Recognition and domination on intersection and overlap graphs of rectangles

by

Dibyayan Chakraborty
Advanced Computing and Microelectronics Unit
Indian Statistical Institute
Kolkata, India

A thesis submitted in partial fulfillment
of the requirement for the degree of
Doctor of Philosophy
in
Computer Science

Thesis Advisor: Prof. Sandip Das

Abstract

A *rectangle intersection representation* $\mathcal{R}$ of a graph G with vertex set $V(G)$ and edge set $E(G)$ is a collection of axis-parallel rectangles $\{R_u\}_{u \in V(G)}$ such that $uv \in E(G)$ if and only if R_u, R_v intersect. If a graph has a rectangle intersection representation then it is a *rectangle intersection graph*. We introduce a parameter called *stab number* of rectangle intersection graphs and study the structural properties of rectangle intersection graphs with bounded stab number. We introduce a natural generalisation of "asteroidal triples" and show that certain structures are forbidden in rectangle intersection graphs. We also propose polynomial-time recognition algorithms for several subclasses of rectangle intersection graphs with stab number at most 3.

A *rectangle overlap representation* $\mathcal{R}$ of a graph G is a collection of rectangles $\{R_u\}_{u \in V(G)}$ such that $uv \in E(G)$ if and only if the boundaries of R_u, R_v intersect. A graph is a rectangle overlap graph if it has a rectangle overlap representation. The classes of rectangle intersection graphs and rectangle overlap graphs are subclasses of *string graphs*, the intersection graphs of simple curves on the plane. We propose constant factor approximation algorithms for the Minimum Dominating Set problem on subclasses of rectangle overlap graphs and string graphs.

Contents

Author List

Contents of Chapter 2 are part of a journal article accepted in Theory of Computing Systems (TOCS). The following authors contributed to Chapter 2 : Mathew C. Francis.

Contents of Chapter 3 are part of a paper published in the proceedings of CALDAM 2019. The following authors contributed to Chapter 3: Sandip Das, Mathew C. Francis and Sagnik Sen.

Contents of Chapter 4 are part of a paper published in the proceedings of COCOA 2016. The following authors contributed to Chapter 4: Sujoy Bhore, Sandip Das and Sagnik Sen.

Contents of Chapter 5 are part of a paper published in the proceedings of COCOON 2019. The following authors contributed to Chapter 5: Sandip Das and Joydeep Mukherjee.

Contents of Chapter 6 are part of a paper published in the proceedings of WG 2019. The following authors contributed to Chapter 6: Sandip Das and Joydeep Mukherjee.

Listing of figures

To my parents.

Acknowledgments

First and foremost, I must acknowledge my thesis supervisor, Dr Sandip Das. I started working with him after he agreed to guide me for the master's thesis. Throughout my time as a graduate student, he has provided me with unending guidance in research and genuine friendship. He gave me the freedom to work on whatever problems I wanted. He allowed me to travel everywhere and work with everyone. The latter was the essential feature that allowed me to reach where I am now. His patience while dealing with my limitations and faults is an admirable quality. I hope to learn that too.

I would like to thank Dr Mathew C. Francis for teaching me how to convert simple observations into essential theorems. I spent a total of 4 months working with him in ISI, Chennai. During this time, he provided me with numerous advice and tips on how to build up my understanding of a difficult problem, one observation at a time. These are a few of the many experiences that I shall relish for the rest of my life.

A special thanks to Dr Sagnik Sen and Dr Joydeep Mukherjee. They have greatly influenced me, providing new problems to work on together, providing new approaches and insights, and teaching me new techniques for solving problems so that I can apply them myself. I have learnt a lot of things from them, and collaboration with them brought an enjoyable social aspect to research. Academic visits to France with Sagnik have created some everlasting memories. Those visits were valuable sources of research experience. Joydeep taught me how to attack a seemingly difficult problem and finally reach a conclusion by proving one observation at a time. Being able to work with him has certainly made me more confident as a researcher. Both Sagnik and Joydeep have been friends, philosophers and guides.

I thank Uma kant Sahoo for many advice that he provided. Food tours with him are few of the many experiences that I shall relish for the rest of my life. I thank Harmendar, Shubhadeep and Arun for making our laboratory a source of happiness.

I thank Dr Sandip Das for creating an environment which enabled me to meet Sagnik, Joydeep, Uma kant, Harmendar, Shubhadeep and Arun.

I thank Dr Arijit Bishnu for arranging the beautiful trek to Ruinsara Tal. It was a calming experience amidst the chaos of thesis writing and career planning. I am incredibly fortunate in having Dr Arijit Bishnu, Dr Subhas C. Nandy, Dr Ansuman Banerjee, Dr Bhargab B. Bhattacharya, Dr Nabanita Das, Dr Sasthi C. Ghosh and Dr Susmita Sur-Kolay as my teachers.

To mother Dr Dipanwita Chakraborty and father Dr Krishnananda Chakraborty, thank you for everything. You will undoubtedly be relieved to see this thesis put to rest.

A PERSON COULD MAKE A WHOLE CAREER ON ALGORITHMS FOR INTERSECTION GRAPHS!

Martin Charles Golumbic

1

Introduction

Contents

Given a collection of sets, $\mathcal{C}$, the *intersection* graph of $\mathcal{C}$ is the graph, whose vertices correspond to the elements of $\mathcal{C}$, and two vertices are joined by an edge if and only if the corresponding sets have a nonempty intersection. When $\mathcal{C}$ is a collection of geometric objects, the corresponding class of intersection graphs is a *geometric intersection* graph.

A popular graph class is the intersection graphs of intervals on the real line, i.e., *interval graphs*. Study of interval graphs dates back to the 1950s. Benzer [17] established a direct relation between interval graphs and arrangements of genes in the chromosome. Later, researchers have encountered interval graphs in applications like scheduling, seriation in archaeology, behavioural psychology, planning, medical diagnosis, temporal reasoning in artificial intelligence, circuit design etc. [89, 91, 92, 105, 133].

Researchers have studied graph classes like *circular-arc graphs, probe interval graphs, rectangle intersection graphs, disk graphs, segment graphs* [145] etc. These graph classes are generalisations of interval graphs and are intersection graphs of simple curves on the plane *i.e. string graphs*. Any intersection graph of arc-connected sets on the plane is a string graph. But not all graphs are string graphs [71]. Hence, there are some graphs that are not intersection graphs of any collection of arc-connected two-dimensional sets. But such graphs are intersection graphs of higher-dimensional geometric objects, and this motivates the study of higher-dimensional analogues of interval graphs. Roberts [138] introduced the notion of *boxicity* of a graph G, which is the minimum integer d such that G is an intersection graph of d-dimensional rectangles. A graph has boxicity 1 if it is an interval graph. Apart from being of theoretical interests, the graph classes mentioned above have proven to be useful tools for modelling applications from various domains [17, 20, 104, 139]. Naturally, researchers have studied geometric intersection graphs. We divide the entire literature on geometric intersection graphs into three main categories depending on whether a result is about (i) finding characterisation of a class of geometric intersection graph or (ii) proving bound on a parameter or (iii) determining the computational complexity of an algorithmic problem on geometric intersection graphs.

1.1 Characterisation of geometric intersection graphs

A *characterisation* of a graph class $\mathcal{G}$ is a set of properties $\mathcal{P}_{\mathcal{G}}$ such that a graph $G \in \mathcal{G}$ if and only if G satisfies all properties in $\mathcal{P}_{\mathcal{G}}$. The problem of characterising interval graphs was first posed independently by Hajos [96] in combinatorics and by Benzer [17] in genetics. Boland and Lekkerkerker [26] gave the first characterisation of interval graphs.

A graph G has vertex set $V(G)$ and edge set $E(G)$. A graph G is a *chordal* graph if it has no induced cycle of length greater than 3. Given a vertex $v \in V(G)$, we say that a path P *misses* v, if no vertex in P is a neighbour of v. Three vertices $a, b, c \in V(G)$ are said to form an *asteroidal triple*, or AT for short, in G if there exists a path between any two vertices in $\{a, b, c\}$ that misses the third. A graph is said to be *AT-free* if it contains no asteroidal triple.

Theorem 1.1.1 ([26]). *A graph G is an interval graph if and only if G is chordal and AT-free.*

Theorem 1.1.1 is a *forbidden structure characterisation* of interval

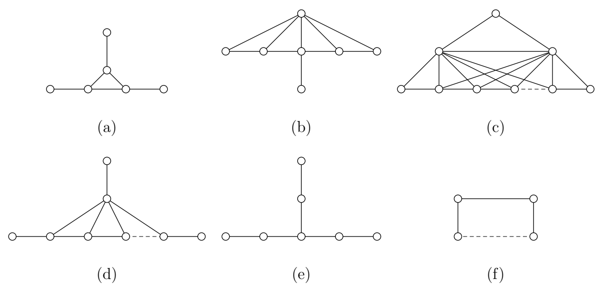


Figure 1.1.1: All Minimal forbidden induced subgraphs of interval graphs.

graphs. Boland and Lekkerkerker [26] found the graph classes shown in Figure 1.1.1. They proved that any chordal graph containing an asteroidal triple must also contain one of the graphs from the graph classes shown in Figure 1.1.1(a)-(e), as induced subgraphs.

Theorem 1.1.2 ([26])**.** *A graph G is an interval graph if and only if G does not contain any graph from the graph classes shown in Figure 1.1.1 as induced subgraphs.*

A *forbidden induced subgraph characterisation* of a graph class $\mathcal{G}$ is a set of graphs $\mathcal{F}_{\mathcal{G}}$ such that a graph $G \in \mathcal{G}$ if and only if G does not contain any graph of $\mathcal{F}_{\mathcal{G}}$ as an induced subgraph. Theorem 1.1.2 is the first instance of a forbidden induced subgraph characterisation and the second instance of a forbidden structure characterisation (after *Kuratowski's Theorem* [151]) of a non-trivial graph class. Theorem 1.1.2 motivated the researchers to prove analogous theorems for other graph classes [91, 145]. However, finding forbidden structure characterisations of graph classes can be challenging tasks. Two classic examples are *circular-arc graphs* and *probe interval graphs*.

Circular-arc graphs are the intersection graphs of *circular arcs of a circle*. The first characterisation of circular-arc graphs appeared in 1970 [149] and finding a forbidden induced subgraph characterisation of circular-arc graphs is a challenging open problem. Many partial results toward this goal have been proposed over the years, but a full answer remains elusive, capturing the interest of many researchers [27, 78, 111, 120, 148, 149]. In 2014, Francis et al. [81] gave a forbidden structure characterisation of circular-arc graphs.

A graph G is a *probe interval graph* if there is a partition of $V(G)$ into sets P and N and a collection $\{I_v : v \in V(G)\}$ of intervals of $\mathbb{R}$ such that, for $u, v \in V(G)$, $uv \in E(G)$ if and only if $I_u \cap I_v \neq \emptyset$ and at least one of u, v belongs to P. Interval probe graphs were introduced by Zang [154] to model certain problems in physical mapping of DNA when only partial data is available on the overlap of clones (i.e., the intervals) [155, 156]. There are some characterisations of probe interval graphs [87, 126] but forbidden structure characterisations are known for some subclasses of probe interval graphs [31–33, 135].

Probe interval graphs are subclasses of intersection graphs of *axis-parallel rectangles* on the plane i.e., *rectangle intersection graphs*. Specifically, the following is an interesting observation. A graph G is a probe interval graph if and only if there is a partition $V(G)$ into sets P and N such that (i) there is a set $\mathcal{R} = \{r_v : v \in P\}$ of rectangles each of whose bottom boundary lies on the x-axis, (ii) there is a set $\mathcal{S} = \{s_v : v \in N\}$ of disjoint horizontal segments each of them lying above x-axis, and (iii) G is an intersection graph of $\mathcal{R} \cup \mathcal{S}$.

The earliest reference to the study of rectangle intersection graphs can be traced back to the work of Bielecki [23]. However, very little is known about the structure of rectangle intersection graphs. Graph classes like *outerplanar* graphs, *planar bipartite* graphs, *halin* graphs, *block* graphs, AT-free graphs with girth at least 5 are all subclasses of rectangle intersection graphs [22, 49, 97, 140]. On the other hand, there are examples of *series-parallel* graphs, AT-free, *split* graphs that are not rectangle intersection graphs [22, 25, 62]. This motivates the following question.

Question 1.1.1. *Is there a forbidden structure characterisation for rectangle intersection graphs?*

Structural properties of other generalisations of interval graphs like intersection graphs of disks (*disk graphs*) and line segments (*segment graphs*) on the plane have been widely studied in the literature [11, 30, 36–38, 45, 53, 103, 123, 124]. *Circle Packing Theorem* implies that planar graphs are disk graphs. Scheinerman [140] conjectured that all planar graphs are segment graphs. Chalopin and Gonçalves [45] settled the above conjecture.

Theorem 1.1.3 ([45])**.** *Every planar graph is a segment graph.*

West [150] conjectured that every planar graph is the intersection graph of line segments using only four directions. Recently, Gonçalves [93] proved that all planar graphs with *chromatic number* [151] at most three are intersection graphs of line segments using only three directions. Since disk graphs and segment graphs are string graphs, one might consider characterising string graphs. Very little is known in this case. This motivates the following question(s).

Question 1.1.2. *Is there a forbidden structure characterisation of segment graphs or string graphs?*

Researchers have studied many subclasses of segment graphs and string graphs. Let $\mathcal{G}$ denote a class of geometric intersection graphs, and $\mathcal{O}$ be another set of geometric objects. A subclass $\mathcal{H} \subseteq \mathcal{G}$ *captures a local structure of* $\mathcal{G}$ *with respect to* $\mathcal{O}$ if every graph $H \in \mathcal{H}$ has an intersection representation $\mathcal{R}$ such that each object in $\mathcal{R}$ interacts with the objects of $\mathcal{O}$ according to some specified notion. The set $\mathcal{O}$ is a *localizer* of $\mathcal{G}$. Below we give two examples that capture local structures of less understood classes of geometric intersection graphs.

Permutation graph: A *permutation graph* is a graph whose vertices represent the elements of a permutation, and whose edges represent pairs of elements that are reversed by the permutation. Observe that, a graph is a permutation graph if and only if it is an intersection graph of segments whose endpoints lie on two parallel lines. Permutation graphs capture a local structure of segment graphs when the localizer is a pair of parallel lines.

Co-comparability graph: A graph G is a *comparability* graph if edges of G admit an *transitive orientation* [91]. A graph G is a *co-comparability graph* if $\overline{G}$ is a comparability graph. In 1983, Golumbic et al. [90] proved that a *co-comparability graph* on n vertices are intersection graphs of n continuous curves set $f_i\colon [0,1] \to \mathbb{R}(1 \leq i \leq n)$. Co-comparability graphs capture a local structure of string graphs when the localizer is a pair of parallel lines. Interestingly, Fox and Pach [80] proved that every "dense" string graph contains a "dense" spanning subgraph that is a co-comparability graph. Eventually, the above observation proves the following extremal property of string graphs.

Theorem 1.1.4 ([80])**.** *For every $\epsilon > 0$, there exists $\delta > 0$ such that every string graph with n vertices and at least ϵn^2 edges contains a bi-clique with parts of size $\frac{\delta n}{\log n}$.*

Characterisations of co-comparability graphs and permutation graphs [58, 91, 116, 145] motivated us to study graph classes that capture local structures of rectangle intersection graphs. See Section 1.4 for details.

1.2 Bounds on various parameters of geometric intersection graphs

Researchers have used the rich structural properties and characterisations of geometric intersection graphs to prove bounds on various graph parameters. A well-studied parameter is the chromatic number of geometric intersection graphs. Even before the formal introduction of interval graphs, Bielecki [23] and Rado [136] proved that the maximum number of pairwise intersecting intervals is equal to the minimum number of classes in a partition into subclasses containing pairwise disjoint intervals. This implies the following theorem.

Theorem 1.2.1 ([23, 136])**.** *The chromatic number an interval graph equals its clique number.*

Observe that every induced subgraph of an interval graph is an interval graph. Therefore the above theorem implies that the chromatic number of every induced subgraph of an interval graph equals its clique number. This means that interval graphs are *perfect graphs* [91]. A graph G is a *perfect graph* if the chromatic number of every induced subgraph of G equals its clique number. Graph classes like probe interval graphs and co-comparability graphs, are perfect graphs. *The Strong Perfect Graph Theorem* [55] states that a graph G is a perfect graph if and only if neither G nor $\overline{G}$ contain an induced odd cycle. Therefore rectangle intersection graphs are not perfect graphs (since the cycle of order 5 is a rectangle intersection graph). This raises the natural question: Can the chromatic number of rectangle intersection graphs be bounded in terms of the clique number? Asplund and Grünbaum [10] gave an answer to the above question.

Theorem 1.2.2 ([10])**.** *The chromatic number of a rectangle intersection graph with clique number ω is at most $8\omega^2$.*

Asplund and Grünbaum [10] posed the following question.

Question 1.2.1. *Is there an absolute constant c such that chromatic number of any rectangle intersection graph G is at most c times the clique number of G.*

The above question has remained open for about 60 years. However, for some subclasses of rectangle intersection graphs, researchers have answered the above question in affirmative [43, 114]. Asplund and

Grünbaum [10] asked if the chromatic number of triangle-free intersection graphs of 3-dimensional rectangles is bounded by a constant. Burling [34] answered the above question in negative. The chromatic number of other geometric intersection graphs have been studied extensively. See [114, 142] for a detailed survey on the topic.

Chandran et al. [47] proved that if the boxicity of a graph G with n vertices is equal to $\frac{n}{2} - s, s \geq 0$, then the chromatic number of G is at least $\frac{n}{2s+2}$. It is natural to ask if the converse is true. Formally, is it true that for all graphs G, the boxicity of G, denoted by $box(G)$, is upper bounded by some function of the chromatic number of G? The answer to the above question is negative. Adiga et al. [6] proved that for any positive constant $c < 1$, almost all balanced bipartite graphs on $2n$ vertices and $m \leq cn^2$ edges have boxicity $\Omega(m/n)$. Hence, the difference between the chromatic number and boxicity of a graph can be arbitrarily high. Chandran et al. [48] proved that for any graph G, $box(G)$ is at most the chromatic number of G^2 where G^2 is the graph obtained from G by adding edges between vertices having some common neighbours in G.

For special classes of graphs, it might be possible to bound the boxicity in terms of its chromatic number. Bhowmick and Chandran [22] proved that the boxicity of an AT-free graph is at most its chromatic number. Chandran et al. [51] proved that for a *line graph* G, $box(G) = O(\chi(G) \log\log(\chi(G))$, where $\chi(G)$ denotes the chromatic number of G. Their proof depends on the fact that for a *line graph* G, $box(G) = 2\Delta(G)(\lceil \log\log \Delta(G) \rceil + 3) + 1$ where $\Delta(G)$ denotes the maximum degree of G. This raises the natural question of bounding the boxicity of a graph in terms of its maximum degree. Adiga et al. [5] proved the following theorem.

Theorem 1.2.3 ([5]). *For any graph G having maximum degree Δ, there exists a constant c' such that $box(G) < c'\Delta(log\ \Delta)^2$. Moreover, there exists a graph G with maximum degree Δ and $box(G) = \Omega(\Delta \log \Delta)$.*

The above theorem was improved by Scott and Wood [143] who proved that, for every graph G with maximum degree Δ, as $\Delta \to \infty$, $box(G) \leq 6(180 + o(1))\Delta \log(\Delta)(2e)^{\sqrt{\log\log\Delta}} \log\log\Delta$. Boxicity of special graph classes like chordal graphs, circular-arc graphs, AT-free graphs and co-comparability graphs are bounded by linear functions of the maximum degree. Chandran and Sivadasan [52] proved the above results by relating boxicity and *treewidth*.

Theorem 1.2.4 ([52])**.** *For a graph G, $box(G) \leq tw(G)+2$, where $tw(G)$ is the treewidth of G.*

Researchers have also studied the relationships between boxicity and other graph parameters. See [48, 50, 52, 74–76, 143] for more results.

1.3 Computational complexity of algorithmic problems on geometric intersection graphs

Researchers have studied the computational complexities of many algorithmic problems on geometric intersection graphs. We classify the literature into two sections *viz.* (i) recognition algorithms for geometric intersection graphs and (ii) algorithms for optimisation problems on geometric intersection graphs.

1.3.1 Recognition algorithms for geometric intersection graphs

For a graph class $\mathcal{G}$, the *recognition problem* for $\mathcal{G}$ is to decide whether some given input graph belongs to $\mathcal{G}$ and any algorithm to solve a recognition problem is a *recognition algorithm*. Theorem 1.1.1 provides a $O(n^4)$ time algorithm to recognise interval graphs with n vertices. Booth and Lueker [28] proved the following.

Theorem 1.3.1 ([28])**.** *There is an $O(n+m)$ time algorithm to recognise an interval graph with n vertices and m edges.*

Since the result of Booth and Lueker [28], researchers have proposed different recognition algorithms for interval graphs. Korte and Möhring [113] proposed a simpler incremental algorithm to recognise interval graphs. Hsu [101] gave a recognition algorithm for interval graphs that directly placed the intervals without precomputing all maximal cliques. Habib et al. [95] gave a *Lex-BFS* based linear time algorithm to recognise interval graphs. Corneil et al. [59] gave a *4-sweep Lex-BFS* based algorithm to recognise interval graphs.

When the input graph is an interval graph, all the algorithms discussed so far produce an interval representation. However, if the input graph is not an interval graph, then these algorithms do not output any structure which is forbidden for interval graphs. In other words, the algorithms are not *certifying* in nature.

A *certifying algorithm* for a decision problem is an algorithm that provides a certificate with each answer that it produces [122]. A *certifying recognition algorithm* for a class $\mathcal{G}$ of geometric intersection graphs must produce either a valid intersection representation of the input graph G that acts as the evidence that $G \in \mathcal{G}$ or output some structures that are known to be absent in any graph that belongs to $\mathcal{G}$. Finding a forbidden structure characterisation of a graph class $\mathcal{G}$ is important for designing a certifying recognition algorithm for $\mathcal{G}$. Kratsch et al. [116] used Theorem 1.1.1 to give the first certifying recognition algorithm for interval graphs.

Researchers have studied the computational complexities of recognition problems for many generalisations of interval graphs. A particular case of interest is the class of rectangle intersection graphs. Yannakakis [153] questioned the existence of polynomial-time recognition algorithms for rectangle intersection graphs and Kratochvíl [115] proved the following theorem.

Theorem 1.3.2 ([115]). *Recognising rectangle intersection graphs is* NP-*Complete.*

There is a recognition algorithm for rectangle intersection graphs based on the following principle. For any rectangle intersection graph G there exist two interval graphs I_1 and I_2, both having the same vertex set and any edge of G is an edge in both I_1 and I_2. Since there are only $O(2^{n \log n})$ interval graphs having n vertices, we can decide if G is a rectangle intersection graph or not, in $\tilde{O}(4^n)$ time. The above algorithm produces a rectangle intersection representation of the input graph if it exists. If the input graph is not a rectangle intersection graph, then the algorithm does not provide a certificate. This motivates the following question.

Question 1.3.1. *Is there a certifying recognition algorithm for rectangle intersection graphs?*

To answer Question 1.3.1, we need a forbidden structure characterisation of rectangle intersection graphs.

Researchers have studied the computational complexities of recognition problems for many subclasses of rectangle intersection graphs. A particular case of interest is the class of intersection graphs of axis-parallel unit squares on the plane, i.e., *unit square intersection* graphs. Recognising unit square intersection graphs is an NP-complete problem [30]. Since all rectangle intersection graphs are not unit square intersection graphs (e.g. $K_{1,5}$), it makes sense to study the computational complexity of recognising unit square intersection graphs when the inputs are restricted to rectangle intersection graphs. Since all trees are rectangle intersection graphs, the following question is interesting.

Question 1.3.2. *Is there a polynomial-time algorithm to recognise trees that are also unit square intersection graphs?*

In this thesis, we study the computational complexities of the recognition problems for graph classes that capture local structures of rectangle intersection graphs and unit square intersection graphs. See Section 1.4 for details.

1.3.2 Algorithms for optimisation problems on geometric intersection graphs

Suppose there is a resource, and there are some requests to use the resource for specific time intervals. Assuming that the resource can serve only one request at a time, the job of a scheduler is to choose the maximum number of such requests that can be served by the resource. Solving the above problem is equivalent to solving the Maximum Independent Set [84] problem on interval graphs. The above example is one of many real-world applications that motivate the study of computational complexities of optimisation problems on geometric intersection graphs [2, 8, 41, 43, 46, 66, 86, 104, 108].

An *independent set* of an undirected graph G is a set of pairwise non-adjacent vertices. The *independence number* of a graph G is the maximum integer k such that G has an independent set with cardinality k. The Maximum Independent Set (MIS) problem is to find an independent set of an input graph with maximum cardinality. There are several polynomial-time algorithms to solve the MIS problem on interval graphs [21, 144, 147]. On the other hand, the MIS problem is NP-complete for rectangle intersection graphs [102]. Therefore researchers have focused on designing *approximation algorithms* for the MIS problem on rectangle intersection graphs.

An *α-approximation algorithm* for an optimisation problem Π is a polynomial-time algorithm that for all instances of the problem produces a solution whose value is within a factor of α of the value of an optimal solution for that instance. If $ALG(I)$ is the value of the solution computed by an algorithm and $OPT(I)$ is the value of the optimal solution on input instance $I \in \Pi$ then, $OPT(I) \leq ALG(I) \leq \alpha \cdot OPT(I)$ (for minimization problems) or $OPT(I) \geq ALG(I) \geq \alpha \cdot OPT(I)$ (for maximization problems) for every instance I.

The MIS problem turns out to be significantly harder in case of rectangle intersection graphs. No constant factor approximation algorithms is known for the MIS problem on rectangle intersection graphs. The best-known hardness result is on *strong NP-hardness* [102]. Chalmersook and Chuzhoy [44] gave an $O(\log \log n)$-approximation algorithm for the MIS problem on rectangle intersection graphs. Lewin-Eytan et al. [119] gave a $4q$-approximation algorithm for the MIS problem on rectangle intersection graphs having clique number q. When the optimal independent set of a rectangle intersection graph with n vertices has size βn for some $\beta \leq 1$, Agarwal and Mustafa [7] gave an algorithm that computes an independent set of size $\beta^2 n$. Adamaszek et al. [3] gave an $(1-\epsilon)$-approximation algorithm for the MIS problem on intersection graphs of n polygons on the plane with a quasi-polynomial running time of $O(2^{poly(\log n, \frac{1}{\epsilon})} \cdot n \log n)$. Adamaszek et al. [3] also gave a *polynomial-time approximation scheme* (PTAS) for the MIS problem on intersection graphs of δ-large rectangles for any $\delta > 0$, i.e., for the case when each rectangle has at least one side of length at least δN, assuming that in the input only integer coordinates within $\{0, \dots, N\}$ occur. But the following question remains open.

Question 1.3.3. *Is there a polynomial-time constant-factor approximation algorithm for the* MIS *problem on rectangle intersection graphs?*

A closely related minimisation problem is the Minimum Vertex Cover (MVC) problem. A *vertex cover* of an undirected graph G is a subset D of vertices such that each edge of G has at least one endpoint in D. The *vertex cover number* of a graph G is the minimum integer k such that G has a vertex cover with cardinality k. The Minimum Vertex Cover (MVC) problem is to find a vertex cover of an input graph with minimum cardinality. Observe that, the number of vertices of a graph is equal to its vertex cover number plus its independence number. The observation implies a polynomial-time algorithm for the MVC problem on interval graphs and NP-hardness for the MVC problem on rectangle intersection graphs, respectively.

There is a polynomial-time 2-approximation algorithm for the MVC problem on general graphs. Bar-Yehuda et al. [16] gave an *efficient polynomial-time approximation scheme* (EPTAS) for the MVC problem on intersection graphs of non-crossing rectangles, i.e., the case where $R_1 \setminus R_2$ is connected for every pair of input rectangles R_1, R_2. Bar-Yehuda et al. [16] also gave an $(1.5+\epsilon)$-approximation algorithm for the MVC problem on rectangle intersection graphs. But the following question remains open.

Question 1.3.4. *Is there a PTAS for the* MVC *problem on rectangle intersection graphs?*

Observe that a vertex cover of a graph G contains at least one neighbour of each vertex of G. In other words, a vertex cover of a graph G is also a *dominating set* of G. Formally, a dominating set of an undirected graph G is a subset D of vertices such that each vertex in $V(G) \setminus D$ is adjacent to some vertex in D. The Minimum Dominating Set (MDS) problem is to find a dominating set of an input graph with minimum cardinality. Since the formal introduction of dominating set [19, 130], more than 500 research papers have been written on the topic. For a survey see [98–100].

Due to the NP-complete nature of the MDS problem [84], researchers have focused on designing approximation algorithms for the MDS problem. Unless $P = NP$, for any $\epsilon > 0$, there is no polynomial-time $o(\log n)$-approximation algorithm for the MDS on general graphs [69]. Therefore, researchers have focused on studying the complexity of the MDS problem on geometric intersection graphs. Intersection graphs of disks with unit radius i.e. *unit disk graphs* have attracted a significant amount of research in this direction. In 1995, Marathe et al. [121] gave the first constant factor approximation algorithm for the MDS problem on unit disk graphs. Nieberg and Hurink [129] proposed a PTAS for the MDS problem on unit disk graphs. Carmi et al. [39] proposed several approximation algorithms for the MDS problem on unit disk graphs. Gibson and

Pirwani [88] used the *local search technique* introduced by Mustafa and Ray [128] to give a PTAS for the MDS problem on disk graphs. The same technique was later generalised to obtain PTAS for the MDS problem on intersection graphs of *non-piercing* 2D objects and intersection graphs of homothets of convex objects [67, 94]. The complexity of the MDS problem on rectangle intersection graphs is less understood. Erlebach and Van Leeuwen [73] proved that MDS problem is *APX-hard* [152] on rectangle intersection graphs and proposed an $O(1)$-approximation algorithm for the MDS problem on intersection graphs of rectangles with bounded aspect ratio. The following remains a challenging open problem.

Question 1.3.5. *Is there a constant-factor approximation algorithm for the* MDS *problem on rectangle intersection graphs?*

Pandit [132] introduced the intersection graphs of *diagonally anchored* rectangles. A set $\mathcal{R}$ of rectangles is a set of *diagonally anchored* rectangles if there is a straight line l with slope -1 such that intersection of any $R \in \mathcal{R}$ with l is exactly one corner of R. The MDS problem remains NP-Hard on intersection graphs of diagonally anchored rectangles [132]. Bandyapadhyay et al. [13] gave a $(2+\epsilon)$-approximation algorithm for the same using the local search technique of Mustafa and Ray [128]. Intersection graphs of diagonally anchored rectangles is a subclass of *rectangle overlap graphs*. A *rectangle overlap representation* $\mathcal{R}$ of a graph G is a set of rectangles $\{R_u\}_{u \in V(G)}$ such that $uv \in E(G)$ if and only if the boundaries of R_u and R_v intersect. A graph G is a rectangle overlap graph if G has a rectangle overlap representation. Fulkerson and Gross [82] introduced a more general notion of *overlap* graphs of a collection of sets. Rim and Nakajima [137] formally initiated the study of rectangle overlap graphs, but few results are known [117, 137]. We focus on the following question.

Question 1.3.6. *Is there a constant-factor approximation algorithm for the* MDS *problem on rectangle overlap graphs?*

As the rectangle overlap graphs and the rectangle intersection graphs are subclasses of string graphs, one might hope to propose a constant-factor approximation algorithm for the MDS problem on string graphs. However, the existence of such an algorithm is unlikely. It is known that all chordal graphs are string graphs and solving the MDS problem on chordal graphs are asymptotically equivalent to solving the MDS problem on general graphs. Therefore, unless $P = NP$, there is no polynomial-time $o(\log n)$-approximation algorithm for the MDS problem on string graphs.

Asinowski et al. [9] introduced the concept of *B_k-VPG* graphs and *bend number*. A *path* is a simple rectilinear curve and a *k-bend* path is a path having k bends. The *B_k-VPG* graphs are intersection graphs of k-bend paths. A graph G is a *VPG* graph if G is a B_k-VPG graph for some k. The *bend number* of a graph G, denoted by *bend*(G), is the minimum integer k for which G has an intersection representation of k-bend paths. Asinowski et al. [9] proved that VPG graphs are equivalent to string graphs. Chaplick et al. [54] proved that for every $k \geq 0$, B_k-VPG $\subsetneq$ B_{k+1}-VPG. Therefore, it makes sense to investigate the complexity of the MDS problem on B_k-VPG graphs, for a fixed $k \geq 0$. Katz et al. [107] have already studied the MDS problem on B_0-VPG graph under the name of *orthogonal segment domination* problem. Their result implies that it is NP-Hard to solve the MDS problem on B_k-VPG graphs with $k \geq 0$. However, the following question is open.

Question 1.3.7. *Is there an $f(k)$-approximation algorithm for the* MDS *problem on B_k-VPG graphs for any $k \geq 0$.*

An affirmative answer to the Question 1.3.7 would give an affirmative answer to Question 1.3.6.

The MDS problem remains difficult in subclasses of B_1-VPG graphs. An L-path is a 1-bend path having the shape 'L'. A set of L-paths is *vertically-stabbed* if all L-paths in the set intersect a common vertical line.

A graph G is a *vertically-stabbed* L*-graph* if G is an intersection graph of vertically-stabbed L-paths. McGuinness [125] introduced the class of vertically-stabbed L-graphs. It contains *interval graphs*, *outerplanar graphs*, *permutation graphs*, *circle graph* as subclasses. Researchers have studied the MDS problem on these classes of graphs ([29, 57, 64, 65, 77]). Bandyapadhyay et al. [13] proved that the MDS problem is APX-hard on vertically-stabbed L-graphs. An algorithm of Mehrabi [127] implies an $O(1)$-approximation algorithm for the MDS problem on vertically-stabbed L-graphs. This motivates the following question.

Question 1.3.8. *What is the optimal approximation ratio for the* MDS *problem on vertically-stabbed* L*-graphs?*

In this thesis, we shall address Questions 1.3.6-1.3.8.

1.4 Contributions and thesis overview

We now present a summary of the results presented in the thesis.

1.4.1 Stab number of rectangle intersection graphs

In Chapter 2, we introduce the following graph classes. A graph G is said to be a *k-stabbable rectangle intersection* graph, or *k-SRIG*, if it has a rectangle intersection representation in which k horizontal lines can be chosen such that each rectangle is intersected by at least one of them. If there exists such a representation with the additional property that each rectangle intersects exactly one of the k horizontal lines, then the graph G is said to be a *k-exactly stabbable rectangle intersection* graph, or *k-ESRIG*. The *stab number*, *stab*(G), of a graph G is the minimum integer k such that G is a k-SRIG. Similarly, the *exact stab number*, *estab*(G), of a graph G, is the minimum integer k such that G is a k-ESRIG. Observe that the class 2-SRIG captures a local structure of rectangle intersection graphs when the localizer is a pair of horizontal lines.

We provide tight upper bounds on the exact stab number of several classes of rectangle intersection graphs. We show that for $k \leq 3$, k-SRIG is equivalent to k-ESRIG. We introduce a natural generalisation of asteroidal triples and show that certain structures are forbidden in rectangle intersection graphs. We believe this to be a positive step towards answering Question 1.1.1. We show that these forbidden structures are sufficient to characterise block graphs that are 2-ESRIG and trees that are 3-ESRIG. Our observations lead to polynomial-time recognition algorithms for these two classes of graphs. We show that these forbidden structures are not sufficient to characterise block graphs that are 3-SRIG or trees that are k-SRIG for any $k \geq 4$. We also show that for any $k \geq 10$, there is a tree that is a k-SRIG but not a k-ESRIG.

1.4.2 Rectangle intersection graphs of stab number at most 2

In Chapter 3, we introduce some natural subclasses of 2-SRIG and study the containment relationships among them. We show that one of these subclasses can be recognised in linear-time if the input graphs are restricted to be triangle-free. We also show that the Chromatic Number problem is NP-complete for 2-SRIGs.

1.4.3 Recognising trees that are 2-SUIG

A graph G is a *2-stabbable unit square intersection graph* or *2-SUIG*, if G is an intersection graph of axis-parallel unit squares on the plane with stab number at most two. Observe that, 2-SUIG captures a local structure of unit square intersection graphs when the localizer is a pair of horizontal lines. In Chapter 4, we give a linear-time algorithm to recognise trees that are 2-SUIG. This addresses Question 1.3.2.

1.4.4 Dominating set of stabbed rectangle overlap graphs

In Chapter 5, we show that if the *Unique Games Conjecture* [110] is true, it is not possible to have a polynomial-time $(2-\epsilon)$-approximation algorithm for the MDS problem on rectangle overlap graphs. A set $\mathcal{R}$ of rectangles is *stabbed* if there is a straight line that intersects all rectangles in $\mathcal{R}$. A graph G is a *stabbed rectangle overlap* graph if G has a stabbed rectangle overlap representation. Observe that, the class of stabbed overlap graphs captures a local structure of rectangle overlap graphs when the localizer is a straight line. We give a 768-approximation algorithm on *stabbed rectangle overlap* graph. The above results address Question 1.3.6.

1.4.5 Dominating set of vertically-stabbed L-graphs and unit B_k-VPG graphs

In Chapter 6, we give an 8-factor approximation algorithm for the MDS problems on vertically-stabbed L-graphs. This addresses Question 1.3.8. For $k \geq 0$, *unit B_k-VPG* graphs are intersection graphs of simple rectilinear curves on the plane such that each curve in the set has at most k bends, and each segment of each of the curves have unit length. We show that the MDS problem on unit B_0-VPG graphs is NP-hard, strengthening a result of Katz et al. [107]. We propose an $O(k^4)$-approximation algorithm for the MDS problem on unit B_k-VPG graphs. This solves a special case of Question 1.3.7.

1.4.6 Conclusion

Finally, in Chapter 7, we discuss some open problems and possible directions for future research to conclude the thesis.

2

Stab number of rectangle intersection graphs

Contents

We study the structure of rectangle intersection graphs by introducing the notion of *stab number* and *exact stab number* of rectangle intersection graphs. Below we recall some definitions from Chapter 1 and introduce some new definitions also.

A *rectangle intersection representation* of a graph is a collection of axis-parallel rectangles on the plane such that each rectangle in the collection represents a vertex of the graph and two rectangles intersect if and only if the vertices they represent are adjacent in the graph. The graphs that have rectangle intersection representation are called *rectangle intersection graphs*. A *k-stabbed rectangle intersection representation* is a rectangle intersection representation, along with a collection of k horizontal lines called *stab lines*, such that every rectangle intersects at least one of the stab lines. A graph G is a *k-stabbable rectangle intersection* graph (*k-SRIG*), if there exists a k-stabbed rectangle intersection representation of G. The *stab number* of a rectangle intersection graph, denoted by $stab(G)$, is the minimum integer k such that there exists a k-stabbed rectangle intersection representation of G. In other words $stab(G)$ is the minimum integer k such that G is k-SRIG.

A *k-exactly stabbed rectangle intersection representation* is a k-stabbed rectangle intersection representation in which every rectangle intersects exactly one of the stab lines. A graph G is a *k-exactly stabbable rectan-*

gle intersection graph, or k-*ESRIG* for short, if there exists a k-exactly stabbed rectangle intersection representation of G. The *exact stab number* of a rectangle intersection graph, denoted by $estab(G)$, is the minimum integer k such that there exists a k-exactly stabbed rectangle intersection representation of G. In other words, $estab(G)$ is the minimum integer k such that G is k-ESRIG. Note that for a graph G, $stab(G) \leq estab(G)$ and that a graph G is an interval graph if and only if $stab(G) = estab(G) = 1$, or in other words, the class of interval graphs, the class of 1-SRIGs, and the class of 1-ESRIGs are all the same.

A *unit height rectangle intersection* graph G is a graph that has a rectangle intersection representation in which all rectangles have equal height.

For a subclass $\mathcal{C}$ of rectangle intersection graphs, $stab(\mathcal{C}, n)$ is the minimum integer k such that any graph $G \in \mathcal{C}$ with n vertices satisfies $stab(G) \leq k$, and $estab(\mathcal{C}, n)$ is the minimum integer k such that for any graph $G \in \mathcal{C}$ with n vertices satisfies $estab(G) \leq k$.

2.1 Chapter overview

In Section 2.2, we give some definitions and notation that will be used throughout the chapter. We prove some basic results about k-SRIGs and k-ESRIGs in Section 2.3. We first show a simple necessary and sufficient condition for a graph to be a k-ESRIG and also show why the classes k-SRIG and k-ESRIG are equivalent when $k \leq 3$ (Theorem 2.3.2). Then we prove that the class of unit height rectangle intersection graphs is a proper subset of the class of graphs which have a k-exactly stabbed rectangle intersection representation (Theorem 2.3.3), which is a proper subset of rectangle intersection graphs (Theorem 2.3.4).

In Section 2.4, we show a lower bound on the stab number of rectangle intersection graphs in terms of the clique number and the pathwidth, and then study upper bounds on the stab number of rectangle intersection graphs that are also (a) split graphs, or (b) block graphs. In particular, we show (a) that all rectangle intersection graphs that are also split graphs have exact stab number at most 3 and that this bound is tight, and (b) an upper bound of $\lceil \log m \rceil$ on the exact stab number of block graphs with m blocks (this bound is shown to be asymptotically tight in Section 2.7).

Then in Section 2.5, we describe a forbidden structure for k-SRIG and k-ESRIG, which we call "asteroidal-(non-($k-1$)-SRIG)" subgraphs and "asteroidal-(non-($k-1$)-ESRIG)" subgraphs respectively. These obstructions are a natural generalization of the well-known asteroidal-triples of Lekkerkerker and Boland [26], which are obstructions for interval graphs. In Section 2.5.2, we discuss some general properties possessed by the block-trees of graphs without these kinds of obstructions. In Section 2.6, we show that the absence of these forbidden structures is enough to characterize block graphs that are 2-ESRIG (Theorem 2.6.1) and trees that are 3-ESRIG (Theorem 2.6.2). These results lead to polynomial-time algorithms to recognize block graphs that are 2-SRIG and trees that are 3-SRIG.

Then we explore the natural question of whether there exists a constant c such that every tree is a c-SRIG. We give a negative answer to this question in Section 2.7.

We use the machineries developed in Section 2.7 to show that the forbidden structure characterizations of Theorems 2.6.1 and 2.6.2 do not extend to block graphs that are 3-ESRIG (equivalently 3-SRIG, by Theorem 2.3.2) or trees that are k-SRIG for any $k \geq 4$. We prove the above results in Section 2.8.

In Theorem 2.3.3, we proved that $K_{4,4}$ is not an exactly stabbable rectangle intersection graph. This leads us to the natural question of finding exactly stabbable graphs whose exact stab number is strictly greater than the stab number. Using several lemmas proved in Section 2.7 and 2.8, we show that for each $k \geq 10$, there exist trees which are k-SRIG but not k-ESRIG (Theorem 2.9.1). Therefore, even for graphs that are exactly stabbable, like trees (Theorem 2.4.4), the stab number and the exact stab number may differ. We prove this result in Section 2.9. Finally, we draw conclusions in Section 2.10.

2.2 Preliminaries

We present some definitions in this section. Let G be a graph with vertex set $V(G)$ and edge set $E(G)$. Let $N(v) = \{u \in V(G) \colon uv \in E(G)\}$ and $N[v] = N(v) \cup \{v\}$ denote the *open neighbourhood* and the *closed neighbourhood* of a vertex v, respectively. For $S \subseteq V(G)$, we denote by $G[S]$ the subgraph induced in G by the vertices in S, and by $G - S$ the graph obtained by removing the vertices in S from G. For an edge $e \in E(G)$, we denote by $G - e$ the graph on vertex set $V(G)$ having edge set $E(G) \setminus \{e\}$.

Let G be a rectangle intersection graph with rectangle intersection representation $\mathcal{R}$. A rectangle in $\mathcal{R}$ corresponding to the vertex v is denoted as r_v. All rectangles considered in this article are closed rectangles. Denote by x_v^+ (x_v^-), the $x-$coordinate of the right (left) bottom corner of r_v. Also y_v^+ (y_v^-) is the $y-$coordinate of the left top (bottom) corner of r_v. In other words, $r_v = [x_v^-, x_v^+] \times [y_v^-, y_v^+]$. The *span* of a vertex u, denoted as *span*(u), is the projection of r_u on the $X-$axis, i.e. *span*$(u) = [x_u^-, x_u^+]$. For two intervals $I_1 = [a_1, b_1]$ and $I_2 = [a_2, b_2]$, we write $I_1 < I_2$ to indicate that $b_1 < a_2$. Clearly, $I_1 \cap I_2 = \emptyset$ if and only if $I_1 < I_2$ or $I_2 < I_1$. For an edge $uv \in E(G)$, we define *span*$(uv) =$ *span*$(u) \cap$ *span*(v).

Let G be a k-SRIG with a k-stabbed rectangle intersection representation $\mathcal{R}$ in which the stab lines are $y = a_1$, $y = a_2$, …, $y = a_k$, where $a_1 < a_2 < \cdots < a_k$. The *top* (resp. *bottom*) stab line of $\mathcal{R}$ is the stab line $y = a_k$ (resp. $y = a_1$). For $1 \leq i < k$, we say that $y = a_{i+1}$ is the stab line "just above" the stab line $y = a_i$ and that $y = a_i$ is the stab line "just below" the stab line $y = a_{i+1}$. We also say that the stab lines $y = a_i$ and $y = a_{i+1}$ are "consecutive". A vertex $u \in V(G)$ is said to be "on" a stab line if r_u intersects that stab line. Two vertices u, v of G "have a common stab" if there is some stab line that intersects both r_u and r_v. Similarly, a set of vertices is said to have a common stab if there is one stab line that intersects the rectangles corresponding to each of them. It is easy to see that if $uv \in E(G)$, then there must be either a stab line such that u and v are on it or two consecutive stab lines such that u is on one of them and v is on the other. Whenever the k-stabbed rectangle intersection representation of a graph G under consideration is clear from the context, the terms r_u, x_u^-, x_u^+, y_u^-, y_u^+, for every vertex $u \in V(G)$ and usages such as "on a stab line", "have a common stab", "span" etc. are considered to be defined with respect to this representation. Clearly, both the classes k-SRIG and k-ESRIG are closed under taking induced subgraphs. We say that a graph is a non-k-SRIG (resp. non-k-ESRIG) if it is not a k-SRIG (resp. k-ESRIG). Similarly, we say that a graph is a non-interval graph if it is not an interval graph.

2.3 Basic results

Given a collection $\mathcal{I}$ of intervals, a *hitting set* X of $\mathcal{I}$ is a subset of $\mathbb{R}$ such that each interval in $\mathcal{I}$ contains at least one element of X. The set X is an *exact hitting set* of $\mathcal{I}$ if each interval in $\mathcal{I}$ contains exactly one element of X. An interval graph G is said to have an exact hitting set *of size* k if there exists an interval representation $\mathcal{I}$ of G that has an exact hitting set of cardinality k. Note that some collections of intervals may not have an exact hitting set of any cardinality. Also, there are interval graphs (for example, $K_{1,4}$) that have no exact hitting set.

Theorem 2.3.1. *A graph G is a k-ESRIG if and only if there exists two interval graphs I_1 and I_2 such that $V(G) = V(I_1) = V(I_2)$ and $E(G) = E(I_1) \cap E(I_2)$ and at least one of I_1, I_2 has an exact hitting set of size k.*

Proof. First we prove that if G has a k-exactly stabbed rectangle intersection representation, then there exist two interval graphs I_1 and I_2 such that $V(G) = V(I_1) = V(I_2)$ and $E(G) = E(I_1) \cap E(I_2)$ and at least one of them has an exact hitting set of size k. Let $\mathcal{R}$ be a k-exactly stabbed rectangle intersection representation of G and $\{y = a_1, y = a_2, \ldots, y = a_k\}$ be the set of stab lines in $\mathcal{R}$. Let I_x, I_y be the interval graphs formed by taking the projections of the rectangles in $\mathcal{R}$ on the X and Y axes, respectively. In other words, I_x is the interval graph given by the interval representation $\{[x_u^-, x_u^+]\}_{u \in V(G)}$ and I_y is the interval graph given by the interval representation $\{[y_u^-, y_u^+]\}_{u \in V(G)}$. It is clear that $V(G) = V(I_x) = V(I_y)$ and $E(G) = E(I_x) \cap E(I_y)$. Furthermore, the set $S = \{a_1, a_2, \ldots, a_k\}$ is an exact hitting set of the interval representation $\{[y_u^-, y_u^+]\}_{u \in V(G)}$ of I_y. Hence, I_y has an exact hitting set of size k.

Now assume that there exist two interval graphs I_1 and I_2 such that $V(G) = V(I_1) = V(I_2)$ and $E(G) = E(I_1) \cap E(I_2)$ and at least one of them, say I_1, has an exact hitting set of size k. Let $S = \{a_1, a_2, \ldots, a_k\}$ be an exact hitting set of an interval representation $\{[c_u, d_u]\}_{u \in V(G)}$ of I_1. Also, let $\{[c'_u, d'_u]\}_{u \in V(G)}$ be an interval representation of I_2. For each $u \in V(G)$, define $r_u = [c'_u, d'_u] \times [c_u, d_u]$. It is easy to see that $\mathcal{R} = \{r_u\}_{u \in V(G)}$ is a rectangle intersection representation of G. Further, the lines $y = a_1$, $y = a_2$, $\ldots$, $y = a_k$ are horizontal lines such that each rectangle in $\mathcal{R}$ intersects exactly one of them. Hence, $\mathcal{R}$, together with these lines, is a k-exactly stabbed rectangle intersection representation of G and therefore, G is a k-ESRIG. This completes the proof. $\square$

Theorem 2.3.2. *When $k \leq 3$, the classes k-SRIG and k-ESRIG are equivalent.*

Proof. If a graph G is k-ESRIG for some k, then G is also k-SRIG. Therefore it suffices to prove that if a graph G has a k-stabbed rectangle intersection representation for some $k \leq 3$, then G also has a k-exactly stabbed rectangle intersection representation. If $k = 1$, then there is nothing to prove. So we shall assume that $k \in \{2, 3\}$. Let $\mathcal{R}$ be a k-stabbed rectangle intersection representation of a graph G with $k \leq 3$ with stab lines $y = 0$, $y = 1$, …, $y = k - 1$. We can assume without loss of generality that for any two distinct vertices $u, v \in V(G)$, we have $\{y_u^+, y_u^-\} \cap \{y_v^+, y_v^-\} = \emptyset$ and that for any vertex $v \in V(G)$, we have $\{y_v^+, y_v^-\} \cap \{0, 1, 2\} = \emptyset$ (note that if this is not the case, then the rectangles in $\mathcal{R}$ can be perturbed slightly so that these conditions are satisfied). Let $S = \{y_v^+, y_v^-\}_{v \in V(G)} \cup \{0, 1, 2\}$ and ϵ be a positive real number such that $\epsilon < \min\{|a - b| \colon a, b \in S, a \neq b\}$. Let $M = \{u \in V(G) \colon r_u$ intersects the stab line $y = 1\}$. For each vertex $u \in M$, define $r'_u = [x_u^-, x_u^+] \times [y_u'^-, y_u'^+]$, where $y_u'^- = \max\{\epsilon, y_u^-\}$ and $y_u'^+ = \min\{2 - \epsilon, y_u^+\}$. Let $\mathcal{R}'$ be the rectangle intersection representation given by the collection of rectangles $(\mathcal{R} \setminus \{r_u \colon u \in M\}) \cup \{r'_u \colon u \in M\}$. It is now easy to verify that $\mathcal{R}'$ is a k-exactly stabbed rectangle intersection representation of G. Indeed, $\mathcal{R}'$ is obtained from $\mathcal{R}$ by the vertical shortening of some of the rectangles intersecting the stab line $y = 1$, and we only need to show that every rectangle that is so shortened still intersects with all the rectangles with which it originally has an intersection. The definition of ϵ guarantees that in $\mathcal{R}$, the bottom edge of any rectangle is no higher than $2 - \epsilon$ and the top edge of any rectangle is no lower than ϵ. So when a rectangle is shortened in the manner described above, it does not become disjoint from a rectangle with which it previously had a nonempty intersection. Therefore $\mathcal{R}$ is a valid rectangle intersection representation of G. It is clear that any rectangle that intersects the stab line $y = 1$ in $\mathcal{R}$ intersects only the stab line $y = 1$ in $\mathcal{R}'$. This implies that $\mathcal{R}'$ is a k-exactly stabbed rectangle intersection representation of G. $\square$

In the following theorem, we show that for $k = 4$, the classes k-SRIG and k-ESRIG differ.

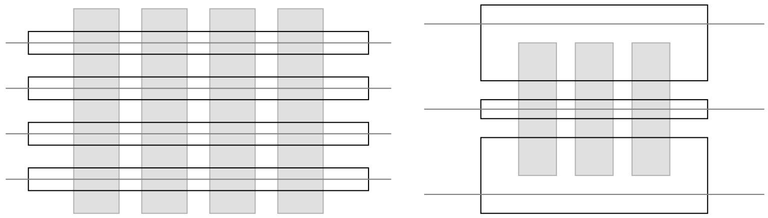


Figure 2.3.1: (a) A 4-stabbed rectangle intersection representation of $K_{4,4}$, (b) a 3-exactly stabbed rectangle intersection representation of $K_{3,3}$.

Theorem 2.3.3. *There is a graph G such that $stab(G) \leq 4$ and G is not a k-ESRIG for any k.*

Proof. We let $G = K_{4,4}$, i.e. the complete bipartite graph in which each partite set contains four vertices each. Clearly, G is a rectangle intersection graph with $stab(G) \leq 4$ (see Figure 2.3.1(a)). We shall prove that G is not an exactly stabbable rectangle intersection graph. First we prove the following claim.

Claim. Let C be a cycle of length four and $E(C) = \{ab, bc, cd, da\}$. There is no k-exactly stabbed rectangle intersection representation of C, for any integer k, in which a, c have a common stab and b, d have a common stab.

Assume for the sake of contradiction that there is a k-exactly stabbed rectangle intersection representation $\mathcal{R}$ of C, for some integer k, in which a, c have a common stab and b, d have a common stab. Clearly, a, b, c, d cannot all be on one stab line (as C is not an interval graph). Since every vertex is on exactly one stab line and because $ab \in E(C)$, we can assume without loss of generality that a, c are on the stab line just below the stab line on which b, d are. Since a, c and b, d are nonadjacent in C, again without loss of generality we can assume that $span(a) < span(c)$. Since $b \in N(a) \cap N(c)$, we can infer that $[x_a^+, x_c^-] \subset span(b)$. Similarly, we can show that $[x_a^+, x_c^-] \subset span(d)$. But this implies that $[x_a^+, x_c^-] \subset$

$span(b) \cap span(d)$. Since b, d are on the same stab line, this means that $r_b \cap r_d \neq \emptyset$. As $bd \notin E(C)$, this contradicts the fact that $\mathcal{R}$ is a rectangle intersection representation of C. This proves the claim.

Now suppose that G has a k-exactly stabbed rectangle intersection representation $\mathcal{R}$ for some k. Let V_1, V_2 be the two partite sets of G (recall that G is isomorphic to $K_{4,4}$) and $v \in V_1$ be a vertex on some stab line ℓ. Since each vertex is on exactly one stab line, and all vertices of V_2 are adjacent to v, we know that each vertex of V_2 must be on the stab line ℓ, on the stab line just above ℓ, or on the stab line just below ℓ. By Pigeon Hole Principle, there exists $u, w \in V_2$ such that u and w are both on one of these stab lines, say ℓ_1. Now, for the same reason as before, each vertex of V_1 must be on the stab line ℓ_1, on the stab line just above ℓ_1, or on the stab line just below ℓ_1. Again by Pigeon Hole Principle, there are two vertices $u', w' \in V_1$ such that u' and w' are both on one of these stab lines. Now, consider the cycle C of length four with $E(C) = \{u'u, uw', w'w, wu'\}$, that is an induced subgraph of G. It can be seen that the rectangles in $\mathcal{R}$ corresponding to the vertices of C form a k-exactly stabbed rectangle intersection representation of C in which u', w' have a common stab and u, w have a common stab. This contradicts the claim proved above. Therefore, G cannot have a k-exactly stabbed rectangle intersection representation for any k. □

Corollary 1. *The class of exactly stabbable rectangle intersection graphs is a proper subset of the class of rectangle intersection graphs.*

The above theorem shows that there are graphs whose stab number is a constant but their exact stab number is infinite. Later on, in Section 2.9 (Theorem 2.9.1), we shall show that there are even trees whose stab number and exact number differ.

Theorem 2.3.4. *The class of unit height rectangle intersection graphs is a proper subset of the class of exactly stabbable rectangle intersection graphs.*

Proof. We shall first give a proof for the well-known fact that every unit height rectangle intersection graph is an exactly stabbable rectangle intersection graph. We shall prove the following stronger claim.

Claim. Given a unit height rectangle intersection representation $\mathcal{R}$ for a graph G, there exists a set of horizontal lines $y = a_1$, $y = a_2$, $\ldots$, $y = a_k$ (for some integer k), where $a_1 < a_2 < \cdots < a_k$, such that each rectangle in $\mathcal{R}$ intersects exactly one of them and $a_1 = \min_{u \in V(G)}\{y_u^+\}$.

Let $a = \min_{u \in V(G)}\{y_u^+\}$ and let $S = \{u \colon u \in V(G) \text{ and } a \in [y_u^-, y_u^+]\}$. Now consider the unit height rectangle intersection representation $\mathcal{R}' = \mathcal{R} \setminus \{r_u\}_{u \in S}$ of $G' = G - S$. By the induction hypothesis, there exists a set of horizontal lines $y = a'_1$, $y = a'_2$, $\ldots$, $y = a'_{k'}$, for some integer k', where $a'_1 < a'_2 < \cdots < a'_{k'}$, such that each rectangle in $\mathcal{R}'$ intersects exactly one of them and $a'_1 = \min_{u \in V(G')}\{y_u^+\}$. Since every rectangle in $\mathcal{R}'$ lies completely above the horizontal line $y = a$, we have that $\min_{u \in V(G')}\{y_u^+\} > a + 1$. Therefore, we have $a'_1 - a > 1$. Since $a'_1 < a'_2 < \cdots < a'_{k'}$, this means that for $1 \leq i \leq k'$, no rectangle of $\mathcal{R}$ intersects both the horizontal lines $y = a'_i$ and $y = a$. Since every rectangle in $\{r_u\}_{u \in S}$ intersects the horizontal line $y = a$, and every rectangle in $\{r_u\}_{u \in V(G')}$ intersects exactly one of the horizontal lines $y = a'_1$, $y = a'_2$, $\ldots$, $y = a'_{k'}$, it follows that each rectangle of $\mathcal{R}$ intersects exactly one of the horizontal lines $y = a$, $y = a'_1$, $y = a'_2$, $\ldots$, $y = a'_{k'}$. This proves the claim.

We shall now show the existence of an exactly stabbable rectangle intersection graph that is not a unit height rectangle intersection graph. Consider the graph $K_{3,3}$, i.e. the complete bipartite graph in which each partite set contains three vertices each. Clearly, $K_{3,3}$ is an exactly stabbable rectangle intersection graph (see Figure 2.3.1(b)). We shall prove that $K_{3,3}$ is not a unit height rectangle intersection graph.

A rectangle intersection representation $\mathcal{R}$ of a graph G is *crossing-free* if for any two rectangles r_u and r_v in $\mathcal{R}$, the regions $r_u \setminus r_v$ and $r_v \setminus r_u$

are both arc-connected. Note that a unit height rectangle intersection representation of a graph is crossing-free. We shall show that if a triangle-free graph G has a crossing-free rectangle intersection representation, then G must be a planar graph. It then follows directly that $K_{3,3}$ is not a unit height rectangle intersection graph.

Let $\mathcal{R}$ be a crossing-free rectangle intersection representation of a triangle-free graph G and let $S \subseteq V(G)$ be the set of vertices of G having degree one. Let $H = G - S$. Clearly, G is planar if and only if H is planar. Let $\mathcal{R}'$ be obtained from $\mathcal{R}$ by removing all the rectangles corresponding to the vertices in S. Note that H is a triangle-free graph and $\mathcal{R}'$ is crossing-free.

Claim. *There is no rectangle in $\mathcal{R}'$ which is contained in some other rectangle of $\mathcal{R}'$.*

Assume for the sake of contradiction that for vertices $u, v \in V(H)$ we have $r_u \subseteq r_v$ in $\mathcal{R}'$. Since u is a vertex of H, we know that u must have degree at least two in G. Let w be a neighbour of u other than v in G. Then in $\mathcal{R}$, we have $r_w \cap r_u \neq \emptyset$. Since $r_u \subseteq r_v$, this implies that $r_w \cap r_v \neq \emptyset$. But now u, v, w form a triangle in G, contradicting the fact that G is triangle-free. This proves the claim.

