, we are not interested in the objective function, but just want to know whether a valid assignment of the variables that satisfies all constraints exists. For solving the square dual recognition problem using a linear program, Felsner used variables for each edge of the input PTP graph G which would decide the length of the contact of the two corresponding rectangles. Felsner's approach is the following: For every vertex $u \in V(G)$ we define $B(u)$ as the set of neighbors of u where for all $v \in B(u)$ the edge (u, v) is blue in REL(G). That is $S(u)$ and $S(v)$ make bottom-to-top contact with $S(u)$ being the bottom square. Analogously we define $R(u)$ as the set of neighbors of u where for all $v \in B(u)$ the edge (u, v) is red in REL(G). Further, let $\alpha(u)$ be to right-most top neighbor of u, $\beta(u)$ be the left-most top one, $\gamma(u)$ the top-most right neighbor and $\delta(u)$ the bottom-most right neighbor of u, see Figure 4.2 left. We define $B'(u)$ and $R'(u)$ analogously for the neighbors of u with incoming blue and incoming red edges respectively.

Felsner then uses variables l_e, with $e \in E(G)$ an edge of G, that describe the length of a contact between the squares of the two vertices of e, see Figure 4.2 right. Then the sum $\sum_{v \in B(u)} l_{(u,v)}$ is the side length of $S(u)$. The same is true for $R(u)$, $B'(u)$, and $R'(u)$ for the right, bottom, and left side of $S(u)$ respectively. To get proper squares, all four sides of a square have to be the same length:

$$\sum_{a \in B(u)} l_{(u,a)} = \sum_{b \in R(u)} l_{(u,b)} = \sum_{c \in B'(u)} l_{(u,c)} = \sum_{d \in R'(u)} l_{(u,d)}. \tag{4.1}$$

This approach is however not sufficient for the partial square dual extension problem, since it is difficult to represent the given, fixed squares in the model. Since the length of each fixed square is known, it is possible to use the sums in 4.1 to guarantee that the rectangle has the correct size. However there is no easily observable correlation between the length of a contact of two squares and their exact geometric position, i.e. their coordinates. We therefore propose a different linear program.

4.1 Linear Program for the Partial Square Dual Extension Problem

A square can be described by the x and y coordinates of one of its corners and the length of its side, which are three numeric properties. Since a variable in a linear program holds a numeric value, it is therefore beneficial to choose variables that directly represent these

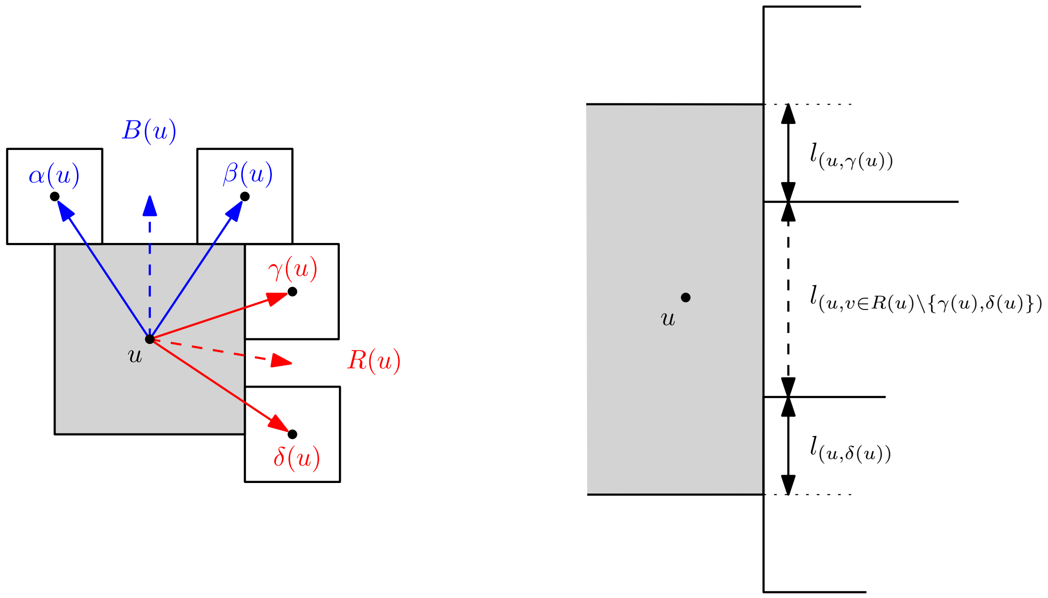


Fig. 4.2: The sets $B(u)$ and $R(u)$ of a vertex u (left). The variables of type l_e representing the length of a contact (right).

properties. We introduce, for a vertex $u \in V(G)$, variables of type x_u and y_u that define the coordinates of the bottom left corner of $S(u)$ and variables w_u, with linear constraint $w_u > 0$, to represent the side length of $S(u)$. In the context of the partial square dual extension problem it is now straightforward to model the fixed squares: Let $u \in U$ be a vertex of the subset $U \subset V(G)$, $A(G[U])$ a partial square dual and $S(u) \in A(G[U])$ the fixed square of u. Further, let $x'_{S(u)}$ and $y'_{S(u)}$ be the x-y-coordinates of the lower left corner of $S(u)$ and $w'_{S(u)}$ its side length. Then the following constraints will guarantee that the fixed squares are represented properly in the linear program and the resulting drawing:

$$x_u = x'_{S(u)} \qquad \forall u \in U, \tag{4.2}$$

$$y_u = y'_{S(u)} \qquad \forall u \in U, \tag{4.3}$$

$$w_u = w'_{S(u)} \qquad \forall u \in U. \tag{4.4}$$

Any assignment of the variables that satisfies these constraints yields a drawing where the fixed squares are properly represented and where every rectangle is a square. However these constraints do not make the resulting drawing a square dual, as they do not constrain the contact of two squares.

Vertical Alignment. A *vertical* contact is the contact of two squares at their left and right side respectively. They are the contacts represented by the red edges in the regular edge labeling REL(G). Let $u, v \in V(G)$ be two vertices. For their respective squares

$S(u) = (x'_{S(u)}, y'_{S(u)}, w'_{S(u)})$ and $S(v) = (x'_{S(v)}, y'_{S(v)}, w'_{S(v)})$ to touch at their left and right side respectively, the following two conditions must be met:

- The x-coordinates of the left and right side of $S(u)$ and $S(v)$ respectively must be equal as can be seen in Figure 4.3 (left).
- The closed intervals $[y'_{S(u)}, y'_{S(u)} + w'_{S(u)}]$ and $[y'_{S(v)}, y'_{S(v)} + w'_{S(v)}]$ must overlap as can be seen in Figure 4.3 (right).

