

On Interval Routing Schemes and Treewidth^{*}

Hans L. Bodlaender¹, Jan van Leeuwen¹, Richard Tan^{1,2}, and Dimitrios M. Thilikos¹

¹ Department of Computer Science, Utrecht University,
P.O. Box 80.089
3508 TB Utrecht, the Netherlands
{hansb,jan,rbtan,sedthilk}@cs.ruu.nl

² Department of Computer Science, University of Sciences and Arts of Oklahoma
Chickasha, Oklahoma 73018, USA

Abstract. In this paper, we investigate which processor networks allow k -label Interval Routing Schemes, under the assumption that costs of edges may vary. We show that for each fixed $k \geq 1$, the class of graphs allowing such routing schemes is closed under minor-taking in the domain of connected graphs, and hence has a linear time recognition algorithm. This result connects the theory of compact routing with the theory of graph minors and treewidth.

We show that every graph that does not contain $K_{2,r}$ as a minor has treewidth at most $2r - 2$. In case the graph is planar, this bound can be lowered to $r + 2$. As a consequence, graphs that allow k -label Interval Routing Schemes under dynamic cost edges have treewidth at most $4k$, and treewidth at most $2k + 3$ if they are planar.

Similar results are shown for other types of Interval Routing Schemes.

1 Introduction

A common problem in processor networks is that messages that are sent from one processor to another processor must be routed through the network. The classical solution is to give each processor a routing table, with an entry for each possible destination specifying over which link the message must be forwarded. A disadvantage of this method is that these tables grow with network size, and may become too large for larger processor networks.

^{*}This research was partially supported by the ESPRIT Basic Actions Program of the EC under contract No. 8141 (project ALCOM II). The research of the second author was also partially supported by the Netherlands Organisation for Scientific Research (NWO) under contract NF 62-376 (NFI project ALADDIN: *Algorithmic Aspects of Parallel and Distributed Systems*). The research of the last author was supported by the Training and Mobility of Researchers (TMR) Program, (EU contract no ERBFMBICT950198). Correspondence on this paper to the first author.

Several different routing methods have been proposed that do not have this disadvantage. One such method is the interval routing method, together with its generalisation k -label interval routing and variants of these. An overview of these and other compact routing methods can be found in [19].

Interval routing was introduced by Santoro and Khatib [24] and van Leeuwen and Tan [18]. Several well-known classes of networks allow interval routing schemes that are optimal, in the sense that messages always follow the shortest path to their destination. The method was applied in the C104 Router Chip, used in the INMOS T9000 Transputer design [16].

Frederickson and Janardan [15] considered interval routing in the setting of dynamic cost links (i.e., in the case that the cost of edges is variable). Actually, they considered a variant of interval routing, called *strict* interval routing. For these, they gave a precise characterisation of the graphs with dynamic cost links which allow optimum strict interval routing schemes: these are exactly the *outerplanar* graphs. Bakker, van Leeuwen and Tan [2] obtained a similar result for general interval routing: a graph with dynamic cost links has an optimum interval routing scheme, if and only if it is outerplanar or K_4 . Another restriction of interval routing was introduced by Bakker, van Leeuwen, and Tan in [3]: linear interval routing. It has also been applied in concrete networks. Here, also a precise characterisation exists of the graphs which allow optimum linear interval routing schemes with dynamic cost links.

All of the interval routing schemes assumes that each link has one unique label, which is a (possibly cyclic) interval of processor names. All can be generalised to multi-label schemes, where each link has a number of labels. We consider the k -label schemes: each link has at most k labels. The issue we study in this paper is: *which graphs allow k -label interval routing schemes in the setting of dynamic cost links.*

Surprisingly, new and deep graph theoretical results on graph minors of Robertson and Seymour (see Section 2.1) can be used for the analysis of this problem. With the help of these results, we show *non-constructively* the existence of finite characterisations of which graphs allow certain routing schemes. Also, we give a non-constructive proof of the *existence* of linear time algorithms that check whether a desired routing scheme exists for a given graph. These algorithms heavily depend on the use of tree-decompositions. We show that graphs, allowing a k -label interval routing scheme (in the setting of dynamic cost links) have treewidth at most $4k$. This not only gives a partial characterisation of the graphs which have such routing schemes, but also, as the hidden constant factor of these algorithms is exponential in the treewidth of the tree-decomposition, it helps to decrease the running time of algorithms that would test the property.

As a main lemma, we show that every graph either contains $K_{2,r}$ as a minor, or has treewidth at most $2r - 2$. This can be seen as a special case of a result of Robertson and Seymour [21]: every planar graph $H = (V(H), E(H))$ has an associated constant c_H , such that any graph G either contains H as a minor or has

treewidth at most c_H . The best general bound for c_H known is $20^{2(2|V(H)|+4|E(H)|)^5}$ [23]. Our result gives a much better bound in the case of graphs of the form $K_{2,r}$. Also, this result is constructive, and can be turned into an $O(rn)$ time algorithm, that either outputs that the input graph G has $K_{2,r}$ as a minor, or that outputs a tree-decomposition of G of treewidth at most $2r - 2$. Similar results for other specific graphs can be found in [4] (trees), [14] (cycles and subgraphs of cycles), [7] (disjoint copies of K_3), and [6] (graphs that are minor of a circus graph and $(2 \times k)$ -grid). The result of this main lemma can be seen as an additional result, fitting into this framework. Applied to the routing problem, it gives the first graph-theoretic complexity bound on the graphs that admit optimal k -label interval routing schemes. Another consequence we discuss is that ‘most’ random graphs (even ‘sparse random graphs’) do not allow k -label interval routing schemes under the dynamic cost edges assumption, for small values of k . Additionally, we give variants of the results when the graphs are restricted to be planar.

This paper is organised as follows. In Section 2, we give most necessary definitions and some preliminary results. In Section 3, we establish minor-closedness of the considered classes of graphs, each class containing those networks allowing certain types of k -label interval routing schemes. As a consequence, we obtain a non-existential proof of the existence of linear time membership algorithms for these classes. Also, slower, but constructive algorithms for these problems are given. In Section 4, we give the result on the treewidth of graphs, avoiding $K_{2,r}$ as a minor (as discussed above). In Section 5, a similar result is given, but with a restriction to planar graphs, and with a better bound. Some open problems are mentioned in Section 6.

2 Definitions and preliminary results

In this section, we introduce the most important definitions and mention some known results. In Section 2.1, we introduce graph-theoretic notions and results, and in Section 2.2, concepts and results from interval routing and its variants.

2.1 Graph theoretic definitions and preliminary results

All graphs in this paper will be assumed to be undirected, simple and finite. Given a graph G we denote as $V(G)$ and $E(G)$ the set of its vertices and edges respectively. The number of vertices of a graph $G = (V, E)$ will be denoted by $n = |V(G)|$. The notion of treewidth was introduced by Robertson and Seymour [21].

Definition. A tree-decomposition of a graph $G = (V, E)$ is a pair $D = (X, T)$ with $T = (I, F)$ a tree and $X = \{X_i \mid i \in I\}$ a family of subsets of V , one for each node of T , such that

- $\bigcup_{i \in I} X_i = V$.
- for all edges $\{v, w\} \in E$, there exists an $i \in I$ with $v \in X_i$ and $w \in X_i$.
- for all $i, j, k \in I$: if j is on the path from i to k in T , then $X_i \cap X_k \subseteq X_j$.

The treewidth of a tree-decomposition $(\{X_i \mid i \in I\}, T = (I, F))$ is $\max_{i \in I} |X_i| - 1$. The treewidth of a graph G is the minimum treewidth over all possible tree-decompositions of G .

There are several well known equivalent characterisations of the notion of treewidth; for instance, a graph has treewidth at most k , if and only if it is a partial k -tree, or a subgraph of a chordal graph with maximum clique size at most $k + 1$ (see [17]).

A graph $G = (V, E)$ is said to be a *minor* of a graph $H = (W, F)$, if G can be obtained from H by a series of vertex deletions, edge deletions, and edge contractions; where an edge contraction is the operation that takes two adjacent vertices v and w , and replaces it by a new vertex, adjacent to all vertices that were adjacent to v or w . A class of graphs \mathcal{G} is said to be *closed under taking of minors*, if for every $G \in \mathcal{G}$, every minor H of G belongs to \mathcal{G} . For classes of graphs \mathcal{G}, \mathcal{H} , we say that \mathcal{G} is *closed under taking of minors in the domain \mathcal{H}* , if for every graph $G \in \mathcal{G} \cap \mathcal{H}$, every minor H of G with $H \in \mathcal{H}$ belongs to \mathcal{G} .

In a long series of papers, Robertson and Seymour proved their famous graph minor theorem (formerly ‘Wagner’s conjecture’):

Theorem 1 (See [20].) *For every class of graphs \mathcal{G} , that is closed under taking of minors, there exists a finite set of graphs, called the obstruction set of \mathcal{G} , $ob(\mathcal{G})$, such that for all graphs H , $H \in \mathcal{G}$, if and only if there is no graph G in the obstruction set of \mathcal{G} that is a minor of H .*

Fellows and Langston [13] derived the following consequence and variant of this result.

Theorem 2 *Let \mathcal{G} be a class of graphs, closed under taking of minors in the domain \mathcal{H} , with $\mathcal{G} \subseteq \mathcal{H}$. There exists a finite set of graphs, the obstruction set of \mathcal{G} in \mathcal{H} , $ob_{\mathcal{H}}(\mathcal{G})$, such that for all graphs $H \in \mathcal{H}$, $H \in \mathcal{G}$, if and only if there is no graph $G \in ob_{\mathcal{H}}(\mathcal{G})$ that is a minor of H .*

It should be noted that the proofs of these results are (inherently) non-constructive. As for every fixed graph H , there exists an $O(n^3)$ time algorithm that tests whether H is a minor of a given graph G with n vertices [22], it follows that every minor-closed class of graphs has a cubic recognition algorithm, and every minor-closed class of graphs in a domain \mathcal{H} has a cubic algorithm that tests whether graphs from \mathcal{H} belong to \mathcal{G} . However, as the proof of Theorem 1 is non-constructive, we only know the algorithm exists, but we do not have the algorithm itself.

In several cases, faster algorithms exist.

Theorem 3 ([21]) *For every planar graph H , there exists a constant c_H , such that for every graph G , either H is a minor of G , or the treewidth of G is at most c_H .*

Moreover, for every fixed integer k and graph H , there exists a linear time algorithm, such that when given a graph $G = (V, E)$ with a tree-decomposition of treewidth at most k , decides whether H is a minor of G , using standard methods for graphs with bounded treewidth (see e.g. [1].) As such tree-decompositions can be found in linear time [5], when existing, the following result holds:

Theorem 4 *Let \mathcal{G} be a class of graphs that is closed under taking of minors, and that does not contain all planar graphs. Then there exists a linear time algorithm that tests whether a given graph G belongs to \mathcal{G} .*

Proof: (This proof is basically taken from [13], but we now use the algorithm of [5] for finding tree-decomposition of small treewidth.) Suppose G is a planar graph that does not belong to \mathcal{G} . First test whether the treewidth of input graph G is at most c_H . If not, we can safely conclude that $G \notin \mathcal{G}$. Otherwise, find a tree-decomposition of G of treewidth at most c_H with the algorithm of [5], and use this tree-decomposition to test whether a graph in $ob(\mathcal{G})$ is a minor of G . \square

Theorem 5 *Let \mathcal{G} be a class of graphs that is closed under taking of minors in the domain \mathcal{H} , $\mathcal{G} \subseteq \mathcal{H}$. Suppose there is at least one planar graph that belongs to \mathcal{H} but not to \mathcal{G} . Then there exists a linear time algorithm that tests whether a given graph $G \in \mathcal{H}$ belongs to \mathcal{G} .*

Proof: (We again use an only slightly modified variant of a proof from [13].) Suppose H is a planar graph with $H \in \mathcal{H}$, $H \notin \mathcal{G}$. If $G \in \mathcal{G}$, then G does not contain H as a minor, hence has treewidth at most c_H . So, again we can first test whether the treewidth of G is at most k . If not, we are done. Otherwise, we compute a tree-decomposition of G with treewidth at most c_H , and then use this tree-decomposition to test in linear time whether G contains a graph in $ob_{\mathcal{H}}(\mathcal{G})$ as a minor. \square

The constant factor of the linear time algorithms mentioned above is exponential in the treewidth of the tree-decomposition used, i.e., in c_H , H a planar graph not in \mathcal{G} (but in \mathcal{H}). The constant factor in the original result of Robertson and Seymour was ‘astronomically large’. In a later paper, Robertson et al. [23] improved this result, and obtained a constant factor of $20^{2(2|V(H)|+4|E(H)|)^5}$.

Still, in most, if not all, practical cases, this constant factor is much too large, and makes the algorithm practically infeasible. This is the motivation, why we looked for much smaller values of c_H for graphs of the form $K_{2,r}$, as these graphs are planar, connected and can be shown to be ‘outside’ the considered classes of graphs.

2.2 Definitions and preliminary results on interval routing

Unless stated otherwise, intervals will be assumed to be ‘cyclic’ in the set $\{0, 1, \dots, n-1\}$, ($n = |V|$); thus if $a > b$ then the interval $[a, b)$ denotes the set $\{a, a+1, \dots, n-1, 0, \dots, b-1\}$.

The shortest distance from vertex $u \in V$ to a vertex $v \in V$ in a graph $G = (V, E)$ when edges have costs given by edge cost function $c : E \rightarrow \mathbf{R}$, is denoted by $d_{G,c}(u, v)$. When G and/or c are clear from the context, we drop them from the subscript. The cost of a path p under edge cost function c is denoted by $c(p)$.

A node labelling of a graph $G = (V, E)$ is a bijective mapping $nb : V \rightarrow \{0, 1, \dots, n-1\}$. An interval labelling scheme (ILS) of a graph $G = (V, E)$ is a node labelling nb of G , together with a labelling l , mapping each link to an interval $[a, b)$, $a, b \in \{0, 1, \dots, n-1\}$, such that for every vertex v , the set of all labels of links outgoing from v partitions the set $\{0, 1, \dots, n-1\}$.

Given an ILS, routing is done as follows. Each message contains, amongst others, the node label $nb(w)$ of its destination node w . When a node x receives a message with destination-label $dest$, it first looks whether $nb(x) = dest$. If so, the message has reached its destination, and is not routed any further. Otherwise, the message is transferred over the link with label $[a, b)$ such that $dest \in [a, b)$. An ILS is *valid*, if for all nodes v, w , messages sent from v to w eventually reach w by this procedure. An *interval routing scheme* (in short: IRS) is a valid ILS.

