Transactions on Combinatorics
ISSN (print): 2251-8657, ISSN (on-line): 2251-8665
Vol. 12 No. 3 (2023), pp. 143-163.

https://toc.ui.ac.ir

www.ui.ac.ir

TRANSFORMATIONS AMONG RECTANGULAR PARTITIONS

VINOD KUMAR* AND KRISHNENDRA SHEKHAWAT

Abstract. We first prove that there always exists a maximal rectangularly dualizable graph for a given rectangularly dualizable graph and present an algorithm for its construction. Further, we show that a maximal rectangularly dualizable graph can always be transformed to an edge-irreducible rectangularly dualizable graph and present an algorithm that transforms a maximal rectangularly dualizable graph to an edge-irreducible rectangularly dualizable graph.

1. Introduction

Due to advances in VLSI technology, interconnection optimization has become a major concern [18]. Sometimes, it is required to customize interconnections of an existing VLSI circuit in two ways: to customize interconnection among modules and interconnection of modules to the outside units. To deal with the first case, we need to increase/decrease the adjacencies among the modules as much as possible and the second requires to increase the length of exterior of the layout which depends on the number of modules adjacent to the exterior. In architectural terms, a legacy rectangular floorplan can be reconstructed to suit modern lifestyles by changing the adjacency relations of its rooms. In this paper, we present graph theoretic characterizations for customizing interconnections in floorplans.

A *generic rectangular partition* is a partition of a rectangle into a finite set of rectangles provided that no four rectangles share the same point. Now and onward, a rectangular partition, we mean a generic rectangular partition. A graph $\mathcal{H}$ is called the *dual* of a plane graph $\mathcal{G}$ if there is one-to-one correspondence between the vertices of $\mathcal{G}$ and the regions of $\mathcal{H}$, and two vertices of $\mathcal{G}$ are adjacent if and only if the corresponding regions of $\mathcal{H}$ share a common boundary segment. A plane graph

Communicated by Manouchehr Zaker.

MSC(2010): Primary: 68R10; Secondary: 68U05.

Keywords: planar graph, rectangular dual, rectangularly dualizable graph, Rectangular Partitions.

Manuscript Type: Research Paper.

Received: 09 Aprill 2022, Accepted: 03 August 2022.

*Corresponding author.

http://dx.doi.org/10.22108/TOC.2022.133242.1984 .

is a *rectangularly dualizable graph* (RDG) if its dual can be embedded as a *rectangular partition*. A *rectangular dual* $\mathcal{R}$ of a plane graph $\mathcal{G}$ is a partition of a rectangle into a finite set of rectangles such that (i) no four rectangles of $\mathcal{R}$ share a common corner point, (ii) rectangles in $\mathcal{R}$ are mapped to vertices of $\mathcal{G}$, and (iii) two rectangles in $\mathcal{R}$ share a common boundary segment if and only if the corresponding vertices are adjacent.

1.1. **Related work.** In 1985, Kozminski and Kinnen [7] characterized rectangular partitions in terms of graphs. In 1987, Bhasker and Sahni [3, 4] provided a linear-time algorithm for checking the existence of a rectangular partition for a plane triangulated graph and in 1988, they provided a linear-time algorithm that constructs a rectangular partition for a given plane triangulated graph. In the same year, Rinsma [11] showed that it is not always possible to construct a rectangular partition with an arbitrary assignment of areas to its rectangles. In 1990, Lai and Leinwand [10] showed that the construction of a rectangular partition is equivalent to a matching problem in a bipartite graph derived from a given planar graph. In 1993, He [20] presented a new linear time algorithm for finding a rectangular partition for a given plane triangulated graph, which is conceptually simpler than the previously known algorithms. In 2000, Recuero et al. [2] gave a heuristic method that maps a graph into rectangles.

A rectangular partition is area-universal if no matter what assignment of areas to its rectangles be given, a combinatorial equivalent rectangular partition can be constructed. In 2012, Eppstein et al. [6] characterized area universal rectangular partitions for the given RDG. In fact, they showed that a rectangular partition is area universal if and only if it is one-sided. In 2014, Felsner [8] proved that every rectangular partition with n internal segments can be embedded on every set of n points in generic position.

Transformations among rectangular partitions were studied [9, 12, 14] where a topologically distinct rectangular partition was constructed from a given rectangular partition such that the resulting rectangular partition admits the same dual graph. Intuitively speaking, these transformations change vertical (horizontal) shared line segment of two rectangles of a rectangular partition to the horizontal(vertical) shared line segment of the two rectangles in the resulting rectangular partition. Wang *et al.* [21] customized rectangular partitions by adding or removing rectangles to it. They presented a transformation algorithm by which rectangles can be added or deleted from a given rectangular partition to generate a feasible rectangular partition. But they did not provide any necessary and sufficient condition to do so. The shortfall is that while removing or adding a rectangle to a rectangular partition, some existing adjacencies of rectangles either lose or new adjacencies are introduced among rectangles. But we presents algorithms that transform a rectangular partition to a new rectangular partition by introducing new adjacencies among rectangles of a rectangular partition can be introduced or by removing adjacencies of rectangles. In fact, for any two given adjacent rectangles of a rectangular partition, it is interesting to identify whether they can be made non-adjacent in such a way that adjacencies of remaining rectangles of the rectangular partition remains sustained and the resultant plan is a rectangular partition. Conversely, can any two given non-adjacent rectangles of a

http://dx.doi.org/10.22108/TOC.2022.133242.1984

rectangular partition be made adjacent in such a way that adjacencies of the remaining rectangles of the rectangular partition remains preserved and the resulting partition is a rectangular partition. The answer to these questions are given in this paper.

Enumerating of all rectangular partitions, each composed of n-rectangles, has been remained a major issue in combinatorics [15, 1, 16, 5, 13, 17, 19]. But these enumerating methods are not preferable because of producing large solution sets and therefore, it is computationally hard to pick a rectangular partition suiting adjacency requirements of rectangles from such a large solution set.

1.2. **Our Results.** For a given RDG, we aim to construct a new RDG as follows:

i. By introducing new adjacency relations while preserving all existing adjacency relations until no more adjacency relation can be added.
ii. By deleting adjacency relations while preserving all the other adjacency relations until no more adjacency relation can be removed.

We first prove that these transformations are feasible and then present quadratic time algorithms for them. The class of maximal RDGs (MRDGs) can play an important role in floorplanning because they are rich in adjacency relations (an MRDG is an RDG having maximal adjacencies among its vertices). We show that there always exists an MRDG for a given RDG. Then we present a polynomial time algorithm that constructs an MRDG for a given RDG by adding new edges among its non-adjacent vertices. We further show that a given MRDG is always edge-reducible and can be reduced to an RDG and present an algorithm that deletes the edges of a given RDG until it is an RDG.

A brief description of the rest of the paper is as follows. In this article, we first survey the existing facts about RDGs in Section 2. In Section 3, we introduce MRDGs and edge-reducible RDGs. Then we show that an MRDG has $2n-2$ or $3n-7$ edges. MRDGs with $2n-2$ edges are wheel graphs whereas MRDGs with $3n-7$ edges are obtained from the class of maximal plane graphs with the property that they do not have any separating triangles in their interiors by deleting one of their exterior edges. In Section 4, we prove that it is always possible to construct an MRDG for a given RDG and present an algorithm for its construction from the given RDG. In Section 5, we show that it is always possible to transform an edge-reducible RDG to another RDG and present an algorithm for its reduction to a minimal one. A conclusive summary and future direction is given in Section 6.

A list of notations used in this paper can be seen in Table 1.

Symbol	Description
RDG	rectangularly dualizable graph
MRDG	maximal rectangularly dualizable graph
BNG	block neighborhood graph
v_i	i^{th} vertex of a graph
$d(v_i)$	degree of v_i
(v_i, v_j)	an edge incident to vertices v_i and v_j
3-cycle	a cycle of length 3
$v_iv_jv_k$	a cycle passing through vertices v_i, v_j and v_k of an RDG
R_i	a rectangle corresponding v_i
E_i	edge set of graph $\mathcal{G}_i$
$\|E\|$	cardinality of a set E
CIP(s)	corner implying path(s)

Table 1. List of Notations

2. Preliminaries

A planar graph is a graph that can be embedded in the plane without crossings. A plane graph is a planar graph with a fixed planar embedding. It partitions the plane into connected regions called faces; the unbounded region is the exterior face (the outermost face) and all other faces are interior faces. The vertices lying on the exterior face are exterior vertices and all other vertices are interior vertices. A planar (plane) graph is maximal if no new edge can be added to it without disturbing planarity. Thus each face of a maximal plane graph is a triangle. If a connected graph has a cut-vertex, then it is called a separable graph, otherwise it is called a nonseparable graph.