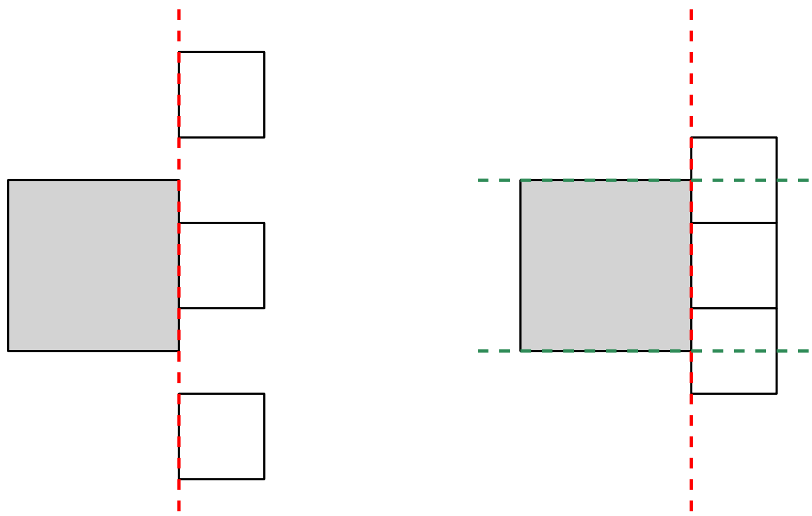


Fig. 4.3: Vertical alignment conditions for a square.

The first condition can be directly translated to a single type of linear constraint as indicated by the red (vertical) lines in Figure 4.3:

$$x_u + w_u = x_v \qquad \forall u \in V(G'),\ v \in R(u). \tag{4.5}$$

The second condition requires two constraints as indicated by the green (horizontal) lines in Figure 4.3. It is sufficient to only constrain the first and last element of $R(u)$ since the correct vertical contact of $S(u)$ and $S(v)$ with $v \in R(u) \setminus \{\gamma(u), \delta(u)\}$ is covered by the horizontal alignment of the squares of all vertices in $R(u)$:

$$y_{\delta(u)} + w_{\delta(u)} > y_u \qquad \forall u \in V(G'), \tag{4.6}$$
$$y_u + w_u > y_{\gamma(u)} \qquad \forall u \in V(G'). \tag{4.7}$$

Note that $S(u)$ may have only one right neighbor $\gamma(u) = \delta(u)$.

Horizontal Alignment. Analogously to vertical contacts a *horizontal* contact is is the contact of two squares at their top and bottom side respectively. These contacts are represented by the blue edges in $\mathrm{REL}(G)$. For two squares $S(u) = (x'_{S(u)}; y'_{S(u)}; w'_{S(u)})$ and $S(v) = (x'_{S(v)}; y'_{S(v)}; w'_{S(v)})$ to touch at their top and bottom side respectively, the following two conditions must be met. Note the close relationship to vertical contacts:

- The y-coordinates of the top and bottom side of $S(u)$ and $S(v)$ respectively must be equal as can be seen in Figure 4.4 (left).
- The closed intervals $[x'_{S(u)}, x'_{S(u)} + w'_{S(u)}]$ and $[x'_{S(v)}, x'_{S(v)} + w'_{S(v)}]$ must overlap as can be seen in Figure 4.4 (right).

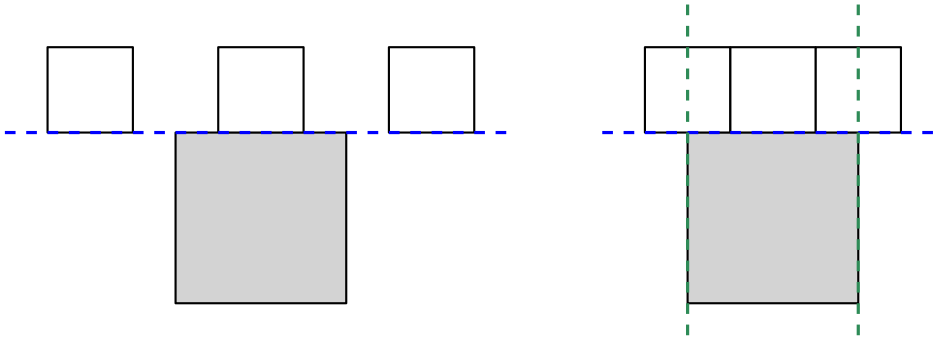


Fig. 4.4: Horizontal alignment conditions for a square.

The first condition is again represented by a single constraint that is visualized by the blue (horizontal) lines in Figure 4.4:

$$y_u + w_u = y_v \qquad \forall u \in V(G'),\ v \in B(u), \tag{4.8}$$

whereas the second condition requires two constraints represented by the green (vertical) lines in Figure 4.4:

$$x_{\beta(u)} < x_u + w_u \qquad \forall u \in V(G'), \tag{4.9}$$

$$x_u < x_{\alpha(u)} + w_{\alpha(u)} \qquad \forall u \in V(G'). \tag{4.10}$$

This concludes the set of constraints for the contact of all inner squares. A summary of the vertical and horizontal constraints can be seen in Figure 4.5.

Outer Squares. For a square dual, the union of all squares is a square. We introduced four squares $S(v_\mathrm{N}), S(v_\mathrm{E}), S(v_\mathrm{S}), S(v_\mathrm{W})$ placed around the dual that act as a frame as seen in Figure 4.1. We use those squares to force the resulting drawing into a proper square dual by utilizing the analogous constraints for horizontal and vertical alignment.

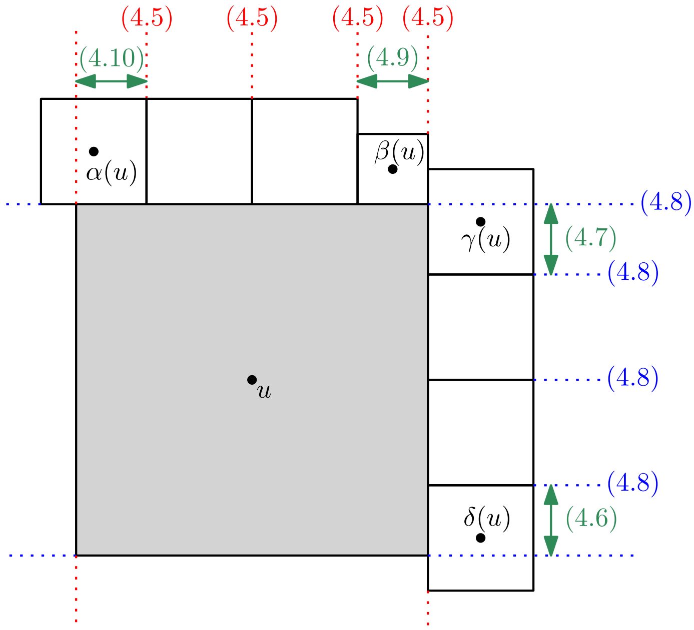


Fig. 4.5: Visual summary of the vertical and horizontal alignment constraints.