The notion of strict interval labelling schemes is obtained in a similar way: modify the definition of ILS in the sense that all labels of links associated with nodes v must partition the set $\{0, 1, \dots, n-1\} - \{nb(v)\}$, i.e., the label of v may not appear in the labels of any of its outgoing links. A *linear interval labelling scheme* is an ILS where no interval label ‘wraps’ around, i.e., for all interval labels $[a, b)$ $a < b$. Strict linear interval labelling schemes, strict interval routing schemes (SIRS), linear interval routing schemes (LIRS), and strict linear interval routing schemes (SLIRS) are defined in the obvious way.

For each of these notions, we also define k -label variants. Here, each link is labelled with at most k (cyclic) intervals. All (cyclic) intervals associated with links of a node v must together partition $\{0, 1, \dots, n-1\}$ (or $\{0, 1, \dots, n-1\} - \{nb(v)\}$, in the case of strict labellings.) Again, a message is transferred over the link e for which one of its labels is an interval that contains the destination-number. k -label interval routing schemes, k -label linear interval routing schemes, etc., are defined as can be expected, and abbreviated as k -IRS, k -LIRS, etc. Note that an IRS is an 1-IRS, etc.

A routing scheme is *optimal* for a graph $G = (V, E)$, together with an assignment of non-negative costs to each edge $e \in E$, if, whenever a message is sent from node v to node w , the path taken by this message is a minimum cost path from v to w .

Costs of edges denote the time needed to send a message over the edge. How-

ever, in many practical cases, this time may vary. This situation is modelled by the dynamic cost links setting.

We say that graph $G = (V, E)$ with dynamic cost links has an optimum k -IRS, if there exists a node labelling nb of G , such that for all assignments of non-negative costs to edges of E , there exists an IRS (nb, l) that is optimal for this cost assignment.

The class of graphs k -IRS is defined as the set of all graphs G that have an optimum k -IRS with dynamic cost links. In the same way, we define classes k -LIRS, k -SIRS, k -SLIRS. See [19] for an overview of several results on these classes. We have the following relationships.

- Theorem 6** (i) (Frederickson, Janardan [15]) k -IRS \subset $(k + 1)$ -IRS.
(ii) (Bakker, van Leeuwen, Tan [3]) k -LIRS \subset $(k + 1)$ -LIRS.
(iii) (Bakker, van Leeuwen, Tan [3]) k -IRS \subset $(k + 1)$ -LIRS.
(iv) k -SIRS \subseteq k -IRS \subseteq $(k + 1)$ -SIRS.
(v) k -SLIRS \subseteq k -LIRS \subseteq k -IRS.

3 Closedness under minor taking

In this section we prove that for each fixed integer $k \geq 1$, each of the classes k -IRS, k -LIRS, k -SIRS and k -SLIRS is closed under taking of minors in the domain of connected graphs. The reason that this result is interesting is that it enables us to apply results from the theory of graph minors and of graphs of bounded treewidth to the theory of interval routing. We first prove a lemma which will be used later.

Lemma 7 Let $G = (V, E)$ be a graph with edge costs $c : E \rightarrow \mathbf{R}^+ \cup \{0\}$. There exists an edge cost function $c' : E \rightarrow \mathbf{Z}^+$, such that for all $u, v \in V$: each shortest path p from u to v in G under edge costs c' is also a shortest path from u to v in G under edge costs c .

Proof: Let \mathcal{P} be the set of all simple paths in G . Define

$$\epsilon = \min\{|c(p) - c(p')| \mid p, p' \in \mathcal{P}, c(p) \neq c(p')\}$$

Note that $\epsilon > 0$. Define $c' : E \rightarrow \mathbf{Z}^+$ by taking for all $e \in E$:

$$c'(e) = \left\lfloor \frac{|V|c(e)}{\epsilon} \right\rfloor + 1$$

Suppose p is a shortest path from u to v under edge costs c' , but not under edge costs c . Let p' be another path from u to v with $c(p') < c(p)$. By definition of ϵ ,

we have $c(p') \leq c(p) - \epsilon$. Let $n = |V|$. Now

$$\begin{aligned}
c'(p') &= \sum_{e \in p'} \left(\left\lfloor \frac{n \cdot c(e)}{\epsilon} \right\rfloor + 1 \right) \\
&\leq n - 1 + \frac{n}{\epsilon} c(p') < n + \frac{n}{\epsilon} (c(p) - \epsilon) \\
&= \frac{n}{\epsilon} c(p) = \sum_{e \in p} \frac{n \cdot c(e)}{\epsilon} \\
&\leq \sum_{e \in p} \left(\left\lfloor \frac{n \cdot c(e)}{\epsilon} \right\rfloor + 1 \right) = c'(p)
\end{aligned}$$

So, $c'(p') < c'(p)$, hence p was not a shortest path from u to v under edge cost c' , contradiction. \square

Theorem 8 *Let $k \in \mathbf{N}$ be a fixed constant.*

- (i) *k -IRS is closed under minor taking in the domain of connected graphs.*
- (ii) *k -LIRS is closed under minor taking in the domain of connected graphs.*
- (iii) *k -SIRS is closed under minor taking in the domain of connected graphs.*
- (iv) *k -SLIRS is closed under minor taking in the domain of connected graphs.*

Proof: (i) It is sufficient to prove, that if a connected graph $G = (V', E')$ is obtained from a graph $H = (V, E) \in k$ -IRS by one of the following operations: removal of a vertex, removal of an edge, contraction of an edge, then $G \in k$ -IRS. Suppose $H \in k$ -IRS; let nb be a vertex labelling, such that for any cost assignment, there exists a k -label interval routing scheme (nb, l) for H .

First, suppose that G is obtained from H by removing an edge e_0 . Use the same numbering nb for G . For any cost assignment $c : E_G \rightarrow \mathbf{R}^+ \cup \{0\}$, consider the cost assignment $c' : E_H \rightarrow \mathbf{R}^+ \cup \{0\}$, where for all $e \in E_G : c'(e) = c(e)$, and take $c(e_0) = 1 + \sum_{e \in E_G} c(e)$, i.e., the cost of e_0 is chosen so large that no minimum cost path will ever use the edge e_0 . Hence, any k -label interval routing scheme (nb, l) for H with costs c' will also be a k -label interval routing scheme for G with costs c .

Next, suppose that G is obtained from H by removing a vertex $v \in V$ and all of its adjacent edges. By first removing all edges adjacent to v but one, as in the previous case, it follows that we may assume v has degree 1. Now, no shortest path between two vertices w and x , $x \neq v$, $x \neq w$ uses v . Label the vertices in V' as follows: if $nb(w) < nb(v)$, then take $nb'(w) = nb(w)$, and if $nb(w) > nb(v)$, then $nb'(w) = nb(w) - 1$. For any edge cost function c on G , we can make an IRS as follows: consider the same edge cost function c on H , giving the unique edge from v some arbitrary cost, and find an IRS (nb, l) for this function on H . Applying the same relabelling (decrease all labels larger than $nb(v)$ by one) on labelling l , we obtain a labelling l' such that (nb', l') is an IRS for G with edge costs c .

Finally, suppose G is obtained from H by contracting the edge $(v, w) = e_0 \in E_H$ to a vertex, say v' . Let $nb' : V \rightarrow \{0, 1, \dots, |V(H)| - 1\}$ be the function, obtained by taking for all $x \in V(G) - \{v\}$, $nb'(x) = nb(x)$. Actually, there is a ‘gap’ in nb' : there is no vertex x with number $nb'(x) = nb(v)$. This is resolved by decreasing all labels larger than $nb(v)$ by one, as in the case of removing a vertex.

Let $c : E_G \rightarrow \mathbf{R}^+ \cup \{0\}$ be a cost assignment for G . By Lemma 7, there exists a cost assignment $c' : E_G \rightarrow \mathbf{N}^+$, such that all shortest paths under cost assignment c' are shortest paths under cost assignment c . Let $\alpha = 1 + \sum_{e \in E_G} c'(e)$, a forbidding weight. Now let $c'' : E_H \rightarrow \mathbf{R}^+$ be defined as follows: for all edges $\{x, y\} \in E_H$ with $x, y \notin \{v, w\}$, let $c''(\{x, y\}) = c'(\{x, y\})$. For $y \neq w$, if $\{v, y\} \in E(H)$, take $c''(\{v, y\}) = c'(\{v', y\})$. For $y \neq v$, if $\{w, y\} \in E_H$, then if $\{v, y\} \in E_H$, then let $c''(\{w, y\}) = \alpha$, otherwise let $c''(\{w, y\}) = c'(\{v', y\}) + 1/4$. Finally, we let $c''(\{v, w\}) = 1/8$.

Let (nb, l) be a k -IRS for H with cost c'' . We can use l to build a k -IRS (nb, l') for G with cost c' . First note that H without the edges of cost α is still connected. So, no shortest path takes an edge of cost α , and all links corresponding to these edges have an empty label. For every link $(x, \{x, y\})$ with $x \notin \{v, w\}$, take in l' the same labels as in l . For a link $(v', \{v', y\})$, take in l' the union of the labels of links $(v, \{v, y\})$ and $(w, \{w, y\})$. Note that one of these links is either non-existing or empty, so this label will not consist of more than k intervals. Also, note that for every node x , the shortest path from v to x does not use w , if and only if the shortest path from w to x uses v . The same holds with roles of v and w reversed. It follows that no vertex label will appear in more than one label of a link outgoing from v' . We now have shown that l' is a k -ILS.

It remains to be shown that l' gives shortest paths in G . Consider nodes x and y in $V(G)$. Let p be a shortest path in H between nodes x and y following links as directed by l' . If $x = v'$, then take $x = v$ in H . Similar, if $y = v'$. Note that if both v and w appear in p , then they must occur as consecutive nodes on this path, as all edges except $\{v, w\}$ have cost at least 1. Let p' be the path in G , obtained from p by replacing a possible occurrence of v , w or both by one occurrence of v' . Observe that l' will direct a message from x to y via path p' . Finally, observe that p' is a shortest path from x to y in G with costs c' , hence also with costs c .

(ii) (iii) (iv) Similar. □

Theorem 9 (i) (Frederickson, Janardan [15]) $K_{2,2k+1} \notin k$ -SIRS.

(ii) $K_{2,2k+1} \notin k$ -IRS.

Proof: (i) is shown in [15]. The proof of (ii) is very similar. □

It follows now from Theorem 2 that for each fixed $k \geq 1$, the classes k -IRS, k -LIRS, k -SIRS, and k -SLIRS have a finite characterisation in terms of obstruction sets. Combining Theorem 9, Theorem 8 and Theorem 5 gives the following result.

Corollary 10 *For each fixed $k \in \mathbf{N}$, there exists a linear time algorithm that decide whether given a graph $G = (V, E)$ belongs to the class k -IRS (or: k -SIRS, k -LIRS, k -SLIRS).*

It should be noted that this result is *non-constructive*: we know the algorithm exists, but to write down the algorithm, we must know the corresponding finite obstruction set, which we do not know. Unfortunately, we only know of much slower constructive versions of these results. For establishing these constructive version, we first need the following lemma.

Lemma 11 *Let $G = (V, E)$ be a connected graph, and let nb be a node labelling of G . The following statements are equivalent:*

1. *For every cost assignment $c : E \rightarrow \mathbf{R}^+ \cup \{0\}$, there exists an optimal k -SIRS.*
2. *For every vertex v , and for every edge $\{v, w\} \in E$, there does not exist vertices $a_1, \dots, a_{k+1}, b_1, \dots, b_{k+1} \in V$, and a spanning tree $T = (V, F)$ of G such that*
 - $nb(a_1) < nb(b_1) < nb(a_2) < nb(b_2) < \dots < nb(a_k) < nb(b_k) < nb(a_{k+1}) < nb(b_{k+1})$ or $nb(b_1) < nb(a_1) < nb(b_2) < nb(a_2) < \dots < nb(b_k) < nb(a_k) < nb(b_{k+1}) < nb(a_{k+1})$.
 - For each i , $1 \leq i \leq k + 1$, the path in T from v to a_i uses the edge $\{v, w\}$.
 - For each i , $1 \leq i \leq k + 1$, the path in T from v to b_i does not use the edge $\{v, w\}$.
 - $nb(w) \in \{a_1, \dots, a_{k+1}\}$.

Proof: $2 \rightarrow 1$: Suppose that $v, \{v, w\}, a_1, \dots, a_{k+1}, b_1, \dots, b_{k+1}$ and T are as stated. Now, let c be the cost assignment that assigns cost 1 to every edge in T , and $|V| + 1$ to every other edge, i.e., all shortest paths follow T . Now, each $nb(a_i)$ must be in a different interval for the link $(v, \{v, w\})$, as when $nb(a_i)$ and $nb(a_{i+1})$ would be in the same interval, then $nb(b_i)$ or $nb(b_{i+1})$ also would belong to the interval, and messages to this node b_i or b_{i+1} would be routed in the wrong direction.

$1 \rightarrow 2$: Suppose for cost assignment c , there is no optimal k -IRS. Note that we may assume that between every two pairs of nodes, there is a unique shortest path. (If not, then we can change the weights of some edges with very small amounts, such that there some non-unique shortest paths disappear, but no new shortest path routes are created.) Now, there is a vertex $v \in V$, and an adjacent edge $\{v, w\} \in E$, such that at least $k+1$ intervals, say $[c_1, d_1], \dots, [c_r, d_r]$, $r \geq k+1$ are necessary to give the set of numbers of nodes whose shortest paths from v use the edge $\{v, w\}$. For each interval $[c_i, d_i]$, $1 \leq i \leq k$, choose a vertex a_i with

$nb(a_i) \in [c_i, d_i]$, and choose a vertex a_{k+1} with $nb(a_{k+1}) \in [c_{k+1}, d_{k+1}] \cup \dots \cup [c_r, d_r]$, such that $w \in \{a_1, \dots, a_{k+1}\}$. Next, choose b_1, \dots, b_{k+1} , such that no $nb(b_i)$ belongs to an interval $[c_j, d_j]$ ($1 \leq i \leq k+1$, and $1 \leq j \leq r$), and that $nb(a_1) < nb(b_1) < nb(a_2) < nb(b_2) < \dots < nb(a_k) < nb(b_k) < nb(a_{k+1}) < nb(b_{k+1})$ or $nb(b_1) < nb(a_1) < nb(b_2) < nb(a_2) < \dots < nb(b_k) < nb(a_k) < nb(b_{k+1}) < nb(a_{k+1})$. (It is easy to see that this can be done: in general, pick vertices whose number is between d_j and c_{j+1} .)