A *generic rectangular partition* is a partition of a rectangle into a finite set of rectangles provided that no four rectangles share the same point. Now and onward, a rectangular partition, we mean a generic rectangular partition. A graph $\mathcal{H}$ is *dual* of a plane graph $\mathcal{G}$ if there is one-to-one correspondence between the vertices of $\mathcal{G}$ and the regions of $\mathcal{H}$, and two vertices of $\mathcal{G}$ are adjacent if and only if the corresponding regions of $\mathcal{H}$ share a common boundary segment. A plane graph is a *rectangularly dualizable graph* (RDG) if its dual can be embedded as a *rectangular partition*. A *rectangular dual* $\mathcal{R}$ of a plane graph $\mathcal{G}$ is a partition of a rectangle into a finite set of rectangles such that (i) no four rectangles of $\mathcal{R}$ share a common corner point, (ii) rectangles in $\mathcal{R}$ are mapped to vertices of $\mathcal{G}$, and (iii) two rectangles in $\mathcal{R}$ share a common boundary segment if and only if the corresponding vertices are adjacent.

Before presenting well known results contained in Theorems 2.1 and 2.2, we first some definitions which as follows.

Definition 2.1. [4] A separating triangle is a cycle of length 3 in a plane graph that encloses atleast one vertex inside as well as outside. It is also known as a complex triangle.

http://dx.doi.org/10.22108/TOC.2022.133242.1984

The graph shown in Fig. 1b that has a cycle $\mathcal{C}$ of length 3 passing through the vertices v_1, v_2 and v_9, enclosing the vertices v_4 and v_5, and having many vertices outside $\mathcal{C}$. Therefore $\mathcal{C}$ is a separating triangle.

Definition 2.2. [4] The block neighborhood graph (BNG) of a planar graph $\mathcal{G}$ is a graph in which each component of $\mathcal{G}$ is represented by a vertex and there is an edge between two vertices of the BNG if and only if the two corresponding components have a vertex in common.

Definition 2.3. [12] A shortcut in a nonseparable plane graph $\mathcal{G}$ is an edge incident to two vertices on the outermost cycle of $\mathcal{G}$ and it is not a part of this cycle. A corner implying path (CIP) in $\mathcal{G}$ is a $v_1 - v_k$ path on the outermost cycle of $\mathcal{G}$ such that it does not contain any vertex of a shortcut other than v_1 and v_k and the shortcut (v_1, v_k) is called a critical shortcut. A critical CIP in a nonseparable component H_k of a separable plane graph $\mathcal{G}_1$ is a CIP of H_k that does not contain cut-vertices of $\mathcal{G}_1$ in its interior.

For a better understanding to Definition 2.3, consider the graph shown in Fig. 1a.

- Edges (v_1, v_3), (v_6, v_8) and (v_4, v_9) are shortcuts,
- $v_1v_2v_3$ and $v_6v_7v_8$ are CIPs,
- $v_9v_1v_2v_3v_4$ is not a CIP because it contains the endpoints of the shortcut (v_1, v_3) and hence (v_9, v_4) is not a critical shortcut (CIPs may have the same endpoints, but they are edge disjoint).

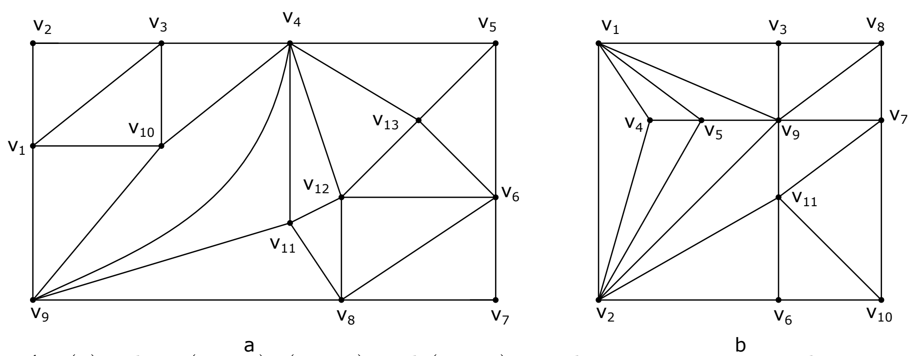

FIGURE 1. (a) Edges (v_1, v_3), (v_6, v_8) and (v_4, v_9) are shortcuts. $v_1v_2v_3$ and $v_6v_7v_8$ are CIPs and (b) $\triangle v_1v_2v_9$ is a separating triangle.

Theorem 2.1. [7, Theorem 3] A nonseparable plane graph $\mathcal{G}$ with triangular interior faces (regions) is an RDG if and only if it has at most 4 CIPs and has no separating triangle.

The graph shown in Fig. 1a is an RDG while the graph in Fig. 1b is not an RDG because of the presence of a separating triangle.

Theorem 2.2. [7, Theorem 5] A separable connected plane graph $\mathcal{G}$ with triangular interior faces (regions) is an RDG if and only if

i. $\mathcal{G}$ has no separating triangle,

ii. BNG of $\mathcal{G}$ is a path,

iii. each maximal block[1] corresponding to the endpoints of the BNG contains at most 2 critical CIPs,

iv. no other maximal block contains a critical CIP.

3. Introducing MRDGs and edge-reducible RDGs

Definition 3.1. An RDG $\mathcal{G} = (V, E)$ is called maximal RDG (MRDG) if there does not exist an RDG $\mathcal{G}' = (V, E')$ with $E' \supset E$.

Definition 3.2. An RDG $\mathcal{G} = (V, E)$ is said to be edge-reducible if there exists an RDG $\mathcal{G}' = (V, E')$ such that $E \supset E'$. If an RDG is not edge-reducible, it is said to be an edge-irreducible RDG.

For a better understanding of Definitions 3.1 and 3.2, refer to Fig. 2. In Fig. 2a, the given RDG is an MRDG since adding a new edge to it, its exterior becomes triangular and hence one of the exterior modules in its floorplan will be non-rectangular. The corresponding rectangular partition of the MRDG is shown in 2b where R_i corresponds to v_i. In Fig. 2c, the given graph $\mathcal{G}$ is an edge-irreducible nonseparable RDG since after removing an edge from it, it no longer remains an RDG. In fact, on removing an exterior edge from $\mathcal{G}$, it transforms to a separable connected graph where one of its blocks contains three critical CIPs. It is noted that we can not remove an interior edge from $\mathcal{G}$ since the resultant graph must have triangular interior regions. This is because we only consider plane graphs with triangular interior regions.

[1]A maximal block of a graph $\mathcal{G}$ is a maximal nonseparable subgraph of $\mathcal{G}$.

http://dx.doi.org/10.22108/TOC.2022.133242.1984

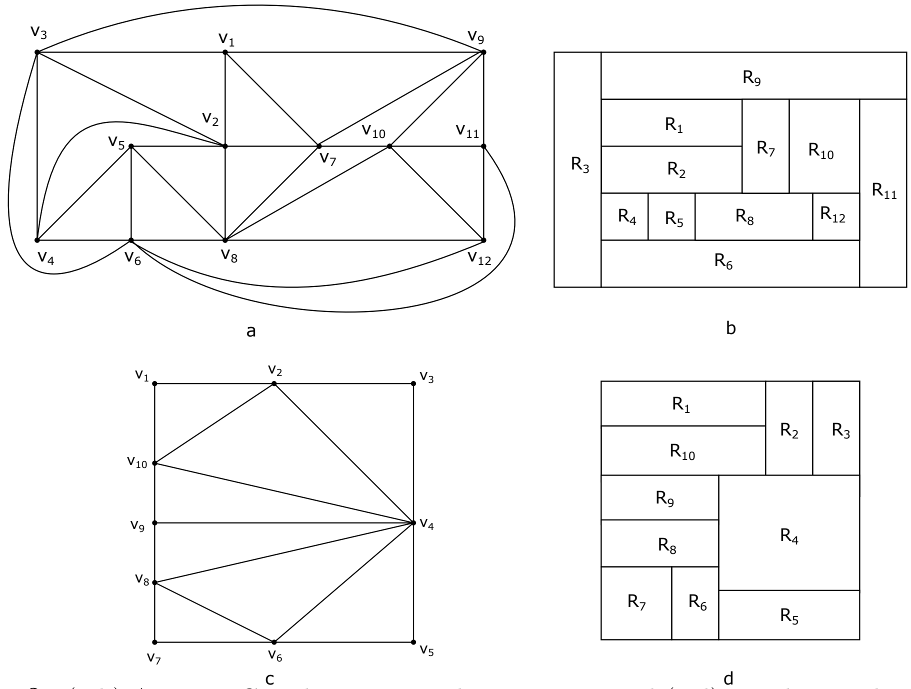

FIGURE 2. (a-b) An MRDG and its rectangular partition, and (c-d) an edge-irreducible non-separable RDG and its rectangular partition.

Theorem 3.1. The number of edges in an MRDG is $2n-2$ or $3n-7$ where n denotes the number of vertices in the MRDG.