Since v_{S} has only blue outgoing edges in $\mathrm{REL}(G)$ and v_{W} has only red outgoing edges, it is possible to the define the sequences $R(v_{\mathrm{W}})$ and $B(v_{\mathrm{S}})$ and use constraints analogously to inner squares.

$$x_{v_{\mathrm{W}}} + w_{v_{\mathrm{W}}} = x_u \qquad \forall u \in R(v_{\mathrm{W}}) \tag{4.11}$$
$$y_{\delta(v_{\mathrm{W}})} + w_{\delta(v_{\mathrm{W}})} > y_{v_{\mathrm{W}}} \tag{4.12}$$
$$y_{v_{\mathrm{W}}} + w_{v_{\mathrm{W}}} > y_{\gamma(v_{\mathrm{W}})} \tag{4.13}$$

$$y_{v_{\mathrm{S}}} + w_{v_{\mathrm{S}}} = y_u \qquad \forall u \in B(v_{\mathrm{S}}) \tag{4.14}$$
$$x_{\beta(v_{\mathrm{S}})} < x_{v_{\mathrm{S}}} + w_{v_{\mathrm{S}}} \tag{4.15}$$
$$x_{v_{\mathrm{S}}} < x_{\alpha(v_{\mathrm{S}})} + w_{\beta(v_{\mathrm{S}})} \tag{4.16}$$

For v_{E} and v_{N} an equivalent set of constraints can be defined using the sets of neighbors with incoming edges $R'(v_{\mathrm{E}})$ and $B'(v_{\mathrm{N}})$ respectively. Let $\gamma'(v_{\mathrm{E}})$ be the topmost left neighbor of v_{E} and $\delta'(v_{\mathrm{E}})$ the bottom-most left one. Then the following constraints confine the vertical contact of $S(v_{\mathrm{E}})$ and each $S(u)$ with $u \in R'(v_{\mathrm{E}})$:

$$x_u + w_u = x_{v_{\mathrm{E}}} \qquad \forall u \in R'(v_{\mathrm{E}}), \tag{4.17}$$
$$y_{\delta'(v_{\mathrm{E}})} + w_{\delta'(v_{\mathrm{E}})} > y_{v_{\mathrm{E}}}, \tag{4.18}$$
$$y_{v_{\mathrm{E}}} + w_{v_{\mathrm{E}}} > y_{\gamma'(v_{\mathrm{E}})}. \tag{4.19}$$

Further let $\alpha'(v_\mathrm{N})$ be the leftmost bottom neighbor of v_N and $\beta'(v_\mathrm{N})$ the rightmost bottom one. Then the following constraints confine the horizontal contact of $S(v_\mathrm{N})$ and each $S(u)$ with $u \in B'(v_\mathrm{N})$:

$$\begin{aligned} y_u + w_u &= y_{v_\mathrm{N}} & \forall u \in B'(v_\mathrm{N}), \quad &(4.20)\\ x_{\beta'(v_\mathrm{N})} &< x_{v_\mathrm{N}} + w_{v_\mathrm{N}}, & &(4.21)\\ x_{v_\mathrm{N}} &< x_{\alpha'(v_\mathrm{N})} + w_{\alpha'(v_\mathrm{N})}. & &(4.22) \end{aligned}$$

Lastly the outer squares have the same side length:

$$w_{v_\mathrm{N}} = w_{v_\mathrm{E}} = w_{v_\mathrm{S}} = w_{v_\mathrm{W}}. \quad (4.23)$$

These sets of horizontal and vertical contact constraints for the outer squares guarantee that the union of all squares is a rectangle. It is possible, that there is a gap between two outer squares as seen in Figure 4.6. To only allow square duals to be the result of the LP, constraints that force the outer squares to make contact at their corners are added, that is we force exceptional touchings:

$$\begin{aligned} x_{v_\mathrm{S}} + w_{v_\mathrm{S}} &= x_{v_\mathrm{E}}, & (4.24)\\ y_{v_\mathrm{E}} + w_{v_\mathrm{E}} &= y_{v_\mathrm{N}}, & (4.25)\\ x_{v_\mathrm{N}} &= x_{v_\mathrm{W}} + w_{v_\mathrm{W}}, & (4.26)\\ y_{v_\mathrm{W}} &= y_{v_\mathrm{S}} + w_{v_\mathrm{S}}. & (4.27) \end{aligned}$$

This concludes the set of constraints for the linear program. Since the partial square dual extension problem is not an optimization problem, there is no need for an objective function. To show the existence of a partial square dual extension it is just of interest whether there is a configuration of the variables that satisfies all constraints. If such a configuration exists, then it is trivial to get the resulting drawing of the complete square dual by checking the assigned values of a solution to the linear program. For a vertex $u \in V(G')$ the values of x_u and y_u designate the x-y-coordinates of its square $S(u)$ and the value of w_u decides its side length. Since a linear program can be solved in polynomial time, we can conclude this section with the following theorem:

Theorem 11. *The partial square dual extension problem can be solved in polynomial time. For yes-instances a square dual extension can be computed within the same time bound.*

4.2 Linear Program as a System of Difference Constraints

In this section, we study the relationship of the LP proposed in Section 4.1 and a solution to the partial rectangular dual extension problem. A *system of difference constraints* is a linear program where, for two variables x_i and x_j and a constant b_k, every constraint can be written as $x_i - x_j \leq b_k$. A system of difference constraints can be interpreted

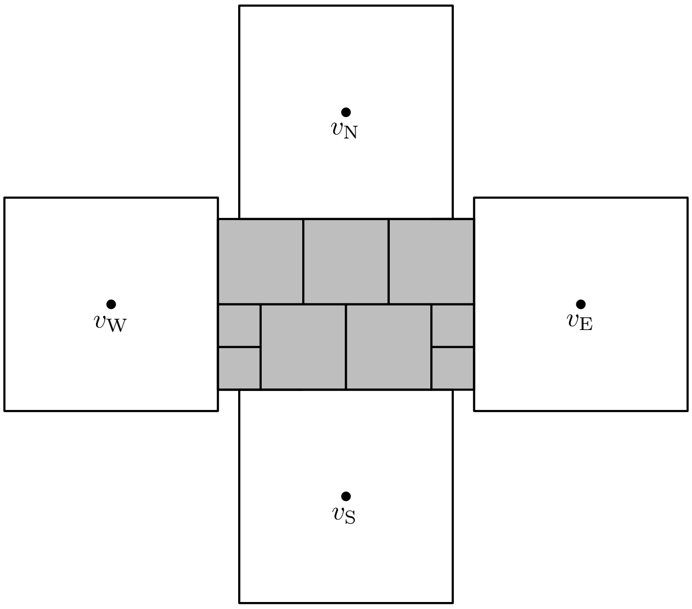


Fig. 4.6: Gap between outer squares. Note that the dual in the middle is a rectangular dual, but not a square dual.