Let T be the shortest paths tree containing shortest paths from v to all other vertices. (See e.g. [11], Chapter 25.) The paths in T from v to a vertex a_i , $1 \leq i \leq k+1$ must use the edge $\{v, w\}$, while the paths in T from v to a vertex b_i do not use this edge. \square

Similar results can be shown for k -IRS, k -LIRS, and k -SLIRS: in case of non-strict versions, additionally we require that $v \notin \{a_1, \dots, a_{k+1}, b_1, \dots, b_{k+1}\}$, and in case of linear versions, b_{k+1} is not used, and the condition on the numbers of vertices a_i, b_i becomes: $nb(a_1) < nb(b_1) < nb(a_2) < nb(b_2) < \dots < nb(a_k) < nb(b_k) < nb(a_{k+1})$.

Theorem 12 *For any fixed $k \geq 1$, one can construct algorithms that test whether for a given graph $G = (V, E)$ with a node labelling nb and for all costs assignments $c : E \rightarrow \mathbf{R}^+ \cup \{0\}$, there exists an optimal k -IRS (or: k -SIRS, k -LIRS, k -SLIRS) (nb, l) for G with costs c , in $O(n^{2k+3})$ ($O(n^{2k+3})$, $O(n^{2k+2})$, $O(n^{2k+2})$) time.*

Proof: We consider the algorithm for checking existence of an optimal k -SIRS. First, we use the algorithm from [5] to check in linear time whether the treewidth of G is at most $4k$, and if so, to build a tree-decomposition of G of treewidth at most $4k$. If the treewidth of G is more than $4k$, then by Corollary 22, $G \notin k$ -SIRS, so also for the node labelling nb , there exists a cost assignment which requires at least $k+1$ intervals for some link: we can output ‘no’, and stop.

So, now suppose we have a tree-decomposition of G of treewidth at most $4k$. It is well known that $|E| \leq 4k|V|$. Now, for every vertex $v \in V$, and for every $(v, w) \in E$, and for all vertices $a_1, \dots, a_{k+1}, b_1, \dots, b_{k+1} \in V$, with $nb(a_1) < nb(b_1) < nb(a_2) < nb(b_2) < \dots < nb(a_k) < nb(b_k) < nb(a_{k+1}) < nb(b_{k+1})$ or $nb(b_1) < nb(a_1) < nb(b_2) < nb(a_2) < \dots < nb(b_k) < nb(a_k) < nb(b_{k+1}) < nb(a_{k+1})$ and $w \in \{a_1, \dots, a_{k+1}\}$, we check whether there exists a spanning tree $T = (V, F)$ of G such that

- For each i , $1 \leq i \leq k+1$, the path in T from v to a_i uses the edge $\{v, w\}$.
- For each i , $1 \leq i \leq k+1$, the path in T from v to b_i does not use the edge $\{v, w\}$.

If one of these checks is true, we know by Lemma 11 that there is a cost assignment for which no k -SIRS (nb, l) exists; otherwise we know that for all cost assignments such a k -SIRS does exist.

Each check can be done in linear time, with help of the tree-decomposition: notice, that for fixed $v, w, a_1, \dots, a_{k+1}, b_1, \dots, b_{k+1}$, the existence of T fulfilling the given properties can be formulated in *Monadic Second Order Logic*, and hence be decided (with an algorithm that can be constructed) in linear time for graphs of bounded treewidth (see [1, 10, 12]). As we must make in total less than $|E| \cdot n^{2k+1} \cdot k = O(n^{2k+2})$ checks, the time bound follows.

The algorithms for the cases of k -IRS, k -LIRS, and k -SLIRS are similar: because b_{k+1} is not used, the time bounds for k -LIRS and k -SLIRS are a linear factor smaller. \square

Corollary 13 *One can construct an algorithm that tests whether for a given integer $k \in \mathbf{N}$, and graph $G = (V, E)$, $G \in k$ -IRS (or: $G \in k$ -SIRS, $G \in k$ -LIRS, $G \in k$ -SLIRS).*

Proof: Use the algorithm of Theorem 12 for each permutation (numbering) of the vertices of G . \square

4 The treewidth of graphs with k -label interval routing schemes

The main object of this section is to prove the following result.

Theorem 14 *Every graph $G = (V, E)$ contains $K_{2,r}$ as a minor or has treewidth at most $2r - 2$.*

A variant of these results with a sharper bound for the case that G is planar is discussed in the next section.

Given a graph $G = (V, E)$ and a set $S \subseteq V$, let $\partial S = \{v \in V - S \mid \exists u \in S, \{u, v\} \in E\}$ (i.e., the neighbours of vertices in S that do not belong to S).

Definition. A set $S \subseteq V$ is an s - t -separator in $G = (V, E)$ ($s, t \in V$), if s and t belong to different connected components of $G[V - S]$. S is a *minimal s - t -separator*, if it does not contain another s - t -separator as a proper subgraph. S is a *minimal separator*, if there exist vertices $s, t \in V$ for which S is a minimal s - t -separator.

Note that minimal separators can contain other minimal separators as proper subgraphs. We will use in fact a different property of minimal separators, as given in the following lemma, which is easy to proof.

Lemma 15 *A non-empty set S is a minimal separator in G , if and only if there are at least two connected components, G_1, G_2 of $G[V - S]$ such that $S \subseteq \partial V(C_i)$, $i = 1, 2$ (i.e. each vertex in S has a neighbour in both G_1 and G_2). We call two such components separated components.*

Lemma 16 *If G contains a minimal separator S , with $|S| \geq r$, then $K_{2,r}$ is a minor of G .*

Proof: Let S be a minimal separator and consider two separated components G_A and G_B of $G[V - S]$. Remove any vertex from any other component and $|S| - r$ vertices from S . If we now contract all edges in G_A and G_B that are not incident with a vertex in S , we obtain $K_{2,r}$. \square

Definition. Let $G = (V, E)$ be a graph and \mathcal{S} a collection of subsets of $V(G)$. Denote by $\text{CL}(G, \mathcal{S})$ the graph obtained from G by making every set $S_i \in \mathcal{S}$ into a clique, i.e., $\text{CL}(G, \mathcal{S}) = (V, E \cup \{\{v, w\} \mid v \neq w, \exists S_i \in \mathcal{S} : v, w \in S_i\})$.

Definition. Let G be a graph and \mathcal{S} a collection of subsets of $V(G)$. Denote by $\text{EX}(G, \mathcal{S})$ the graph obtained from G by adding to every set $S_i \in \mathcal{S}$ a new vertex $v_{\text{new},i}$ which is adjacent to all vertices in S_i . (In case $|\mathcal{S}| = 1$, we denote the “new” vertex as v_{new}).

Definition. For given $r \geq 1$, let \mathcal{D}_r be the class of all graphs $G = (V_0 \cup V_1 \cup V_2 \cup V_3, E)$, such that

- V_0, V_1, V_2, V_3 are disjoint sets.
- $V_0 = \{v_0\}$. v_0 is adjacent to all vertices in V_1 and no vertices in $V_2 \cup V_3$.
- $|V_1| < r$. Every vertex in V_1 is adjacent to at least one vertex in V_2 and to no vertex in V_3 .
- Every vertex in V_2 is adjacent to at least one vertex in V_1 .
- Every vertex in V_3 is adjacent to less than r vertices in V_2 , and is not adjacent to vertices in $V_0 \cup V_1 \cup V_3$.

Finally, if $R \in \mathcal{D}_r$, we define $\text{CL}(R) = \text{CL}(R, \{\partial\{v\} : v \in V_0(R) \cup V_3(R)\})$. Also, we define $\text{CL}(\mathcal{D}_r) = \{\text{CL}(R) : R \in \mathcal{D}_r\}$.

Let $V_3 = \{v_1^3, \dots, v_m^3\}$.

Lemma 17 (See [6].) *For any graph $G = (V, E)$, either $K_{1,r}$ is a minor of G or $\text{treewidth}(G) \leq r - 1$.*

Proof: W.l.o.g., suppose that G is connected. Take an arbitrary depth first search tree T of G . For any vertex v , let Y_v be the set of ancestors of v in T that are adjacent to v or to a descendant of v , and let $X_v = \{v\} \cup Y_v$. One can show that if $|Y_v| \geq r$, then G contains $K_{1,r}$ as a minor (contract v with all its descendants, and then remove all vertices not in X_v .) For all v , if $|Y_v| \leq r - 1$ then $(\{X_v \mid v \in V\}, T)$ is a tree-decomposition of G of treewidth at most $r - 1$. \square

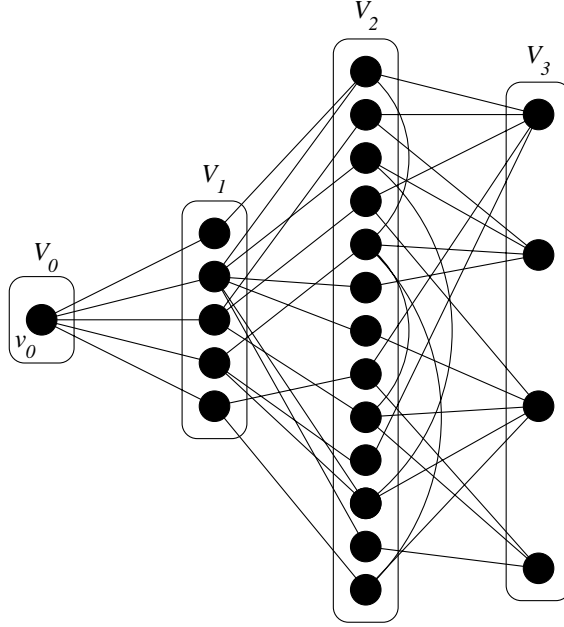


Figure 1: Example of a graph in D_6 .

Lemma 18 (See e.g., [8].) *Let $(\{X_i, i \in I\}, T)$ be a tree-decomposition of graph $G = (V, E)$. For any clique K of G , there exists an $i \in I$ with $V(K) \subseteq X_i$.*

Lemma 19 *For any graph $G \in \mathcal{D}_r$, either $K_{2,r}$ is a minor of G or G has a tree-decomposition of treewidth $\leq 2r - 2$ which is also a tree-decomposition of $\text{CL}(G)$.*

Proof: Let $G_{\text{clique}} = \text{CL}(G)$, $V(K_{1,r}) = \{w_0, w_1, \dots, w_r\}$ and $E(K_{1,r}) = \{\{w_0, w_1\}, \dots, \{w_0, w_r\}\}$. From Lemma 17, either $K_{1,r}$ is a minor of $G_{\text{clique}}[V_2]$ or $\text{treewidth}(G_{\text{clique}}[V_2]) \leq r - 1$. We consider these cases separately.

Case 1. $K_{1,r}$ is a minor of $G_{\text{clique}}[V_2]$: Let $i = 0, \dots, r$ and S_{w_i} be the set of vertices in $G_{\text{clique}}[V_2]$ that were identified to w_i when creating $K_{1,r}$ as a minor. Notice that any set S_{w_i} induces a connected subgraph in $G_{\text{clique}}[V_2]$. Denote by R the set of vertices in V_3 that are adjacent to vertices in S_{w_0} . Finally let w_i be a vertex in S_{w_i} that is adjacent to a vertex in S_{w_0} . (Note that these vertices w_i exist, by the construction of $K_{1,r}$ as a minor.) We observe that $G_{\text{clique}}[R \cup S_{w_0}]$ is connected.

Claim I: $G[R \cup S_{w_0}]$ is connected.

Suppose not. As $E(G[R \cup S_{w_0}]) \subseteq E(G_{\text{clique}}[R \cup S_{w_0}])$, we can add edges in $E(G_{\text{clique}}[R \cup S_{w_0}]) - E(G[R \cup S_{w_0}])$ to $G[R \cup S_{w_0}]$ until an edge, say $\{x_1, x_2\}$

makes the graph connected. As $\{x_1, x_2\}$ belongs to $E(G_{\text{clique}}[R \cup S_{w_0}])$, but not to $E(G[R \cup S_{w_0}])$, the edge is in one of the added cliques, i.e., there must be a vertex $x_3 \in V_3$ that is adjacent to both x_1 and x_2 . Now we have a contradiction, as $x_3 \in R \subseteq R \cup S_{w_0}$.

Claim II: For all i , w_i is adjacent to a vertex in $R \cup S_{w_0}$ in G .

For all i , there exists a vertex $x_i \in S_{w_0}$ that is adjacent to w_i in G_{clique} . If $\{w_i, x_i\} \in E(G)$, then we are done. If $\{w_i, x_i\} \notin E(G)$, then there is a vertex $x_i^3 \in V_3$ with $\{x_i, x_i^3\}, \{w_i, x_i^3\} \in E(G)$, and the claim is true, as w_i is adjacent to $x_i^3 \in R$.

We can now show that $K_{2,r}$ is a minor of G . First contract all vertices in $R \cup S_{w_0}$ to a single vertex z_0 . Next for each i , contract all vertices in S_{w_i} to a single vertex, say z_i . Then contract all vertices in $V_0 \cup V_1$ to a single vertex z_{r+1} . (We can do all of these contractions, as each of these sets induces a connected subgraph of G .) By claim II, z_0 is adjacent to each vertex in $\{z_1, \dots, z_r\}$. Also for each i , as $S_{w_i} \subseteq V_2$ and each vertex in V_2 is adjacent to at least one vertex in V_1 , z_{r+1} is adjacent to each vertex z_i , $1 \leq i \leq r$. We now have a $K_{2,r}$ minor.

Case 2. $\text{treewidth}(G_{\text{clique}}[V_2]) \leq r - 1$: We now show that $\text{treewidth}(G_{\text{clique}}) \leq 2r - 2$. Take a tree-decomposition $(\{X_i \mid i \in I\}, T = (I, F))$ of $G_{\text{clique}}[V_2]$ with $\text{treewidth} \leq r - 1$. Observe that $(\{X_i \cup V_1 \mid i \in I\}, T)$ is a tree-decomposition of $G_{\text{clique}}[V_1 \cup V_2]$ with treewidth at most $r - 1 + |V_1| \leq 2r - 2$. Using this tree-decomposition, we can build a tree-decomposition of G_{clique} of $\text{treewidth} \leq 2r - 2$, as follows. Add nodes j_0, j_1^3, \dots, j_m^3 to I , with $X_{j_0} = \{v_0\} \cup \partial\{v_0\}$ and $X_{j_i^3} = \{v_i^3\} \cup \partial\{v_i^3\}$. By Lemma 18 there exists for each i a node $j'_i \in I$, with $\partial\{v_i^3\} \subseteq X_{j'_i}$. We make j_i^3 adjacent to this node j'_i . Finally, make j_0 adjacent to an arbitrary node $j'_0 \in I$. We now have a tree-decomposition of G of treewidth at most $2r - 2$. \square

Definition. A *terminal graph* is a triple $G = (V, E, S)$ where (V, E) is a graph and $S \subseteq V$ is an ordered subset of its vertices. We call S the *terminal set* of G .