Proof. Let $\mathcal{M}$ be an MRDG with n vertices. We have two cases:

(1) When some vertex of the MRDG has degree $n-1$,
(2) When no vertex of the MRDG has degree $n-1$.

Case 1. Let v_i be a vertex of the MRDG $\mathcal{M}$ such that $d(v_i) = n-1$. Then it is a wheel graph W_n[2]. W_n is independent of separating triangles and CIPs. By Theorem 2.1, it is an RDG. Note that adding a new edge to W_n creates a separating triangle passing through its central vertex and two of its exterior vertices. This implies that it is a maximal RDG. Now by degree sum formula, we have $3(n-1)+(n-1) = 2$(the number of edges in W_n) and hence W_n has $2n-2$ edges.

Case 2. Now suppose that $d(v_i) < n-1,\ \forall\ v_i \ \in \mathcal{M}$. Consider a maximal plane graph $\mathcal{G}$ with n vertices such that there does not exist any separating triangle in its interior. We claim that $\mathcal{M} = \mathcal{G}-(v_i, v_j)$ for some exterior edge (v_i, v_j) of $\mathcal{G}$, i.e., we need to show that $\mathcal{G}-(v_i, v_j)$ is an MRDG with $3n-7$ edges.

By our assumption on $\mathcal{G}$, it is evident that $\mathcal{G}-(v_i, v_j)$ has no separating triangle. Suppose that $\mathcal{G}-(v_i, v_j)$ has a CIP. Then there is a shortcut (v_s, v_t) in $\mathcal{G}-(v_i, v_j)$. This implies that $\mathcal{G}$ has a separating triangle $v_s v_t v_e$ in its interior where v_e is an exterior vertex. This is a contradiction to our assumption that $\mathcal{G}$ has no separating triangle in its interior. Thus we see that $\mathcal{G}-(v_i, v_j)$ has no CIP. Thus By Theorem 2.1, it is an RDG.

[2]A wheel graph W_n is a graph in which a single vertex is adjacent to $n-1$ vertices lying on a cycle.

The number of edges in a maximal plane graph is $3n-6$. This implies that $\mathcal{G}-(v_i,v_j)$ has $3n-7$ edges. Note that adding a new edge to $\mathcal{G}-(v_i,v_j)$, creates a separating triangle in $\mathcal{G}-(v_i,v_j)$ and hence it is an MRDG.

Thus we see that $\mathcal{G}-(v_i,v_j)$ is an RDG with $3n-7$ edges and hence $\mathcal{M}$ is an MRDG with $3n-7$ edges. □

Corollary 3.1. The number of vertices on the outermost cycle of an MRDG with n vertices is 4 or $n-1$.

Proof. Let $\mathcal{M}$ be an MRDG with n vertices. If $\mathcal{M}$ is a wheel graph W_n, then it has $n-1$ vertices on its exterior. Otherwise it can obtained from a maximal plane graph that has no separating triangle in its interior by deleting one of its exterior edges. But a maximal plane graph has 3 vertices on its exterior. Then it is evident that $\mathcal{M}$ has 4 vertices on its exterior. □

4. MRDG Construction

In this section, we first prove that it is always possible to construct an MRDG $\mathcal{M}=(V,E)$ for a given RDG $\mathcal{G}=(V,E_1)$ such that $E_1 \subset E$. Then we present an algorithm for its construction corresponding to $\mathcal{G}$.

In order to prove the main result, we first need to prove some lemmas. Denote $|N(v_i)\cap N(v_j)|$ by s for any two adjacent vertices v_i and v_j of an RDG $\mathcal{G}$.

Lemma 4.1. If $s=0$, then (v_i,v_j) is an exterior edge of $\mathcal{G}$.

Proof. For $s=0$, (v_i,v_j) is a cut-edge (bridge) and hence is an exterior edge. □

Lemma 4.2. For all adjacent vertices v_i and v_j, we have $s \leq 2$.

Proof. To the contrary, suppose that $s=3$. Consider a plane embedding $\mathcal{G}_e$ of $\mathcal{G}$. Since, $s=3$, $N(v_i)$ and $N(v_j)$ must have 3 common vertices v_k, v_l and v_m which results in 3 cycles $v_iv_jv_k$, $v_iv_jv_l$ and $v_iv_jv_m$ in $\mathcal{G}_e$. Now (v_i,v_j) is a common edge in these 3 cycles. This implies that atleast 2 of 3 cycles would lie on the same side of (v_i,v_j) in $\mathcal{G}_e$. This means that one of the cycles encloses some vertex v_t of the other cycle and hence is not a face in $\mathcal{G}_e$. Therefore its removal results $\mathcal{G}_e$ in a disconnected graph and hence it is a separating triangle in $\mathcal{G}$, which is a contradiction to Theorems 2.1 and 2.2 since $\mathcal{G}$ is an RDG. Similarly, if $s \geq 3$, we arrive at the contradiction. □

Lemma 4.3. (v_i,v_j) is an interior edge of $\mathcal{G}$ if and only if $s=2$.

Proof. First suppose that $s=2$. We need to show that (v_i,v_j) is an interior edge in $\mathcal{G}$. To the contrary, suppose that (v_i,v_j) is an exterior edge of $\mathcal{G}$. Let $N(v_i)\cap N(v_j)=\{v_k,v_l\}$. Since (v_i,v_j) is an exterior edge, there exist two triangles $v_iv_jv_k$, $v_iv_jv_l$ in the plane embedding of $\mathcal{G}$ such that both lie on the same side of (v_i,v_j). This implies that one of them contains the other and hence is not a region (face) and is a separating triangle. This is a contradiction to Theorems 2.1 and 2.2 since $\mathcal{G}$ is an RDG.

Conversely, suppose that (v_i, v_j) is an interior edge in $\mathcal{G}$. Since $\mathcal{G}$ is an RDG, each of its interior regions is triangular. This implies that there exist two triangles $v_iv_jv_k$ and $v_iv_jv_l$ in the plane embedding of $\mathcal{G}$. Hence $N(v_i)$ and $N(v_j)$ have atleast two vertices in common, i.e., $s \geq 2$. By Lemma 4.2, we have $s \leq 2$. Hence $s = 2$. □

Corollary 4.1. If $s = 1$, then (v_i, v_j) is an exterior edge of $\mathcal{G}$.

Proof. It is the direct consequence of Lemmas 4.1 and 4.3. □

Lemma 4.4. It is always possible to construct a nonseparable (nonseparable) RDG from a separable connected RDG by adding edges to it.

Proof. Let $\mathcal{G}_1 = (V, E_1)$ be a separable connected RDG such that it has atleast one bridge (cut-edge). Suppose that $L_1 = \{(v_i, v_j) \in E_1 \mid |N(v_i) \cap N(v_j)| = 0\}$ and $L_2 = \{(v_a, v_b) \in E_1 \mid |N(v_a) \cap N(v_b)| = 1\}$. Consider two adjacent edges, (v_i, v_j) from L_1 and (v_j, v_k) from L_2 such that $|N(v_i) \cap N(v_k)| = 1$. Such selection is always possible since both edges belongs to different blocks and $N(v_i) \cap N(v_k) = \{v_j\}$.

Construct a graph $\mathcal{G}_2 = (V, E_2)$ where $E_2 = E_1 \cup \{(v_i, v_k)\}$. To prove $\mathcal{G}_2$ is an RDG, we prove the following:

- there does not exist a separating triangle passing through (v_i, v_k) in $\mathcal{G}_2$

  There would be a separating triangle passing through (v_i, v_k) in $\mathcal{G}_2$ if $|N(v_i) \cap N(v_k)| = 2$ in $\mathcal{G}_2$. In this case, $v_j, v_r \in N(v_i) \cap N(v_k)$ such that v_r lies inside the triangle passing through (v_i, v_k). But $(v_i, v_j) \in L_1$ and $(v_j, v_k) \in L_2$. Therefore by Lemma 4.1 and Corollary 4.1, both (v_i, v_j) and (v_j, v_k) are the exterior edges and v_i and v_k belongs to different blocks in $\mathcal{G}_1$. Hence in $\mathcal{G}_2$, $|N(v_i) \cap N(v_k)| = 1$ is the only possibility.
- the number of critical CIPs in $\mathcal{G}_2$ can not exceed the number of critical CIPs in $\mathcal{G}_1$

  In $\mathcal{G}_2$, a critical CIP can only pass through (v_i, v_k), which already passes through (v_i, v_j) and (v_j, v_k) in $\mathcal{G}_1$. But v_j is a cut vertex which is a contradiction to the fact that a critical CIP never passes through a cut vertex.

Since $\mathcal{G}_1$ is an RDG, each of its region is triangular. The new edge (v_i, v_k) is added with the property that $|N(v_i) \cap N(v_k)| = 1$. By Corollary 4.1, (v_i, v_k) is exterior edge in $\mathcal{G}_2$. Therefore, the new region $v_iv_jv_k$ is triangular in $\mathcal{G}_2$. By Theorem 2.2, $\mathcal{G}_2$ is an RDG.