from a graph-theoretic point of view. Let n be the number of variables in a system of difference constraints. The *constraint graph* of the system of difference constraints is a weighted, directed graph and has a vertex u_i for each variable x_i with $i = 1, 2, \ldots, n$. The constraints are modeled as directed edges. For each difference constraint $x_i - x_j \leq b_k$ there is an edge (u_j, u_i). Formally, let there be a system of difference constraint, then the constraint graph is a weighted, directed graph G where

- $V(G) = \{u_0, u_1, u_2, \ldots, u_n\}$ are the vertices of G and
- $E(G) = \{(u_j, u_i) \colon x_i - x_j \leq b_k \text{ is a constraint}\} \cup \{(u_0, u_1), (u_0, u_2), \ldots, (u_0, u_n)\}$ are the edges of G.

Note that G has an additional vertex u_0 from which all other vertices are reachable. If $x_i - x_j \leq b_k$ is a constraint, then the weight of an edge (u_j, u_i) is $\omega(u_j, u_i) = b_k$. The weight of all edge (u_0, u_i) with $i = 1, 2, \ldots, n$ is $\omega(u_0, u_i) = 0$. A solution for the linear program can then be obtained by finding the shortest path weights $\delta(u_0, u_i)$. Precisely the vector $x = (\delta(u_0, u_1), \delta(u_0, u_2), \delta(u_0, u_3), \ldots, \delta(u_0, u_n))$ is a solution for the system of difference constraints and the system does not have a feasible solution when G contains a negative cycle. Let m be the number of constraints, then the shortest path weights can be found by using a slightly modified version of the Bellman–Ford algorithm in $O(mn)$ time. This is an improvement over algorithms that solve general linear programs.

We attempt to express the linear program proposed in Section 4.1 as a system of difference constraints. Consider a slight adjustment of the proposed linear program. Instead of having a square be defined by the coordinates of its bottom-left corner and its width, we now define it as the coordinates of its bottom-left and top-right corner. This definition of squares closely follows the definition of rectangles in Chapter 3 as the cross product of two bounded closed intervals. Let G be a PTP graph and $u \in V(G)$ a vertex of G. For the square $S(u)$, let x_1^u and y_1^u be variables for the coordinates of the bottom-left corner of $S(u)$ and x_2^u and y_2^u be variables for the coordinates of the top-right one. A rectangle is a square if it has equal side lengths, which can be expressed as the following linear constraint:

$$x_2^u - x_1^u = y_2^u - y_1^u \qquad \forall u \in V(G). \tag{4.28}$$

Using an additional variable w_u, we can break up constraint 4.28 into multiple constraints that resemble difference constraints:

$$\begin{aligned} &x_2^u - x_1^u \leq w_u, \\ &y_2^u - y_1^u \leq w_u, \\ &x_1^u - x_2^u \leq -w_u, \\ &y_1^u - y_2^u \leq -w_u \qquad \forall u \in V(G). \end{aligned} \tag{4.29}$$

Sadly, these are not difference constraints, since w_u is a variable and not a constant. However, considering the constraints for vertical and horizontal contact in this new system, it becomes evident that the partial *rectangular* dual extension problem can be modeled as a system of difference constraints. For the vertical contacts:

$$x_2^u - x_1^v = 0 \qquad \forall (u, v) \in E(G), (u, v) \text{ is a red edge in REL}(G), \tag{4.30}$$

$$y_1^{\gamma(u)} - y_2^u < 0, \tag{4.31}$$

$$y_1^u - y_2^{\delta(u)} < 0 \qquad \forall u \in V(G), \tag{4.32}$$

and for the horizontal contacts:

$$y_2^u - y_1^v = 0 \qquad \forall (u, v) \in E(G), (u, v) \text{ is a blue edge in REL}(G), \tag{4.33}$$

$$x_1^{\beta(u)} - x_2^u < 0, \tag{4.34}$$

$$x_1^u - x_2^{\alpha(u)} < 0 \qquad \forall u \in V(G). \tag{4.35}$$

Note that an equality constraint $x_i - x_j = b_k$ can be transformed into two inequalities $x_i - x_j \leq b_k$ and $x_j - x_i \leq -b_k$. A strict inequality constraint $x_i - x_j < b_k$ can be augmented by subtracting a sufficiently small constant ϵ, such that the constraint becomes $x_i - x_j \leq b_k - \epsilon$. Fixed rectangles can be modeled similarly to the LP in Section 4.1. In revision of their work, Chaplick et al. [CKK+21] proposed a system of difference constraints using this idea.

Since it seems like we cannot model the partial square dual extension problem as a system of difference constraint, we conjecture that it cannot be solved in linear time.

5 Triangle Contact Representations

In this chapter we consider contact representations using triangles. A *right-angled axis-aligned triangle* is a right-angled triangle, for which the two short sides are x-axis aligned and y-axis aligned respectively, and the hypotenuse faces towards the upper-left. Another way to imagine a right-angled axis-aligned triangle is as the lower-right half of an axis-aligned rectangle. Similarly to rectangles in Chapter 3, when we refer to a triangle, we always mean a right-angled axis-aligned triangle. Then a *triangle contact system* C is a finite set of triangles, such that no two triangles intersect and C does not have an exceptional touching. For a triangulated graph G, a *triangle contact representation* $C(G)$ is a triangle contact system C, such that the contact graph $G^*(C) = G$. Note that, similarly to rectangles in Chapter 3, the types of contact between two rectangles in a triangle contact system is limited. Where for rectangles there was only side and corner contact, every contact between triangles is a single-point contact. Similar to rectangle representations we can augment a graph to have exactly three vertices on the outer face which will be denoted as v_1, v_2 and v_n throughout this chapter. Analogously to Chapter 3 we call these three vertices *outer vertices* and all other vertices *inner vertices* and assume that the triangles representing the outer vertices are arranged as seen in Figure 5.1 (right). For a triangle contact representation $C(G)$, we further require that each triangle that represents an inner vertex makes contact at its three corners. Note that for a triangulated graph, every inner vertex has at least three neighbors.