Definition. Consider two terminal graphs $G_i = (V_i, E_i, S_i)$, $i = 1, 2$ such that $|S_1| = |S_2|$. Define $G_1 \oplus G_2$ as the graph obtained by taking the disjoint union of G_1 and G_2 and then identifying the corresponding terminal vertices in S_1 and S_2 .

Lemma 20 Consider two terminal graphs $G_i = (V_i, E_i, S_i)$, $i = 1, 2$ such that $|S_1| = |S_2|$. Suppose that for $i = 1, 2$, $G_i[S_i]$ is a clique. If $\text{treewidth}(G_i) \leq k_i$, $i = 1, 2$, then there is a tree-decomposition with $G_1 \oplus G_2$ of treewidth at most $\max\{k_1, k_2\}$.

Proof: Take tree-decompositions $(\{X_j^i \mid j \in I^i\}, T^i = (I^i, F^i))$ of G^i of treewidth at most k_i , $i = 1, 2$. By Lemma 18, there are $j_0^i \in I^i$ with $S_i \subseteq X_{j_0^i}^i$, $i = 1, 2$.

Taking the disjoint union of the two tree-decompositions and connecting nodes j_0^1 and j_0^2 yields the desired tree-decomposition: one easily verifies that $(\{X_j^1 \mid j \in I^1\} \cup \{X_j^2 \mid j \in I^2\}, T = (I^1 \cup I^2, F^1 \cup F^2 \cup \{\{j_0^1, j_0^2\}\}))$ is a tree-decomposition with $G_1 \oplus G_2$ with treewidth at most $\max\{k_1, k_2\}$. \square

We are now ready to prove Theorem 14. In fact, we prove the following, slightly stronger result.

Theorem 21 *Let $G = (V, E)$ be a graph that is not a clique. Then, for any $r \geq 1$, either $K_{2,r}$ is a minor of G or for any minimal separator S where $|S| < r$, G has a tree-decomposition with treewidth $\leq 2r - 2$ that is also a tree-decomposition of $\text{CL}(G, \{S\})$.*

Proof: We use induction on $|V|$. The theorem clearly holds for $|V| = 3$. Assume that the theorem holds for any graph with less than n vertices. Let $G = (V, E)$ be a graph with n vertices and let S be a minimal separator with $|S| < r$ (in the case where $|S| \geq r$, we have by Lemma 16 that $K_{2,r}$ is a minor of G). Let $G_i = (V_i, E_i)$, $i = 1, \dots, m$, be the connected components of $G[V - S]$ and $\bar{G}_i = \text{EX}(G[V_i \cup \partial V_i], \{\partial V_i\})$. We denote the corresponding “new” nodes as v_{new}^i . Notice that each graph \bar{G}_i has $\partial\{v_{\text{new}}^i\}$ as a minimal separator and $|\partial\{v_{\text{new}}^i\}| < r$. We consider two cases.

Case 1. $|V(\bar{G}_i)| < n$ for all $i \leq i \leq m$. From the induction hypothesis it follows that either $K_{2,r}$ is a minor of \bar{G}_i for some i or for all i , \bar{G}_i has a tree-decomposition of treewidth $\leq 2r - 2$ that is also a tree-decomposition of $\text{CL}(\bar{G}_i, \{\partial\{v_{\text{new}}^i\}\})$. In the first case, as \bar{G}_i is a minor of G , $K_{2,r}$ is also a minor of G .

So suppose that for all i , \bar{G}_i has a tree-decomposition of treewidth $\leq 2r - 2$ which is also a tree-decomposition of $\text{CL}(\bar{G}_i, \{\partial\{v_{\text{new}}^i\}\})$. We now construct a tree-decomposition of $\text{CL}(G, \{S\})$ of treewidth $\leq 2r - 2$. Let H_i be the graph obtained from $\text{CL}(\bar{G}_i, \{\partial\{v_{\text{new}}^i\}\})$ by removing the “new” vertex v_{new}^i . Clearly any graph H_i , $i = 1, \dots, m$ has a tree-decomposition of treewidth $\leq 2r - 2$ and is a subgraph of $\text{CL}(G, \{S\})$. Consider now the graphs H_1 and H_2 . We have that $S_{1,2} = V(H_1) \cap V(H_2) \subseteq S$ and thus $S_{1,2}$ induces a clique in H_1 and H_2 . Make H_1 and H_2 into terminal graphs with terminal set $S_{1,2}$. From Lemma 20 the graph $H_{1,2} = H_1 \oplus H_2$ has also a tree-decomposition of treewidth $\leq 2r - 2$. Notice that $H_{1,2}$ is a subgraph of $\text{CL}(G, \{S\})$ and $S_{1,2,3} = V(H_{1,2}) \cap V(H_3) \subseteq S$, thus $S_{1,2,3}$ induces a clique in $H_{1,2}$ and H_3 . Now make $H_{1,2}$ and H_3 terminals with terminal set $S_{1,2,3}$ and apply again Lemma 20 to obtain a tree-decomposition of $H_{1,2,3} = H_{1,2} \oplus H_3$ with treewidth $\leq 2r - 2$. In this manner, by repeatedly applying Lemma 20, we can merge all the tree decompositions of the graphs H_i , $i = 1, \dots, m$ and thus construct a tree-decomposition of $\text{CL}(G, \{S\})$ that has treewidth $\leq 2r - 2$.

Case 2. We now examine the remaining case where there are only two connected components in $G[V - S]$ and at least one of them contains only one vertex v_0 . Consider the set $D = \partial S - \{v_0\}$, and assume that it does not contain a minimal separator of cardinality $\geq r$ (if it does, then by Lemma 16, $K_{2,r}$ is a minor of G). Let G_i be the connected components of $G[V - D - S - \{v_0\}]$ and $N_i = \partial V(G_i) \subseteq D$, $i = 1, \dots, m$. Notice that, as D does not contain minimal separators of cardinality $\geq r$, $|N_i| < r$. Let $\bar{G}_i = \text{EX}(G_i, \{N_i\})$, $i = 1, \dots, m$, and denote the corresponding “new” vertices as v_{new}^i . As each \bar{G}_i has less than n vertices, by the induction hypothesis, either $K_{2,r}$ is a minor of \bar{G}_i for some i or \bar{G}_i has a tree-decomposition of treewidth $\leq 2r - 2$ which is also tree-decomposition of $\text{CL}(\bar{G}_i, \{N_i\})$ for each i . In the first case, as before, \bar{G}_i is a minor of G and thus $K_{2,r}$ is a minor of G . In the second case, observe that $F = \text{EX}(G[D \cup S], \{S, N_1, \dots, N_m\})$ is a member of D_r . From Lemma 19, either $K_{2,r}$ is a minor of F (which implies that $K_{2,r}$ is a minor of G , as F is a minor of G) or F has a tree-decomposition of treewidth $\leq 2r - 2$ which is also a tree-decomposition of $\text{CL}(F, \{S, N_1, \dots, N_m\})$. We now construct a tree-decomposition of $\text{CL}(G, \{S, N_1, \dots, N_m\})$ with treewidth $\leq 2r - 2$. For each i let H_i be the graph obtained from $\text{CL}(\bar{G}_i, \{N_i\})$ if we eliminate the “new” vertex v_{new}^i . Also let F_0 be the graph obtained from F by eliminating the “new” vertices $v_{\text{new},i}$ corresponding to the sets N_1, \dots, N_m . Clearly each H_i and F_0 have tree decompositions of treewidth $\leq 2r - 2$. We observe that for F_0 and H_1 , $V(F_0) \cap V(H_1) = N_1$ induces a clique in F_0 and H_1 . Make F_0 and H_1 into terminal graphs with terminal set N_1 . By Lemma 20 the graph $F_1 = F_0 \oplus H_1$ has also a tree-decomposition of treewidth $\leq 2r - 2$. Now N_2 induces a clique on F_1 and H_2 , so we can make them terminals with terminal set N_2 , and apply Lemma 20 again to obtain a tree decomposition of $F_2 = F_1 \oplus H_2$ of treewidth $\leq 2r - 2$. Continuing in this fashion we can merge all the tree decompositions of the graphs H_i , $i = 1, \dots, r$ to the tree-decomposition of F_0 , and thus construct a tree-decomposition of $\text{CL}(G, \{S, N_1, \dots, N_m\})$ of treewidth $\leq 2r - 2$. As $\text{CL}(G, \{S\})$ is a subgraph of $\text{CL}(G, \{S, N_1, \dots, N_m\})$, this completes the proof of the theorem. \square

Corollary 22 (i) *Every graph in k -IRS has treewidth at most $4k$.*

(ii) *Every graph in k -SIRS has treewidth at most $4k$.*

Proof: If $G \in k$ -SIRS or $G \in k$ -IRS, then $K_{2,2k+1}$ is not a minor of G hence G has treewidth at most $2(2k + 1) - 2 = 4k$. \square

A direct consequence is that graphs in the classes k -LIRS and k -SLIRS also have treewidth at most $4k$. These results can be seen as partial characterisations of graphs which allow k -label interval routing schemes (with dynamic edge costs). The result also indicates a limitation of the interval routing method: as ‘most graphs have large treewidth’ (see e.g. [17], chapter 5), the set of graphs in k -IRS only covers a small part of all graphs (or even of all sparse graphs, see [17]).

Interestingly, the proof of Theorem 14 can be made constructive, and can be used to build an algorithm, that either outputs that input graph G has $K_{2,r}$ as a minor, or that outputs a tree-decomposition of G of treewidth at most $2r - 2$, and that uses $O(rn)$ time. Combined with the results of Lemma 12 this can lead to a practical algorithm that checks whether for a given node labelling, an k -IRS (or k -SIRS, k -LIRS, k -SLIRS) exists for this labelling and all possible cost assignments, especially when additional optimisations are used, and k is small (e.g., $k = 2$, or $k = 3$.)

In the case when $k = 1$, more precise bounds are known. As 1-SIRS equals the class of connected outerplanar graphs [15], and outerplanar graphs have treewidth at most 2, every graph in 1-SIRS has treewidth at most 2. Similarly, the characterisations of 1-LIRS in [3] and of 1-SIRS in [2] show that every graph in 1-LIRS has treewidth (and even pathwidth) at most 2, and every graph in 1-IRS has treewidth at most 3, and has treewidth at most 2 if it is not equal to K_4 .

The results also have consequences for *random graphs*. We mention some results, obtained by Kloks [17]. Let $G_{n,m}$ denote a random graph with n vertices and m edges. For a precise meaning of the term ‘almost every’ we refer to [17] or [9].

Theorem 23 (Kloks[17]) (i) Let $\delta > 1.18$. Then almost every graph $G_{n,m}$ with $m \geq \delta n$ has treewidth $\Theta(n)$.

(ii) For all $\delta > 1$ and $0 < \epsilon < (\delta - 1)/(\delta + 1)$, almost every graph $G_{n,m}$ with $m \geq \delta n$ has treewidth at least n^ϵ .

Corollary 24 (i) Let $\delta > 1.18$. Then for almost every graph $G_{n,m}$ with $m \geq \delta n$, the smallest k for which $G_{n,m} \in k$ -IRS is of size $\Theta(n)$.

(ii) Let $\delta > 1$ and $0 < \epsilon < (\delta - 1)/(\delta + 1)$. Then for almost every graph $G_{n,m}$ with $m \geq \delta n$, the smallest k for which $G_{n,m} \in k$ -IRS fulfils $k \geq n^\epsilon$.

5 The treewidth of planar graphs with k -label interval routing schemes

In this section we prove results, similar to those of the previous section, for the case that the graph G is planar.

We first need some definitions and lemmas. The following lemma is easy.

Lemma 25 Let G be a graph with $\text{treewidth}(G) \leq k$, $k \geq 2$ and $w \notin V(G)$. Suppose $G' = (V(G) \cup \{w\}, E(G) \cup \{\{v, w\}, \{u, w\}\})$ where $\{u, v\} \in E(G)$ or $G' = (V(G) \cup \{w\}, E(G) \cup \{\{v, w\}\})$ where $v \in V(G)$. Then $\text{treewidth}(G') \leq k$.

Lemma 26 Let G be a graph and $\mathcal{C} = \{C_1, \dots, C_r\}$ be a collection of vertex sets in G where $G[C_i]$ is a clique for each i . If $\text{treewidth}(G) \leq k$ and $|C_i| \leq k$ for each i then $\text{treewidth}(\text{EX}(G, \mathcal{C})) \leq k$.

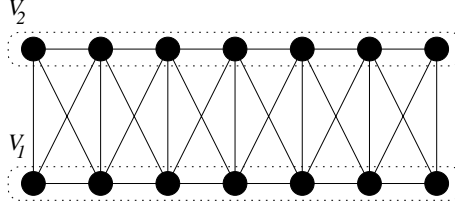


Figure 2: A 7-fence.

Proof: Let $\text{treewidth}(G) \leq k$. There is a tree-decomposition of G of $\text{treewidth} \leq k$. By Lemma 18, for any $C_i \in \mathcal{C}$ there exists a node X_i in the decomposition such that $C_i \subseteq X_i$. Now construct a tree-decomposition of $\text{EX}(G, \mathcal{C})$ by introducing r new nodes $X'_i, i = 1, \dots, r$, where $X'_i = C_i \cup \{v_{\text{new},i}\}$ and make X'_i adjacent to X_i in the new decomposition tree. It is easy to check that the tree-decomposition thus obtained has $\text{treewidth} \leq k$. \square

Definition. A graph $G = (V_1 \cup V_2, E)$ is called an r -fence, if it can be written in the following form: $V = V_1 \cup V_2$, with $V_i = \{v_1^i, \dots, v_r^i\}, i = 1, 2$ and $E = \{(v_j^i, v_{j'}^{i'}) \mid v_j^i \neq v_{j'}^{i'}, |j - j'| \leq 1, i, i' \in \{1, 2\}\}$.

An example of a 7-fence is given in Figure 2.