After adding (v_i, v_k) to $\mathcal{G}_1$, the edge (v_i, v_j) from L_1 belongs to L_2 since $|N(v_i) \cap N(v_j)| = 1$ $(|N(v_i) \cap N(v_j)| = \{v_k\})$. Therefore a recursive process shows that at the iteration until L_1 is empty, $\mathcal{G}_{k+1} = (V, E_{k+1})$ becomes a separable connected RDG with cut-vertices (vertex), but no cut edge where $E_{k+1} = E_k \cup (v_a, v_c)$ such that (v_a, v_b) is from L_1 and (v_b, v_c) is from L_2 with the property $|N(v_a) \cap N(v_c)| = 1$. In this way, we can construct a separable connected RDG having cut-vertices of the given separable connected RDG only.

It now remains to show that it is always possible to construct a nonseparable (nonseparable) RDG of the given separable connected RDG $\mathcal{G}_1 = (V, E_1)$ having cut-vertices but no cut-edges. Let v_t be its cut-vertex. Since it has no cut-edge, $d(v_t) \geq 4$. A plane embedding of $\mathcal{G}_1$ with exterior cycles C_1 and C_2 sharing a cut vertex v_t is shown in Fig. 3a. It is evident from this embedding that there is no

http://dx.doi.org/10.22108/TOC.2022.133242.1984

separating triangle passing though the new added edges (red edges) in the resultant graph shown in Fig. 3b. Since v_t is a cut-vertex, none of the vertices v_1, v_2, v_3 and v_4 in Fig. 3, which are adjacent to v_t, can be the endpoints of a shortcut in $\mathcal{G}_1$. This implies that the number of CIPs in the resultant graph (shown in Fig. 3b) can not exceed than the number of critical CIPs in $\mathcal{G}_1$. This proves the required result. □

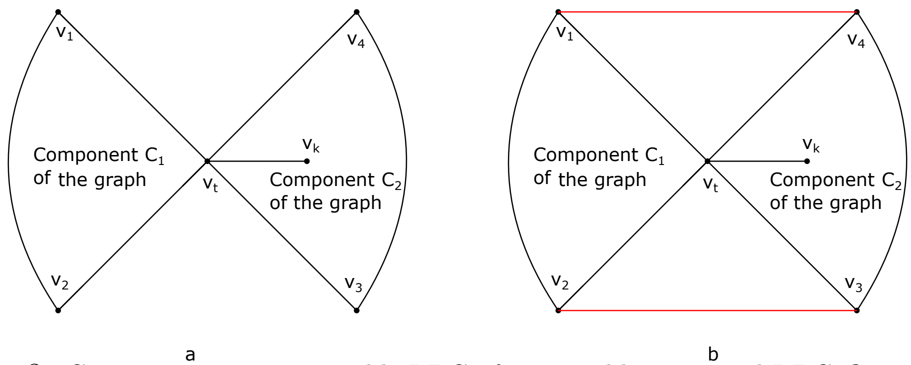

Figure 3. Constructing a nonseparable RDG of a separable connected RDG $\mathcal{G}_1$ with a cut-vertex v_t shared by its two components C_1 and C_2.

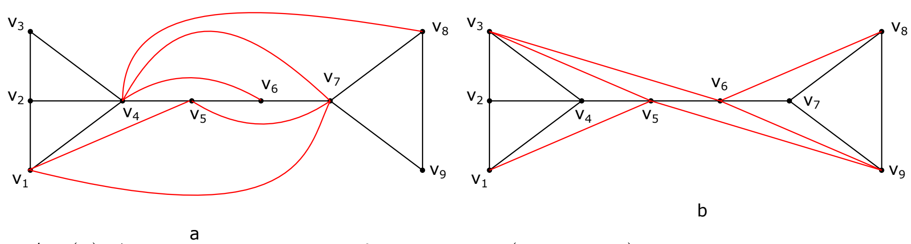

Figure 4. (a) A random addition of new edges (red edges) to a separable connected RDG destroy the RDG property because of generation of a separating triangle $v_4v_5v_7$ in the resultant graph (b) while adding new edges (red edges) by following Lemma 4.4 do not destroy RDG property (no separating triangle and CIP in the resultant graph).

Remark 4.1. It is not straight forward to add edges to a separable connected RDG maintaining RDG property. Randomly adding new edges to an RDG can destroy the RDG property, i.e., can produce either a CIP or a separating triangle in the resultant graph. In the lie of this, we have shown the procedure of adding edges by Lemma 4.4. For instance, consider a separable connected RDG shown in Fig. 4a. It is transformed to a nonseparable graph by adding new edges (red edges) randomly. As a result, the nonseparable graph thus obtained is not an RDG. In fact, it contains a separating triangle $v_4v_5v_7$. On the other hand, red edges are added to the same graph by using Lemma 4.4 in order to construct a nonseparable RDG shown in Fig. 4b. Thus Lemma 4.4 suggests a sequence for adding new edges to a separable connected RDG for transforming it into a nonseparable RDG.

Lemma 4.5. It is always possible to construct an MRDG from a nonseparable RDG by adding edges to it.

Proof. Let $\mathcal{G}_1 = (V, E_1)$ be a nonseparable RDG. If $|E_1| = 3|V| - 7$ or $\mathcal{G}_1 = W_n$, then $\mathcal{G}_1$ is itself an MRDG and the proof is obvious.

Suppose that $|E_1| < (3|V| - 7)$ and $\mathcal{G}_1 \neq W_n$. Assume that $L_2 = \{(v_i, v_j) \in E_1 \mid |N(v_i) \cap N(v_j)| = 1\}$. By Lemma 4.1, L_2 is a list of all exterior edges of $\mathcal{G}_1$.

We now prove that there exists atleast a pair of adjacent edges (v_i, v_j) and (v_j, v_k) in L_2 such that $|N(v_i) \cap N(v_k)| = 1$. If such pair does not exist, then $|N(v_a) \cap N(v_c)| = 2$ for each pair (v_a, v_b) and (v_b, v_c) in L_2. In fact, since $\mathcal{G}_1$ is a nonseparable graph, by Lemma 4.2, we have $|N(v_a) \cap N(v_c)| \in \{1, 2\}$. Let $v_1 v_2 \cdots v_p$ be the outermost cycle of $\mathcal{G}_1$. Note that all edges (v_1, v_2), (v_2, v_3),$\ldots$,(v_{p-1}, v_p) and (v_p, v_1) are exterior and hence by Lemma 4.2, all these edges belongs to L_2. Now if we choose (v_1, v_2) and (v_2, v_3), then $|N(v_1) \cap N(v_3)| = \{v_2, v_c\}$. Again if we choose (v_2, v_3) and (v_3, v_4), then $|N(v_2) \cap N(v_4)| = \{v_3, v_c\}$. Continuing in this way, we see that all the exterior vertices are adjacent to v_c. Observe that the vertex v_c and every adjacent exterior vertices v_i and v_j forms a triangle. Therefore, if $\mathcal{G}_1$ has any other vertex (except $v_1, v_2, \ldots, v_p$ and v_c), it would lie inside the triangle $v_i v_j v_c$, which is a separating triangle. This contradicts the fact that $\mathcal{G}_1$ is an RDG. This implies that $\mathcal{G}_1$ cannot have any other vertex (except $v_1, v_2, \ldots, v_p$ and v_c) which concludes that $\mathcal{G}_1$ is a wheel graph W_n which is again a contradiction since we assumed that $\mathcal{G}_1 \neq W_n$. This proves our claim.

Choose two adjacent edges (v_i, v_j) and (v_j, v_k) from L_2 such that $|N(v_i) \cap N(v_k)| = 1$ and construct a graph $\mathcal{G}_2 = (V, E_2)$ where $E_2 = E_1 \cup \{(v_i, v_k)\}$.

Now we show that the number of CIPs in $\mathcal{G}_2$ can not exceed the number of CIPs in $\mathcal{G}_1$. For $\mathcal{G}_2$, there are the following possibilities:

i. None of vertices v_i and v_k is the endpoint of a shortcut in $\mathcal{G}_1$,
ii. One of vertices v_i and v_k is the endpoint of a shortcut in $\mathcal{G}_1$,
iii. Both vertices v_i and v_k are the endpoints of a shortcut in $\mathcal{G}_1$.