Since H is triangle-free, we have that in H, for any vertex $u \in V(H)$ and any two vertices in $v, w \in N(u)$, $r_v \cap r_w = \emptyset$. This, together with the fact that $\mathcal{R}'$ is crossing free, implies that the region $r_u \setminus \bigcup_{w \in N(u)} r_w$ is arc-connected and non-empty. (To see this, observe that if $r_u \setminus \bigcup_{w \in N(u)} r_w$ is non-empty, but is not arc-connected, then there exists two points $x, y \in r_u$ and a simple curve $\mathbf{c} \subseteq \bigcup_{w \in N(u)} r_w$ such that x and y are in different arc-connected components of $r_u \setminus \mathbf{c}$. Since for any two vertices in $v, w \in N(u)$, we have $r_v \cap r_w = \emptyset$, we know that there exists some $z \in N(u)$ such that $\mathbf{c} \subseteq r_z$. But this means that x and y are in different arc-connected components of $r_u \setminus r_z$, contradicting the fact that $\mathcal{R}'$ is crossing-free. If $r_u \setminus \bigcup_{w \in N(u)} r_w$ is empty, then $r_u \subseteq \bigcup_{w \in N(u)} r_w$. Again, since for any two

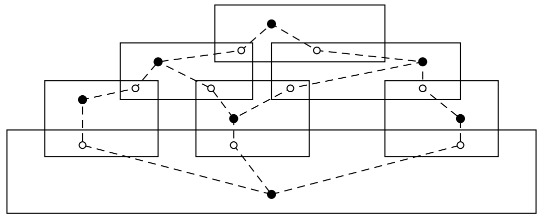


Figure 2.3.2: The dotted curves along with the solid points endpoints, give a planar embedding of the intersection graph of the rectangles in the figure. The hollow circle contained in the intersection region of two rectangles, say r_u and r_v, represents the point p_{uv}.

vertices in $v, w \in N(u)$, we have $r_v \cap r_w = \emptyset$, it must be the case that there exists some $z \in N(u)$ such that $r_u \subseteq r_z$. But this contradicts the claim proved above.) Now choose for every vertex $u \in V(H)$, a point p_u in $r_u \setminus \bigcup_{w \in N(u)} r_w$. In other words, p_u is a point in r_u which is not contained in any rectangle other than r_u. For every edge $uv \in E(H)$, choose a point p_{uv} that is contained in the rectangular region $r_u \cap r_v$. Further, for each edge $uv \in E(H)$, choose a simple curve $\mathbf{s_{u,v}}$ between p_u and p_{uv} that is completely contained in r_u and a simple curve $\mathbf{s_{v,u}}$ between p_v and p_{uv} that is completely contained in r_v such that for any curve in the collection $\{\mathbf{s_{u,v}}, \mathbf{s_{v,u}}\}_{uv \in E(H)}$, none of its interior points are contained in any other curve in the collection. Now the set of simple curves $\{\mathbf{s_{u,v}} \cup \mathbf{s_{v,u}}\}_{uv \in E(H)}$ corresponds to the edges of H and gives a planar embedding of H (please see Figure 2.3.2 for an example). Hence, G is a planar graph. □

2.4 Bounds on the stab number for some graph classes

In this section, we study the stab number of some subclasses of rectangle intersection graphs. We show a lower bound on $stab(G)$ for any rectangle intersection graph G, which is used to derive an asymptotically tight lower bound for the stab number of grids. We also derive upper bounds on $estab(G)$ when G is a split graph or a block graph.

2.4.1 Lower bounds

A *c-coloring* of G is a mapping $\phi\colon V(G) \to \{1, 2, \ldots, c\}$ such that $\phi(u) \neq \phi(v)$ when $uv \in E(G)$. A graph is *c-colorable* if it has a c-coloring. The *chromatic number* $\chi(G)$ of G is the minimum c such that G is c-colorable.

It is clear that given a k-stabbed rectangle intersection representation of a graph G, a set of $\omega(G)$ colours can be used to properly colour the vertices whose rectangles have a common stab (since the subgraph induced in G by these vertices is an interval graph). This means that if G is exactly stabbable, we can use two sets of $\omega(G)$ colours each to colour the vertices on alternate stab lines of a k-exactly stabbed representation of G (for some k) to obtain a proper colouring of G. Thus, if G is an exactly stabbable rectangle intersection graph, then $\chi(G) \leq 2\omega(G)$. For general rectangle intersection graphs, we can adapt the same colouring strategy to get the following observation.

Observation 2.4.1. *For a rectangle intersection graph G, we have $\chi(G) \leq stab(G) \cdot \omega(G)$, or in other words, $stab(G) \geq \frac{\chi(G)}{\omega(G)}$.*

We now strengthen the above observation and show that the $\chi(G)$ in the lower bound can be replaced by $pw(G) + 1$, where $pw(G)$ is the "pathwidth" of G. A *path decomposition* of a graph G is a collection $X_1, X_2, \ldots, X_t$ of subsets of $V(G)$, where t is some positive integer, such that for each edge $uv \in E(G)$, there exists $i \in \{1, 2, \ldots, t\}$ such that $u, v \in X_i$ and for each vertex $u \in V(G)$, if $u \in X_i \cap X_j$, where $i < j$, then $u \in X_k$ for $i \leq k \leq j$. The *width* of a path decomposition $X_1, X_2, \ldots, X_t$ of G is defined to be $\max_{1 \leq i \leq t}\{|X_i|\} - 1$. The *pathwidth* of a graph G, denoted by $pw(G)$, is the width of a path decomposition of G of minimum width.

We adapt a proof by Suderman [146] to show that if a graph G is k-SRIG then G has pathwidth at most $k \cdot \omega(G) - 1$.

Theorem 2.4.1. *Let G be a rectangle intersection graph. Then $pw(G) \leq \omega(G) \cdot stab(G) - 1$, or in other words, $stab(G) \geq \frac{pw(G)+1}{\omega(G)}$.*

Proof. Let G be a rectangle intersection graph with $stab(G) = k$. We shall show that $pw(G) \leq k \cdot \omega(G) - 1$. Let $\mathcal{R}$ be a k-stabbed rectangle intersection representation of G. Let $V(G) = \{u_1, u_2, \ldots, u_n\}$ such that $x^+_{u_1} \leq x^+_{u_2} \leq \cdots \leq x^+_{u_n}$. For $i \in \{1, 2, \ldots, n\}$, let us define the subset $X_i = \{v \in V(G) \colon x^+_{u_i} \in span(v)\}$. We claim that $X_1, X_2, \ldots, X_n$ is a path decomposition of G. To see this, note that for any edge $u_iu_j \in E(G)$, where $i < j$, $u_i, u_j \in X_i$. Also, if some vertex $v \in X_i \cap X_j$, where $i < j$, then $span(v)$ contains both $x^+_{u_i}$ and $x^+_{u_j}$, implying that it also contains $x^+_{u_k}$, for $i \leq k \leq j$. Therefore, $v \in X_k$, for $i \leq k \leq j$. To complete the proof, we only need to show that $\max_{1 \leq i \leq n}\{|X_i|\} \leq k \cdot \omega(G)$. Suppose that for some $i \in \{1, 2, \ldots, n\}$, there exists $S \subseteq X_i$ such that $|S| \geq \omega(G) + 1$ and all the vertices of S have a common stab. Since $x^+_{u_i} \in \bigcap_{u \in S} span(u)$ and the rectangles corresponding to the vertices of S all intersect a common stab line, we have that the vertices of S form a clique in G, which is a contradiction to the fact that $\omega(G)$ is the clique number of G. Therefore, for any $i \in \{1, 2, \ldots, n\}$, there exists at most $\omega(G)$ vertices in X_i that have a common stab. Since there are only k stab lines in $\mathcal{R}$, we now have that $|X_i| \leq k \cdot \omega(G)$ for each $i \in \{1, 2, \ldots, n\}$. □

The (h, w)-*grid* is the undirected graph G with $V(G) = \{(x, y) \colon x, y \in \mathbb{Z}, 1 \leq x \leq h, 1 \leq y \leq w\}$ and $E(G) = \{(u, v)(x, y) \colon |u - x| + |v - y| = 1\}$.

Corollary 2. *Let G be the (h, w)-grid. Then $\frac{1}{2}(\min\{h, w\} + 1) \leq stab(G) \leq estab(G) \leq \min\{h, w\}$.*

Proof. It is clear that $\omega(G) \leq 2$ and from a result of [72] we know that the pathwidth of the (h, w)-grid is $\min\{h, w\}$. From these facts and Theorem 2.4.1, we can infer that, $\frac{1}{2}(\min\{h, w\} + 1) \leq stab(G)$. It is easy to see that the (h, w)-grid graph has a $\min\{h, w\}$-exactly stabbed rectangle intersection representation as shown in Figure 2.4.1, and therefore

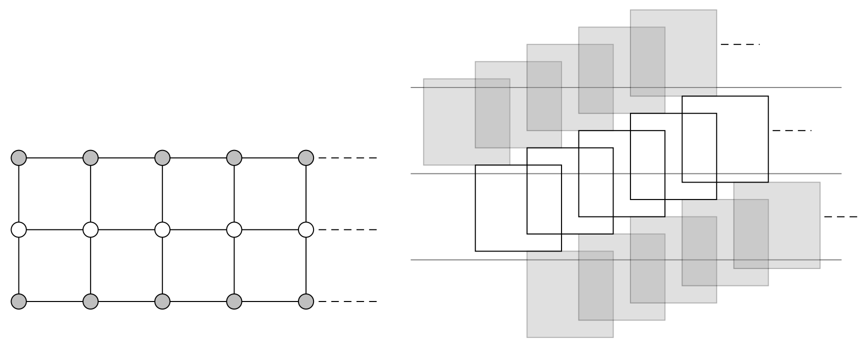


Figure 2.4.1: Illustration of $\min\{h, w\}$-exactly stabbed rectangle intersection representation of the (h, w)-grid: (a) The $(3, n)$-grid with $n \geq 3$; (b) a 3-exactly stabbed rectangle intersection representation of the $(3, n)$-grid.

$estab(G) \leq \min\{h, w\}$. The statement of the corollary now follows from the fact that $stab(G) \leq estab(G)$. □

The above corollary shows that $stab(\text{GRIDS}, n) = \Theta(\sqrt{n})$. This also shows that there are triangle-free rectangle intersection graphs on n vertices whose stab number can be $\Omega(\sqrt{n})$. Moreover, these triangle-free rectangle intersection graphs are exactly stabbable.

2.4.2 Split graphs

A split graph is a graph whose vertex set can be partitioned into a clique and an independent set. It is known that split graphs can have arbitrarily high boxicity [62]. So it is natural to ask whether the split graphs within rectangle intersection graphs are all exactly stabbable rectangle intersection graphs. We show that any split graph with boxicity at most 2 is 3-ESRIG (Theorem 2.4.2) and that there exists a split graph with boxicity at most 2 which is not 2-ESRIG (Theorem 2.4.3). From Theorem 2.3.2, it then follows that the stab number and exact stab number are equal for any split graph that has boxicity at most 2. Adiga et al.

showed that deciding whether a split graph has boxicity at most 3 is NP-complete [4]. But as far as we know, the problem of deciding whether the boxicity of a split graph is at most 2 is not known to be polynomial-time solvable or NP-complete. By our observations below, it follows that this problem is equivalent to deciding whether a given split graph is 3-ESRIG (or equivalently, 3-SRIG).

Theorem 2.4.2. *A split graph G is a rectangle intersection graph if and only if G is a 3-ESRIG.*

Proof. As G is a split graph, there exists a partition of $V(G)$ into sets C and I such that C is a clique and I is an independent set. If G is a 3-ESRIG then G is a rectangle intersection graph. Now let G be a split graph having a rectangle intersection representation $\mathcal{R}$ such that for any two vertices $u, v \in V(G)$, $\{x_u^-, x_u^+, y_u^-, y_u^+\} \cap \{x_v^-, x_v^+, y_v^-, y_v^+\} = \emptyset$ (note that such a rectangle intersection representation exists for any rectangle intersection graph). We shall assume without loss of generality that in this representation, the origin is contained in $\bigcap_{v \in C} r_v$. For every vertex $u \in I$, define the region $A_u = \bigcap_{v \in N[u]} r_v$. It is easy to see that $A_u \subseteq r_u$. It follows that for vertices $u, v \in V(G)$ such that $u \in I$ and $v \notin N[u]$, $A_u \cap r_v = \emptyset$. Also, A_u is a rectangle (by the Helly property of rectangles) with non-zero height and width. This means that we can choose a point p_u in A_u that is not on the X-axis for each vertex $u \in I$, while satisfying the additional property that no two points in $\{p_u\}_{u \in I}$ have the same x-coordinate. Consider $u \in I$. Since the degenerate rectangle given by the point p_u intersects all the rectangles in $\{r_v\}_{v \in N(u)}$, we can replace the rectangle r_u with the degenerate rectangle given by the point p_u to obtain a new rectangle intersection representation of G. Let $\mathcal{R}'$ be the rectangle intersection representation of G obtained in this fasion, i.e. $\mathcal{R}' = (\mathcal{R} \setminus \{r_u\}_{u \in I}) \cup \{p_u\}_{u \in I}$ (see Figure 2.4.2(a)).

Let I^+ (respectively I^-) be the set of vertices $\{u \in I \colon p_u$ is above (respectively, below) the X-axis $\}$. Let $y_{max} = \max\{y_v^+ \colon v \in C\}$ and

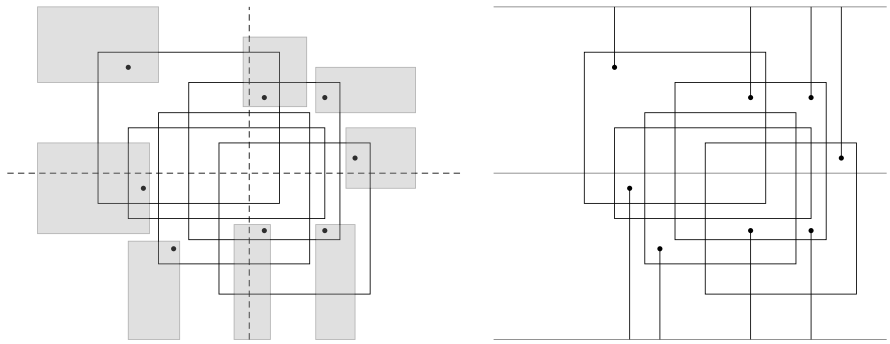
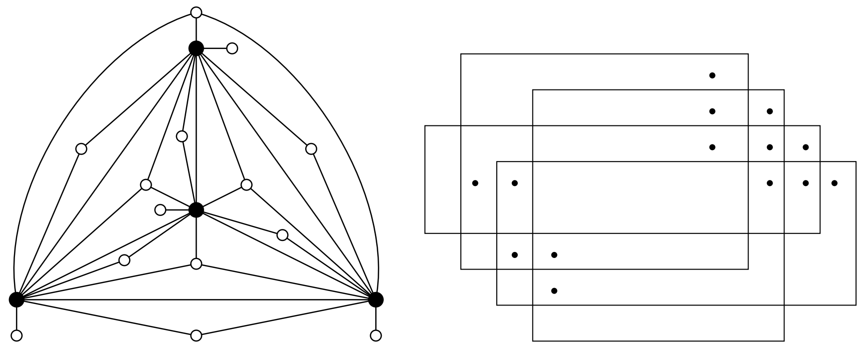


Figure 2.4.2: Representation of split graphs with boxicity at most 2. (a) The shaded rectangles represent vertices of the independent set of the split graph and the dots indicate the points p_u, for each vertex u in the independent set. (b) The 3-ESRIG representation derived from the rectangle intersection representation given in (a).

$y_{min} = \min\{y_v^- : v \in C\}$. For each vertex $u \in I^+$, we define s_u to be the degenerate rectangle given by the vertical line segment whose bottom end point is p_u and top end point has y-coordinate $y_{max}+1$. Similarly, for each vertex $u \in I^-$, we define s_u to be the degenerate rectangle given by the vertical line segment whose top end point is p_u and bottom end point has y-coordinate $y_{min} - 1$. As each rectangle in $\mathcal{R}'$ corresponding to a vertex in C contains the origin, we have that for any $u, v \in V(G)$ such that $u \in I$ and $v \in C$, the rectangle r_v intersects s_u if and only if r_v contains p_u. Therefore, the collection of rectangles given by $(\mathcal{R}' \setminus \{p_u\}_{u\in I}) \cup \{s_u\}_{u\in I}$ is a rectangle intersection representation of G. It is easy to see that this rectangle intersection representation, together with the horizontal lines $y = y_{min} - 1$, $y = 0$, and $y = y_{max} + 1$, forms a 3-ESRIG representation of G (see Figure 2.4.2(b)). □

Theorem 2.4.3. *There is a split graph G which is a rectangle intersection graph but not a 2-ESRIG.*

Proof. Let G be the split graph whose vertex set is partitioned into a clique C on four vertices and an in independent set I of 14 vertices, and



(a) (b)

Figure 2.4.3: (a) A planar split graph which is 3-ESRIG but not 2-ESRIG. The clique vertices are coloured black and the remaining vertices are independent vertices. (b) A rectangle intersection representation of the graph shown in (a). The vertices corresponding to the independent set are represented as points.

whose edges are defined as follows. Let $\mathcal{X}$ be the set of all subsets X of C with $1 \leq |X| \leq 3$. For every $X \in \mathcal{X}$, there is exactly one vertex $u_X \in I$ such that $N(u_X) = X$. See Figure 2.4.3(a) for a drawing of the graph G. Clearly, G has a rectangle intersection representation as shown in Figure 2.4.3(b).

Now assume for the sake of contradiction that G has a 2-ESRIG representation $\mathcal{R}$. We can assume that the stab lines are $y = 0$ and $y = 1$. We shall further assume that all the rectangles are contained in the strip of the plane between the two stab lines, i.e. for each $v \in V(G)$, we have $y_v^- \geq 0$ and $y_v^+ \leq 1$ (it is easy to see that every 2-ESRIG representation can be converted to such a 2-ESRIG representation by "trimming" the parts of the rectangles that lie above the top stab line and below the bottom stab line).

Observe that for each $X \in \mathcal{X}$, the rectangle r_{u_X} intersects all the rectangles in $\{r_v\}_{v \in X}$ and is disjoint from each rectangle in $\{r_v\}_{v \in C \setminus X}$. Now choose a point $p_X \in r_{u_X} \cap \bigcap_{v \in X} r_v$. Clearly, $p_X \in \bigcap_{v \in X} r_v$ and

$p_X \notin \bigcup_{v \in C \setminus X} r_v$.

Let $a, b \in C$ (not necessarily distinct) such that $x_a^- = \max\{x_v^-\}_{v \in C}$ and $x_b^+ = \min\{x_v^+\}_{v \in C}$. Let c, d be two distinct vertices in $C \setminus \{a, b\}$. By our choice of a and b, we have $[x_a^-, x_b^+] \subseteq span(c)$ and $[x_a^-, x_b^+] \subseteq span(d)$, or in other words $[x_a^-, x_b^+] \subseteq span(c) \cap span(d)$.

Claim. *The vertices c and d have a common stab.*

Suppose for the sake of contradiction that c and d do not have a common stab. Then, since $[x_a^-, x_b^+] \subseteq span(c) \cap span(d)$ and $r_c \cap r_d \neq \emptyset$, it follows that the rectangle $[x_a^-, x_b^+] \times [0, 1] \subseteq r_c \cup r_d$. We thus have $r_a \cap r_b \subseteq [x_a^-, x_b^+] \times [0, 1] \subseteq r_c \cup r_d$. But this contradicts the fact that there exists a point $p_{\{a,b\}}$ such that $p_{\{a,b\}} \in r_a \cap r_b$ and $p_{\{a,b\}} \notin r_c \cup r_d$. This proves the claim.

By the above claim, we shall assume without loss of generality that c and d are on the stab line $y = 0$ and that $y_c^+ \leq y_d^+$. This implies that $[x_a^-, x_b^+] \times [0, y_c^+] \subseteq [x_a^-, x_b^+] \times [0, y_d^+] \subseteq r_d$ (recall that $[x_a^-, x_b^+] \subseteq span(d)$). Note that $r_a \cap r_b \cap r_c \subseteq [x_a^-, x_b^+] \times [0, y_c^+]$, implying that $r_a \cap r_b \cap r_c \subseteq r_d$. But this contradicts the fact that there exists a point $p_{\{a,b,c\}}$ such that $p_{\{a,b,c\}} \in r_a \cap r_b \cap r_c$ and $p_{\{a,b,c\}} \notin r_d$. □

2.4.3 Block graphs

A graph G is a block graph if every block (i.e 2-connected component) of G is a clique. Note that all trees are block graphs. It is not hard to see that all trees, and indeed all block graphs, are rectangle intersection graphs. We show that all block graphs are exactly stabbable rectangle intersection graphs and give an upper bound of $\lceil \log m \rceil$ for the exact stab number of block graphs with m blocks, where $m \geq 2$. Note that this implies an upper bound of $\lceil \log n \rceil$ for the exact stab number of trees on n vertices. We shall show in Section 2.7 that this bound is asymptotically tight, by constructing trees whose stab number is $\Omega(\log n)$.

Let G be a block graph. Given a k-exactly stabbed rectangle intersection representation $\mathcal{R}$ of G, we say that a set of vertices $S \subseteq B$, where B is a block in G, is *accessible* if all vertices in S are on the bottom stab line of $\mathcal{R}$ and for any vertex $v \notin S$ either v is not on the bottom stab line or $x_u^- < x_v^-$ for every vertex $u \in S$.

Theorem 2.4.4. *For any block graph G with m blocks, $estab(G) \leq \max\{1, \lceil \log m \rceil\}$.*

Proof. Note that we only need the statement of the theorem to be proved for connected graphs. In fact, we shall prove the following stronger claim for connected graphs.

Claim. Let G be any connected block graph with m blocks and let $k = \max\{1, \lceil \log m \rceil\}$. Then for any block B of G, any subset S of B, any $a, b \in \mathbb{R}$ such that $a < b$, and any $h \in \mathbb{R}$ such that $0 \leq h < 1$, there is a k-exactly stabbed rectangle intersection representation $\mathcal{R}(S, a, b, h)$ of G with stab lines $y = 0$, $y = 1$, $y = 2$, ..., $y = k - 1$ such that:

- *S is accessible,*
- *for every vertex $u \in V(G)$, $span(u) \subseteq (a, b)$,*
- *for every vertex $u \in V(G)$ that is on the bottom stab line, we have $y_u^+ > h$, and*
- *for every vertex $u \in V(G)$ that is not on the bottom stab line, we have $y_u^- > h$.*

Proof. We prove the claim by induction on m. When $m \leq 2$, G is an interval graph. It is not hard to see that the statement of the claim is true in this case. From here onwards, we shall assume that $m \geq 3$, and that the statement of the claim is true when the number of blocks in the graph is lesser than m.

Let $\mathcal{H}$ be the set of components of $G - B$. It is easy to see that each graph $H \in \mathcal{H}$ is a block graph and at most one of them can have greater than $\frac{m}{2}$ blocks. We shall denote the graph in $\mathcal{H}$ that has greater than $\frac{m}{2}$ blocks, if it exists, as H^*. For a vertex $u \in B$, let $\mathcal{H}_u = \{H \in \mathcal{H} \colon N(u) \cap V(H) \neq \emptyset\}$. Note that for $u, v \in B$ such that $u \neq v$, $\mathcal{H}_u \cap \mathcal{H}_v = \emptyset$. Also, since G is connected, $\{\mathcal{H}_u\}_{u \in B}$ is a partition of $\mathcal{H}$. If H^* exists, let $u^* \in B$ be the vertex such that $H^* \in \mathcal{H}_{u^*}$.

Let $\mathcal{I}_B = \{[c_u, d_u]\}_{u \in B}$ be an interval representation of $G[B]$ (which is a complete graph) such that all endpoints of intervals are distinct, $[c_u, d_u] \subseteq (a, \frac{a+b}{2})$ for any $u \in V(G)$, and for any $u \in S$ and $v \in B \setminus S$, we have $c_u < c_v$. Let $B = \{u_1, u_2, \ldots, u_{|B|}\}$, where $c_{u_1} < c_{u_2} < \cdots < c_{u_{|B|}}$. We shall define $c_{u_{|B|+1}} = d_{u_{|B|}}$ (this shall be used later on). Choose $|B|$ real numbers $h_1, h_2, \ldots, h_{|B|}$ such that $h < h_1 < h_2 < \cdots < h_{|B|} < 1$. We define r_u for every vertex $u \in B$ other than u^* as follows: $r_u = [c_u, d_u] \times [0, h_i]$, where $i \in \{1, 2, \ldots, |B|\}$ is such that $u = u_i$. We shall show how to define r_{u^*}, in case u^* exists, later.

For each $i \in \{1, 2, \ldots, |B|\}$, let $\mathcal{H}'_i = \mathcal{H}_{u_i} \setminus \{H^*\}$, if H^* exists, and $\mathcal{H}'_i = \mathcal{H}_{u_i}$ otherwise. Let $t_i = |\mathcal{H}'_i|$. For each $i \in \{1, 2, \ldots, |B|\}$, let $\mathcal{H}'_i = \{H_{i,1}, H_{i,2}, \ldots, H_{i,|t_i|}\}$ and for each $j \in \{1, 2, \ldots, t_i\}$, let $S_{i,j} = N(u_i) \cap V(H_{i,j})$ (which is nonempty by the definition of $\mathcal{H}_{u_i}$). For each $i \in \{1, 2, \ldots, |B|\}$, choose $t_i + 1$ real numbers $c_{u_i} < q_{i,1} < q_{i,2} < \cdots < q_{i,t_i+1} < c_{u_{i+1}}$ (recall that $c_{u_{|B|+1}} = d_{u_{|B|}}$). Now consider any $i \in \{1, 2, \ldots, |B|\}$ and any $j \in \{1, 2, \ldots, t_i\}$. As the number of blocks in $H_{i,j}$ is at most $\frac{m}{2}$, we can apply the induction hypothesis on $H_{i,j}$ to conclude that there is a $\max\{1, \lceil \log m \rceil - 1\}$-exactly stabbed rectangle intersection representation $\mathcal{R}'_{i,j} = \mathcal{R}(S_{i,j}, q_{i,j}, q_{i,j+1}, 0)$ of $H_{i,j}$. Since $m \geq 3$, we know that $k \geq 2$ and that $\max\{1, \lceil \log m \rceil - 1\} = k - 1$. Thus, $\mathcal{R}'_{i,j}$ uses the stab lines $y = 0$, $y = 1$, …, $y = k - 2$. For each vertex $v \in V(H_{i,j})$, let $r'_v = [x'^-_v, x'^+_v] \times [y'^-_v, y'^+_v]$ be the rectangle corresponding to v in $\mathcal{R}'_{i,j}$. We now define r_v for each vertex $v \in V(H_{i,j})$ for all $i \in \{1, 2, \ldots, |B|\}$ and $j \in \{1, 2, \ldots, t_i\}$ as follows. If $v \in S_{i,j}$, then $r_v = [x'^-_v, x'^+_v] \times [h_i, y'^+_v + 1]$.

If $v \in V(H_{i,j}) \backslash S_{i,j}$ and v is on the bottom stab line of $\mathcal{R}'_{i,j}$, then we define $r_v = [x_v'^-, x_v'^+] \times [1, y_v'^+ + 1]$. Lastly, if $v \in V(H_{i,j}) \backslash S_{i,j}$, but v is not on the bottom stab line of $\mathcal{R}'$, then we define $r_v = [x_v'^-, x_v'^+] \times [y_v'^- + 1, y_v'^+ + 1]$.

(*) For an integer $i \in \{1, 2, \ldots, |B|\}$ and a vertex v of some $H \in \mathcal{H}'_i$, we have $[x_v^-, x_v^+] \subset [c_{u_i}, c_{u_{i+1}}]$.

(+) For an integer $i \in \{1, 2, \ldots, |B|\}$ and for any two distinct integers $j, k \in \{1, 2, \ldots, t_i\}$ let u be a vertex in $V(H_{i,j})$ and v be a vertex in $V(H_{i,k})$. Then $r_u \cap r_v = \emptyset$ (since $[x_u^-, x_u^+]$, $[x_v^-, x_v^+]$ belong respectively to the intervals $(q_{i,j}, q_{i,j+1}), (q_{i,k}, q_{i,k+1})$ which are disjoint).

(++) Let i, j be two distinct integers in $\{1, 2, \ldots, |B|\}$. Let u be a vertex in some graph in $\mathcal{H}'_i$ and v be a vertex in some graph in $\mathcal{H}'_j$. Then $r_u \cap r_v = \emptyset$ (since $[x_u^-, x_u^+]$, $[x_v^-, x_v^+]$ belong respectively to the intervals $(c_{u_i}, c_{u_{i+1}}), (c_{u_j}, c_{u_{j+1}})$ which are disjoint).

We now define a rectangle r_v for each vertex $v \in V(H^*)$ and the rectangle r_{u^*} for u^*, in case H^* exists. Let $S^* = N[u^*] \cap V(H^*)$. Since H^* contains less than m blocks, and recalling that $k = \max\{1, \lceil \log m \rceil\}$, we have by the induction hypothesis that H^* has a k-exactly stabbed rectangle intersection representation $\mathcal{R}^* = \mathcal{R}(S^*, \frac{a+b}{2}, b, h_{|B|})$ that uses the stab lines $y = 0$, $y = 1$, ..., $y = k-1$. Let the rectangle in $\mathcal{R}^*$ corresponding to a vertex $v \in V(H^*)$ be denoted by $r_v^* = [x_v^{*-}, x_v^{*+}] \times [y_v^{*-}, y_v^{*+}]$. We define $r_v = r_v^*$ for every vertex $v \in V(H^*)$. We now let $r_{u^*} = [c_{u^*}, \max\{x_v^{*-} \colon v \in S^*\}] \times [0, h_i]$, where $i \in \{1, 2, \ldots, |B|\}$ is such that $u^* = u_i$.

(+++) Let i be any integer in $\{1, 2, \ldots, |B|\}$. Let u be a vertex of some graph in $\mathcal{H}'_i$ and v be a vertex of H^*. Then $r_u \cap r_v = \emptyset$ (since $[x_u^-, x_u^+]$, $[x_v^-, x_v^+]$ belong respectively to the intervals $(a, \frac{a+b}{2}), (\frac{a+b}{2}, b)$ which are disjoint).

We now verify that $\mathcal{R} = \{r_u\}_{u \in V(G)}$ forms a $\lceil \log m \rceil$-exactly stabbed rectangle intersection representation of G that satisfies all the requirements to be $\mathcal{R}(S, a, b, h)$. For a vertex $u \in V(G)$, let $x_u^-, x_u^+, y_u^-, y_u^+$ be such that $r_u = [x_u^-, x_u^+] \times [y_u^-, y_u^+]$.

From the construction of $\mathcal{R}$, it is clear that all the vertices in B, and therefore all the vertices in S, are on the bottom stab line. It is also easy to see that the only vertices on the bottom stab line other than the vertices in B are some vertices in $V(H^*)$. For any vertex $u \in B$ and $v \in V(H^*)$, we have $x_u^- < \frac{a+b}{2} < x_v^-$. Note that for any vertex $u \in B$, we have $x_u^- = c_u$. Therefore, for vertices $u, v \in B$ such that $u \in S$ and $v \in B \setminus S$, we have $x_u^- < x_v^-$ (recall that $c_u < c_v$ in this case). From this, we can infer that S is accessible in $\mathcal{R}$.

It is clear that for each $u \in V(G)$, $r_u \subset (a, b)$. Now consider any vertex v that is on the bottom stab line in $\mathcal{R}$. As explained before, v is either in B or in $V(H^*)$. If $v \in B$, then $v = u_i$ for some $i \in \{1, 2, \ldots, |B|\}$, and $y_v^+ = h_i > h$. On the other hand, if $v \in V(H^*)$, then $r_v = r_v^*$, the rectangle corresponding to v in $\mathcal{R}^*$. Since $\mathcal{R}^* = \mathcal{R}(S^*, \frac{a+b}{2}, b, h_{|B|})$, we know that $y_v^{*+} > h_{|B|} > h$, and therefore we have $y_v^+ > h$. Therefore, for every vertex $v \in V(G)$ that is on the bottom stab line, we have $y_v^+ > h$. Now consider a vertex $v \in V(G)$ that is not on the bottom stab line in $\mathcal{R}$. It is clear that $v \notin B$. If $v \in V(H)$, where $H \neq H^*$ and $H \in \mathcal{H}_{u_i}$, for some $i \in \{1, 2, \ldots, |B|\}$, then by our construction, $y_v^- \geq h_i > h$. If $v \in V(H^*)$, then we know that since v is not on the bottom stab line of $\mathcal{R}$, it is also not on the bottom stab line of $\mathcal{R}^*$. Since $\mathcal{R}^* = (S^*, \frac{a+b}{2}, b, h_{|B|})$, this means that $y_v^{*-} > h_{|B|} > h$. As $y_v^- = y_v^{*-}$, we now have $y_v^- > h$. This shows that $\mathcal{R}$ satisfies the four conditions to be chosen as $\mathcal{R}(S, a, b, h)$.

As it can be easily verified that each rectangle in $\mathcal{R}$ is intersected by exactly one of the stab lines $y = 0$, $y = 1$, …, $y = k - 1$, it only remains to be shown that $\mathcal{R}$ is a rectangle intersection representation of G. Even though this is more or less clear from the construction, we give a proof for the sake of completeness. Consider $u, v \in V(G)$. We shall show that $uv \in E(G)$ if and only if $r_u \cap r_v \neq \emptyset$.

(i) First, let us consider the case when $u, v \in V(H^*)$. Since we have

$r_u = r^*_u$ and $r_v = r^*_v$, $r_u \cap r_v \neq \emptyset \Leftrightarrow r^*_u \cap r^*_v \neq \emptyset$. Since $\mathcal{R}^*$ is a valid representation of H^*, we have $r_u \cap r_v \neq \emptyset \Leftrightarrow uv \in E(H^*) \Leftrightarrow uv \in E(G)$.

(ii) Next, let us consider the case when $u \in B$ and $v \in V(H^*)$. If $u \neq u^*$ then $H^* \notin \mathcal{H}_u$ and thus $uv \notin E(G)$. Also, we have $[x^-_u, x^+_u] \subseteq (a, \frac{a+b}{2})$ (since $u \neq u^*$) and $[x^-_v, x^+_v] \subseteq (\frac{a+b}{2}, b)$. Hence $r_u \cap r_v = \emptyset$. Now assume that $u = u^* = u_i$ (for some $i \in \{1, 2, \ldots, |B|\}$). Recall that $S^* = N(u) \cap V(H^*)$. Suppose first that $v \in S^*$. Then $uv \in E(G)$. Now from the definition of $\mathcal{R}^*$ and $r_{u^*} = r_u$, we have that both r_v and r_u intersect the stab line $y = 0$, $x^-_v = x^{*-}_v$ and that $x^+_u = \max\{x^{*-}_w : w \in S^*\}$. Combining these, we have $x^-_v \leq x^+_u$. This gives us $x^-_u < \frac{a+b}{2} < x^-_v \leq x^+_u$, implying that $r_u \cap r_v \neq \emptyset$. Now assume that $v \notin S^*$, from which it follows that $uv \notin E(G)$. If $r_v = r^*_v$ intersects the stab line $y = 0$, then since $\mathcal{R}^* = \mathcal{R}(S^*, \frac{a+b}{2}, b, h_{|B|})$, we have that $\max\{x^-_w : w \in S^*\} < x^-_v$, implying that $x^+_u < x^-_v$ (recall that $u = u^*$). Therefore, $r_u \cap r_v = \emptyset$. The only remaining case is if r_v does not intersect the bottom stab line. Then, since $r_v = r^*_v$ and $\mathcal{R}^* = \mathcal{R}(S^*, \frac{a+b}{2}, b, h_{|B|})$, we have $y^{*-}_v > h_{|B|} \geq h_i = y^+_u$, where $i \in \{1, 2, \ldots, |B|\}$ is such that $u = u^* = u_i$. Therefore $r_u \cap r_v = \emptyset$.

(iii) Next, let u be a vertex of some graph in $\mathcal{H}'_i$ for some $i \in \{1, 2, \ldots, |B|\}$ and v be a vertex in H^*. Then clearly $uv \notin E(G)$ and by (+++) we have that $r_u \cap r_v = \emptyset$.

(iv) Next, suppose that $u, v \in B$. Note that for every vertex $u \in B \setminus \{u^*\}$, we have $x^-_u = c_u$ and $x^+_u = d_u$. Since we have $x^-_{u^*} = c_{u^*}$ and $x^+_{u^*} = \max\{x^{*-}_v : v \in S^*\} > \frac{a+b}{2} > d_{u^*}$, we can conclude that for every vertex $u \in B$, $[c_u, d_u] \subseteq [x^-_u, x^+_u]$. As $G[B]$ is a clique, we have $uv \in E(G)$. By our construction, both u and v are on the bottom stab line, and since $[c_u, d_u] \cap [c_v, d_v] \neq \emptyset$, we have $[x^-_u, x^+_u] \cap [x^-_v, x^+_v] \neq$

$\emptyset$. We thus have $r_u \cap r_v \neq \emptyset$.

(v) Next, let us consider the case when $u \in B$ and v is a vertex of some graph in $\mathcal{H}'_i$. First, let us consider the case when $u = u_i$. Let v be a vertex in $H_{i,j}$ for some $j \in \{1, 2, \ldots, t_i\}$. If $uv \in E(G)$, then $v \in S_{i,j}$ (recall that $S_{i,j} = N(u_i) \cap V(H_{i,j})$). In this case, we have by (*) that $[x_v^-, x_v^+] \subset [c_{u_i}, c_{u_{i+1}}]$ and thus $[x_v^-, x_v^+] \subset [x_u^-, x_u^+]$. Furthermore, we have by construction that $y_v^- = h_i = y_u^+$, allowing us to conclude that $r_u \cap r_v \neq \emptyset$. If $uv \notin E(G)$, then $v \notin S_{i,j}$, and therefore by construction, we know that $y_v^- \geq 1$ whereas $y_u^+ = h_i < 1$. Therefore the two rectangles r_u and r_v do not intersect. Now let us consider the case when $u \neq u_i$. In this case, we have $uv \notin E(G)$. Let $u = u_j$ and assume $j < i$. Then from our construction, we have that $y_v^- \geq h_i > h_j = y_u^+$ and therefore $r_u \cap r_v = \emptyset$. Now assume $j > i$. Then from (*), we know that $x_v^+ < c_{u_{i+1}} \leq x_u^-$ and therefore conclude that $r_u \cap r_v = \emptyset$.

(vi) Next, let i, j be two distinct integers in $\{1, 2, \ldots, |B|\}$. Let u be a vertex of some graph in $\mathcal{H}'_i$ and v be a vertex of some graph in $\mathcal{H}'_j$. Then clearly $uv \notin E(G)$ and by (++) we have that $r_u \cap r_v = \emptyset$.

(vii) Next, let i be an integer in $\{1, 2, \ldots, |B|\}$ and j, k be two distinct integers in $\{1, 2, \ldots, t_i\}$. Let u be a vertex in $H_{i,j}$ and v be a vertex of $H_{i,k}$. Then clearly $uv \notin E(G)$ and by (+) we have that $r_u \cap r_v = \emptyset$.

(viii) Finally, let i be an integer in $\{1, 2, \ldots, |B|\}$ and j be an integer in $\{1, 2, \ldots, t_i\}$. Let $u, v \in V(H_{i,j})$. Let $\{r'_w\}_{w \in V(H_{i,j})} = \mathcal{R}'_{i,j}$. Also, let $r'_w = [x'^-_w, x'^+_w] \times [y'^-_w, y'^+_w]$. Then we have $[x_u^-, x_u^+] = [x'^-_u, x'^+_u]$, $[x_v^-, x_v^+] = [x'^-_v, x'^+_v]$, $y_u^+ = y'^+_u + 1$, $y_v^+ = y'^+_v + 1$, $y_u^- \in \{1, h_i, y'^-_u + 1\}$, and $y_v^- \in \{1, h_i, y'^-_v + 1\}$. Let us assume without loss of generality that $y_u^- \leq y_v^-$. We now have $[y_u^-, y_u^+] \cap [y_v^-, y_v^+] = \emptyset \Leftrightarrow y_u^+ < y_v^- \Leftrightarrow y'^+_u + 1 < y_v^-$. Recall that $y_v^- \in \{1, h_i, y'^-_v + 1\}$. If $y'^+_u + 1 < y_v^-$ and $y_v^- \in \{1, h_i\}$, then we have $y'^+_u < 0$, which is not possible (as no stab line of $\mathcal{R}'_{i,j}$ could have intersected r'_u). We can thus continue the derivation as $y'^{+}_u + 1 < y^{-}_v \Leftrightarrow y'^{+}_u + 1 < y'^{-}_v + 1 \Leftrightarrow y'^{+}_u < y'^{-}_v \Leftrightarrow [y'^{-}_u, y'^{+}_u] \cap [y'^{-}_v, y'^{+}_v] = \emptyset$. Since we have $[x^{-}_u, x^{+}_u] = [x'^{-}_u, x'^{+}_u]$ and $[x^{-}_v, x^{+}_v] = [x'^{-}_v, x'^{+}_v]$, it is clear that $[x^{-}_u, x^{+}_u] \cap [x^{-}_v, x^{+}_v] = \emptyset \Leftrightarrow [x'^{-}_u, x'^{+}_u] \cap [x'^{-}_v, x'^{+}_v] = \emptyset$. We can thus conclude that $r_u \cap r_v = \emptyset \Leftrightarrow r'_u \cap r'_v = \emptyset$. Since $\mathcal{R}'_{i,j}$ is a valid representation of $H_{i,j}$, we have $r_u \cap r_v = \emptyset \Leftrightarrow uv \notin E(H_{i,j}) \Leftrightarrow uv \notin E(G)$.

This completes the proof. □

2.5 Asteroidal subgraphs in a graph

In this section, we present a forbidden structure for k-SRIGs and k-ESRIGs that generalizes the "asteroidal triples" of Lekkerkerker and Boland [26]. We then study the block-trees of graphs in the context of these forbidden structures, to derive some preliminary observations which shall be used in the proofs in Section 2.6. First, we give some basic definitions.

We say that two subgraphs G_1, G_2 of a graph G are *neighbour-disjoint* if for any vertex $v \in V(G_1)$, $N[v] \cap V(G_2) = \emptyset$. In other words, $V(G_1)$ and $V(G_2)$ are disjoint and there is no edge between a vertex in $V(G_1)$ and a vertex in $V(G_2)$.

Let G be any graph. Given a vertex $v \in V(G)$, we say that a path P *misses* v, if no vertex in P is a neighbour of v. Similarly, given a subgraph H of G we say that P *misses* H if P misses each vertex in $V(H)$; in other words, P misses H exactly when P and H are neighbour-disjoint.

Recall that, given a graph G, three vertices $a, b, c \in V(G)$ are said to form an *asteroidal triple*, or AT for short, in G if there exists a path between any two vertices in $\{a, b, c\}$ that misses the third. A graph is said to be *AT-free* if it contains no asteroidal triple.

Definition 2.5.1. *Three connected induced subgraphs G_1, G_2, G_3 of G that are pairwise neighbour-disjoint are said to be* asteroidal *in G if for each $i \in \{1, 2, 3\}$, for any i, j, k such that $\{i, j, k\} = \{1, 2, 3\}$, there is a path from some vertex of G_i to some vertex of G_j that misses G_k.*

Suppose G_1, G_2, G_3 are asteroidal in a graph G. Then from the above definition, they are pairwise neighbour-disjoint and each of them is connected. This implies that for any i, j, k such that $\{i, j, k\} = \{1, 2, 3\}$, and for any $u \in V(G_i)$ and any $v \in V(G_j)$, there is some path between u and v that misses G_k.

Definition 2.5.2. *Let $\mathcal{C}$ be a class of graphs and let G be any graph. Let G_1, G_2, G_3 be asteroidal in G and let $G_i \in \mathcal{C}$ for $i \in \{1, 2, 3\}$. Then we say that G_1, G_2, G_3 are* asteroidal-$\mathcal{C}$ *in G.*

Definition 2.5.3. *We say that a graph G is* asteroidal-$\mathcal{C}$-free *if there are no three subgraphs that are asteroidal-$\mathcal{C}$ in G.*

2.5.1 A forbidden structure for k-SRIGs and k-ESRIGs

We now show that no k-SRIG can contain three subgraphs that are asteroidal-(non-($k-1$)-SRIG) in it. The same technique can be used to show that a k-ESRIG cannot contain three subgraphs that are asteroidal-(non-($k-1$)-ESRIG) in it. The intuition is that if a k-SRIG G contains subgraphs G_1, G_2, G_3 which are asteroidal-(non-($k-1$)-SRIG) in G, then in any k-stabbed rectangle intersection representation of G, the rectangles corresponding to vertices in G_i, for each $i \in \{1, 2, 3\}$, together occupy all the stab lines (as each G_i is a non-($k-1$)-SRIG). Coupled with the fact that the three subgraphs are pairwise neighbour-disjoint, this enforces a kind of "left-to-right" order on the subgraphs: that is, in the k-SRIG representation, for distinct $i, j \in \{1, 2, 3\}$, the collection of rectangles corresponding to vertices of G_i can be thought of as being "to the left of" or "to the right of" the collection of rectangles corresponding to the vertices of G_j. If we take this left-to-right order of subgraphs to be G_1, G_2, G_3, then it can be shown that any path from a vertex of G_1 to a vertex of G_3 must contain a vertex whose rectangle intersects a rectangle belonging to a vertex of G_2, thus contradicting the fact that G_1, G_2, G_3 are asteroidal in G. We give the formal proof below.

Theorem 2.5.1. *k-SRIGs are asteroidal-(non-$(k-1)$-SRIG)-free.*

Proof. Assume for the sake of contradiction that G is a k-SRIG with a k-stabbed rectangle intersection representation $\mathcal{R}$ and has three connected induced non-$(k-1)$-SRIG subgraphs G_1, G_2, G_3 that are asteroidal in G. As each of G_1, G_2, G_3 are non-$(k-1)$-SRIGs, but are k-SRIGs (as they are induced subgraphs of G), for each $i \in \{1, 2, 3\}$, there exists a walk W_i in G_i such that W_i contains at least one vertex on each stab line of $\mathcal{R}$ (for example, W_i can be chosen to be any path in G_i between a vertex on the top stab line and a vertex on the bottom stab line). This further implies that for each $i \in \{1, 2, 3\}$, there exists a vertex v_i in W_i that is on the bottom stab line. As G_1, G_2, G_3 are pairwise neighbour-disjoint, we know that $span(v_1)$, $span(v_2)$, $span(v_3)$ are pairwise disjoint. Therefore we can assume without loss of generality that $span(v_1) < span(v_2) < span(v_3)$. Now consider the set of vertices $S = \{w \colon w \in N[w'] \text{ for some } w' \in W_2\}$.

Consider the region X of the plane defined by $X = \bigcup_{u \in W_2} r_u$. Since W_2 is connected and has a vertex on each stab line, X is an arc-connected region that intersects all the stab lines. Clearly, for any vertex x such that $r_x \cap X \neq \emptyset$ we can conclude that $x \in S$. Now let B be the rectangle with diagonally opposite corners (x_1, y_1) and (x_2, y_2) where $x_1 = \min\{x_v^- \colon v \in V(G)\}$, $x_2 = \max\{x_v^+ \colon v \in V(G)\}$, $y = y_1$ is the bottom stab line and $y = y_2$ is the top stab line of $\mathcal{R}$.

Claim. The rectangles $B \cap r_{v_1}$ and $B \cap r_{v_3}$ are completely contained in different arc-connected regions of $B \setminus X$.

Since v_1 and v_3 have no neighbours in W_2, and therefore are not in S, we can infer from our earlier observation that the rectangles $B \cap r_{v_1}$ and

$B \cap r_{v_3}$ are disjoint from X. This means that each of these rectangles are completely contained in some arc-connected region of $B \setminus X$. Assume for the sake of contradiction that the rectangles $B \cap r_{v_1}$ and $B \cap r_{v_3}$ are completely contained in the same connected region of $B \setminus X$. This implies that there exists a curve $\mathbf{s}$ in $B \setminus X$ that connects some point in $B \cap r_{v_1}$ that is on the bottom stab line to some point in $B \cap r_{v_3}$ that is also on the bottom stab line. Now consider the points $p, q \in X$ such that p is on the top stab line and q is a point in r_{v_2} that is on the bottom stab line. Since X is connected, there is a curve $\mathbf{s}'$ in X that connects p, q. Since $span(v_1) < span(v_2) < span(v_3)$ and $\mathbf{s}, \mathbf{s}'$ are curves that are completely contained in B, we can conclude that the curves $\mathbf{s}$ and $\mathbf{s}'$ intersect. But this is a contradiction, as $\mathbf{s}$ is a curve in $B \setminus X$ and hence cannot contain any point in $\mathbf{s}' \subseteq X$. This completes the proof of the claim.

As G_1, G_2, G_3 are asteroidal in G, there is a path P between v_1 and v_3 that misses G_2. This means that the path P does not contain any vertex from S, and therefore the rectangle corresponding to no vertex in P intersects X. Since every rectangle in the representation intersects B, this means that $\bigcup_{w \in V(P)} B \cap r_w$ is an arc-connected set in $B \setminus X$ that contains both $B \cap r_{v_1}$ and $B \cap r_{v_3}$. This is a contradiction to the above claim. □

The following theorem can be proved using the similar arguments, and hence we omit the proof.

Theorem 2.5.2. *k-ESRIGs are asteroidal-(non-$(k-1)$-ESRIG)-free.*

Remark 2.5.1. *The stab number of a rectangle intersection graph with n vertices is at most n. Therefore, rectangle intersection graphs with n vertices must be asteroidal-(non-$(n-1)$-SRIG)-free.*

2.5.2 The coloured block-tree of a graph

A *hereditary* class of graphs is a class of graphs that is closed under taking induced subgraphs. A class of graphs is said to be *closed under vertex addition* if adding a vertex (and an arbitrary set of edges incident on it) to any graph in the class results in another graph that is in the class. It can be seen that a class of graphs is closed under vertex addition if and only if its complement class (the set of graphs that are not in the class) is hereditary. Therefore, the class of non-k-SRIGs and the class of non-k-ESRIGs, for any positive integer k, are both closed under vertex addition. In this section, we study the block-tree (defined below) of an asteroidal-$\mathcal{C}$-free graph, where $\mathcal{C}$ is some graph class that is closed under vertex addition. The lemmas derived in this section will be useful in the next section.

For any graph G, let $\mathcal{B}(G)$ be the set of blocks in it and $C(G)$ the set of cut-vertices in it. The *block-tree* of G (denoted as T_G) is the graph with $V(T_G) = \mathcal{B}(G) \cup C(G)$ and $E(T_G) = \{Bc\colon B \in \mathcal{B}(G),\ c \in C(G),$ and $c \in B\}$. For any graph G, the graph T_G turns out to be a tree, justifying the name "block-tree of G" [68].

For $e = Bc \in E(T_G)$, where $c \in C(G)$ and $B \in \mathcal{B}(G)$, we denote by $T_G(e)$ the connected component of $T_G - e$ containing B. Also, let us define

$$G_e = G[\bigcup_{B \in T_G(e)} B \setminus \{c\}]$$

In other words, G_e is the component of $G - \{c\}$ that contains the vertices of B other than c. Note that G_e is a connected induced subgraph of G. The following observation is a direct consequence of the structure of the block-tree.

Observation 2.5.1. *The vertices of G other than c that belong to blocks not in $T_G(e)$ are neither in G_e nor are adjacent to any vertex in G_e.*

Let $\mathcal{C}$ be a class of graphs. Let us now colour red those edges e of T_G

such that $G_e \in \mathcal{C}$. Further, let us colour red those cut-vertices in T_G that have at least two red edges incident on them. Note that if two red edges e_1 and e_2 are incident on a cut-vertex u in T_G, then G_{e_1} and G_{e_2} are two components of $G - \{u\}$. As the final step of colouring, we colour red those block-vertices of T_G that are adjacent to at least two cut-vertices that are red. We now say that the tree T_G *is coloured with respect to* $\mathcal{C}$.

Lemma 2.5.1. *Let $\mathcal{C}$ be a class of graphs that is closed under vertex addition. Let G be any graph and let T_G be coloured with respect to $\mathcal{C}$. Then the subgraph of T_G induced by the set of red vertices is connected.*

Proof. We only need to prove that for any $u, v \in V(T_G)$ that are coloured red, any vertex $w \in V(T_G)$ that lies on the path in T_G between u and v is also red. Let P be the path between u and v in T_G. If u is a cut-vertex, then let $u' = u$ and if u is a block-vertex, then let u' be a red cut-vertex that is adjacent to u but is not on P. Similarly, if v is a cut-vertex, then we let $v' = v$ and if v is a block-vertex, we let v' be a red cut-vertex that is adjacent to v but is not on P. Clearly, the path P' in T_G between u' and v' also contains w. It can be seen that there is a red edge e_u that is incident on u' but does not belong to P' and a red edge e_v that is incident on v' but does not belong to P'. As e_u and e_v are red edges, we know that $G_{e_u}, G_{e_v} \in \mathcal{C}$. Now consider any edge e that is in P'. From the structure of the block-tree, it follows that either $V(G_{e_u}) \subseteq V(G_e)$ or $V(G_{e_v}) \subseteq V(G_e)$. (To see this, let z be the cut-vertex in e and assume that u' is closer to z than v' in T_G. Then, $T_G(e_v)$ is a subtree of $T_G(e)$. Note that z is not adjacent to any block-vertex of $T_G(e_v)$, implying that z is not contained in any block that appears as a block-vertex in $T_G(e_v)$. We now have that $V(G_{e_v}) \subseteq V(G_e)$.) Since $\mathcal{C}$ is closed under vertex addition, we now have that $G_e \in \mathcal{C}$, which implies that e is red. Therefore, every edge in P' is red. It now follows that every cut-vertex in P' other than u' and v' are incident with at least two red edges. Therefore every cut-vertex in P' is red (recall that u' and v' are red by definition). This tells us that every block-vertex in P' is adjacent to two red cut-vertices, and is therefore red. This proves that w is red. □

Lemma 2.5.2. *Let G be a graph and $\mathcal{C}$ a class of graphs closed under vertex addition. Let T_G be coloured with respect to $\mathcal{C}$ and let $\mathcal{B}$ be the set of block-vertices of T_G that have at least one red neighbour (or equivalently, the blocks of G that contain at least one cut-vertex that is red in T_G). Furthermore, assume that T_G has at least one red vertex. Let H be any component of $G - \bigcup_{B \in \mathcal{B}} B$. Then:*

(a) there exists exactly one vertex $u \in V(G) \setminus V(H)$ such that $N(u) \cap H \neq \emptyset$, and

(b) $H \notin \mathcal{C}$.

Proof. Let us mark the block-vertices in T_G corresponding to blocks of G that contain at least one vertex of H and also mark the cut-vertices in T_G corresponding to cut-vertices of G that are in H. Clearly, the block-vertices that are marked are not in $\mathcal{B}$. Since H is connected, it follows from the structure of the block-tree that the marked vertices of T_G form a subtree of T_G whose leaves are all marked block-vertices. Further, it is clear that any unmarked cut-vertex that is adjacent to a marked block-vertex belongs to some block in $\mathcal{B}$ (otherwise, that cut-vertex would have been in H and therefore marked). Now suppose there exist two distinct edges $e = uX$ and $e' = u'X'$ of T_G where X, X' are marked block-vertices and u, u' are unmarked cut-vertices. Let B, B' be the blocks in $\mathcal{B}$ that contain u, u' respectively. As $B, B' \in \mathcal{B}$, there exist red cut-vertices v, v' adjacent to B, B' respectively where $u \neq v$ and $u' \neq v'$. From Lemma 2.5.1, we know that the red vertices in T_G induce a connected subtree of T_G. Therefore, every vertex in the path in T_G between v and v' has to be red. This implies that u is red, which further implies that $X \in \mathcal{B}$. But this contradicts the fact that X is a marked block-vertex.

We can therefore conclude that there exist at most one marked block-vertex X that has an unmarked neighbour in T_G. Since T_G contains at least one marked vertex and at least one unmarked vertex (as $V(H) \neq \emptyset$ and $\mathcal{B} \neq \emptyset$), we have that there is exactly one marked block-vertex X such that it has an unmarked neighbour u in T_G. It now follows from the structure of the block-tree that $H = G_{uX}$. This implies that no vertex in H can have a neighbour in $V(G) \setminus V(H)$ other than u. This proves (a).

We shall now prove (b). Suppose for the sake of contradiction that $H \in \mathcal{C}$, or in other words, $G_{uX} \in \mathcal{C}$. So, the edge uX is red in T_G.

Claim. *The cut-vertex u of T_G is red.*

As observed earlier, u is in some block that is in $\mathcal{B}$. Let $B \in \mathcal{B}$ be a block containing u. So uB is an edge of T_G. Since $B \in \mathcal{B}$, there must be some red cut-vertex u' in T_G that is adjacent to B. Clearly, $u' \neq u$, as otherwise, X would have been adjacent to a red cut-vertex, and hence it would have been in $\mathcal{B}$. But this cannot happen as X contains vertices from H. Since u' is a red cut-vertex, it has at least two red edges incident on it and therefore there is a red edge e incident on u' that is different from $u'B$. From the definition of red edges, we have that $G_e \in \mathcal{C}$. It follows from the structure of the block-tree that G_e is an induced subgraph of G_{uB}. As $\mathcal{C}$ is closed under vertex addition, we have that $G_{uB} \in \mathcal{C}$, implying that the edge uB is red in T_G. We now have two red edges, uX and uB, incident on u, which means that u is a red cut-vertex of T_G.

From the above claim, it follows that X is a block-vertex of T_G that is incident to a red cut-vertex u, and hence it is in $\mathcal{B}$. But this is a contradiction as B contains vertices of H. This proves (b). $\square$

Lemma 2.5.3. *Let $\mathcal{C}$ be a class of graphs that is closed under vertex addition. Let G be an asteroidal-$\mathcal{C}$-free graph and let T_G be coloured with respect to $\mathcal{C}$. Then the subgraph T_r of T_G induced by the set of red vertices is either empty or is a path.*

Proof. If there are no red vertices in T_G, then there is nothing to prove. So let us suppose that T_r is not empty. From Lemma 2.5.1, it follows that T_r is connected. It only remains to be shown that every vertex has degree at most two in T_r. Suppose for the sake of contradiction that u is a red vertex that has three red neighbours u_1, u_2, u_3.

Let us first consider the case when u is a block-vertex. Then, clearly u_1, u_2, u_3 are all cut-vertices. Since each u_i, for $i \in \{1, 2, 3\}$ is red, we know that there are two red edges incident on each of them. This means that for each $i \in \{1, 2, 3\}$ there is a red edge e_i different from uu_i that is incident on u_i. It is clear from Observation 2.5.1 that $G_{e_1}, G_{e_2}, G_{e_3}$ are pairwise neighbour-disjoint connected induced subgraphs of G. Because e_1, e_2, e_3 are red, we know that $G_{e_1}, G_{e_2}, G_{e_3} \in \mathcal{C}$. For each $i \in \{1, 2, 3\}$, let v_i be a neighbour of u_i in G_{e_i}. Let the block-vertex u in T_G correspond to a block B in G. From the definition of the block-tree, we know that $u_1, u_2, u_3 \in B$. Since B is a 2-connected subgraph of G, for any i, j, k such that $\{i, j, k\} = \{1, 2, 3\}$, there exists a path P_{ij} in B between u_i and u_j that does not contain u_k. Let $P'_{ij} = P_{ij} \cup \{u_i v_i, u_j v_j\}$. From Observation 2.5.1, it follows that P'_{ij} misses G_{e_k}. This means that $G_{e_1}, G_{e_2}, G_{e_3}$ are asteroidal-$\mathcal{C}$ in G, contradicting the fact that G is asteroidal-$\mathcal{C}$-free.

Next, let us consider the case when u is a cut-vertex. Then, u_1, u_2, u_3 are block-vertices that are coloured red. Since each of them have to be adjacent to at least two red cut-vertices, we know that for each $i \in \{1, 2, 3\}$, there is a red cut-vertex u'_i different from u that is adjacent to u_i. Then again, as for each $i \in \{1, 2, 3\}$, u'_i is red, we can infer that there is a red edge e_i different from $u'_i u_i$ that is incident on u'_i. As before, $G_{e_1}, G_{e_2}, G_{e_3}$ form neighbour-disjoint connected induced subgraphs of G that all belong to $\mathcal{C}$. For each $i \in \{1, 2, 3\}$, let v_i be a neighbour of u'_i in G_{e_i}. It is now clear from the structure of the block-tree that for any i, j, k such that $\{i, j, k\} = \{1, 2, 3\}$, there is a path P_{ij} in G between u'_i and u'_j that does not contain u'_k. We can now infer using Observation 2.5.1 that the path $P'_{ij} = P_{ij} \cup \{v_i u'_i, v_j u'_j\}$ misses G_{e_k}. So we again have that $G_{e_1}, G_{e_2}, G_{e_3}$ are asteroidal-$\mathcal{C}$ in G, contradicting the fact that G is asteroidal-$\mathcal{C}$-free. □

Lemma 2.5.4. *Let $\mathcal{C}$ be a class of graphs that is closed under vertex addition. Let G be a graph and let T_G be coloured with respect to $\mathcal{C}$. If there are no red vertices in T_G, then there exists a block B in G such that no component of $G - B$ is in $\mathcal{C}$.*

Proof. Note that if there exists a cut-vertex u in G such that each component of $G - \{u\}$ is not in $\mathcal{C}$, then clearly, removal of any block that contains u from G will result in a graph whose components are not in $\mathcal{C}$ (recall that $\mathcal{C}$ is closed under vertex addition). Therefore, we shall assume that for any cut-vertex u of G, there is some component of $G - \{u\}$ that is in $\mathcal{C}$. Since $\{G_e : e \text{ incident on } u\}$ are the components of $G - \{u\}$, this implies that in T_G, every cut-vertex has at least one edge e incident on it such that $G_e \in \mathcal{C}$. In other words, every cut-vertex of T_G has at least one red edge incident on it. Since T_G contains no red vertices, we can now conclude that every cut-vertex in T_G has exactly one red edge incident on it.