Lemma 27 *If $G = (V_1 \cup V_2, E)$ is an r -fence then $\text{treewidth}(\text{CL}(G, \{V_1 \cup V_2\})) \leq r + 1$.*

Proof: Take the tree-decomposition $(\{X_i : i \in I\}, T)$ where T is a path with r nodes and $X_1 = \{v_1^1, \dots, v_r^1, v_1^2, v_2^2\}, X_i = X_{i-1} \cup \{v_{i+1}^2\} - \{v_{i-1}^1\}, i = 2, \dots, r - 1$. It is easy to see that this is a tree-decomposition of G with $\text{treewidth} \leq r + 1$. \square

Definition. Let \mathcal{Z}_r be the collection of graphs $G = (V_1 \cup V_2, E)$ that can be constructed as follows:

- Take two disjoint sets of vertices $V_1 = \{v_1^1, \dots, v_{k_1}^1\}, V_2 = \{v_1^2, \dots, v_{k_2}^2\}$ with $k_1, k_2 < r$. We call $V_i, i = 1, 2$, the *parts* of the graph under construction.
- Add edges $\{v_1^i, v_2^i\}, \dots, \{v_{k_i-1}^i, v_{k_i}^i\}, i = 1, 2$ and edges $\{v_1^1, v_1^2\}, \{v_{k_1}^1, v_{k_2}^2\}$.
- Add a maximal set of edges such that
 1. the graph stays planar,
 2. any vertex in V_1 (resp. V_2) is adjacent to at least one vertex in V_2 (resp. V_1),

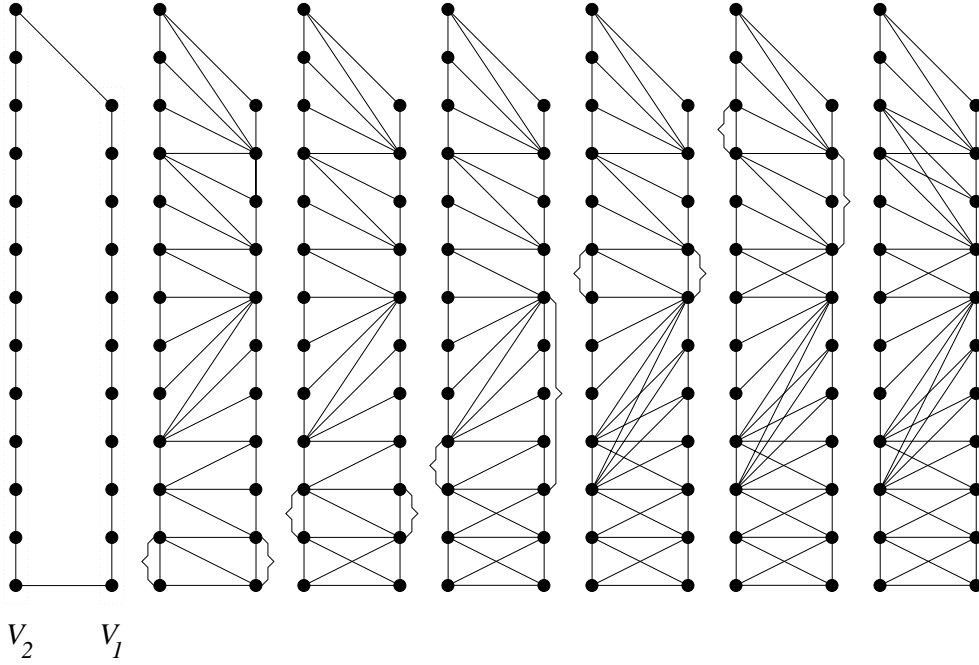


Figure 3: The construction of a graph in \mathcal{Z}_{14} .

3. the resulting planar graph can be embedded such that the outer face is formed by the cycle $(v_1^1, \dots, v_{k_1}^1, v_{k_2}^2, v_{k_2-1}^2, \dots, v_1^2, v_1^1)$.

Notice that the graph R constructed so far is outerplanar.

- The construction is completed by setting $E^j = E(G[V_j]), j = 1, 2$ and applying the following operation for an arbitrary number of times:

For some edge $\{v_i^2, v_{i+1}^2\} \in E^2$ and a set of vertices $V_{l,r}^1 = \{v_l^1, \dots, v_{l+r}^1\} \subseteq V_1, l, r \geq 1$ such that $E(G[V_{l,r}^1]) \subseteq E^1$ and $\{v_i^2, v_l^1\}, \{v_{i+1}^2, v_{l+r}^1\} \in E(G)$ we set

- (i) $E^1 \leftarrow E^1 - E(G[V_{l,r}^1])$
- (ii) $E^2 \leftarrow E^2 - \{\{v_i^2, v_{i+1}^2\}\}$
- (iii) $E(G) \leftarrow E(G) \cup \{\{v_i^2, v_l^1\}, \dots, \{v_i^2, v_{l+r}^1\}\}$
 $\cup \{\{v_{i+1}^2, v_l^1\}, \dots, \{v_{i+1}^2, v_{l+r}^1\}\}.$

We call the edges in $E(G[V_2]) - E^2$ *unmarked edges* of G .

For an example of the construction of a graph in \mathcal{Z}_{14} see Figure 3.

From now on, given a graph $G = (V_0 \cup \dots \cup V_\sigma, E)$, we will use the notation $V_i(G) = V_i, i = 0, \dots, \sigma$ (we call V_0, \dots, V_σ *parts* of G).

Definition. If $R \in \mathcal{Z}_r$, then define $\text{CL}(R) = \text{CL}(R, \{V_1(R), V_2(R)\})$. We call $V_i(R), i = 1, 2$, the *parts* of $\text{CL}(R)$. Also we define $\text{CL}(\mathcal{Z}_r) = \{\text{CL}(R) : R \in \mathcal{Z}_r\}$.

Definition. Let $G \in \mathcal{Z}_r$ with parts V_1, V_2 . $D(v_i^j, G), i = 1, \dots, k_j, j = 1, 2$ is defined as the number of vertices that are adjacent to v_i^j and belong to part V_{3-j} .

Lemma 28 *Let $G \in \mathcal{Z}_r$ with parts $V_i = \{v_1^i, \dots, v_{k_i}^i\}, i = 1, 2$, and $k_1, k_2 < r$. Then, either $D(v_1^1, G) \leq 2$ or $D(v_1^2, G) \leq 2$.*

Proof: Let R be the outerplanar graph from which G has been created according to the definition of \mathcal{Z}_r . Let also E^1, E^2 be the edge sets defined by the construction of G according to this definition. Notice that $R \in \mathcal{Z}_r$ and that either $D(v_1^1, R) = 1$ or $D(v_1^2, R) = 1$. Assume first that $D(v_1^1, R) = 1$. If $\{v_1^1, v_2^1\} \notin E^1$, then clearly $D(v_1^1, G) = 1$. If $\{v_1^1, v_2^1\} \in E^1$, then it is easy to see that edge $\{v_1^1, v_2^1\}$ is the only edge in $E(G) - E(R)$ that is incident to v_1^1 and thus $D(v_1^1, G) = 2$. Assume now that $D(v_1^2, R) = 1$. If $\{v_1^2, v_2^2\} \notin E^2$, then $D(v_1^2, G) = 1$. If $\{v_1^2, v_2^2\} \in E^2$, then it is easy to see that $D(v_1^2, R) = 2$ and no edge in $E(G) - E(R)$ is incident to v_1^2 . Thus, $D(v_1^2, G) = 2$. \square

Lemma 29 *If $G \in \text{CL}(\mathcal{Z}_r)$, then $\text{treewidth}(G) \leq r$.*

Proof: Suppose that $R = (V_1 \cup V_2, E) \in \mathcal{Z}_r$. We will show that $G = \text{CL}(R, \{V_1, V_2\})$ has $\text{treewidth} \leq r$.

We will use induction on r . If $r \leq 3$, then the proposition is trivial. We assume that lemma holds for any $r \leq k$. We will prove that if $R = (V_1 \cup V_2, E) \in \mathcal{Z}_{k+1}$, then $\text{treewidth}(\text{CL}(R)) \leq k + 1$.

Let $R \in \mathcal{Z}_{k+1}$ with parts V_1 and V_2 . Recall that $k_i = |V_i|, i = 1, 2$. If $k_1 < k$ and $k_2 < k$ then $R \in \mathcal{Z}_k$ and the induction step is obtained easily. We claim that it suffices to prove the induction step for $k_1 = k_2 = k$. This is so because in case $k = k_j > k_{3-j}$ for some $j = 1, 2$, then we set $i = 3 - j$ and construct a graph R^* , containing R as a subgraph, as follows: add $k_j - k_i$ vertices $v_{k_i+1}^i, \dots, v_{k_j}^i$ in part V_i and then add the edge sets $\{\{v_{k_i}^i, v_{k_i+1}^i\}, \dots, \{v_{k_j-1}^i, v_{k_j}^i\}\}$ and $\{\{v_{k_j}^j, v_{k_i+1}^i\}, \dots, \{v_{k_j}^j, v_{k_j}^i\}\}$. Clearly, $R^* \in \mathcal{Z}_{k+1}$ and $\text{treewidth}(\text{CL}(R)) \leq \text{treewidth}(\text{CL}(R^*))$ (see Figures 4.i, 4.ii). Thus, it is sufficient to examine the case where $k_1 = k_2 = k$.

If R is a k -fence then the result follows from Lemma 27. Suppose that R is not an k -fence.

If $D(v_1^1, R) \neq 2$ or $D(v_1^2, R) \neq 2$ then set $q = 1$, otherwise set

$$q = \max\{ i : R[\{v_1^1, \dots, v_i^1, v_1^2, \dots, v_i^2\}] \text{ is an } i\text{-fence} \\ \text{and the vertex set } \{v_i^1, v_i^2\} \text{ is a separator of } R \}.$$

Notice that $q < k$. Let $R' = R[\{v_q^1, \dots, v_k^1, v_q^2, \dots, v_k^2\}]$ and observe that $R' \in \mathcal{Z}_{k-q+2}$ (see Figure 4.ii).

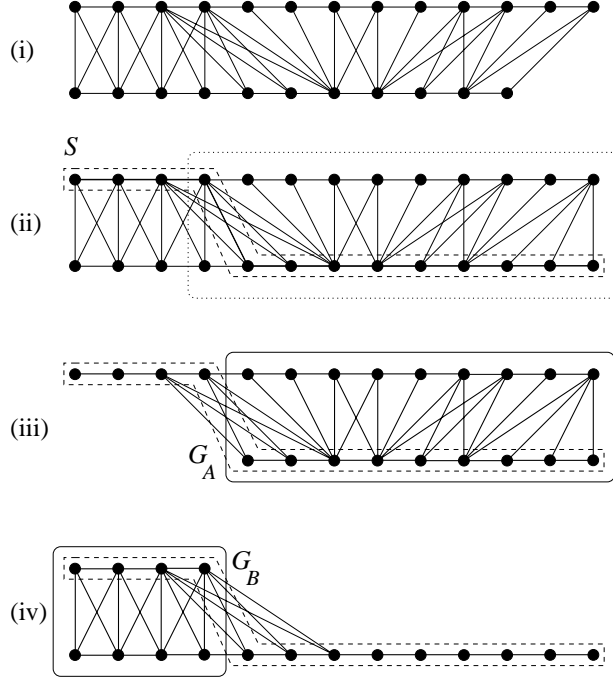


Figure 4: An example of the proof of Lemma 29.

Using Lemma 28 we define m according to the following cases:

Case (i) $D(v_q^1, R') = 1$. In this case we set $m = q$.

Case (ii) $D(v_q^2, R') = 1$. In this case we also set $m = q$.

Case (iii) $D(v_q^1, R') > 1$ and $D(v_q^2, R') > 1$. By a case analysis similar to Lemma 28 we can see that $D(v_q^1, R') = 2$ and $D(v_q^2, R') > 2$ and we set $m = q + 1$.

Observe that in Cases (i) and (iii) the set $S = \{v_1^2, \dots, v_m^2, v_{m+1}^1, \dots, v_k^1\}$ is a separator of $G = \text{CL}(R)$ (see Figure 4.ii). Also, in Case (ii) the set $T = \{v_1^1, \dots, v_m^1, v_{m+1}^2, \dots, v_k^2\}$ is a separator of $G = \text{CL}(R)$.

We will present the proof for the Cases (i) and (iii). We set $G_A = G[\{v_{m+1}^1, \dots, v_k^1, v_{m+1}^2, \dots, v_k^2\}]$. Notice that G_A is a subgraph of a graph in $\text{CL}(\mathcal{Z}_{k-m+1})$ and, as $k - m + 1 \leq k$, by the induction hypothesis, G_A has a tree-decomposition $D_A = (\{X_i : i \in I\}, T)$ of treewidth $\leq k - m + 1$. Notice that from Lemma 18 it follows that D_A contains a node X_A such that $X_A \supseteq \{v_{m+1}^1, \dots, v_k^1\}$. Clearly, $D'_A = (\{X_i \cup \{v_1^2, \dots, v_m^2\} : i \in I\}, T)$ is a tree-decomposition of $G'_A = G[V_2 \cup S]$ with treewidth $\leq k + 1$. Notice also that D'_A contains node $X'_A = X_A \cup \{v_1^2, \dots, v_m^2\} \supseteq S$ and thus $\text{treewidth}(\text{Cl}(G'_A, \{S\})) \leq k + 1$ (see Figure 4.iii).

Let now $G_B = G[\{v_1^1, \dots, v_m^1, v_1^2, \dots, v_m^2\}]$.

We define a tree-decomposition D_B of G_B as follows: $D_B = (\{X_i : i \in I\}, T)$ where T is a path consisting of $m - 1$ nodes and $X_1 = \{v_1^1, \dots, v_m^1, v_1^2, v_2^2\}$,

$X_i = X'_{i-1} - \{v_{i-1}^1\} \cup \{v_{i+1}^2\}, i = 2, m - 1$. It is easy to see that D_B is a tree-decomposition of G_B with treewidth $\leq m + 1$. Notice that D_B contains node $X_B = X_{m-1} \supseteq \{v_1^2, \dots, v_m^2\}$. Clearly, $D'_B = (\{X_i \cup \{v_{m+1}^1, \dots, v_k^1\} : i \in I\}, T)$ is a tree-decomposition of $G'_B = G[V_1 \cup S]$ with treewidth $\leq k + 1$. Notice also that D'_B contains node $X'_B = X_B \cup \{v_{m+1}^1, \dots, v_k^1\} \supseteq S$ and thus $\text{treewidth}(\text{Cl}(G'_B, \{S\})) \leq k + 1$ (see Figure 28.iv).