These 3 possibilities are shown in Fig. 5a-5c respectively. In the first case, clearly there is no CIP passing through (v_i, v_k) in $\mathcal{G}_2$. In the second case, $v_i v_k v_{k+1} \cdots v_{q-1} v_q$ becomes a CIP in $\mathcal{G}_2$ and $v_i v_j v_k v_{k+1} \cdots v_{q-1} v_q$ no longer remains a CIP in $\mathcal{G}_2$. In fact, the edges (v_i, v_j) and (v_j, v_k) of the existing CIP in $\mathcal{G}_1$ are replaced by (v_i, v_k) in $\mathcal{G}_2$. Thus, in this case, the number of CIPs do not get increased. The third case is not possible since $|N(v_i) \cap N(v_k)| = 2$. In fact, $N(v_i) \cap N(v_k) = \{v_j, v_s\}$ and $\mathcal{G}_2$ is obtained from $\mathcal{G}_1$ by adding an edge (v_i, v_k) such that $|N(v_i) \cap N(v_k)| = 1$. This proves our claim.

Now we claim that there does not exist a separating triangle passing through (v_i, v_k) in $\mathcal{G}_2$. Since $|N(v_i) \cap N(v_k)| = 1$, i.e., $N(v_i) \cap N(v_k) = \{v_j\}$. Therefore $v_i v_j v_k$ is the only cycle of length 3 having no vertex inside and passing through (v_i, v_k) in $\mathcal{G}_2$. This shows that $v_i v_j v_k$ is not a separating triangle, it is a new added triangular region (face) in $\mathcal{G}_2$. By Theorem 2.1, $\mathcal{G}_2$ is an RDG. A recursive process shows that each $\mathcal{G}_i = (V, E_i)$, $(i \geq 3)$ is an RDG where $E_i = E_{i-1} \cup \{(v_a, v_c)\}$ such that $|N(v_a) \cap N(v_c)| = 1$ for some edges $(v_a, v_b), (v_b, v_c)$ belong to L_2 which is defined as $L_2 = \{(v_i, v_j) \in E_{i-1} \mid |N(v_i) \cap N(v_j)| = 1\}$.

It can be noted that the recursive process will terminate when the outermost cycle has four vertices for some RDG $\mathcal{G}_k$. In fact, (v_i, v_j), (v_j, v_k), (v_k, v_l) and (v_l, v_i) are four edges constituted by the four exterior vertices v_i, v_j, v_k and v_l of some RDG $\mathcal{G}_k$. For any two edges (v_a, v_b) and (v_b, v_c), we have $|N(v_a) \cap N(v_c)| = 2$. This terminate our process. On the other hand, there does not exist any other way for adding a new edge such that the resultant graph is an RDG with a new triangular region. Recall that a maximal plane graph has $3|V_1| - 6$ edges where V_1 denotes its vertex set and has all triangular regions including exterior. In our case, every region of $\mathcal{G}_k$ is triangular, but exterior is quadrangle. This implies that the number of edges in $\mathcal{G}_k$ is $3|V| - 7$ and hence it is an MRDG. This completes the proof of lemma. □

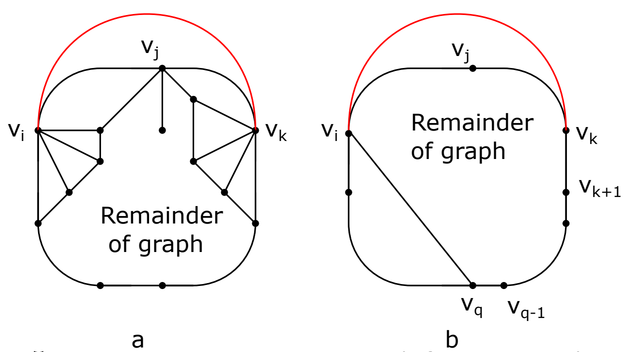

FIGURE 5. Three possible depictions of $\mathcal{G}_2$ obtained from $\mathcal{G}_1$ (consists of black edges) by adding a red edge.

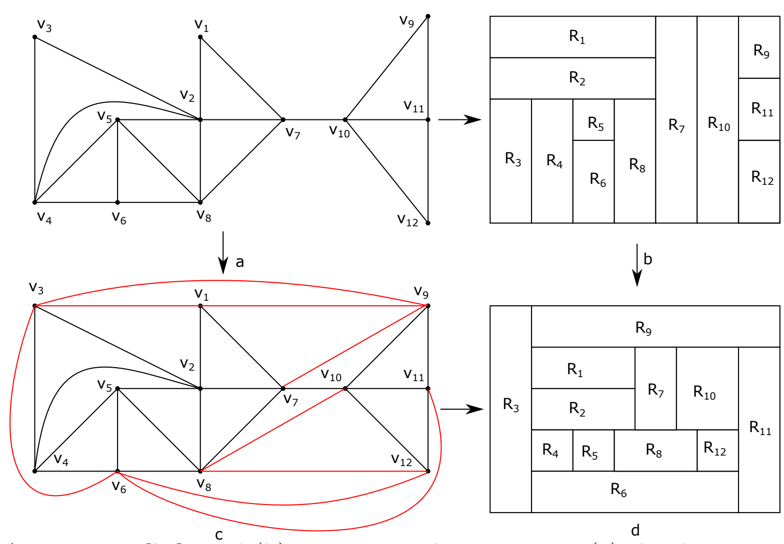

FIGURE 6. A given RDG $\mathcal{G}_1$ and (b) its rectangular partition, (c) the derivation of an MRDG $\mathcal{M}_2$ from $\mathcal{G}_1$, and (d) its rectangular partition.

From Lemmas 4.4-4.5, we conclude that the following main result of the paper.

Theorem 4.1. It is always possible to construct an MRDG for a given RDG.

http://dx.doi.org/10.22108/TOC.2022.133242.1984

Algorithm 1 **Constructing an MRDG of a given RDG**

Input: An RDG $\mathcal{G}_1 = (V, E_1)$

Output: An MRDG $\mathcal{M} = (V, E)$ for $\mathcal{G} = (V, E_1)$ such that $|E_1| < |E|$

1: $L_1 \leftarrow \phi$
2: $L_2 \leftarrow \phi$
3: **for all** $(v_i, v_j) \in E_1$ **do**
4: $\quad s \leftarrow |N(v_i) \cap N(v_j)|$
5: $\quad$ **if** $s == 0$ **then**
6: $\qquad L_1 \leftarrow L_1 \cup \{(v_i, v_j)\}$
7: $\quad$ **else if** $s == 1$ **then**
8: $\qquad L_2 \leftarrow L_2 \cup \{(v_i, v_j)\}$
9: $\quad$ **else**
10: $\qquad$ **continue**
11: $\quad$ **end if**
12: **end for**
13: **for all** $(v_i, v_j) \in L_1$ **do**
14: $\quad$ **if** $(v_j, v_k) \in L_2$ **then**
15: $\qquad L_2 \leftarrow (L_2 \cup \{(v_i, v_j), (v_i, v_k)\}) - \{(v_j, v_k)\}$
16: $\qquad E_1 \leftarrow E_1 \cup \{(v_i, v_k)\}$
17: $\quad$ **else**
18: $\qquad$ **continue**
19: $\quad$ **end if**
20: **end for**
21: **for all** $(v_i, v_j), (v_j, v_k) \in L_2$ **do**
22: $\quad$ **if** $|N(v_i) \cap N(v_k)| == 1$ **then**
23: $\qquad L_2 \leftarrow L_2 \cup \{(v_i, v_k)\} - \{(v_i, v_j), (v_j, v_k)\}$
24: $\qquad E_1 \leftarrow E_1 \cup \{(v_i, v_k)\}$
25: $\quad$ **else**
26: $\qquad$ **continue**
27: $\quad$ **end if**
28: **end for**
29: **for all** $(v_i, v_j) \in E_1$ **do**
30: $\quad$ **if** $(v_i, v_j) \in (E_1 - \{(v_i, v_j)\})$ **then**
31: $\qquad E_1 \leftarrow E_1 - \{(v_i, v_j)\}$
32: $\quad$ **else**
33: $\qquad$ **continue**
34: $\quad$ **end if**
35: **end for**
36: **return** $\mathcal{G}$

Since the output of Algorithm 1 is an MRDG having four vertices on its exterior, the corresponding rectangular partition would have four rectangles on the exterior. It may not always be desirable to transform a given RDG to an MRDG. In such a case, we can replace L_2 by $L_2 - A$ where A is the set of edges not to be added to the given RDG. Thus we can obtain the required RDG from a given RDG.

For a better understanding to Algorithm 1, we explain its steps through an example. Consider an RDG $\mathcal{G}_1 = (V, E_1)$ shown in Fig. 6a. First of all, Algorithm 1 computes two sets L_1 and L_2 (the lines $3 - 12$) from $\mathcal{G}_2$ such that L_1 contains those edges whose endpoints have no common vertex and

L_2 contains those edges whose endpoints have exactly one common vertex. Then $L_1 = \{(v_7, v_{10})\}$ and $L_2 = \{(v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_6), (v_6, v_8), (v_7, v_8), (v_{10}, v_{12}), (v_{11}, v_{12}, (v_9, v_{11}), (v_9, v_{10}), (v_7, v_1)\}$.