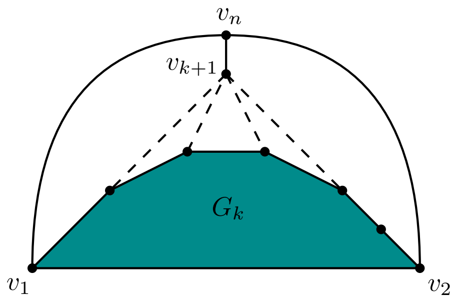
Following the definition of triangle contact representations we can define the partial triangle contact representation extension problem as a special case of the partial contact representation extension problem.

Definition 12 (Partial Triangle Contact Representation Extension Problem). *Given a triangulated graph G, a subset $U \subset V$, a partial triangle contact representation $A(G[U])$ and a combinatorial description of a triangle contact representation $C(G)$, can $A(G[U])$ be extended to $C(G)$.*

While we cannot give a solution to this problem, we define a set of necessary conditions for an extension to exist in this chapter. Throughout this chapter, we denote the triangle T of a vertex $v \in V(G)$ that represents v in a triangle contact representation $C(G)$ as $T(v)$.

5.1 Canonical Order Constraints

For a triangulated graph G, the *canonical order* $\pi(G) = (v_1, v_2, \ldots, v_n)$ is an ordering of the vertices $V(G)$ with v_1, v_2 and v_n being the outer vertices, such that for each k with $2 \leq k \leq n$ the following holds true:



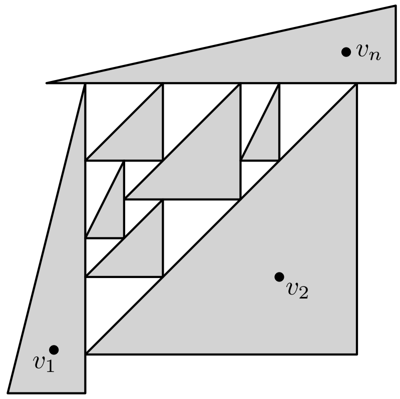


Fig. 5.1: Visualization how a canonical order is defined (left). An example triangle contact representation and the three outer vertices v_1, v_2 and v_n (right).

1. The vertices $\{v_1, v_2, \ldots, v_k\}$ induce an internally triangulated, bi-connected graph G_k. Let $V(G_k)$ be the vertices of G_k.
2. The edge $\{v_1, v_2\}$ is on the outer face of G_k.
3. For $k < n$, the vertex v_{k+1} lies on the outer face of G_k and all neighbors $v_i \in V_k$ of v_{k+1} lie on the boundary of G_k consecutively.

Such an order was first used by De Fraysseix et al. [dFPP90] and later by Kant [Kan96] to create straight line drawings of planar graphs on a grid, see Figure 5.1. De Fraysseix et al. have shown, that for any triangulated graph there exists a canonical order. We call a canonical order $\pi(G) = (v_1, v_2, \ldots, v_n)$ *based on* v_1 and v_2 since those are the first two vertices. By rotating the embedded graph it is also possible to base a canonical order on v_1 and v_n or on v_2 and v_n. The definitions of those canonical orders with different bases follows the definition above analogously.

For a triangle contact system C and and a triangle $T \in C$, let $(x(T), y(T))$ be the lower right corner of T.

Lemma 13. *Let G be a triangulated graph and $C(G)$ a triangle contact representation of G. Further let T_i, T_j be two triangles in $C(G)$ with $y(T_i) \leq y(T_j)$. Then there exists a canonical order $\pi(G) = (v_1, v_2, \ldots, v(T_i), \ldots, v(T_j), \ldots, v_n)$.*

Proof. To prove this Lemma, we make a crucial observation. Since for every triangle contact representation $C(G)$, each inner triangle $T_p \in C(G)$ makes contact at its three corners, we can deduce that T_p has at least two adjacent triangles T_l and T_o with $y(T_l) < y(T_p)$ and $y(T_o) < y(T_p)$. Precisely, T_l and T_o are the triangles that make contact at the two lower corners of T_p.

We build the desired canonical order by reverse induction, removing the triangle T with the largest $y(T)$ and its corresponding vertex $v(T)$ in each step and we show that the resulting sub-graph of G satisfies all required conditions of a canonical order. By definition, $T(v_n)$ is the topmost triangle in $C(G)$. By the maximality of G, the graph

G_{n-1} is a triangulated, bi-connected graph and the neighbors of v_n form the boundary of G_{n-1}.

Let T_k be the triangle that is removed in step k with $2 < k < n$. By our previous observation, removing $v(T_k)$ from G_k will yield the triangulated, bi-connected graph G_{k-1} since all remaining inner triangles have at least two adjacent triangles remaining. Again by the maximality of G_k, the graph G_{k-1} has all neighbors N of $v(T_k)$ with $N \subseteq V(G_{k-1})$ on its outer boundary.

Since v_1 and v_2 are the triangles with the smallest y-coordinates, they will be removed last and are on the outer boundary of each G_k. □

An analogous result, for two triangles $T_i, T_j \in C(G)$, regarding the order of the x-coordinates $x(T_i)$ and $x(T_j)$ for canonical orders based on v_2 and v_n can be shown. For that, imagine removing the triangle T_k with the smallest $x(T_k)$ in each step. The proof then follows analogously to Lemma 13.

Lemma 14. *Let G be a triangulated graph and $C(G)$ a triangle contact representation of G. Further let $T_i, T_j \in C(G)$ be two triangles in $C(G)$ with $x(T_i) \leq x(T_j)$. Then there exists a canonical order $\pi(G) = (v_2, v_n, \ldots, v(T_j), \ldots, v(T_i), \ldots, v_1)$.*

By combining Lemma 13 and Lemma 14, we conclude this section with a constraint that needs to be satisfied for a solution to the partial triangle contact representation problem to exist.

Theorem 15. *Let G be a triangulated graph, $U \subset V(G)$ a subset of vertices and $A(G[U])$ a partial triangle contact representation. Then for an extension of $A(G[U])$ to exist, both a canonical order $\pi(G) = (v_1, v_2, \ldots, v_n)$ that admits the y-coordinate order of the triangles in $A(G[U])$ and a canonical order $\pi(G) = (v_2, v_n, \ldots, v_1)$ that admits the x-coordinate order of the triangles in $A(G[U])$ must exist.*

5.2 Schnyder Wood Constraints

Regular edge labelings and corner edge labelings in Chapter 3 describe the combinatorial aspects of rectangle contact representations. Such a description also exists for triangle contact representations in the form of *Schnyder woods* [Sch89, Sch90], which are also known as *Schnyder Realizers*. Let G be a triangulated graph. A Schnyder wood of G is a partition of the edges $E(G)$ that contain at least one inner vertex, into three subsets S_1, S_2 and S_3 of directed edges, such that for each inner vertex $v \in V(G)$,

1. v has exactly one outgoing edge in S_1, S_2 and S_3 each and
2. the counter clockwise order of edges incident to v are outgoing S_1, incoming S_3, outgoing S_2, incoming S_1, outgoing S_3 and incoming S_2 as seen in Figure 5.2 (right).