Now, if we consider the terminal graphs, obtained by taking $\text{Cl}(G'_A, \{S\})$ and $\text{Cl}(G'_B, \{S\})$ with S as set of terminals, and then apply Lemma 20, we obtain that $\text{treewidth}(G) \leq \text{treewidth}(\text{Cl}(G, \{S\})) = \text{treewidth}(\text{Cl}(G'_A, \{S\}) \oplus \text{Cl}(G'_B, \{S\})) \leq k + 1$.

Case (ii) is similar to Case (i). (We use separator T instead of S). \square

Definition. Let \mathcal{Q}_r be the collection of graphs $G = (V_1 \cup V_2, E)$ that can be constructed as follows:

Let R be a graph in \mathcal{Z}_{r+1} containing an unmarked edge $\{v_i^2, v_{i+1}^2\}, 1 \leq i < |V_2|$. We apply the following two operations:

(i) Let $R' = (V(R), E')$ where $E' = E(R) \cup \{\{v_i^2, v_1^1\}, \dots, \{v_i^2, v_{|V_1|}^1\}\} \cup \{\{v_{i+1}^2, v_1^1\}, \dots, \{v_{i+1}^2, v_{|V_1|}^1\}\}$.

(ii) Identify vertices v_1^i and $v_{|V_1|}^i, i = 1, 2$ in R' .

If $R \in \mathcal{Q}_r$ with parts V_1, V_2 , we define $\text{Cl}(R) = \text{Cl}(R, \{V_1, V_2\})$. Also, we define $\text{Cl}(\mathcal{Q}_r) = \{\text{Cl}(R) : R \in \mathcal{Q}_r\}$.

For an example of the construction of a graph in \mathcal{Q}_{13} see Figure 5.

Lemma 30 *If $G \in \text{Cl}(\mathcal{Q}_r)$ then $\text{treewidth}(G) \leq r + 2$.*

Proof: We prove that if $R \in \mathcal{Q}_r$, then $G = \text{Cl}(R)$ has $\text{treewidth} \leq r + 2$. According to the definition of \mathcal{Q}_r , R is constructed by a graph $R_0 \in \mathcal{Z}_{r+1}$ containing an unmarked edge $\{v_i^2, v_{i+1}^2\}$. Let a be a vertex in $V_1(R_0)$ connected with both v_i^2 and v_{i+1}^2 and let b be a neighbour of a in $R[V_1]$. Let also $R_1 = R[V(R) - \{v_i^2, v_{i+1}^2, a, b\}]$. It is easy to see that R_1 is a subgraph of a graph $R_2 \in \mathcal{Z}_{r-1}$ (we use the fact that $\{v_i^2, v_{i+1}^2\}$ is an unmarked edge in R_0). From Lemma 29 we have that $\text{treewidth}(\text{Cl}(R_2)) \leq r - 2$ and $\text{Cl}(R_2)$ has a tree-decomposition of $\text{treewidth} \leq r - 2$. We can now obtain a tree-decomposition of $G = \text{Cl}(R)$ by adding vertex set $\{a, b, c, d\}$ in each node of the tree-decomposition of R_2 . Clearly, this tree-decomposition has $\text{treewidth} \leq r + 2$; this completes the proof of the lemma. \square

Note that if $G = (V_0 \cup V_1 \cup V_2 \cup V_3, E)$ is a planar graph in \mathcal{D}_r , then $G[V_2]$ is outerplanar, because otherwise there would exist a vertex in $G[V_2]$ not adjacent with some vertex in V_1 contradicting the definition of \mathcal{D}_r . Using this fact we can easily see that if G is a planar graph in \mathcal{D}_2 , then $\text{Cl}(G)$ has $\text{treewidth} \leq 3$. Also it is not hard to see that if G is a planar graph in \mathcal{D}_3 , then $\text{Cl}(G)$ has $\text{treewidth} \leq 4$.

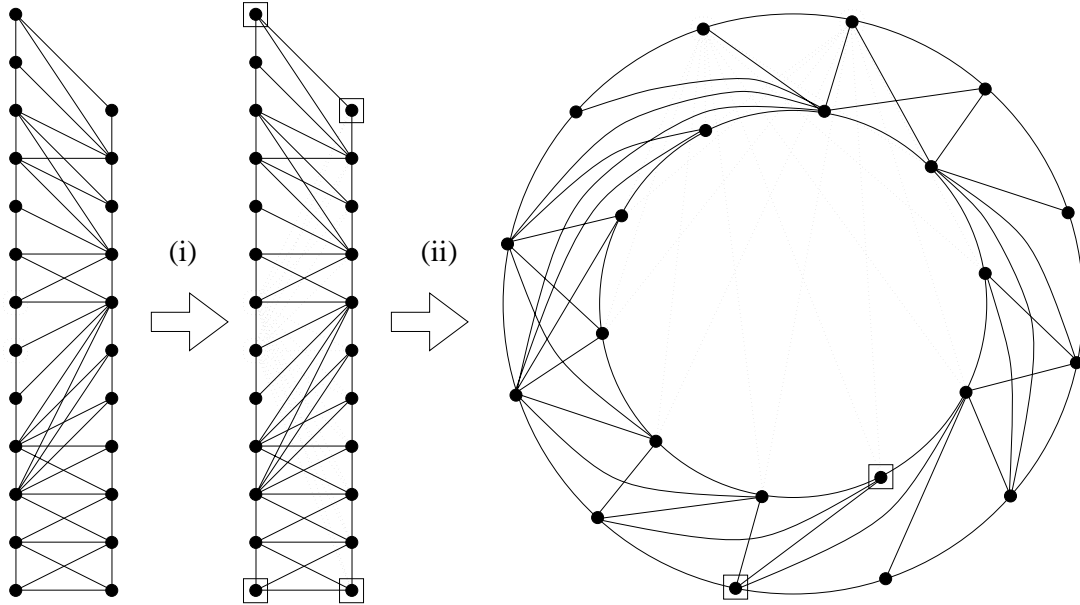


Figure 5: The construction of a graph in \mathcal{Q}_{13}

In what follows, we will prove that if G is a planar graph in \mathcal{D}_r , $r \geq 4$, then $\text{CL}(G)$ has treewidth $\leq r + 2$.

In the remainder of this section, we let \mathcal{D}_r be as defined in the previous section, but we remove from the class all graphs that contain at least one vertex of degree at most two. Lemma 25 shows that we can actually remove these graphs (the purpose of the proof below is to show that planar graphs in \mathcal{D}_r have treewidth at most $r + 2$.)

Definition. We define \mathcal{P}_r , $r \geq 4$ as the set of planar graphs that can be constructed from a planar graph $G \in \mathcal{D}_r$, $r \geq 4$, by applying the following four steps:

- (a) Consider a planar embedding of G . W.l.o.g. we assume that the vertices in V_1 are arranged around v_0 in the cyclic order $v_1^1, \dots, v_{|V_1|}^1, v_1^1$. Let $G' = (V(G'), E(G'))$ be defined by $V(G') \leftarrow V(G)$ and $E(G') \leftarrow E(G) - E(G[V_1]) \cup \{\{v_1^1, v_2^1\}, \dots, \{v_{|V_1|-1}^1, v_{|V_1|}^1\}, \{v_{|V_1|}^1, v_1^1\}\}$. Notice that we have basically the same planar embedding of G' as of G , and that $\text{CL}(G') = \text{CL}(G)$. The resulting graph is now further denoted as G .
- (b) If there exist an edge $\{a, b\} \notin E(G)$ such that at least one of a, b belongs to V_2 , $a, b \notin V_3$, and $G' = (V(G), E(G) \cup \{\{a, b\}\})$ is a planar graph, then add this edge to G . We repeat this step until no such edge can be added in G .

Notice that: (i) all faces in the planar embedding of the resulting graph G are triangles. (ii) $G[V_2]$ is outerplanar and connected.

- (c) If there is a biconnected component in $G[V_2]$ that consists of a single edge $\{a, b\}$ then it is easy to see that there is a vertex $d \in V_1$ such that $\{a, d\}, \{b, d\} \in E(G)$. In this case, add a new vertex c to V_2 , and add edges $\{\{a, c\}, \{b, c\}, \{c, d\}\}$. Repeat this step until there exist no biconnected component in $G[V_2]$ that is an edge. The resulting graph is now further denoted as G .

Notice that $G[V_2]$ is outerplanar and G has an embedding such that all vertices of $V_3(G)$ are at the inside of the cycle(s) formed by $V_2(G)$. We call this planar embedding the *outerplanar embedding*.

In the outerplanar embedding, we identify the *regions of $G[V_2]$* as the areas of the plane that constitute an interior face in the embedding of $V_2(G)$, obtained by restricting the embedding of G to $G[V_2]$.

Notice that each region of V_2 has at most one vertex of V_3 inside and that each region of V_2 that does not contain a vertex of V_3 inside is a triangle.

- (d) If there is a triangle in $G[V_2]$ with vertices a, b, c , whose region does not contain a vertex of V_3 inside, then add a new vertex d to V_3 , and add edges $\{a, d\}, \{b, d\}, \{c, d\}$. Repeat this step until all the regions in $G[V_2]$ contain a vertex of V_3 inside in the outerplanar embedding of G . The resulting graph G becomes a member of \mathcal{P}_r .

After step (d) any region F of $G[V_2]$ contains a vertex of V_3 . We denote this vertex by v_F .

Finally, if $R \in \mathcal{P}_r$, we define $\text{CL}(R) = \text{CL}(R, \{\partial\{v\} : v \in V_0(R) \cup V_3(R)\})$. Also, we define $\text{CL}(\mathcal{P}_r) = \{\text{CL}(R) : R \in \mathcal{P}\}$.

For an example of the construction of a graph in \mathcal{P}_r see Figure 6.

From the definition of \mathcal{P}_r we can see that if any graph in $\text{CL}(\mathcal{P}_r)$ has $\text{treewidth} \leq r + 2$, then for any planar graph $R \in \mathcal{D}_r$, with $r \geq 4$, it holds that the $\text{treewidth CL}(R)$ is at most $r + 2$. In what follows, we prove that if $G \in \text{CL}(\mathcal{P}_r)$, then $\text{treewidth}(G) \leq r + 2$.

Definition. Let $G = (V_0 \cup V_1 \cup V_2 \cup V_3, E)$ be a graph in \mathcal{P}_r , $r \geq 4$. Consider the outerplanar embedding of G . We call the edges that are incident to the exterior face of $G[V_2]$ *exterior edges*. If e is an exterior edge of G , then we denote as $F(e)$ the set of vertices in V_2 that belong to the unique interior face adjacent to e , in the embedding of $G[V_2]$, obtained by restricting the embedding of G .

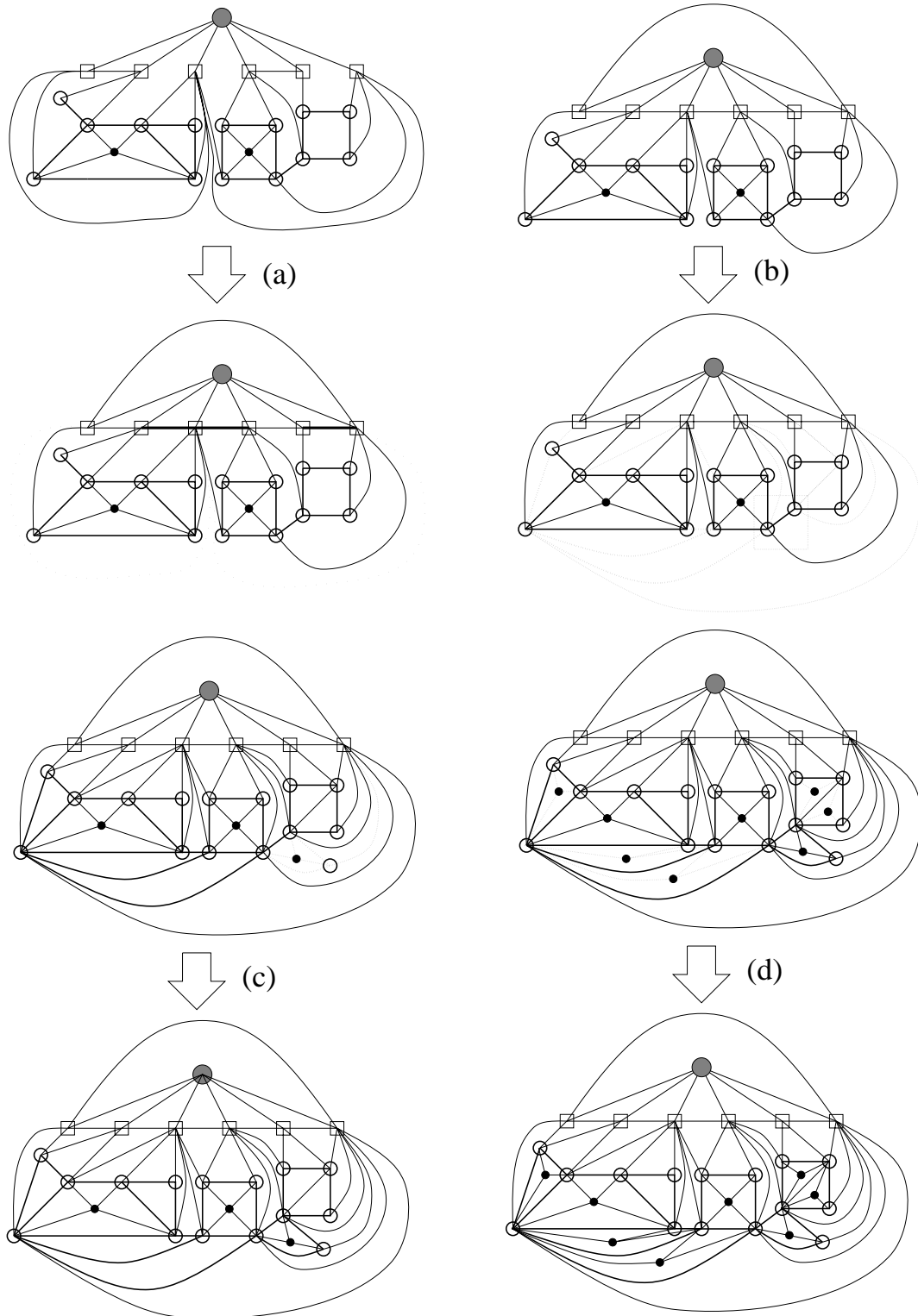


Figure 6: The construction of a graph in \mathcal{P}_7

Definition. Let G be a graph in $\mathcal{P}_r, r \geq 4$. Consider the outerplanar embedding of G . For each region F in G_2 we let \mathcal{S}_F the collection of sets S_e , over all edges $e = \{v, v'\} \in E(G[F])$ that belong to another region of $G[V_2]$, where S_e is the set $\{v, a, \dots, b, v'\}, a, \dots, b \in V_1$ such that the cycle (v, a, \dots, b, v', v) does not separate v_0 and v_F . (Vertices a, \dots, b are consecutive vertices in the cyclic order $v_1^1, \dots, v_{|V_1|}^1, v_1^1$). We denote the region of (v, a, \dots, b, v', v) that does not contain v_0 and v_F the *induced region* of S_e .