For a cut-vertex u in G, let us define $f(u)$ to be the only red edge incident on u in T_G. Let v be the cut-vertex in G that minimizes $|V(G_{f(v)})|$. Let $f(v) = vB$, where B is a block-vertex of T_G. Recall that in T_G, every edge incident on v other than vB is a non-red edge. In other words, none of the components of $G - v$ other than $G_{f(v)}$ belong to $\mathcal{C}$. We now claim that every edge in T_G incident on B is red. Suppose that there is a non-red edge wB in T_G. Since w is a cut-vertex, there is a red edge $f(w)$ incident on w. Since wB is non-red, $f(w)$ is different from wB. From the structure of the block-tree, it is evident that $V(G_{f(w)}) \subset V(G_{f(v)})$ $(w \in V(G_{f(v)} \setminus V(G_{f(w)}))$. But this contradicts our choice of v as we now have $(|V(G_{f(w)})| < |V(G_{f(v)})|$. Therefore, every edge that is incident on B in T_G is red.

For any block-vertex X in T_G, we shall define $F_X = \{wY : wX \in E(T_G)$

and $X \neq Y\}$. In other words, F_X consists of exactly those edges of T_G that are not incident on X but are incident on some cut-vertex adjacent to X. Note that $\{G_e : e \in F_X\}$ are exactly the components of $G - X$. Since for the block-vertex B under consideration, we know that every edge incident on it is red, we can infer that every edge in F_B is non-red (as every cut-vertex has exactly one red edge incident on it). This means that each of $\{G_e : e \in F_B\}$ is a graph that is not in $\mathcal{C}$; in other words, no component of $G - B$ belongs to $\mathcal{C}$. We have thus found the required block. □

2.6 Trees and block graphs

A question asked in Babu et al. [12] is whether it can be determined in polynomial-time if an input tree has a rectangle intersection representation in which each rectangle is a square of unit height and width. Instead of restricting the rectangles to be unit squares, we study a different restriction. In particular, we ask if, given a tree and an integer k, it can be determined in polynomial-time whether the tree has a k-SRIG or k-ESRIG representation. We show that the problem is polynomial-time solvable if $k \leq 3$. In fact, we show that we can determine in polynomial-time if the input graph G is 2-ESRIG (equivalently 2-SRIG, by Theorem 2.3.2) if G is guaranteed to be a block graph. We also show that it can be determined in polynomial-time if an input tree is 3-ESRIG (equivalently 3-SRIG, by Theorem 2.3.2). Our algorithms depend on a forbidden structure characterization for block graphs that are 2-ESRIG and trees that are 3-ESRIG. In fact, in both cases, the algorithm is a search for the presence of these forbidden structures in the input graph, and therefore it is a "certifying algorithm", in the sense that the algorithm outputs a representation whenever the answer is "Yes" and a forbidden structure in the graph whenever the answer is "No".

The forbidden structure characterizations of block graphs that are 2-

ESRIG and trees that are 3-ESRIG are obtained as follows. In the previous section, we showed that a necessary condition for a graph to be a 2-ESRIG is that it has to be asteroidal-(non-interval)-free. We show in this section that for block graphs, this necessary condition is also sufficient. We later on show that for trees that are 3-ESRIG, the necessary condition of being asteroidal-(non-2-ESRIG)-free is again a sufficient condition. First, we need the following lemma.

Lemma 2.6.1. *Let $\mathcal{C}$ be a class of graphs that is closed under vertex addition. Let G be a block graph that is asteroidal-$\mathcal{C}$-free and let T_G be coloured with respect to $\mathcal{C}$. Then there exists a set $S \subseteq V(G)$ such that $G[S]$ is an interval graph and no component of $G - S$ is in $\mathcal{C}$.*

Proof. When T_G contains at least one red vertex, let $\mathcal{B}$ be the set of block-vertices of T_G that have at least one red neighbour. If T_G contains no red vertices, then by Lemma 2.5.4, there is a block B in G whose removal gives us components, none of which are in $\mathcal{C}$. In this case, let $\mathcal{B} = \{B\}$. We shall let S be the set of vertices which are contained in some block in $\mathcal{B}$, or in other words, $S = \bigcup_{B \in \mathcal{B}} B$. By the above observation and Lemma 2.5.2, we can assume from here onwards that no component of $G - S$ is in $\mathcal{C}$. If there are no red vertices in T_G, then $G[S]$ is a complete graph, and therefore an interval graph. To complete the proof, we only need to show that if T_G contains at least one red vertex, then $G[S]$ is an interval graph.

Suppose that T_G contains at least one red vertex. Then from Lemma 2.5.3, we know that the red vertices in T_G form a path. Since block graphs are chordal, by Theorem 1.1.1, we need to only show that $G[S]$ is AT-free in order to prove that $G[S]$ is an interval graph. Suppose for the sake of contradiction that there exists an asteroidal triple $\{a, b, c\} \subseteq S$ in $G[S]$. Since $\{a, b, c\}$ has to be an independent set in G, we know that there is no block that contains any two of them. We shall say that a cut-vertex in G is red if that cut-vertex is coloured red in

T_G. Note that from the definition of S, every vertex in S is adjacent to at least one red cut-vertex (since each vertex of S is in some block that also contains a cut-vertex that is coloured red in T_G, and each block is a complete graph). Let a', b', c' denote red cut-vertices that are adjacent to a, b, c respectively. Suppose that $a' = b' = x$. Then, it is clear from the structure of the block-tree that either every path between a and c contains x or every path between b and c contains x. But this contradicts the fact that a, b, c form an AT in $G[S]$, since x is a neighbour of both a and b. We can therefore assume that a', b', c' are distinct red cut-vertices. Since the red vertices form a path in T_G, the vertices a', b', c' must lie on a path in T_G. Let us assume without loss of generality that b' lies on the path in T_G between a' and c'. This means that every path between a' and c' in $G[S]$ contains b'. We now claim that every path in $G[S]$ between a and c goes through b'. Suppose for the sake of contradiction that there exists a path P between a and c in $G[S]$ that does not contain b'. Then the path $a'a \cup P \cup cc'$ is a path between a' and c' in $G[S]$ that does not contain b', contradicting the fact that every path in $G[S]$ between a' and c' contains b'. So, we have that every path between a and c in $G[S]$ contains b', which is a neighbour of b. This contradicts the fact that a, b, c forms an AT in $G[S]$. $\square$

Theorem 2.6.1. *A block graph G is 2-ESRIG if and only if G is asteroidal-(non-interval)-free.*

Proof. Let G be a block graph. We know by Theorem 2.5.2 that if G is a 2-ESRIG then G is asteroidal-(non-interval)-free. Now we prove that if G is asteroidal-(non-interval)-free then G is a 2-ESRIG.

By letting $\mathcal{C}$ be the class of non-interval graphs, we have by Lemma 2.6.1 that there exists a set $S \subseteq V(G)$ such that $G[S]$ is an interval graph and each component of $G - S$ is also an interval graph.

Let $\mathcal{R} = \{[c_u, d_u]\}_{u \in S}$ be an interval representation of $G[S]$ such that all endpoints of intervals are distinct. Let $\epsilon \in \mathbb{R}^+$ be such that $\epsilon <$

$\min\{|c_u - c_v|\colon u, v \in S, u \neq v\}$. Also, let $L, R \in \mathbb{R}$ such that $L < \min_{u \in S} c_u$ and $R > \max_{u \in S} d_u$. For each vertex $u \in S$, define $t_u = \frac{c_u - L}{R - L}$. Let $\mathcal{H}$ be the set of components of $G - S$. For a vertex $u \in S$, let $\mathcal{H}_u = \{H \in \mathcal{H}\colon N(u) \cap H \neq \emptyset\}$. From Lemma 2.5.2(a), it is clear that for each component $H \in \mathcal{H}$, there is a exactly one vertex in S that has neighbours in H. Therefore, it follows that $\{\mathcal{H}_u\}_{u \in S}$ is a partition of $\mathcal{H}$ (recall that G is connected). Since each component of $\mathcal{H}$ is an interval graph, and because disjoint unions of interval graphs are again interval graphs, we know that for $u \in S$, the graph I_u formed by the disjoint union of the components in $\mathcal{H}_u$ is an interval graph. It is easy to see that $\{I_u\}_{u \in S}$ is a collection of neighbour-disjoint interval graphs. For each $u \in S$, let $\mathcal{R}_u$ be an interval representation $\{[c'_v, d'_v]\}_{v \in V(I_u)}$ for the interval graph I_u such that every interval in it is contained in the interval $[c_u, c_u + \epsilon]$. Note that for distinct $a, b \in S$, no interval of $\mathcal{R}_a$ intersects with any interval of $\mathcal{R}_b$. Also let $b'_v = 1$ if $v \notin N(u)$ and $b'_v = t_u$ if $v \in N(u)$. From here onwards, we shall assume that for every vertex $v \in V(G) \setminus S$, the interval $[c'_v, d'_v]$ and the value b'_v are defined.

We shall now define a rectangle $r_u = [x_u^-, x_u^+] \times [y_u^-, y_u^+]$ for each vertex $u \in V(G)$. For a vertex $u \in S$, we let $x_u^- = c_u$, $x_u^+ = d_u$, $y_u^- = 0$ and $y_u^+ = t_u$. For a vertex $u \in V(G) \setminus S$, we let $x_u^- = c'_u$, $x_u^+ = d'_u$, $y_u^- = b'_u$ and $y_u^+ = 1$. We leave it to the reader to verify that the rectangles $\{r_u\}_{u \in V(G)}$ form a 2-exactly stabbed rectangle intersection representation of G. □

Remarks. Let $\mathcal{C}$ be the class of non-interval graphs and G be a block graph with n vertices and m edges. Since checking whether G is in $\mathcal{C}$ or not is possible in $O(n+m)$ time [60], we can infer that coloring the edges of T_G with respect to $\mathcal{C}$ is possible in $O(n^2+nm)$ time. The construction procedure described in the above proof can also be performed in $O(n^2 + nm)$ time, thus giving a polynomial-time algorithm to recognize block graphs that are 2-ESRIG.

Theorem 2.6.2. *A tree G is 3-ESRIG if and only if G is asteroidal-(non-2-ESRIG)-free.*

Proof. Let G be a tree. We know by Theorem 2.5.2 that if G is a 3-ESRIG then G is asteroidal-(non-2-ESRIG)-free. Now we prove that if G is asteroidal-(non-2-ESRIG)-free then G is a 3-ESRIG.

By letting $\mathcal{C}$ be the class of non-2-ESRIGs, we have by Lemma 2.6.1 that there exists a set $S \subseteq V(G)$ such that $G[S]$ is an interval graph and each component of $G - S$ is a 2-ESRIG.

Let $\mathcal{R} = \{[c_u, d_u]\}_{u \in S}$ be an interval representation of $G[S]$ such that all endpoints of intervals are distinct. Let $\epsilon \in \mathbb{R}^+$ be such that $\epsilon < \min\{|c_u - c_v| \colon u, v \in S, u \neq v\}$. Also, let $L, R \in \mathbb{R}$ such that $L < \min_{u \in S} c_u$ and $R > \max_{u \in S} d_u$. For each vertex $u \in S$, define $t_u = \frac{c_u - L}{R - L}$. Let $\mathcal{H}$ be the set of components of $G - S$. For a vertex $u \in S$, let $\mathcal{H}_u = \{H \in \mathcal{H} \colon N(u) \cap H \neq \emptyset\}$. From Lemma 2.5.2(a), it is clear that for each component $H \in \mathcal{H}$, there is exactly one vertex in S that has neighbours in H. Therefore, it follows that $\{\mathcal{H}_u\}_{u \in S}$ is a partition of $\mathcal{H}$ (recall that G is connected). Now let H be a component of $\mathcal{H}_u$. Since G is a tree, there is exactly one vertex w of H which is adjacent to u in G. It is easy to see that there is a 2-exactly stabbed rectangle intersection representation of H such that w is on the bottom stab line (take any 2-exactly stabbed rectangle intersection representation of H, and if the rectangle corresponding to w does not intersect the bottom stab line, then reflect the whole representation about the X-axis).

Since each component of $\mathcal{H}$ is a 2-ESRIG, and because disjoint unions of 2-ESRIGs are again 2-ESRIG, we know that for $u \in S$, the graph I_u formed by the disjoint union of the components in $\mathcal{H}_u$ is a 2-ESRIG. Let $\mathcal{R}_u = \{r'_v\}_{v \in I_u}$ be a 2-exactly stabbed rectangle intersection representation of I_u with the stab lines $y = 1$ and $y = 2$ such that for any vertex v of I_u, $span(v) \subset [c_u, c_u + \epsilon]$, and for each vertex $w \in N(u) \cap V(I_u)$ the rectangle r'_w intersects the stab line $y = 1$. Let I_u^1 be the subgraph induced in I_u by the vertices that are on the stab line $y = 1$ in $\mathcal{R}_u$. Similarly, I_u^2 be the subgraph induced in I_u by the vertices that are on the stab line $y = 2$ in $\mathcal{R}_u$. For any vertex $v \in I_u$, let c'_v, d'_v, t'_v, b'_v be such that $r'_v = [c'_v, d'_v] \times [b'_v, t'_v]$.

We shall now define a rectangle r_u for each vertex $u \in V(G)$ as follows. For a vertex $u \in S$, we let $r_u = [c_u, d_u] \times [0, t_u]$. Consider a vertex $v \in V(G) \backslash S$. Let u be the vertex in S such that $v \in V(I_u)$. If $v \in V(I_u^2)$, then we let $r_v = r'_v$. If $v \in V(I_u^1)$ and $v \notin N(u)$, then we let $r_v = [c'_v, d'_v] \times [1, t'_v]$. If $v \in V(I_u^1)$ and $v \in N(u)$, then we let $r_v = [c'_v, d'_v] \times [t_u, t'_v]$. We leave it to the reader to verify that the rectangles $\{r_u\}_{u \in V(G)}$ form a 3-exactly stabbed rectangle intersection representation of G. □

Remarks. Let $\mathcal{C}$ be the class of non-2-ESRIG graphs and T be a tree with n vertices. Since checking whether T is in $\mathcal{C}$ or not is possible in $O(n^2)$ time, we can infer that coloring the edges of block-tree of T with respect to $\mathcal{C}$ is possible in $O(n^3)$ time. The construction procedure described in the above proof can also be performed in $O(n^3)$ time, thus giving a polynomial-time algorithm to recognize trees that are 3-ESRIG.

2.7 Constructing trees with high stab number

For a rooted tree T, let *root*(T) be the root vertex of T. The following observation is easy to see.

Observation 2.7.1. *Let T be a tree and T' be a subtree of T such that $T - V(T')$ has only one component.*

(i) For any edge $e \in E(T')$, at least one component of $T - e$ is a proper subtree of T'.

(ii) For any vertex $v \in V(T')$, all but one component of $T - \{v\}$ are proper subtrees of T'.

First we describe a recursive procedure to construct a rooted tree G_l for all $l \geq 1$. For $l = 1$, let G_1 be the rooted tree containing only one vertex.

For any integer l greater than 1, we construct G_l as follows. Let T_1, T_2 and T_3 be three rooted trees each isomorphic to G_{l-1}. Take a $K_{1,3}$ with vertex set $\{u, u_1, u_2, u_3\}$, where u_1, u_2, u_3 are the pendant vertices, and construct G_l by adding edges between u_i and $root(T_i)$ for each $i \in \{1, 2, 3\}$. Also let $root(G_l) = u$. For any rooted tree T with root r, we can define the "ancestor" relation on $V(T)$ in the usual way: i.e., for $u, v \in V(T)$, u is an *ancestor* of v if and only if the path in T between r and v contains u. We prove the following lemma.

Lemma 2.7.1.

(i) *For $l > 1$, G_l is not $(l-1)$-SRIG.*

(ii) *For $l \geq 1$, there is an l-exactly stabbed rectangle intersection representation $\mathcal{R}$ of G_l such that for $v, w \in V(G_l)$, $span(v) \subseteq span(w)$ if w is an ancestor of v and the vertices on the top stab line of $\mathcal{R}$ are exactly the vertices in $N[root(G_l)]$.*

(iii) *Let T and T' be two trees each isomorphic to G_l, for some $l \geq 1$. Let F_l be the tree obtained by taking a new vertex u and joining it to the root vertices of T, T' using paths of length two.*

  (a) *For $l \geq 1$, there is an l-exactly stabbed rectangle intersection representation $\mathcal{R}'$ of F_l such that for $v, w \in V(F_l)$, $span(v) \subseteq span(w)$ if w is an ancestor of v in T or T', and all vertices in the path between $root(T)$ and $root(T')$ are on the top stab line of $\mathcal{R}'$.*

  (b) *For $l \geq 2$, there is an l-exactly stabbed rectangle intersection representation $\mathcal{R}''$ of F_l such that for $v, w \in V(F_l)$, $span(v) \subseteq span(w)$ if w is an ancestor of v in T or T', and only the vertices in $N[root(T)] \cup N[root(T')]$ are on the top stab line of $\mathcal{R}''$.*

(iv) For $l \geq 2$, there are no two vertex-disjoint subtrees in G_l such that they are both non-$(l-1)$-ESRIG.

(v) For $l \geq 1$, $estab(G_l) = stab(G_l) = \log_3(n+2)$, where $n = |V(G_l)|$.

Proof. We prove each statement separately by induction on l. When $l = 1$, G_l consists of a single vertex and therefore all the statements are true. Now we assume that the above statements are true for all integers less than l.

Recall that G_l is obtained by taking three rooted trees T_1, T_2, T_3, each isomorphic to G_{l-1}, and then making each root adjacent to a unique pendant vertex of a $K_{1,3}$. Let u be the vertex of degree 3 and u_1, u_2, u_3 be the pendant vertices of the $K_{1,3}$. Also recall that $root(G_l) = u$.

To prove (i), note that as T_i is isomorphic to G_{l-1} for each $i \in \{1, 2, 3\}$, we have by our induction hypothesis that T_i is not $(l-2)$-SRIG. Therefore, T_1, T_2, T_3 are asteroidal-(non-$(l-2)$-SRIG) in G_l. Using Theorem 2.5.1, we can conclude that G_l is not $(l-1)$-SRIG.

To prove (ii), note that by our induction hypothesis, for each $i \in \{1, 2, 3\}$, T_i has an $(l-1)$-exactly stabbed rectangle intersection representation $\mathcal{R}_i$ such that for $v, w \in V(T_i)$, $span(v) \subseteq span(w)$ if w is an ancestor of v and only the vertices in $N[root(T_i)]$ are on the top stab line of $\mathcal{R}_i$. Since T_1, T_2, T_3 are vertex disjoint, it is easy to see that there is an $(l-1)$-exactly stabbed rectangle intersection representation $\mathcal{R}$ of the subgraph induced in G_l by $\cup_{i=1}^{3} V(T_i)$ such that only the vertices in $\cup_{i=1}^{3} N[root(T_i)]$ are on the top stab line of $\mathcal{R}$: we can just place $\mathcal{R}_1$, $\mathcal{R}_2$ and $\mathcal{R}_3$ side by side as shown in Figure 2.7.1(a). Now by introducing a new stab line ℓ above the top stab line of $\mathcal{R}$ and new rectangles corresponding to the vertices in $N[root(G_l)] = \{u, u_1, u_2, u_3\}$ into the representation such that they all intersect ℓ, and for each $i \in \{1, 2, 3\}$, the rectangle corresponding to u_i intersects the rectangle corresponding to $root(T_i)$ as shown in Figure 2.7.1(a), we can get the desired l-exactly

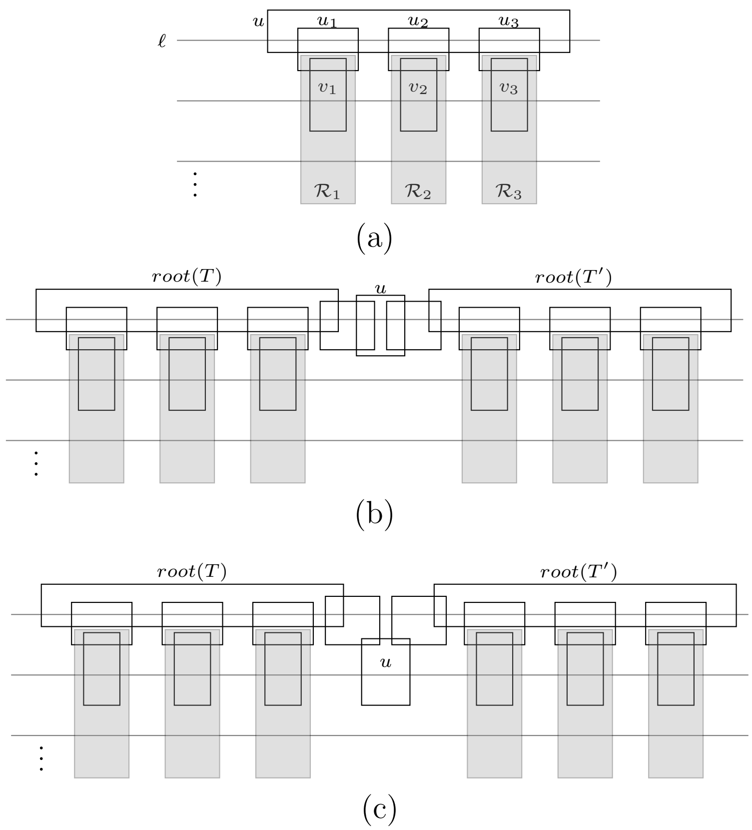


Figure 2.7.1: Construction of G_l and F_l. The shaded region denotes a collection of rectangles. In (a), for $i \in \{1, 2, 3\}$, v_i is the vertex $root(T_i)$. Figures (b) and (c) show different l-exactly stabbed rectangle intersection representations of F_l as described in Lemma 2.7.1(iii)(a) and Lemma 2.7.1(iii)(b).

stabbed rectangle intersection representation of G_l.

Now we prove (iii)(a) and (iii)(b). Since T and T' are both isomorphic to G_l and vertex disjoint, we can infer using (ii) that there is an l-exactly stabbed rectangle intersection representation $\mathcal{R}$ of $F_l[V(T) \cup V(T')]$ such that for $v, w \in V(F_l)$, $span(v) \subseteq span(w)$ if w is an ancestor of v in T or T', and only the vertices in $N[root(T)] \cup N[root(T')]$ are on the top stab line of $\mathcal{R}$ (we can obtain $\mathcal{R}$ by placing the representations of T and T' as given by (ii) side by side as shown in Figure 2.7.1(b)). Let P be the path that joins $root(T)$ and $root(T')$ in F_l. As shown in Figure 2.7.1(b), we can represent P such that all the vertices in P are on the top stab line of $\mathcal{R}$. This proves (iii)(a). Similarly, if $l \geq 2$, then as shown in Figure 2.7.1(c), we can represent P such that only the vertices in $N[root(T)] \cup N[root(T')]$ are on the top stab line of $\mathcal{R}$. This proves (iii)(b).

Now we prove (iv). Assume for the sake of contradiction that X_1, X_2 are two vertex-disjoint subtrees in G_l such that they are both non-$(l-1)$-ESRIG. Since G_l is connected, there exists an edge e in G_l such that if X'_1 and X'_2 are the two components in $G_l - e$, then for each $i \in \{1, 2\}$, X_i is a subtree of X'_i. This implies that both X'_1 and X'_2 are non-$(l-1)$-ESRIG. Suppose that $e \in E(T_i)$ for some $i \in \{1, 2, 3\}$. Note that $G_l - V(T_i)$ has only one component. Therefore, using Observation 2.7.1(i) we can infer that there exists $X \in \{X'_1, X'_2\}$ such that X is a proper subtree of T_i. But as T_i, being isomorphic to G_{l-1}, is $(l-1)$-ESRIG by (ii), this implies that X is $(l-1)$-ESRIG. This contradicts the fact that both X'_1 and X'_2 are non-$(l-1)$-ESRIG. Therefore, we can assume without loss of generality that e is either uu_1 or the edge between u_1 and $root(T_1)$. If e is the edge between u_1 and $root(T_1)$, then one of the components of $T-e$ is T_1, which is $(l-1)$-ESRIG by (ii), contradicting the fact that both components of $T-e$ are non-$(l-1)$-ESRIG. If e is the edge uu_1, then one of the components of $T-e$ is isomorphic to F_{l-1}, and therefore by (iii), is $(l-1)$-ESRIG. This again contradicts the fact that both components of $T-e$ are non-$(l-1)$-ESRIG.

To prove (v), we can solve the recurrence $|V(G_l)| = 3|V(G_{l-1})| + 4$ to obtain $n = |V(G_l)| = 3^l - 2$. Now, using (i) and (ii), we can conclude that $estab(G_l) = stab(G_l) = \log_3(n+2)$. $\square$

From Theorem 2.4.4, we have that for any tree T on n vertices with $n \geq 3$, $estab(T) \leq \lceil \log(n-1) \rceil$. Also, using Theorem 2.4.4 and Lemma 2.7.1(v), we have the following corollary.

Corollary 3. *$estab(\textsc{Trees}, n) = \Theta(\log n)$, $stab(\textsc{Trees}, n) = \Theta(\log n)$, $estab(\textsc{Block Graphs}, n) = \Theta(\log n)$, and $stab(\textsc{Block Graphs}, n) = \Theta(\log n)$.*

Although the stab number and exact stab number were equal for the trees that we constructed in this section, we shall show in Theorem 2.9.1 there are trees for which these parameters differ. The graph G_l and the observations in Lemma 2.7.1 will be used frequently in the remainder of the chapter.

2.8 Absence of asteroidal subgraphs is not sufficient

We showed in Theorem 2.5.1 that being asteroidal-(non-$(k-1)$-SRIG)-free is a necessary condition for a graph to be k-SRIG. Theorem 2.6.1 showed that this necessary condition is also sufficient for block graphs when $k \leq 2$ and Theorem 2.6.2 demonstrated that this necessary condition is sufficient for trees when $k \leq 3$. In this section, we shall show that this necessary condition is not sufficient for block graphs for any $k \geq 3$ and it is not sufficient for trees for any $k \geq 4$. In particular, we shall prove the following two theorems.

Theorem 2.8.1. *There exists a block graph that is asteroidal-(non-2-SRIG)-free, but is not 3-SRIG.*

Note that by Theorem 2.3.2, the above theorem also means that there exists a block graph that is asteroidal-(non-2-ESRIG)-free, but is not 3-ESRIG.

Theorem 2.8.2. *For each integer $k \geq 4$, there exists a tree T that is asteroidal-(non-$(k-1)$-ESRIG)-free, but is not k-SRIG.*

It is easy to see that Theorem 2.8.2 directly gives the following two corollaries, which tell us that the necessary conditions derived in Theorem 2.5.1 and Theorem 2.5.2 for a tree to be a k-SRIG and k-ESRIG respectively, are not sufficient for any $k \geq 4$.

Corollary 4. *For each integer $k \geq 4$, there exists a tree T that is asteroidal-(non-$(k-1)$-SRIG)-free, but is not k-SRIG.*

Corollary 5. *For each integer $k \geq 4$, there exists a tree T that is asteroidal-(non-$(k-1)$-ESRIG)-free, but is not k-ESRIG.*

In order to prove these theorems, we develop some tools to study k-stabbed rectangle intersection representations using special kinds of curves in the representation that are derived from induced paths in the graph.

Consider a k-stabbed rectangle intersection representation $\mathcal{R}$ of a graph G. In this representation, we say that a curve is *rectilinear* if it consists of vertical and horizontal line segments and each horizontal line segment in it lies on a stab line. Given an induced path $P = u_1 u_2 \dots u_s$ in G and two distinct points $p \in r_{u_1}$ and $p' \in r_{u_s}$ such that p, p' lie on stab lines, a rectilinear curve *through P from p to p'* is a simple rectilinear curve $\mathbf{p}$ that starts at p and ends at p', where $\mathbf{p} \subseteq \bigcup_{i=1}^{s} r_{u_i}$ and $\mathbf{p} \cap r_{u_i}$ is arc-connected (and nonempty) for each $i \in \{1, 2, \dots, s\}$. Note that such a curve always exists and that for each $i \in \{1, 2, \dots, s\}$, the curve contains some point in r_{u_i} that is on a stab line.

Given a set X of consecutive stab lines $y = a_1$, $y = a_2$, $\dots$, $y = a_t$, such that $a_1 < a_2 < \dots < a_t$, we say that $y = a_1$ is the bottom stab line in X

and $y = a_t$ is the top stab line in X. Further, we say that a connected induced subgraph H of G is *X-spanning* if there is some vertex in H on each stab line in X. An induced path in G is said to be an *X-spanning path* if its starting and ending vertices are on the top and bottom stab lines of X respectively. Note that if a subgraph H of G is X-spanning, then there is an X-spanning path in H (to see this, consider the shortest path between two vertices u and v in H such that u is on the top stab line in X and v is on the bottom stab line in X).

In the following, we use the term "region" to denote an arc-connected region of the plane that is bounded by a closed rectilinear curve which is the union of four simple rectilinear curves that satisfy some special properties (we assume that a region does not contain the points on its boundary). Suppose $\mathbf{t}$, $\mathbf{l}$, $\mathbf{b}$, and $\mathbf{r}$ are four simple rectilinear curves such that $\mathbf{l} \cap \mathbf{r} = \emptyset$, $\mathbf{t} \cap \mathbf{b} = \emptyset$, and for each $(\mathbf{x}, \mathbf{y}) \in \{(\mathbf{t}, \mathbf{l}), (\mathbf{l}, \mathbf{b}), (\mathbf{b}, \mathbf{r}), (\mathbf{r}, \mathbf{t})\}$, the curves $\mathbf{x}$ and $\mathbf{y}$ have exactly one point in common which is also an end point of both of them. Then, the region $R = (\mathbf{t}, \mathbf{l}, \mathbf{b}, \mathbf{r})$ is the bounded arc-connected component of $\mathbb{R}^2 \setminus (\mathbf{t} \cup \mathbf{l} \cup \mathbf{b} \cup \mathbf{r})$. The closed rectilinear curve $\mathbf{t} \cup \mathbf{l} \cup \mathbf{b} \cup \mathbf{r}$ is called the "boundary" of R. For a region R, we let $\mathcal{L}_{\mathcal{R}}(R)$ denote the set of stab lines of $\mathcal{R}$ that intersect R. Also, let G_R denote the subgraph induced in G by the vertices whose rectangles lie completely inside R.

Observation 2.8.1. *Let ℓ_t, ℓ_b be the stab lines just above and just below the top and bottom stab lines in $\mathcal{L}_{\mathcal{R}}(R)$ respectively. Then, no point on the boundary of R lies above ℓ_t or below ℓ_b.*

Proof. Suppose that the boundary of R contains a point p that is above ℓ_t. Let p' be an arbitrary point in R that is on the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$. It is easy to see that there exists a simple curve from p' to p all of whose points except p belong to R. Since p' is below ℓ_t and p above it, there must be a point on this curve that lies on ℓ_t. But this would mean that R intersects ℓ_t, contradicting the fact that $\ell_t \notin \mathcal{L}_{\mathcal{R}}(R)$. Using similar arguments, we can prove that no point on the boundary of R lies below ℓ_b. $\square$

Definition 2.8.1. *A region* $R = (\mathbf{t}, \mathbf{l}, \mathbf{b}, \mathbf{r})$ *is said to be "good" if it has the following properties:*

(i) the parts of $\mathbf{l}$ *and* $\mathbf{r}$ *that are above the top stab line in* $\mathcal{L}_{\mathcal{R}}(R)$ *and below the bottom stab line in* $\mathcal{L}_{\mathcal{R}}(R)$ *consist of just a vertical segment each, or in other words, every horizontal segment of* $\mathbf{l}$ *and* $\mathbf{r}$ *lies on a stab line in* $\mathcal{L}_{\mathcal{R}}(R)$*,*

(ii) no point of $\mathbf{t}$ *lies below the bottom stab line of* $\mathcal{L}_{\mathcal{R}}(R)$*, and*

(iii) no point of $\mathbf{b}$ *lies above the top stab line of* $\mathcal{L}_{\mathcal{R}}(R)$*.*

For a good region $R = (\mathbf{t}, \mathbf{l}, \mathbf{b}, \mathbf{r})$, we let $\mathbf{top}(R) = \mathbf{t}$ and $\mathbf{bottom}(R) = \mathbf{b}$.

Let $R = (\mathbf{t}, \mathbf{l}, \mathbf{b}, \mathbf{r})$ be a good region with $|\mathcal{L}_{\mathcal{R}}(R)| \geq 1$. Let P_1 and P_2 be two neighbour-disjoint $\mathcal{L}_{\mathcal{R}}(R)$-spanning paths in G_R. For $i \in \{1, 2\}$, let u_i, v_i be the endvertices of P_i that are on the top and bottom stab lines in $\mathcal{L}_{\mathcal{R}}(R)$ respectively. For $i \in \{1, 2\}$, let $\mathbf{p_i}$ be a rectilinear curve that starts at a point $(x_i, y_i) \in r_{u_i}$ on the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$ and ends at a point $(x'_i, y'_i) \in r_{v_i}$ on the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$, with the following additional properties:

(i) The only point in $\mathbf{p_i}$ that is in r_{u_i} and is also on the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$ is (x_i, y_i), and

(ii) The only point in $\mathbf{p_i}$ that is in r_{v_i} and is also on the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ is (x'_i, y'_i).

It is not difficult to see that the curves $\mathbf{p_1}, \mathbf{p_2}$ always exist. (Take any rectilinear curve $\mathbf{q}$ through P_i between some point on the top stab line in r_{u_i} and some point on the bottom stab line in r_{v_i}. Let (x_i, y_i) be the last point in $\mathbf{q}$ that is both in r_{u_i} and is on the top stab line and let (x'_i, y'_i) be the first point in $\mathbf{q}$ that is both in r_{v_i} and is on the bottom stab line. Then the subcurve of $\mathbf{q}$ between (x_i, y_i) and (x'_i, y'_i) can be taken as $\mathbf{p_i}$.)

Suppose that there is a path in G_R between a vertex of P_1 and a vertex of P_2. Then, let P be the induced path in G_R between a vertex w_1 in P_1 and a vertex w_2 in P_2 such that all other vertices of P belong to neither P_1 nor P_2. Let p_1, p_2 be points on stab lines where for $i \in \{1, 2\}$, $p_i \in \mathbf{p_i} \cap r_{w_i}$, such that there exists a rectilinear curve $\mathbf{p}$ through P from p_1 to p_2, whose interior points belong to neither $\mathbf{p_1}$ nor $\mathbf{p_2}$ (note that p_1, p_2 and $\mathbf{p}$ always exist — take arbitrary points p, p' on stab lines such that $p \in \mathbf{p_1} \cap r_{w_1}$, $p' \in \mathbf{p_2} \cap r_{w_2}$ and consider the rectilinear curve $\mathbf{p'}$ through P between p and p'; p_1, p_2 can be chosen to be the closest pair of points on $\mathbf{p'}$ such that $p_1 \in \mathbf{p_1}$, $p_2 \in \mathbf{p_2}$, and the part of $\mathbf{p'}$ between p_1 and p_2 can be chosen as $\mathbf{p}$). Please refer to Figure 2.8.1(a) for an example showing the different curves in R. For $i \in \{1, 2\}$, let $\mathbf{s_i}$ be the shortest vertical line segment with its bottom endpoint being (x_i, y_i) and top endpoint being a point on the boundary of R. Similarly, for $i \in \{1, 2\}$, let $\mathbf{s'_i}$ be the shortest vertical line segment with its top endpoint being (x'_i, y'_i) and bottom endpoint being a point on the boundary of R (refer Figure 2.8.1(b)).

Observation 2.8.2. *For each $i \in \{1, 2\}$, the top endpoint of $\mathbf{s_i}$ lies on the stab line just above the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$ and on a horizontal segment of $\mathbf{t}$ and the bottom endpoint of $\mathbf{s'_i}$ lies on the stab line just below the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ and on a horizontal segment of $\mathbf{b}$.*

Proof. For $i \in \{1, 2\}$, we know that the top endpoint of $\mathbf{s_i}$ lies on the boundary of R, and hence on a horizontal segment of the boundary of R. This implies that the top endpoint of $\mathbf{s_i}$ lies on a stab line. Also, note that the bottom endpoint of $\mathbf{s_i}$ is a point in R that is on the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$. This means that the top endpoint of $\mathbf{s_i}$ lies above the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$. Since the top endpoint of $\mathbf{s_i}$ lies on the boundary of R, we immediately have from Observation 2.8.1 that it lies on the stab line

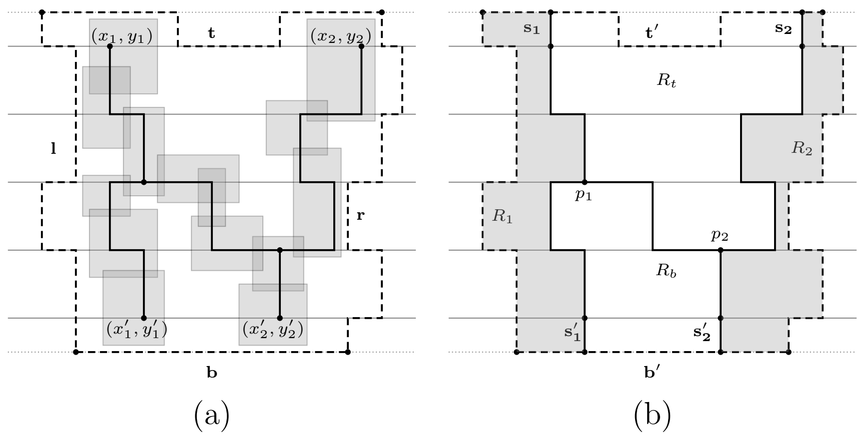


Figure 2.8.1: An example of a good region $R = (\mathbf{t}, \mathbf{l}, \mathbf{b}, \mathbf{r})$ (whose boundary is shown using thick dashed lines) containing the rectangles corresponding to minimal spanning paths P_1 and P_2 and a path P connecting them. (a) shows the rectilinear curves $\mathbf{p_1}$, $\mathbf{p_2}$ and $\mathbf{p}$ through these paths using thick solid lines. (b) shows the partition of R into the four regions R_1, R_2, R_t and R_b.

just above the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$. Also, since it lies on a horizontal segment of the boundary of R, it lies on some horizontal segment that belongs to one of the curves $\mathbf{t}, \mathbf{l}, \mathbf{b}, \mathbf{r}$. Since R is good, we know that no horizontal segment of $\mathbf{l}$, $\mathbf{r}$ or $\mathbf{b}$ lies above the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$. This means that the top endpoint of $\mathbf{s_i}$ lies on a horizontal segment of $\mathbf{t}$. Using similar reasoning, it can be seen that for $i \in \{1, 2\}$, the bottom endpoint of $\mathbf{s'_i}$ lies on the stab line just below the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ and on a horizontal segment of $\mathbf{b}$. □

Let $\mathbf{t'} \subseteq \mathbf{t}$ be the portion of the curve $\mathbf{t}$ that starts at the top endpoint of $\mathbf{s_1}$ and ends at the top endpoint of $\mathbf{s_2}$. Similarly, let $\mathbf{b'} \subseteq \mathbf{b}$ be the portion of the curve $\mathbf{b}$ that starts at the bottom endpoint of $\mathbf{s'_1}$ and ends at the bottom endpoint of $\mathbf{s'_2}$.

For $i \in \{1, 2\}$, let the curve $\mathbf{p_i^t}$ be the connected portion of $\mathbf{p_i}$ that starts at (x_i, y_i) and ends at the common point of $\mathbf{p_i}$ and $\mathbf{p}$ (denoted as p_i previously) and let the curve $\mathbf{p_i^b}$ be the connected portion of $\mathbf{p_i}$ that starts at the common point of $\mathbf{p_i}$ and $\mathbf{p}$ and ends at (x'_i, y'_i).

Let R_1, R_2, R_t, R_b be the regions into which the region R gets split by the union of the curves $\mathbf{p_1}, \mathbf{p_2}, \mathbf{p}, \mathbf{s_1}$, $\mathbf{s'_1}, \mathbf{s_2}, \mathbf{s'_2}$, where R_i, for $i \in \{1, 2\}$, is the region whose boundary contains $\mathbf{p_i}$, $R_t = (\mathbf{t'}, \mathbf{s_1} \cup \mathbf{p_1^t}, \mathbf{p}, \mathbf{p_2^t} \cup \mathbf{s_2})$, and $R_b = (\mathbf{p}, \mathbf{p_1^b} \cup \mathbf{s'_1}, \mathbf{b'}, \mathbf{s'_2} \cup \mathbf{p_2^b})$ (please refer to Figure 2.8.1(b)).

Observation 2.8.3. *From the definition of R_t and R_b, we have:*

(i) $\mathbf{top}(R_t) \subseteq \mathbf{top}(R)$ *and* $\mathbf{bottom}(R_b) \subseteq \mathbf{bottom}(R)$.

(ii) $\mathbf{bottom}(R_t) = \mathbf{top}(R_b)$.

(iii) If x is a vertex in P, then r_x intersects $\mathbf{bottom}(R_t)$ *(*$= \mathbf{top}(R_b)$*).*

(iv) Let $x \in V(G)$ such that r_x intersects $\mathbf{bottom}(R_t)$ *(*$= \mathbf{top}(R_b)$*). Then x has a neighbour in P.*

For the rest of this section, for a good region R and paths P_1, P_2, P such that:

- P_1 and P_2 are two neighbour-disjoint $\mathcal{L}_{\mathcal{R}}(R)$-spanning paths in G_R, and
- P is an induced path in G_R between a vertex in P_1 and a vertex in P_2 such that all vertices of P other than its end vertices belong to neither P_1 nor P_2 (note that such a path will exist if there is some path in G_R between a vertex of P_1 and a vertex of P_2),

we shall denote by $\Delta(\mathcal{R}, R, P_1, P_2, P)$ the ordered pair (R_t, R_b), where the regions R_t and R_b are obtained using the procedure described above. We shall now prove some observations about the regions R_t and R_b.

Lemma 2.8.1.

(a) The curve $\mathbf{t'}$ *(resp.* $\mathbf{b'}$*) does not intersect the bottom (resp. top) stab line in $\mathcal{L}_{\mathcal{R}}(R)$.*

(b) *R_t does not intersect the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ and R_b does not intersect the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$.*

Proof. Let us first prove (a). We shall only show that the curve $\mathbf{t'}$ does not intersect the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ as the other case is similar. Let the rectilinear curve $\mathbf{q}$ be $\mathbf{p_1^t} \cup \mathbf{p} \cup \mathbf{p_2^t}$. Note that $\mathbf{q}$ is a simple rectilinear curve. Let ℓ be the stab line just above the top stab line of $\mathcal{L}_{\mathcal{R}}(R)$. From Observation 2.8.2, we have that the top endpoints of $\mathbf{s_1}$ and $\mathbf{s_2}$ lie on ℓ. Let the horizontal line segment (that lies entirely on ℓ) between these two points be denoted by $\mathbf{s}$. Let R' be the region bounded by $\mathbf{s_1} \cup \mathbf{q} \cup \mathbf{s_2} \cup \mathbf{s}$. From Observation 2.8.1, it is then clear that $\mathbf{t'}$ lies entirely in $R' \cup \mathbf{s}$ (recall that R' consists only of the points in the interior of the region bounded by $\mathbf{s_1} \cup \mathbf{q} \cup \mathbf{s_2} \cup \mathbf{s}$). Since the points in $\mathbf{q}$ all belong to rectilinear curves through paths in G_R, every horizontal segment of $\mathbf{q}$ is on a stab line in $\mathcal{L}_{\mathcal{R}}(R)$. Since the endpoints of $\mathbf{q}$ lie on the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$, and $\mathbf{q}$ is a simple rectilinear curve, it follows that every point in $\mathbf{q}$ is on or above the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$. As the points in $\mathbf{s_1} \cup \mathbf{s_2} \cup \mathbf{s}$ lie on or above the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$, this means that all the points on the boundary of R' lie on or above the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$, implying that R' does not intersect the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$. As $\mathbf{s}$ lies on the stab line just above the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$, we now have that $R' \cup \mathbf{s}$ does not intersect the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$. From our earlier observation that $\mathbf{t'}$ lies entirely in $R' \cup \mathbf{s}$, we now have that $\mathbf{t'}$ does not intersect the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$.

To prove (b), we shall only prove that R_t does not intersect the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ as the case for R_b involves similar arguments. Note that the boundary of R_t is $\mathbf{t'} \cup \mathbf{s_1} \cup \mathbf{q} \cup \mathbf{s_2}$. From the arguments in the previous paragraph, it is easy to see that all the points in $\mathbf{s_1} \cup \mathbf{q} \cup \mathbf{s_2}$ lie on or above the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$. Combining this with (a), we now have that all the points on the boundary of R_t lie on or above the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$. Hence we can conclude that the bottom stab line in $\mathcal{L}_{\mathcal{R}}(R)$ does not intersect R_t. □

An $\mathcal{L}_\mathcal{R}(R)$-spanning path P is said to be a *minimal $\mathcal{L}_\mathcal{R}(R)$-spanning path* if there is no $\mathcal{L}_\mathcal{R}(R)$-spanning path P' such that $V(P') \subset V(P)$. Note that the existence of an $\mathcal{L}_\mathcal{R}(R)$-spanning path in a graph implies the existence of a minimal $\mathcal{L}_\mathcal{R}(R)$-spanning path in the graph.

Lemma 2.8.2. *Suppose that P_1 and P_2 are minimal $\mathcal{L}_\mathcal{R}(R)$-spanning paths. Let $R' \in \{R_t, R_b\}$ such that $|\mathcal{L}_\mathcal{R}(R')| \geq |\mathcal{L}_\mathcal{R}(R)| - 1$. Then R' is good.*

Proof. We shall prove this only for the case when $R' = R_t$ as the other case is similar. As $|\mathcal{L}_\mathcal{R}(R_t)| \geq |\mathcal{L}_\mathcal{R}(R)| - 1$, and by Lemma 2.8.1(b), R_t does not intersect the bottom stab line, we know that $\mathcal{L}_\mathcal{R}(R_t)$ consists of all the stab lines in $\mathcal{L}_\mathcal{R}(R)$ other than the bottom stab line in $\mathcal{L}_\mathcal{R}(R)$.

Recall that $R_t = (\mathbf{t'}, \mathbf{s_1} \cup \mathbf{p_1^t}, \mathbf{p}, \mathbf{p_2^t} \cup \mathbf{s_2})$. Since the paths P_1 and P_2 are minimal, we know that for $i \in \{1, 2\}$, u_i is the only vertex on P_i that is on the top stab line in $\mathcal{L}_\mathcal{R}(R)$ and v_i is the only vertex on P_i that is on the bottom stab line in $\mathcal{L}_\mathcal{R}(R)$. Therefore, from the definition of curves $\mathbf{p_1}$ and $\mathbf{p_2}$, we have that for $i \in \{1, 2\}$, the only points of $\mathbf{p_i}$ that lie on the top and bottom stab lines in $\mathcal{L}_\mathcal{R}(R)$ are the endpoints of $\mathbf{p_i}$, which further implies that $\mathbf{p_i}$ does not contain any horizontal segment on the top or bottom stab lines in $\mathcal{L}_\mathcal{R}(R)$. It follows that for $i \in \{1, 2\}$, $\mathbf{p_i^t}$, and therefore $\mathbf{s_i} \cup \mathbf{p_i^t}$, also does not contain any horizontal segment on the top or bottom stab lines in $\mathcal{L}_\mathcal{R}(R)$. As $\mathcal{L}_\mathcal{R}(R_t)$ consists of all the stab lines in $\mathcal{L}_\mathcal{R}(R)$ other than the bottom stab line in $\mathcal{L}_\mathcal{R}(R)$, we have that $\mathbf{s_1} \cup \mathbf{p_1^t}$ and $\mathbf{s_2} \cup \mathbf{p_2^t}$ do not contain any horizontal segment that lies above the top stab line in $\mathcal{L}_\mathcal{R}(R_t)$ or below the bottom stab line in $\mathcal{L}_\mathcal{R}(R_t)$. Therefore, R_t satisfies property (i) of Definition 2.8.1. From Lemma 2.8.1(a), we have that $\mathbf{t'}$ does not intersect the bottom stab line in $\mathcal{L}_\mathcal{R}(R)$. Since the endpoints of $\mathbf{t'}$ lie above the top stab line in $\mathcal{L}_\mathcal{R}(R)$, we can then conclude using the definition of rectilinear curves that no point of $\mathbf{t'}$ lies below the bottom stab line of $\mathcal{L}_\mathcal{R}(R_t)$. Thus, R_t satisfies property (ii) of Definition 2.8.1. Since the points in $\mathbf{p}$ all belong to rectangles contained in R and $\mathbf{p}$ is a simple rectilinear curve, we know that all of them are on or below the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$ and hence on or below the top stab line in $\mathcal{L}_{\mathcal{R}}(R_t)$. Therefore, R_t satisfies property (iii) of Definition 2.8.1 as well. This completes the proof. □

Observation 2.8.4. *Let $v \in V(G)$. For $i \in \{1,2\}$, if $r_v \cap \mathbf{t'} = \emptyset$ (resp. $r_v \cap \mathbf{b'} = \emptyset$) and r_v intersects $\mathbf{s_i}$ (resp. $\mathbf{s'_i}$), then r_v contains the point (x_i, y_i) (resp. (x'_i, y'_i)).*

Proof. Suppose that $r_v \cap \mathbf{t'} = \emptyset$, but r_v intersects $\mathbf{s_i}$. As the top endpoint of $\mathbf{s_i}$ is contained in $\mathbf{t'}$, we can infer that r_v does not contain the top endpoint of $\mathbf{s_i}$. If r_v also does not contain the bottom endpoint of $\mathbf{s_i}$, then there is no stab line that intersects r_v, as the top and bottom endpoints of $\mathbf{s_i}$ are on consecutive stab lines. We can therefore conclude that the bottom endpoint of $\mathbf{s_i}$, which is (x_i, y_i), is contained in r_v. The arguments for the other case are similar and are therefore omitted. □

Lemma 2.8.3. *Let $v \in V(G)$ such that r_v intersects the boundary of R_t (resp. R_b). Then either r_v intersects $\mathbf{t'} = \mathbf{top}(R_t)$ (resp. $\mathbf{b'} = \mathbf{bottom}(R_b)$) or v has a neighbour on at least one of the paths P_1, P_2, or P.*

Proof. We shall prove this lemma only for R_t as the arguments for R_b are similar. Suppose there exists a vertex $v \in V(G)$ such that r_v intersects the boundary of R_t, but r_v does not intersect $\mathbf{t'}$ and v does not have a neighbour on any of the paths P_1, P_2, or P. Then r_v does not intersect any of the curves $\mathbf{p_1^t}$, $\mathbf{p}$, or $\mathbf{p_2^t}$. From this, it follows that r_v does not contain the points (x_1, y_1) or (x_2, y_2). By Observation 2.8.4, we now have that r_v does not intersect $\mathbf{s_1}$ or $\mathbf{s_2}$. Since the boundary of R_t is $\mathbf{t'} \cup \mathbf{s_1} \cup \mathbf{p_1^t} \cup \mathbf{p} \cup \mathbf{p_2^t} \cup \mathbf{s_2}$, this means that r_v does not intersect the boundary of R_t, which is a contradiction. □

Lemma 2.8.4. *Let $v \in V(G_R)$ such that P misses v and there is a path in G_R from v to a vertex in P that misses both P_1 and P_2. Then r_v is contained in R_t or R_b.*

Proof. As v is not adjacent to any vertex in P, P_1 or P_2, the rectangle r_v does not intersect $\mathbf{p}$, $\mathbf{p_1}$ or $\mathbf{p_2}$. Also, as $v \in V(G_R)$, r_v does not intersect $\mathbf{t}'$ or $\mathbf{b}'$. Then, by Observation 2.8.4, we can further infer that r_v does not intersect $\mathbf{s_1}$, $\mathbf{s_2}$, $\mathbf{s_1'}$ or $\mathbf{s_2'}$. This means that r_v is contained in one of the regions R_1, R_2, R_t or R_b. Now suppose for the sake of contradiction that r_v is contained in R_1. We know from the statement of the lemma that there is at least one path in G_R from v to some vertex in P that misses both P_1 and P_2. Let Q be such a path of minimum length and let u be the endpoint of Q other than v. It is clear that $V(P) \cap V(Q) = \{u\}$. Let p' be a point in $r_u \cap \mathbf{p}$ that is on a stab line (recall from the definition of rectilinear curves through paths that such a point exists). As u has no neighbour on P_1 or P_2, it can be seen that p' is not an endpoint of $\mathbf{p}$, i.e., p' is an interior point of $\mathbf{p}$. Now consider the rectilinear path through Q from some point in r_v (that is on a stab line) to p'. As the point p' is not inside or on the boundary of R_1, this rectilinear curve must cross the boundary of R_1 at some point p''. It is clear that there is a vertex x in Q such that $p'' \in r_x$. Since r_x is contained in R, we can infer that p'' is on $\mathbf{s_1} \cup \mathbf{p_1} \cup \mathbf{s_1'}$ and also that r_x does not intersect $\mathbf{t}'$ or $\mathbf{b}'$. If p'' is on $\mathbf{s_1}$ or $\mathbf{s_1'}$, we have by Observation 2.8.4 that r_x intersects $\mathbf{p_1}$. So we can conclude that in any case, r_x intersects $\mathbf{p_1}$. Since from the definition of $\mathbf{p_1}$, every point of $\mathbf{p_1}$ belongs to the rectangle corresponding to some vertex of P_1, this implies that x is adjacent to some vertex of P_1. This contradicts the fact that Q misses P_1. We can thus conclude that r_v is not contained in R_1. Using similar arguments, we can also infer that r_v is not contained in R_2. This completes the proof. □

Lemma 2.8.5. *Let $v, w \in V(G_R)$ such that r_v is contained in $R' \in \{R_t, R_b\}$ and there is a path in G_R between v and w that misses P_1, P_2 and P. Then r_w is contained in R'.*

Proof. We shall prove the statement of the lemma only for the case $R' = R_t$ as the proof for the case $R' = R_b$ is similar. Let Q be the path between v and w in G_R that misses P_1, P_2 and P. Let x be any vertex on Q. Clearly, x has no neighbour on P_1, P_2 or P. As $x \in V(G_R)$, the rectangle r_x is contained in R, implying that r_x does not intersect the boundary of R. As we have $\mathbf{top}(R_t) \subseteq \mathbf{top}(R)$ by Observation 2.8.3(i), this means that r_x does not intersect $\mathbf{top}(R_t)$. By Lemma 2.8.3, we now have that r_x does not intersect the boundary of R_t. Therefore, no rectangle corresponding to a vertex in Q can intersect the boundary of R_t. Since r_v is contained in R_t, this means that the rectangle corresponding to each vertex of Q, and hence r_w, is contained in R_t. □

We shall use the technical details about good regions and rectilinear curves only for the proof of Theorem 2.8.2. We now give a lemma that shall be sufficient for most of the other proofs. Given a graph G and a representation $\mathcal{R}$ of G, we shall define $\mathcal{L}_{\mathcal{R}}(H)$, for any connected induced subgraph H of G, to be the set of stab lines of $\mathcal{R}$ that intersect the rectangle corresponding to some vertex in $V(H)$. Note that $\mathcal{L}_{\mathcal{R}}(H)$ will contain a consecutive set of stab lines of $\mathcal{R}$.

Lemma 2.8.6. *Let G be a connected k-SRIG and $\mathcal{R}$ a k-stabbed rectangle intersection representation of it. Let H_1 and H_2 be two neighbour-disjoint connected induced subgraphs of G such that $\mathcal{L}_{\mathcal{R}}(H_1) = \mathcal{L}_{\mathcal{R}}(H_2) = \mathcal{L}_{\mathcal{R}}(G) = k$. Let P be an induced path in G between some vertex in $V(H_1)$ and some vertex in $V(H_2)$ such that no internal vertex of P is in $V(H_1)$ or $V(H_2)$. Let H be a connected induced subgraph of G that is neighbour-disjoint from H_1, H_2 and P such that there is a vertex in H from which there is a path to a vertex of P that misses both H_1 and H_2. Then, $\mathcal{L}_{\mathcal{R}}(H) \subset \mathcal{L}_{\mathcal{R}}(G)$.*

Proof. We shall augment $\mathcal{R}$ to a new representation $\mathcal{R}'$ by adding two new stab lines, one above the top stab line and the other below the bottom stab line of $\mathcal{R}$. Notice that for any connected induced subgraph G' of G, we have $\mathcal{L}_{\mathcal{R}'}(G') = \mathcal{L}_{\mathcal{R}}(G')$. Let A be a good region that contains all the rectangles of $\mathcal{R}'$, i.e., $G_A = G$ (note that such a region exists; we can consider a rectangle with top and bottom edges on the top and bottom stab lines such that it contains all the rectangles of $\mathcal{R}'$). As the only two stab lines that are not intersected by any rectangle in $\mathcal{R}'$ are the top and bottom stab lines (recall that $\mathcal{L}_{\mathcal{R}}(G)$ contains all the stab lines of $\mathcal{R}$), it follows that $\mathcal{L}_{\mathcal{R}'}(A) = \mathcal{L}_{\mathcal{R}'}(G)$. It is clear that for any induced subgraph G' of G, $\mathcal{L}_{\mathcal{R}'}(G') = \mathcal{L}_{\mathcal{R}}(G')$. Therefore, we have $\mathcal{L}_{\mathcal{R}'}(H_1) = \mathcal{L}_{\mathcal{R}'}(H_2) = \mathcal{L}_{\mathcal{R}'}(G)$, which implies that there are $\mathcal{L}_{\mathcal{R}'}(A)$-spanning paths in each of them. Let P_1 and P_2 be minimal $\mathcal{L}_{\mathcal{R}'}(A)$-spanning paths in H_1 and H_2 respectively. As H_1 and H_2 are neighbour-disjoint, P_1 and P_2 are neighbour-disjoint. It is not hard to see that there exists an induced path P' in $G[V(H_1) \cup V(P) \cup V(H_2)]$ that contains P as a subpath, such that P' connects some vertex of P_1 to some vertex of P_2 and no internal vertex of P' belongs to either P_1 or P_2. Let $(A_t, A_b) = \Delta(\mathcal{R}', A, P_1, P_2, P')$.

We know that there exists a vertex, say v, in H such that there is a path from v to a vertex of P that misses both H_1 and H_2. Clearly, this is also a path from v to a vertex in P' that misses both P_1 and P_2. As H is neighbour-disjoint from P, we know that P misses v. By Lemma 2.8.4, we know that r_v is contained in A_t or A_b. Let us assume without loss of generality that r_v is contained in A_t. Since H is a connected induced subgraph of G that is neighbour-disjoint from H_1, H_2 and P, we know that there is a path from v to each vertex of H that misses H_1, H_2 and P. This means that there is a path from v to each vertex of H that misses P_1, P_2 and P'. Now, we can use Lemma 2.8.5 to conclude that the rectangles corresponding to the vertices of H are all contained in A_t. Since by Lemma 2.8.1(b), we know that $\mathcal{L}_{\mathcal{R}'}(A_t) \subset \mathcal{L}_{\mathcal{R}'}(A)$, we can now conclude that $\mathcal{L}_{\mathcal{R}'}(H) \subset \mathcal{L}_{\mathcal{R}'}(G)$, and therefore $\mathcal{L}_{\mathcal{R}}(H) \subset \mathcal{L}_{\mathcal{R}}(G)$. □

Proof of Theorem 2.8.1.

Let T be the block graph obtained by taking a copy of the tree G_2 (defined in Section 2.7) and then introducing a true twin for one of the leaves. Let w, w' be the two true twins in T, v be their common neigh-

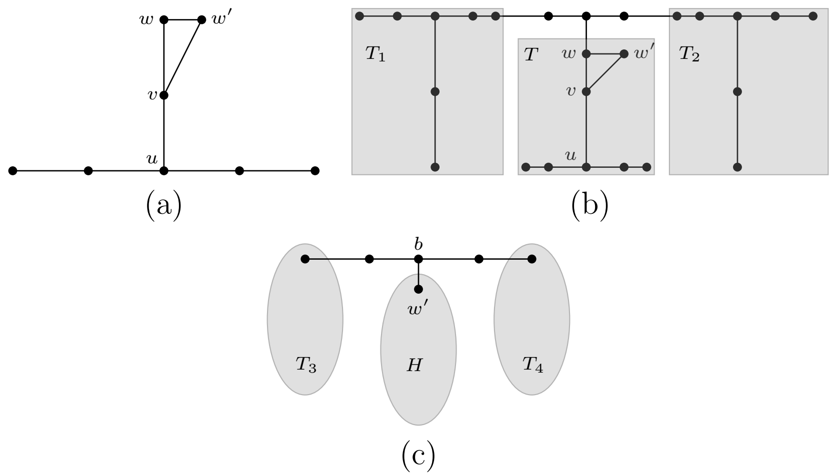


Figure 2.8.2: Construction of block graph for Proof of Theorem 2.8.1. (a) Construction of T. (b) Construction of H. T_1 and T_2 are isomorphic to G_2. (c) Construction of G. T_3 and T_4 are isomorphic to G_3 and H is the block graph shown in (b).

bour and u the degree 3 vertex adjacent to v. See Figure 2.8.2(a) for a drawing of T. Notice that the graph G_2 is non-interval (folklore, or by Lemma 2.7.1(i)).