Figure 5.2 (left) shows a complete Schnyder wood. The subsets S_1, S_2 and S_3 are trees rooted at the three outer vertices respectively. Throughout the rest of this chapter, we assume that S_1 is rooted at v_n, that S_2 is rooted at v_2 and that S_3 is rooted at v_1. Note that the edges $(v_1, v_2), (v_2, v_n)$ and (v_n, v_1) are not in any of the three subsets. Canonical orders and Schnyder woods are closely related. In fact, it is well-known that both are in bijection, for example, see [dFdMR94].

In triangle contact representations, there are only single-point contacts. By mapping the outgoing edges of each tree S_1, S_2 and S_3 to one of the three corners of a triangle, a Schnyder wood induces a combinatorial description as seen in Figure 5.3. Following up from the proof of Lemma 13 and Schnyder woods, we can define a set of constraints on colored paths in a Schnyder wood.

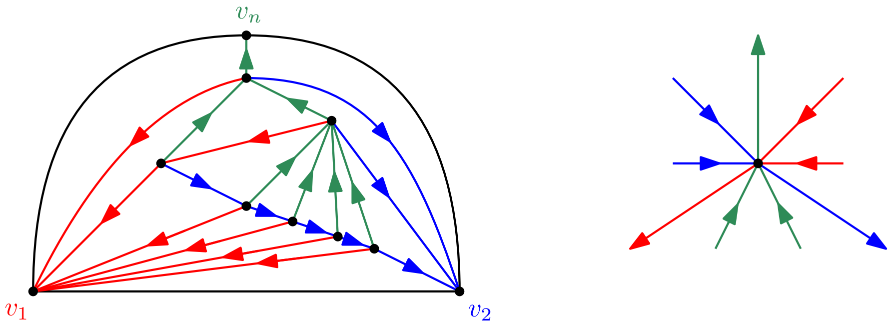


Fig. 5.2: A Schnyder wood (left) and its local coloring rules (right).

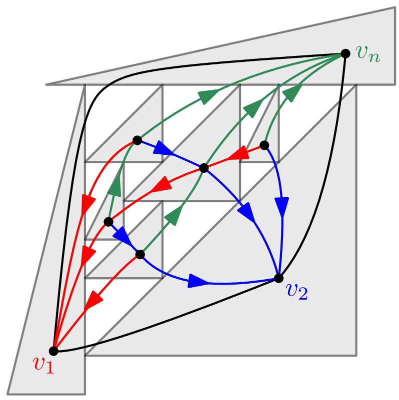


Fig. 5.3: A graph G and a Schnyder wood as a combinatorial description of a triangle contact representation $C(G)$.

Colored Paths. From the observation in the proof of Lemma 13, we have seen that, for a triangulated graph G, a triangle contact representation $C(G)$ and two touching triangles $T_i, T_j \in C(G)$, there are certain limitations for the x-y-coordinates of T_i and T_j depending on the corner that is involved in the contact. Precisely, let T_j be the triangle that touches T_i at the lower right corner of T_i. Then the point $(x(T_j), y(T_j))$ is bound to be in the region defined by the half-spaces $x(T_j) - x(T_i) > 0$ and $y(T_i) - y(T_j) > 0$. These half-spaces are open, that is, the linear inequalities are strict, since non-strict behavior could lead to degenerate triangles, for $x(T_i) = x(T_j)$ and $y(T_i) = y(T_j)$, and exceptional touchings, for either $x(T_i) = x(T_j)$ or $y(T_i) = y(T_j)$.

Since the color of an edge in a Schnyder wood decides which corner of a triangle makes contact, we can extract a set of constraints for the coordinates of a triangle from colored paths in a Schnyder wood. Let G be a triangulated graph, $v_i, v_j \in V(G)$ vertices of G and $\{S_1, S_2, S_3\}$ be a Schnyder wood of G. Further, let there be a path $(v_i, \ldots, v_j) \in S_2$ from v_i to v_j in the Schnyder wood. Then, for any vertex v_k on that path with $v_k \neq v_i$, the values of $x\left(T(v_k)\right)$ and $y\left(T(v_k)\right)$ need to iteratively get larger and smaller respectively. For paths in S_1 and S_3, similar region constraints can be deduced. This observation suggests the following set of rules for partial triangle contact representations.

Corollary 16. *For an instance of the partial triangle contact representation extension problem with a given Schnyder wood $\{S_1, S_2, S_3\}$ and two fixed triangles $T_i, T_j \in A(G[U])$, the coordinates of the lower right corner of T_i and T_j must admit the following rules for an extension to exist:*

- *If there is a path $(v(T_i), \ldots, v(T_j)) \in S_1$, then $x(T_j) - x(T_i) > 0$ and $y(T_j) - y(T_i) > 0$.*
- *If there is a path $(v(T_i), \ldots, v(T_j)) \in S_2$, then $x(T_j) - x(T_i) > 0$ and $y(T_i) - y(T_j) > 0$.*
- *If there is a path $(v(T_i), \ldots, v(T_j)) \in S_3$, then $x(T_i) - x(T_j) > 0$ and $y(T_i) - y(T_j) > 0$.*

For paths in S_1 and S_3 it is obvious that these rules can be further refined by defining the constricting half-planes not by the coordinates of the lower right corner, but by the coordinates of the lower left corner for S_3 and the top corner for S_1. For sake of simplicity, we will refrain from explicitly establishing those more strict rules. We will however take a closer look at paths containing edges of two Schnyder trees.

In the case of a path containing edges of two Schnyder trees, an analogous set of rules can be defined. Precisely, we observe that in the rules stated in Corollary 16, the rules for S_1 and S_2 and the rules for S_2 and S_3 share an inequality each and can be combined. This yields the following set of rules.