Notice that \mathcal{S}_F is a collection of separators of G (see Figure 7). We first prove the following result:

Lemma 31 *If $G = (V_0 \cup V_1 \cup V_2 \cup V_3, E) \in \text{CL}(\mathcal{P}_r), r \geq 4$ such that $G[V_2]$ has only one biconnected component, then for any exterior edge $e = \{v, v'\} \in G[V_2]$, $\text{treewidth}(\text{CL}(G, \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\})) \leq r + 2$.*

Proof: Let G be a graph in $\text{CL}(\mathcal{P}_r)$. Let also $R \in \mathcal{P}_r$ be a planar graph such that $\text{CL}(R) = G$. Recall that $G[V_2]$ is an outerplanar graph. We use induction on the number of faces in $G[V_2]$.

Suppose that the planar embedding of $G[V_2]$ contains only one region. Notice that in this case $F(e) = V_2, \mathcal{S}_{F(e)} = \emptyset$ and thus $G' = \text{CL}(G[V_1 \cup V_2], \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\}) \in \text{CL}(\mathcal{Q}_r)$. From Lemma 26 we have the required as $G = \text{EX}(G', \{\partial\{u\} : u \in V_0 \cup V_3\})$.

Suppose now that lemma holds for any graph $G \in \text{CL}(\mathcal{P}_r)$ where $G[V_2]$ contains less than l faces. Let now G be a graph in $\text{CL}(\mathcal{P}_r)$ where $G[V_2]$ contains l faces. We now prove that lemma also holds for G .

Let $\mathcal{S}_{F(e)} = \{S_1, \dots, S_t\}$. For each $S_i \in \mathcal{S}_{F(e)}$ we define the vertex set $U_i \subseteq V_2 \cup V_3$ to be the set of all the vertices contained in the induced region of S_i . Let $G_i^o = G[\partial U_i \cup U_i], i = 1, \dots, t$ and $G^o = G[V(G) - \bigcup_{i=1, \dots, t} U_i]$. Notice that $V(G^o) \cap V(G_i^o) = S_i, i = 1, \dots, t$. Observe that, as $|F(e)| < r$ and $|V_1| < r$, we have that $\text{CL}(G[V(G^o) - (\{v_{F(e)}\} \cup V_0)], \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\}) \in \mathcal{Q}_r$ and, by Lemmas 30 and 26, $\text{treewidth}(\text{CL}(G^o, \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\})) \leq r + 2$.

Let $S_i = \{u_i, a_i, \dots, b_i, u'_i\}, i = 1, \dots, t$ where $\{u_i, u'_i\} = F(e) \cap V(G_i^o)$ and $\{a_i, \dots, b_i\} = V_1(G) \cap V(G_i^o)$ for $i = 1, \dots, t$ (see the definition of \mathcal{S}_F above). Observe that G_i^o is a subgraph of a graph $G_i'^o$ in $\text{CL}(\mathcal{P}_r)$ and the number of regions in the planar embedding of $V_2(G_i'^o)$ is less than l for $i = 1, \dots, t$ (w.l.o.g. we assume that $V(G_i'^o) = V(G_i^o)$). Notice also that $\{u_i, u'_i\} = S_i - V_1(G_i'^o)$ and that the edge $e_i = \{u_i, u'_i\}$ is an exterior edge in the planar embedding of $V_2[G_i'^o]$ for $i = 1, \dots, t$. From the induction hypothesis we obtain that $\text{treewidth}(\text{CL}(G_i'^o, \mathcal{S}_{F(e_i)} \cup \{\{u_i, u'_i\} \cup V_1(G_i'^o)\})) \leq r + 2, i = 1, \dots, t$. Finally, using that $S_i = \{u_i, u'_i\} \cup V_1(G_i'^o)$, we conclude that $\text{treewidth}(\text{CL}(G_i^o, \{S_i\})) \leq \text{treewidth}(\text{CL}(G_i'^o, \{S_i\})) \leq \text{treewidth}(\text{CL}(G_i'^o, \mathcal{S}_{F(e_i)} \cup \{S_i\})) \leq r + 2, i = 1, \dots, t$.

We now set $H_i = \text{CL}(G_i^o, \{S_i\}), i = 1, \dots, t$, and $H' = \text{CL}(G^o, \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\})$. We make terminal graphs of H_i by taking as set of terminals S_i , for $i =$

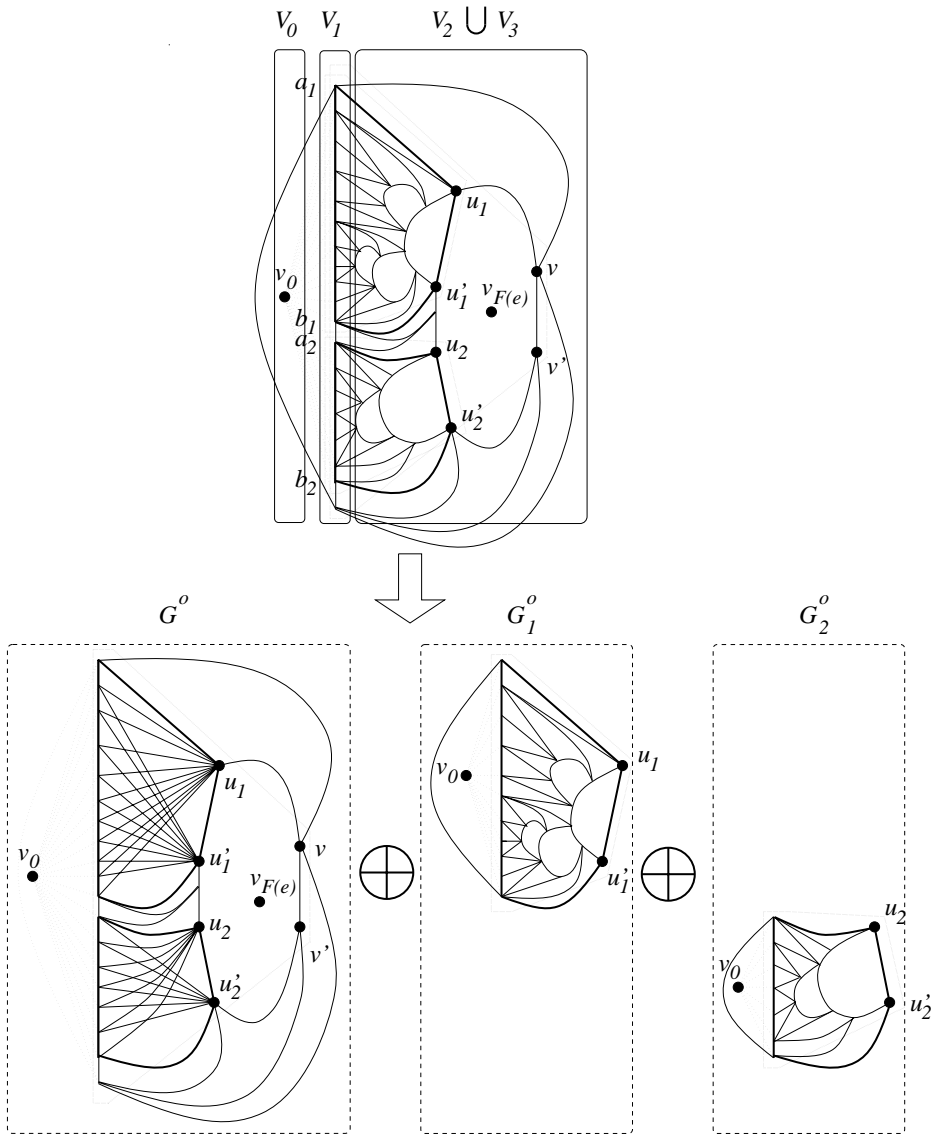


Figure 7: An example of the proof of Lemma 31.

$1, \dots, t$. If now we successively take S_i , $i = 1, \dots, t$ as terminals in H and apply repeatedly Lemma 20 we conclude that the graph $(\dots((H' \oplus H_1) \oplus H_2) \dots \oplus H_t) = \text{CL}(G, \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\})$, (where during each \oplus operation, a different set of terminals is used), has also treewidth $\leq r + 2$ (this construction is the same as the one used in the proof of Theorem 21). For an example of the construction of the proof see Figure 7. \square

Definition. Let G be a graph in \mathcal{P}_r , $r \geq 4$. Given a vertex $v \in V_2$, we define \mathcal{C}_v as the collection of the biconnected components in $G[V_2]$ that contain v . If $|\mathcal{C}_v| > 1$, we call v *rich*, otherwise we call it *poor*.

Definition. Let G be a graph in \mathcal{P}_r . Consider the outerplanar embedding of G . For each vertex $v \in G[V_2]$ we define a collection \mathcal{S}_v of sets of vertices, as follows:

- (i) Set $\mathcal{S}_v \leftarrow \emptyset$. Also, if v is a rich vertex, then set $\mathcal{A} = \emptyset$.
- (ii) We examine two cases:

Case (a) If v is a rich vertex, then for each biconnected component $C_i \in \mathcal{C}_v - \mathcal{A}$, $i = 1, \dots, t$ we set $\mathcal{S}_v \leftarrow \mathcal{S}_v \cup \{S_i\}$ where $G[S_i]$ defines a cycle (v, a, \dots, b, v) , $a, \dots, b \in V_1$ that separates the vertices in $\{v_0\} \cup_{j=1, \dots, t, j \neq i} V(C_j) - \{v\}$ from the vertices in $V(C_i) - \{v\}$ (vertices a, \dots, b are consecutive vertices in the cyclic order $v_1^1, \dots, v_{|V_1|}^1, v_1^1$). We call the area of the plane inside or outside the cycle, that contains the vertices in $V(C_i) - \{v\}$ the *induced region* of S_i .

Case (b) If v is a poor vertex, then there is a unique biconnected component $C_v \in \mathcal{C}_v$ that contains it. Let W_v be the set of the rich vertices in C_v . For each reach vertex $v_i \in W_v$ we set $\mathcal{S}_v \leftarrow \mathcal{S}_v \cup \mathcal{S}_{v_i}$ where \mathcal{S}_{v_i} is defined by setting $\mathcal{A} = C_v$ and applying Case (a) for v_i , $i = 1, \dots, |W_v|$.

Notice that \mathcal{S}_v is a collection of separators of G (see Figure 8 for Case (a) and Figure 9 for Case (b)). We now prove the following:

Lemma 32 *Let $G = (V_0 \cup V_1 \cup V_2 \cup V_3, E)$ be a graph in $\text{CL}(\mathcal{P}_r)$, $r \geq 4$. Then for any vertex $v \in G[V_2]$, $\text{treewidth}(\text{CL}(G, \mathcal{S}_v \cup \{\{v\} \cup V_1\})) \leq r + 2$.*

Proof: Let G be a graph in $\text{CL}(\mathcal{P}_r)$. Let also R be a planar graph such that $G = \text{CL}(R)$. We use induction on the number of biconnected components in $G[V_2]$.

Suppose that $G[V_2]$ contains only one biconnected component. Clearly, for any vertex $v \in G[V_2]$ we have that there exists an exterior edge $e = \{v, v'\}$, $v' \in G[V_2]$ that is incident to v . From Lemma 31 we have that $\text{treewidth}(\text{CL}(G, \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\})) \leq k + 2$. As $\text{CL}(G, \{\{v\} \cup V_1\})$ is a subgraph of $\text{CL}(G, \mathcal{S}_{F(e)} \cup \{\{v, v'\} \cup V_1\})$ and $\mathcal{S}_v = \emptyset$, we have that $\text{treewidth}(\text{CL}(G, \mathcal{S}_v \cup \{\{v\} \cup V_1\})) \leq r + 2$.

Suppose now that the lemma holds for any graph $G \in \text{CL}(\mathcal{P}_r)$ where $G[V_2]$ contains less than l biconnected components. Let now G be a graph in $\text{CL}(\mathcal{P}_r)$

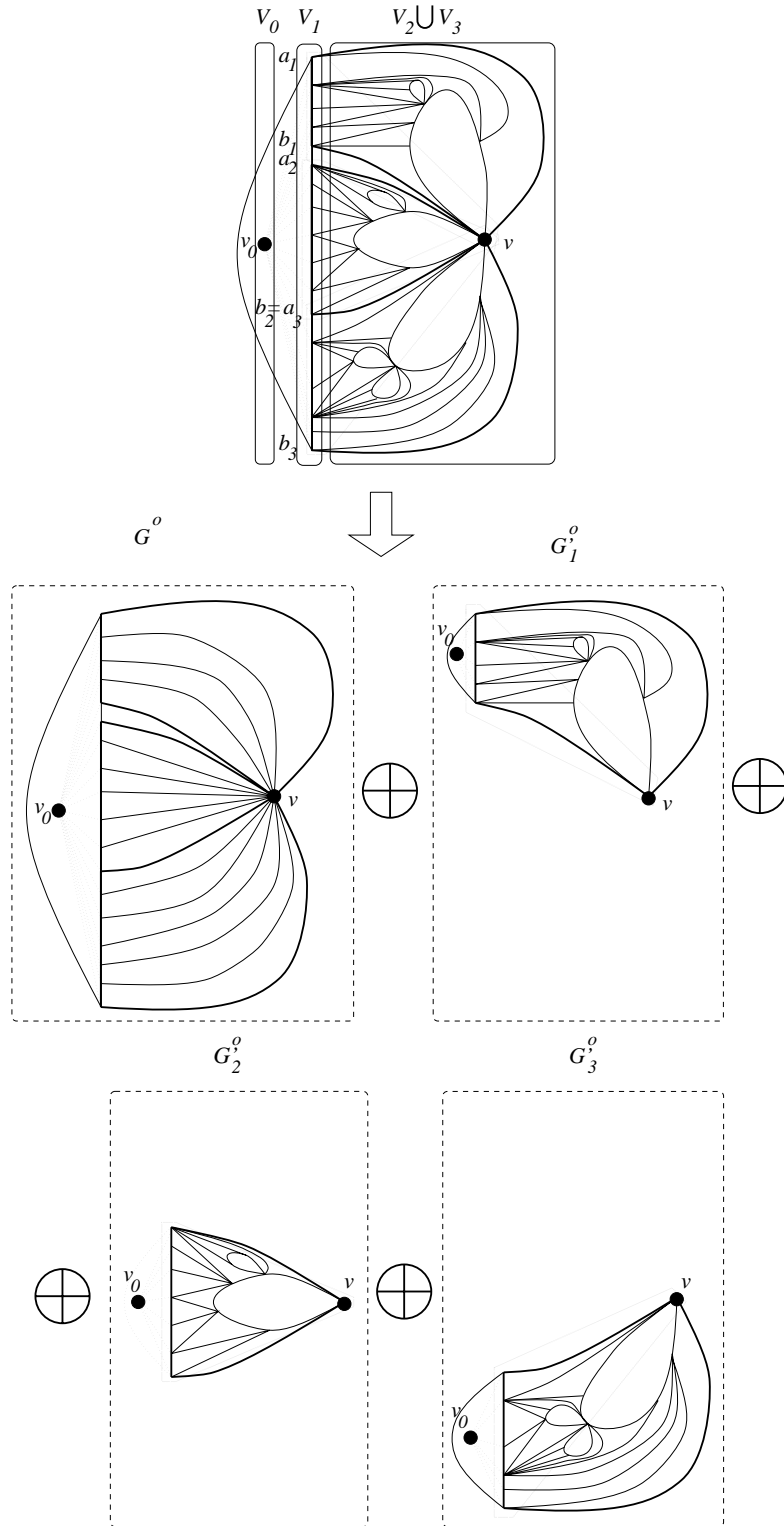


Figure 8: Case (i) of the proof of Lemma 32.

where $G[V_2]$ contains l biconnected components. We prove that the lemma also holds for G .