Let T_1 and T_2 be trees each isomorphic to G_2. Let H be the graph obtained by taking the disjoint union of T_1, T_2 and T and then doing the following: introduce a new vertex a, connect a to a leaf of T_1 and to a leaf of T_2 using paths of length 2 and then make a adjacent to w (see Figure 2.8.2(b)).

Claim 1. H is non-(2-SRIG).

Proof. Note that $T - \{w\}$ is isomorphic to G_2, and hence is non-interval. As T_1, T_2, $T - \{w\}$ are asteroidal-(non-interval) in H, by Theorem 2.5.1, we have that H is non-(2-SRIG).

It is easy to see that $H - \{w'\}$ is asteroidal-(non-interval)-free. Hence, by Theorem 2.6.1, we have that $H - \{w'\}$ is 2-SRIG.

Claim 2. The vertices w and v do not have a common stab in any 2-stabbed rectangle intersection representation of $H - \{w'\}$.

Proof. Let $H' = H - \{w'\}$. Let $\mathcal{R}$ be any 2-stabbed rectangle intersection representation of H'. Since T_1 and T_2 are neighbour-disjoint connected induced subgraphs of H' that are non-interval, we have that $|\mathcal{L}_{\mathcal{R}}(T_1)| = |\mathcal{L}_{\mathcal{R}}(T_2)| = 2$. Let P be the (induced) path between T_1 and T_2 in H'. Notice that $T - \{w, w'\}$ is a connected induced subgraph of H' that is neighbour-disjoint from T_1, T_2 and P. Moreover, there is a path from the vertex v of $T - \{w, w'\}$ to the vertex a of P that misses T_1 and T_2. We can now use Lemma 2.8.6 to conclude that $|\mathcal{L}_{\mathcal{R}}(T - \{w, w'\})| = 1$. Let $\mathcal{L}_{\mathcal{R}}(T - \{w, w'\}) = \{\ell\}$. It is clear that for each vertex of $T - \{w, w'\}$, and hence also for v, the only stab line that intersects the rectangle corresponding to it is ℓ. If r_w also intersects ℓ, then the collection $\{\ell \cap r_x\}_{x \in V(T - \{w'\})}$ would form an interval representation of G_2, which contradicts the fact that G_2 is non-interval. This completes the proof of the claim.

We shall now construct the desired block graph G that satisfies the requirements in the statement of Theorem 2.8.1. Let T_3 and T_4 be trees that are isomorphic to G_3 (defined in Section 2.7). Let G' be the graph formed by taking the disjoint union of T_3 and T_4 and then doing the following: add a new vertex b and connect it to a vertex of T_3 using a path of length 2 and a vertex of T_4 using a path of length 2. The graph G is constructed by taking the disjoint union of H and G' and then adding an edge between b and w' (see Figure 2.8.2(c) for a schematic diagram of G).

Claim 3. G is not 3-SRIG.

Proof. Suppose for the sake of contradiction that G is 3-SRIG. Let $\mathcal{R}$ be a 3-stabbed rectangle intersection representation of G. Since T_3 and T_4 are neighbour-disjoint connected induced subgraphs of G that are non-(2-SRIG) (recall that T_3 and T_4 are isomorphic to G_3 and that G_3 is non-(2-SRIG) by Lemma 2.7.1(i)), we have that $|\mathcal{L}_{\mathcal{R}}(T_3)| = |\mathcal{L}_{\mathcal{R}}(T_4)| = 3$. Let P

be the path between T_3 and T_4 in G. Notice that $H - \{w'\}$ is a connected induced subgraph of G that is neighbour-disjoint from T_3, T_4 and P. Moreover, there is a path from the vertex w of $H - \{w'\}$ to the vertex b of P that misses T_3 and T_4. We can now use Lemma 2.8.6 to conclude that $|\mathcal{L}_{\mathcal{R}}(H - \{w'\})| = 2$. This means that in $\mathcal{R}$, the rectangles corresponding to $H - \{w'\}$ form a 2-stabbed rectangle intersection representation of $H - \{w'\}$. Then, by Claim 2, we know that neither of the two stab lines in $\mathcal{L}_{\mathcal{R}}(H - \{w'\})$ intersects both r_w and r_v. Since w' is adjacent to both w and v, this implies that $r_{w'}$ intersects at least one of the two stab lines in $\mathcal{L}_{\mathcal{R}}(H - \{w'\})$. But then, the rectangles corresponding to the vertices of H, together with the stab lines in $\mathcal{L}_{\mathcal{R}}(H - \{w'\})$, form a 2-stabbed rectangle intersection representation of H. This contradicts Claim 1.

To complete the proof of the theorem, we only need to show that G is asteroidal-(non-2-SRIG)-free. Suppose for the sake of contradiction that there exist induced subgraphs X_1, X_2, X_3 that are asteroidal-(non-2-SRIG) in G. First we need the following claim, whose proof is left to the reader.

Claim. In any block graph that contains three induced subgraphs that are asteroidal-$\mathcal{C}$ in it, for some graph class $\mathcal{C}$, there exists either a cutvertex that has no neighbour in each of the three subgraphs, or a triangle, whose removal results in a graph in which each of the three subgraphs is in a different component.

From the above claim, we have that either there exists a vertex $x \in V(G)$ such that $G - \{x\}$ has three components X'_1, X'_2, X'_3 such that for each $i \in \{1, 2, 3\}$, $V(X_i) \subseteq V(X'_i) \setminus N[x]$, or X_1, X_2, X_3 are each contained in a different component of $G - \{w, w', v\}$ (since the only triangle in G is formed by w, w' and v). Let us first suppose that X_1, X_2, X_3 are each contained in a different component of the three components in $G - \{w, w', v\}$. It is easy to see that the component of $G - \{w, w', v\}$ that contains a neighbour of v is a path and is therefore 1-SRIG, contradicting the fact that it contains one of the non-(2-SRIG) graphs X_1, X_2, X_3. So we can assume that there exists a vertex $x \in V(G)$ such that $G - x$ has three components X_1', X_2', X_3' such that for each $i \in \{1, 2, 3\}$, $V(X_i) \subseteq V(X_i') \setminus N[x]$. Note that since $G - \{x\}$ contains at least three components, degree of x is at least 3 and $x \notin \{w, w', v\}$.

Let us first suppose that $x \in V(G')$. If $x = b$, one of the three components of $G - \{x\}$, say X_1', is H. But now, $V(X_1') \setminus N[x] = H - \{w'\}$, which is 2-SRIG by our earlier observation. This contradicts the fact that $V(X_1) \subseteq V(X_1') \setminus N[x]$ as X_1 is non-(2-SRIG). If $x \neq b$, then $x \in V(T_3)$ or $x \in V(T_4)$. Suppose that $x \in V(T_3)$. As G' is a tree, we know that $G' - \{x\}$ contains at least three components. Also, as $G' - V(T_3)$ has only one component, we can use Observation 2.7.1(ii) to conclude that all components of $G' - \{x\}$ except the component Y that contains b are proper subtrees of T_3. Since the only edge between $V(G) \setminus V(G')$ and $V(G')$ is $w'b$, we can see that every component of $G' - \{x\}$ other than Y is also a component of $G - \{x\}$. This means that at least two components, say X_1', X_2', of $G - \{x\}$ are also components of $G' - \{x\}$. Since $V(X_1) \subseteq V(X_1')$ and $V(X_2) \subseteq V(X_2')$, we have that X_1' and X_2' are non-(2-SRIG) neighbour-disjoint induced subgraphs of T_3. As T_3 is isomorphic to G_3, this is a contradiction to Lemma 2.7.1(iv). For the same reason, we can also conclude that $x \notin V(T_4)$. This means that $x \in V(H)$.

But if $x \in V(H)$, then since $x \notin \{w, w', v\}$, it is clear from the construction of G that at least one of the components, say X_1', of $G - \{x\}$ is an induced subgraph of $H - \{w'\}$. As $H - \{w'\}$ is 2-SRIG by our earlier observation, this means that X_1' is 2-SRIG, which contradicts the fact that it contains the non-(2-SRIG) graph X_1 as an induced subgraph. This shows that G is asteroidal non-(2-SRIG)-free and hence completes the proof. □

We shall now prove a general theorem that will later be used to prove Theorem 2.8.2.

Theorem 2.8.3. *Let $k \geq 4$. For each $i \in \{k, k-1, k-2\}$, let T_i, T'_i be two graphs that are i-SRIG but not $(i-1)$-SRIG and let $a_i \in V(T_i)$ and $a'_i \in V(T'_i)$. For $i \in \{k, k-1, k-2\}$, let H_i be the graph obtained by adding a new vertex b_i to the disjoint union of T_i and T'_i and connecting it to a_i and a'_i using paths of length at least two. Let T be the graph obtained by adding a new vertex c to the disjoint union of H_k, H_{k-1} and H_{k-2} and then connecting c to each of b_k, b_{k-1} and b_{k-2} using paths of length at least two. Then T is not k-SRIG.*

Proof. Suppose for the sake of contradiction that T is k-SRIG. Let $\mathcal{R}$ be a $(k+2)$-stabbed rectangle intersection representation of T in which the top and bottom stab lines do not intersect any rectangle. Let A be a good region that contains all the rectangles of $\mathcal{R}$, i.e., $T_A = T$ (note that such a region exists; we can consider a rectangle with top and bottom edges on the top and bottom stab lines such that it contains all the rectangles of $\mathcal{R}$). As the only two stab lines that are not intersected by any rectangle in $\mathcal{R}$ are the top and bottom stab lines (recall that T_A is not $(k-1)$-SRIG as it contains T_k and T'_k), it follows that $|\mathcal{L}_{\mathcal{R}}(A)| = k$. As T_k and T'_k are k-SRIG but not $(k-1)$-SRIG, we know that there are $\mathcal{L}_{\mathcal{R}}(A)$-spanning paths in each of them. Let X_1 and X_2 be minimal $\mathcal{L}_{\mathcal{R}}(A)$-spanning paths in T_k and T'_k respectively. It is easy to see that X_1 and X_2 are neighbour-disjoint. Let X be an induced path in T_A that connects some vertex of X_1 and some vertex of X_2 such that no internal vertex of X belongs to either X_1 or X_2. Note that X is a subgraph of H_k that contains b_k. Let $(A_t, A_b) = \Delta(\mathcal{R}, A, X_1, X_2, X)$.

(+) By Observation 2.8.3(iv), if for $x \in V(T)$, the rectangle r_x intersects **bottom**(A_t), then x has a neighbour on X.

Since there is a path in T_A from $c \in V(T_A)$ to a vertex in X (in this case, b_k) that misses both X_1 and X_2, we know by Lemma 2.8.4 that r_c is contained in A_t or A_b. We shall assume without loss of generality that r_c is contained in A_t (see Figure 2.8.3(a)). Let us define $B = A_t$. Let T^* be the graph obtained by removing the vertices in $V(H_k)$ and

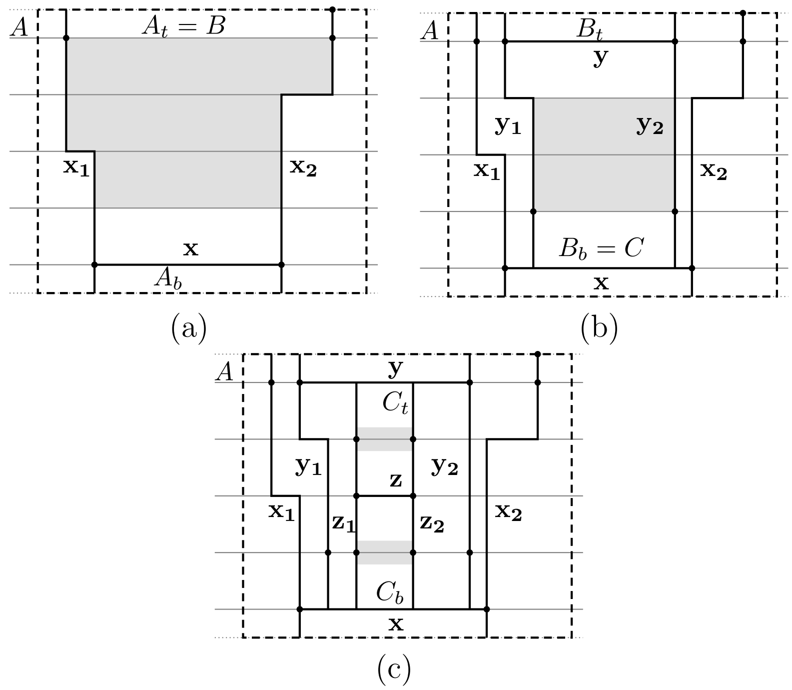


Figure 2.8.3: An illustration of various stages of the proof of Theorem 2.8.3. The region bounded by the dashed curve is A. The solid curves represent the rectilinear curves through paths chosen in the proof to split the regions. For example, the solid curve labelled $\mathbf{x_1}$ is the rectilinear curve through the path X_1, the solid curve labelled $\mathbf{y}$ is the rectilinear curve through the path Y and so on. The shaded region indicates the possible locations of the rectangle r_c as the proof proceeds.

their neighbours from T, or in other words, $T^* = T - (V(H_k) \cup N[b_k])$. Note that there is a path in T_A from c to each vertex of T^* that misses X_1, X_2 and X. We can now infer using Lemma 2.8.5 that the rectangles corresponding to the vertices in T^* are all contained in $A_t = B$. In other words, T^* is a connected induced subgraph of T_B.

Since T^* contains T_{k-1} and T'_{k-1} as induced subgraphs, and is therefore not $(k-2)$-SRIG, we have $|\mathcal{L}_{\mathcal{R}}(B)| \geq k-1$. By Lemma 2.8.2, this means that $B = A_t$ is a good region. Since B does not contain the bottom stab line in $\mathcal{L}_{\mathcal{R}}(A)$ by Lemma 2.8.1(b), we can conclude that $|\mathcal{L}_{\mathcal{R}}(B)| = k-1$. Now, T_{k-1} and T'_{k-1} are two neighbour-disjoint subgraphs of T^* that are $(k-1)$-SRIG but not $(k-2)$-SRIG. Since the rectangles corresponding to the vertices in them are all contained in B (recall that T^* is an induced subgraph of T_B), there is at least one vertex of T_{k-1} and at least one vertex of T'_{k-1} on every stab line in $\mathcal{L}_{\mathcal{R}}(B)$. This means that there exist minimal $\mathcal{L}_{\mathcal{R}}(B)$-spanning paths Y_1 in T_{k-1} and Y_2 in T'_{k-1}, and it is clear that Y_1 and Y_2 are neighbour-disjoint. Let Y be an induced path in T^* that connects some vertex of Y_1 and some vertex of Y_2 such that no internal vertex of Y belongs to either Y_1 or Y_2. Note that Y is a subgraph of H_{k-1} that contains b_{k-1}. Let $(B_t, B_b) = \Delta(\mathcal{R}, B, Y_1, Y_2, Y)$.

(++) By Observation 2.8.3(iv), if for $x \in V(T)$, the rectangle r_x intersects $\mathbf{top}(B_b)$, then x has a neighbour on Y.

Since there is a path in T^* from c to a vertex in Y (in this case, b_{k-1}) that misses both Y_1 and Y_2, we know by Lemma 2.8.4 that r_c is contained in B_t or B_b. Suppose that r_c is contained in B_t. Note that the path Q in T between c and b_k misses Y_1, Y_2 and Y. As b_k lies on the path X, we know by Observation 2.8.3(iii) that r_{b_k} intersects $\mathbf{bottom}(B)$. This means that r_{b_k} contains some points from outside B and hence some points from outside B_t. Since r_c is contained in B_t, this can only mean that there exists some vertex x in Q such that the rectangle r_x intersects the boundary of B_t. Since x has no neighbour on Y_1, Y_2 or Y, we know by Lemma 2.8.3 that r_x intersects $\mathbf{top}(B_t)$. Since $B = A_t$ and A are good regions, we have by Observation 2.8.3(i) that $\mathbf{top}(B_t) \subseteq \mathbf{top}(A_t) \subseteq \mathbf{top}(A)$. This implies that r_x intersects the boundary of A, which is a contradiction to the fact that $T = T_A$ (or in other words, all rectangles corresponding to vertices of T are contained in A). Thus, we can conclude that r_c is not contained in B_t, and hence is contained in B_b (See Figure 2.8.3(b)). Let us define $C = B_b$.

Let T^{**} be the graph obtained by removing the vertices in $V(H_{k-1})$ and their neighbours from T^*, or in other words, $T^{**} = T^* - (V(H_{k-1}) \cup$

$N[b_{k-1}]$). Note that $c \in V(T^{**})$ and that there is a path in T^* from c to each vertex of T^{**} that misses Y_1, Y_2 and Y. We can now infer using Lemma 2.8.5 that the rectangles corresponding to the vertices in T^{**} are all contained in C. In other words, T^{**} is a connected induced subgraph of T_C.

Since T^{**} contains T_{k-2} and T'_{k-2} as induced subgraphs, and is therefore not $(k-3)$-SRIG, we have $|\mathcal{L}_{\mathcal{R}}(C)| \geq k-2$. By Lemma 2.8.2, this means that C is a good region. Since C does not contain the top stab line in $\mathcal{L}_{\mathcal{R}}(B)$ by Lemma 2.8.1(b), we can conclude that $|\mathcal{L}_{\mathcal{R}}(C)| = k-2$. Now, T_{k-2} and T'_{k-2} are two neighbour-disjoint subgraphs of T^{**} that are $(k-2)$-SRIG but not $(k-3)$-SRIG. Since T^{**} is an induced subgraph of T_C, at least one vertex of T_{k-2} and at least one vertex of T'_{k-2} are on every stab line in $\mathcal{L}_{\mathcal{R}}(C)$. This means that there exist minimal $\mathcal{L}_{\mathcal{R}}(C)$-spanning paths Z_1 in T_{k-2} and Z_2 in T'_{k-2}, which are neighbour-disjoint. Let Z be an induced path in T^{**} that connects some vertex of Z_1 and some vertex of Z_2 such that no internal vertex of Z belongs to either Z_1 or Z_2. Note that Z is a subgraph of H_{k-2} that contains b_{k-2}. Let $(C_t, C_b) = \Delta(\mathcal{R}, C, Z_1, Z_2, Z)$.

Since there is a path in T^{**} from c to a vertex in Z (in this case, b_{k-2}) that misses both Z_1 and Z_2, we know by Lemma 2.8.4 that r_c is contained in C_t or C_b (See Figure 2.8.3(c)). Suppose that r_c is contained in C_t. Note that the path Q in T between c and b_k misses Z_1, Z_2, Z and Y. As b_k lies on the path X, we know by Observation 2.8.3(iii) that r_{b_k} intersects **bottom**(B). This means that r_{b_k} contains some points from outside B, and hence some points from outside C_t. Since r_c is contained in C_t, this can only mean that there exists some vertex x in Q such that the rectangle r_x intersects the boundary of C_t. Since x has no neighbour on Z_1, Z_2 or Z, we know by Lemma 2.8.3 that r_x intersects **top**(C_t). Since $C = B_b$ is a good region, we have by Observation 2.8.3(i) that **top**$(C_t) \subseteq$ **top**(B_b), implying that r_x intersects **top**(B_b). By (++), we now have that x has a neighbour on Y, which is a contradiction to the fact that Q misses Y. This means that r_c is contained in C_b.

Now consider the path Q in T between c and b_{k-1}. It is clear that Q misses Z_1, Z_2, Z and X. As b_{k-1} lies on the path Y, we know by Observation 2.8.3(iii) that $r_{b_{k-1}}$ intersects $\mathbf{top}(C)$. This means that $r_{b_{k-1}}$ contains some points from outside C, and hence some points from outside C_b. Since r_c is contained in C_b, this can only mean that there exists some vertex x in Q such that the rectangle r_x intersects the boundary of C_b. Since x has no neighbour on Z_1, Z_2 or Z, we know by Lemma 2.8.3 that r_x intersects $\mathbf{bottom}(C_b)$. Since $C = B_b$ and $B = A_t$ are good regions, we have by Observation 2.8.3(i) that $\mathbf{bottom}(C_b) \subseteq \mathbf{bottom}(B_b) \subseteq \mathbf{bottom}(A_t)$, implying that r_x intersects $\mathbf{bottom}(A_t)$. By (+), we now have that x has a neighbour on X, which is a contradiction to the fact that Q misses X. This completes the proof. □

Proof of Theorem 2.8.2.

Let k be any integer greater than or equal to 4. For each $i \in \{k, k-1, k-2\}$, let T_i, T'_i be two rooted trees that are each isomorphic to G_i (defined in Section 2.7). From Lemma 2.7.1(i) and Lemma 2.7.1(ii) we know that T_i and T'_i are i-SRIG but not $(i-1)$-SRIG. Let $a_i = \mathit{root}(T_i)$ and $a'_i = \mathit{root}(T'_i)$. For $i \in \{k, k-1, k-2\}$, let H_i be the tree obtained by adding a new vertex b_i to the disjoint union of T_i and T'_i and connecting it to a_i and a'_i using paths of length two. Note that H_i is isomorphic to F_i (also defined in Section 2.7). Let T be the tree obtained by adding a new vertex c to the disjoint union of H_k, H_{k-1} and H_{k-2} and then connecting c to each of b_k, b_{k-1} and b_{k-2} using paths of length at least two. See Figure 2.8.4 for a schematic diagram of T. From Theorem 2.8.3, we know that T is not k-SRIG.

We now show that T is asteroidal-(non-$(k-1)$-ESRIG)-free. For the sake of contradiction, assume that there are three subtrees X_1, X_2, X_3 that are asteroidal-(non-$(k-1)$-ESRIG) in T. The following claim is

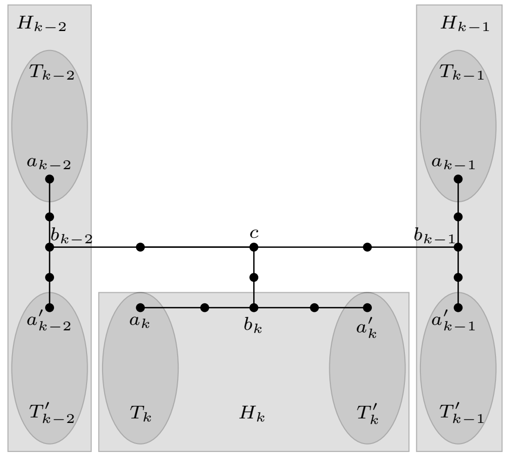


Figure 2.8.4: A schematic diagram of T. For each $i \in \{k, k-1, k-2\}$, let T_i, T'_i be two rooted trees that are each isomorphic to G_i (defined in Section 2.7) and rooted at a_i and a'_i respectively.

easy to see.

Claim. There is a vertex v in T of degree at least 3 such that $T - \{v\}$ contains three components X'_1, X'_2, X'_3 where for each $i \in \{1, 2, 3\}$, X_i is an induced subtree of $X'_i - N[v]$.

Let v be the vertex in T of degree at least 3 such that $T - \{v\}$ contains three components X'_1, X'_2, X'_3 where for each $i \in \{1, 2, 3\}$, X_i is an induced subtree of $X'_i - N[v]$. For each $i \in \{1, 2, 3\}$, since X_i is non-$(k-1)$-ESRIG, we also have that X'_i is non-$(k-1)$-ESRIG. Let us assume that v is a vertex of T_k. Note that $T - V(T_k)$ has only one component. Then by Observation 2.7.1(ii), all but one component of $T - \{v\}$ are proper subtrees of T_k. This implies that there exist distinct $X, Y \in \{X'_1, X'_2, X'_3\}$ such that X, Y are proper subtrees of T_k. Therefore, X and Y are vertex-disjoint (in fact, neighbour-disjoint) subtrees of T_k that are both non-$(k-1)$-ESRIG. But since T_k is isomorphic to G_k, this is a contradiction to Lemma 2.7.1(iv). Hence, v is not a vertex of T_k and for similar reasons, v is not a vertex of T'_k. Let $T^* = T - (V(H_k) \cup N[b_k])$.

Claim. The tree T^* is $(k-1)$-ESRIG.

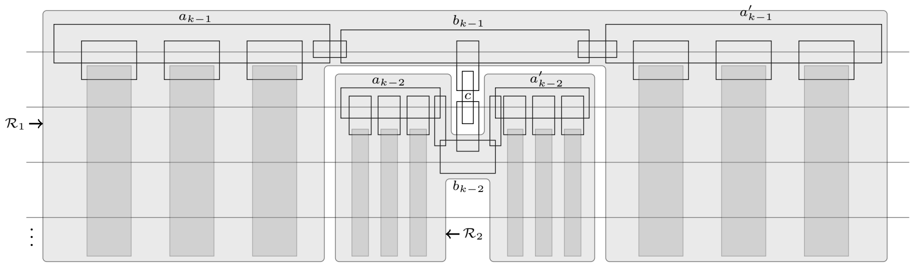


Figure 2.8.5: A schematic diagram of the $(k-1)$-stabbed rectangle intersection representation $\mathcal{R}$ of T^*.

Proof. From the definition of T, we know that T^* is the union of H_{k-1}, H_{k-2} and the path in T between b_{k-1} and b_{k-2} (which contains the vertex c). Recall that H_{k-1} is obtained by adding a new vertex b_{k-1} to the disjoint union of T_{k-1} and T'_{k-1} and connecting their roots (i.e. a_{k-1} and a'_{k-1} respectively) to b_{k-1} using paths of length two. Therefore, H_{k-1} is isomorphic to F_{k-1}. Let $\mathcal{R}_1$ be the $(k-1)$-exactly stabbed rectangle intersection representation of H_{k-1} that is given by Lemma 2.7.1(iii)(a). Similarly, H_{k-2} is isomorphic to F_{k-2}. Let $\mathcal{R}_2$ be the $(k-2)$-exactly stabbed rectangle intersection representation of H_{k-2} that is given by Lemma 2.7.1(iii)(b), in which the only vertices on the top stab line are those in $N[a_{k-2}] \cup N[a'_{k-2}]$. It can now be seen that the two representations $\mathcal{R}_1$ and $\mathcal{R}_2$ can be combined as shown in Figure 2.8.5 to obtain a $(k-1)$-exactly stabbed rectangle intersection representation $\mathcal{R}$ of $T^*[V(H_{k-1}) \cup V(H_{k-2})]$ that satisfies the following properties: (i) all vertices of the path between a_{k-1} and a'_{k-1} are on the top stab line of $\mathcal{R}$, (ii) a vertex $u \in V(H_{k-2})$ is on the stab line just below the top stab line of $\mathcal{R}$ if and only if $u \in N[a_{k-2}] \cup N[a'_{k-2}]$, and (iii) for any vertex $u \in V(H_{k-2})$, we have that $span(u) \subset span(b_{k-1})$. We leave it to the reader to verify that $\mathcal{R}$ can be extended to a $(k-1)$-exactly stabbed rectangle intersection representation of T^* by adding the rectangles corresponding to the three vertices in the path between b_{k-1} and b_{k-2} (refer to Figure 2.8.5).

Therefore we conclude that T^* is $(k-1)$-ESRIG.

Now suppose v is a vertex of H_k. Since we have already concluded that $v \notin V(T_k) \cup V(T'_k)$, we can infer that v must be the vertex b_k. Recalling the definition of T, we can infer that $T - \{b_k\}$ has exactly three components and since $b_k = v$ we know that they are X'_1, X'_2, X'_3. Also from the definition of T, it follows that there exists $i \in \{1, 2, 3\}$ such that $X'_i = T - V(H_k)$. We know from the definition of v that X_i is a subtree of $X'_i - N[v] = T^*$. But then by the above claim, we have that X_i is $(k-1)$-ESRIG, which contradicts the fact that X_i is non-$(k-1)$-ESRIG.

From the above arguments, we infer that v must lie in the tree $T - V(H_k)$. Since v has degree at least 3, we can infer from the construction of T that $v \in V(T^*)$. Notice that $T - V(T^*)$ has only one component. Then by Observation 2.7.1(ii), all but one component of $T - \{v\}$ are proper subtrees of T^*. This implies that there is a component $X \in \{X'_1, X'_2, X'_3\}$ such that X is a proper subtree of T^*. But by the above claim, we now have that X is $(k-1)$-ESRIG, contradicting our earlier observation that X'_1, X'_2, X'_3 are all non-$(k-1)$-ESRIG. This completes the proof. □

2.9 Trees that are k-SRIG but not k-ESRIG

We define the tree D_l, for $l > 1$, as follows. Let $T_1, T_2, \ldots, T_7$ be seven rooted trees, each isomorphic to G_{l-1}. Take a $K_{1,7}$ with vertex set $\{u, u_1, u_2, \ldots, u_7\}$, where $u_1, u_2, \ldots, u_7$ are the leaves, and add edges between u_i and $root(T_i)$ for each $i \in \{1, 2, \ldots, 7\}$. The resulting graph is D_l and we let $root(D_l) = u$.

Lemma 2.9.1. *Let $l > 1$.*

(i) D_l is not $(l-1)$-SRIG.

(ii) There is an l-exactly stabbed rectangle intersection representation $\mathcal{R}$ of D_l such that for $v, w \in V(D_l)$, $span(v) \subseteq span(w)$ if w is an

ancestor of v and the rectangles intersecting the top stab line of $\mathcal{R}$ are exactly the vertices in $N[root(D_l)]$.

(iii) Let T and T' be two trees each isomorphic to D_l. Let J_l be the tree obtained by taking a new vertex u and joining it to the root vertices of T, T' using paths of length two.

(a) There is an l-exactly stabbed rectangle intersection representation $\mathcal{R}'$ of J_l such that for $v, w \in V(J_l)$, $span(v) \subseteq span(w)$ if w is an ancestor of v in T or T', and all vertices in the path between $root(T)$ and $root(T')$ are on the top stab line of $\mathcal{R}'$.

(b) If $l \geq 6$, then in any l-exactly stabbed rectangle intersection representation of J_l, $root(T)$ and $root(T')$ are either both on the top stab line or both on the bottom stab line.

(iv) In any l-exactly stabbed rectangle intersection representation $\mathcal{R}$ of D_l, $root(D_l)$ is on the top or bottom stab line of $\mathcal{R}$.

Proof. For (i), it is easy to see that G_l is an induced subgraph of D_l, and therefore by Lemma 2.7.1(i), D_l is not $(l-1)$-SRIG. It is also easy to see that the constructions in the proofs of Lemma 2.7.1(ii) and Lemma 2.7.1(iii)(a) can be easily extended to prove (ii) and (iii)(a) respectively.

We shall now prove (iv). Suppose for the sake of contradiction that there exists an l-exactly stabbed rectangle intersection representation $\mathcal{R}$ of D_l in which $root(D_l)$ is not on the top or bottom stab lines. Recall that D_l is constructed by taking a $K_{1,7}$ with vertex set $\{u, u_1, u_2, \ldots, u_7\}$ with leaves $u_1, u_2, \ldots, u_7$ and making each u_i adjacent to the root of a tree T_i that is isomorphic to G_{l-1}. For each $i \in \{1, 2, \ldots, 7\}$, let $T'_i = D_l[\{u_i\} \cup V(T_i)]$. Suppose that there exists $I \subseteq \{1, 2, \ldots, 7\}$ with $|I| = 3$ such that for each $i \in I$, there is no vertex in T'_i that is on the top stab line. Then, since $u = root(D_l)$ is not on the top stab line, the rectangles corresponding to the vertices of $\{u\} \cup \bigcup_{i \in I} V(T'_i)$ form an $(l-1)$-(exactly) stabbed rectangle intersection representation of a tree isomorphic to G_l. This contradicts Lemma 2.7.1(i). Therefore, there are at most two trees in $\{T_1', T_2', \ldots, T_7'\}$ such that none of their vertices are on the top stab line. In similar fashion, we can conclude that there are at most two trees in $\{T_1', T_2', \ldots, T_7'\}$ such that none of their vertices are on the bottom stab line. This means that there are at least three trees in $\{T_1', T_2', \ldots, T_7'\}$, say T_1', T_2', T_3', such that $|\mathcal{L}_\mathcal{R}(T_1')| = |\mathcal{L}_\mathcal{R}(T_2')| = |\mathcal{L}_\mathcal{R}(T_3')| = l$. For $i \in \{1, 2, 3\}$, let P_i be an $\mathcal{L}_\mathcal{R}(T_i')$-spanning induced path in T_i' starting at a vertex x_i that is on the top stab line and ending at a vertex y_i that is on the bottom stab line. Let $\mathbf{p_i}$ be a rectilinear curve through P_i starting at some point on the top stab line in r_{x_i} and ending at some point on the bottom stab line in r_{y_i}. As T_1', T_2', T_3' are pairwise neighbour-disjoint, we know that P_1, P_2, P_3 are also pairwise neighbour-disjoint, implying that the curves $\mathbf{p_1}, \mathbf{p_2}, \mathbf{p_3}$ are pairwise disjoint. Therefore one of the curves, say $\mathbf{p_2}$, is between the other two. Then, it is easy to see that any path between a vertex of T_1' and a vertex of T_3' contains a vertex whose rectangle intersects $\mathbf{p_2}$, which means that this vertex has a neighbour on P_2. Now consider the path u_1uu_3. As the only vertex on this path that has a neighbour in $V(T_2)$ is $u = root(D_l)$, we can infer that r_u intersects $\mathbf{p_2}$. It follows from the definition of rectilinear curves that there is a point $q \in r_u \cap \mathbf{p_2}$ that is also on a stab line, say ℓ. As $u = root(D_l)$ is on ℓ, we can conclude that ℓ is neither the top nor the bottom stab line of $\mathcal{R}$. Since the point $q \in \mathbf{p_2}$, it belongs to the rectangle corresponding to a vertex on P_2 that intersects r_u. Note that if u has a neighbour on P_2, then it has to be u_2. This lets us conclude that u_2 is on P and also that $q \in r_{u_2}$, which implies that u_2 is on ℓ. As $\mathcal{R}$ is an l-exactly stabbed rectangle intersection representation, we infer that u_2 is neither on the top nor the bottom stab line. Then, $u_2 \notin \{x_i, y_i\}$. But this means that $x_i, y_i \in V(T_i)$, implying that the path P_2 does not contain u_2. This contradicts our earlier observation that u_2 is on P_2.

It only remains to prove (iii)(b). Let $l \geq 6$ and let $\mathcal{R}$ be any l-exactly stabbed rectangle intersection representation of J_l. Let ℓ, ℓ' be the stab lines that intersect $r_{root(T)}$ and $r_{root(T')}$ respectively. By (iv), we know that each of ℓ, ℓ' is either the top stab line or the bottom stab line. Since there is a path of length 4 between $root(T)$ and $root(T')$ in J_l, we can infer that ℓ and ℓ' have no more than 3 stab lines between them. Since $l \geq 6$, this means that it is not possible that one of ℓ, ℓ' is the top stab line and the other the bottom stab line. So ℓ, ℓ' are either both the top stab line or both the bottom stab line. □

Lemma 2.9.2. *Let $\mathcal{R}$ be a k-exactly stabbed rectangle intersection representation of a graph G and let R be a good region in this representation. Let P_1 and P_2 be minimal $\mathcal{L}_{\mathcal{R}}(R)$-spanning paths in G_R that are neighbour-disjoint and let P be an induced path in G_R between some vertex in $V(P_1)$ and some vertex in $V(P_2)$ such that no internal vertex of P is on P_1 or P_2. Let $(R_t, R_b) = \Delta(\mathcal{R}, R, P_1, P_2, P)$. Suppose that there are two nonadjacent vertices $x_1, x_2 \in V(P)$ that are on the top (bottom) stab line in $\mathcal{L}_{\mathcal{R}}(R)$ such that the subpath P' of P between x_1 and x_2 has length at most d, for some $d \geq 2$. Then there are no connected induced subgraph H of G_{R_t} (G_{R_b}) which is neighbour-disjoint from P_1, P_2, P and satisfies the following properties:*

(i) $|\mathcal{L}_{\mathcal{R}}(H)| > \lceil \frac{d-1}{2} \rceil$*, and*

(ii) *H contains a vertex c such that there exists a path in G_R from c to some vertex in P' that misses x_1, x_2, P_1, P_2 and $P - V(P')$.*

Proof. We shall prove the lemma only for the case when x_1 and x_2 are on the top stab line in $\mathcal{L}_{\mathcal{R}}(R)$, as the other case can be proved in similar fashion. Suppose there exists a connected component H of G_{R_t} that is neighbour-disjoint from P_1, P_2 and P such that $|\mathcal{L}_{\mathcal{R}}(H)| > \lceil \frac{d-1}{2} \rceil$, and there exists $c \in V(H)$ from which there is a path Q in G_R to some vertex in P' that misses x_1, x_2, P_1, P_2 and $P - V(P')$. For $i \in \{1, 2\}$, let u_i, v_i be the endvertices of P_i on the top and bottom stab lines in $\mathcal{L}_{\mathcal{R}}(R)$

respectively, and let $V(P_i) \cap V(P) = \{w_i\}$. Let us assume without loss of generality that x_1 appears before x_2 when traversing the path P from w_1 to w_2. For $i \in \{1, 2\}$, define P'_i to be the path obtained by the union of the subpath of P_i between v_i and w_i and the subpath of P between w_i and x_i. It is clear that P'_1 and P'_2 are neighbour-disjoint $\mathcal{L}_\mathcal{R}(R)$-spanning paths in G_R and that P' is an induced path in G_R between a vertex in $V(P'_1)$ and a vertex in $V(P'_2)$ none of whose internal vertices are on either P'_1 or P'_2. Let $(R'_t, R'_b) = \Delta(\mathcal{R}, R, P'_1, P'_2, P')$. As $\mathcal{R}$ is a k-exactly stabbed rectangle intersection representation and P' has length d, it follows that $|\mathcal{L}_\mathcal{R}(R'_t)| \leq \lceil \frac{d-1}{2} \rceil$.

Since P' misses c and there is the path Q in G_R between c and a vertex of P' that misses both P'_1 and P'_2, we can apply Lemma 2.8.4 to conclude that r_c is contained in R'_t or R'_b. It is easy to see that for any vertex z that misses P_1, P_2 and P, the rectangle r_z is contained in R'_b if and only if it is contained in R_b. As we know that $c \in V(G_{R_t})$, which implies that r_c is contained in R_t and therefore not in R_b, we can now conclude that r_c is contained in R'_t. Since H is neighbour-disjoint from P_1, P_2 and P, it is also neighbour-disjoint from P'_1, P'_2 and P'. As H is connected, this means that there is a path in G_R from c to each vertex of H that misses P'_1, P'_2 and P'. By Lemma 2.8.5, we now have that H is an induced subgraph of $G_{R'_t}$. This means that $|\mathcal{L}_\mathcal{R}(R'_t)| > \lceil \frac{d-1}{2} \rceil$, contradicting our earlier observation. □

Theorem 2.9.1. *For every $k \geq 10$, there is a tree which is k-SRIG but not k-ESRIG.*

Proof. Let k be any integer greater than or equal to 10. For each $i \in \{k, k-1\}$, let T_i, T'_i be two rooted trees that are each isomorphic to D_i and let T_{k-2} be a tree isomorphic to D_{k-2}. From Lemma 2.9.1(i) and Lemma 2.9.1(ii), we know that for $i \in \{k, k-1\}$, T_i and T'_i are i-SRIG but not $(i-1)$-SRIG. For $i \in \{k, k-1\}$, let $a_i = root(T_i)$ and $a'_i = root(T'_i)$. Further, let H_i be the tree obtained by adding a new vertex b_i to the

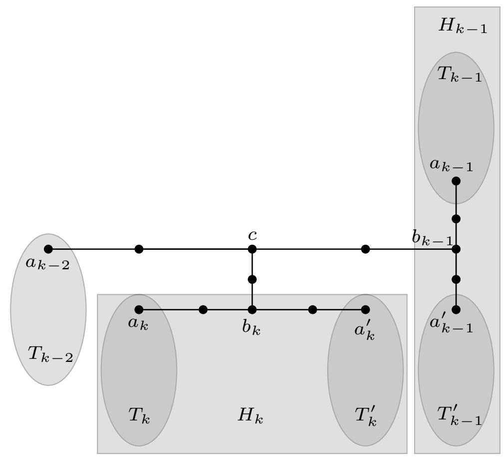


Figure 2.9.1: A schematic diagram of T. For each $i \in \{k, k-1\}$, let T_i, T'_i two rooted trees that are each isomorphic to D_i and rooted at a_i and a_i respectively. T_{k-2} is isomorphic to D_{k-2} and is rooted at a_{k-2}.

disjoint union of T_i and T'_i and connecting it to a_i and a'_i using paths of length two. Let $a_{k-2} = \mathit{root}(T_{k-2})$. Let T be the tree obtained by adding a new vertex c to the disjoint union of H_k, H_{k-1} and T_{k-2} and then connecting c to each of b_k, b_{k-1} and a_{k-2} using paths of length two. See Figure 2.9.1 for a schematic diagram of T. We claim that T is k-SRIG but not k-ESRIG.

We will first show that T is k-SRIG. Let $\ell_1, \ell_2, \ldots, \ell_k$ be k horizontal lines, ordered from bottom to top. Since H_k is isomorphic to J_k, we know from Lemma 2.9.1(iii)(a) that there is a k-(exactly) stabbed rectangle intersection representation $\mathcal{R}_1$ of H_k using stab lines $\ell_1, \ell_2, \ldots, \ell_k$ such that for $v, w \in V(H_k)$, $\mathit{span}(v) \subseteq \mathit{span}(w)$ if w is an ancestor of v in T_k or T'_k, and all vertices in the path in T between a_k and a'_k are on the bottom stab line ℓ_1. Similarly, there is a $(k-1)$-(exactly) stabbed rectangle intersection representation $\mathcal{R}_2$ of H_{k-1} using stab lines $\ell_2, \ell_3, \ldots, \ell_k$ such that for $v, w \in V(H_{k-1})$, $\mathit{span}(v) \subseteq \mathit{span}(w)$ if w is an ancestor of v in T_{k-1} or T'_{k-1}, and all vertices in the path in T between a_{k-1} and a'_{k-1} are on the top stab line ℓ_k. By Lemma 2.9.1(ii), there exists a $(k-2)$-(exactly)

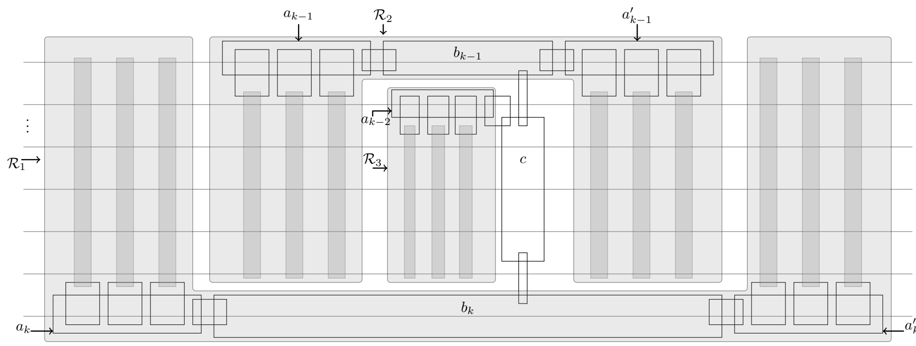


Figure 2.9.2: A schematic diagram of a k-stabbed rectangle intersection representation of T.

stabbed rectangle intersection representation $\mathcal{R}_3$ of T_{k-2} using stab lines $\ell_2, \ell_3, \ldots, \ell_{k-1}$ such that for $v, w \in V(T_{k-2})$, $span(v) \subseteq span(w)$ if w is an ancestor of v in T_{k-2}, and the only vertices in T_{k-2} that are on the stab line ℓ_{k-2} are the ones in $N[a_{k-2}]$. It can be seen as shown in Figure 2.9.2 that $\mathcal{R}_1$, $\mathcal{R}_2$ and $\mathcal{R}_3$ can be combined and rectangles for the vertices in $N[c]$ can be added to obtain a k-stabbed rectangle intersection representation of T in which for any $x \in V(H_{k-1})$, $span(x) \subseteq span(b_k)$ and for any $x \in V(T_{k-2})$, $span(x) \subseteq span(b_{k-1})$.

Suppose for the sake of contradiction that T is k-ESRIG. This part of the proof proceeds very similarly to the proof of Theorem 2.8.3. As in that proof, we let $\mathcal{R}$ be a $(k+2)$-exactly stabbed rectangle intersection representation of T in which the top and bottom stab lines do not intersect any rectangle and let A be a good region that contains all the rectangles of $\mathcal{R}$. As T_k and T'_k are k-SRIG but not $(k-1)$-SRIG, we have $|\mathcal{L}_{\mathcal{R}}(A)| = k$ and there are $\mathcal{L}_{\mathcal{R}}(A)$-spanning paths in both T_k and T'_k. Let X_1 and X_2 be minimal $\mathcal{L}_{\mathcal{R}}(A)$-spanning paths in T_k and T'_k respectively. Let X be an induced path in T that connects some vertex of X_1 and some vertex of X_2 such that no internal vertex of X belongs to either X_1 or X_2. Note that X is a subgraph of H_k that contains b_k. Let $(A_t, A_b) = \Delta(\mathcal{R}, A, X_1, X_2, X)$.

Since there is a path in $T_A = T$ from $c \in V(T_A)$ to a vertex in X (in this case, b_k) that misses both X_1 and X_2, we know by Lemma 2.8.4 that r_c is contained in A_t or A_b. We shall assume without loss of generality that r_c is contained in A_t. Let $T^* = T - (V(H_k) \cup N[b_k])$. Since there is a path in T_A from c to each vertex of T^* that misses X_1, X_2 and X, we can use Lemma 2.8.5 to infer that T^* is a connected induced subgraph of T_{A_t}.

Claim. *Both the vertices a_k and a'_k are on the bottom stab line in $\mathcal{L}_{\mathcal{R}}(A)$.*

Proof. Let X' be the path in T_A between a_k and a'_k. Clearly, X' has length 4 and is a subpath of X. The tree T^* contains T_{k-1} as an induced subgraph, and is therefore not $(k-2)$-SRIG by Lemma 2.9.1(i). Hence, $|\mathcal{L}_{\mathcal{R}}(T^*)| \geq k-1$. Since T^* contains the vertex c that has a path to a vertex in X' which misses a_k, a'_k, X_1, X_2 and $X - V(X')$, we can use Lemma 2.9.2 to infer that at least one of a_k and a'_k is not on the top stab line in $\mathcal{L}_{\mathcal{R}}(A)$. Notice that the graph induced by $V(T_k) \cup V(T'_k) \cup V(X')$ in $T = T_A$ is isomorphic to J_k. This means that there is a k-exactly stabbed rectangle intersection representation of J_k contained in the region A. Using Lemma 2.9.1(iii)(b), we can now conclude that both a_k and a'_k are on the bottom stab line in $\mathcal{L}_{\mathcal{R}}(A)$. This completes the proof.

From here onwards, we shall let $B = A_t$, for ease of notation. From the above arguments, we know that $|\mathcal{L}_{\mathcal{R}}(T^*)| \geq k-1$ and T^* is a connected induced subgraph of T_B. Therefore, $|\mathcal{L}_{\mathcal{R}}(B)| \geq k-1$. By Lemma 2.8.2, this means that B is a good region and by Lemma 2.8.1(b), we can conclude that $|\mathcal{L}_{\mathcal{R}}(B)| = k-1$. Now, T_{k-1} and T'_{k-1} are two neighbour-disjoint subtrees of T^* that are $(k-1)$-SRIG but not $(k-2)$-SRIG. This means that there exist minimal $\mathcal{L}_{\mathcal{R}}(B)$-spanning induced paths Y_1 in T_{k-1} and Y_2 in T'_{k-1}. Let Y be an induced path in T^* that connects some vertex of Y_1 and some vertex of Y_2 such that no internal vertex of Y belongs to either Y_1 or Y_2. Note that Y is a subgraph of H_{k-1} that contains b_{k-1}. Let $(B_t, B_b) = \Delta(\mathcal{R}, B, Y_1, Y_2, Y)$.

Since there is a path in T^* from c to a vertex in Y (in this case, b_{k-1})

that misses both Y_1 and Y_2, we know by Lemma 2.8.4 that r_c is contained in B_t or B_b. As explained in the proof of Theorem 2.8.3, it can be shown that r_c is contained in B_b (if r_c is contained in B_t, then there could not have been a path in T between c and the vertex b_k in X that misses Y_1, Y_2 and Y). Let $T^{**} = T^* - (V(H_{k-1}) \cup N[b_{k-1}])$. Since there is a path in T_B from c to each vertex of T^{**} that misses Y_1, Y_2 and Y, we can use Lemma 2.8.5 to infer that T^{**} is a connected induced subgraph of T_{B_b}.

Claim. Both the vertices a_{k-1} and a'_{k-1} are on the top stab line in $\mathcal{L}_{\mathcal{R}}(A)$.

Proof. Let Y' be the path in T between a_{k-1} and a'_{k-1}. Clearly, Y' has length 4 and is a subpath of Y. The tree T^{**} contains T_{k-2} as an induced subgraph, and is therefore not $(k-3)$-SRIG, implying that $|\mathcal{L}_{\mathcal{R}}(T^{**})| \geq k-2$. Since T^{**} contains the vertex c that has a path to a vertex in Y' which misses a_{k-1}, a'_{k-1}, Y_1, Y_2 and $Y - V(Y')$, we can use Lemma 2.9.2 to infer that at least one of a_{k-1} and a'_{k-1} is not on the bottom stab line in $\mathcal{L}_{\mathcal{R}}(B)$. Notice that the graph induced by $V(T_{k-1}) \cup V(T'_{k-1}) \cup V(Y')$ in T^* is isomorphic to J_{k-1}. This means that there is a $(k-1)$-exactly stabbed rectangle intersection representation contained in the region B. Using Lemma 2.9.1(iii)(b), we can now conclude that both a_{k-1} and a'_{k-1} are on the top stab line in $\mathcal{L}_{\mathcal{R}}(B)$. Now since $B = A_t$ and $|\mathcal{L}_{\mathcal{R}}(B)| = |\mathcal{L}_{\mathcal{R}}(A)| - 1$, we know by Lemma 2.8.1(b) that the top stab line in $\mathcal{L}_{\mathcal{R}}(B)$ is also the top stab line in $\mathcal{L}_{\mathcal{R}}(A)$. This completes the proof of the claim.

Let $\ell_1, \ell_2, \ldots, \ell_k$ be the stab lines in $\mathcal{L}_{\mathcal{R}}(A)$ in order from bottom to top. Now, the fact that each rectangle in $\mathcal{R}$ intersects exactly one stab line gives us several observations. Since there is a path of length 2 between b_k and a_k in T, and because our first claim tells us that a_k is on ℓ_1, we can conclude that b_k is not on any of the stab lines in $\{\ell_4, \ell_5, \ldots, \ell_k\}$. Similarly, our second claim tells us that a_{k-1} is on ℓ_k, and then the fact that there is a path of length 2 between a_{k-1} and b_{k-1} implies that b_{k-1} cannot be on any stab line in $\{\ell_{k-3}, \ell_{k-4}, \ldots, \ell_2, \ell_1\}$. Now, since there is a path of length 4 between b_k and b_{k-1}, there can be at most 3 stab lines between ℓ_3 and ℓ_{k-2}. But this contradicts the fact that $k \geq 10$. □

2.10 Concluding remarks and open problems

In this chapter, we introduced the stab number of rectangle intersection graphs. We constructed polynomial-time algorithm to check if $stab(G) \leq 2$ for any block graph G. However, the structure of 2-SRIG remains a mystery.

Therefore, the direction of further research could be to investigate the subclasses of 2-SRIGs and try to characterise these classes of graphs.

Question 2.10.1. *Develop a forbidden structure characterization and/or a polynomial-time recognition algorithm for 2-SRIG.*

Note that Theorem 2.6.1 gives a characterisation of the 2-SRIGs within the class of block graphs. This theorem shows that within the class of block graphs, those graphs that do not contain asteroidal-(non-interval) subgraphs are exactly the 2-SRIGs. From the characterisation of interval graphs by Boland and Lekkerkerker (Theorem 1.1.1), we know that the absence of asteroidal triples characterises the 1-SRIGs within chordal graphs. Therefore, a natural question is whether the absence of asteroidal-(non-interval) subgraphs is enough to characterise the 2-SRIGs within chordal graphs (note that block graphs are a subclass of chordal graphs). The answer to this question is negative, as we have shown in Theorem 2.4.3 that there are split graphs that are not 2-SRIG. Split graphs are chordal and clearly, no split graph can contain asteroidal-(non-interval) subgraphs, as for any three connected induced subgraphs that are pairwise neighbour-disjoint in a split graph, at least two of them will contain just one vertex each. This gives rise to the following question.

Question 2.10.2. *Find a forbidden structure characterisation for chordal graphs (resp. split graphs) that are 2-SRIG. Can chordal graphs* *(resp. split graphs) that are 2-SRIG be recognised in polynomial-time?*

We have shown that any split graph with boxicity at most 2 is 3-SRIG and that there exists a split graph which is 3-SRIG but not 2-SRIG. Therefore, the following question is interesting.

Question 2.10.3. *What is the complexity of recognising split graphs that are 3-SRIG?*

Note that by Theorem 2.4.2, the above problem is equivalent to the problem of recognising split graphs that have boxicity at most 2. This problem assumes significance because recognising split graphs that have boxicity at most 3 is NP-complete [4].

We constructed polynomial-time algorithm to check if $stab(T) \leq 3$ for any tree T. Therefore, the following are essential questions in this direction.

Question 2.10.4. *For a given block graph G, is it possible to determine $stab(G)$ in polynomial-time?*

Question 2.10.5. *For a given tree T, is it possible to determine $stab(T)$ in polynomial-time?*

We showed that $K_{4,4}$ is not k-ESRIG for any finite k, but is 4-SRIG. Here, the question arises as to how high the exact stab number of an exactly stabbable graph can be with respect to its stab number. Theorem 2.4.4 shows that trees are exactly stabbable and Theorem 2.9.1 shows a tree T such that $estab(T) > stab(T)$ (in fact, it is an easy exercise to show that $estab(T) = stab(T)+1$). The following questions are, therefore of interest.

Question 2.10.6. *Is there a constant c such that for any tree T we have, $estab(T) - stab(T) \leq c$ or $\frac{estab(T)}{stab(T)} \leq c$?*

Question 2.10.7. *For a given tree T, is it possible to determine $estab(T)$ in polynomial-time?*

3

Rectangle intersection graphs with stab number at most 2

Contents

In this chapter, we focus on rectangle intersection graphs of stab number at most two and its subclasses. First, we recall and introduce some definitions. Let G be a 2-SRIG with a 2-stabbed rectangle intersection representation $\mathcal{R}$ in which the stab lines are $y = a_1$ and $y = a_2$ where $a_1 < a_2$. The *top* (resp., *bottom*) stab line of $\mathcal{R}$ is the stab line $y = a_2$ (resp., $y = a_1$). A vertex $u \in V(G)$ is "on" a stab line if r_u intersects that stab line. Note that the vertices on the top (resp., bottom) stab line induce an interval subgraph of G, and the projection of their corresponding rectangles on the X-axis provide an interval representation $\mathcal{R}_t$ (resp., $\mathcal{R}_b$) of this subgraph. Now by putting restrictions on $\mathcal{R}_t$ and $\mathcal{R}_b$, we have several subclasses of 2-SRIG as defined below.

Given a geometric intersection representation $\mathcal{R}$ of a graph, the notation $\mathcal{R} \in \mathcal{I}$ means that $\mathcal{R}$ is an interval representation. Similarly, the notation $\mathcal{R} \in \mathcal{P}$ means that $\mathcal{R}$ is an interval representation where no interval is a proper subset of another, $\mathcal{R} \in \mathcal{E}$ means that $\mathcal{R}$ is an interval representation where the intervals have equal lengths, $\mathcal{R} \in \mathcal{U}$ means that $\mathcal{R}$ is an interval representation where the intervals have unit lengths. Note that $\mathcal{R} \in \mathcal{E}$ and $\mathcal{R} \in \mathcal{U}$ are equivalent notions up to scaling for interval graphs [24]. Nevertheless, the above distinction is needed for the definitions that follow.

If $\mathcal{R}$ is a 2-stabbed rectangle intersection representation of G with $\mathcal{R}_t \in \mathcal{X}$ and $\mathcal{R}_b \in \mathcal{Y}$, then $\mathcal{R}$ is said to be an $(\mathcal{X}, \mathcal{Y})$-representation of G, where $\mathcal{X}, \mathcal{Y} \in \{\mathcal{I}, \mathcal{P}, \mathcal{E}, \mathcal{U}\}$. Moreover, G is an $(\mathcal{X}, \mathcal{Y})$-graph if it admits an $(\mathcal{X}, \mathcal{Y})$-representation. Note that the class of $(\mathcal{I}, \mathcal{I})$-graphs is the same as 2-SRIG, while the classes $(\mathcal{I}, \mathcal{P})$, $(\mathcal{I}, \mathcal{E})$, $(\mathcal{I}, \mathcal{U})$, $(\mathcal{P}, \mathcal{P})$, $(\mathcal{P}, \mathcal{E})$, $(\mathcal{P}, \mathcal{U})$, $(\mathcal{E}, \mathcal{E})$, $(\mathcal{E}, \mathcal{U})$, and $(\mathcal{U}, \mathcal{U})$-graphs are all subclasses of 2-SRIG. A note of caution in this context is the following: in an $(\mathcal{E}, \mathcal{E})$-representation $\mathcal{R}$ of G, the lengths of the intervals of $\mathcal{R}_t$ may not be equal to the lengths of the intervals of $\mathcal{R}_b$. Furthermore, a graph G is a 2-*stabbable unit square intersection graph* or 2-SUIG, if G has a 2-stabbed rectangle intersection representation $\mathcal{R}$ in which all rectangles are unit squares.

3.1 Chapter overview

In Section 3.2, we study the containment relationship among the graph classes defined above. In Section 3.3, we prove that given a triangle-free graph it is possible to figure out if it is a $(\mathcal{P}, \mathcal{P})$-graph or not in linear time.

In Section 3.4, we prove that triangle-free 2-SRIGs are 3-colorable. Moreover, we show that For every natural number c, there exists a polynomial-time algorithm that decides whether an input 2-SRIG graph is c-colorable.

The Chromatic Number problem is to decide, on given a graph G and an integer c as input, whether G is c-colorable. In Section 3.5, we prove that the problem of finding the chromatic number of a 2-SRIG graph G, that is, deciding if G is c-colorable when c is part of the input, remains NP-Hard, even if a 2-stabbed rectangle intersection representation of G is available. (In contrast, 1-SRIGs are nothing but the interval graphs for which the Chromatic Number problem can be solved in polynomial time.) This is a strengthening of a result of Imai and Asano [102], who proved that the Chromatic Number problem is NP-complete for rectangle intersection graphs, a superclass of 2-SRIGs. We show that the Chromatic Number problem is NP-complete for a subclass of 2-SRIGs known as *2-row* B_0*-VPGs*.

A B_0*-VPG graph* is an intersection graph of vertical and horizontal line segments on the plane. Asinowski, Cohen, Golumbic and Limouzy [9] proved that the Chromatic Number problem is NP-complete for B_0-VPG graphs. Chaplick, Cohen and Stacho [53] claimed in the conclusion of their work that the Chromatic Number problem is NP-complete even for the subclass of B_0-VPG graphs where the horizontal line segments are contained in two horizontal reference lines. Observe that these graphs, called “2-row B_0-VPG graphs”, form a subclass of 2-SRIGs (the horizontal reference lines are the stab lines in our definition). Note that

1-row B_0-VPG graphs are the same as interval graphs. As a proof for the claim in [53] does not seem to have been published, in Section 3.5, we prove that deciding if a 2-row B_0-VPG graph G is c-colorable when c is part of the input, remains NP-Hard. Finally, we draw conclusions in Section 3.6.

3.2 Containment relationship among subclasses of 2-SRIG

In this section, we prove the following theorem.

Theorem 3.2.1. *2-SUIG = $(\mathcal{U},\mathcal{U})$-graphs $\subset$ $(\mathcal{E},\mathcal{U})$-graphs = $(\mathcal{E},\mathcal{E})$-graphs $\subset$ $(\mathcal{P},\mathcal{U})$-graphs = $(\mathcal{P},\mathcal{E})$-graphs = $(\mathcal{P},\mathcal{P})$-graphs $\subset$ $(\mathcal{I},\mathcal{U})$-graphs = $(\mathcal{I},\mathcal{E})$-graphs = $(\mathcal{I},\mathcal{P})$-graphs $\subset$ 2-ESRIG = 2-SRIG.*

We first note that 2-SRIG = 2-ESRIG. This is proved in [42], but we give a proof below for the sake of completeness.