Now it executes the rest of its steps ($13 - 20$) as follows:

Since for $(v_7, v_{10}) \in L_1$, there is an edge (v_{10}, v_9) belonging to L_2, the loop $(13-20)$ adds (v_7, v_9) and (v_7, v_{10}) to L_2, and adds (v_7, v_9) to E_1. Further, it subtracts (v_{10}, v_9) from L_2. Thus, $L_2 = \{(v_7, v_{10}), (v_7, v_9), (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_6), (v_6, v_8), (v_7, v_8), (v_{10}, v_{12}), (v_{11}, v_{12}, (v_9, v_{11}), (v_7, v_1)\}$ and $E_1 = E_1 \cup \{(v_7, v_9)\}$. Since L_1 has exactly one edge, this loop terminates (here we have transformed the given separable connected RDG to an nonseparable RDG using the method proposed by Lemma 4.4) and Algorithm 1 executes the next loop $(21 - 28)$ as follows:

Suppose that Algorithm 1 picks (v_9, v_7) and (v_7, v_1) from L_2. Since $N(v_1) \cap N(v_9) = \{v_7\}$, $|N(v_1) \cap N(v_9)| = 1$. Then it subtracts (v_9, v_7) and (v_7, v_1) from L_2 and adds (v_9, v_1) to both L_2 and E_1 (the lines 23 and 24). Thus $L_2 = \{(v_9, v_1), (v_7, v_{10}), (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_6), (v_6, v_8), (v_7, v_8), (v_{10}, v_{12}), (v_{11}, v_{12}, (v_9, v_{11})\}$ and $E_1 = E_1 \cup \{(v_9, v_1), (v_7, v_9)\}$.

Again it picks (v_{10}, v_7) and (v_7, v_8) from L_2. Since $N(v_{10}) \cap N(v_8) = \{v_7\}$, $|N(v_{10}) \cap N(v_8)| = 1$, it subtracts (v_{10}, v_7) and (v_7, v_8) from L_2, and adds (v_{10}, v_8) to both L_2 and E_1 (the lines 23 and 24).Thus, $L_2 = \{(v_{10}, v_8), (v_9, v_1), (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_6), (v_6, v_8), (v_{10}, v_{12}), (v_{11}, v_{12}, (v_9, v_{11})\}$ and $E_1 = E_1 \cup \{(v_{10}, v_8), (v_9, v_1), (v_7, v_9)\}$.

Thus a recursive process of this loop adds $\{(v_6, v_{11}), (v_3, v_6), (v_9, v_3), (v_9, v_{11})\}$ to L_2 and $E_1 = E_1 \cup \{(v_9, v_3), (v_6, v_{12}), (v_8, v_{12}), (v_3, v_6), (v_9, v_3), (v_8, v_{10}), (v_9, v_1), (v_3, v_1), (v_9, v_7)\}$ respectively.

Since no duplicate edge (multiple edges) has been added, the loop $(30 - 35)$ is skipped.

Thus we see that the output is an MRDG $\mathcal{M}_2$ shown in Fig. 6c where red edges are the new edges which are added to $\mathcal{G}_1$. Note that $\mathcal{G}_1$ admits a rectangular partition $\mathcal{F}_1$ shown in 6b and $\mathcal{M}_2$ admits a rectangular partition $\mathcal{F}_2$ shown in 6d. Consequently to this, $\mathcal{F}_1$ can be transformed to $\mathcal{F}_2$ (a maximal one).

Analysis of computational complexity. Let v_s be a vertex of the largest degree in the input RDG $\mathcal{G}$. This implies that $|N(v_i| \leq K$ where $d(v_s) = K$. Now we consider each of the following loops:

i. The computational complexity of the lines $3 - 12$ is $(|N(v_i)||N(v_j)||E_1|) \cong K^2|E_1| \cong \mathrm{O}(n)$.
ii. The computational complexity of the lines $14 - 20$ is $(|L_1|.|L_2|) \cong |L_2| \cong \mathrm{O}(n)$. In fact, L_1 contains edges whose endpoints are cut vertices. $|L_1| \cong \mathrm{O}(n)$ if the given RDG is a path graph. In that case, L_2 is empty. Both L_1 and L_2 can not be large simultaneously.
iii. The computational complexity of the lines $21 - 28$ is $(|N(v_i)||N(v_j)| + |L_2|)|L_2| \cong K^2|L_2|^2 \cong \mathrm{O}(n^2)$.
iv. The computational complexity of the lines $29 - 35$ is $|E_1|^2 \cong \mathrm{O}(n^2)$.

Hence, the computational complexity of Algorithm 1 is quadratic.

http://dx.doi.org/10.22108/TOC.2022.133242.1984

5. **Reduction Method**

In this section, we show that an edge-reducible nonseparable RDG can always be transformed to an edge-irreducible nonseparable RDG. Then we present Algorithm 2 that computes the number of CIPs in a given RDG which is a an input requirement for Algorithm 3 as a call function. Algorithm 3 transforms an edge- reducible nonseparable RDG to an edge-irreducible nonseparable RDG.

Theorem 5.1. If an RDG $\mathcal{G} = (V, E)$ is edge-reducible to an RDG $\mathcal{G}' = (V, E)$ such that $E = E' \cup \{(v_i, v_j)\}$, then (v_i, v_j) is an exterior edge of $\mathcal{G}$.

Proof. Assume that C and C' be the exterior faces of $\mathcal{G}$ and $\mathcal{G}'$ respectively. Since both $\mathcal{G}$ and $\mathcal{G}'$ are RDGs, each interior face of both $\mathcal{G}$ and $\mathcal{G}'$ are of equal length (i.e., of length 3). But $E' \subsetneq E$ and, $\mathcal{G}$ and $\mathcal{G}'$ have the same number of vertices. This implies that C and C' have different length, i.e., $|C| < |C'|$. Also, when (v_i, v_j) is removed from C, the two other edges of the triangle passing through (v_i, v_j) becomes a part of C', i.e., removing an edge from C increases the size of C' by one. Thus we see that $|C'| - |C| = 1$ and (v_i, v_j) belongs to C. Hence (v_i, v_j) is an exterior edge of $\mathcal{G}$. $\square$

Theorem 5.1 suggests us that an edge-reducible RDG can be transformed to any other RDG by deleting some of its exterior edges. Then further such resultant RDG can also be transformed to another RDG by deleting some of its exterior edges. Such recursion process can be continued until the graph remains an RDG. Following this, now we are going to prove another main result of this paper.

Theorem 5.2. It is always possible to transform an edge-reducible nonseparable RDG to an edge-irreducible nonseparable RDG.

Proof. Let $\mathcal{G}_1 = (V, E_1)$ be an edge-reducible nonseparable RDG. Let Z_1 be the set of all exterior edges of $\mathcal{G}_1$. If for each edge $(v_i, v_j) \in Z_1$, $|N(v_i)| \leq 2$ or $|N(v_j)| \leq 2$, then $\mathcal{G}_1 = (V, E_1)$ is an edge-irreducible nonseparable RDG which is a contradiction. Thus, there exists an edge $(v_i, v_j) \in Z_1$ such that $|N(v_i)| > 2, |N(v_j)| > 2$.

Consider a nonseparable graph $\mathcal{G}_2 = (V, E_2)$ with atmost 4 CIPs where $E_2 = E_1 - \{(v_i, v_j)\}$. Such a nonseparable graph always exists otherwise it has more than four CIPs and $\mathcal{G}_1$ becomes an edge-irreducible RDG which contradicts our assumption. Since (v_i, v_j) is an exterior edge and $\mathcal{G}_1 = (V, E_1)$ is a nonseparable RDG, $|N(v_i) \cap N(v_j)| = 1$. Suppose that $(N(v_i) \cap N(v_j)) = \{v_t\}$.

Now we claim that $\mathcal{G}_2$ is an RDG. Since $\mathcal{G}_1$ is an RDG, each of its interior regions are triangular. On removing an exterior edge from an RDG, the remaining interior regions remains triangular. This implies that each interior region of $\mathcal{G}_2$ is triangular. It is evident that the removal of an exterior edge from $\mathcal{G}_1$ does not produce a separating triangle. This shows that $\mathcal{G}_2$ is independent of separating triangles. Also, by our assumption, $\mathcal{G}_2$ has atmost four CIPs. By Theorem 2.1, $\mathcal{G}_2$ is a nonseparable RDG.