Let $\mathcal{S}_v = \{S_1, \dots, S_t\}$. For each $S_i \in \mathcal{S}_v, i = 1, \dots, t$ we define the vertex set $U_i \subseteq V_2 \cup V_3$ to be the set of all the vertices contained in the induced region of S_i .

Let $G_i^o = G[\partial U_i \cup U_i], i = 1, \dots, t$ and $G^o = G[V(G) - \bigcup_{i=1, \dots, t} U_i]$. Notice that $V(G_i^o) \cap V(G_j^o) = S_i, i = 1, \dots, t$. We examine two cases:

(i) v is a rich vertex. Notice that $|V(G^o)| \leq r + 1$ and, clearly $\text{treewidth}(\text{CL}(G^o, \mathcal{S}_v \cup \{\{v\} \cup V_1\})) \leq r + 2$ (see Figure 8).

(ii) v is a poor vertex. Let C_v be the unique biconnected component of $G[V_2]$ that contains v . Let also V_3^v be the set of vertices in V_3 that are adjacent with at least two vertices in $V(C_v)$. We define $G^o = G[\{v_0\} \cup V_1 \cup V(C_v) \cup V_3^v]$. It is easy to see that $G' = \text{CL}(G^o, \mathcal{S}_v) \in \text{CL}(\mathcal{P}_r)$ and that $G'[V_2] = C_v$ has only one biconnected component. Let also $e' = \{v, v'\} \in E(C_v)$ be an exterior edge of C_v containing vertex v . From Lemma 31 we have that $\text{treewidth}(\text{CL}(G', \mathcal{S}_{F(e')} \cup \{\{v, v'\} \cup V_1\})) \leq r + 2$. It is now easy to see that, as $\text{CL}(G^o, \mathcal{S}_v \cup \{\{v\} \cup V_1\})$ is a subgraph of $\text{CL}(G', \mathcal{S}_{F(e')} \cup \{\{v, v'\} \cup V_1\})$, we have that $\text{treewidth}(\text{CL}(G^o, \mathcal{S}_v \cup \{\{v\} \cup V_1\})) \leq r + 2$. (See Figure 9).

Observe now that G_i^o is an induced subgraph of a graph $G_i'^o \in \text{CL}(\mathcal{P}_r)$ such that $V(G_i^o) = V(G_i'^o) - \{v_0\}$. Notice also that the number of biconnected components in the planar embedding of $V_2(G_i'^o)$ is $< l$ for $i = 1, \dots, t$. Set $\{u_i\} = S_i - V_1(G_i'^o)$ (notice that if v is a rich vertex then $u_i = v, i = 1, \dots, t$). From the induction hypothesis, we have that $\text{treewidth}(\text{CL}(G_i'^o, \mathcal{S}_{u_i} \cup \{\{u_i\} \cup V_1(G_i'^o)\})) \leq r + 2$ and, as $S_i = \{u_i\} \cup V_1(G_i'^o)$, we obtain that $\text{treewidth}(\text{CL}(G_i'^o, \{S_i\})) \leq r + 2, i = 1, \dots, t$. Finally, since G_i^o is a subgraph of $G_i'^o$ we conclude that $\text{treewidth}(\text{CL}(G_i^o, \{S_i\})) \leq r + 2$.

We set $H_i = \text{CL}(G_i^o, \{S_i\}), i = 1, \dots, t$, and $H' = \text{CL}(G^o, \mathcal{S}_v \cup \{\{v\} \cup V_1\})$. For $i = 1, \dots, t$, we make H_i a terminal graph by taking S_i as set of terminals. If now we successively take $S_i, i = 1, \dots, t$ as set of terminals in H and apply repeatedly Lemma 20 as in the end of the proof of Lemma 31, we conclude that the graph $\text{CL}(G, \mathcal{S}_v \cup \{\{v\} \cup V_1\})$ has also $\text{treewidth} \leq r + 2$. \square

Lemma 33 *If $G \in \mathcal{D}_r$ and G is planar, then $\text{treewidth}(\text{CL}(G)) \leq r + 2$.*

Proof: We have already examined the cases where $r = 2, 3$. For the case where $r \geq 4$, the theorem holds because of Lemma 32 as, for any planar graph $G \in \mathcal{D}_r$ there is a graph $G' \in \text{Cl}(\mathcal{P}_r)$ such that $\text{CL}(G)$ is a subgraph of G' . \square

Now, we can use the proof of Theorem 21, but with the result of Lemma 33 instead of the one of Lemma 19, and obtain the following result.

Theorem 34 *If G is planar, then either G contains $K_{2,r}$ as a minor or the treewidth of G is at most $r + 2$.*

Corollary 35 (i) *Every planar graph in k -IRS has treewidth at most $2r + 3$.*
(ii) *Every planar graph in k -SIRS has treewidth at most $2r + 3$.*

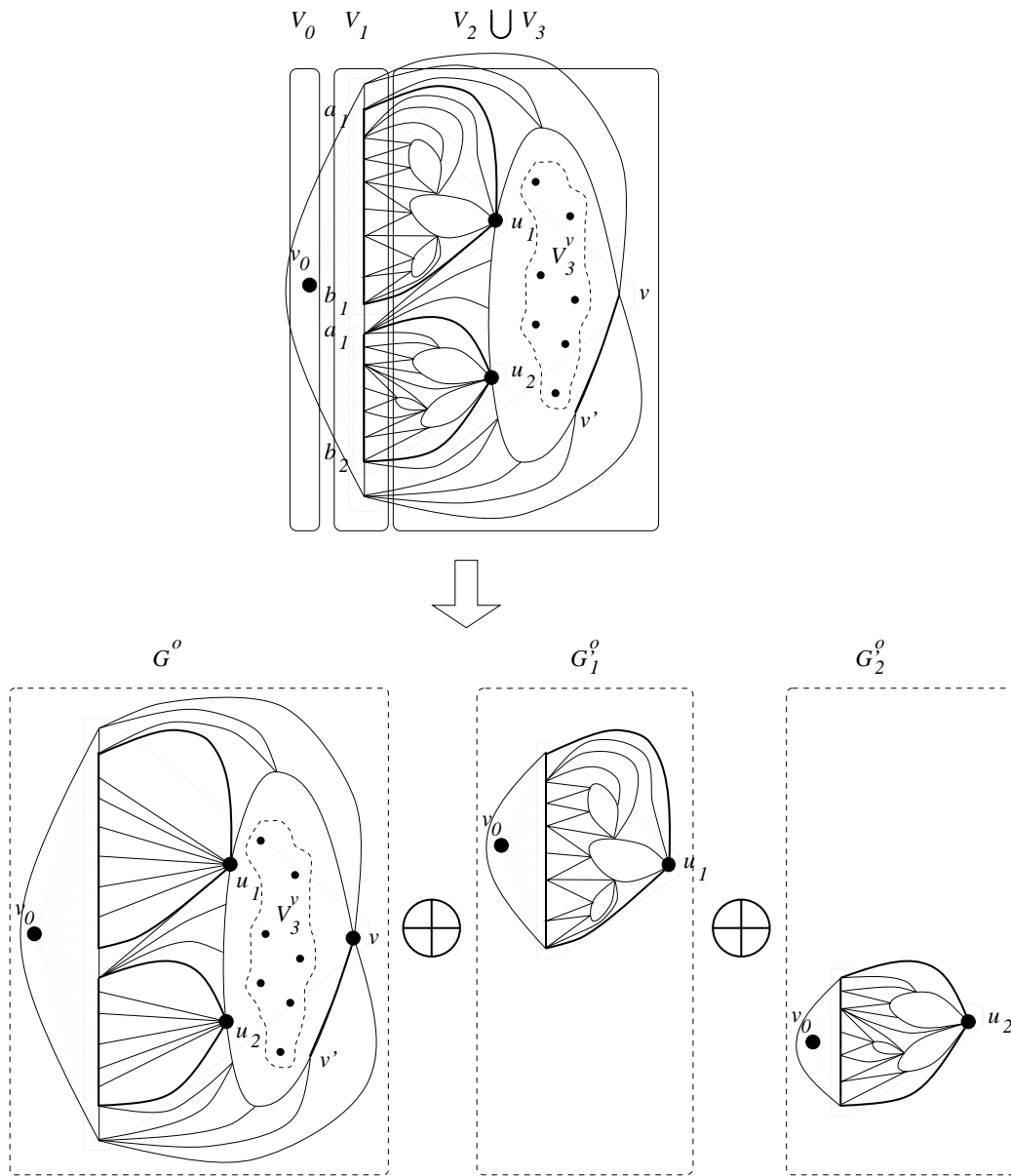


Figure 9: Case (ii) of the proof of Lemma 32.

6 Conclusions

In this paper, we made a perhaps somewhat surprising and interesting connection between the theory of compact routing schemes, and the theory of graph minors and treewidth of graphs. Several angles of this connection are still left unexplored.

As main open problems, we like to mention several issues that deal with constructivity. Is it possible to *construct* linear time algorithms that test whether a given graph belongs to k -IRS or one of its variants, for a fixed k ? In several other cases, a non-constructive proof of a linear or small degree polynomial time bound was only the first step towards a fully constructive solution (e.g., [5]). Will our Corollary 10 also be such a first step? But even if we know that a graph belongs to k -IRS (or a related class), we do not have a corresponding node labelling. How much time does it cost to construct such a node labelling? And, given a node labelling, how much time does it cost to verify that it has a k -label IRS (or variant) for every edge cost assignment? More related open problems are mentioned e.g. in [19].

References

- [1] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *J. Algorithms*, 12:308–340, 1991.
- [2] E. M. Bakker, J. van Leeuwen, and R. B. Tan. Work in progress.
- [3] E. M. Bakker, J. van Leeuwen, and R. B. Tan. Linear interval routing schemes. *Algorithms Review*, 2:45–61, 1991.
- [4] D. Bienstock, N. Robertson, P. D. Seymour, and R. Thomas. Quickly excluding a forest. *J. Comb. Theory Series B*, 52:274–283, 1991.
- [5] H. L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *Proceedings of the 25th Annual Symposium on Theory of Computing*, pages 226–234, New York, 1993. ACM Press. To appear in: *SIAM J. Comput.*, 1996.
- [6] H. L. Bodlaender. On linear time minor tests with depth first search. *J. Algorithms*, 14:1–23, 1993.
- [7] H. L. Bodlaender. On disjoint cycles. *Int. J. Found. Computer Science*, 5(1):59–68, 1994.
- [8] H. L. Bodlaender and R. H. Möhring. The pathwidth and treewidth of cographs. *SIAM J. Disc. Meth.*, 6:181–188, 1993.
- [9] B. Bollobas. *Random Graphs*. Academic Press, London, 1985.

- [10] R. B. Borie, R. G. Parker, and C. A. Tovey. Automatic generation of linear-time algorithms from predicate calculus descriptions of problems on recursively constructed graph families. *Algorithmica*, 7:555–581, 1992.
- [11] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, Mass., USA, 1989.
- [12] B. Courcelle. The monadic second-order logic of graphs I: Recognizable sets of finite graphs. *Information and Computation*, 85:12–75, 1990.
- [13] M. R. Fellows and M. A. Langston. Nonconstructive tools for proving polynomial-time decidability. *J. ACM*, 35:727–739, 1988.
- [14] M. R. Fellows and M. A. Langston. On search, decision and the efficiency of polynomial-time algorithms. *J. Comp. Syst. Sc.*, 49:769–779, 1994.
- [15] G. N. Frederickson and R. Janardan. Designing networks with compact routing tables. *Algorithmica*, 3:171–190, 1988.
- [16] Inmos. *The T9000 Transputer Products Overview Manual*, 1991.
- [17] T. Kloks. *Treewidth. Computations and Approximations*. Lecture Notes in Computer Science. Springer Verlag, 1994.
- [18] J. van Leeuwen and R. B. Tan. Computer networks with compact routing tables. In G. Rozenberg and A. Salomaa, editors, *The Book of L*, pages 298–307. Springer-Verlag, Berlin, 1986.
- [19] J. van Leeuwen and R. B. Tan. Compact routing methods: A survey. Technical Report UU-CS-1995-05, Department of Computer Science, Utrecht University, Utrecht, 1995.
- [20] N. Robertson and P. D. Seymour. Graph minors — a survey. In I. Anderson, editor, *Surveys in Combinatorics*, pages 153–171. Cambridge Univ. Press, 1985.
- [21] N. Robertson and P. D. Seymour. Graph minors. II. Algorithmic aspects of tree-width. *J. Algorithms*, 7:309–322, 1986.
- [22] N. Robertson and P. D. Seymour. Graph minors. XIII. The disjoint paths problem. *J. Comb. Theory Series B*, 63:65–110, 1995.
- [23] N. Robertson, P. D. Seymour, and R. Thomas. Quickly excluding a planar graph. *J. Comb. Theory Series B*, 62:323–348, 1994.
- [24] N. Santoro and R. Khatib. Labelling and implicit routing in networks. *Computer Journal*, 28:5–8, 1985.