Lemma 3.2.1. *2-SRIG = 2-ESRIG.*

Proof. Let G be a 2-SRIG with a 2-stabbed rectangle intersection representation $\mathcal{R}$ which is not a 2-exactly stabbed rectangle intersection representation. Thus some of the rectangles, say, $r_{u_1}, r_{u_2}, \ldots, r_{u_s}$ intersect both the top and bottom stab lines. Furthermore, let $r_{v_1}, r_{v_2}, \ldots, r_{v_t}$ be the rectangles that intersect only the bottom stab $y = a_1$. Now modify the representation $\mathcal{R}$ by replacing the rectangles

$$r_{u_i} = [x_{u_i}^-, x_{u_i}^+] \times [y_{u_i}^-, y_{u_i}^+] \text{ by } r'_{u_i} = [x_{u_i}^-, x_{u_i}^+] \times [a_1, y_{u_i}^+]$$

for all $i \in \{1, 2, \ldots, s\}$ and by replacing the rectangles

$$r_{v_j} = [x_{v_j}^-, x_{v_j}^+] \times [y_{v_j}^-, y_{v_j}^+] \text{ by } r'_{v_j} = [x_{v_j}^-, x_{v_j}^+] \times [a_1 - 2, y_{v_j}^+]$$

for all $j \in \{1, 2, \ldots, t\}$. Now replace the bottom stab line by $y = a_1 - 1$ to obtain a 2-exactly stabbed rectangle intersection representation of G. $\square$

Observe that as 2-SRIG = 2-ESRIG, we have $(\mathcal{I}, \mathcal{P})$-graphs $\subseteq$ 2-ESRIG. To show that the class of $(\mathcal{I}, \mathcal{P})$-graphs is a proper subclass of 2-ESRIG, we show that the graph depicted in Figure 3.2.1(a) is a 2-ESRIG but not an $(\mathcal{I}, \mathcal{P})$-graph. The 2-exactly stabbed rectangle intersection representation of this graph shown in Figure 3.2.1(b) proves the former. For proving the latter, we need to work a bit more. First of all we need to define some notations and terminologies.

Let G be a 2-SRIG with a 2-stabbed rectangle intersection representation $\mathcal{R}$. The *span* of a vertex u is $span(u) = [x_u^-, x_u^+]$. The span of $S \subseteq V(G)$ is $span(S) = \cup_{u \in S} span(u)$. Observe that when $G[S]$, the subgraph of G induced by S, is connected, $span(S)$ is an interval. The span of an edge $uv \in E(G)$ is $span(uv) = span(u) \cap span(v)$. We write $I_1 = [a_1, b_1] < I_2 = [a_2, b_2]$ if $b_1 < a_2$. A set of vertices of G have a *common stab* if all of them are on a particular stab line. A *bridge edge* $uv \in E(G)$ in $\mathcal{R}$ is an edge such that there are two different stab lines having u on one of them and v on the other. Whenever the 2-stabbed rectangle intersection representation of a graph G under consideration is clear from the context, the terms r_u, x_u^-, x_u^+, y_u^-, y_u^+, for every vertex $u \in V(G)$ and usages such as "on a stab line", "have a common stab", "span" etc. are considered to be defined with respect to this representation.

Observation 3.2.1. *Let $\mathcal{R}$ be a 2-exactly stabbed rectangle intersection representation of a graph G. Let uv be a bridge edge in $\mathcal{R}$ and let $S = \{w \in V(G)\colon span(w) \cap span(uv) \neq \emptyset\}$. Let $a, b \in V(G)$ such that $span(a) < span(uv) < span(b)$. Then a and b are in different connected components of $G - S$.*

Proof. Suppose that a and b are in the same connected component C of $G - S$. As $span(V(C))$ is an interval that contains both $span(a)$ and

$span(b)$, it is clear that $span(V(C))$ also contains $span(uv)$. But this means that C contains some vertex w such that $span(w) \cap span(uv) \neq \emptyset$, which is a contradiction. $\square$

Note that in the above observation, if $a \notin N[u] \cup N[v]$ and $b \notin N[u] \cup N[v]$, then as $S \subseteq N[u] \cup N[v]$, we can conclude that a and b are in different connected components of $G - (N[u] \cup N[v])$. We shall use this form of Observation 3.2.1 in several places.

Observation 3.2.2. *Let $\mathcal{R}$ be a 2-exactly stabbed rectangle intersection representation of a graph G. Let $ua, uv, ab \in E(G)$ such that uv and ab are bridge edges in $\mathcal{R}$ while u and a are on the same stab line. Further, let $ub, av \notin E(G)$. Then either $span(ua) \cap span(v) = \emptyset$ or $span(ua) \cap span(b) = \emptyset$.*

Proof. Let $I = span(ua)$. Clearly, $I \subseteq span(u)$ and $I \subseteq span(a)$. Suppose for the sake of contradiction that $I \cap span(v) \neq \emptyset$ and $I \cap span(b) \neq \emptyset$. Then, $span(u) \cap span(b) \neq \emptyset$ and $span(a) \cap span(v) \neq \emptyset$. Assume by symmetry that u and a are on the bottom stab line in $\mathcal{R}$. As $uv, ab \in E(G)$, we have $y_v^- \leq y_u^+$ and $y_b^- \leq y_a^+$. If $y_u^+ \leq y_a^+$, then we have $y_v^- \leq y_a^+$. Since $span(a) \cap span(v) \neq \emptyset$, this implies that $av \in E(G)$, which is a contradiction. On the other hand, if $y_a^+ < y_u^+$, then $y_b^- < y_u^+$, and since $span(u) \cap span(b) \neq \emptyset$, this gives $ub \in E(G)$, which is again a contradiction. $\square$

Observation 3.2.3. *Let $\mathcal{R}$ be a 2-exactly stabbed rectangle intersection representation of a triangle-free graph G. Let $e_1, e_2 \in E(G)$ be two bridge edges in $\mathcal{R}$. Then, $span(e_1) \cap span(e_2) = \emptyset$.*

Proof. Let $e_1 = uv$, $e_2 = ab$, and let us assume without loss of generality that u, a are on the bottom stab line and v, b are on the top stab line. Since e_1 and e_2 are distinct edges, we shall also assume, by symmetry, that $u \neq a$. Suppose that $I = span(uv) \cap span(ab) \neq \emptyset$. Then $ua \in E(G)$ and ua is not a bridge edge. As G is triangle-free, we have $ub, av \notin E(G)$.

Further, notice that $I \subseteq span(u), span(v), span(a), span(b)$, which also implies that $I \subseteq span(ua)$. But then $I \subseteq span(ua) \cap span(v)$ and $I \subseteq span(ua) \cap span(b)$, contradicting Observation 3.2.2. □

Proposition 3.2.1. *In any 2-exactly stabbed rectangle intersection representation of a cycle of order greater than 3, there are exactly two bridge edges.*

Proof. Let G be a cycle of order greater than three having a 2-exactly stabbed rectangle intersection representation $\mathcal{R}$. Observe that, all the vertices of G cannot have a common stab as G is not an interval graph. This implies that in any 2-exactly stabbed rectangle intersection representation of G, there are at least two bridge edges. Suppose that $\mathcal{R}$ has more than two bridge edges. As G is triangle-free, we can use Observation 3.2.3 to order the bridge edges $e_1 = u_1v_1, e_2 = u_2v_2, ..., e_k = u_kv_k$ of $\mathcal{R}$ such that $span(e_1) < span(e_2) < \cdots < span(e_k)$. As $span(e_1) < span(e_2) < span(e_k)$ there exists a vertex $w_1 \in \{u_1, v_1\}$ and a vertex $w_2 \in \{u_k, v_k\}$ such that $span(w_1) < span(e_2) < span(w_k)$. We can now apply Observation 3.2.1 to conclude that w_1 and w_k are in different connected components of $G - (N[u_2] \cup N[v_2])$. But this is a contradiction as in any cycle of order greater than 3, it is not possible to remove the closed neighbourhoods of two adjacent vertices to obtain a disconnected graph. □

Let $W_{n+1,d}$ denote the class of triangle-free graphs consisting of a cycle on n vertices and a central vertex of degree d where $n \geq 4$ and $d \geq 2$. In other words, a graph in the class $W_{n+1,d}$ is obtained by taking a wheel on $n+1$ vertices and deleting $n-d$ edges incident to its central vertex in such a way that it becomes triangle-free (and the central vertex ends up having degree d).

Proposition 3.2.2. *Let $\mathcal{R}$ be any 2-exactly stabbed rectangle intersection representation of a graph $G \in W_{n+1,d}$ with central vertex v. Then the* *number of bridge edges incident to v is $d-1$. Moreover, if $d \geq 3$ and uv be an edge such that u, v have a common stab, then $span(v) \subset span(u)$.*

Proof. Consider an arbitrary 2-exactly stabbed rectangle intersection representation $\mathcal{R}$ of G. Let

$$C = w_0 w_1 \ldots w_{n-2}$$

be the cycle obtained by removing the central vertex v from G and let $u_0, u_1, \ldots, u_{d-1}$ be the neighbours of v. We shall assume without loss of generality that $u_0 = w_{j_0}$, $u_2 = w_{j_1}$, $\ldots$, $u_{d-1} = w_{j_{d-1}}$, where $j_0 < j_1 < \ldots < j_{d-1}$. For $0 \leq i < d-1$, let P_i denote the path $u_i = w_{j_i} w_{j_i+1} \ldots w_{j_{i+1}} = u_{i+1}$ and let P_{d-1} be the path $u_{d-1} = w_{j_{d-1}} w_{j_{d-1}+1} \ldots w_{n-2} w_0 w_1 \ldots w_{j_0} = u_0$. Moreover, let C_i denote the cycle $\{vu_i\} \cup P_i \cup \{u_{i+1}v\}$ (here and in the following, we consider indices of u_is, P_is and C_is to be modulo d).

Notice that each of $C, C_0, C_1, \ldots, C_{d-1}$ contain exactly two bridge edges due to Proposition 3.2.1. Let us first suppose that for some $i \in \{0, 1, \ldots, d-1\}$, the two bridge edges of C_i are both edges of P_i. Then these two edges are exactly the two bridge edges of C, implying that none of the paths P_j, for $j \neq i$, contain any bridge edges. This means that the two bridge edges of C_{i+1} are vu_i and vu_{i+1}. Thus vu_i is a third bridge edge in C_i other than the two bridge edges in P_i, which is a contradiction. So we can assume that there is at most one bridge edge in P_i, for each i.

As C has exactly two bridge edges, it follows that there are exactly two values in $\{0, 1, \ldots, d-1\}$, say t and t', such that P_t and $P_{t'}$ contain a bridge edge each. This implies that for $i \in \{0, 1, \ldots, d-1\} \setminus \{t, t'\}$, the edges vu_i and vu_{i+1} are the two bridge edges in C_i. Now if $t - t' \neq \pm 1$, then by our previous observation, both vu_t and vu_{t+1} must be bridge edges, which is a contradiction as we would then have three bridge edges in C_t. Therefore, without loss of generality we assume that $t = 0$ and

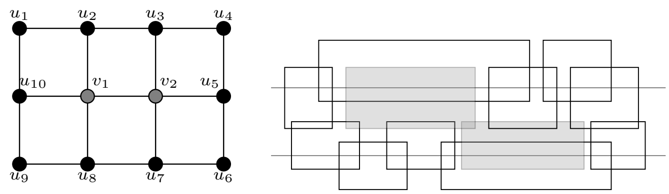
$t' = 1$. If $d = 2$, then since C_0 and C_1 both contain exactly two bridge edges, it follows that one of vu_0, vu_1 is a bridge edge and the other is not. If $d > 2$, then vu_i is a bridge edge, for all $i \neq 1$, as it belongs to some $C_i \neq C_0, C_1$. Moreover, vu_1 is not a bridge edge, as otherwise, the cycles C_0 and C_1 will have more than two bridge edges. Therefore, for any value of $d \geq 2$, all but one of the edges incident on v are bridge edges. This proves the first part of the statement of the proposition.

We now prove the next part of the statement of the proposition. When $d \geq 3$, we can assume by our previous observations that P_0, P_1 contain one bridge edge each, P_i, for $i \notin \{0, 1\}$, does not contain any bridge edge, and that vu_1 is the only edge incident on v that is not a bridge edge. We claim that $span(v) \subset span(u_1)$. Without loss of generality let v, and therefore u_1, be on the bottom stab line. Then u_0, u_2 are on the top stab line and thus $span(u_0) \cap span(u_2) = \emptyset$ as they are non-adjacent. Let us assume by symmetry that $span(u_0) < span(u_2)$. Since v is a neighbour of both u_0 and u_2, $[x^+_{u_0}, x^-_{u_2}] \subseteq span(v)$.

Notice that there are no bridge edges in the path $P = P_2 \cup P_3 \cup \cdots \cup P_{d-1}$. As $u_0, u_2 \in V(P)$, all the vertices of P are on the top stab line. Let $P' = P - \{u_0, u_2\}$. Since $span(P)$ is an interval, we have that $[x^+_{u_0}, x^-_{u_2}] \subseteq span(V(P'))$.

Let ww' and zz' be the bridge edges on P_0 and P_1 respectively, where w, z, u_0, u_2 are on the top stab line and w', z', u_1, v are on the bottom stab line. Let P'_0 be the path $P_0 - \{u_0, u_1\}$ and P'_1 the path $P_1 - \{u_1, u_2\}$. As the vertices of P'_0 are non-adjacent to the vertices in $V(P') \cup \{v\}$, we can conclude that $span(V(P'_0)) \cap [x^+_{u_0}, x^-_{u_2}] = \emptyset$. As there is a neighbour of u_0 on P'_0, $span(V(P'_0))$ intersects $span(u_0)$, leading us to the conclusion that $span(V(P'_0)) < [x^+_{u_0}, x^-_{u_2}]$. Since at least one of w, w' is on P'_0, this means that $span(ww') < [x^+_{u_0}, x^-_{u_2}]$. With the same kind of arguments, we can also deduce that $[x^+_{u_0}, x^-_{u_2}] < span(V(P'_1))$ and that $[x^+_{u_0}, x^-_{u_2}] < span(zz')$.

By Observation 3.2.3, we know that the spans of any two bridge edges of G are disjoint. Since it is clear that $x^+_{u_0} \in span(vu_0)$ and $x^-_{u_2} \in span(vu_2)$,



(a) (b)

Figure 3.2.1: A 2-exactly stabbed rectangle intersection representation of $(3,4)$-grid graph.

we now have $span(ww') < span(vu_0) < span(vu_2) < span(zz')$. As $span(ww') < span(vu_0)$, there exists a vertex $w'' \in \{w, w'\}$ such that $span(w'') < span(vu_0)$. Let $S = \{u \in V(G) \colon span(u) \cap span(vu_0) \neq \emptyset\}$. Observe that $S \subseteq N[v] \cup N[u_0]$. Now, by Observation 3.2.1, w'' and u_2 are in two connected components of $G - S$ (note that $span(vu_0) < span(u_2)$). This implies that $u_1 \in S$, as otherwise, the path between w'' and u_2 in $P_0 \cup P_1$ exists also in $G - S$. Therefore, $span(u_1) \cap span(vu_0) \neq \emptyset$.

Using the same kind of reasoning for the bridge edges vu_2 and zz', we can conclude that $span(u_1) \cap span(vu_2) \neq \emptyset$. Together, we get $[x^+_{u_0}, x^-_{u_2}] \subseteq span(u_1)$. Recall that $[x^+_{u_0}, x^-_{u_2}] \subseteq span(v)$. As u_1 and v are both on the bottom stab line and because $u_0, u_2 \in N(v) \setminus N(u_1)$, we can conclude that $y^+_{u_1} < y^+_v$. Now suppose that $x^-_v \leq x^-_{u_1}$. Let a be the neighbour of u_1 on P'_0. As $span(V(P'_0)) < [x^+_{u_0}, x^-_{u_2}]$, we have $span(a) < [x^+_{u_0}, x^-_{u_2}]$. Since r_a intersects r_{u_1}, it then follows that r_a also intersects r_v. But this contradicts the fact that a and v are nonadjacent. Therefore, we have $x^-_{u_1} < x^-_v$. Arguing symmetrically, we can show that $x^+_{u_1} > x^+_v$. Thus $span(v) \subset span(u_1)$. □

The (h, w)-*grid* is the undirected graph G with $V(G) = \{(x, y) \colon x, y \in \mathbb{Z}, 1 \leq x \leq h, 1 \leq y \leq w\}$ and $E(G) = \{(u, v)(x, y) \colon |u - x| + |v - y| = 1\}$.

Lemma 3.2.2. *The class of* $(\mathcal{I}, \mathcal{P})$*-graphs is a proper subset of* 2*-ESRIG.*

Proof. Note that an $(\mathcal{I}, \mathcal{P})$-graph is a 2-ESRIG by definition.

Let H be the $(3,4)$-grid as shown in Figure 3.2.1(a). There is a 2-exactly stabbed rectangle intersection representation of H as shown in Figure 3.2.1(b). Thus we will be done if we can show that H is not an $(\mathcal{I}, \mathcal{P})$-graph.

Suppose for the sake of contradiction that there is an $(\mathcal{I}, \mathcal{P})$-representation $\mathcal{R}$ of H. Observe that both $H - \{u_4, u_5, u_6\}$ and $H - \{u_1, u_9, u_{10}\}$ are graphs that belong to $W_{9,4}$ having central vertices v_1 and v_2 respectively. Hence, by Proposition 3.2.2, in $\mathcal{R}$, for each $i \in \{1, 2\}$, there is exactly one vertex $w_i \in N(v_i)$ that is on the same stab line as v_i and further, $span(v_i) \subset span(w_i)$. Thus, if v_1 and v_2 are on different stab lines in $\mathcal{R}$, then for some $i \in \{1, 2\}$, v_i and w_i are on the bottom stab line. As $span(v_i) \subset span(w_i)$, we have a contradiction to the fact that $\mathcal{R}_b$ is a proper set of intervals. Therefore, v_1 and v_2 have a common stab. Then by definition of w_1 and w_2, we have $w_1 = v_2$ and $w_2 = v_1$, implying that $span(v_1) \subset span(v_2) \subset span(v_1)$, which is again a contradiction. □

Lemma 3.2.3. *$(\mathcal{I}, \mathcal{U})$-graphs $=$ $(\mathcal{I}, \mathcal{E})$-graphs $=$ $(\mathcal{I}, \mathcal{P})$-graphs.*

Proof. Let $\mathcal{R}$ be an $(\mathcal{I}, \mathcal{P})$-representation of G having V_b as its set of vertices on the bottom stab line. Thus $\mathcal{R}_b$ is a proper interval representation of $G[V_b]$. Assume that $a = \min span(V_b)$ and $b = \max span(V_b)$. Then, it can be seen that there exists a strictly increasing function $f : [a, b] \to [a, b']$ for some $b' \in \mathbb{R}$ such that $f(x_u^+) - f(x_u^-) = 1$ for all $u \in V_b$. Such an f could be constructed by induction on $|V_b|$ as follows. Note that since $\mathcal{R}_b$ is a proper interval representation, there are no two distinct vertices $u, v \in V_b$ such that $x_u^- = x_v^-$ or $x_u^+ = x_v^+$. If $|V_b| = 1$, then there exists a single vertex $u \in V_b$ such that $x_u^- = a$ and $x_u^+ = b$. In this case, define $f : [a, b] \to [a, a+1]$ to be an arbitrary strictly increasing function. Suppose that $|V_b| > 1$. Let v be the vertex in V_b such that $x_v^- = a$ and let $V_b' = V_b \setminus \{v\}$. Further, let $c = \min span(V_b')$. Clearly, $a < c \leq b$. By the inductive hypothesis applied on the set V_b' and interval $[c, b]$, there exists a strictly increasing function $f' : [c, b] \to [c, b']$ for some

$b' \in \mathbb{R}$ such that for every $u \in V_b'$, $f'(x_u^+) - f'(x_u^-) = 1$. If $x_v^+ \geq c$, then define f as follows. Let $f_1 : [a, c] \to [f'(x_v^+) - 1, c]$ be an arbitrary strictly increasing function (notice that $f'(x_v^+) - 1 < c$). Define

$$f(x) = \begin{cases} a + f_1(x) - (f'(x_v^+) - 1) & a \leq x < c \\ f'(x) - (f'(x_v^+) - 1 - a) & c \leq x \leq b \end{cases}$$

If $x_v^+ < c$, then define f as follows. Let $f_1 : [a, x_v^+] \to [a, a+1]$ be an arbitrary strictly increasing function. Define

$$f(x) = \begin{cases} f_1(x) & a \leq x \leq x_v^+ \\ x - (x_v^+ - a - 1) & x_v^+ < x < c \\ f'(x) - (x_v^+ - a - 1) & c \leq x \leq b \end{cases}$$

It can be verified that f satisfies the necessary conditions.

Now we extend the function f to the whole of the real line by defining

$$\hat{f}(x) = \begin{cases} x & \text{if } x < a, \\ f(x) & \text{if } a \leq x \leq b, \\ x + (b' - b) & \text{if } x > b. \end{cases}$$

For each vertex $u \in V(G)$, replace the rectangle $r_u = [x_u^-, x_u^+] \times [y_u^-, y_u^+]$ from the representation $\mathcal{R}$ by the rectangle $r_u' = [\hat{f}(x_u^-), \hat{f}(x_u^+)] \times [y_u^-, y_u^+]$ to obtain an $(\mathcal{I},\mathcal{U})$-representation $\mathcal{R}'$ of G. □

From the proof of Lemma 3.2.3, it is clear that the left and right edges of the rectangles of $\mathcal{R}'$ are in the same order as they are in $\mathcal{R}$. Let G be a $(\mathcal{P},\mathcal{P})$-graph and $\mathcal{R}$ be a $(\mathcal{P},\mathcal{P})$-representation of G. The construction procedure described in Lemma 3.2.3 when applied on $\mathcal{R}$ gives us a $(\mathcal{P},\mathcal{U})$-representation $\mathcal{R}'$ of G. This gives us the following lemma.

Lemma 3.2.4. *$(\mathcal{P},\mathcal{U})$-graphs $=$ $(\mathcal{P},\mathcal{E})$-graphs $=$ $(\mathcal{P},\mathcal{P})$-graphs.*

When we are given an $(\mathcal{E},\mathcal{E})$-representation $\mathcal{R}$ of a graph G, we can al-

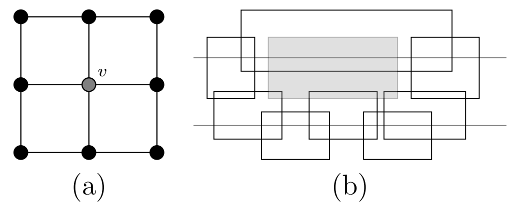


Figure 3.2.2: $(\mathcal{I}, \mathcal{P})$-representation of $(3,3)$-grid graph.

ways scale the whole representation horizontally by an appropriate factor so that the intervals of $\mathcal{R}_b$ becomes unit intervals. Note that the intervals of $\mathcal{R}_t$ will all continue to be of the same length even after this operation. We thus have the following lemma.

Lemma 3.2.5. *$(\mathcal{E},\mathcal{U})$-graphs $=$ $(\mathcal{E},\mathcal{E})$-graphs.*

By definition, a $(\mathcal{P}, \mathcal{P})$-graph is an $(\mathcal{I}, \mathcal{P})$-graph. Consider the $(3,3)$-grid graph H shown in Figure 3.2.2(a). Clearly, there is an $(\mathcal{I}, \mathcal{P})$-representation of H (Figure 3.2.2(b)). Note that H is a graph belonging to the $W_{9,4}$ class. By Proposition 3.2.2, in any 2-exactly stabbed rectangle intersection representation of H, there is a vertex $w \in N(v)$ such that w, v have a common stab and $span(v) \subset span(w)$. Hence, H is not a $(\mathcal{P}, \mathcal{P})$-graph. So we have the following lemma.

Lemma 3.2.6. *The class of $(\mathcal{P}, \mathcal{P})$-graphs is a proper subset of the class of $(\mathcal{I}, \mathcal{P})$-graphs.*

Now we will show in Lemma 3.2.8 that the class of $(\mathcal{E}, \mathcal{E})$-graphs is a proper subset of the class of $(\mathcal{P}, \mathcal{P})$-graphs. Essentially, we shall show that the graph shown in Figure 3.2.4(a) is a $(\mathcal{P}, \mathcal{P})$-graph (representation given in Figure 3.2.4(b)) but not an $(\mathcal{E}, \mathcal{E})$-graph. To prove Lemma 3.2.8, we need some observations and propositions.

The following observation can be observed to be true along the lines of Observation 3.2.1.

Observation 3.2.4. *Let $\mathcal{R}$ be a $(\mathcal{P},\mathcal{P})$-representation of a triangle free graph G. Let $e = uv$ be a bridge edge and $a, b \in V(G)$ such that $span(a) < span(uv) < span(b)$. Then a and b are in different connected components of $G - \{u, v\}$.*

Observation 3.2.5. *Let C be the cycle $v_0v_1 \cdots v_{n-1}v_0$, where $n \geq 4$, and let $\mathcal{R}$ be a 2-exactly stabbed rectangle intersection representation of C. If for some $i, j \in \{0, 1, \ldots, n-1\}$ and $i \neq j$, v_iv_{i+1} and v_jv_{j+1} are both bridge edges in $\mathcal{R}$ (subscripts modulo n), then $span(x) \cap span(Y) \neq \emptyset$ for each $x \in X$ and $span(y) \cap span(X) \neq \emptyset$ for each $y \in Y$, where $X = \{v_{i+1}, v_{i+2}, \ldots, v_j\}$ and $Y = \{v_{j+1}, v_{j+2}, \ldots, v_i\}$.*

Proof. By Proposition 3.2.1, v_iv_{i+1} and v_jv_{j+1} are the only two bridge edges in C. Therefore, all the vertices in X are on one stab line and all the vertices in Y are on the other stab line. By Observation 3.2.3, we shall assume without loss of generality that $span(v_iv_{i+1}) < span(v_jv_{j+1})$. Clearly, there exists $u \in \{v_j, v_{j+1}\}$ such that $span(v_iv_{i+1}) < span(u)$. Suppose for the sake of contradiction that there is a vertex $y \in Y$ such that $span(y) < span(X)$. Then, we have $span(y) < span(v_iv_{i+1}) < span(u)$, which implies by Observation 3.2.1 that y and u lie in different components of $G - (N[v_i] \cup N[v_{i+1}])$. This is a contradiction as one cannot remove the endpoints of any edge and their neighbours to disconnect C. Repeating the same argument, we get that there exists no vertex $y \in Y$ such that $span(X) < span(y)$ and that no vertex $x \in X$ can have $span(x) < span(Y)$ or $span(Y) < span(x)$. This completes the proof. □

Consider the cycle $C = v_0v_1 \cdots v_{n-1}v_0$ on $n \geq 4$ vertices. Now add a new vertex w adjacent only to v_0 to obtain the graph $C_{n,1}$.

Proposition 3.2.3. *Let $\mathcal{R}$ be a $(\mathcal{P},\mathcal{P})$-representation of $C_{n,1}$. Then at least one of v_0v_1 and v_0v_{n-1} must be a bridge edge.*

Proof. Suppose that neither of v_0v_1 or v_0v_{n-1} is a bridge edge and without loss of generality assume that all of v_0, v_1, v_{n-1} are on the top stab line. As a proper interval graph cannot have a $K_{1,3}$ as an induced subgraph, the edge v_0w is a bridge edge with w on the bottom stab line. Moreover, by Proposition 3.2.1, C has exactly two bridge edges. Let the bridge edges in C be z_1w_1 and z_2w_2. As neither of the two edges v_0v_1, v_0v_{n-1} of C that are incident on v_0 are bridge edges, we have that $v_0 \notin \{z_1, z_2, w_1, w_2\}$. From Observation 3.2.3, we know that the spans of the bridge edges v_0w, z_1w_1 and z_2w_2 are pairwise disjoint. Let e_1, e_2, e_3 be the edges in $\{v_0w, z_1w_1, z_2w_2\}$ such that $span(e_1) < span(e_2) < span(e_3)$. Clearly, there is an endpoint a of e_1 and an endpoint b of e_3 such that $span(a) < span(e_2) < span(b)$. By Observation 3.2.4, we can infer that a and b become disconnected if the two endpoints of e_2 are removed from G. But this is a contradiction as the graph G cannot be disconnected by removing the two endpoints of any of the edges e_1, e_2 or e_3. Therefore, at least one of v_0v_1 or v_0v_{n-1} must be a bridge edge. □

For a graph G, let $\alpha(G)$ denote the *independence number*, that is, the cardinality of the maximum independent set of G. For an interval I, let $|I|$ denote the length of the interval.

Observation 3.2.6. *Let $\mathcal{R}$ be an $(\mathcal{E}, \mathcal{E})$-representation of a graph G. Let ℓ_1, ℓ_2 be the two stab lines in $\mathcal{R}$ and let l_1, l_2 be the lengths of the spans of each vertex on ℓ_1 and ℓ_2 respectively. Let G_1 be a connected subgraph of G such that all vertices in $V(G_1)$ are on ℓ_1. Let G_2 be the subgraph induced in G by the set $\{w \in V(G)\colon span(w) \cap span(V(G_1)) \neq \emptyset$ and w is on $\ell_2\}$. Then $l_1 \cdot |V(G_1)| > l_2 \cdot (\alpha(G_2) - 2)$.*

Proof. Note that $span(V(G_1))$ is an interval since G_1 is connected. As the span of every vertex in G_1 is l_1, we have that $|span(V(G_1))| \leq l_1 \cdot |V(G_1)|$. Let $S = \{w_1, w_2, \ldots, w_{\alpha(G_2)}\}$ be a maximum independent set of G_2 such that $span(w_1) < span(w_2) < \cdots < span(w_{\alpha(G_2)})$. Hence, we have $(x^-_{w_{\alpha(G_2)}} - x^+_{w_1}) > l_2 \cdot (\alpha(G_2) - 2)$. Since $span(V(G_1))$ intersects with both

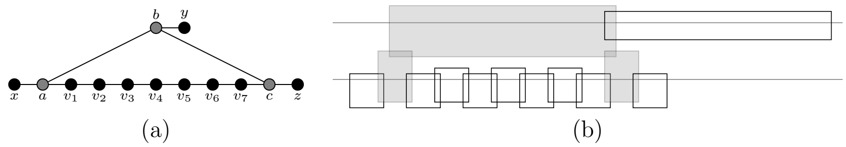


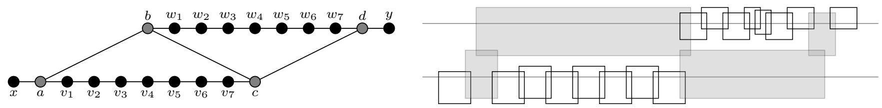
Figure 3.2.3: (a) The graph F, and (b) an $(\mathcal{E},\mathcal{E})$-representation of F.

span(w_1) and *span*$(w_{\alpha(G_2)})$, we must have $|span(V(G_1))| > l_2 \cdot (\alpha(G_2) - 2)$. This completes the proof. □

The observations and propositions proved above are enough to show that the class of $(\mathcal{U},\mathcal{U})$-graphs is properly contained in the class of $(\mathcal{E},\mathcal{E})$-graphs. By definition, a $(\mathcal{U},\mathcal{U})$-graph is an $(\mathcal{E},\mathcal{E})$-graph. We show that the graph F shown in Figure 3.2.3(a) is not a $(\mathcal{U},\mathcal{U})$-graph.

Proposition 3.2.4. *Let $\mathcal{R}$ be an $(\mathcal{E},\mathcal{E})$-representation of F having stab lines ℓ_1 and ℓ_2. Further, let l_1 and l_2 be the lengths of the spans of the vertices on ℓ_1 and ℓ_2 respectively in $\mathcal{R}$. If b is on ℓ_1, then $l_1 > l_2$.*

Proof. Since each of $F - \{y,z\}$, $F - \{x,z\}$, $F - \{x,y\}$ is isomorphic to $C_{10,1}$, we know by Proposition 3.2.3 that at least one edge in each of $\{v_1a, ab\}$, $\{ab, bc\}$ and $\{bc, cv_7\}$ is a bridge edge in $\mathcal{R}$. By Proposition 3.2.1 applied to the cycle $C = F - \{x,y,z\}$, it then follows that there are exactly two bridge edges $e_1 \in \{v_1a, ab\}$ and $e_2 \in \{bc, cv_7\}$ in C and that $\{e_1, e_2\} \neq \{v_1a, cv_7\}$. Let P_1, P_2 be the two paths that form the components in the graph obtained by removing e_1 and e_2 from C, where $b \in V(P_1)$. Observe that $|V(P_1)| \leq 2$ and that $\alpha(P_2) \geq 4$. By Observation 3.2.5, for each $u \in V(P_2)$, we have $span(u) \cap span(V(P_1)) \neq \emptyset$. It is clear by Proposition 3.2.1 that in $\mathcal{R}$, the vertices of $V(P_1)$ are on the stab line ℓ_1 (recall that b is on ℓ_1) while the vertices of $V(P_2)$ are on the stab line ℓ_2. Then by Observation 3.2.6 applied to P_1, we have that $l_1 \cdot |V(P_1)| > l_2 \cdot (\alpha(P_2) - 2)$, which gives $l_1 > l_2$. □



(a) (b)

Figure 3.2.4: (a) The graph H, and (b) a $(\mathcal{P}, \mathcal{P})$-representation of H.

It is clear from Proposition 3.2.4 that the graph F has no $(\mathcal{U},\mathcal{U})$-representation. On the other hand, there is an $(\mathcal{E}, \mathcal{E})$-representation of F as shown in Figure 3.2.3(b). Hence, we have the following lemma.

Lemma 3.2.7. *The class of $(\mathcal{U},\mathcal{U})$-graphs is a proper subset of the class of $(\mathcal{E}, \mathcal{E})$-graphs.*

Next, we show that the class of $(\mathcal{E}, \mathcal{E})$-graphs is a proper subset of the class of $(\mathcal{P}, \mathcal{P})$-graphs. By definition, an $(\mathcal{E}, \mathcal{E})$-graph is a $(\mathcal{P}, \mathcal{P})$-graph. Let H be the graph depicted in Figure 3.2.4(a). As shown in Figure 3.2.4(b), there is a $(\mathcal{P}, \mathcal{P})$-representation of H. We shall show that H cannot have any $(\mathcal{E}, \mathcal{E})$-representation.

First, we prove the following observation about the graph H.

Proposition 3.2.5. *In any $(\mathcal{P}, \mathcal{P})$-representation of H, bc is a bridge edge.*

Proof. Suppose for the sake of contradiction that there is a $(\mathcal{P}, \mathcal{P})$-representation $\mathcal{R}$ of H in which bc is not a bridge edge. Since the subgraph of H induced by the set $\{a, v_1, v_2, \ldots, v_7, b, c, w_1\}$ is isomorphic to $C_{10,1}$, by Proposition 3.2.3, the edge ba must be a bridge edge in $\mathcal{R}$. The same argument applied on the subgraphs in H induced by $\{b, w_1, w_2, \ldots, w_7, c, d, v_7\}$, $\{b, w_1, w_2, \ldots, w_7, c, d, a\}$, $\{a, v_1, v_2, \ldots, v_7, b, c, d\}$ gives that cd, bw_1, cv_7 respectively are bridge edges in $\mathcal{R}$. Without loss of generality assume that both b and c are on the bottom stab line in $\mathcal{R}$ and that $x_c^- < x_b^- < x_c^+ < x_b^+$. By Observation 3.2.3, we have that $span(cv_7) \cap span(cd) = \emptyset$ and

$span(bw_1) \cap span(ba) = \emptyset$. Moreover, if for any $e \in \{cv_7, cd\}$ and $e' \in \{bw_1, ba\}$, we have $span(e') < span(e)$, then $span(e') \cap span(bc) \neq \emptyset$ and $span(e) \cap span(bc) \neq \emptyset$, which is a contradiction to Observation 3.2.2. Therefore, there exist $e \in \{cv_7, cd\}$ and $e_1, e_2 \in \{bw_1, ba\}$ such that $span(e) < span(e_1) < span(e_2)$. Then there is an endpoint u of e and an endpoint u' of e_2 such that $span(u) < span(e_1) < span(u')$. Clearly, $u \in \{c, d, v_7\}$, $u' \in \{a, w_1\}$ and $e_1 = bu''$, where $\{u''\} = \{a, w_1\} \setminus \{u'\}$. By Observation 3.2.4, u and u' must lie in different connected components in $H - \{b, u''\}$, a contradiction. This completes the proof of the claim. □

Proposition 3.2.6. *The graph H has no $(\mathcal{E}, \mathcal{E})$-representation.*

Proof. Suppose for the sake of contradiction that there is an $(\mathcal{E}, \mathcal{E})$-representation $\mathcal{R}$ of the graph H. Let l_t, l_b be the lengths of the spans of the vertices on the top and bottom stab lines respectively in $\mathcal{R}$. Let H_1 and H_2 be the subgraphs induced in H by the vertices in $\{x, a, b, c, d, v_1, v_2, \ldots, v_7, w_1\}$ and $\{a, b, c, d, y, w_1, w_2, \ldots, w_7, v_7\}$ respectively. Clearly, H_1 and H_2 are both isomorphic to the graph F shown in Figure 3.2.3(a). By Proposition 3.2.5, we can assume by symmetry that b is on the top stab line and c is on the bottom stab line in $\mathcal{R}$. We then have $|span(b)| = l_t$ and $|span(c)| = l_b$. By Proposition 3.2.4 applied to H_1, we get that $l_t > l_b$ and by applying the same proposition to H_2, we get that $l_b > l_t$. This contradiction completes the proof. □

The graph H has a $(\mathcal{P}, \mathcal{P})$-representation as shown in Figure 3.2.4(b). From Proposition 3.2.6, we therefore have the following lemma.

Lemma 3.2.8. *The class of $(\mathcal{E}, \mathcal{E})$-graphs is a proper subset of the class of $(\mathcal{P}, \mathcal{P})$-graphs.*

Finally, we are ready to prove Theorem 3.2.1.

Proof of Theorem 3.2.1. Observe from the definitions that the class 2-SUIG is the same as the class of $(\mathcal{U}, \mathcal{U})$-graphs. Then the proof of Theorem 3.2.1 follows from Lemmas 3.2.1–3.2.8. □

3.3 Recognition algorithm

In this section, we prove the following theorem.

Theorem 3.3.1. *Let G be a triangle-free graph. There is an $O(|V(G)|)$ time algorithm to decide if G is a $(\mathcal{P}, \mathcal{P})$-graph.*

An *LL-drawing* of a planar graph G is a straight line planar embedding of G such that the point corresponding to each vertex of G lies on one of two given horizontal lines. A planar graph G is an *LL-graph* if G has an LL-drawing.

Observation 3.3.1. *A graph G is an LL-graph if and only if there exists a partition of $V(G)$ into two ordered sets $A = \{a_1, a_2, \ldots, a_k\}$ and $B = \{b_1, b_2, \ldots, b_t\}$ such that:*

(a) there does not exist $1 \leq i < j \leq k$ and $1 \leq i' < j' \leq t$ with the property that $a_i b_{j'}, a_j b_{i'} \in E(G)$, and

(b) for any a_i, $N(a_i) \cap A \subseteq \{a_{i-1}, a_{i+1}\}$ and for any b_i, $N(b_i) \cap B \subseteq \{b_{i-1}, b_{i+1}\}$.

Indeed, the two ordered sets A and B referred to in the above observation consist of the vertices lying on each of the two horizontal lines in an LL-drawing of the graph, sorted according to their increasing X-coordinate. We show that all LL-graphs are $(\mathcal{P}, \mathcal{P})$-graphs and that all triangle-free $(\mathcal{P}, \mathcal{P})$-graphs are LL-graphs.

Lemma 3.3.1. *If G is an LL-graph then G is a $(\mathcal{P}, \mathcal{P})$-graph.*

Proof. Let $(A = \{a_1, a_2, \ldots, a_s\}, B = \{b_1, b_2, \ldots, b_t\})$ be the partition of $V(G)$ as given by Observation 3.3.1. For any $a_j \in A$, define $l(a_j) = \min\{i\colon b_i \in N(a_j)\}$ and $r(a_j) = \max\{i\colon b_i \in N(a_j)\}$ if $N(a_j) \cap B \neq \emptyset$ and $l(a_j) = r(a_j) = 0$ otherwise. Then by Observation 3.3.1, for any a_i, a_j

such that $i < j$, both having neighbours in B, we have $r(a_i) \leq l(a_j)$ and $a_i a_j$ is an edge only if $|i - j| = 1$.

Let us define a $(\mathcal{P}, \mathcal{P})$-representation $\mathcal{R} = \{[x_u^-, x_u^+] \times [y_u^-, y_u^+]\}_{u \in V(G)}$ having stab lines $y = 0$ and $y = t + 1$ as follows. For each b_i, we set

$$x_{b_i}^- = i, \quad x_{b_i}^+ = \begin{cases} i + \frac{1}{2} & \text{if } b_i b_{i+1} \notin E(G) \\ i + 1 & \text{otherwise} \end{cases}$$
$$y_{b_i}^- = 0, \quad y_{b_i}^+ = \begin{cases} i & \text{if } N(b_i) \cap A \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

Let $\epsilon = \frac{1}{2|V(G)|}$ and

$$x_{a_1}^- = 1$$

$$x_{a_1}^+ = \begin{cases} r(a_1) + \epsilon & \text{if } N(a_1) \cap B \neq \emptyset \\ 1 + \frac{\epsilon}{2} & \text{otherwise} \end{cases}$$

$$y_{a_1}^- = \begin{cases} l(a_1) & \text{if } N(a_1) \cap B \neq \emptyset \\ t + 1 & \text{otherwise} \end{cases}$$

$$y_{a_1}^+ = t + 1$$

For each a_i, $i > 1$, we inductively define

$$x_{a_i}^- = \begin{cases} x_{a_{i-1}}^+ & \text{if } a_{i-1} a_i \in E(G) \\ x_{a_{i-1}}^+ + \frac{\epsilon}{2} & \text{otherwise} \end{cases}$$

$$x_{a_i}^+ = \begin{cases} r(a_i) + i \cdot \epsilon & \text{if } N(a_i) \cap B \neq \emptyset \\ x_{a_i}^- + \frac{\epsilon}{2} & \text{otherwise} \end{cases}$$

$$y_{a_i}^- = \begin{cases} l(a_i) & \text{if } N(a_i) \cap B \neq \emptyset \\ t + 1 & \text{otherwise} \end{cases} ,$$

$$y_{a_i}^+ = t + 1$$

Observe that $\mathcal{R}$ is a valid $(\mathcal{P}, \mathcal{P})$-representation of G. A proof is given below for the sake of completeness.

For each $a_i \in A$, define $P_i = \{l : l < i \text{ and } N(a_l) \cap B \neq \emptyset\}$.

Claim 1. Let $a_i \in A$. If $P_i \neq \emptyset$, then $x^-_{a_i} \leq x^+_{a_j} + (i - j - \frac{1}{2}) \cdot \epsilon$, where $j = \max P_i$. If $P_i = \emptyset$, then $x^-_{a_i} \leq 1 + (i-1) \cdot \epsilon$.

First, suppose that $P_i \neq \emptyset$. Then $x^-_{a_{j+1}} \leq x^+_{a_j} + \frac{\epsilon}{2} \Rightarrow x^+_{a_{j+1}} \leq x^+_{a_j} + \epsilon \Rightarrow x^+_{a_{j+2}} \leq x^+_{a_j} + 2 \cdot \epsilon \Rightarrow \cdots \Rightarrow x^+_{a_{i-1}} \leq x^+_{a_j} + (i - 1 - j) \cdot \epsilon \Rightarrow x^-_{a_i} \leq x^+_{a_j} + (i - j - \frac{1}{2}) \cdot \epsilon$. Next, suppose that $P_i = \emptyset$. Then $x^-_{a_1} = 1$ and $x^+_{a_1} \leq 1 + \frac{\epsilon}{2} \Rightarrow x^-_{a_2} \leq 1 + \epsilon \Rightarrow \cdots \Rightarrow x^-_{a_i} \leq 1 + (i-1) \cdot \epsilon$. This proves the claim.

First, let us prove that $\mathcal{R}$ is a valid rectangle intersection representation by showing that for every $u \in V(G)$, $x^+_u \geq x^-_u$ and $y^+_u \geq y^-_u$. This is clearly true for $u \in B$. Also, for each $u \in A$, we have $y^+_u \geq y^-_u$. Now let us prove that for each $a_i \in A$, we have $x^+_{a_i} > x^-_{a_i}$. Clearly, this is true for a_1. Suppose that for some $i > 1$, we have $x^+_{a_i} \leq x^-_{a_i}$. Clearly, $N(a_i) \cap B \neq \emptyset$. Then $x^+_{a_i} = r(a_i) + i \cdot \epsilon$. Suppose that $P_i \neq \emptyset$. Let $j = \max P_i$. Then, $x^+_{a_j} = r(a_j) + j \cdot \epsilon$. Using Claim 1, we now get $x^+_{a_j} + (i - j - \frac{1}{2}) \cdot \epsilon \geq x^-_{a_i} \geq x^+_{a_i} = r(a_i) + i \cdot \epsilon \Rightarrow x^+_{a_j} \geq r(a_i) + (j + \frac{1}{2}) \cdot \epsilon \geq r(a_j) + j \cdot \epsilon + \frac{1}{2} \cdot \epsilon$, which is a contradiction (here we use the fact that $r(a_j) \leq l(a_i) \leq r(a_i)$ by Observation 3.3.1). If $P_i = \emptyset$, then by Claim 1, we have $1 + (i-1) \cdot \epsilon \geq x^-_{a_i} \geq x^+_{a_i} = r(a_i) + i \cdot \epsilon$. This implies that $r(a_i) \leq 1 - \epsilon$, which is a contradiction.

The arguments above show that $\mathcal{R}$ is a 2-exactly stabbed rectangle intersection representation with stab lines $y = 0$ and $y = t + 1$. It is easy to see that there does not exist $u, v \in B$ such that $span(u) \subseteq span(v)$. Suppose that there exist $a_i, a_j \in A$ such that $span(a_j) \subseteq span(a_i)$. Then $x^-_{a_i} \leq x^-_{a_j} < x^+_{a_j} \leq x^+_{a_i}$. If $i > j$, then we have $x^-_{a_i} \geq x^+_{a_{i-1}} \geq x^-_{a_{i-1}} \geq x^+_{a_{i-2}} \geq \cdots \geq x^+_{a_j}$, which is a contradiction. If $i < j$, then we have $x^-_{a_j} \geq x^+_{a_{j-1}} \geq x^-_{a_{j-1}} \geq x^+_{a_{j-2}} \geq \cdots \geq x^+_{a_i}$, which is again a contradiction. Therefore, $\mathcal{R}$ is a $(\mathcal{P}, \mathcal{P})$-representation.

It is easy to see using Observation 3.3.1 that for $u, v \in B$ and for $u, v \in A$, $r_u \cap r_v \neq \emptyset$ if and only if $uv \in E(G)$. Let $a_i \in A$ and $b_k \in B$. Suppose that $a_i b_k \in E(G)$. Then $l(a_i) \leq k \leq r(a_i)$. Suppose that $P_i \neq \emptyset$. Let $j = \max P_i$. Then $x^+_{a_j} = r(a_j) + j \cdot \epsilon$. By Claim 1, $x^-_{a_i} \leq x^+_{a_j} + (i - j - \frac{1}{2}) \cdot \epsilon = r(a_j) + (i - \frac{1}{2}) \cdot \epsilon$. By Observation 3.3.1, $r(a_j) \leq l(a_i)$. So, by our choice of ϵ, we get $x^-_{a_i} \leq l(a_j) + \frac{1}{2} \leq k + \frac{1}{2}$. If $P_i = \emptyset$, then by Claim 1, we have $x^-_{a_i} \leq 1 + (i-1) \cdot \epsilon$, which again gives us $x^-_{a_i} \leq k + \frac{1}{2}$. On the other hand, $x^-_{b_k} = k$ and $x^+_{b_k} \geq k + \frac{1}{2}$. As $x^+_{a_i} = r(a_i) + i \cdot \epsilon \geq k$, we have that $span(a_i) \cap span(b_k) \neq \emptyset$. Observe that $[y^-_{b_k}, y^+_{b_k}] = [0, k]$, $y^+_{a_i} = t + 1$ and $y^-_{a_i} \leq k$. We therefore conclude that $r_{a_i} \cap r_{b_k} \neq \emptyset$.

We now show that if $uv \notin E(G)$, then $r_u \cap r_v = \emptyset$. It is obvious from the definition of $\mathcal{R}$ and Observation 3.3.1 that this holds if $u, v \in A$ or $u, v \in B$. Suppose that $a_i \in A$ and $b_k \in B$ such that $a_i b_k \notin E(G)$. If $N(a_i) \cap B = \emptyset$, then $y^-_{a_i} = t + 1$ while $y^+_{b_k} \leq t$, implying that $r_{a_i} \cap r_{b_k} = \emptyset$. Similarly, if $N(b_k) \cap A = \emptyset$, then $y^+_{b_k} = 0$ while $y^-_{a_i} \geq 1$, again implying that $r_{a_i} \cap r_{b_k} = \emptyset$. So let us assume that $N(a_i) \cap B \neq \emptyset$ and $N(b_k) \cap A \neq \emptyset$. Then by Observation 3.3.1, we have that either $r(a_i) < k$ or $l(a_i) > k$. In the former case, we have $x^+_{a_i} = r(a_i) + i \cdot \epsilon < k = x^-_{b_k}$ (by our choice of ϵ), and in the latter case, we have $y^-_{a_i} = l(a_i) > k$ and $y^+_{b_k} = k$. Therefore, in both cases, we get $r_{a_i} \cap r_{b_k} = \emptyset$.

This shows that $\mathcal{R}$ is a valid $(\mathcal{P}, \mathcal{P})$-representation of G. □

Lemma 3.3.2. *If a triangle-free graph G is a $(\mathcal{P}, \mathcal{P})$-graph then G is an LL-graph.*

Proof. Let $\mathcal{R} = \{r_u = [x^-_u, x^+_u] \times [y^-_u, y^+_u]\}_{u \in V(G)}$ be a $(\mathcal{P}, \mathcal{P})$-representation of G having stab lines $y = 0$ and $y = 1$. Let $A = \{u \colon r_u$ intersects the stab line $y = 1\}$ and $B = V(G) \setminus A$. Clearly, the rectangles corresponding to each vertex in B intersects the stab line $y = 0$.

Let $s = |A|$ and $t = |B|$. Also let $a_1, a_2, \ldots, a_s$ be the vertices of A and let $b_1, b_2, \ldots, b_t$ be the vertices of B, such that for $1 \leq i < j \leq s$, $x^-_{a_i} \leq x^-_{a_j}$ and for $1 \leq i < j \leq t$, $x^-_{b_i} \leq x^-_{b_j}$. Observe that the sets A and

B also satisfy all conditions of Observation 3.3.1, proving that G is an LL-graph. A proof is given below for sake of completeness.

Suppose that the sets A and B with these orderings on their vertices violate condition (a) of Observation 3.3.1. Then there exist i, j, i', j' such that $1 \leq i < j \leq s$ and $1 \leq i' < j' \leq t$ and $a_i b_{j'}, a_j b_{i'} \in E(G)$. Therefore, $r_{a_i} \cap r_{b_{j'}} \neq \emptyset$, implying that $x^+_{a_i} \geq x^-_{b_{j'}}$. Similarly, we have $x^+_{b_{i'}} \geq x^-_{a_j}$. As $x^-_{a_i} \leq x^-_{a_j}$, $x^-_{b_{i'}} \leq x^-_{b_{j'}}$ and $\mathcal{R}$ is a $(\mathcal{P}, \mathcal{P})$-representation, we have $x^+_{a_i} \leq x^+_{a_j}$ and $x^+_{b_{i'}} \leq x^+_{b_{j'}}$.

Combining with previous inequalities, we now get $x^+_{a_i} \geq x^-_{b_{j'}} \geq x^-_{b_{i'}}$ and $x^+_{b_{i'}} \geq x^-_{a_j} \geq x^-_{a_i}$. This implies that the intervals $[x^-_{a_i}, x^+_{a_i}]$ and $[x^-_{b_{i'}}, x^+_{b_{i'}}]$ intersect. Similarly, we get $x^+_{a_j} \geq x^+_{a_i} \geq x^-_{b_{j'}}$ and $x^+_{b_{j'}} \geq x^+_{b_{i'}} \geq x^-_{a_j}$ which implies that the intervals $[x^-_{a_j}, x^+_{a_j}]$ and $[x^-_{b_{j'}}, x^+_{b_{j'}}]$ intersect.

Now let us consider the case when $y^+_{b_{i'}} \leq y^+_{b_{j'}}$ (the case when $y^+_{b_{i'}} > y^+_{b_{j'}}$ is symmetric and will not be discussed). As $r_{b_{i'}} \cap r_{a_j} \neq \emptyset$, we have $y^+_{b_{i'}} \geq y^-_{a_j}$ and therefore $y^+_{b_{j'}} \geq y^-_{a_j}$. Combined with our previous observation that $[x^-_{a_j}, x^+_{a_j}] \cap [x^-_{b_{j'}}, x^+_{b_{j'}}] \neq \emptyset$, this implies that $r_{a_j} \cap r_{b_{j'}} \neq \emptyset$ (recall that $0 \in [y^-_{b_{j'}}, y^+_{b_{j'}}]$ and $1 \in [y^-_{a_j}, y^+_{a_j}]$). Since this means that $a_j b_{j'} \in E(G)$, we have $a_i a_j, b_{i'} b_{j'} \notin E(G)$ as otherwise, either $a_i a_j b_{j'}$ or $b_{i'} b_{j'} a_j$ will be a triangle in G.

Therefore, we can conclude that $x^+_{a_i} < x^-_{a_j}$ and $x^+_{b_{i'}} < x^-_{b_{j'}}$. Combining these with our earlier observation that $x^+_{a_i} \geq x^-_{b_{j'}}$, we get $x^-_{a_j} > x^+_{b_{i'}}$, which contradicts the fact that $x^+_{b_{i'}} \geq x^-_{a_j}$. This shows that the sets A and B do not violate condition (a) of Observation 3.3.1.

Now suppose that $a_i a_j \in E(G)$, where $i < j$, then $x^+_{a_i} \geq x^-_{a_j}$. Since $\mathcal{R}$ is a $(\mathcal{P}, \mathcal{P})$-representation, we then have that $x^-_{a_i} \leq x^-_{a_{i+1}} \leq \cdots \leq x^-_{a_j} \leq x^+_{a_i} \leq x^+_{a_{i+1}} \leq \cdots \leq x^+_{a_j}$. Therefore, the rectangles $r_{a_i}, r_{a_{i+1}}, \ldots, r_{a_j}$ will intersect pairwise, implying that $\{a_i, a_{i+1}, \ldots, a_j\}$ induces a complete graph in G. Since G is triangle-free, we can conclude that $j = i+1$. It can be similarly proven that if $b_i b_j \in E(G)$, where $i < j$, we have $j = i + 1$. Therefore, the sets A and B also satisfy condition (b) of Observation 3.3.1, proving that G is an LL-graph. □

Now we are ready to prove Theorem 3.3.1.

Proof of Theorem 3.3.1. Due to Lemma 3.3.1 and Lemma 3.3.2, we can infer that a triangle-free graph G is a $(\mathcal{P}, \mathcal{P})$-graph if and only if G is an LL-graph. Cornelsen, Schank and Wagner [61] proved that given a graph G, there is an $O(|V(G)|)$ time algorithm to decide if G is an LL-graph. Combining this with Lemma 3.3.1 and 3.3.2, we are done. □

3.4 Coloring 2-SRIGs

In this section, we prove the following two propositions.

Proposition 3.4.1. *Triangle-free* 2*-SRIGs are 3-colorable.*

Proposition 3.4.2. *For every natural number c, there exists a polynomial-time algorithm that decides whether an input* 2*-SRIG graph is c-colorable.*

These two propositions follow directly from a few known results making the proofs rather short. We will first prove Proposition 3.4.1.

Proof of Proposition 3.4.1. Let H be a triangle-free 2-SRIG. First, we will show that H is a planar graph.

Observe that it is possible to find 2-stabbed rectangle intersection representation $\mathcal{R}$ of H such that given any two vertices $u, v \in V(H)$, $r_v \setminus r_u$ is nonempty and connected. Thus H is a planar graph due to Perepelitsa (Theorem 7 [134]).

The Grötzsch's Theorem [151] says that every triangle-free planar graph is 3-colorable. Moreover, we can get such a coloring in $O(|V(H)|)$ time due to Dvořák, Kawarabayashi and Thomas [70]. □

Next we prove Proposition 3.4.2.

Proof of Proposition 3.4.2. To prove this, observe that when a 2-SRIG graph G is c-colorable, the treewidth [63] of G is at most $2c$. This can be seen as follows. Consider a 2-stabbed rectangle intersection representation $\mathcal{R}$ of G. Let the vertices of G be $v_1, v_2, \ldots, v_n$ such that $x^-_{v_1} \leq x^-_{v_2} \leq \cdots \leq x^-_{v_n}$. For $1 \leq i \leq n$, define $X_i = \{v_j : x^-_{v_i} \in span(v_j)\}$. It is easy to see that $\{X_i\}_{1 \leq i \leq n}$ is a path decomposition of G (where the underlying path is X_1-X_2-$\cdots$-X_n). Notice that for each i, the vertices of X_i that are on one stab line induce a complete graph in G. Since G is c-colorable, this implies that X_i contains at most c vertices that are on one stab line. Therefore, $|X_i| \leq 2c$, implying that the X_i-s form a path decomposition of G having width at most $2c$. Thus the pathwidth, and hence also the treewidth, of G is at most $2c$. Now, for every constant c, there is a polynomial-time algorithm $\mathcal{A}_c$ that checks if an input graph G has treewidth at most $2c$ and constructs a tree decomposition of G width at most $2c$ if it exists (see Theorem 7.17 in [63]). For a natural number c, let $\mathcal{B}_c$ be the algorithm that does the following. The algorithm takes a graph G as input and first runs $\mathcal{A}_c$ on G. It returns a "No" if $\mathcal{A}_c$ concludes that the treewidth of G is greater than $2c$; otherwise it runs a standard polynomial-time dynamic programming algorithm on the tree decomposition of G of width at most $2c$ constructed by $\mathcal{A}_c$ to decide if G is c-colorable (see Theorem 7.9 in [63]). □

3.5 NP-completeness of coloring 2-SRIGs

In this section, we prove the following theorem.

Theorem 3.5.1. *The* Chromatic Number *problem is NP-complete for l-row B_0-VPG graphs for all $l \geq 2$ even if the l-row B_0-VPG representation is given, and is polynomial time solvable for $l = 1$.*

As an immediate consequence of the above, we have the following corollary.

Corollary 6. *The* Chromatic Number *problem is NP-complete for k-SRIGs for all $k \geq 2$, even if the k-SRIG representation is given, and* *is polynomial time solvable for* $k = 1$.

A *circular arc graph* is an intersection graph of arcs of a circle. Given a circular arc graph G along with a circular arc representation $\mathcal{C}$ and an integer k as input, the CIRCULAR ARC COLORING problem is to decide whether the chromatic number of G is at most k. The decision problem CIRCULAR ARC COLORING is known to be NP-complete [83]. To prove Theorem 3.5.1, we will use a reduction from CIRCULAR ARC COLORING using a strategy similar to those used in [40, 53].

Let G be a circular arc graph given along with a circular arc representation $\mathcal{C}$ and an integer k. We describe the construction of a B_0-VPG graph $H_{G,\mathcal{C},k}$ whose chromatic number is k if and only if the chromatic number of G is at most k.

We assume that the circular-arcs of $\mathcal{C}$ are drawn on the unit circle and that no circular-arc in this representation is a point (i.e., degenerate). The "left end-point" of a circular-arc is the first point of the arc that is encountered during an anti-clockwise traversal of the unit circle starting from a point not contained in the arc. The other end-point is the "right end-point" of the arc. For a vertex $v \in V(G)$, let C_v denote the circular-arc representing v in $\mathcal{C}$. Let $\theta_1(v), \theta_2(v) \in [0, 2\pi)$ denote the angles formed by the positive X-axis and the line segments joining the origin to the left and right end-points of C_v respectively (i.e., the end-points of C_v are $(\cos\theta_1, \sin\theta_1)$ and $(\cos\theta_2, \sin\theta_2)$). Our plan is to "cut" the circle at the point $(1, 0)$ and "stretch" it onto the X-axis so that the arcs become straight line segments lying on the X-axis. Observe that during this procedure, the arcs in $\mathcal{C}$ that contain the point $(1, 0)$ get split into two line segments whereas every other arc becomes a line segment. Also note that the arcs C_v that contain the point $(1, 0)$ are exactly those for which $\theta_1(v) > \theta_2(v)$. Let $T = \{v \in V(G) \colon \theta_1(v) > \theta_2(v)\}$. Further, let $|T| = t$ and $T = \{u_1, u_2, \ldots, u_t\}$. We now formally define the B_0-VPG representation of the graph $H_{G,\mathcal{C},k}$.