By continuously defining $\mathcal{G}_k$ $(k \geq 3)$ as above, a recursive process shows that at the iteration until the above defined conditions remains true, $\mathcal{G}_k$ is an edge-reducible nonseparable RDG. Hence the proof. $\square$

Algorithm 2 NumberOfCIPs($\mathcal{G} = (V, E), W$)

Input: A nonseparable RDG $\mathcal{G} = (V, E)$

Output: Number of CIPs in $\mathcal{G}$

$W \leftarrow \phi$, $L \leftarrow \phi$, $U \leftarrow \phi$, $X \leftarrow \phi$
for all $(v_i, v_j) \in E$ **do**
  $s \leftarrow |N(v_i) \cap N(v_j)|$
  if $s == 1$ **then**
    $L \leftarrow L \cup \{(v_i, v_j)\}$
    $U \leftarrow U \cup \{v_i, v_j\}$
  else
    continue
  end if
end for
for all $(v_i, v_j) \in (E - L)$ **do**
  if $v_i, v_j \in U$ **then**
    $W \leftarrow W \cup \{(v_i, v_j)\}$
    $X \leftarrow X \cup \{v_i, v_j\}$
  else
    continue
  end if
end for
for all $(v_i, v_j) \in W$ **do**
  if $(v_i, v_{i+1}), (v_{i+1}, v_{i+2}), \ldots, (v_{j-1}, v_j) \in L$ **then**
    if $v_k \in X$, $i + 1 \leq k \leq j - 1$ **then**
      $W \leftarrow W - \{(v_i, v_j)\}$
    else if $(v_i, v_{i-1}), (v_{i-1}, v_{i-2}), \ldots, (v_{j+1}, v_j) \in L$ **then**
      if $v_k \in X$, $i + 1 \leq k \leq j - 1$ **then**
        $W \leftarrow W - \{(v_i, v_j)\}$
      end if
    end if
  else
    continue
  end if
end for
return W

In the most design problems, graphs structures of floorplans are nonseparable. Therefore abiding by common design practice, we have described Algorithm 3 for transforming a nonseparable RDG to another nonseparable RDG.

Algorithm 3 **Restoring an edge-reducible nonseparable RDG to an edge-irreducible nonseparable RDG**

Input: A nonseparable RDG $\mathcal{G} = (V, E)$
Output: An edge-irreducible nonseparable RDG $\mathcal{G}' = (V, E')$

1: $Z \leftarrow \phi$
2: **for all** $(v_i, v_j) \in E$ **do**
3:     $s \leftarrow |N(v_i) \cap N(v_j)|$
4:     **if** $s == 1$ **then**
5:         $Z \leftarrow Z \cup \{(v_i, v_j)\}$
6:     **else**
7:         **continue**
8:     **end if**
9: **end for**
10: **for all** $(v_i, v_j) \in Z$ **do**
11:     **if** $|N(v_i)| > 2 \wedge |N(v_j)| > 2 \wedge (N(v_i) \cap N(v_j)) = \{v_t\}$ **then**
12:         NumberOfCIPs($\mathcal{G} = (V, E - \{(v_i, v_j)\}), W$)
13:         **if** $|W| \leq 4$ **then**
14:             $E \leftarrow E - \{(v_i, v_j)\}$
15:             $Z \leftarrow Z \cup \{(v_i, v_t), (v_t, v_j)\} - \{(v_i, v_j)$
16:         **else**
17:             **print** $\mathcal{G}$ is an edge-irreducible nonseparable RDG.
18:         **end if**
19:     **end if**
20: **end for**
21: **return** $\mathcal{G}$

For a better understanding of Algorithm 3, we illustrate its steps through an example. Consider a nonseparable RDG $\mathcal{G}_1 = (V, E_1)$ shown in Fig. 7a. First of all, the first loop (the lines $3 - 9$) of Algorithm 3 computes a set $Z = \{(v_1, v_3), (v_1, v_7), (v_5, v_7), (v_3, v_5)\}$. Now Algorithm 3 executes the steps of second loop (the lines $10 - 20$) as follows:

Suppose that the second loop picks (v_3, v_5) from Z randomly. Then 11^{th} line is executed since $N(v_3) = 4 > 2$, $N(v_5) = 4 > 2$ and $N(v_3) \cap N(v_5) = \{v_4\}$. Next it executes 12^{th} line to determine a set W of CIPs in $\mathcal{G}_2 = (V, E_2)$ where $E_2 = E_1 - \{(v_3, v_5)\}$. Using Algorithm 2, $|W| = 0 \leq 4$. Then $E_1 = E_1 - \{(v_3, v_5)\}$ and it adds both (v_3, v_4) and (v_4, v_5) to Z and remove (v_3, v_5) from Z. Thus $Z = \{(v_3, v_4), (v_4, v_5), (v_1, v_3), (v_1, v_7), (v_5, v_7)\}$.

Next suppose that it picks (v_1, v_3) from Z. Then 11^{th} line is executed since $N(v_1) = 5 > 2$, $N(v_3) = 4 > 2$ and $N(v_1) \cap N(v_3) = \{v_2\}$. Next it executes 12^{th} line to determine the set W for $\mathcal{G}_3 = (V, E_3)$ where $E_3 = E_2 - \{(v_1, v_3)\}$. Using Algorithm 2, we have $W = \{(v_2, v_4)\}$, i.e., $|W| = 1 \leq 4$. Therefore it subtracts (v_1, v_3) from both E_1 and Z, and adds both (v_1, v_2) and (v_2, v_3) to Z. Thus $Z = \{(v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_5), (v_1, v_7), (v_5, v_7)\}$ and $E_1 = E_1 - \{(v_1, v_3), (v_3, v_5)\}$.

In this way, the second loop continues until $|W| \leq 4$ with the condition in 11^{th}. Finally we obtain an edge-irreducible nonseparable RDG shown in Fig. 7b with the edge set $E - \{(v_7, v_9), (v_1, v_9), (v_1, v_7), (v_7, v_5), (v_1, v_3), (v_3, v_5)\}$ and Z (set of the exterior edges) becomes $\{(v_7, v_8), (v_8, v_9), (v_1, v_{10}), (v_{10}, v_9), (v_5, v_6), (v_6, v_7), (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_5)\}$.

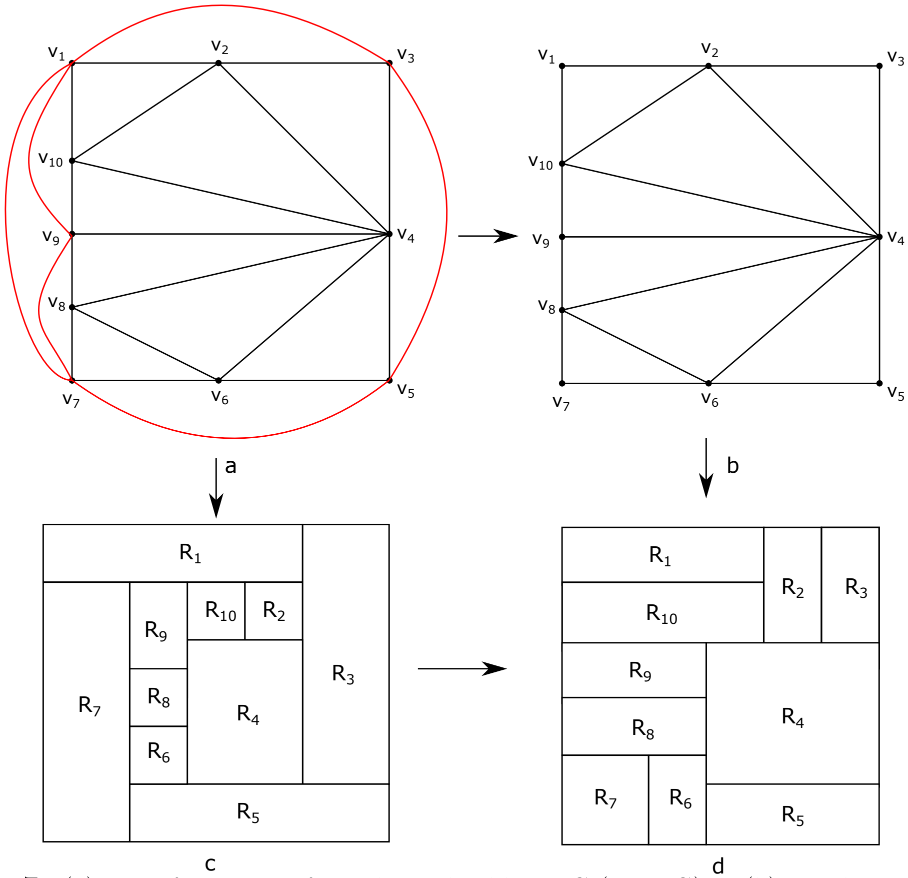

FIGURE 7. (a) Transformation of an edge-reducible RDG (MRDG) to (b) an edge-irreducible RDG. The respective rectangular partitions are shown in (c) and (d).

It can be noted that the edge-irreducible nonseparable RDG in Fig. 7b can not be further transformed to an edge-irreducible separable connected RDG. In fact, the removal of a single edge from it violates RDG property (refer to Theorems 2.1 and 2.2). From this, it follows that a nonseparable edge-irreducible RDG may not necessarily be transformed to a separable connected RDG. It may be sometimes possible to restore it to an edge-reducible separable connected RDG by relaxing the conditions $N(v_i) > 2$ and $N(v_j) > 2$ ($11^{\mathbf{th}}$ line)[3]. Once if it is transformed to an edge-reducible separable connected RDG, Algorithm 3 can be made applicable for separable connected RDGs with a slight modifications as follows.