Corollary 17. *For an instance of the partial triangle contact representation extension problem with a given Schnyder wood $\{S_1, S_2, S_3\}$ and two fixed triangles $T_i, T_j \in A(G[U])$, the coordinates of the lower right corner of T_i and T_j must admit the following rules for an extension to exist:*

- *If there is a path $(v(T_i), \ldots, v(T_j)) \in S_1 \cup S_2$, then $x(T_j) - x(T_i) > 0$.*
- *If there is a path $(v(T_i), \ldots, v(T_j)) \in S_2 \cup S_3$, then $y(T_i) - y(T_j) > 0$.*

See Figure 5.4 for a visual representation of the rules. If any two fixed triangles violate one of those rules, then there exists no extension. This concludes the set of necessary constraints that need to be satisfied for an extension of a partial triangle contact representation to exist. We are unaware whether the constraints proposed in Section 5.1 and Section 5.2 are sufficient to prove that a triangle contact representation extension exists, and hence, whether they solve the partial triangle contact representation extension problem or not.

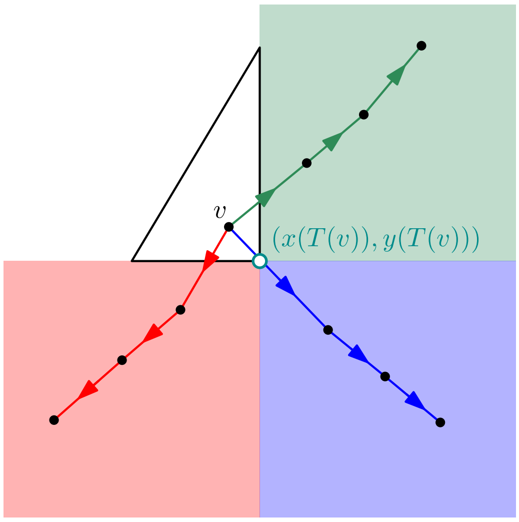


Fig. 5.4: Colored paths leading from a vertex v. The colored regions depict where the triangles representing the vertices can appear.

6 Conclusion

Contact representations of graphs remain an actively studied topic. In this work, we have taken a closer look at different variants of the partial polygon contact representation problem.

We have seen that, for an MTP graph G, the corner edge labeling is closely related to the regular edge labeling of an inner triangulation of G. Based on this, we have shown that, using the linear time algorithm to solve the partial rectangular dual extension problem introduced by Chaplick et al. [CKK+21], the partial rectangle contact representation problem can be solved in linear time as well and that for yes-instances an extension can be computed within the same time bound.

We further proposed a linear program for the partial square dual extension problem, such that a solution to the LP yields an extension, if an extension exists, otherwise there is no solution. We then have seen that the proposed linear program can be slightly modified to form a system of difference constraints, such that solving the modified linear program solves the partial rectangular dual extension problem.

Both the partial rectangle contact representation problem and the partial square dual extension problem remain open when no corner edge labeling or regular edge labeling respectively is specified. For the square dual contact representation recognition problem, given a regular edge labeling, Felsner [Fel13] proposed a system of linear equations and, should there be no solution that satisfies all equations, a method to alter the specified regular edge labeling and try again. While Felsner conjectured that this iterative process terminates independent of the choice of the initial regular edge labeling, they could not prove the termination. It might be possible to adapt Felsner's iteration process to solve the partial rectangle contact representation extension problem and the partial square dual extension problem without given combinatorial descriptions.

The partial polygon contact representation problem remains open as well. For that, K-gon contact structures introduced by Felsner et al. [FSS18] for equiangular contact systems could be a good candidate for a combinatorial description.

For right-angle axis-aligned triangle contact representations we have proposed a set of necessary constraints that need to be satisfied for a triangle contact extension to exist. We do not know whether these constraint are sufficient. However, we conjecture that they are not. The partial triangle contact representation problem remains open.

Bibliography

[ABF$^+$12] Muhammad Jawaherul Alam, Therese Biedl, Stefan Felsner, Michael Kaufmann, and Stephen G. Kobourov. Proportional contact representations of planar graphs. In Marc van Kreveld and Bettina Speckmann, editors, *Proc. Graph Drawing and Network Visualization (GD)*, pages 26–38. Springer, 2012. `doi:10.1007/978-3-642-25878-7_4`.

[ACC$^+$19] Patrizio Angelini, Steven Chaplick, Sabine Cornelsen, Giordano Da Lozzo, and Vincenzo Roselli. Morphing contact representations of graphs. In Gill Barequet and Yusu Wang, editors, *Proc. 35th International Symposium on Computational Geometry (SoCG)*, volume 129 of *LIPIcs*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. `arXiv:1903.07595`.

[BETT98] Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall PTR, 1998.

[BL76] Kellogg S. Booth and George S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using *PQ*-tree algorithms. *Journal of Computer and System Sciences*, 13(3):335–379, 1976. `doi:10.1016/S0022-0000(76)80045-1`.

[BSST40] R. L. Brooks, C. A. B. Smith, A. H. Stone, and W. T. Tutte. The dissection of rectangles into squares. *Duke Mathematical Journal*, 7(1):312–340, 1940. `doi:10.1215/S0012-7094-40-00718-9`.

[CDK$^+$14] Steven Chaplick, Paul Dorbec, Jan Kratochvíl, Mickael Montassier, and Juraj Stacho. Contact representations of planar graphs: Extending a partial representation is hard. In Dieter Kratsch and Ioan Todinca, editors, *Proc. Graph-Theoretic Concepts in Computer Science*, pages 139–151. Springer, 2014. `doi:10.1007/978-3-319-12340-0_12`.

[CFK13] Steven Chaplick, Radoslav Fulek, and Pavel Klavík. Extending partial representations of circle graphs. In Stephen Wismath and Alexander Wolff, editors, *Proc. Graph Drawing and Network Visualization (GD)*, pages 131–142. Springer, 2013. `doi:10.1007/978-3-319-03841-4_12`.

[CGG$^+$18] Steven Chaplick, Grzegorz Guśpiel, Grzegorz Gutowski, Tomasz Krawczyk, and Giuseppe Liotta. The partial visibility representation extension problem. *Algorithmica*, 80(8):2286–2323, 2018. `doi:10.1007/s00453-017-0322-4`.

[CKK+21] Steven Chaplick, Philipp Kindermann, Jonathan Klawitter, Ignaz Rutter, and Alexander Wolff. Extending partial representations of rectangular duals with given contact orientations. In *Proc. Algorithms and Complexity: 12th International Conference*, pages 340–353. Springer International Publishing, 2021. arXiv:2102.02013, doi:10.1007/978-3-030-75242-2_24.

[CKU13] Steven Chaplick, Stephen G. Kobourov, and Torsten Ueckerdt. Equilateral L-contact graphs. In Andreas Brandstädt, Klaus Jansen, and Rüdiger Reischuk, editors, *Proc. Graph-Theoretic Concepts in Computer Science*, pages 139–151. Springer, 2013. doi:10.1007/978-3-642-45043-3_13.