If $t > k$, then we let $H_{G,\mathcal{C},k}$ be an arbitrary B_0-VPG graph with chromatic number greater than k (for example, a complete graph with more than k vertices). So from now on, we assume that $t \leq k$. For every vertex $v \in V(G) \setminus T$, define the horizontal line segment $L_v = [\theta_1(v), \theta_2(v)] \times \{0\}$. For each $i \in \{1, 2, \ldots, t\}$, define horizontal line segments $L_{u_i} = [-i, \theta_2(v)] \times \{0\}$ and $L'_{u_i} = [\theta_1(v), 2\pi + i] \times \{0\}$. Further, define t horizontal line segments $A_1, A_2, \ldots, A_t$, where $A_i = [-i, 2\pi+i] \times \{1\}$ for each $i \in \{1, 2 \ldots, t\}$. Finally, for each $i \in \{1, 2, \ldots, t\}$, define a collection $\mathcal{B}_i$ of $k + i - t - 1$ vertical line segments each of which is $\{-i\} \times [0, 1]$, and a collection $\mathcal{B}'_i$ of $k + i - t - 1$ line segments each of which is $\{2\pi + i\} \times [0, 1]$. Clearly, the line segments in $\{L_v \colon v \in V(G)\} \cup (\bigcup_{i=1}^{t} \{L_{u_i}, L'_{u_i}, A_i\} \cup \mathcal{B}_i \cup \mathcal{B}'_i)$ form a 2-row B_0-VPG representation of a graph, and this graph is what we define to be $H_{G,\mathcal{C},k}$.

Lemma 3.5.1. *The graph G is k-colorable if and only if $H_{G,\mathcal{C},k}$ is k-colorable.*

Proof. For convenience, we will call the vertices of $H_{G,\mathcal{C},k}$ by their corresponding line segments. Furthermore, we will use the terms and notations used to describe the construction of $H_{G,\mathcal{C},k}$ inside this proof as well.

Observe that if t is greater than k, then both G and $H_{G,\mathcal{C},k}$ are not k-colorable. Thus we may assume that $t \leq k$.

Claim. For each $i \in \{1, 2, \ldots, t\}$, the line segments L_{u_i}, A_i and L'_{u_i} get the same color in any k-coloring of $H_{G,\mathcal{C},k}$.

We show this by induction on $t-i$. If $t-i = 0$, i.e. $i = t$, then $\mathcal{B}_i \cup \{L_{u_i}\}$ and $\mathcal{B}_i \cup \{A_i\}$ form cliques of size k in $H_{G,\mathcal{C},k}$, implying that L_{u_i} and A_i have the same color in any k-coloring of $H_{G,\mathcal{C},k}$. In the same way, since $\mathcal{B}'_i \cup \{A_i\}$ and $\mathcal{B}'_i \cup \{L'_{u_i}\}$ form cliques of size k in $H_{G,\mathcal{C},k}$, A_i and L'_{u_i} have the same color in any k-coloring of $H_{G,\mathcal{C},k}$. Suppose that $t - i > 0$, i.e., $1 \leq i < t$. Consider any k-coloring of $H_{G,\mathcal{C},k}$. By the induction hypothesis, for each $i < j \leq t$, L_{u_j}, A_j and L'_{u_j} have the same color, implying that the vertices in the sets $\{L_{u_t}, L_{u_{t-1}}, \ldots, L_{u_{i+1}}\}$, $\{A_{u_t}, A_{u_{t-1}}, \ldots, A_{u_{i+1}}\}$, and $\{L'_{u_t}, L'_{u_{t-1}}, \ldots, L'_{u_{i+1}}\}$ are all colored with the same $t-i$ colors. Since $\{L_{u_t}, L_{u_{t-1}}, \ldots, L_{u_i}\} \cup \mathcal{B}_i$ and $\{A_t, A_{t-1}, \ldots, A_i\} \cup \mathcal{B}_i$ are both cliques of size k in $H_{G,\mathcal{C},k}$, this implies that L_{u_i} and A_i have the same color. Similarly, since $\{A_t, A_{t-1}, \ldots, A_i\} \cup \mathcal{B}'_i$ and $\{L'_{u_t}, L'_{u_{t-1}}, \ldots, L'_{u_i}\} \cup \mathcal{B}'_i$ are both cliques of size k in $H_{G,\mathcal{C},k}$, we get that L'_{u_i} and A_i have the same color. This proves the claim.

From the claim above, it is clear that given any k-coloring of $H_{G,\mathcal{C},k}$, one can generate a valid k-coloring of G by giving each vertex v in $V(G) \setminus T$ the color of L_v and each vertex u_i in $\{u_1, u_2, \ldots, u_t\}$ the color of L_{u_i}. Also, given a k-coloring of G, one can color the line segments L_v, for each $v \in V(G) \setminus T$, with the color of v, the line segments L_{u_i}, L'_{u_i} and A_i, for each $i \in \{1, 2, \ldots, t\}$, with the color of u_i, and then color the line segments in $\bigcup_{i=1}^{t} \mathcal{B}_i \cup \mathcal{B}'_i$ greedily to generate a valid k-coloring of $H_{G,\mathcal{C},k}$. □

Now we are ready to prove Theorem 3.5.1.

Proof of Theorem 3.5.1. Note that a 2-row B_0-VPG representation for the graph $H_{G,\mathcal{C},k}$ can be constructed in polynomial-time given a circular-arc graph G, a circular-arc representation $\mathcal{C}$ of it and an integer k as input. Lemma 3.5.1 shows that this is a polynomial-time reduction from the CIRCULAR ARC COLORING problem to the CHROMATIC NUMBER problem on 2-row B_0-VPGs. □

3.6 Concluding remarks and open problems

In this chapter, we studied different subclasses of 2-SRIG. A direction of further research could be to investigate the subclasses of 2-SRIGs and try to characterise these classes of graphs.

Question 3.6.1. *Develop a forbidden structure characterization and/or a polynomial-time recognition algorithm for any of the classes* $(\mathcal{I},\mathcal{U})$*-graphs,* $(\mathcal{P},\mathcal{P})$*-graphs,* $(\mathcal{E},\mathcal{E})$*-graphs, or* $(\mathcal{U},\mathcal{U})$*-graphs.*

In this chapter, we studied the complexity of the CHROMATIC NUMBER problem on 2-SIRGs and observed it to be NP-hard on 2-SRIGs. Observe that the chromatic number of any 2-SRIG is at most twice its clique number. However, this bound is not known to be tight. There exists a class of 2-SRIGs containing graphs G with arbitrarily large clique number satisfying the property that $\chi(G) = \frac{5\omega(G)}{4}$. To see this, consider the following graph G which appears in [106]. Take a cycle C of five vertices. Replace each vertex in C with a complete graph on k vertices (where k is a positive even number) and add all possible edges between vertices belonging to consecutive cliques. Observe that $\chi(G) = \frac{5\omega(G)}{4}$ and that G is a 2-SRIG. In fact G is also 2-row B_0-VPG graph.

Question 3.6.2. *Obtain tight upper bounds on the chromatic number in terms of the clique number for 2-SRIG and its subclasses.*

4

Recognising trees that are 2-SUIG

Contents

In this chapter, we study the graph class 2-SUIG introduced in Chapter 3. First, we recall some definitions. A *2-stabbed unit square inter-*

section representation $\mathcal{R}$ of a graph G is a collection of axis-parallel unit squares on the plane and two horizontal lines called *stab lines* such that each unit square in the collection corresponds to a vertex of G, intersects exactly one of the stab lines and two unit squares intersect if and only if the corresponding vertices are adjacent in G. A graph G is a *2-stabbable unit square intersection graph* or *2-SUIG*, if G has a 2-stabbed unit square intersection representation. In this chapter, we shall prove the following theorem.

Theorem 4.0.1. *For a tree T, there is an $O(|V(T)|)$ time algorithm to decide if T is a 2-SUIG.*

4.1 Chapter overview

In Section 4.2, we present some definitions that will be used through out this chapter. In Section 4.3 we prove some properties of trees that are 2-SUIG. Finally in Section 4.4 we present our algorithm for recognising trees that are 2-SUIG. Finally, we draw conclusions in Section 4.5.

4.2 Preliminaries

For an interval $I = [a, b]$, let $|I| = (b - a)$. A vertex of a tree T with degree more than 2 is a *branch vertex* of T. A *branch edge* is an edge incident to a branch vertex.

Let $\mathcal{R}$ be a 2-stabbed unit square intersection representation of a graph G and $y = a_1, y = a_2$ be the stab lines with $a_1 < a_2$. The horizontal line $y = a_2$ is the *top stab line* of $\mathcal{R}$ and the horizontal line $y = a_1$ is the *bottom stab line* of $\mathcal{R}$. For a vertex $u \in V(G)$, s_u shall denote the unit square in $\mathcal{R}$ corresponding to u. The pair (x_u, y_u) shall denote the left bottom corner of s_u. For two unit squares s_u and s_v, we have $s_u <_x s_v$ if $x_u < x_v$. An edge uv is a *bridge edge* if s_u and s_v intersect different stab lines. The vertices corresponding to the unit squares intersecting the top stab line of $\mathcal{R}$ are called *upper vertices* and the vertices corresponding to the unit squares intersecting the bottom stab line of $\mathcal{R}$ are called *lower vertices*. For a connected subgraph H of G, let $span(H) = \bigcup_{u \in V(H)} [x_u, x_u + 1]$.

4.3 Some properties of trees that are 2-SUIG

In this section, we prove some properties of trees that are 2-SUIG.

Observation 4.3.1. *If T is a unit square intersection graph, then the maximum degree of T is at most four.*

A graph H is a *subdivision* of a graph G if H can be obtained by replacing some edges of G with paths.

Lemma 4.3.1. *Let T be a tree with at most one branch vertex. The tree T is a 2-SUIG if and only if maximum degree of T is at most four.*

Proof. If T is a 2-SUIG then by Observation 4.3.1, the maximum degree of T is at most four. For the other direction, notice that if T has no branch vertices, then T is a path and therefore a 2-SUIG. When T contains exactly one branch vertex of degree at most four, then T must be a subdivision of $K_{1,3}$ or $K_{1,4}$. In this case also T is a 2-SUIG. □

An edge e of T is a *red edge* if each component of $T - \{e\}$ contains a $K_{1,3}$ as a subtree. It is possible that a tree with more than one branch vertex do not have a red edge. If T is a 2-SUIG, then the following lemma follows from Observation 4.3.1.

Lemma 4.3.2. *If T is a 2-SUIG and does not have any red edge, then the number of branch vertices in T is at most 5.*

Lemma 4.3.3. *If T is a 2-SUIG, then either T has no red edge or the set of red edges of T induce a connected path.*

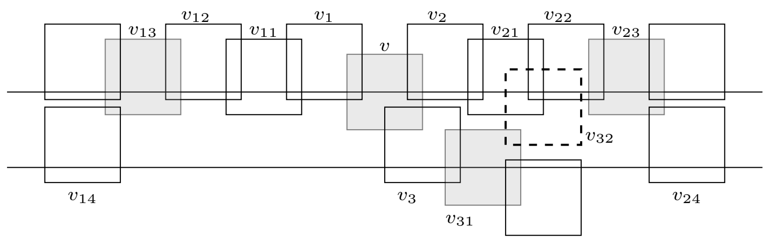


Figure 4.3.1: Illustration of proof of Lemma 4.3.3

Proof. Let T has at least one red edge and T' be the graph induced by all red edges. First we will show that T' is connected. Thus, assume that T' has at least two components T_1 and T_2. Then there is a path P in T connecting T_1 and T_2. Note that removing an edge e of P creates two components of T each of which contains a $K_{1,3}$ (a claw). Thus, e should be a red edge. Therefore, T' is connected.

We will show that T' is a path. Assume that v is a vertex of T' with degree at least 3. Also, let v_1, v_2 and v_3 be three neighbours of v in T'. Let $\mathcal{R}$ be any 2-stabbed unit square intersection representation $\mathcal{R}$ of T. Without loss of generality, we assume that s_{v_1} intersects the upper-left corner of s_v, s_{v_2} intersect the upper-right corner of s_v, and s_{v_3} intersects the lower-right corner of s_v. This implies that s_{v_1}, s_{v_2} intersect the top stab line while s_{v_3} intersects the bottom stab line (see Figure 4.3.1). As each component of $T - \{vv_1\}$ has a claw, there must be a path of the form $v_1v_{11}v_{12}...v_{1t}$ in T such that $s_{v_{1i}} <_x s_{v_1}$ for all $i \in [t]$ and $s_{v_{1t}}$ intersects the bottom stab line. Similarly, as each component of $T - \{vv_2\}$ has a claw, there must be a path of the form $v_2v_{21}v_{22}...v_{2t'}$ in T such that $s_{v_2} <_x s_{v_{2i}}$ for all $i \in [t']$ where $s_{v_{2t'}}$ intersects the bottom stab line. Moreover, as each component of $T - \{vv_3\}$ has a claw, there must be a path of the form $v_3v_{31}v_{32}...v_{3t''}$ in T where $s_{v_{3t''}}$ intersects the top stab line. But this contradicts the fact that $\mathcal{R}$ is a valid 2-stabbed unit square representation of T. Thus, T' must be a path. □

For a tree T, we shall say that T *has a red path* if and only if the set of red edges in T induce a path (which may be empty). A *maximal red path* is a red path that is not properly contained in another red path. Let $P = v_1v_2 \dots v_k$ be a maximal red path in T. The vertices v_1 and v_k are endpoints of P. Lemma 4.3.3 leads us to two cases: when T has a red path and when T does not have any red edge. To deal with both cases in a uniform framework, we construct the *extended red path* of a tree T as follows.

Let T be a tree such that the set of red edges is either empty or induce a connected path. If the red edges of T induces a path P, then construct the *extended red path* $A = a_1a_2...a_k$ by including the edge(s), that are not red, incident to the endpoint(s) of P that have degree two in T. In particular, if both the end points of P are branch vertices, then the extended red path A is identical to P. On the other hand, if T has no red edges, then distance between any two branch vertices is at most 2. Thus, there exist a vertex v in T whose closed neighbourhood $N[v]$ contains all the branch vertices of T. Choose (if not found to be unique) one such special vertex v. If v has degree two then consider the path uvw induced by the closed neighbourhood of v and call it the extended red path of T. If v does not have degree two, then the extended red path of T is the singleton vertex v. In any case, rename the vertices of the extended red path $A = a_1a_2...a_k$ so that we can speak about it in an uniform framework along with the case T having red edges. The vertices of the extended red path A are called the *red vertices* of T. See Figure 4.3.2(a) for an example.

Lemma 4.3.4. *Let T be a tree having an extended red path. A branch vertex of T is either a red vertex or is adjacent to a red vertex.*

Proof. If T has no red edge, then there exists a red vertex v in T such that all the branch vertices are in the closed neighbourhood $N[v]$ of v. Assume for contadiction that a red vertex v and a non-red branch vertex u are

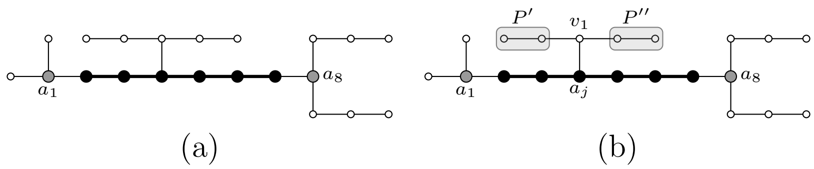


Figure 4.3.2: (a) The thick edges are the red edges and the path between a_1 and a_8 is the extended red path, (b) v_1 is an agent of a_j and P', P'' are tails of v_1.

connected by a path P containing at least one non-red vertex. Observe that, for each edge e in P, both components in $T - \{e\}$ contains a claw. This is a contradiction. □

The next lemma follows from Observation 4.3.1, Lemma 4.3.3 and 4.3.4.

Lemma 4.3.5. *Let T be a tree with at least one branch vertex and an extended red path A. If T is 2-SUIG then $T - V(A)$ induces a set of disjoint paths.*

Lemma 4.3.6. *Let T be a tree with an extended red path $A = a_1a_2 \dots a_k$ and a_i, a_{i+1} be two red vertices both having degree four in T. In any 2-stabbed unit square intersection representation $\mathcal{R}$ of T, a_i and a_{i+1} must intersect different stab lines.*

Proof. For contradiction assume that a_i, a_{i+1} are lower vertices. Without loss of generality assume $s_{a_i} <_x s_{a_{i+1}}$. Then $s_{a_{i+1}}$ intersects either the upper-right corner of s_{a_i} or the lower-right corner of s_{a_i}. If $s_{a_{i+1}}$ intersects the upper-right corner of s_{a_i}, then for some neighbour z of a_i, the squares $s_{a_i}, s_{a_{i+1}}, s_z$ will intersect each other in $\mathcal{R}$. This is a contradiction. We can argue similarly for the other cases as well. □

Now we introduce some more definitions. Let T be a tree with an extended red path A. The vertices adjacent to the red vertices of T that are not red are called *agents*. Observe that, if v is an agent then it must be adjacent to exactly one red vertex, say, u. In this case, we shall say that "v is an agent of u" (see Figure 4.3.2(b)).

If a tree T is a 2-SUIG, then due to Lemma 4.3.5, deleting all the red vertices from T we will induce a set $\mathcal{P}$ of disjoint paths. Each path $P \in \mathcal{P}$ contains exactly one vertex which is also an agent. Let v_1 be an agent of a red vertex a_j and contained in a path $P \in \mathcal{P}$. Also let $\{P', P''\} = P - \{v_1\}$. Then the path P' (and similarly P'') is called a *tail* of the agent v_1 (see Figure 4.3.2(b) for an example). Sometimes we will also use the term "tail P' of the red vertex a_j". Note that an agent has exactly two tails by allowing tails with zero vertices. Whenever an extended red path P of the tree T under consideration is clear form the context, the terms "red vertices", "agents", "tail of an agent", etc. are considered to be defined with respect to this extended red path.

Let $P = v_1 v_2 ... v_k$ be a path and $\mathcal{R}$ be a 2-stabbed unit square intersection representation of P. The path P is called a *folded path* if it has a degree two vertex u such that either $s_u <_x s_v$ for all $v \in V(P) \setminus \{u\}$ or $s_v <_x s_u$ for all $v \in V(P) \setminus \{u\}$. Note that a folded path has at least one bridge edge. The path P is a *right monotone* if $s_{v_1} <_x s_{v_2} <_x ... <_x s_{v_k}$ holds. Furthermore, a right monotone path P is *upper right monotone* (resp. *lower right monotone*) if all vertices of P are upper (resp. lower) vertices in $\mathcal{R}$. Similarly, P is a *left monotone* if $s_{v_k} <_x s_{v_{k-1}} <_x ... <_x s_{v_1}$ holds. The terminologies *upper left monotone paths* and *lower left monotone paths* are defined analogously. The path P is a *monotone* if P if right or left monotone.

Let $P = v_1 v_2 ... v_k$ be a path and $\mathcal{R}$ be a 2-stabbed unit square intersection representation of P such that P is a monotone path in $\mathcal{R}$. Observe that $\alpha(P) = \lceil \frac{k}{2} \rceil < |span(P)| \leq k$. The representation $\mathcal{R}$ is a *stretched representation* if $|span(P)| = k$ and is a *shrinked representation* if $|span(P)| = \alpha(P) + \epsilon$ where $0 < \epsilon << 1$.

Lemma 4.3.7. *If a tree T is a 2-SUIG, then there exists a 2-stabbed unit square intersection representation $\mathcal{R}$ of T where the extended red path of*

T is monotone and stretched.

Proof. Let $\mathcal{R}$ be a 2-stabbed unit square intersection representation of T having an extended red path $A = a_1a_2 \dots a_k$. If A is folded path in $\mathcal{R}$ then there is a vertex $u \in V(A)$ with $s_v <_x s_u$ for all $v \in V(A) \setminus \{u\}$. Then there will be two claws C_1, C_2 in two different components of $T - \{u\}$ with $s_v <_x s_u$ for all $v \in V(C_1) \cup V(C_2)$. But this configuration is not possible in any valid 2-stabbed unit square intersection representation of T. Hence A cannot be a folded path in $\mathcal{R}$.

Suppose A is not monotone in $\mathcal{R}$. Then without loss of generality we can assume that, there is a vertex $a_i \in V(A)$ such that s_{a_i} intersects the bottom stab line, $s_{a_{i+1}} <_x s_{a_i}$ and $s_{a_{i+1}} <_x s_{a_{i+2}}$ in $\mathcal{R}$. Then a_{i+1} and a_{i+2} must be on the top stab line. Let T_{i+1} be the component of T obtained by deleting the edge a_ia_{i+1} and contains a_{i+1}. There is a claw C_3 in T_{i+1} with $s_{a_{i+1}} <_x s_w$ for all $w \in V(C_3)$. As $\mathcal{R}$ is a valid representation of T, an agent z of a_i with $s_{a_i} <_x s_z$ cannot be a branch vertex. Hence the tail of z must be a lower right monotone path in $\mathcal{R}$. Therefore, we can translate the squares corresponding to the vertices of T_{i+1} to obtain a 2-stabbed unit square intersection representation of T where $s_{a_i} <_x s_{a_{i+1}} <_x s_{a_{i+2}}$. By performing the above operation for each vertex in A we can get an alternative 2-stabbed unit square intersection representation $\mathcal{R}'$ of T where A is monotone.

If A is not stretched in $\mathcal{R}'$ we apply the following procedure. Let $e = a_ia_{i+1}$ be an edge of the extended red path with $s_{a_i} <_x s_{a_{i+1}}$ in $\mathcal{R}'$. Let T_i and T_{i+1} be the components of $T - \{e\}$ containing a_i and a_{i+1}, respectively. Now translate the squares corresponding to the vertices of T_{i+1} to obtain a new 2-stabbed unit square intersection representation $\mathcal{R}''$ of T where $x_{a_{i+1}} = x_{a_i} + 1$. By performing this operation on every edge of A we can get an alternative 2-stabbed unit square intersection representation of T where A is monotone and stretched. □

Lemma 4.3.8. *If a tree T is a 2-SUIG, then there exists a 2-stabbed unit* *square intersection representation $\mathcal{R}$ of T where each tail is a shrinked monotone path and all its vertices have a common stab.*

Proof. Let $P = v_2v_3...v_{l-1}$ be a tail of an agent v_1. If all vertices of the tail P have a common stab then P must be monotone. In this case, we can obtain a new 2-stabbed unit square intersection representation of T where P is shrinked, and we are done. Now assume P has at least one bridge edge. Then any brige edge e of P divides the stab lines into two parts, left and right. Assume, without loss of generality, that s_{v_1} is in the left part. As there are no branch vertices in the tail, we do not have any vertex $w \in V(T) \setminus V(P)$ with s_w lying in the right part. Thus, we can modify $\mathcal{R}$ by such that all the vertices of P have a common stab and then obtain a new 2-stabbed unit square intersection representation of T where P is shrinked. □

Lemma 4.3.9. *Let T be tree which is a 2-SUIG and A be its extended red path. Then there exists a 2-stabbed unit square intersection representation of T where endpoints of any bridge edge is a branch vertex of T.*

Proof. By Lemma 4.3.7, there is a 2-stabbed unit square intersection representation $\mathcal{R}$ of T where A is stretched and monotone. Let $A = a_1a_2 \ldots a_k$ and without loss of generality assume A to be a right monotone path in $\mathcal{R}$. Let i be the smallest integer such that A have a bridge edge $e = a_ia_{i+1}$ in $\mathcal{R}$ but there is a $w \in \{a_i, a_{i+1}\}$ which is not a branch vertex. We shall apply the following procedures to obtain an alternative 2-stabbed unit square intersection representation of T where a_i and a_{i+1} have a common stab.

Consider the case when $w = a_{i+1}$. Let $S = \{v \in V(T) \colon s_{a_i} <_x s_v\}$. Let $T' = T[S]$ and $\mathcal{R}' = \{s_u\}_{u \in S}$ be the 2-stabbed unit square intersection representation of T' induced in $\mathcal{R}$. Note that $\mathcal{R}'$ contains $s_{a_{i+1}}$. Let $\mathcal{R}''$ be 2-stabbed unit square intersection representation of T' obtained by reflectinq the unit squares in $\mathcal{R}'$ with respect to the x-axis and having the $y = -z_2$ and $y = -z_1$ as stab lines. Now translate the unit squares in $\mathcal{R}''$ upwards until all the upper vertices (with respect to $\mathcal{R}''$) of T' intersects $y = z_2$ and all the lower vertices (with respect to $\mathcal{R}''$) intersect $y = z_1$. Note that a_i can have at most one degree 2 agent in T' and thus, that agent can have at most one tail. After what we did above, we can adjust the y-coordinates of that agent and its tail, if needed, to obtain a 2-stabbed unit square intersection representation of T where a_i and a_{i+1} have a common stab.

Consider the case when $w = a_i$. Let $S' = \{v \in V(T) \colon s_v <_x s_{a_{i+1}}\}$ and let T'' be the graph induced by S'. Now we apply similar procedure as above on T''. This will give a 2-stabbed unit square intersection representation of T where a_i and a_{i+1} have a common stab.

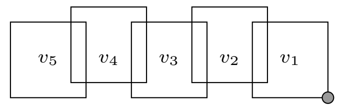
Now for each bridge edge of A that have a degree two vertex of T incident on it, we shall apply the above procedure inductively to get a 2-stabbed unit square intersection representation of T that satisy the statement of the lemma. □

4.4 The algorithm

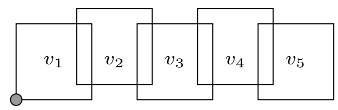
Now we shall describe our algorithm for deciding if a tree is a 2-SUIG. First we need the following definitions. Given a tree T, for the remainder of this section we shall assume that $\epsilon = \frac{1}{|V(T)|}$ and extended red path of T is $A = a_1 a_2 \dots a_k$. Also, whenever we refer to shrinked representation of a path P, we shall refer to a shrinked representation of P such that $|span(P)| = |\alpha(P)| + \epsilon^{|V(T)|}$. For a vertex $v \in V(T)$, let $d(v)$ denote the degree of v in T.

For an agent v of T, $lt(v)$ and $st(v)$ are the two tails of v where the number of vertices in $st(v)$ is at most that of $lt(v)$. In the remainder of this section, $|lt(v)|$ and $|st(v)|$ shall denote the number of vertices of $lt(v)$ and $st(v)$, respectively.

Let $P = v_1 v_2 \dots v_t$ be a path having at least one vertex. The *starting point* of a left monotone representation $\mathcal{R}$ of P is the point $(x_{v_1} + 1, y_{v_1})$



(a) starting point of a left monotone representation



(b) starting point of a right monotone representation.

Figure 4.4.1: Definition of starting point of a monotone representation of $P = v_1v_2v_3v_4v_5$.

(Figure 4.4.1(a)). Notice that the starting point of $\mathcal{R}$ is the lower-right corner of s_{v_1}. Similarly, the starting point of a right monotone representation $\mathcal{R}'$ of P is the point (x_{v_1}, y_{v_1}) (Figure 4.4.1(b)).

Now we shall define a *nice* representation of a path P. Let $\mathcal{R}$ be a shrinked monotone representation of P, v be a vertex of P and q, q' be point on the plane. Then $\mathcal{R}$ is a *nice*$-UL_{(q,q')}$ representation of P with respect to v if $\mathcal{R}$ is left (L) monotone, all vertices of P are upper (U) vertices in $\mathcal{R}$, $(x_v, y_v) = q$ and $y_u = y_{q'}$ for all $u \in V(P) \setminus \{v\}$ where $q' = (x_{q'}, y_{q'})$. Similarly, we define *nice*$-LL_{(q,q')}$,*nice*$-UR_{(q,q')}$ and *nice*$-LR_{(q,q')}$ representation of a path P with respect to a vertex v (Figure 4.4.2). When q and q' both equals to the starting point of $\mathcal{R}$, notice that the left bottom corner of all unit squares in $\mathcal{R}$ have the same y-coordinate as the starting point. In this case, we define $\mathcal{R}$ to be simply an *nice*$-UL$ representation of P. In other words, $\mathcal{R}$ is an *nice*$-UL$ representation of P if $\mathcal{R}$ is left monotone, all vertices of P are upper vertices in $\mathcal{R}$, and $y_u = y$ for all $u \in V(P)$ where (x, y) is the starting point of $\mathcal{R}$ Similarly, we can define *nice*$-LL$, *nice*$-UR$ and *nice*$-LR$ representation of P.

For a 2-stabbed unit square intersection representation $\mathcal{R}$ of a graph G, let $\Phi_t(\mathcal{R}) = \max\{x_u + 1\colon u \in V(G), s_u$ intersects the top stab line of $\mathcal{R}\}$, and $\Phi_b(\mathcal{R}) = \max\{x_u +$

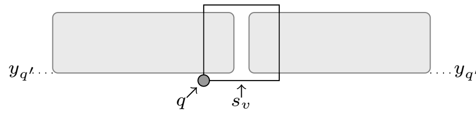


Figure 4.4.2: A nice-$UL_{(q,q')}$ representation of a path P with respect to a vertex $v \in V(P)$. Here $q' = (x_{q'}, y_{q'})$.

1: $u \in V(G), s_u$ intersects the bottom stab line of $\mathcal{R}\}$.

Definition 4.4.1. *For all $1 \leq i < k$, let T_i be the subtree of T induced by the vertices in $V(T') \cup \{a_{i+1}\}$ where T' is the connected component in $T - a_i a_{i+1}$ that contains a_i. Also define $T_k = T$.*

Let $\mathcal{R}_0$ be a 2-stabbed unit square intersection representation of the single vertex a_1 where $y = 0$ and $y = 1$ are the stab lines and $s_{a_1} = [1, 2] \times [-\epsilon, 1 - \epsilon]$.

Definition 4.4.2. *For a fixed $i \in \{1, 2, \ldots, k\}$, a 2-stabbed unit square intersection representation $\mathcal{R}$ is an* optimised representation of T_i *if $\mathcal{R}$ satisfies the following.*

1. *the path $a_1 a_2 \ldots a_i$ in T_{i-1} is a stretched left monotone path in $\mathcal{R}_{i-1}$,*
2. *when a vertex u of T_{i-1} is a lower vertex we have $y_u < 0$,*
3. *when a vertex u of T_{i-1} is an upper vertex we have $0 < y_u < 1$,*
4. *for any 2-stabbed unit square intersection representation $\mathcal{S}'' = \{s''_u\}_{u \in V(T_{i-1}) \setminus \{a_i\}}$ of $T_{i-1} - \{a_i\}$ which satisfies the following properties:*
    - *the set of upper and lower vertices in $\mathcal{S}''$ are same as that of $\mathcal{S}'$ and*
    - *$s''_v = s'_v$ for all $v \in \{a_1, a_2, \ldots, a_{i-1}\}$*

    we have that $\Phi_t(\mathcal{S}') - \Phi_t(\mathcal{S}'') < \epsilon$ and $\Phi_b(\mathcal{S}') - \Phi_b(\mathcal{S}'') < \epsilon$.

Below we give a sketch for our $O(|V(T)|)$-time algorithm to check if T has an optimised representation.

4.4.1 Sketch of our algorithm

(a) Check if maximum degree of T is at most 4. If not, then report T is not a 2-SUIG (by Observation 4.3.1).

(b) Check if there at most one branch vertex in T. If yes, then report T is a 2-SUIG (by Lemma 4.3.1).

(c) Find out if T has an extended red path or not. If not, report that T is not a 2-SUIG (by Lemma 4.3.5). Otherwise let $A = a_1 a_2 \ldots a_k$ be the extended red path of T and $T_1, T_2 \ldots, T_k$ be subtrees of T as defined in Definition 4.4.1.

(d) Let $\mathcal{R}_0$ be a 2-stabbed unit square intersection representation of the single vertex a_1 where $y = 0$ and $y = 1$ are the stab lines and s_{a_1} intersects the bottom stab line.

(e) For $i = 1$ to k find out the optimised representations $\mathcal{R}_i$ of T_i. If we fail to find such a representation for some $i \in \{1, 2, ..., k\}$, then report T is not a 2-SUIG. Otherwise return $\mathcal{R}_k$.

In the next section, we show how to decide if there is an optimised representation of T_1 when $k \neq 1$.

4.4.2 Optimised representation of T_1 when $k \neq 1$

Depending on the degree of a_1, a_2 in T and the lengths of the tails of their agents, we shall describe a procedure to decide if there is an optimised representation of T_1. Depending on the degree of a_1, the degree of a_2 and the lengths of the tails of the agents of a_1 we consider several cases. Since the degrees of both a_1, a_2 are at most 4 and the each agent of a_1 has at most two tails, the total number of cases that need to be considered are constant. Moreover, for each case to decide if there is an optimised representation of T_1 we need only constant amount of time. If the process described below fails, we shall report that T is not a 2-SUIG. Otherwise we shall construct an optimised representation $\mathcal{R}_1$ of T_1.

Case 1: $d(a_1) = 4, d(a_2) = 4$. In this case, we shall first take a unit square s_{a_2} such that $x_{a_2} = x_{a_1} + 1$ and $y_{a_2} = y_{a_1} + 1$. Notice that $s_{a_2} = [2, 3] \times [1 - \epsilon, 2 - \epsilon]$ and therefore s_{a_2} intersects ℓ_1. Then we shall find if there are three agents z_1, z_2, z_3 of a_1 such that there are unit squares s_{z_1}, s_{z_2} and s_{z_3} satisfying the following properties:

(a) $s_{z_1} = [x_{a_1} - 1, x_{a_1}] \times [y_{a_1} - \epsilon, y_{a_1} + 1 - \epsilon]$, $s_{z_2} = [x_{a_1} - 1 + \epsilon, x_{a_1} + \epsilon] \times [y_{a_1} + 1, y_{a_1} + 2]$ and $s_{z_3} = [x_{a_1} + \epsilon, x_{a_1} + 1 + \epsilon] \times [y_{a_1} - \epsilon, y_{a_1} + 1 - \epsilon]$,

(b) if $|lt(z_1)| \geq 1$, then there is a nice-LL representation $\mathcal{I}_{11}$ of $lt(z_1)$ whose starting point is $(x_{z_1}, y_{z_1} - \epsilon)$,

(c) if $|st(z_1)| \geq 1$, then there is a nice-UL representation $\mathcal{I}_{12}$ of $st(z_1)$ whose starting point is $(x_{z_1}, y_{z_1} + 1)$ (the top-left corner of s_{z_1}),

(d) $|st(z_3)| = 0$ and if $|lt(z_3) \geq 1|$ then there is a nice-LR representation $\mathcal{I}_{31}$ of $lt(z_3)$ whose starting point is $(x_{z_3} + 1, y_{z_3})$ (the bottom-right corner of $s(z_3)$),

(e) $|st(z_2)| \leq 1$ and moreover, if $|st(z_1)| \geq 1$ then $|lt(z_2)| \leq 1$ and $|st(z_2)| = 0$,

(f) $|lt(z_2)| \geq 1$, then there is a nice-UL representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point q satisfies the following property

- if $|st(z_1)| \geq 1$ then $q = (x_{z_2} + 1 - \epsilon^2, y_{z_2} + \epsilon^2)$ (Figure 4.4.3(a)), otherwise $q = (x_{z_2}, y_{z_2})$ (Figure 4.4.3(b)).

(g) if $|st(z_2)| = 1$, then there is a nice-UR representation $\mathcal{I}_{22}$ of $st(z_2)$ whose starting point is $(x_{z_2} + \epsilon^2, y_{z_2} + \epsilon^2)$.

Case 2: $d(a_1) = 4, d(a_2) = 3$. In this case, we shall first take a unit square s_{a_2} such that $x_{a_2} = x_{a_1} + 1$ and $y_{a_2} = y_{a_1} - \epsilon$. Then we shall find if there are three agents z_1, z_2, z_3 of a_1 such that there are unit squares s_{z_1}, s_{z_2} and s_{z_3} satisfying the following properties (See Figure 4.4.4):

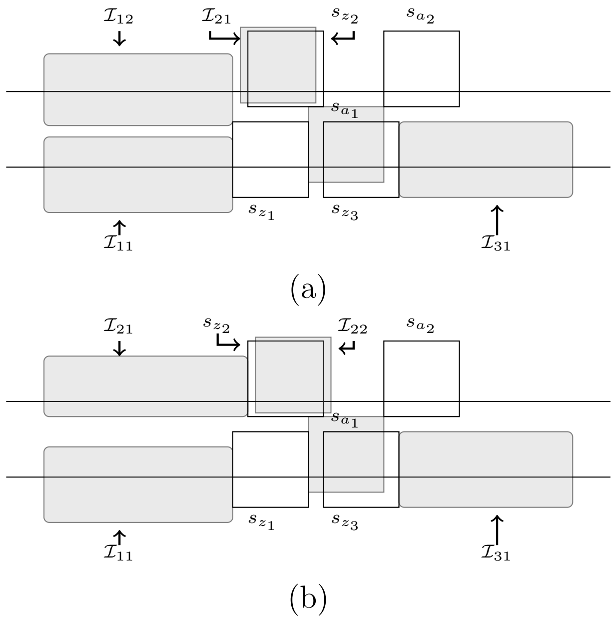


Figure 4.4.3: Case 1 for representation of T_1 when $k \neq 1$. (a) $|st(z_1)| \geq 1$, (b) $|st(z_1)| = 0$.

(a) $s_{z_1} = [x_{a_1}-1, x_{a_1}]\times[y_{a_1}-\epsilon, y_{a_1}+1-\epsilon]$, $s_{z_2} = [x_{a_1}-1+\epsilon, x_{a_1}+\epsilon]\times[y_{a_1}+1, y_{a_1}+2]$ and $s_{z_3} = [x_{a_1}+2\epsilon, x_{a_1}+1+2\epsilon]\times[y_{a_1}+1, y_{a_1}+2]$,

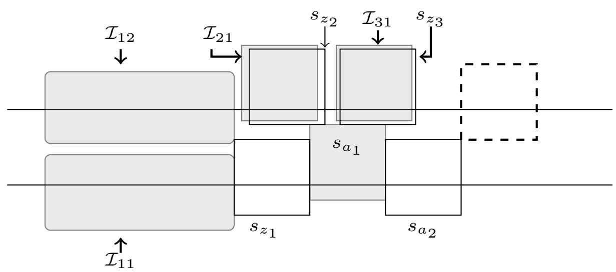
(b) the properties (b), (c), (e), (f) and (g) of Case 1,

(c) $|lt(z_3)| \leq 1$, and

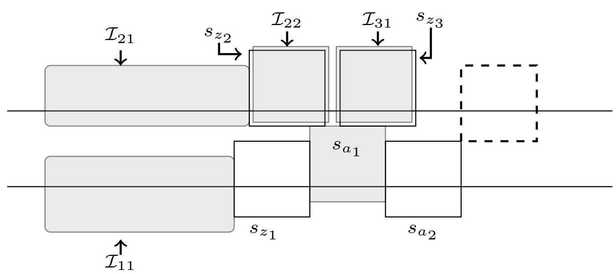
(d) if $|lt(z_3)| = 1$, then there is a there is a nice-UR representation $\mathcal{I}_{31}$ of $lt(z_3)$ whose starting point is $(x_{z_3} - \epsilon^2, y_{z_3} + \epsilon^2)$.

Case 3: $d(a_1) = 4, d(a_2) = 2$. In this case, we shall first take a unit square s_{a_2} such that $x_{a_2} = x_{a_1} + 1$ and $y_{a_2} = y_{a_1} - \epsilon$. Then we shall find if there are three agents z_1, z_2, z_3 of a_1 such that there are unit squares s_{z_1}, s_{z_2} and s_{z_3} satisfying the following properties (See Figure 4.4.5).

(a) $s_{z_1} = [x_{a_1}-1, x_{a_1}]\times[y_{a_1}-\epsilon, y_{a_1}+1-\epsilon]$, $s_{z_2} = [x_{a_1}-1+\epsilon, x_{a_1}+\epsilon]\times[y_{a_1}+1, y_{a_1}+2]$ and $s_{z_3} = [x_{a_1}+2\epsilon, x_{a_1}+1+2\epsilon]\times[y_{a_1}+1, y_{a_1}+2]$,



(a)



(b)

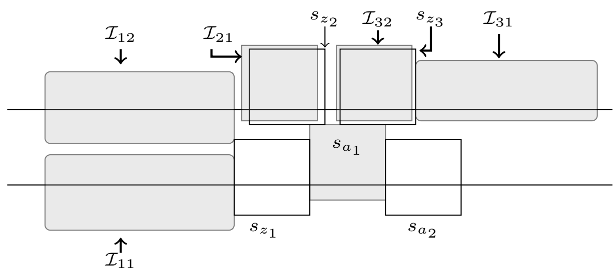
Figure 4.4.4: Case 2 for representation of T_1 when $k \neq 1$. (a) $|st(z_1)| \geq 1$, (b) $|st(z_1)| = 0$. In both figures, the dotted square represents a neighbour of a_2. Since, $d(a_2) = 3$, such a vertex always exists.

(b) the properties (b), (c), (e), (f) and (g) of Case 1,

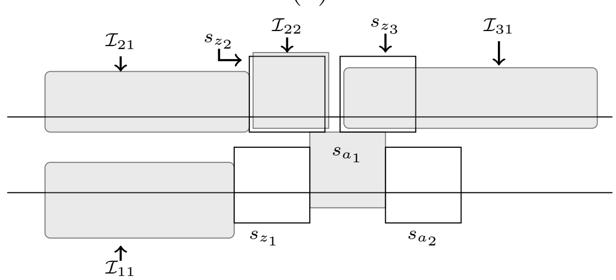
(c) $|st(z_3)| \leq 1$ and if $|st(z_3)| = 1$ then there is a there is a nice-UR representation $\mathcal{I}_{32}$ of $st(z_3)$ whose starting point is $(x_{z_3} - \epsilon^2, y_{z_3} + \epsilon^2)$ (See Figure 4.4.5(a)).

(d) if $|lt(z_3)| \geq 1$

- if $|st(z_3)| = 0$ and $|lt(z_3)| = 1$ then there is a nice-UR representation $\mathcal{I}_{31}$ of $lt(z_3)$ whose starting point is $(x_{z_3} - \epsilon^2, y_{z_3} + \epsilon^2)$
- if $|st(z_3)| = 0$ and $|lt(z_3)| \geq 2$ then there is a nice$-UR_{(q,q')}$ representation $\mathcal{I}_{31}$ of $lt(z_3) \cup \{z_3\}$ with respect to z_3 where $q = (x_{z_3}, y_{z_3})$ and $q' = (x_{z_3} + \epsilon^2, y_{z_3} + \epsilon^2)$ (See Figure 4.4.5(b)).
- if $|st(z_3)| = 1$ then there is a nice-UR representation $\mathcal{I}_{31}$



(a)



(b)

Figure 4.4.5: Case 3 for representation of T_1 when $k \neq 1$. (a) $|st(z_1)| \geq 1$, (b) $|st(z_1)| = 0$.

of $lt(z_3)$ whose starting point is $(x_{z_3} + 1, y_{z_3} + \epsilon^2)$

Case 4: $d(a_1) = 3, d(a_2) = 4$. In this case, we shall first take a unit square s_{a_2} such that $x_{a_2} = x_{a_1} + 1$ and $y_{a_2} = y_{a_1} + \epsilon^2$. Then we shall find if there are two agents z_1, z_2 of a_1 such that there are unit squares s_{z_1} and s_{z_2} satisfying the following properties (See Figure 4.4.6):

(a) $s_{z_1} = [x_{a_1} - 1, x_{a_1}] \times [y_{a_1} - \epsilon, y_{a_1} + 1 - \epsilon]$ and $s_{z_2} = [x_{a_1} - 1 + \epsilon, x_{a_1} + \epsilon] \times [y_{a_1} + 1, y_{a_1} + 2]$,

(b) properties (b), (c), (e), (f) and (g) of case 1.

Case 5: $d(a_1) = 3, d(a_2) = 3$. In this case, we shall first take a unit square s_{a_2} such that $x_{a_2} = x_{a_1} + 1$ and $y_{a_2} = y_{a_1} - \epsilon$. Then we shall find

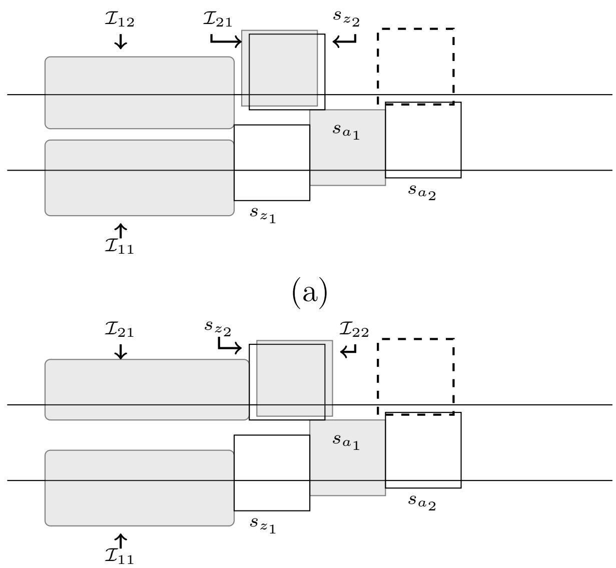


Figure 4.4.6: Case 4 for representation of T_1 when $k \neq 1$. (a) $|st(z_1)| \geq 1$, (b) $|st(z_1)| = 0$. In both figures, the dotted square represents an agent of a_2. Since, $d(a_2) = 4$, such an agent always exists.

if there are two agents z_1, z_2 of a_1 such that there are unit squares s_{z_1} and s_{z_2} satisfying the following properties:

(a) $s_{z_1} = [x_{a_1} - 1, x_{a_1}] \times [y_{a_1} - \epsilon, y_{a_1} + 1 - \epsilon]$ and $s_{z_2} = [x_{a_1} - 1 + \epsilon, x_{a_1} + \epsilon] \times [y_{a_1} + 1, y_{a_1} + 2]$,

(b) properties (b) and (c) of case 1,

(c) if $|st(z_1)| \geq 1$ then $|lt(z_2)| + |st(z_2)| \leq 3$,

(d) if $|st(z_1)| = 0$, then $|st(z_2)| \leq 2$,

(e) if $|st(z_2)| \geq 1$ then

- if $|st(z_1)| \geq 1$ then there is a nice-UL representation $\mathcal{I}_{22}$ of $st(z_2)$ whose starting point is $(x_{z_2} + 1 - 2\epsilon^2, y_{z_2} + \epsilon^2)$ (see Figure 4.4.7(a)),
- otherwise, there is a nice$-UR_{(q,q')}$ representation $\mathcal{I}_{22}$ of $st(z_2) \cup \{z_2\}$ with respect to z_2 where $q = (x_{z_2}, y_{z_2})$ and

$q' = (x_{z_2} + \epsilon^2, y_{z_2} + \epsilon^2)$ (see Figure 4.4.7(b)),

(f) if $|lt(z_2)| \geq 1$, then

- if $|st(z_1)| \geq 1$ then
  - if $|st(z_2)| = 1$ then there is a nice-UR representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is $(x_{z_2} + 1, y_{z_2} + \epsilon^2)$ (see Figure 4.4.7(a))
  - if $|st(z_2)| = 0$ and $|lt(z_2)| \geq 2$ then there is a nice$-UR_{(q,q')}$ representation $\mathcal{I}_{21}$ of $lt(z_2) \cup \{z_2\}$ with respect to z_2 where $q = (x_{z_2}, y_{z_2})$ and $q' = (x_{z_2} + \epsilon^2, y_{z_2} + \epsilon^2)$,
  - if $|st(z_2)| = 0$ and $|lt(z_2)| = 1$ then there is a nice-UL representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is $q = (x_{z_2} - \epsilon^2, y_{z_2} + \epsilon^2)$
- otherwise, there is a nice-UL representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is (x_{z_2}, y_{z_2}) (see Figure 4.4.7(b)),

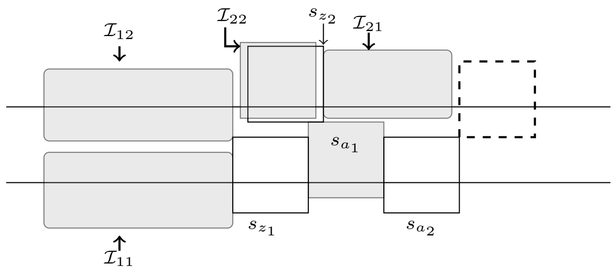
Case 6: $d(a_1) = 3, d(a_2) = 2$. In this case, we shall first take a unit square s_{a_2} such that $x_{a_2} = x_{a_1} + 1$ and $y_{a_2} = y_{a_1} - \epsilon$. Then we shall find if there are two agents z_1, z_2 of a_1 such that there are unit squares s_{z_1} and s_{z_2} satisfying the following properties:

(a) properties (b), (c) of case 1.

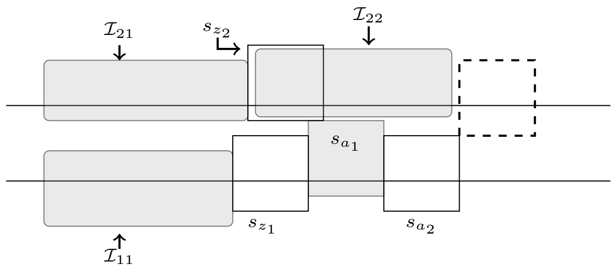
(b) if $|st(z_1)| \geq 1$ then $|st(z_2)| \leq 1$,

(c) if $|st(z_2)| \geq 1$ then

- if $|st(z_1)| \geq 1$ then there is a nice-UL representation $\mathcal{I}_{22}$ of $st(z_2)$ whose starting point is $(x_{z_2} + 1 - \epsilon^2, y_{z_2} + \epsilon^2)$ (see Figure 4.4.8(a)),
- otherwise, there is a nice$-UR_{(q,q')}$ representation $\mathcal{I}_{22}$ of $st(z_2) \cup \{z_2\}$ with respect to z_2 where $q = (x_{z_2}, y_{z_2})$ and $q' = (x_{z_2} + \epsilon^2, y_{z_2} + \epsilon^2)$ (see Figure 4.4.8(b)),



(a)



(b)

Figure 4.4.7: Case 5 for representation of T_1 when $k \neq 1$. (a) $|st(z_1)| \geq 1$, (b) $|st(z_1)| = 0$. In both figures, the dotted square represents a neighbour of a_2. Since, $d(a_2) = 3$, such a vertex always exists.

(d) if $|lt(z_2)| \geq 1$, then

- if $|st(z_1)| \geq 1$ then
  - if $|st(z_2)| = 1$ then there is a nice-UR representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is $(x_{z_2} + 1, y_{z_2} + \epsilon^2)$ (see Figure 4.4.8(a))
  - if $|st(z_2)| = 0$ and $|lt(z_2)| \geq 2$ then there is a nice-UR representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is $q = (x_{z_2} + \epsilon^2, y_{z_2} + \epsilon^2)$
  - if $|st(z_2)| = 0$ and $|lt(z_2)| = 1$ then there is a nice-UL representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is $q = (x_{z_2} - \epsilon^2, y_{z_2} + \epsilon^2)$
- otherwise, there is a nice-UL representation $\mathcal{I}_{21}$ of $lt(z_2)$ whose starting point is (x_{z_2}, y_{z_2}) (see Figure 4.4.8(b)),

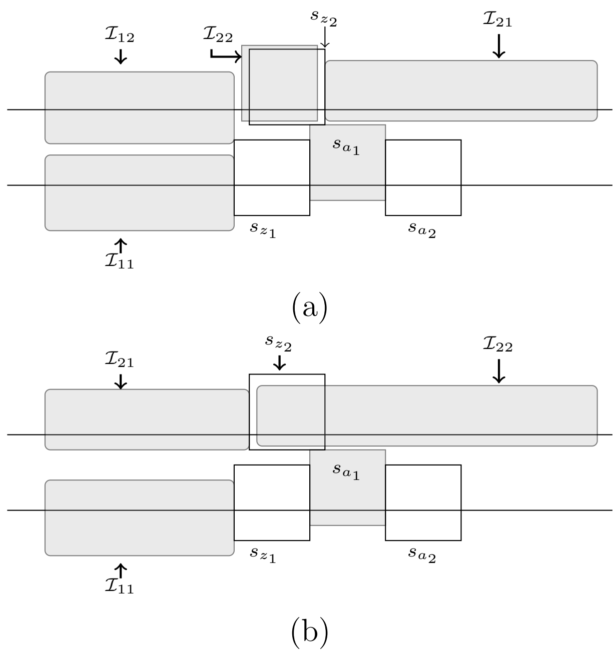


Figure 4.4.8: Case 6 for representation of T_1 when $k \neq 1$. (a) $|st(z_1)| \geq 1$, (b) $|st(z_1)| = 0$.

Let $\mathcal{R}_1 = \mathcal{R}_0 \cup \{s_{a_2}, s_{z_1}, s_{z_2}, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{21}, \mathcal{I}_{22}, \mathcal{I}_{31}, \mathcal{I}_{32}\}$. It is not difficult to verify that if T is a 2-SUIG then all the properties of exactly one of the above case must be true and indeed $\mathcal{R}_1$ is a 2-stabbed unit square intersection representation of T_1. Moreover, $\mathcal{R}_1$ can be obtained in $O(|V(T_1|)$ time. Observe that $\mathcal{R}_1$ is an optimised representation of T_1.

4.4.3 Optimised representation of T_i for $1 < i \leq k$

Using analogous case analysis as above, we can check whether there are optimised representations of $T_2, T_3, \ldots, T_k$. For each i, we shall take into consideration the degree of a_i, the degree of a_{i+1} and the lengths of all tails of all agents a_i. Recall that the degree of a_i is at most 4 and each agent of a_i has at most two tails. Hence we need to consider only constant number of cases for each $1 < i \leq k$ to decide if there is an optimised representation of T_i. For each $1 < i \leq k$, the total time taken by the algorithm is $O(|V(T_i)|)$. Hence the total running time of the algorithm is $O(|V(T)|)$. This completes the proof of Theorem 4.0.1.

4.5 Conclusion and open problems

In this chapter, we gave a linear time algorithm to recognise trees that are 2-SUIG. However, the complexity of recognising trees that are unit square intersection graphs is still unkown. A simpler algorithm to recognise trees that are 2-SUIG would be interesting.

5

Dominating set of stabbed rectangle overlap graphs

Contents

In this chapter, we shall study the Minimum Dominating Set (MDS) problem on rectangle overlap graphs and its subclasses. Recall that, given a set of rectangles, $\mathcal{C}$, the *overlap graph* G of $\mathcal{C}$ is the graph, whose vertices correspond to the elements of $\mathcal{C}$, and two vertices are joined by an edge if and only if the boundaries of the corresponding rectangles have a nonempty intersection. Here G is called a *rectangle overlap* graph and $\mathcal{C}$ is a *rectangle overlap representation* of G. A *dominating set* of an undirected graph G is a subset D of vertices such that each vertex in $V(G) \setminus D$ is adjacent to some vertex in D. The Minimum Dominating Set (MDS) problem is to find a minimum cardinality dominating set of a graph G.

Linear programs are problems that can be expressed as

$$\begin{array}{ll} \text{Minimize} & \mathbf{c}^T\mathbf{x} \\ \text{Subject to} & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ \text{and} & \mathbf{x} \geq \mathbf{0} \end{array}$$

where $\mathbf{x}$ represents the vector of *variables*, $\mathbf{c}$ and $\mathbf{b}$ are vectors of (known) *coefficients*, A is a (known) matrix of coefficients, and $(\cdot)^T$ is the matrix transpose. The expression to be maximized or minimized is called the *objective function*.

If all of the variables are required to be integers, then the problem is called an *integer linear programming* (ILP) problem. The *relaxation* of an integer linear program is the linear program obtained by removing the integrality constraint of each variable.

The *integrality gap* is the maximum ratio between the optimum solution of the integer program and of its relaxation. In an instance of a minimization problem, if the real minimum (the minimum of the integer problem) is M_{int}, and the relaxed minimum (the minimum of the linear programming relaxation) is M_{frac}, then the integrality gap of that instance is $\frac{M_{int}}{M_{frac}}$. For an integer linear program Q, $OPT(Q)$ denote the optimum cost of the objective funtion. For a linear program Q_l, $OPT(Q_l)$ is defined analogously.

5.1 Chapter overview

In Section 5.2, we show that, assuming the *Unique Games Conjecture* [110] to be true, it is not possible to have a polynmial time $(2-\epsilon)$-approximation algorithm for the MDS problem on rectangle overlap graphs, even if a rectangle overlap representation is given as input (Theorem 5.2.1).

A set of rectangles is *stabbed* if all rectangles in the set intersects a common straight line. A rectangle overlap representation $\mathcal{R}$ of a graph G is a *stabbed rectangle overlap representation* if $\mathcal{R}$ is stabbed. A graph G is a *stabbed rectangle overlap graph* if G has a stabbed rectangle overlap representation. In Section 5.7, we give a 768-approximation algorithm for the MDS problem on stabbed rectangle overlap graphs. To prove the above result of this chapter, first we need to prove two lemmas. The first lemma is about the *stabbing segment with rays* (SSR) problem and the second lemma is about the *stabbing rays with segment* (SRS) problem.

In the SSR problem, the inputs are a set of disjoint leftward-directed horizontal rays and a set of disjoint vertical segments. The objective is to select a minimum number of leftward-directed horizontal rays that intersect all vertical segments. Throughout this chapter, we let $\mathcal{SSR}(R,V)$ denote an SSR instance where R is a given set of disjoint leftward-directed horizontal rays and V is a given set of disjoint vertical segments. Using a novel "token passing" based *iterative rounding* scheme [118], we observe the following lemma in Section 5.3.

Lemma 5.1.1. *Let R be a set of leftward-directed rays and V be a set of disjoint vertical segments. Let $\mathcal{C}$ be an ILP formulation of an $\mathcal{SSR}(R,V)$ instance. There is an $O((n+m)\log(n+m))$-time algorithm to compute* *a set $D \subseteq R$ which gives a feasible solution of $\mathcal{C}$ and $|D| \le 2 \cdot OPT(\mathcal{C}_l)$ where $n = |R|, m = |V|$ and $\mathcal{C}_l$ is the relaxed LP formulation of $\mathcal{C}$.*

As a consequence of the above Lemma 5.1.1, we have a subquadratic 2-approximation algorithm for the SSR problem.

Theorem 5.1.1. *There is an $O((n+m)\log(n+m))$-time 2-approximation algorithm for the* SSR *problem where n and m are the number of horizontal rays and vertical segments, respectively.*

In the SRS problem, the inputs are a set of disjoint leftward-directed horizontal rays and a set of disjoint vertical segments. The objective is to select a minimum number of segments that intersect all leftward-directed horizontal rays. Throughout this chapter, we let $\mathcal{SRS}(R, V)$ denote an SRS instance where R is a given set of disjoint leftward-directed horizontal rays and V is a given set of disjoint vertical segments. We observe the following lemma in Section 5.4.

Lemma 5.1.2. *Let $\mathcal{C}$ be an ILP formulation of an $\mathcal{SRS}(R, V)$ instance. There is an $O(n \log n)$ time algorithm to compute a set $D \subseteq V$ which gives a feasible solution of $\mathcal{C}$ and $|D| \le 2 \cdot OPT(\mathcal{C}_l)$ where $n = |V|$ and $\mathcal{C}_l$ is the relaxed LP formulation of $\mathcal{C}$.*

Before we can prove our main result, we use Lemma 5.1.1 and 5.1.2 to prove upper bounds on the integrality gap of the following optimisation problems.

1. **The *local vertical segment covering* (LVSC) problem:** In this problem, the inputs are a set H of disjoint horizontal segments intersecting a common straight line and a set V of disjoint vertical segments. The objective is to select a minimum number of horizontal segments that intersect all vertical segments. Throughout this article, we let $\mathcal{LVSC}(V, H)$ denote an LVSC instance.

2. **The *local horizontal segment covering* (LHSC) problem:** In this problem, the inputs are a set H of disjoint horizontal segments all intersecting a common straight line and a set V of disjoint vertical segments. The objective is to select a minimum number of vertical segments that intersect all horizontal segments. Throughout this article, we let $\mathcal{LHSC}(V, H)$ denote an LHSC instance.

We note that Bandyapadhyay and Mehrabi [14] considered restricted cases of the LVSC and the LHSC problem. They proved that LVSC problem remains NP-hard even if all horizontal segments in the input instance intersect a common vertical line. We also note that PTAS are known for both the LVSC and the LHSC problems [15].

In Section 5.5 and 5.6, we prove Lemma 5.1.3 and 5.1.4, respectively.

Lemma 5.1.3. *Let $\mathcal{C}$ be an ILP formulation of an $\mathcal{LVSC}(V, H)$ instance. There is an $O(n^5)$ time algorithm to compute a set $D \subseteq H$ which gives a feasible solution of $\mathcal{C}$ and $|D| \leq 8 \cdot OPT(\mathcal{C}_l)$ where $n = |V \cup H|$ and $\mathcal{C}_l$ is the relaxed LP formulation of $\mathcal{C}$.*

Lemma 5.1.4. *Let $\mathcal{C}$ be an ILP formulation of an $\mathcal{LHSC}(V, H)$ instance. There is an $O(n^5)$ time algorithm to compute a set $D \subseteq V$ which gives a feasible solution of $\mathcal{C}$ and $|D| \leq 8 \cdot OPT(\mathcal{C}_l)$ where $n = |V \cup H|$ and $\mathcal{C}_l$ is the relaxed LP formulation of $\mathcal{C}$.*

Finally, we draw conclusions in Section 5.8.