Consider a separable connected RDG and construct its BNG. Theorem 2.2 tells us that it is a path. Consider each component corresponding to the vertices of the BNG one by one as an input to Algorithm 3. To proceed, first consider the components corresponding to the initial and end vertices of the BNG as an input to Algorithm 3 and follow Theorem 2.2, i.e., restrict the line 13 of Algorithm 3 by $|W| \leq 2$ such that no CIP should pass through a cut vertex.[4] and restore it to an edge-irreducible one, it is possible using Algorithm 3 since the corresponding block is nonseparable. Then, the remaining blocks

[3]This condition sustains non-separableness of the output RDG.

[4]Such CIPs are called critical CIPs.

http://dx.doi.org/10.22108/TOC.2022.133242.1984

must be restricted to $|W| = 0$ in the line 13 of Algorithm 3. In this way, each nonseparable component can be restored to an edge-irreducible one. Finally gluing these edge-irreducible components in the same order as we decomposed them, we obtain an edge-irreducible separable connected RDG from the edge-reducible separable connected RDG.

The ability of Algorithm 3 in the design process is visible as follows.

Algorithm 3 gives an edge-irreducible nonseparable RDG as an output for an input nonseparable RDG. But it may not be always preferred/required. Of course, one can obtain output as the edge-reducible RDG by imposing some restrictions on Z or P_c. Suppose that one desires that a particular set X of adjacency relations must not be removed from the given RDG. Then Z or P_c (in the lines 10 and 13 respectively) needs to be replaced by $Z - X$ or $P_c - X$. This makes Algorithm 3 more practical to design problems. Consequent to this, a particular set of adjacency relations of component rectangles of an existing rectangular partition can be sustained by removing the set X of adjacency relations (edges) of the corresponding vertices in its dual graph as discussed above. Further, Algorithm 3 can give also output as a separable connected RDG for the input nonseparable RDG.

Analysis of computational complexity.

- The computational complexity of Algorithm 2 is linear

  The computational complexity of the lines $2 - 10$ is $|N(v_s)||N(v_t)||E| = K_1K_2|E| \cong \mathrm{O}(n)$. The computational complexity of the lines $11 - 18$ is $|U||E - L| \cong \mathrm{O}(n)$. The computational complexity of the lines $19 - 31$ is $|W||L||X|^2 \cong \mathrm{O}(n)$. Hence the computational complexity of Algorithm 2 is linear.
- The computational complexity of Algorithm 3 is $\mathrm{O}(n^2)$.

  The computational complexity of the lines $3 - 9$ is $|N(v_s)||N(v_t)||E| = K_1K_2|E| \cong \mathrm{O}(n)$. The computational complexity of the lines $10 - 20$ is the product of $|N(v_i)||N(v_j)||Z||P_c||A|$ and the computational complexity of Algorithm 2. But $|N(v_i)||N(v_j)||Z||P_c||A|| \cong \mathrm{O}(n^2)$. Hence the computational complexity of Algorithm 3 is quadratic.

6. Concluding remarks and future work

In this paper, we characterized rectangular partitions in terms of graphs. We showed how to transform an RDG into another RDG of which the edge set is a superset or a subset of the first one in quadratic-time.

We proved that it is always possible to construct an MRDG from a given RDG, where MRDG represents an RDG with maximum adjacency relations among its rectangles. Then we presented an algorithm for its construction from the given RDG. Since adding new edges to an RDG without disturbing RDG property reduce distances among its vertices (usually it is measured by the shortest path between vertices) and hence it is useful in reducing wire-length interconnections among the modules of VLSI floorplans. This method adds new edges to an RDG in bulk if it is a path graph (minimal one that is an RDG). In other words, if we pick a Hamiltonian path of an RDG, then a new desired form of the RDG can constructed by adding edges in bulk. If it is not possible to make some pair of vertices of a given RDG adjacent in its MRDG without disturbing RDG property, then it would be interesting to find a method that can minimize distance between these vertices. In this case, it is equivalent to finding a minimal spanning tree for routability of interconnections.

We also showed that an edge-reducible RDG can be restored to a minimal one (an edge-irreducible RDG) and presented an algorithm to restore the first one to the minimal one. The removal of an edge from a reducible RDG takes an interior vertex to the exterior. Thus it can be very useful for enhancing the input-output connections between VLSI circuit and the outside world. It would be interesting to derive a necessary and sufficient condition for a given RDG to admit an edge-irreducible RDG.

Consequent to these algorithms, we can construct an efficient RDG by deleting or adding edges to a given RDG.

References

[1] E. Ackerman, G. Barequet and R. Y. Pinter, A bijection between permutations and floorplans, and its applications, *Discrete Appl. Math.*, **154** (2006) 1674–1684.

[2] A. Recuero, O. Ríó and M. Alvarez, Heuristic method to check the realisability of a graph into a rectangular plan, *Adv. Eng. Soft.*, **31** (2000) 223–231.

[3] J. Bhasker and S. Sahni, A linear time algorithm to check for the existence of a rectangular dual of a planar triangulated graph, *Networks*, **17** (1987) 307–317.

[4] J. Bhasker and S. Sahni, A linear algorithm to find a rectangular dual of a planar triangulated graph, *Algorithmica*, **3** (1988) 247–278.

[5] B. D. He, A simple optimal binary representation of mosaic floorplans and Baxter permutations, *Theor. Comput. Sci.*, **532** (2014) 40–50.

[6] D. Eppstein, E. Mumford, B. Speckmann and K. Verbeek, Area-universal and constrained rectangular layouts, *SIAM J. Comput.*, **41** (2012) 537–564.

[7] K. Koźmiński and E. Kinnen, Rectangular dual of planar graphs, *Networks*, **15** (1985) 145–157.

[8] S. Felsner, Exploiting air-pressure to map oorplans on point sets, *J. Graph Algorithms Appl.*, **18** (2014) 233–252.

[9] Y.-T. Lai and S. M. Leinwand, Algorithms for floorplan design via rectangular dualization, *IEEE Trans. Comput. Aided Des.*, **7** (1988) 1278–1289.

[10] Y.-T. Lai and S. M. Leinwand, A theory of rectangular dual graphs, *Algorithmica*, **5** (1990) 467–483.

[11] I. Rinsma, Rectangular floorplan and orthogonal floorplans with required room areas and tree adjacency, *Environment and Planning B: Planning and Design*, **15** (1988) 111–118.

[12] K. Koźmiński and E. Kinnen, Rectangular dualization and rectangular dissections, *IEEE Trans. Circuits and Systems.*, **35** (1988) 1401–1416.

[13] K. Yamanaka, M. S. Rahman and S.-I. Nakano, *Floorplans with columns*, Combinatorial optimization and applications. Part I, Lecture Notes in Comput. Sci., Springer, Cham, (2017) 33–40.

[14] H. Tang and W.-K. Chen, Generation of rectangular duals of a planar triangulated graph by elementary transformations, *IEEE Int. Symp. Circuits and Systems*, **4** (1990) 2857–2860.

[15] B. Yao, and H. Chen, C.-k. Cheng and R. Graham, Floorplan Representations: Complexity and Connections, *ACM Trans. Des. Autom. Electron. Syst.*, **8** (2003) 55–80.

[16] N. Reading, Generic rectangulations, *European J. Combin.*, **33** (2012) 610–623.

[17] K. Shekhawat, Enumerating generic rectangular floor plans, *Autom. Constr.*, **92** (2018) 151–165.

[18] C.-W. Sham and E. F. Y. Young, Area reduction of dead space utilization on interconnect optimized floorplan, *ACM Trans. Des. Automat. Elect. Syst.*, **12** (2007) 1–11.

[19] V. Kumar and K. Shekhawat, A transformation algorithm to construct a rectangular floorplan, *Theoret. Comput. Sci.*, **871** (2021) 94–106.

[20] X. He, On finding the rectangular duals of planar triangular graphs, *SIAM J. Comput.*, **22** (1993) 1218–1226.

[21] X.-Y. Wang, Y. Yang and K. Zhang, Customization and generation of floor plans based on graph transformations, *Autom. Constr.*, **94** (2018) 405–416.

Vinod Kumar

Department of Mathematics, Birla Institute of Technology and Science, Pilani, Pilani Campus, Rajasthan-333031, India

Email: `vinodchahar04@gmail.com`

Krishnendra Shekhawat

Department of Mathematics, Birla Institute of Technology and Science, Pilani, Pilani Campus, Rajasthan-333031, India

Email: `krishnendra.shekhawat@pilani.bits-pilani.ac.in`