[dFdM07] Hubert de Fraysseix and Patrice Ossona de Mendez. Representations by contact and intersection of segments. *Algorithmica*, 47(4):453–463, 2007. doi:10.1007/s00453-006-0157-x.

[dFdMR94] Hubert de Fraysseix, Patrice Ossona de Mendez, and Pierre Rosenstiehl. On triangle contact graphs. *Combinatorics, Probability and Computing*, 3(2):233–246, 1994. doi:10.1017/CBO9780511662034.019.

[dFPP90] Hubert de Fraysseix, János Pach, and Richard Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990. doi:10.1007/BF02122694.

[Fel09] Stefan Felsner. Triangle contact representations. In *Midsummer Combinatorial Workshop, Praha*, 2009. URL: http://page.math.tu-berlin.de/~felsner/Paper/prag-report.pdf.

[Fel13] Stefan Felsner. Rectangle and square representations of planar graphs. In *Thirty Essays on Geometric Graph Theory*, pages 213–248. Springer, 2013. doi:10.1007/978-1-4614-0110-0_12.

[FF11] Stefan Felsner and Mathew C Francis. Contact representations of planar graphs with cubes. In *Proc. 27th Annual Symposium on Computational Geometry*, pages 315–320, 2011. doi:10.1145/1998196.1998250.

[FSS18] Stefan Felsner, Hendrik Schrezenmaier, and Raphael Steiner. Equiangular polygon contact representations. In Andreas Brandstädt, Ekkehard Köhler, and Klaus Meer, editors, *Proc. Graph-Theoretic Concepts in Computer Science*, pages 203–215. Springer, 2018. doi:10.1007/978-3-030-00256-5_17.

[Fus09] Éric Fusy. Transversal structures on triangulations: A combinatorial study and straight-line drawings. *Discrete Mathematics*, 309(7):1870–1894, 2009. doi:10.1016/j.disc.2007.12.093.

[GLP11] Daniel Gonçalves, Benjamin Lévêque, and Alexandre Pinlou. Triangle contact representations and duality. In Ulrik Brandes and Sabine Cornelsen, editors, *Proc. Graph Drawing and Network Visualization (GD)*, pages 262–273. Springer, 2011. doi:10.1007/978-3-642-18469-7_24.

[Kan96] G. Kant. Drawing planar graphs using the canonical ordering. *Algorithmica*, 16(1):4–32, 1996. doi:10.1007/BF02086606.

[KH97] Goos Kant and Xin He. Regular edge labeling of 4-connected plane graphs and its applications in graph drawing problems. *Theoretical Computer Science*, 172(1-2):175–193, 1997. doi:10.1016/S0304-3975(95)00257-X.

[KK85] Krzysztof Koźmiński and Edwin Kinnen. Rectangular duals of planar graphs. *Networks*, 15(2):145–157, 1985.

[KKO+17] Pavel Klavík, Jan Kratochvíl, Yota Otachi, Toshiki Saitoh, and Tomáš Vyskočil. Extending partial representations of interval graphs. *Algorithmica*, 78(3):945–967, 2017. doi:10.1007/s00453-016-0186-z.

[KNU15] Jonathan Klawitter, Martin Nöllenburg, and Torsten Ueckerdt. Combinatorial properties of triangle-free rectangle arrangements and the squarability problem. In *Proc. Graph Drawing and Network Visualization*, pages 231–244. Springer, 2015. doi:10.1007/978-3-319-27261-0_20.

[Koe36] Paul Koebe. Kontaktprobleme der konformen Abbildung. *Berichte über die Verhandlungen der Sächsischen Akademie der Wissenschaften zu Leipzig. Math.-Phys. Klasse*, 88:141–164, 1936.

[KS15] Vincent Kusters and Bettina Speckmann. Towards characterizing graphs with a sliceable rectangular dual. In *Proc. Graph Drawing and Network Visualization*, pages 460–471. Springer, 2015. doi:10.1007/978-3-319-27261-0_38.

[KUV13] Stephen Kobourov, Torsten Ueckerdt, and Kevin Verbeek. Combinatorial and geometric properties of planar laman graphs. In *Proc. ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1668–1678. SIAM, 2013. doi:10.1137/1.9781611973105.120.

[KW03] Michael Kaufmann and Dorothea Wagner. *Drawing Graphs: Methods and Models*. Springer, 2003. doi:10.1007/3-540-44969-8.

[KW17] Tomasz Krawczyk and Bartosz Walczak. Extending partial representations of trapezoid graphs. In Hans L. Bodlaender and Gerhard J. Woeginger, editors, *Proc. Graph-Theoretic Concepts in Computer Science*, pages 358–371. Springer, 2017. doi:10.1007/978-3-319-68705-6_27.

[LL84] Sany M Leinwand and Yen-Tai Lai. An algorithm for building rectangular floor-plans. In *21st Design Automation Conference Proceedings*, pages 663–664. IEEE, 1984. doi:10.1109/DAC.1984.1585874.

[NR04] Takao Nishizeki and Md Saidur Rahman. *Planar Graph Drawing*, volume 12. World Scientific Publishing, 2004. doi:10.1142/5648.

[Sch89] Walter Schnyder. Planar graphs and poset dimension. *Order*, 5(4):323–343, 1989. doi:10.1007/BF00353652.

[Sch90] Walter Schnyder. Embedding planar graphs on the grid. In *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 138–148, 1990.

[Sch93] Oded Schramm. Square tilings with prescribed combinatorics. *Israel Journal of Mathematics*, 84(1):97–118, 1993. doi:10.1007/BF02761693.

[Sch07] Oded Schramm. *Combinatorically Prescribed Packings and Applications to Conformal and Quasiconformal Maps*. PhD thesis, Princeton University, 2007. Modified version of PhD thesis from 1990. URL: https://arxiv.org/abs/0709.0710.

[Ste73] Philip Steadman. Graph theoretic representation of architectural arrangement. *Architectural Research and Teaching*, 2(1):161–172, 1973.

[Tho86] Carsten Thomassen. Interval representations of planar graphs. *Journal of Combinatorial Theory, Series B*, 40(1):9–20, 1986. doi:10.1016/0095-8956(86)90061-4.

[YS95] Gary K. H. Yeap and Majid Sarrafzadeh. Sliceable floorplanning by graph dualization. *SIAM Journal on Discrete Mathematics*, 8(2):258–280, 1995. doi:10.1137/S0895480191266700.

Erklärung

Hiermit versichere ich die vorliegende Abschlussarbeit selbstständig verfasst zu haben, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben, und die Arbeit bisher oder gleichzeitig keiner anderen Prüfungsbehörde unter Erlangung eines akademischen Grades vorgelegt zu haben.

Würzburg, den December 14, 2021

. . . .
Peter Markfelder