5.2 Hardness result

In this section, we prove the following theorem.

Theorem 5.2.1. *Assuming the* Unique Games Conjecture, *for any $\epsilon > 0$, it is not possible to have a polynomial time $(2 - \epsilon)$-approximation algorithm for the* MDS *problem on rectangle overlap graphs, even if a rectangle overlap representation is given as input.*

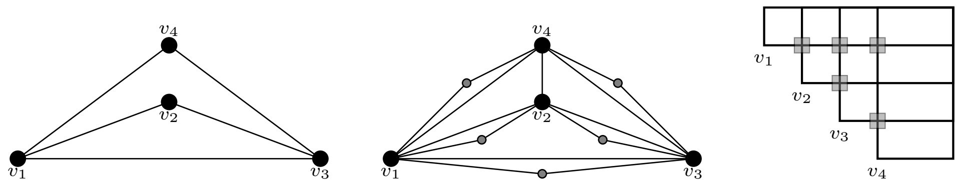


Figure 5.2.1: Reduction procedure for Theorem 5.2.1. (a) Input graph G, (b) The graph G' and (c) rectangle overlap representation of G'.

A *vertex cover* of a graph G is a subset C of $V(G)$ such that each edge in $E(G)$ has an endvertex which lies in C. The Minimum Vertex Cover problem is to find a minimum cardinality vertex cover of a graph. Assuming *Unique Games Conjecture* to be true, the Minimum Vertex Cover has no polynomial-time $(2-\epsilon)$-approximation algorithm for any $\epsilon > 0$ [109]. We shall reduce the Minimum Vertex Cover problem to the MDS problem on rectangle overlap graphs.

Given a graph G, construct another graph G' as follows. Define $V(G') = V(G) \cup E(G)$. Define $E(G') = \{uv\colon u, v \in V(G)\} \cup \{ue\colon u \in V(G), e \in E(G)$ and u is an endvertex of e in $G\}$. We have the following observation

Observation 5.2.1. *The graph G has a vertex cover of size k if and only if G' has a dominating set of size k.*

Proof. Let C be a vertex cover of G. Then at least one endpoint of every edge of G belongs to C. From construction of G', it follows that C is a dominating set of G'. Now let D be a dominating set of G'. Since E induces an independent set in G', we can assume that $D \subseteq V(G)$. Therefore, D is a vertex cover of G. □

Therefore, we will be done by showing that G' is a rectangle overlap graph. Let $V(G) = \{v_1, v_2, \ldots, v_n\}$ and for each $v_i \in V(G)$ define $R_{v_i} = [i, n+1] \times [-i, 0]$ (See Figure 5.2.1(c) for illustration).

Notice that, each vertex $u \in V(G') \setminus V(G)$, has degree two and is adjacent to exactly two vertices of $V(G)$. For each vertex $u \in V(G') \setminus V(G)$, introduce a rectangle R_u which overlaps only with R_{v_i} and R_{v_j} where $\{v_i, v_j\}$ is the set of vertices adjacent to u with $i < j$. This is possible as R_u can be kept around the unique intersection point of the bottom boundary of R_{v_i} and the left boundary of R_{v_j} (see Figure 5.2.1(c) for illustration). Formally, for each $u \in V(G') \setminus V(G)$, define $R_u = [p-\epsilon, p+\epsilon] \times [q-\epsilon, q+\epsilon]$ where $\epsilon = \frac{1}{|V(G)|}$ and (p, q) is the intersection point of the bottom boundary of R_{v_i} and the left boundary of R_{v_j}. Observe that the set of rectangles $\mathcal{R}' = \{R_{v_i} : v_i \in V(G)\} \cup \{R_u : u \in V(G') \setminus V(G)\}$ is a rectangle overlap representation of G'. This completes the proof of Theorem 5.2.1.

5.3 Integrality gap of the SSR problem

In this section, we shall prove Lemma 5.1.1 and Theorem 5.1.1 by showing that the integrality gap of the *stabbing segments with segments* (SSR) problem is at most two. Recall in the SSR problem, the inputs are a set of disjoint leftward-directed horizontal rays and a set of disjoint vertical segments. The objective is to select a minimum number of leftward-directed horizontal rays that intersect all vertical segments.

In this section, we represent a *leftward-directed horizontal ray* by simply a *ray* and a *vertical segment* by a *segment* in short. Let R be a set of disjoint rays and V be a set of disjoint vertical segments. We assume each segment intersects at least one ray in R and no two segments in V have the same x-coordinate.

To prove Lemma 5.1.1, first we present an iterative algorithm consisting of three main steps. The first step is to include all rays $r \in R$ in solution S whenever some segments in V intersect precisely a single ray r in that iterative step. In the next step, delete all segments intersecting any ray in S from V. In the final step, find a ray in $R \setminus S$ whose x-coordinate of

Algorithm 1 SSR-Algorithm

Input: A set R of leftward-directed rays and a set V of vertical segments.

Output: A subset of R that intersects all segments in V.

$T_r = \{r\}$ for each $r \in R$ and $i \leftarrow 1, V_0 \leftarrow V, R_0 \leftarrow R, S \leftarrow \emptyset, S_0 \leftarrow \emptyset$ $\triangleright$ Initialisation.
while $V_{i-1} \neq \emptyset$ **do**
    $S \leftarrow S \cup \{r\colon r \in R_{i-1}, r$ is critical after $(i-1)^{th}$ iteration$\}$ and $S_i \leftarrow S$.
        $\triangleright$ Critical ray collection.
    $V_i \leftarrow$ the set obtained by deleting all segments from V_{i-1} that intersect a ray in S_i.
    Find a $r \in R_{i-1} \setminus S_i$ whose x-coordinate of the right endpoint is the smallest.
    r discharges the token to its neighbours.
    $R_i \leftarrow$ The set obtained by deleting $\{r\} \cup S_i$ from R_{i-1}.
        $\triangleright$ Discharging token step.
    $i \leftarrow i + 1$;
end while
return S

the right endpoint is the smallest among all rays in $R \setminus S$ and delete it from R (when there are multiple such rays, choose one arbitrarily). We repeat the above three steps until V is empty. The above algorithm takes $O((|R|+|V|)\log(|R|+|V|))$ time (using segment trees [18]) and outputs a set S of rays such that all segments in V intersect at least one ray in S.

We describe the above algorithm formally in Algorithm 1. Below we introduce some notations used to describe the algorithm. We assign *token* $T_r = \{r\}$ for each $r \in R$ initially. For $i \geq 1$, let R_i, V_i, S_i be the set of rays, the set of segments and the solution constructed by this Algorithm 1, respectively at the *end* of i^{th} iteration. A ray $r \in R_i$ is *critical* if there is a segment $v \in V_i$ such that r is the only ray in R_i that intersects v. We describe a *discharging technique* below.

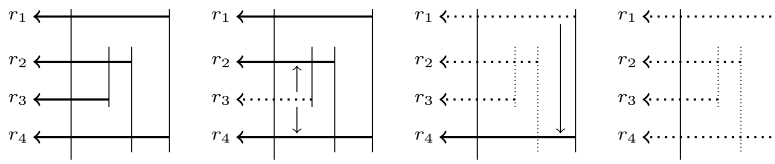


Figure 5.3.1: (a) An input SSR instance, (b) 1^{st} iteration, (c) 2^{nd} iteration and (d) 3^{rd} iteration of the MOD-SSR-Algorithm with (a) as input. A dotted ray (or segment) indicates that it is deleted.

Let D be a subset of R. A ray $r \in D$ lies *between* two rays $r', r'' \in D$ if the y-coordinate of r lies between those of r', r''. A ray $r \in D$ lies *just above* (resp. *just below*) a ray $r' \in D$ if y-coordinate of r is greater (resp. smaller) than that of r' and no other ray lies between r, r' in D. Two rays $r, r' \in D$ are *neighbours* of each other if r lies just above or below r'.

Discharging Method: Let $r \in R_{i-1} \setminus S_i$ be a ray whose x-coordinate of the right endpoint is the smallest. The phrase "r discharges the token to its neighbours" in the i^{th} iteration means the following operations in the given order.

(i) Let r' lie just above r and r'' lie just below r in $R_{i-1} \setminus S_i$. For all $x \in T_r$ (x and r not necessarily distinct) do the following. If there is a segment in V_i that intersects x, r' and r then assign $T_{r'} = T_{r'} \cup \{x\}$ and if there is a segment in V_i that intersects x, r'' and r then $T_{r''} = T_{r''} \cup \{x\}$.

(ii) Make $T_r = \emptyset$ after performing the above step.

For an illustration, consider the input instance shown in Figure 5.3.1(a). At the first iteration of Algorithm 1, r_3 passes the token to its neighbours (r_2, r_4) and gets deleted. After the 1^{st} iteration, notice that r_2 has become critical. So, at the begining of the 2^{nd} iteration Algorithm 1 put r_2 in the solution. Then all segment intersecting r_2 is deleted and r_2 itself is also deleted. Also in the second iteration r_1 passes the token to its neighbour (r_4) and gets deleted. Finally in the third iteration r_4 is put in the solution. We have the following observation.

Observation 5.3.1. *For some $v \in V_k$, $k \geq 1$, if some ray $r \in R_0$ intersects v, then either $r \in R_k$ or there exists some ray $r' \in R_k$ such that $r \in T_{r'}$.*

Proof. Assume $r \notin R_k$. Let $< r_1, r_2, \ldots, r_k >$ be a sorted order of the rays such that for $i < j$, r_i discharged the token to the neighbours before r_j. Due to step 5 of SSR-algorithm, $X =< r_1, r_2, \ldots, r_k >$ is an increasing sequence based on the x-coordinate of their right endpoint. Observe that, whenever a ray $r_i \in X$ discharged its token to its neighbours in the i^{th} iteration, all the vertical segments in V_i intersected by r_i also intersects one of the immediate neighbours of r_i. Again as $v \in V_k$, v is not intersected by critical ray within k iteration. Hence the result follows. □

Lemma 5.3.1. *For a ray r, there are at most two tokens containing r.*

Proof. If r never discharged its token to its neighbours, then the statement is true. Let r discharged the token to its neighbours at iteration i. Note that, r discharged tokens to at most two of its neighbours. Since r gets deleted after the discharging step, the rays whose token contain r become neighbours of each other.

Let j be the minimum integer with $i < j$ such that at the end of $(j-1)^{th}$ iteration, there is a ray $p \in R_{j-1}$ which is critical and $r \in T_p$. Note that iteration of SSR-Algorithm may stop before encountering such events. However, within iteration i to $j-1$, there may exist some rays which discharged their tokens containing r due to step 5 of SSR-Algorithm.

To prove the lemma, we use induction to show that there are at most two tokens containing r in any iteration from i upto $j-1$. Consider some k, $i < k < j$, such that x_1, $x_2 \in R_{k-1}$ be only two rays where $r \in T_{x_1}$

and $r \in T_{x_2}$. Notice that, x_1 and x_2 are neighbours of each other and without loss of generality assume x_1 lies just above x_2 in V_{k-1}. Assume x_1 discharged its token at k^{th} iteration. If there exists a neighbour of x_1 (say x_3) which is different from x_2, then due to the discharging step of k^{th} iteration, x_1 passes the token to its neighbours (i.e x_2 and x_3) and gets deleted from R_{k-1} to create R_k. If x_3 does not exist, then x_1 shall pass the token only to x_2. Therefore x_2 becomes the top-most ray among those rays in R_k which intersect some segment intersecting r.

Moreover, if x was the only ray in R_{k-1} such that $r \in T_x$, then x was the top-most (or bottom-most) ray among those rays in R_{k-1} which intersect some segment intersecting r. Therefore, at the end of k^{th} iteration there is exactly one ray $x' \in R_k$ such that $r \in T_{x'}$ and x' must be the top-most (resp. bottom-most) ray among those rays in R_k which intersect some segment intersecting r.

Hence we conclude that for each k with $i \leq k < j$, there is at most two rays $r', r'' \in R_k$ such that $r \in T_{r'} \cap T_{r''}$ and they are neighbours. If there is exactly one ray $r''' \in R_k$ such that $r \in T_{r'''}$ then r''' must be the top-most or bottom-most ray among those rays in R_k which intersect some segment intersecting r.

In iteration j, ray p is critical and $r \in T_p$ and p is put in the solution. If p is the only ray whose token contained r, only T_p will contain r after the termination of Algorithm 1. Let $r', p \in R_{j-1}$ be the rays whose token contained r. They must be neighbours. Without loss of generality assume that p lies just above r'. If both r', p are selected in S_j, then there is nothing to prove. Now consider the set A of segments in V_j that intersects r but not p. Note that, no ray above p intersects any segment in A. Hence r' becomes the only ray in next iterative step whose token contains r and r' turns to be bottom most ray among those rays in R_{j-1} which intersect some segment intersecting r. Now consider any iteration $k > j$. By similar arguments as above, there would be at most one ray in R_k that contains the token r. Hence the lemma follows. □

For a segment $v \in V$, let $N(v) \subseteq R$ be the set of rays that intersect v. Let $r \in S$ be a ray, i be the minimum integer such that $r \in S_i$. There must exist a segment $\nu_r \in V_{i-1}$ such that r is the only ray in R_{i-1} that intersects ν_r and all rays in $N(\nu_r) \setminus \{r\}$ must have passed the token to its neighbours. So, for each ray $r \in S$, there exists a segment ν_r such that for all $x \in N(\nu_r) \setminus \{r\}$ we have $T_x = \emptyset$. We call ν_r a *critical segment with respect to* r.

Observation 5.3.2. *For a ray $r \in S$ let ν_r be a critical segment with respect to r. Then $N(\nu_r) \subseteq T_r$.*

Proof. Consider any arbitrary but fixed deleted ray $y \in N(\nu_r) \setminus \{r\}$ which was deleted at some j^{th} iteration. By Observation 5.3.1, there exists a ray $y' \in R_j$ such that y' intersects v and $y \in T_{y'}$. Applying the above argument for all rays in $N(\nu_r) \setminus \{r\}$, we have the proof. □

Lemma 5.3.2. *If S is the set returned by the* SSR-*algorithm with rays R and segments V, then $|S| \leq 2|OPT|$, where OPT is an optimum solution of $\mathcal{SSR}(R, V)$.*

Proof. Let R be the set of rays and V be the set of segments with $|R| = n, |V| = m$. Consider the ILP formulation Q of $\mathcal{SSR}(R, V)$. For each ray $r \in R$, let $x_r \in \{0, 1\}$ denote the variable corresponding to r. Objective is to minimize $\sum\limits_{r \in R} x_r$ with constraints $\sum\limits_{r \in N(v)} x_r \geq 1$ for all $v \in V$. Let the corresponding relaxed LP formulation be Q_l.

Let $\mathbf{Q}_l = \{x_r\}_{r \in R}$ be an optimal solution of Q_l. Consider SSR-algorithm. Here, define $y_r = 1$ if $r \in S$, $y_r = 0$ if $r \notin S$ and $\mathbf{Q}' = \{y_r\}_{r \in R}$, obtained by the algorithm. This is a feasible solution of Q as SSR-algorithm terminates only when no segments are left in V_i. Now we fix any arbitrary $r \in S$ and ν_r be a critical segment with respect to r. Then due to Observation 5.3.2, we know that for all $z \in N(\nu_r) \setminus \{r\}$ we have $T_z = \emptyset$ and $N(\nu_r) \subseteq T_r$. Since $N(\nu_r) \subseteq T_r$ by Observation 5.3.2, we have for the constraint corresponding to ν_r in Q_l,

$$\sum_{z \in N(\nu_r)} y_z = 1 \leq \sum_{z \in N(\nu_r)} x_z \leq \sum_{z \in T_r} x_z$$

Therefore, from above argument and from Lemma 5.3.1 we conclude that

$$|S| = \sum_{r \in S} y_r = \sum_{r \in S} \sum_{z \in N(\nu_r)} y_z \leq \sum_{r \in S} \sum_{z \in T_r} x_z \leq 2 \sum_{z \in R} x_z \leq 2|OPT|.$$

Hence we have the proof. □

The proofs of Lemma 5.1.1 and Theorem 5.1.1 follows directly from the proof of Lemma 5.3.2.

5.4 Integrality gap of the SRS problem

In this section we shall prove Lemma 5.1.2 by showing that the upper bound of the integrality gap of the *stabbing rays with segments* (SRS) problem is at most two. Recall in the SRS problem, the inputs are a set of disjoint leftward-directed horizontal rays and a set of disjoint vertical segments. The objective is to select a minimum number of vertical segments that intersect all leftward-directed horizontal rays.

2-approximation algorithm for the SRS problem: With each segment $v \in V$, we associate a token T_v which is a subset of V. Initialise $T_v = \emptyset$ for each $v \in V$. Let r_i be the ray whose right-endpoint, (x_i, y_i), has the smallest x-coordinate. We assume without loss of generality that x- and y-coordinates of the endpoints of the rays are all distinct. Assuming that there is a feasible solution to the SRS instance, there must exist a segment of V that intersects r_i. Let $N(r_i) \subseteq V$ be the set of segments that intersect r_i. Let v_{top} (resp. v_{bot}) be a segment in $N(r_i)$ whose top endpoint is top-most (resp., bottom endpoint is bottom-most); it may be that $v_{top} = v_{bot}$. We add both v_{top} and v_{bot} to our heuristic solution set S. Also we set $T_{v_{top}} = T_{v_{bot}} = N(r_i)$. We remove from R all of the rays that intersect v_{top} or v_{bot}, delete all segments in $N(r_i)$ and then repeat the above steps untill $R = \emptyset$. Observe that for each ray r, there is a segment $v \in S$ that intersects r. Also observe that for each segment $v \in V$, there are at most two tokens such that both of them contains v. Observe that, the running time of the above algorithm is $O(n \log n)$ where $n = |V|$.

Lemma 5.4.1. *Let Q be the ILP of an SRS instance with a set of rays R and set of segments V as input and Q_l be the corresponding relaxed LP. Then $OPT(Q) \leq 2 \cdot OPT(Q_l)$.*

Proof. Let $\mathbf{X} = \{x_v\}_{v \in V}$ be an optimal solution of Q_l where x_v denotes the value of the variable in Q_l corresponding to $v \in V$. Let S be the solution returned by the above algorithm with R, V as input. Now define for each $v \in V$, $y_v = 1$ if $v \in S$, $y_v = 0$ if $v \notin S$ and let $\mathbf{Y} = \{y_v\}_{v \in V}$. Observe that $\mathbf{Y}$ is a feasible solution of Q. For each $z \in S$, there is a ray r_i such that $T_z = N(r_i)$. Therefore, $y_z = 1 \leq \sum_{v \in N(r_i)} x_v = \sum_{v \in T_z} x_v$

As a segment v is contained in at most two tokens, using the above inequality we have

$$|S| = \sum_{v \in S} y_v \leq \sum_{v \in S} \sum_{v' \in T_v} x_{v'} \leq 2 \sum_{v' \in V} x_{v'} = 2 \cdot OPT(Q_l)$$

Hence the result follows. □

To complete the proof of Lemma 5.1.2 observe that the approximation algorithm stated above returns a feasible solution D for the ILP formulation $\mathcal{C}$ of an *SRS* instance in $O(n \log n)$ time such that $|D| \leq 2 \cdot OPT(\mathcal{C}_l)$.

5.5 Integrality gap of the LVSC problem

In this section, we shall prove Lemma 5.1.3 by showing that the upper bound of the integrality gap of the LVSC problem is at most 8.

Let l be the straight line that intersects all horizontal segment in H. Notice that if l is a horizontal line then any vertical line segment intersects at most one horizontal line segment in H. This is because horizontal lines in H are disjoint. But, in this case, there is nothing to prove.

Therefore, without loss of generality, we assume that l passes through the origin. at an angle in $[\frac{\pi}{2}, \pi)$. For a vertical segment $v \in V$, let $N(v)$ denote the set of horizontal segments intersecting v, $A(v)$ be the set of horizontal segments that intersect v *above* l and $B(v) = N(v) \setminus A(v)$. Observe that for a vertical segment v and a horizontal segment $h \in B(v)$, h intersects v *on* or *below* l.

Based on these consider the following ILP formulation, Q, of the $\mathcal{LVSC}(V, H)$ instance. For each horizontal segment $h \in H$ let $x_h \in \{0, 1\}$ denote the variable corresponding to h. Objective is to minimize $\sum\limits_{h \in H} x_h$ with constraints

$$\sum_{h \in A(v)} x_h + \sum_{h \in B(v)} x_h \geq 1, \ \forall v \in V$$

Let Q_l be the relaxed LP formulation of Q and $\mathbf{Q}_l = \{x_h \colon h \in H\}$ be an optimal solution of Q_l. Since Q_l consists of n variables where $n = |H|$, solving Q_l takes $O(n^5)$ time [147]. Now we define the following sets.

$$V_1 = \left\{ v \in V \colon \sum_{h \in A(v)} x_h \geq \frac{1}{2} \right\}, V_2 = \left\{ v \in V \colon \sum_{h \in B(v)} x_h \geq \frac{1}{2} \right\}$$

$$H_1 = \bigcup_{v \in V_1} A(v), H_2 = \bigcup_{v \in V_2} B(v)$$

Based on these, we consider two integer programs Q' and Q''.

minimize $\sum_{h \in H_1} x'_h$ subject to $\sum_{h \in A(v)} x'_h \geq 1, \forall v \in V_1$ $x'_h \in \{0,1\}, \quad h \in H_1$ Q'	minimize $\sum_{h \in H_2} x''_h$ subject to $\sum_{h \in B(v)} x''_h \geq 1, \forall v \in V_2$ $x''_h \in \{0,1\}, \quad h \in H_2$ Q''

Let Q'_l and Q''_l be the relaxed LP formulation of Q' and Q'' respectively. Clearly, the solutions of Q' and Q'' gives a feasible solution for Q. Hence $OPT(Q) \leq OPT(Q') + OPT(Q'')$. For each $x_h \in \mathbf{Q}_l$, define $y_h = \min\{1, 2x_h\}$ and define $\mathbf{Y}_l = \{y_h\}_{x_h \in \mathbf{Q}_l}$. Notice that $\mathbf{Y}_l$ gives a feasible solution to Q'_l and Q''_l. Therefore, $OPT(Q'_l) + OPT(Q''_l) \leq 2 \cdot OPT(Q_l)$. We have the following claim.

Claim. $OPT(Q') \leq 2 \cdot OPT(Q'_l)$ *and* $OPT(Q'') \leq 2 \cdot OPT(Q''_l)$.

To prove the first part, note that for each segment $v \in V_1$, $A(v)$ is non-empty and for each $h \in A(v)$, h intersects v above the line l (the straight line which intersects all segments in H). Since all segments in H_1 intersect the straight line l we can consider the horizontal segments in H_1 as leftward-directed rays and all vertical segments in V_1 lie above l. Hence, solving Q' is equivalent to solving an ILP formulation, say $\mathcal{E}$, of the problem of finding a minimum cardinality subset of leftward-directed rays in H_1 that intersects all vertical segments in the set V_1. Hence solving $\mathcal{E}$ is equivalent to solving an SSR instance with H_1 and V_1 as input. By Lemma 5.1.1, we have that

$$OPT(Q') = OPT(\mathcal{E}) \leq 2 \cdot OPT(\mathcal{E}_l) \leq 2 \cdot OPT(Q'_l)$$

where $\mathcal{E}_l$ is the relaxed LP formulation of $\mathcal{E}$. Hence we have proof of the first part. For the second part, using similar arguments as above, we can show that solving Q'' is equivalent to solving an SSR instance and therefore by Lemma 5.1.1, we have that $OPT(Q'') \leq 2 \cdot OPT(Q''_l)$. Hence the proof of the claim follows.

By Lemma 5.1.1, we can solve both Q' and Q'' in polynomial time. Let D' and D'' be solutions of Q' and Q'', respectively. Clearly, $D' \cup D''$ is a feasible solution to the $\mathcal{LVSC}(V, H)$ instance. Hence,

$$|D' \cup D''| \leq 4(OPT(Q_l') + OPT(Q_l'') \leq 8 \cdot OPT(Q_l)$$

This completes the proof.

5.6 Integrality gap of the LHSC problem

In this section, we shall prove Lemma 5.1.4 by showing that the upper bound of the integrality gap of the LHSC problem is at most 8. The proof is similar to that of Lemma 5.1.3. For sake of completeness, we present the detailed proof below.

Let l be the straight line that intersects all horizontal segment in H. Without loss of generality, we assume that l passes through the origin at an angle in $[\frac{\pi}{2}, \pi)$. For a horizontal segment $h \in H$, let $N(h)$ denote the set of vertical segments intersecting h, $A(h)$ be the set of vertical segments that intersect h *above* l and $B(h) = N(h) \setminus A(h)$. Observe that for a horizontal segment h and a vertical segment $v \in B(h)$, v intersects h *on* or *below* l.

Based on these we have the following ILP formulation of the $\mathcal{LHSC}(V, H)$ instance.

$$\begin{array}{lll} \text{minimize} & \sum_{v \in V} x_v & \\ \text{subject to} & \sum_{v \in A(h)} x_v + \sum_{v \in B(h)} x_v \geq 1, & \forall h \in H \\ & x_v \in \{0,1\}, & \forall v \in V \end{array}$$

Q

Let Q_l be the the relaxed LP formulation of Q and $\mathbf{Q}_l = \{x_v \colon v \in V\}$

be an optimal solution of Q_l. Now we define the following sets.

$$H_1 = \left\{ h \in H\colon \sum_{v \in A(h)} x_v \geq \frac{1}{2} \right\}, H_2 = \left\{ h \in H\colon \sum_{v \in B(h)} x_v \geq \frac{1}{2} \right\}$$

$$V_1 = \bigcup_{h \in H_1} A(h), V_2 = \bigcup_{h \in H_2} B(h)$$

Based on these, we consider the following two integer programs Q' and Q''.

Q'	Q''
minimize $\sum_{v \in V_1} x'_v$ subject to $\sum_{v \in A(h)} x'_v \geq 1, \forall h \in H_1$ $x'_v \in \{0,1\},\ \ v \in V_1$	minimize $\sum_{v \in V_2} x''_v$ subject to $\sum_{v \in B(h)} x''_v \geq 1, \forall h \in H_2$ $x''_v \in \{0,1\},\ \ v \in V_2$

Let Q'_l and Q''_l be the relaxed LP formulation of Q' and Q'' respectively. Clearly, the solutions of Q' and Q'' gives a feasible solution for Q. Hence $OPT(Q) \leq OPT(Q') + OPT(Q'')$. For each $x_v \in \mathbf{Q}_l$, define $y_v = \min\{1, 2x_v\}$ and define $\mathbf{Y}_l = \{y_v\}_{x_v \in \mathbf{Q}_l}$. Notice that $\mathbf{Y}_l$ gives a feasible solution to Q'_l and Q''_l. Therefore, $OPT(Q'_l) + OPT(Q''_l) \leq 4 \cdot OPT(Q_l)$. We have the following claim.

Claim. $OPT(Q') \leq 2 \cdot OPT(Q'_l)$ *and* $OPT(Q'') \leq 2 \cdot OPT(Q''_l)$.

To prove the first part, note that for each vertex $h \in H_1$, $A(h)$ is non-empty and for each $v \in A(h)$, v intersects h above the line l (the straight line which intersects all segments in H). Since all segments in H_1 intersect the straight line l we can consider the horizontal segments in H_1 as leftward-directed rays and all vertical segments in V_1 lie above l. Hence, solving Q' is equivalent to solving an ILP formulation, say $\mathcal{E}$, of the problem of finding a minimum cardinality subset of vertical segments in V_1 that intersects all leftward-directed rays in the set H_1. Hence solving $\mathcal{E}$

is equivalent to solving an SRS instance with V_1 and H_1 as input. By Lemma 5.1.2, we have that

$$OPT(Q') = OPT(\mathcal{E}) \leq 2 \cdot OPT(\mathcal{E}_l) \leq 2 \cdot OPT(Q'_l)$$

where $\mathcal{E}_l$ is the relaxed LP formulation of $\mathcal{E}$. Hence we have proof of the first part. For the second part, using similar arguments as above, we can show that solving Q'' is equivalent to solving an SRS instance and therefore by Lemma 5.1.2, we have that $OPT(Q'') \leq 2 \cdot OPT(Q''_l)$. Hence the proof of the claim follows.

By Lemma 5.1.2, we can solve both Q' and Q'' in polynomial time. Let D' and D'' be solutions of Q' and Q'', respectively. Clearly, $D' \cup D''$ is a feasible solution to the $\mathcal{LHSC}(V, H)$ instance. Hence,

$$|D' \cup D''| \leq 2(OPT(Q'_l) + OPT(Q''_l) \leq 8 \cdot OPT(Q_l)$$

Hence we have the proof of Lemma 5.1.4.

5.7 Algorithm for stabbed rectangle overlap graphs

Given a stabbed rectangle overlap representation of a graph G with n vertices we shall give a 768-approximation algorithm for the MDS problem on G. Specifically, we shall prove the following theorem.

Theorem 5.7.1. *Given a stabbed rectangle overlap representation of a graph G with n vertices, there is an $O(n^5)$-time 768-approximation algorithm for the* MDS *problem on G.*

We shall use Lemma 5.1.3 and Lemma 5.1.4 to prove the above theorem.

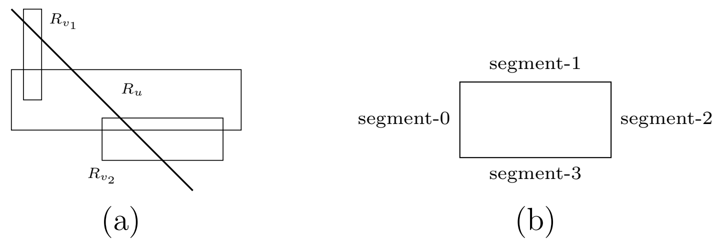


Figure 5.7.1: (a) In this example $R_{v_1} \in N'(u)$ and $R_{v_2} \in N''(u)$. (b) Nomenclature for the four boundadry segments of a rectangle.

Let $\mathcal{R}$ be a stabbed rectangle overlap representation of a graph G and l be the line that intersects all rectangles in $\mathcal{R}$. We shall also refer to l as the *cutting line*.

For a vertex $u \in V(G)$, let R_u denote the rectangle corresponding to u in $\mathcal{R}$. Without loss of generality, we assume that the coordinates of all corner points of all the rectangles in $\mathcal{R}$ are distinct and that the cutting line passes through the origin at an angle in $[\frac{\pi}{2}, \pi)$ with the positive x-axis.

Each rectangle R_u consists of four *boundary segments* i.e. *left segment*, *top segment*, *right segment* and *bottom segment*. Without loss of generality, we assume that the cutting line intersects eactly two boundary segments of each rectangle in $\mathcal{R}$. For a vertex $u \in V$, let $N(u)$ denote the set of rectangles that overlaps with R_u in $\mathcal{R}$. Let $N'(u)$ be the set of rectangles having a boundary segment that intersects both the cutting line and some boundary segment of R_u that does not intersect the cutting line. See Figure 5.7.1(a) for an example. Now define $N''(u) = N(u) \setminus N'(u)$. We have the following observation.

Observation 5.7.1. *For a rectangle $R_u \in \mathcal{R}$ and a rectangle $X \in N''(u)$, there is a boundary segment of R_u that intersects the cutting line and some boundary segment of X.*

Proof. Suppose X has a boundary segment that intersects the cutting line and some boundary segment s of R_u. In this case, s must also intersect the cutting line and we are done. Otherwise, observe that X contains two boundary segments, say s_1 and s_2, such that none of s_1, s_2 intersects the cutting line and R_u intersects both s_1 and s_2. If s_1 and s_2 belong to opposite sides of the cutting line, then both s_1 and s_2 are horizontal or both of them are vertical. In either case, R_u must have a boundary segment t that intersect both s_1, s_2 and the cutting line. Consider the case when both s_1 and s_2 lie below the cutting line. Then there exists $w \in \{s_1, s_2\}$ which is a vertical segment and $z \in \{s_1, s_2\} \setminus \{w\}$ which is a horizontal segment. Hence, R_u must have a horizontal boundary segment w' that intersects w and a vertical boundary segment z' that intersects z. If neither w' nor z' intersects the cutting line, then observe that the top-right corner of R_u must lie below the cutting line, implying that R_u does not intersect the cutting line. This is a contradiction. Similarly, the case when both s_1, s_2 lie above the cutting line also leads to a contradiction. □

We shall denote the left segment of a rectangle $R_u \in \mathcal{R}$ also as the *segment*-0 of R_u. Similarly *segment*-1, *segment*-2 and *segment*-3 of R_u shall refer to the top segment, the right segment and the bottom segment of R_u, respectively. See Figure 5.7.1(b) for an illustration. Let $\mathcal{S} = \{(0,1),(0,3),(1,0),(1,2),(2,1),(2,3),(3,0),(3,2)\}$. Since no two horizontal segments or two vertical segments intersect, we have the following observation.

Observation 5.7.2. *If two rectangles $R_u, R_v \in \mathcal{R}$ overlap there must be a pair $(i,j) \in \mathcal{S}$ such that segment-i of R_u intersects segment-j of R_v.*

Based on the above observation, we partition the sets $N'(u)$ and $N''(u)$ in the following way. For each rectangle $R_u \in \mathcal{R}$ and $(i,j) \in \mathcal{S}$, a rectangle $R_v \in N'(u)$ belongs to the set $X'_u(i,j)$ if and only if (i,j) is the smallest pair in the lexicographic order such that *(a)* segment-i of R_u intersects the segment-j of R_v and *(b)* segment-j of R_v intersects the cutting line.

Similarly, for each rectangle $R_u \in \mathcal{R}$ and $(i,j) \in \mathcal{S}$, a rectangle $R_v \in N''(u)$ belongs to the set $X''_u(i,j)$ if and only if (i,j) is the smallest pair in the lexicographic order such that *(a)* segment-i of R_u intersects the segment-j of R_v and *(b)* segment-i of R_u intersects the cutting line. The next observation follows from the above definitions.

Observation 5.7.3. *For each $R_u \in \mathcal{R}$, $\{X'_u(i,j)\}_{(i,j)\in\mathcal{S}}$ is a partition of $N'(u)$ and $\{X''_u(i,j)\}_{(i,j)\in\mathcal{S}}$ is a partition of $N''(u)$.*

For each $R_u \in \mathcal{R}$, define the sets $\mathcal{S}'_u = \{(i,j) \in \mathcal{S} \colon X'_u(i,j) \neq \emptyset\}$ and $\mathcal{S}''_u(i,j) = \{(i,j) \in \mathcal{S} \colon X''_u(i,j) \neq \emptyset\}$. Recall that according to our assumption, each rectangle intersect the cutting line exactly two times. Since the boundary segment of a retangle intersect exactly two boundary segments of another rectangle, we have the following observation.

Observation 5.7.4. *For each $R_u \in \mathcal{R}$, $|\mathcal{S}'_u| \leq 4$ and $|\mathcal{S}''_u| \leq 4$.*

Proof. Observe that if there is a rectangle $R \in X'_u(i,j)$ for some $(i,j) \in \mathcal{S}'_u$ then R intersects a boundary segment of R_u that does not intersect the cutting line. There are exactly two boundary segments, say segment-i and segment-j, of R_u that do not intersect the cutting line. Hence $\mathcal{S}'_u$ is a subset of $\{(i,i-1),(i,i+1),(j,j-1),(j,j+1)\}$ where all addition operations are modulo 4. Therefore $|\mathcal{S}'_u| \leq 4$. To prove the second part of the observation, first we use Observation 5.7.2 to infer that if a rectangle $R \in X''_u(i,j)$ for some $(i,j) \in \mathcal{S}''_u$ then R intersects a boundary segment of R_u that intersects the cutting line. Now using similar arguments as above we have that $|\mathcal{S}''_u| \leq 4$. $\square$

Let Q denote the following ILP formulation of the MDS problem on G and Q_l be the corresponding relaxed LP formulation.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{R_v \in \mathcal{R}} x_v \\
\text{subject to} \quad & \sum_{(i,j)\in\mathcal{S}'_u} \sum_{R_v \in X'_u(i,j)} x_v + \sum_{(i,j)\in\mathcal{S}''_u} \sum_{R_v \in X''_u(i,j)} x_v \geq 1, \ \forall R_u \in \mathcal{R} \\
& x_v \in \{0,1\}, \qquad \forall R_v \in \mathcal{R}
\end{aligned}
$$

Q

Let $\mathbf{Q}_l = \{x_v \colon R_v \in \mathcal{R}\}$ be an optimal solution of Q_l. By Observation 5.7.4, for each rectangle $R_u \in \mathcal{R}$, we have $|\mathcal{S}'_u| + |\mathcal{S}''_u| \le 8$. Hence, there is a pair $(i,j) \in \mathcal{S}'_u \cup \mathcal{S}''_u$ such that either $\sum_{R_v \in X'_u(i,j)} x_v \ge \frac{1}{8}$ or $\sum_{R_v \in X''_u(i,j)} x_v \ge \frac{1}{8}$. For each pair $(i,j) \in \mathcal{S}$, define

$$A'(i,j) = \left\{ R_u \in \mathcal{R} \colon (i,j) \in \mathcal{S}'_u, \sum_{R_v \in X'_u(i,j)} x_v \ge \frac{1}{8} \right\}$$

$$B'(i,j) = \bigcup_{R_u \in A'(i,j)} X'_u(i,j)$$

$$A''(i,j) = \left\{ R_u \in \mathcal{R} \colon (i,j) \in \mathcal{S}''_u, \sum_{R_v \in X''_u(i,j)} x_v \ge \frac{1}{8} \right\}$$

$$B''(i,j) = \bigcup_{R_u \in A''(i,j)} X''_u(i,j)$$

Based on these we have the following two ILP formulations for each pair $(i,j) \in \mathcal{S}$.

$Q'(i,j)$	$Q''(i,j)$
minimize $\sum_{R_v \in B'(i,j)} x'_v$	minimize $\sum_{R_v \in B''(i,j)} x''_v$
subject to $\sum_{R_v \in X'_u(i,j)} x'_v \ge 1, \forall R_u \in A'(i,j)$	subject to $\sum_{R_v \in X''_u(i,j)} x''_v \ge 1, \forall R_u \in A''(i,j)$
$x'_v \in \{0,1\}, \quad R_v \in B'(i,j)$	$x''_v \in \{0,1\}, \quad R_v \in B''(i,j)$

For each pair $(i,j) \in \mathcal{S}$, let $Q'_l(i,j)$ and $Q''_l(i,j)$ be the relaxed LP formulation of $Q'(i,j)$ and $Q''(i,j)$, respectively. Observe that

$$OPT(Q) \le \sum_{(i,j) \in \mathcal{S}} \left(OPT(Q'(i,j)) + OPT(Q''(i,j)) \right)$$

For each $x_v \in \mathbf{Q}_l$, define $y_v = \min\{1, 8x_v\}$ and $\mathbf{Y}_l = \{y_v\}_{x_v \in \mathbf{Q}_l}$. Due to Observation 5.7.3 and 5.7.4, $\mathbf{Y}_l$ gives a feasible solution to $Q'_l(i,j)$

and $Q''_l(i,j)$ for all $(i,j) \in \mathcal{S}$. Therefore, $OPT(Q'_l(i,j)) \le 8 \cdot OPT(Q_l)$ and $OPT(Q''_l(i,j)) \le 8 \cdot OPT(Q_l)$ for all $(i,j) \in \mathcal{S}$. Now we have the following lemma.

Lemma 5.7.1. *For each $(i,j) \in \mathcal{S}$ there is a set $D'(i,j) \subseteq B'(i,j)$ such that $D'(i,j)$ gives a feasible solution of $Q'(i,j)$ and $|D'(i,j)| \le 8 \cdot OPT(Q'_l(i,j))$.*

Proof. For any $(i,j) \in \mathcal{S}$, solving $Q'(i,j)$ is equivalent to finding a minimum cardinality subset D of $B'(i,j)$ such that each rectangle $R_u \in A'(i,j)$ overlaps a rectangle in $D \cap X'_u(i,j)$. Notice that, for each $R_u \in A'(i,j)$ the set $X'_u(i,j)$ is non-empty. Moreover for each $R_v \in X'_u(i,j)$, the segment-j of R_v intersects the cutting line and segment-i of R_u. Let $S = \{\text{segment-}i \text{ of } R_u \colon R_u \in A'(i,j)\}$, $T = \{\text{segment-}j \text{ of } R_v \colon R_v \in B'(i,j)\}$.

Solving $Q'(i,j)$ is equivalent to the problem finding a minimum cardinality subset D of T such that every segment in S intersect at least one segment in D. Moreover, every segment in T intersects the cutting line. Without loss of generality we can assume that S consists of vertical segments. Therefore T consists of horizontal segments all intersecting the cutting line. Hence solving $Q'(i,j)$ is equivalent to solving the $\mathcal{LVSC}(S,T)$ instance. Hence by Lemma 5.1.3, we have a feasible solution (say $D'(i,j)$) for $Q'(i,j)$ such that $|D'(i,j)| \le 8 \cdot OPT(Q'_l(i,j))$. □

Using similar arguments as in the proof of Lemma 5.7.1, we can prove that solving $Q''(i,j)$ is equivalent to solving an instance of the LHSC problem. Then using Lemma 5.1.4 we can prove the following lemma.

Lemma 5.7.2. *For each $(i,j) \in \mathcal{S}$ there is a set $D''(i,j) \subseteq B''(i,j)$ such that $D''(i,j)$ gives a feasible solution of $Q''(i,j)$ and $|D''(i,j)| \le 8 \cdot OPT(Q''_l(i,j))$.*

For each $R_v \in \mathcal{R}$, let $\mathcal{T}_v = \{(i,j) \in \mathcal{S} \colon R_v \in B'(i,j) \text{ or } R_v \in B''(i,j)\}$. The following observation follows from the definitions of $B'(i,j)$ and

$B''(i,j)$.

Observation 5.7.5. *For each $R_v \in \mathcal{R}$, we have that $|\mathcal{T}_v| \leq 12$.*

For each pair $(i,j) \in \mathcal{S}$, due to Lemma 5.7.1 and Lemma 5.7.2, we have a feasible solution $D'(i,j)$ of $Q'(i,j)$ and a feasible solution $D''(i,j)$ such that $|D'(i,j)| \leq 8 \cdot OPT(Q'_l(i,j))$ and $|D''(i,j)| \leq 8 \cdot OPT(Q''_l(i,j))$. Let D be the union of $D'(i,j)$'s and $D''(i,j)$ for all $(i,j) \in \mathcal{S}$. Using Observation 5.7.5 we have

$$\begin{aligned}
|D| &= \sum_{(i,j)\in\mathcal{S}} (|D'(i,j)| + |D''(i,j)|) \\
&\leq 8 \cdot \sum_{(i,j)\in\mathcal{S}} (OPT(Q'_l(i,j)) + OPT(Q''_l(i,j))) \\
&\leq 768 \cdot OPT(Q_l) \leq 768 \cdot OPT(Q)
\end{aligned}$$

This completes the proof of Theorem 5.7.1.

5.8 Concluding remarks and open problems

In this chapter, we introduce the class of stabbed rectangle overlap graphs and study the MDS problem on stabbed rectangle overlap graphs. We gave an 768-approximation algorithm for the MDS problem on stabbed rectangle overlap graphs. As a corollary to Theorem 5.7.1, we have the following.

Corollary 7. *Let $\mathcal{R}$ be a stabbed rectangle intersection representation of a graph G such that no two rectangles in $\mathcal{R}$ contain each other. There is an $O(|V(G)|^5)$-time 768-approximation algorithm for the* MDS *problem on G.*

We also proved that if the *Unique Games Conjecture* is true then it is not possible to have a polynomial-time $(2-\epsilon)$-approximation algorithm for the MDS problem on rectangle overlap graphs. However, our construction does not work for stabbed rectangle intersection graphs. This leads to the following question(s).

Question 5.8.1. *Is there a c-approximation algorithm for the* MDS *problem on stabbed rectangle overlap graphs with* $c < 768$*?*

Question 5.8.2. *Is there a constant factor approximation algorithm for the* MDS *problem on stabbed rectangle intersection graphs?*

Question 5.8.3. *Is there a constant factor approximation algorithm for the* MDS *problem on rectangle overlap graphs?*

To prove the approximation ratio of our algorithms, we studied the *SSR* problem and the *SRS* and proved that their integrality gaps are at most two. Improvements on the upper bounds of the integrality gaps of the SSR problem and the SRS problem will immediately imply better approximation ratios for several optimisation problems including a few studied by Bandyapadhyay and Meharbi [14]. Therefore the following question might be interesting.

Question 5.8.4. *What is the integrality gap of the* SSR *and* SRS *problems?*

6

Dominating set of vertically-stabbed L-graphs and unit B_k-VPG graphs

Contents

In this chapter, we present approximation algorithms for the Minimum Dominating Set (MDS) problem on *vertically-stabbed* L*-graphs* and *unit* B_k*-VPG graphs*. Recall that an L-path is a simple rectilinear path having the shape 'L'. A set of L-paths is *vertically-stabbed* if all L-paths in the set intersect a common vertical line. A graph G is a *vertically-stabbed* L*-graph* if G is an intersection graph of a set $\mathcal{R}$ of vertically-stabbed L-paths. Here $\mathcal{R}$ is a *vertically-stabbed* L*-representation* of G. For $k \geq 0$, A graph G is a *unit* B_k*-VPG* graph if G is an intersection graph of a set $\mathcal{R}$ of simple rectilinear curves on the plane such that each curve in the set has at most k bends and each segment of each of the curves have the same length as the other. Here $\mathcal{R}$ is a *unit* B_k*-VPG representation* of G.

6.1 Chapter overview

In Section 6.2, we shall prove that it is NP-Hard to solve the MDS problem on unit B_k-VPG graphs with $k \geq 0$ (Theorem 6.2.1).

In Section 6.3, we shall apply Lemma 5.1.1 and Lemma 5.1.2 an propose an 8-approximation algorithm for the MDS problem on vertically-stabbed L-graphs. In Section 6.4, we present an 18-approximation algorithm for the MDS problem on unit B_0-VPG graphs. In Section 6.5, we present an $O(k^4)$-approximation algorithm for the MDS problem on unit B_k-VPG graphs, for each $k \geq 1$. Finally, we draw conclusions in Section 6.6.

6.2 Hardness result

In this section, we prove the following theorem.

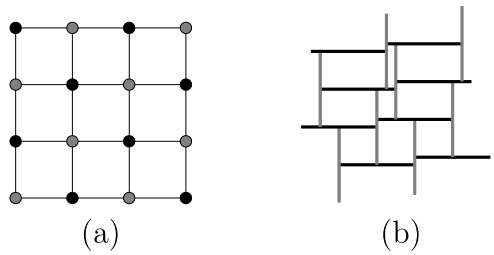


Figure 6.2.1: (a) A $(4,4)$-grid. In this case, X consists of the gray vertices and Y consists of black vertices. (b) A unit B_0-VPG representation of (a).

Theorem 6.2.1. *It is NP-Hard to solve the* MDS *problem on unit B_k-VPG graphs with $k \geq 0$.*

First we prove the NP-hardness for the MDS problem on unit B_0-VPG graphs. The (h,w)-*grid* is the undirected graph G with vertex set $\{(x,y)\colon x,y \in \mathbb{Z}, 1 \leq x \leq h, 1 \leq y \leq w\}$ and edge set $\{(u,v)(x,y)\colon |u-x|+|v-y|=1\}$. A graph G is a *grid* graph if G is an induced subgraph of (h,w)-grid for some positive integers h,w. We shall reduce the NP-complete MDS problem on grid graphs [56] to the MDS problem on unit B_0-VPG graphs.

We shall show that any grid-graph is a unit-B_0-VPG graph and thus prove Theorem 6.2.1. Observe that, it is sufficient to show that for any positive even integer n the (n,n)-grid has a unit B_0-VPG representation. Let n be a fixed positive even integer and H be a (n,n)-grid. Let $X = \{(i,j) \in V(H)\colon i,j \text{ have same parity}\}$ and $Y = V(H) \setminus X$. See Figure 6.2.1(a) for an example. We have the following observation.

Observation 6.2.1. *For any edge $e \in E'$, one of the endpoints of e belongs to X and the other endpoint belongs to Y.*

We shall show that H has a unit B_0-VPG representation $\mathcal{R}$ where the vertical segments represent the pairs in X and the horizontal segments represent the pairs in Y. We describe how to get such a unit B_0-VPG representation of H below.

Let $\epsilon = \frac{1}{n^2}$. For each $(i,j) \in Y$, we define two real values $x_{i,j}$ and $y_{i,j}$ as follows.

$$x_{i,j} = \begin{cases} \lceil \frac{j}{2} \rceil & \text{when } i = 1 \\ \lceil \frac{j}{2} - \epsilon \rceil & \text{when } i = 2 \\ x_{i-1,j+1} + \frac{x_{i-2,j} - x_{i-1,j+1}}{2} & \text{when } i \geq 3, i \equiv 0 \mod 2 \\ x_{i-1,j-1} + \frac{x_{i-2,j} - x_{i-1,j-1}}{2} & \text{when } i \geq 3, i \equiv 1 \mod 2 \end{cases}$$

$$y_{i,j} = \frac{i}{2} + \left\lceil \frac{j}{2} \right\rceil \epsilon$$

Notice that for $i \geq 3$, if $(i,j) \in Y$, then $(i-2,j) \in Y$. Moreover, if i is even then $(i-1,j+1) \in Y$ and if i is odd then $(i-1,j-1) \in Y$. Therefore, the values $x_{i,j}$ for all $(i,j) \in Y$ are well-defined. We have the following observation.

Observation 6.2.2. *Let for some pair (i,j) we have $\{(i,j-1),(i,j+1),(i+1,j),(i-1,j)\} \subseteq Y$. Then*

(i) $x_{i,j-1} + 1 = x_{i,j+1}$ *and* $y_{i,j-1} = y_{i,j+1} - \epsilon$*;*

(ii) $x_{i+1,j} < x_{i,j+1} < (x_{i+1,j}) + 1$ *and* $x_{i-1,j} < x_{i,j+1} < x_{i-1,j} + 1$*;*

(iii) when $i \equiv 1 \mod 2$*,* $y_{i-1,j} = y_{i,j+1} - 0.5$ *and* $y_{i+1,j} = y_{i,j+1} + 0.5$*; and*

(iv) when $i \equiv 0 \mod 2$*,* $y_{i-1,j} = y_{i,j-1} - 0.5$ *and* $y_{i+1,j} = y_{i,j-1} + 0.5$

Now for each $(i,j) \in Y$, we define a horizontal line segment $s_{i,j}$ as follows.

$$s_{i,j} = [x_{i,j}, x_{i,j} + 1] \times [y_{i,j}, y_{i,j}]$$

Let $S = \{s_{i,j}\}_{(i,j)\in Y}$. Observe that no two segment in S intersect each other and length of every segment in S is one. Now for each $(i,j) \in X$, we define the real values $x'_{i,j}$ and $y'_{i,j}$ as follows.

$$x'_{i,j} = \begin{cases} x_{i,j+1} & \text{when } i \equiv 1 \mod 2 \\ x_{i,j-1} + 1 & \text{when } i \equiv 0 \mod 2 \end{cases}$$
$$y'_{i,j} = \begin{cases} y_{i,j+1} - 0.5 & \text{when } i \equiv 1 \mod 2 \\ y_{i,j-1} - 0.5 & \text{when } i \equiv 0 \mod 2 \end{cases}$$

Notice that, for each $(i,j) \in X$ if i is odd then $(i,j+1) \in Y$ and if i is even then $(i,j-1) \in Y$. Therefore, the values $x'_{i,j}$ are well defined. Now for each $(i,j) \in X$, we define a vertical segment $t_{i,j}$ as follows.

$$t_{i,j} = [x'_{i,j}, x'_{i,j}] \times [y'_{i,j}, y'_{i,j} + 1]$$

Let $T = \{t_{i,j}\}_{(i,j)\in X}$. Observe that no two segment in T intersect each other and length of every segment in T is one. Moreover we have the following observation about T.

Observation 6.2.3. *For a pair $(i,j) \in X$, let $S_{i,j}$ be the set of segments in S that intersect $t_{i,j}$. Then*

$$S_{i,j} = \begin{cases} \{s_{i+1,j}, s_{i,j+1}\} & \textit{when } i = 1, j = 1 \\ \{s_{i+1,j}, s_{i,j+1}, s_{i,j-1}\} & \textit{when } i = 1, 2 \le j \le n-1 \\ \{s_{i-1,j}, s_{i,j+1}, s_{i,j-1}\} & \textit{when } i = n, 2 \le j \le n-1 \\ \{s_{i+1,j}, s_{i-1,j}, s_{i,j+1}\} & \textit{when } 2 \le i \le n, j = 1 \\ \{s_{i+1,j}, s_{i-1,j}, s_{i,j-1}\} & \textit{when } 2 \le i \le n, j = n \\ \{s_{i+1,j}, s_{i-1,j}, s_{i,j+1}, s_{i,j-1}\} & \textit{when } 2 \le i \le n-1, 2 \le j \le n-1 \end{cases}$$

Proof. We shall prove the observation only for the case when $2 \le i \le n-1, 2 \le j \le n-1$ and i is odd. For the remaining cases similar arguments will suffice. Notice that when $(i,j) \in X$, we have $\{(i+1,j), (i-1,j), (i,j+1), (i,j-1)\} \subsetneq Y$ and therefore $s_{i+1,j}, s_{i-1,j}, s_{i,j+1}, s_{i,j-1}$ exists.

Since i is odd, the bottom and top endpoints of $t_{i,j}$ are $(x_{i,j+1}, y_{i,j+1} - 0.5)$ and $(x_{i,j+1}, y_{i,j+1} + 0.5)$, respectively. Recall that the left endpoint of $s_{i,j+1}$ is $(x_{i,j+1}, y_{i,j+1})$ and using Observation 6.2.2(i) we can infer that the right endpoint of $s_{i,j-1}$ is $(x_{i,j+1}, y_{i,j+1} - \epsilon)$. These facts imply that the segment $t_{i,j} \cap s_{i,j-1}$ is the right endpoint of $s_{i,j-1}$ and $t_{i,j} \cap s_{i,j-1}$ is the left endpoint of $s_{i,j+1}$. Due to Observation 6.2.2(ii) and 6.2.2(iii), the bottom endpoint of $t_{i,j}$ lies between the left and right endpoints of $s_{i-1,j}$ and has the same y-coordinate as that of $s_{i-1,j}$. Hence $t_{i,j} \cap s_{i-1,j} = \{(x_{i,j+1}, y_{i,j+1} - 0.5)\} = \{(x'_{i,j}, y'_{i,j})\}$. Similarly, we can show that $s_{i+1,j} = \{(x_{i,j+1}, y_{i,j+1} + 0.5)\} = \{(x'_{i,j}, y'_{i,j} + 1)\}$. This completes the proof. □

Using Observation 6.2.1 and Observation 6.2.3 we can infer that $S \cup T$ is a valid unit B_0-VPG representation of H. See Figure 6.2.1(b) for an example.

6.3 Algorithm for vertically-stabbed L-graphs

Given a vertically-stabbed L-representation of a graph G with n vertices, we shall give an $O(n^5)$-time 8-approximation algorithm to solve the MDS problem on G. Specifically, we prove the following theorem.

Theorem 6.3.1. *Given a vertically-stabbed* L*-representation of a graph* G *with* n *vertices, there is an* $O(n^5)$*-time* 8*-approximation algorithm to solve the* MDS *problem on* G.

In the rest of the chapter, $OPT(Q)$ and $OPT(Q_l)$ denote the cost of the optimum solution of an ILP formulation Q and LP formulation Q_l, respectively.

Overview of the algorithm: First, we solve the relaxed LP formulation of the ILP formulation of the MDS problem on the input vertically-stabbed L graph G and create two subproblems. We shall show that one of those two subproblems is equivalent to the SSR problem and the other is equivalent to the SRS problem (defined in Chapter 5). Due to Katz et al [107] we know these subproblems can be solved optimally in polynomial time. Moreover, due to Lemma 5.1.1 and 5.1.2 we know that the integrality gaps of each of these problems are at most two. We shall use the above facts to prove an upper bound on the approximation ratio of our algorithm. The running time of the algorithm becomes $O(n^5)$ where n is the number of vertices in the input graph [147]. We note that such techniques have been previously used to design approximation algorithms [1, 35, 85].

Now we describe our approximation algorithm for MDS problem on vertically-stabbed L graphs. Let G be a graph and $\mathcal{R} = \{\mathrm{L}_u\}_{u \in V}$ be a vertically-stabbed L-representation of G. Without loss of generality, we assume that (i) the vertical line $x = 0$ intersects all the L-paths in $\mathcal{R}$ and the x-coordinate of the corner point of each L-path in $\mathcal{R}$ is strictly less than 0, and (ii) whenever two distinct L-paths intersect in $\mathcal{R}$, they intersect at exactly one point.

For a vertex $u \in V(G)$, let $N[u]$ denote the closed neighbourhood of u in G, $H_u = \{c \in N[u] \colon \mathrm{L}_c \text{ intersects the horizontal segment of } \mathrm{L}_u\}$ and let V_u denote the set $N(u) \setminus H_u$. Based on these we have the following ILP (say Q) of the problem of finding a minimum dominating set of G.

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{v \in V(G)} x_v \\ \text{subject to} & \displaystyle\sum_{v \in H_u} x_v + \sum_{v \in V_u} x_v \geq 1,\ \forall u \in V(G) \\ x_v \in \{0,1\}, & \forall v \in V(G) \end{array}$$

Q

Let Q_l be the the relaxed LP formulation of Q and $\mathbf{Q}_l = \{x_v \colon v \in V(G)\}$ be an optimal solution of Q_l. Now we define the following sets.

$$A_1 = \left\{ u \in V(G) \colon \sum_{v \in H_u} x_v \geq \frac{1}{2} \right\}, A_2 = \left\{ u \in V(G) \colon \sum_{v \in V_u} x_v \geq \frac{1}{2} \right\}$$

$$H = \bigcup_{u \in A_1} H_u, V = \bigcup_{u \in A_2} V_u$$

Based on these, we consider the following two integer programs Q' and Q''.

minimize $\sum_{v \in H} x'_v$ subject to $\sum_{v \in H_u} x'_v \geq 1, \forall u \in A_1$ $x'_v \in \{0,1\}, \quad v \in H$ Q'	minimize $\sum_{v \in V} x''_v$ subject to $\sum_{v \in V_u} x''_v \geq 1, \forall u \in A_2$ $x''_v \in \{0,1\}, \quad v \in V$ Q''

Let Q'_l and Q''_l be the relaxed LP of Q' and Q'' respectively. Clearly, the solutions of Q' and Q'' gives a feasible solution for Q. Hence $OPT(Q) \leq OPT(Q') + OPT(Q'')$. For each $x_v \in \mathbf{Q}_l$, define $y_v = \min\{1, 2x_v\}$ and define $\mathbf{Y}_l = \{y_v\}_{x_v \in \mathbf{Q}_l}$. Notice that $\mathbf{Y}_l$ gives a solution to Q'_l and Q''_l. Therefore, $OPT(Q'_l) + OPT(Q''_l) \leq 4 \cdot OPT(Q_l)$. We have the following lemma.

Lemma 6.3.1. *$OPT(Q') \leq 2 \cdot OPT(Q'_l)$ and $OPT(Q'') \leq 2 \cdot OPT(Q''_l)$.*

Proof. Note that for each vertex $u \in A_1$, H_u is non-empty and for each $v \in H_u$, L_v intersects the horizontal segment of L_u. Let R be the set of horizontal segments of the L-paths representing the vertices in A_1 and S be the set of vertical segments of the L-paths representing the vertices in H. Since all horizontal segments in R intersect the vertical line $x = 0$ and the x-coordinates of the vertical segments in S is strictly less than 0, we can consider the horizontal segments in R as rightward directed rays. Hence, solving Q' is equivalent to solving the ILP, say $\mathcal{E}$, of the problem of finding a minimum cardinality subset of vertical segments S that intersects all rays in the set R of rightward-directed rays. Hence solving $\mathcal{E}$ is equivalent to solving an SRS instance with R and S as input. By Lemma 5.4.1, we have that

$$OPT(Q') = OPT(\mathcal{E}) \leq 2 \cdot OPT(\mathcal{E}_l) \leq 2 \cdot OPT(Q'_l)$$

where $\mathcal{E}_l$ is the relaxed LP of $\mathcal{E}$. Hence we have proof of the first part.

For the second part, using similar arguments as above, we can show that solving Q'' is equivalent to solving an SSR instance. Hence, by Lemma 5.1.1, we have that $OPT(Q'') \leq 2 \cdot OPT(Q''_l)$. Hence the proof follows. □

Proof of Theorem 6.3.1: Lemma 6.3.1 implies that solving Q' (resp. Q'') is equivalent to solving an SRS (resp. SSR) problem instance. Let A be the union of the solutions returned by the optimal algorithms for SRS problem and SSR problem (due to Katz et al. [107]), used to solve Q' and Q'' respectively. Hence,

$$|A| \leq 2(OPT(Q'_l) + OPT(Q''_l)) \leq 8 \cdot OPT(Q_l) \leq 8 \cdot OPT(Q)$$

Since Q_l consists of n variables where $n = |V|$, solving Q_l takes $O(n^5)$ time [147]. Solving both the SSR and SRS instances takes a total of $O(n \log n)$ time and therefore the total running time of the algorithm is $O(n^5)$.

6.4 Algorithm for unit B_0-VPG graphs

Given a unit B_0 representation of a graph G, we shall give an 18-approximation algorithm for the MDS problem on G. In fact, we shall prove the following stronger theorem.

Theorem 6.4.1. *Let S_1 and S_2 be sets of orthogonal unit length segments. Let $\mathcal{C}$ be the ILP of the problem of finding a minimum cardinality subset D of S_2 such that every segment in S_1 intersects some segment in D. There is an $O(n^5)$-time algorithm to compute a set $D' \subseteq S_2$ which gives a feasible solution of $\mathcal{C}$ and $|D'| \leq 18 \cdot OPT(\mathcal{C}_l)$ where $n = |S_1 \cup S_2|$ and $\mathcal{C}_l$ is the relaxed LP of $\mathcal{C}$.*

We shall use Theorem 6.4.1 to prove our result on unit B_k-VPG graphs, with $k \geq 1$. In the next section, we give an overview of the algorithm.

6.4.1 Overview of the algorithm

First, we solve the relaxed LP formulation $\mathcal{C}_l$ of $\mathcal{C}$ and create two subproblems. Since $\mathcal{C}$ consists of n variables where $n = |S_2|$, solving Q_l takes $O(n^5)$ time [147]. We shall show that these subproblems are equivalent to one of the following optimisation problems.

1. ***The Subset Unit Interval Domination*** **(SUID)** problem: In this problem, the inputs are (i) a set X of horizontal unit length segments, (ii) a set Y of vertical unit-length segments, and (iii) two sets X', Y' such that $X' \subseteq X$ and $Y' \subseteq Y$. The objective is to find a minimum cardinality subset D of $X \cup Y$ such that every horizontal (resp. vertical) segment in X' (resp. Y') intersects at least one horizontal (resp. vertical) segment in $D \cap X$ (resp. $D \cap Y$). Through out this article, $\mathcal{SUD}(X', X, Y', Y)$ shall denote an SUID instance.

2. ***The Unit Orthogonal Segment Stabbing*** **(UOSS)** problem: In this problem, the inputs are (i) two sets X_1, X_2 containing horizontal unit length segments and (ii) two sets Y_1, Y_2 containing vertical unit length segments. The objective is to find a minimum cardinality subset D of $X_2 \cup Y_2$ such that every horizontal (resp. vertical) segment in X_1 (resp. Y_1) intersect at least one vertical (resp. horizontal) segment in $D \cap Y_2$ (resp. $D \cap X_2$). Through out this article, $\mathcal{US}(X_1, Y_1, X_2, Y_2)$ shall denote a UOSS instance.

We shall show that the integrality gaps of these subproblems are bounded by some constants and hence admit constant factor approximation algorithms. Specifically, we shall prove the following lemmas.

Lemma 6.4.1. *Let X (resp. Y) be a set of horizontal (resp. vertical) unit length segments. For $X' \subseteq X$ and $Y' \subseteq Y$, let $\mathcal{A}$ be the ILP formulation of the $\mathcal{SUD}(X', X, Y', Y)$ instance. Then $OPT(\mathcal{A}) = OPT(\mathcal{A}_l)$ where $\mathcal{A}_l$ is the relaxed LP of $\mathcal{A}$. Moreover, $OPT(\mathcal{A})$ can be computed in $O(n \log n)$ time where $n = |X| + |Y|$.*

Lemma 6.4.2. *Let X_1, X_2 (resp. Y_1, Y_2) be sets of horizontal (resp. vertical) unit length segments. Let $\mathcal{B}$ be the ILP formulation of the $\mathcal{US}(X_1, Y_1, X_2, Y_2)$ instance. Then there is an $O(n^5)$-time algorithm to compute a set $D' \subseteq X_2 \cup Y_2$ which gives a feasible solution of $\mathcal{B}$ and $|D'| \leq 8 \cdot OPT(\mathcal{B}_l)$ where $n = |X_1 \cup X_2 \cup Y_1 \cup Y_2|$ and $\mathcal{B}_l$ is the relaxed LP of $\mathcal{B}$.*

In Section 6.4.2, we prove Lemma 6.4.1. Then in Section 6.4.3, we shall prove Lemma 6.4.2 using Lemma 5.1.1. Using Lemma 6.4.1 and Lemma 6.4.2, we shall complete the proof of Theorem 6.4.1 in Section 6.4.4.

6.4.2 Proof of Lemma 6.4.1

Recall that X is a set of horizontal unit length segments, Y is a set of vertical unit length segments, $X' \subseteq X, Y' \subseteq Y$ and $\mathcal{A}$ is the ILP formulation of the $\mathcal{SUD}(X', X, Y', Y)$ instance.

Let $\mathcal{A}'$ be the ILP formulation of the problem of finding a subset D_1 of X with minimum cardinality such that any segment in X' intersects a segment in D_1. Let $\mathcal{A}''$ be the ILP formulation of the problem of finding a subset D_2 of Y with minimum cardinality such that any segment in Y' intersects a segment in D_2. Observe that, $OPT(\mathcal{A}) = OPT(\mathcal{A}') + OPT(\mathcal{A}'')$ and $OPT(\mathcal{A}_l) = OPT(\mathcal{A}'_l) + OPT(\mathcal{A}''_l)$ where $\mathcal{A}'_l$ and $\mathcal{A}''_l$ are the relaxed LP formulations of $\mathcal{A}'$ and $\mathcal{A}''$, respectively. Now we have the following observation.

Observation 6.4.1. *$OPT(\mathcal{A}') = \mathcal{A}'_l$ and $OPT(\mathcal{A}'') = OPT(\mathcal{A}''_l)$.*

Proof. We shall only prove the observation for $OPT(\mathcal{A}')$ as similar arguments will suffice for the other case. Let $X_i' \subseteq X'$ be the set of all horizontal segments whose y-coordinate is i. Similarly let $X_i \subseteq X$ be the set of all horizontal segments whose y-coordinate is i. Let $\mathcal{A}_i'$ be the ILP formulation of the problem of finding a subset D_i' of X with minimum cardinality such that any segment in X_i' intersects a segment in D'. Observe that, $OPT(\mathcal{A}') = \sum_i OPT(\mathcal{A}_i')$ and $OPT(\mathcal{A}_l') = \sum_i OPT(\mathcal{A}_{i,l}')$ where $\mathcal{A}_{i,l}'$ is the relaxed LP formulation of $\mathcal{A}_i'$. Now we prove the following claim.

Claim. For each i, $OPT(\mathcal{A}_i') = OPT(\mathcal{A}_{i,l}')$.

To prove the claim first define for each horizontal segment $h \in X_i$, let $l(h)$ denote the left endpoints of h. Let $h_1, h_2, \ldots, h_k$ be the segments in X_i sorted in the ascending order of the x-coordinates of $l(h)$. For a segment $h' \in X_i'$, let $N(h')$ denote the set of intervals in X_i that intersect h'. Let $\mathcal{M}$ be the coefficient matrix of $\mathcal{A}_i'$ such that the i^{th} column of $\mathcal{M}$ corresponds to the variable corresponding to $h_i \in X_i$. Observe that in each row of $\mathcal{M}$, the set of 1's are consecutivel. Therefore, $\mathcal{M}$ is a totally unimodular matrix [141]. Thus any optimal solution of $\mathcal{A}_{i,l}'$ is integral. Thus we have the proof of the claim.

Hence $OPT(\mathcal{A}') = \sum_i OPT(\mathcal{A}_i') = \sum_i OPT(\mathcal{A}_{i,l}') = OPT(\mathcal{A}_l')$. This completes the proof of the observation. ☐

Using the above observation, we have that $OPT(\mathcal{A}) = OPT(\mathcal{A}') + OPT(\mathcal{A}'') = OPT(\mathcal{A}_l') + OPT(\mathcal{A}_l'') = OPT(\mathcal{A}_l)$. This completes the proof of the lemma.

6.4.3 Proof of Lemma 6.4.2

Recall that X_1, X_2 are sets of horizontal unit length segments, Y_1, Y_2 are sets of vertical unit length segments and $\mathcal{B}$ is the ILP formulation of the $\mathcal{US}(X_1, Y_1, X_2, Y_2)$ instance.

Let $\mathcal{B}'$ be an ILP formulation of the problem of finding a subset D' of Y_2 with minimum cardinality such that any segment in X_1 intersects a segment in D'. Let $\mathcal{B}''$ be an ILP formulation of the problem of finding a subset D'' of X_2 with minimum cardinality such that any segment in Y_1 intersects a segment in D''. Observe that, $OPT(\mathcal{B}) = OPT(\mathcal{B}') + OPT(\mathcal{B}'')$ and $OPT(\mathcal{B}_l) = OPT(\mathcal{B}'_l) + OPT(\mathcal{B}''_l)$ where $\mathcal{B}'_l$ and $\mathcal{B}''_l$ are the relaxed LP formulations of $\mathcal{B}'$ and $\mathcal{B}''$, respectively. Now we prove the following proposition.

Lemma 6.4.3. *$OPT(\mathcal{B}') \leq 8 \cdot OPT(\mathcal{B}'_l)$ and $OPT(\mathcal{B}'') \leq 8 \cdot OPT(\mathcal{B}''_l)$.*

Proof. We shall only prove the lemma for $OPT(\mathcal{B}'')$ as similar arguments suffice for the other case. Let $X_2 = S$ and $Y_1 = T$ and let $\mathcal{I}_S$ be the set of intervals obtained by projecting the horizontal segments in S onto the x-axis. Observe that $\mathcal{I}_S$ is set of unit intervals.

Without loss of generality, we assume that (i) no two interval in $\mathcal{I}_S$ contain each other, and (ii) x-coordinate of any vertical segment in T is distinct from the left and right endpoints of any interval in $\mathcal{I}_S$. Since no two interval in $\mathcal{I}_S$ contain each other, there exists a set P of real numbers such that each interval in $\mathcal{I}_S$ contains exactly one real number from P. (To see this, consider the right endpoints of the intervals in the maximum cardinality subset of $\mathcal{I}_S$ with pairwise non-intersecting intervals which is obtained using the greedy algorithm [112]). Add in P two more dummy values q, q' which are not contained in any interval in $\mathcal{I}_S$ such that q (resp. q') is less than (resp. greater than) that of all values in P. Let $p_1, p_2, \ldots, p_t$ be the values in P sorted in the ascending order (notice that $p_1 = q$ and $p_t = q'$). For each $i \in \{1, 2, \ldots, t-1\}$, let T_i denote the vertical segments of T that lies inside the strip bounded by the lines $y = p_i$ and $y = p_{i+1}$. Due to our assumptions, for any $i \neq j$, T_i and T_j are disjoint. For each $i \in \{1, 2, \ldots, t-1\}$, and each vertical segment $v \in T_i$, let S_v^{left} (resp. S_v^{right}) be the subset of S that intersects v and the line $y = p_i$ (resp. $y = p_{i+1}$). Since any interval in $\mathcal{I}_S$ contains exactly one value from $\{p_i, p_{i+1}\}$, we have that $S_v^{left} \cap S_v^{right} = \emptyset$, for each vertical segment $v \in T$. Based on these we have the following equivalent ILP formulation (say W) of $\mathcal{B}''$.

$$\begin{array}{lll} \text{minimize} & \sum_{v \in S} x_v & \\ \text{subject to} & \sum_{v \in S_u^{left}} x_v + \sum_{v \in S_u^{right}} x_v \geq 1, & \forall u \in T \\ & x_v \in \{0,1\}, & \forall v \in S \end{array}$$

W

Let $\mathbf{W}_l = \{x_v : v \in S\}$ be an optimal solution of the relaxed LP formulation (say W_l) of W. Consider the following sets.

$$A_1 = \left\{ u \in T : \sum_{v \in S_u^{left}} x_v \geq \frac{1}{2} \right\}, A_2 = \left\{ u \in T : \sum_{v \in S_u^{right}} x_v \geq \frac{1}{2} \right\}$$

$$L = \bigcup_{v \in A_1} S_v^{left}, R = \bigcup_{v \in A_2} S_v^{right}$$

Based on these, we consider the following two integer programs W' and W''.

$\begin{array}{ll} \text{minimize} & \sum_{v \in L} x'_v \\ \text{subject to} & \sum_{v \in S_u^{left}} x'_v \geq 1, \forall u \in A_1 \\ & x'_v \in \{0,1\}, \quad v \in L \end{array}$ W'	$\begin{array}{ll} \text{minimize} & \sum_{v \in R} x''_v \\ \text{subject to} & \sum_{v \in S_u^{right}} x''_v \geq 1, \forall u \in A_2 \\ & x''_v \in \{0,1\}, \quad v \in R \end{array}$ W''

Let W'_l and W''_l be the corresponding relaxed LPs of W' and W'' respectively. The union of the solutions of W' and W'' gives a solution for W implying $OPT(W) \leq OPT(W') + OPT(W'')$. For each $x_v \in \mathbf{W}_l$, define $y_v = \min\{1, 2x_v\}$ and define $\mathbf{Y}_l = \{y_v\}_{x_v \in \mathbf{W}_l}$. Notice that $\mathbf{Y}_l$ gives a solution to W'_l (and W''_l). Hence, $OPT(W'_l) \leq 2 \cdot OPT(W_l)$ and $OPT(W''_l) \leq 2 \cdot OPT(W_l)$. Therefore, $OPT(W'_l) + OPT(W''_l) \leq 4 \cdot OPT(W_l)$. Notice that, solving W' (resp. W'') is equivalent to the problem of finding a minimum cardinality subset of the horizontal segments in L (resp. R) to intersect all vertical segments in A_1 (resp. A_2). Now we have the following claim.

Claim. $OPT(W') \leq 2 \cdot OPT(W'_l)$ *and* $OPT(W'') \leq 2 \cdot OPT(W''_l)$.

We shall prove the above claim only for W' as proof for the other case is similar. Recall that solving W' is equivalent to the problem of finding a minimum cardinality subset of the horizontal segments in the set L (defined earlier) to intersect all vertical segments in A_1. For each $i \in \{1, 2, \ldots, (t-1)\}$ let $T_{1,i} = A_1 \cap T_i$ and L_i be the set of horizontal segments in L that intersect some vertical segment in $T_{1,i}$. Formally, $L_i = \bigcup_{v \in T_{1,i}} S_v^{left}$. For any $i \neq j$, $T_{1,i} \cap T_{1,j} = \emptyset$ and let $L_i \cap L_j = \emptyset$ (this follows from the fact no horizontal segment in S intersects both $y = p_i$ and $y = p_j$). For each $i \in \{1, 2, \ldots, (t-1)\}$, let $\mathcal{D}_i$ (resp, $\mathcal{D}_{i,l}$) denote the ILP (resp. relaxed LP) of the problem of selecting minimum subset D_i horizontal segments in L_i such that all vertical segments in $T_{1,i}$ intersect at least one horizontal segment in D_i. Clearly, $OPT(W') = \sum_{i=1}^{t-1} OPT(\mathcal{D}_i)$ and $OPT(W'_l) = \sum_{i=1}^{t-1} OPT(\mathcal{D}_{i,l})$. For each $i \in \{1, 2, \ldots, (t-1)\}$ notice that, all horizontal segments intersect the vertical line $y = p_i$ and all vertical segments in $T_{1,i}$ lies to the left of the vertical line $y = p_i$. For each $i \in \{1, 2, \ldots, (t-1)\}$ if we consider the segments in L_i to be leftward-directed rays then solving $\mathcal{D}_i$ is equivalent to solving an SSR instance with $T_{1,i}$ and L_i as input. Due to Lemma 5.1.1, for each $i \in \{1, 2, \ldots, (t-1)\}$, $OPT(\mathcal{D}_i) \leq 2 \cdot OPT(\mathcal{D}_{i,l})$. Hence,

$$OPT(W') = \sum_{i=1}^{t-1} OPT(\mathcal{D}_i) \leq 2 \cdot \sum_{i=1}^{t-1} OPT(\mathcal{D}_{i,l}) = 2 \cdot OPT(W'_l)$$

This completes the proof of the claim.

Using the above claim and previous observations, we can infer that $OPT(W) \leq OPT(W') + OPT(W'') \leq 2(OPT(W'_l) + OPT(W''_l)) \leq$

$8 \cdot OPT(W_l')$. This completes the proof of the proposition. □

Hence, Observe that, $OPT(\mathcal{B}) = OPT(\mathcal{B}') + OPT(\mathcal{B}'') \leq 8(OPT(\mathcal{B}_l') + OPT(\mathcal{B}_l'')) = 8 \cdot OPT(\mathcal{B}_l)$. This completes the proof of the lemma.

6.4.4 Completion of proof of Theorem 6.4.1

Recall that S_1 and S_2 are sets of orthogonal unit length segments, $\mathcal{C}$ is an ILP formulation of the problem of finding a minimum cardinality subset D of S_2 such that every segment in S_1 intersects some segment in D. We shall give an $O(n^5)$-time algorithm to compute a set $D' \subseteq S_2$ which gives a feasible solution of $\mathcal{C}$ and $|D'| \leq 18 \cdot OPT(\mathcal{C}_l)$ where $n = |S_1 \cup S_2|$ and $\mathcal{C}_l$ is the relaxed LP formulation of $\mathcal{C}$.

Let V_1 and H_1 be the sets of vertical and horizontal segments in S_1, respectively. Similarly, let V_2 and H_2 be the sets of vertical and horizontal segments in S_2, respectively. For $v \in V_1 \cup H_1$, let $N(v) \subseteq V_2 \cup H_2$ denote the set of segments that intersects v. For $w \in H_1$, let $N_o(w) = N(w) \cap H_2$ and for $w \in V_1$ let $N_o(w) = N(w) \cap V_2$. Based on these we have the following equivalent ILP formulation (say Z) of $\mathcal{C}$.

$$
\begin{array}{lll}
\text{minimize} & \sum_{w \in V_2 \cup H_2} x_w & \\
\text{subject to} & \sum_{w \in N_o(u)} x_w + \sum_{w \in N(u) \setminus N_o(u)} x_w \geq 1, & \forall u \in V_1 \cup H_1 \\
& x_w \in \{0, 1\}, & \forall w \in V_2 \cup H_2
\end{array}
$$

Z

The first step of our algorithm is to solve the relaxed LP formulation (say Z_l) of Z. Let $\mathbf{Z}_l = \{x_w \colon w \in V_2 \cup H_2\}$ be an optimal solution of Z_l. Let

$$A_1 = \left\{ u \in V_1 \cup H_1 \colon \sum_{w \in N_o(u)} x_w \geq \frac{1}{2} \right\}$$

$$A_2 = \left\{ u \in V_1 \cup H_1 \colon \sum_{w \in N(u) \setminus N_o(u)} x_w \geq \frac{1}{2} \right\}$$

,

$$B_1 = \bigcup_{u \in A_1} N_o(u),\ \ B_2 = \bigcup_{u \in A_2} N(u) \setminus N_o(u)$$

Based on these, we consider the following two integer programs Z' and Z''.

minimize $\sum_{w \in B_1} x'_w$ subject to $\sum_{w \in N_o(v)} x'_w \geq 1, \forall v \in A_1$ $x'_w \in \{0,1\}, \quad w \in B_1$ Z'	minimize $\sum_{w \in B_2} x''_w$ subject to $\sum_{w \in N(v) \setminus N_o(v)} x''_w \geq 1, \forall v \in A_2$ $x''_w \in \{0,1\}, \quad w \in B_2$ Z''

Let Z'_l and Z''_l be the corresponding relaxed LPs of Z' and Z'' respectively. Clearly, the union of the solutions of Z' and Z'' gives a solution for Z. Hence, $OPT(Z) \leq OPT(Z') + OPT(Z'')$. For each $x_v \in \mathbf{Z}_l$, define $y_v = \min\{1, 2x_v\}$ and define $\mathbf{Y}_l = \{y_v\}_{x_v \in \mathbf{Z}_l}$. Notice that $\mathbf{Y}_l$ gives a solution for Z'_l and Z''_l. Hence, $OPT(Z'_l) \leq 2 \cdot OPT(Z_l)$ and $OPT(Z''_l) \leq 2 \cdot OPT(Z_l)$. Now we prove the following lemma.

Lemma 6.4.4. *$OPT(Z') = OPT(Z'_l)$ and $OPT(Z'') \leq 8 \cdot OPT(Z''_l)$.*

Proof. To prove the first part, let X (resp. Y) be the set of horizontal (resp. vertical) segments in B_1 and X' (resp. Y') be the set of horizontal (resp. vertical) segments in A_1. Notice that $X' \subseteq X$ and $Y' \subseteq Y$. Hence, Z' is the ILP formulation of finding minimum cardinality subset D of $X \cup Y$ such that every horizontal (resp. vertical) segment in X' (resp. Y') intersects at least one horizontal (resp. vertical) segment in $D \cap X$ (resp. $D \cap Y$). By Lemma 6.4.1, we have that $OPT(Z') = OPT(Z'_l)$.

To prove the second part, let X_1 and X_2 (resp. Y_1 and Y_2) be the sets of horizontal (resp. vertical) segments in A_2 and B_2, respectively. Notice that Z'' is the ILP formulation of finding minimum cardinality subset D of $X_2 \cup Y_2$ such that every horizontal (resp. vertical) segment in X_1 (resp. Y_1) intersects at least one vertical (resp. horizontal) segment in

$D \cap Y_2$ (resp. $D \cap X_2$). By Lemma 6.4.2, we have that $OPT(Z'') \leq 8 \cdot OPT(Z''_l)$. □

Using Lemma 6.4.4 and previous arguments, we can conclude that in $O(n^5)$ time it is possible to compute a set $D' \subseteq S_2$ which gives a feasible solution of Z where $n = |S_1 \cup S_2|$. Moreover, $|D'| \leq OPT(Z') + OPT(Z'') \leq OPT(Z'_l) + 8 \cdot OPT(Z''_l) \leq 18 \cdot OPT(Z_l) \leq 18 \cdot OPT(\mathcal{C}_l)$. This completes the proof of the theorem.

6.5 Algorithm for unit B_k-VPG graphs

In this section, we present our approximation algorithm for the MDS problem on unit B_k-VPG graphs, for $k \geq 1$.

Let $\mathcal{R}$ be a unit B_k-VPG representation of a unit B_k-VPG graph G. Throughout this section, we assume that the segments of each path $P \in \mathcal{R}$ are numbered consecutively starting from the leftmost segment by $1, 2, \ldots, t$ where $t(\leq k+1)$ is the number of segments in P. For a path $P \in \mathcal{R}$, let $N[P]$ denote the set of paths in $\mathcal{R}$ that intersect P.

Define $\Phi\colon \mathcal{R} \times \mathcal{R} \to \mathbb{N} \times \mathbb{N}$ such that for two paths $P, Q \in \mathcal{R}$, $\Phi(P,Q) = (i,j)$ if and only if the i^{th} segment of P intersects the j^{th} segment of Q, and for all $1 \leq a < i$, the a^{th} segment of P does not intersect any segment of Q.

For a path $P \in \mathcal{R}$, let $\mathcal{X}_P(i,j) = \{Q \in N[P]\colon \Phi(P,Q) = (i,j)\}$. For distinct pairs (i,j) and (i',j') the sets $\mathcal{X}_P(i,j)$ and $\mathcal{X}_P(i',j')$ are disjoint. Let $\mathcal{K}$ denote the set $\{1, 2, \ldots, k+1\} \times \{1, 2, \ldots, k+1\}$. Based on these we have the following ILP formulation of the MDS problem on G.

$$\begin{array}{lll}
\text{minimize} & \displaystyle\sum_{Q \in \mathcal{R}} x_Q & \\
\text{subject to} & \displaystyle\sum_{(i,j) \in \mathcal{K}} \sum_{Q \in \mathcal{X}_P(i,j)} x_Q \geq 1, & \forall P \in \mathcal{R} \\
 & x_Q \in \{0,1\}, & \forall P \in \mathcal{R}
\end{array}$$

Z

The first step of our algorithm is to solve the relaxed LP formulation (say Z_l) of Z. Let $\mathbf{Z}_l = \{x_Q : Q \in \mathcal{R}\}$ be an optimal solution of Z_l. For each path $P \in \mathcal{R}$, there is a pair $(i,j) \in \mathcal{K}$ such that $\sum_{Q \in \mathcal{X}_P(i,j)} x_Q \geq \frac{1}{(k+1)^2}$. For each pair $(i,j) \in \mathcal{K}$, define

$$\mathcal{A}(i,j) = \left\{ P \in \mathcal{R} \colon \sum_{Q \in \mathcal{X}_P(i,j)} x_Q \geq \frac{1}{(k+1)^2} \right\}, \mathcal{B}(i,j) = \bigcup_{P \in \mathcal{A}(i,j)} \mathcal{X}_P(i,j)$$

Based on these we have the following ILP formulation for each pair $(i,j) \in \mathcal{K}$.

$$\begin{array}{ll} \text{minimize} & \sum_{Q \in \mathcal{B}(i,j)} x'_Q \\ \text{subject to} & \sum_{Q \in \mathcal{X}_P(i,j)} x'_Q \geq 1, \ \forall P \in \mathcal{A}(i,j) \\ & x'_Q \in \{0,1\}, \qquad \forall Q \in \mathcal{B}(i,j) \\ & \qquad Z(i,j) \end{array}$$

For each pair $(i,j) \in \mathcal{K}$, let $Z_l(i,j)$ be the relaxed LP formulation of $Z(i,j)$. We have the following

$$OPT(Z) \leq \sum_{(i,j) \in \mathcal{K}} OPT(Z(i,j))$$

For each $x_P \in \mathbf{Z}_l$, define $y_P = \min\{1, x_P(k+1)^2\}$ and define $\mathbf{Y}_l = \{y_P\}_{x_P \in \mathbf{Z}_l}$. Clearly, $\mathbf{Y}_l$ gives a solution to $Z_l(i,j)$ for each $(i,j) \in \mathcal{K}$. Moreover,

$$\sum_{(i,j) \in \mathcal{K}} OPT(Z_l(i,j)) \leq (k+1)^4 \cdot OPT(Z_l)$$

Now we have the following lemma.

Lemma 6.5.1. *For each pair $(i,j) \in \mathcal{K}$, there is a solution $D(i,j)$ for $Z(i,j)$ such that $|D(i,j)| \leq 18 \cdot OPT(Z_l(i,j))$.*

Proof. For any $(i,j) \in \mathcal{K}$, solving $Z(i,j)$ is equivalent to finding a minimum cardinality subset D of $\mathcal{B}(i,j)$ such that each path $P \in \mathcal{A}(i,j)$ intersects at least one path is $D \cap \mathcal{X}_P(i,j)$. Notice that, for each $P \in \mathcal{A}(i,j)$ the set $\mathcal{X}_u(i,j)$ is non-empty and for each $Q \in \mathcal{X}_P(i,j)$, the i^{th} segment of P intersects the j^{th} segment of Q. Let $S_1 = \{i^{th}$ segment of $P\colon P \in \mathcal{A}(i,j)\}$, $S_2 = \{j^{th}$ segment of $Q\colon Q \in \mathcal{B}(i,j)\}$.

Solving $Q(i,j)$ is equivalent to the problem finding a minimum cardinality subset D of S_2 such that every segment in S_1 intersect at least one segment in D. Moreover, every segment in $S_1 \cup S_2$ have unit length. Hence by Theorem 6.4.1, we have a solution (say $D(i,j)$) for $Z(i,j)$ such that $|D(i,j)| \leq 18 \cdot OPT(Z_l(i,j))$. □

For each pair $(i,j) \in \mathcal{K}$, due to Lemma 6.5.1, we have a solution $D(i,j)$ of $Z(i,j)$ such that $|D(i,j)| \leq 18 \cdot OPT(Z_l(i,j))$ in polynomial time. Let D be the union of $D(i,j)$'s for all $(i,j) \in \mathcal{K}$. We have that

$$\begin{aligned}
|D| &= \sum_{(i,j)\in\mathcal{K}} |D(i,j)| \\
&\leq \sum_{(i,j)\in\mathcal{K}} 18 \cdot OPT(Z_l(i,j)) \\
&\leq 18 \cdot (k+1)^4 \cdot OPT(Z_l) \leq 18 \cdot (k+1)^4 \cdot OPT(Z)
\end{aligned}$$

Notice that, in $O(k^2 n^5)$ time it is possible to construct the set D.

6.6 Concluding remarks and open problems

In this chapter, we proposed several approximation algorithms for the MDS problem on subclasses of string graphs. Using our results on the SSR problem and theSRS problem, we gave the first constant factor approximation algorithm for the MDS problem on vertically-stabbed L-graphs and unit B_0-VPG graphs. However, we believe that obtained approximation ratio of 18 to be far from being tight. This motivates the following questions.

Question 6.6.1. *Is there a c-approximation algorithm for the* MDS *problem on vertically-stabbed* L*-graphs with* $c < 8$*?*

Question 6.6.2. *Is there a c-approximation algorithm for the* MDS *problem on unit* B_0*-VPG graphs with* $c < 18$*?*

Using our results on SSR and SRS problems, we have an $O(k^4)$-approximation algorithm for the MDS problem on unit B_k-VPG graphs. It is unlikely that there is $o(\log k)$-approximation algorithm for MDS problem on B_k-VPG graphs. This naturally leads to the following question(s).

Question 6.6.3. *Is there an* $O(\log k)$*-approximation algorithm for the* MDS *problem on* B_k*-VPG graphs or unit* B_k*-VPG graphs ?*

7
Conclusion

In this thesis, we studied the forbidden structures of rectangle intersection graphs in terms of its stab number. We proposed polynomial-time certifying recognition algorithms for several subclasses of rectangle intersection graphs with stab number at most 3. Then we studied the computational complexity of the MDS problem on string graphs and its subclasses. We proposed constant factor approximation algorithms for the MDS problem on stabbed rectangle overlap graphs and other subclasses of string graphs. Apart from the open problems posed at the end of the respective chapters, this thesis motivates the following directions of research.

7.1 Certifying recognition algorithms for string graphs

One of the main objectives of this thesis was to study the forbidden structures of rectangle intersection graphs and see if it is possible to devise a certifying recognition algorithm. But it seems that developing a certifying recognition algorithm of rectangle intersection graph will require more effort. On the other hand, it might be easier to propose a certifying algorithm for string graphs which is a more general graph class than rectangle intersection graphs. It is a widespread belief that any non-string graph will contain an *induced full subdivision* of a non-planar graph. A proof for the above will yield a forbidden structure characterisation of string graphs.

Question 7.1.1. *Is it true that a graph is a string graph if and only if it does not contain an induced full subdivision of a non-planar graph?*

Answer to the above question would increase our understanding of the structure of string graphs. Note that an affirmative answer to the above question does not immediately give a certifying recognition algorithm for string graphs. We also need to study the computational complexity of deciding whether a given graph has an induced full subdivision of some non-planar graph.

7.2 Approximation algorithms for the MDS problem on string graphs

In this thesis, we proposed constant factor approximation algorithms for MDS problems on stabbed rectangle overlap graphs, vertically-stabbed L-graphs and unit B_k-VPG graphs for each fixed $k \geq 0$. All the results mentioned above are consequences of the bounded integrality gap of the

SSR problem. The SSR problem is a special case of the following "stabbing type" problems. Given two collections of geometric objects S and T, let $\textsc{Stab}(S, T)$ denote the problem of finding the minimum cardinality subset D of T such that each object in S intersects some object in D. Observe that when $S = T$, $\textsc{Stab}(S, T)$ is equivalent to solving the MDS problem on intersection graphs of S. Researchers have studied different $\textsc{Stab}(S, T)$ problems by putting restrictions on S and T [85, 107]. Let H be a set of horizontal segments, and V be a set of vertical segments. We noticed that proof of a constant upper bound on the integrality gap of the $\textsc{Stab}(V, H)$ problem and a polynomial time rounding algorithm would give an $f(k)$-approximation factor algorithm for the MDS problem on B_k-VPG graphs, for each $k \geq 0$. This motivates the following questions.

Question 7.2.1. *Prove bounds on the integrality gaps and design approximation algorithms for "stabbing type" problems.*

A constant upper bound on the integrality gap of the $\textsc{Stab}(V, H)$ problem would also give a constant factor approximation algorithm for the MDS problem on rectangle overlap graphs. Since two rectangles intersect at most four times, rectangle overlap graphs are subclasses of *4-string* graphs, i.e. intersection graphs of a set of simple curves where two curves intersect at most four times. An important parameter of a string graph is its *rank*. For any $d > 0$, graph G is a string graph of *rank* d if it is an intersection graph of simple curves where two curves have at most d crossings [131]. It would be interesting to see if there is a $\phi(d)$-approximation algorithm for the MDS problem on string graphs with rank at most d.

Question 7.2.2. *Is there a polynomial-time $\phi(d)$-approximation algorithm for the* MDS *problem on string graphs with rank at most d?*

It turns out that the $\textsc{Stab}$(V,H) problem is a special case of the following graph-theoretic problem. Let G be a planar graph, and $\mathcal{P}$ be a collection of paths of G. The HIT$(G, \mathcal{P})$ problem is to select a minimum cardinality subset S of V such that each path in $\mathcal{P}$ contains at least one vertex from S. Observe that, the HIT$(G, \mathcal{P})$ problem is a generalisation of the Minimum Vertex Cover problem on planar graphs. Studying the HIT$(G, \mathcal{P})$ problem and its possible variants are of independent interest.

Question 7.2.3. *What is the optimal approximation ratio for the the* HIT$(G, \mathcal{P})$ *problem?*

Observe that the HIT$(G, \mathcal{P})$ problem is equivalent to solving the geometric Stab$(\mathcal{C}, P)$ problem where $\mathcal{C}$ is a given set of simple curves and P is a given set of points. Researchers have studied different Stab$(\mathcal{C}, P)$ problems by putting restrictions on $\mathcal{C}$ [79]. This motivates the following direction of research.

Question 7.2.4. *What is the optimal approximation ratio for solving the* Stab$(\mathcal{C}, P)$ *problem where* $\mathcal{C}$ *is a set of simple curves on the plane and* P *is a set of points on the plane?*

Journal publications by the author of the thesis

J1. D. Chakraborty, M. C. Francis: On stab number of rectangle intersection graphs. Theory of Computing Systems (Appeared online).

Conference publications by the author of the thesis

C1. D. Chakraborty, S. Das, J. Mukherjee: Approximating Minimum Dominating Set on String graphs. International Workshop on Graph-Theoretic Concepts in Computer Science 2019, Springer, LNCS, pages = 232–243.

C2. D. Chakraborty, S. Das, J. Mukherjee: Dominating set on overlap graphs of rectangles intersecting a line. Computing and Combinatorics Conference 2019, Springer, LNCS, pages = 65–77.

C3. D. Chakraborty, S. Das, M. C. Francis, S. Sen: On rectangle intersection graphs with stab number at most two. Conference on Algorithms and Discrete Applied Mathematics 2019, Springer, LNCS, pages = 124–137.

C4. S. Bhore, D. Chakraborty, S. Das, S. Sen: On local structures of cubicity 2 graphs. Combinatorial Optimization and Applications 2016, Springer, LNCS, pages = 254–269.

References

[1] A. Acharyya, S. C. Nandy, S. Pandit, and S. Roy. Covering segments with unit squares. *Computational Geometry*, 79:1–13, 2019.

[2] A. Adamaszek, P. Chalermsook, and A. Wiese. How to tame rectangles: solving independent set and coloring of rectangles via shrinking. In *proceedings of Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM*, volume 40 of *LIPIcs*, pages 43–60. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

[3] A. Adamaszek, S. Har-Peled, and A. Wiese. Approximation schemes for independent set and sparse subsets of polygons. *Journal of the ACM*, 66(4):29, 2019.

[4] A. Adiga, D. Bhowmick, and L. S. Chandran. The hardness of approximating the boxicity, cubicity and threshold dimension of a graph. *Discrete Applied Mathematics*, 158(16):1719–1726, 2010.

[5] A. Adiga, D. Bhowmick, and L. S. Chandran. Boxicity and poset dimension. *SIAM Journal on Discrete Mathematics*, 25(4):1687–1698, 2011.

[6] A. Adiga, L. S. Chandran, and N. Sivadasan. Lower bounds for boxicity. *Combinatorica*, 34(6):631–655, 2014.

[7] P. K. Agarwal and N. H. Mustafa. Independent set of intersection graphs of convex objects in 2D. *Computational Geometry*, 34(2):83–95, 2006.

[8] P. K. Agarwal, M. Van Kreveld, and S. Suri. Label placement by maximum independent set in rectangles. *Computational Geometry*, 11(3-4):209–218, 1998.

[9] A. Asinowski, E. Cohen, M. C. Golumbic, V. Limouzy, M. Lipshteyn, and M. Stern. Vertex intersection graphs of paths on a grid. *Journal of Graph Algorithms and Applications*, 16(2):129–150, 2012.

[10] E. Asplund and B. Grünbaum. On a coloring problem. *Mathematica Scandinavica*, 8(1):181–188, 1960.

[11] A. Atminas and V. Zamaraev. On forbidden induced subgraphs for unit disk graphs. *Discrete & Computational Geometry*, 60(1):57–97, 2018.

[12] J. Babu, M. Basavaraju, L. S. Chandran, D. Rajendraprasad, and N. Sivadasan. Approximating the cubicity of trees. *arXiv:1402.6310*, 2014.

[13] S. Bandyapadhyay, A. Maheshwari, S. Mehrabi, and S. Suri. Approximating dominating set on intersection graphs of rectangles and L-frames. In *proceedings of Mathematical Foundations of Computer Science, MFCS*, LIPIcs, pages 1–15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[14] S. Bandyapadhyay and S. Mehrabi. Constrained orthogonal segment stabbing. *arXiv:1904.13369*, 2019.

[15] S. Bandyapadhyay and A. B. Roy. Effectiveness of local search for art gallery problems. In *proceedings of International Symposium*

on Algorithms and Data Structures (WADS), LNCS, pages 49–60. Springer, 2017.

[16] R. Bar-Yehuda, D. Hermelin, and D. Rawitz. Minimum vertex cover in rectangle graphs. *Computational Geometry*, 44(6-7):356–364, 2011.

[17] S. Benzer. On the topology of the genetic fine structure. *Proceedings of the national Academy of Sciences*, 45(11):1607–1620, 1959.

[18] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational geometry: algorithms and applications.* Springer, 2008.

[19] C. Berge. *The Theory of Graphs.* Dover books on mathematics. Dover, 2001.

[20] P. Berman, B. Das Gupta, S. Muthukrishnan, and S. Ramaswami. Efficient approximation algorithms for tiling and packing problems with rectangles. *Journal of Algorithms*, 41(2):443–470, 2001.

[21] B. K. Bhattacharya, M. De, S. C. Nandy, and S. Roy. Maximum independent set for interval graphs and trees in space efficient models. In *proceedings of Canadian Conference on Computational Geometry*, 2014.

[22] D. Bhowmick and L. S. Chandran. Boxicity and cubicity of asteroidal triple free graphs. *Discrete Mathematics*, 310(10-11):1536–1543, 2010.

[23] A. Bielecki. Problem 56. *Colloquium Mathematicum*, 1:333–334, 1948.

[24] K. P. Bogart and D. B. West. A short proof that 'proper= unit'. *Discrete Mathematics*, 201(1-3):21–23, 1999.

[25] A. Bohra, L. S. Chandran, and J. K. Raju. Boxicity of series-parallel graphs. *Discrete mathematics*, 306(18):2219–2221, 2006.

[26] J. Boland and C. Lekkerkerker. Representation of a finite graph by a set of intervals on the real line. *Fundamenta Mathematicae*, 51(1):45–64, 1962.

[27] F. Bonomo, G. Durán, L. N. Grippo, and M. D. Safe. Partial characterizations of circular-arc graphs. *Journal of Graph Theory*, 61(4):289–306, 2009.

[28] K. S. Booth and G. 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.

[29] N. Bousquet, D. Gonçalves, G. B. Mertzios, C. Paul, I. Sau, and S. Thomassé. Parameterized domination in circle graphs. In *proceedings of International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, LNCS, pages 308–319. Springer, 2012.

[30] H. Breu. *Algorithmic aspects of constrained unit disk graphs*. PhD thesis, University of British Columbia, 1996.

[31] D. E. Brown, B. M. Flesch, and J. Richard. A characterization of 2-tree probe interval graphs. *Discussiones Mathematicae Graph Theory*, 34(3):509–527, 2014.

[32] D. E. Brown and L. J. Langley. Forbidden subgraph characterization of bipartite unit probe interval graphs. *Australasian Journal of Combinatorics*, 52:19–32, 2012.

[33] D. E. Brown, J. R. Lundgren, and L. Sheng. A characterization of cycle-free unit probe interval graphs. *Discrete Applied Mathematics*, 157(4):762–767, 2009.

[34] J. P. Burling. *On coloring problems of families of prototypes.* P.hd Thesis, Uni-versity of Colorado, Boulder, 1965.

[35] A. Butman, D. Hermelin, M. Lewenstein, and D. Rawitz. Optimization problems in multiple-interval graphs. *ACM Transactions on Algorithms*, 6(2):40:1–40:18, 2010.

[36] S. Cabello and M. Jejčič. Refining the hierarchies of classes of geometric intersection graphs. *The Electronic Journal of Combinatorics*, 24(1):1–33, 2017.

[37] J. Cardinal, S. Felsner, T. Miltzow, C. Tompkins, and B. Vogtenhuber. Intersection graphs of rays and grounded segments. In *proceedings of International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, LNCS, pages 153–166. Springer, 2017.

[38] P. Carmi, M. K. Chiu, M. J. Katz, M. Korman, Y. Okamoto, A. Van Renssen, M. Roeloffzen, T. Shiitada, and S. Smorodinsky. Balanced line separators of unit disk graphs. In *proceedings of International Symposium on Algorithms and Data Structures (WADS)*, LNCS, pages 241–252. Springer, 2017.

[39] P. Carmi, G. K. Das, R. K. Jallu, S. C. Nandy, P. R. Prasad, and Y. Stein. Minimum dominating set problem for unit disks revisited. *International Journal of Computational Geometry & Applications*, 25(03):227–244, 2015.

[40] D. Catanzaro, S. Chaplick, S. Felsner, B. V. Halldórsson, M. M. Halldórsson, T. Hixon, and J. Stacho. Max point-tolerance graphs. *Discrete Applied Mathematics*, 216:84–97, 2017.

[41] E. Čenek and L. Stewart. Maximum independent set and maximum clique algorithms for overlap graphs. *Discrete Applied Mathematics*, 131(1):77–91, 2003.

[42] D. Chakraborty and M. C. Francis. On the stab number of rectangle intersection graphs. *Theory of Computing Systems*, 2019.

[43] P. Chalermsook. Coloring and maximum independent set of rectangles. In *proceedings of Approximation Algorithms for Combinatorial Optimization (APPROX)*, LNCS, pages 123–134. Springer, 2011.

[44] P. Chalermsook and J. Chuzhoy. Maximum independent set of rectangles. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, pages 892–901. SIAM, 2009.

[45] J. Chalopin and D. Gonçalves. Every planar graph is the intersection graph of segments in the plane. In *proceedings of Symposium on Theory of Computing STOC*, pages 631–638. ACM, 2009.

[46] T. M. Chan and S. Har-Peled. Approximation algorithms for maximum independent set of pseudo-disks. *Discrete & Computational Geometry*, 48(2):373–392, 2012.

[47] L. S. Chandran, A. Das, and C. D. Shah. Cubicity, boxicity, and vertex cover. *Discrete Mathematics*, 309(8):2488–2496, 2009.

[48] L. S. Chandran, M. C. Francis, and N. Sivadasan. Boxicity and maximum degree. *Journal of Combinatorial Theory, Series B*, 98(2):443–445, 2008.

[49] L. S. Chandran, M. C. Francis, and S. Suresh. Boxicity of halin graphs. *Discrete Mathematics*, 309(10):3233–3237, 2009.

[50] L. S. Chandran, R. Mathew, and D. Rajendraprasad. Upper bound on cubicity in terms of boxicity for graphs of low chromatic number. *Discrete Mathematics*, 339(2):443 – 446, 2016.

[51] L. S. Chandran, R. Mathew, and N. Sivadasan. Boxicity of line graphs. *Discrete Mathematics*, 311(21):2359–2367, 2011.

[52] L. S. Chandran and N. Sivadasan. Boxicity and treewidth. *Journal of Combinatorial Theory, Series B*, 97(5):733 – 744, 2007.

[53] S. Chaplick, E. Cohen, and J. Stacho. Recognizing some subclasses of vertex intersection graphs of 0-bend paths in a grid. In *proceedings of Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, LNCS, pages 319–330. Springer, 2011.

[54] S. Chaplick, V. Jelínek, J. Kratochvíl, and T. Vyskočil. Bend-bounded path intersection graphs: Sausages, noodles, and waffles on a grill. In *proceedings of International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, LNCS, pages 274–285. Springer, 2012.

[55] M. Chudnovsky, N. Robertson, P. Seymour, and R. Thomas. The strong perfect graph theorem. *Annals of mathematics*, pages 51–229, 2006.

[56] B. N. Clark, C. J. Colbourn, and D. S. Johnson. Unit disk graphs. *Discrete mathematics*, 86(1-3):165–177, 1990.

[57] C. J. Colbourn and L. K. Stewart. Permutation graphs: connected domination and steiner trees. *Discrete Mathematics*, 86(1-3):179–189, 1990.

[58] D. G. Corneil, J. Dusart, M. Habib, and E. Kohler. On the power of graph searching for cocomparability graphs. *SIAM Journal on Discrete Mathematics*, 30(1):569–591, 2016.

[59] D. G. Corneil, S. Olariu, and L. Stewart. The ultimate interval graph recognition algorithm? In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, pages 175–180. SIAM, 1998.

[60] D. G. Corneil, S. Olariu, and L. Stewart. The LBFS structure and recognition of interval graphs. *SIAM Journal on Discrete Mathematics*, 23(4):1905–1953, 2009.

[61] S. Cornelsen, T. Schank, and D. Wagner. Drawing graphs on two and three lines. In *proceedings of International Symposium on Graph Drawing and Network Visualization (GD)*, LNCS, pages 31–41. Springer, 2002.

[62] M. B. Cozzens and F. S. Roberts. Computing the boxicity of a graph by covering its complement by cointerval graphs. *Discrete Applied Mathematics*, 6(3):217–228, 1983.

[63] M. Cygan, F. V. Fomin, Ł. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized algorithms*, volume 3. Springer, 2015.

[64] M. Damian and S. V. Pemmaraju. APX-hardness of domination problems in circle graphs. *Information processing letters*, 97(6):231–237, 2006.

[65] M. Damian-Iordache and S. V. Pemmaraju. A (2+ϵ)-approximation scheme for minimum domination on circle graphs. *Journal of Algorithms*, 42(2):255–276, 2002.

[66] G. K. Das, G. D. da Fonseca, and R. K. Jallu. Efficient independent set approximation in unit disk graphs. *Discrete Applied Mathematics*, (In Press), 2018.

[67] M. De and A. Lahiri. Geometric dominating set and set cover via local search. *arxiv:1605.02499*, 2016.

[68] R. Diestel. *Graph Theory*. Springer, 2006.

[69] I. Dinur and D. Steurer. Analytical approach to parallel repetition. In *proceedings of Symposium on Theory of Computing STOC*, pages 624–633. ACM, 2014.

[70] Z. Dvořák, K. Kawarabayashi, and R. Thomas. Three-coloring triangle-free planar graphs in linear time. *ACM Transactions on Algorithms*, 7(4):41:1–41:14, 2011.

[71] G. Ehrlich, S. Even, and R. E. Tarjan. Intersection graphs of curves in the plane. *Journal of Combinatorial Theory, Series B*, 21(1):8–20, 1976.

[72] J. Ellis and R. Warren. Lower bounds on the pathwidth of some grid-like graphs. *Discrete Applied Mathematics*, 156(5):545–555, 2008.

[73] T. Erlebach and E. J. Van Leeuwen. Domination in geometric intersection graphs. In *proceedings of Latin American Theoretical Informatics (LATIN)*, LNCS, pages 747–758. Springer, 2008.

[74] L. Esperet. Boxicity and topological invariants. *European Journal of Combinatorics*, 51:495–499, 2016.

[75] L. Esperet. Box representations of embedded graphs. *Discrete & Computational Geometry*, 57(3):590–606, 2017.

[76] L. Esperet and G. Joret. Boxicity of graphs on surfaces. *Graphs and Combinatorics*, pages 1–11, 2013.

[77] M. Farber and J. M. Keil. Domination in permutation graphs. *Journal of algorithms*, 6(3):309–321, 1985.

[78] T. Feder, P. Hell, and J. Huang. List homomorphisms and circular arc graphs. *Combinatorica*, 19(4):487–505, 1999.

[79] S. P. Fekete, K. Huang, J. S. B. Mitchell, O. Parekh, and C. A. Phillips. Geometric hitting set for segments of few orientations. *Theory of Computing Systems*, 62(2):268–303, 2018.

[80] J. Fox and J. Pach. Coloring K_k-free intersection graphs of geometric objects in the plane. *European Journal of Combinatorics*, 33(5):853–866, 2012.

[81] M. Francis, P. Hell, and J. Stacho. Forbidden structure characterization of circular-arc graphs and a certifying recognition algorithm. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, pages 1708–1727. SIAM, 2015.

[82] D. Fulkerson and O. Gross. Incidence matrices and interval graphs. *Pacific journal of mathematics*, 15(3):835–855, 1965.

[83] M. R. Garey, D. S. Johnson, G. L. Miller, and C. H. Papadimitriou. The complexity of coloring circular arcs and chords. *SIAM Journal on Algebraic Discrete Methods*, 1(2):216–227, 1980.

[84] M.R. Garey and D.S. Johnson. *Computers and intractability*, volume 29. W.H. Freeman New York, 2002.

[85] D. R. Gaur, T. Ibaraki, and R. Krishnamurti. Constant ratio approximation algorithms for the rectangle stabbing problem and the rectilinear partitioning problem. *Journal of Algorithms*, 43(1):138–152, 2002.

[86] F. Gavril. Algorithms for a maximum clique and a maximum independent set of a circle graph. *Networks*, 3(3):261–273, 1973.

[87] S. Ghosh, M. Podder, and M. K. Sen. Adjacency matrices of probe interval graphs. *Discrete Applied Mathematics*, 158(18):2004–2013, 2010.

[88] M. Gibson and I. A Pirwani. Algorithms for dominating set in disk graphs: breaking the logn barrier. In *proceedings of European Symposium on Algorithms (ESA)*, LIPIcs, pages 243–254. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2010.

[89] P.W. Goldberg, M.C. Golumbic, H. Kaplan, and R. Shamir. Four strikes against physical mapping of DNA. *Journal of Computational Biology*, 2(1):139–152, 1995.

[90] M. C. Golumbic, D. Rotem, and J. Urrutia. Comparability graphs and intersection graphs. *Discrete Mathematics*, 43(1):37–46, 1983.

[91] M.C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Annals of Discrete Mathematics. Elsevier Science, 2004.

[92] M.C. Golumbic, H. Kaplan, and R. Shamir. Graph sandwich problems. *Journal of Algorithms*, 19(3):449–473, 1995.

[93] D. Gonçalves. 3-colorable planar graphs have an intersection segment representation using 3 slopes. In *proceedings of International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, LNCS, pages 351–363. Springer, 2009.

[94] S. Govindarajan, R. Raman, S. Ray, and A. B. Roy. Packing and covering with non-piercing regions. *Discrete & Computational Geometry*, 60(2):471–492, 2018.

[95] M. Habib, R. McConnell, C. Paul, and L. Viennot. Lex-BFS and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theoretical Computer Science*, 234(1-2):59–84, 2000.

[96] G. Hajós. über eine art von graphen. *Intern. Math. Nachr.*, page Problem 65, 1957.

[97] I. B. Hartman, I. Newman, and R. Ziv. On grid intersection graphs. *Discrete Mathematics*, 87(1):41–52, 1991.

[98] T. W. Haynes, S. Hedetniemi, and P. Slater. *Domination in Graphs: Volume 2: Advanced Topics*. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 1998.

[99] T. W. Haynes, S. Hedetniemi, and P. Slater. *Fundamentals of Domination in Graphs*. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 1998.

[100] S. T. Hedetniemi and R. C. Laskar. *Topics on Domination*. Elsevier Science, 1991.

[101] W. Hsu. A simple test for interval graphs. In *proceedings of International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, LNCS, pages 11–16. Springer, Springer, 1992.

[102] H. Imai and T. Asano. Finding the connected components and a maximum clique of an intersection graph of rectangles in the plane. *Journal of Algorithms*, 4(4):310–323, 1983.

[103] R. J. Kang and T. Müller. Sphere and dot product representations of graphs. *Discrete & Computational Geometry*, 47(3):548–568, 2012.

[104] H. Kaplan, W. Mulzer, L. Roditty, and P. Seiferth. Routing in unit disk graphs. *Algorithmica*, 80(3):830–848, 2018.

[105] R. M. Karp. Mapping the genome: some combinatorial problems arising in molecular biology. In *proceedings of Symposium on Theory of Computing STOC*, pages 278–285. ACM, 1993.

[106] T. Karthick and F. Maffray. Coloring (gem, co-gem)-free graphs. *Journal of Graph Theory*, 89(3):288–303, 2018.

[107] M. J. Katz, J. S. B. Mitchell, and Y. Nir. Orthogonal segment stabbing. *Computational Geometry: Theory and Applications*, 30(2):197–205, 2005.

[108] J. M. Keil, J. S. B Mitchell, D. Pradhan, and M. Vatshelle. An algorithm for the maximum weight independent set problem on outerstring graphs. *Computational Geometry*, 60:19–25, 2017.

[109] S. Khot and O. Regev. Vertex cover might be hard to approximate to within 2-ϵ. *Journal of Computer and System Sciences*, 74(3):335–349, 2008.

[110] S. Khot and N.K. Vishnoi. On the unique games conjecture. In *proceedings of IEEE Annual Symposium on Foundations of Computer Science, FOCS*, page 3. IEEE Computer Society, 2005.

[111] V. Klee. What are the intersection graphs of arcs in a circle? *The American Mathematical Monthly*, 76(7):810–813, 1969.

[112] J. Kleinberg and E. Tardos. *Algorithm design*. Pearson Education India, 2006.

[113] N. Korte and R. H. Möhring. An incremental linear-time algorithm for recognizing interval graphs. *SIAM Journal on Computing*, 18(1):68–81, 1989.

[114] A. Kostochka. Coloring intersection graphs of geometric figures with a given clique number. *Contemporary mathematics*, 342:127–138, 2004.

[115] J. Kratochvíl. A special planar satisfiability problem and a consequence of its NP-completeness. *Discrete Applied Mathematics*, 52(3):233–252, 1994.

[116] D. Kratsch, R. M. McConnell, K. Mehlhorn, and J. P. Spinrad. Certifying algorithms for recognizing interval graphs and permutation graphs. *SIAM Journal on Computing*, 36(2):326–353, 2006.

[117] T. Krawczyk, A. Pawlik, and B. Walczak. Coloring triangle-free rectangle overlap graphs with $O(\log \log n)$ colors. *Discrete & Computational Geometry*, 53(1):199–220, 2015.

[118] L.C. Lau, R. Ravi, and M. Singh. *Iterative methods in combinatorial optimization*, volume 46. Cambridge University Press, 2011.

[119] L. Lewin-Eytan, J. S. Naor, and A. Orda. Routing and admission control in networks with advance reservations. In *proceedings of Approximation Algorithms for Combinatorial Optimization (APPROX)*, LNCS, pages 215–228. Springer, 2002.

[120] M. C. Lin and J. L. Szwarcfiter. Characterizations and linear time recognition of helly circular-arc graphs. In *proceedings of Computing and Combinatorics Conference COCOON*, LNCS, pages 73–82. Springer, 2006.

[121] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25(2):59–68, 1995.

[122] R. M. McConnell, K. Mehlhorn, S. Näher, and P. Schweitzer. Certifying algorithms. *Computer Science Review*, 5(2):119–161, 2011.

[123] C. McDiarmid and T. Müller. Integer realizations of disk and segment graphs. *Journal of Combinatorial Theory, Series B*, 103(1):114–143, 2013.

[124] C. McDiarmid and T. Müller. The number of disk graphs. *European Journal of Combinatorics*, 35:413–431, 2014.

[125] S. McGuinness. On bounding the chromatic number of L-graphs. *Discrete Mathematics*, 154(1-3):179–187, 1996.

[126] F. R. McMorris, C. Wang, and P. Zhang. On probe interval graphs. *Discrete Applied Mathematics*, 88(1-3):315–324, 1998.

[127] S. Mehrabi. Approximating domination on intersection graphs of paths on a grid. In *proceedings of Workshop on Approximation and Online Algorithms (WAOA)*, LNCS, pages 76–89. Springer, 2017.

[128] N. H. Mustafa and S. Ray. PTAS for geometric hitting set problems via local search. In *proceedings of Symposium on Computational Geometry SoCG*, LIPIcs, pages 17–22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2009.

[129] T. Nieberg and J. Hurink. A PTAS for the minimum dominating set problem in unit disk graphs. In *proceedings of Workshop on Approximation and Online Algorithms (WAOA)*, LNCS, pages 296–306. Springer, 2005.

[130] O. Ore. *Theory of graphs*. American Mathematical Society, 1962.

[131] J. Pach and G. Tóth. How many ways can one draw a graph? *Combinatorica*, 26(5):559–576, 2006.

[132] S. Pandit. Dominating set of rectangles intersecting a straight line. In *proceedings of Canadian Conference on Computational Geometry*, pages 144–149, 2017.

[133] I. Pe'er and R. Shamir. Interval graphs with side (and size) constraints. In *proceedings of European Symposium on Algorithms (ESA)*, LIPIcs, pages 142–154. Springer, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 1995.

[134] I. G. Perepelitsa. Bounds on the chromatic number of intersection graphs of sets in the plane. *Discrete Mathematics*, 262(1-3):221–227, 2003.

[135] N. Pržulj and D. G. Corneil. 2-tree probe interval graphs have a large obstruction set. *Discrete Applied Mathematics*, 150(1-3):216–231, 2005.

[136] R. Rado. Covering theorems for ordered sets. *Proceedings of the London Mathematical Society*, 2(1):509–535, 1948.

[137] C. S. Rim and K. Nakajima. On rectangle intersection and overlap graphs. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 42(9):549–553, 1995.

[138] F. S. Roberts. *On the boxicity and cubicity of a graph.* Academic Press, 1969.

[139] F. S. Roberts. *Graph theory and its applications to problems of society*, volume 29. SIAM, 1978.

[140] E. R. Scheinerman. *Intersection Classes and Multiple Intersection Parameters of Graphs.* P.hd Thesis, Princeton University, 1984.

[141] A. Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998.

[142] A. Scott and P. Seymour. A survey of χ-boundedness. *arXiv:1812.07500*, 2018.

[143] A. Scott and D. R. Wood. Better bounds for poset dimension and boxicity. *arXiv:1804.03271*, 2018.

[144] J. Snoeyink. Maximum independent set for intervals by divide and conquer with pruning. *Networks: An International Journal*, 49(2):158–159, 2007.

[145] J. P. Spinrad. *Efficient graph representations*. American Mathematical Society, 2003.

[146] M. Suderman. Pathwidth and layered drawings of trees. *International Journal of Computational Geometry & Applications*, 14(03):203–225, 2004.

[147] E. Tardos. A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research*, 34(2):250–256, 1986.

[148] W. T. Trotter Jr and J. I. Moore Jr. Characterization problems for graphs, partially ordered sets, lattices, and families of sets. *Discrete Mathematics*, 16(4):361–381, 1976.

[149] A. Tucker. Characterizing circular-arc graphs. *Bulletin of the American Mathematical Society*, 76(6):1257–1260, 1970.

[150] D. West. Open problems. *SIAM Journal of Discrete Mathematics Newsletter*, 2(1):10–12, 1991.

[151] D. B. West. *Introduction to Graph Theory*. Prentice Hall, 2000.

[152] D. P. Williamson and D. B. Shmoys. *The design of approximation algorithms*. Cambridge university press, 2011.

[153] M. Yannakakis. The complexity of the partial order dimension problem. *SIAM Journal on Algebraic Discrete Methods*, 3(3):351–358, 1982.

[154] P. Zhang. Probe interval graph and its applications to physical mapping of DNA. *Manuscript*, 1994.

[155] P. Zhang, E. A. Schon, S. G. Fischer, E. Cayanis, J. Weiss, S. Kistler, and P. E. Bourne. An algorithm based on graph theory for the assembly of contigs in physical mapping of DNA. *Bioinformatics*, 10(3):309–317, 1994.

[156] P. Zhang, X. Ye, L. Liao, J. J. Russo, and S. G. Fischer. Integrated mapping package—a physical mapping software tool kit. *Genomics*, 55(1):78–87, 1999.