11th International Symposium on Algorithms and Computation ISAAC 2000, Springer Verlag, Lecture Notes on Computer Science, Vol. 1969, pp. 192–203, 2000.

Constructive linear time algorithms for small carving-width

Dimitrios M. Thilikos[1*], Maria J. Serna[1], and Hans L. Bodlaender[2]

[1] Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Campus Nord – Mòdul C5, c/Jordi Girona Salgado, 1-3. E-08034, Barcelona, Spain
{mjserna,sedthilk}@lsi.upc.es

[2] Department of Computer Science, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands
hansb@cs.uu.nl

Abstract. Consider the following problem: For any constant k and any input graph G, check whether there exists a tree Y with internal vertices of degree 3 and a bijection θ mapping the vertices of G to the leaves of Y such that for any edge of Y, the number of edges of G whose endpoints have preimages in different components of $Y - e$, is bounded by k. This problem is known as the Minimum Routing Tree Congestion problem and is relevant to the design of minimum congestion telephone networks. Recent results of the Graph Minor series of Robertson and Seymour imply (non-constructively) that this problem is fixed parameter tractable. In this paper we give a *constructive proof* of this fact. Moreover, the algorithms of our proof are optimal and able to output the corresponding pair (Y, θ) in case of an affirmative answer.

1 Introduction

Let G be a graph where the existence of an edge in G represents a communication demand (i.e. telephone calls) between its endpoints. A *call routing tree* (or a *carving*) of a graph G is a tree T with internal vertices of degree 3 whose leaves correspond to the vertices of G. We say that T has *congestion* $\leq k$ if, for any edge e of T, the communication demands that need to be routed through e or, more explicitly, the number of edges of G that share endpoints corresponding to different connected components of $T - e$, is bounded by k (we denote as $T - e$ the graph obtained from T after the removal of e). The *carving-width* of a graph G is the minimum k for which there exists a call routing tree T with congestion bounded by k. In [9], Seymour and Thomas proved that computing the carving-width of a graph is NP-complete. Moreover, in the same paper, they give a $O(n^4)$ algorithm computing the carving-with of any planar graph G. Finally, the problem of designing call routing trees of minimum congestion has been studied in [6] where a polynomial time algorithm is given, computing a call routing tree T whose congestion is within a $O(\log n)$ factor from the optimal. It is easy to check that the class of graphs with carving-width bounded by k is immersion-closed for any k. Therefore, the results of Robertson and Seymour in [7, 8, 5] guarantee the existence, for any k, of a polynomial time algorithm deciding whether an input graph G has carving-width at most k. However, no such algorithm has been constructed so far even for small values of k. In this paper, we provide, for any $k \geq 1$, a linear time algorithm that checks whether an input graph G has carving-width$\leq k$ and, if this is the case, outputs a call routing tree of minimum congestion.

* The research of the first author was supported by the Ministry of Education and Culture of Spain, Grant number MEC-DGES SB98 0K148809.

A consequence of our algorithms is that, for any k, there exists an algorithm able to determine the immersion obstruction set for the class of the graphs with carving-width at most k.

The key tool in our algorithm is the notion of a *set of characteristics* used so far for the construction of a fixed parameter algorithm for minor closed parameters such as treewidth and pathwidth [2, 1], agile graph searching parameters [4], linear-width [4], and branch-width [3] and, lastly, for the immersion closed parameter of cutwidth see [10]. In a few words, a characteristic serves to filter the main data structure of a parameter to its essential part, a part that is able to be constructed from node to node of a tree decomposition. Moreover, as we will see, the information encoded by a characteristic depends on the width of the tree decomposition and, therefore, it is constant for graphs with bounded treewidth.

Our algorithm starts with an adequate bounded width tree decomposition of the input graph G. The tree decomposition allows the definition of an appropriate sequence of subgraphs. The algorithm computes, in a bottom-up fashion, a set of characteristics that "represent" the carvings that have carving-width $\leq k$ for any of the subgraphs.

2 Definitions and preliminary results

All the graphs of this paper are finite, undirected, and without loops or multiple edges (our results can be straightforwardly generalized in the case where the last restriction is altered). We will denote as $V(G)$ ($E(G)$) the vertex (edge) set of a graph G. Given two graphs G_1 and G_2 we denote $G_1 \cup G_2 = (V(G_1) \cup V(G_2), E(G_1) \cup E(G_2))$ and $G_1 \cap G_2 = (V(G_1) \cap V(G_2), E(G_1) \cap E(G_2))$.

We proceed with a number of definitions and notations, dealing with finite sequences (i.e., ordered sets) of a given finite set $\mathcal{O}$, for our purposes, $\mathcal{O}$ can be a set of numbers, sequences of numbers, vertices, or vertex sets. Let ω be a sequence of elements from $\mathcal{O}$. We use the notation $[\omega_1, \ldots, \omega_r]$ to represent ω and we define $\omega[i, j]$ as the subsequence $[\omega_i, \ldots, \omega_j]$ of ω (in case $j < i$, the result is the empty subsequence $[\,]$). We use the notation ω^r to denote the inverse of ω, i.e. sequence $[\omega_r, \ldots, \omega_1]$. We also denote as $\omega(i)$ the element of ω indexed by i.

Given a set S containing elements of $\mathcal{O}$, we denote as $\omega[S]$ the subsequence of ω that contains only the elements of ω that are in S. Given two sequences ω^1, ω^2, defined on $\mathcal{O}$, where $\omega^i = [\omega^i_1, \ldots, \omega^i_{r_i}], i = 1, 2$ we define the *concatenation* of ω_1 and ω_2 as $\omega^1 \oplus \omega^2 = [\omega^1_1, \ldots, \omega^1_{r_1}, \omega^2_1, \ldots, \omega^2_{r_2}]$. If n is a non-negative integer, we set $n \times \omega = \overbrace{\omega \oplus \cdots \oplus \omega}^{n}$. Unless mentioned otherwise, we will always consider that the first element of a sequence ω is indexed by 1, i.e. $\omega = \omega[1, |\omega|]$.

Let G be a graph and $S \subseteq V(G)$. We call the graph $(S, E(G) \cap \{\{x, y\} \mid x, y \in S\})$ *subgraph of G induced by S* and we denote it by $G[S]$. For any $e \in E(G)$, we set $G - e = (V(G), E(G) - \{e\})$ and for any $N \subseteq V(G)$ and $u \notin V(G)$, the graph $G \overset{u}{+} N$ is obtained by adding the new vertex u and the edges $\{\{u, v\} \mid v \in N\}$. We denote by $E_G(S)$ the set of edges of G that have an endpoint in S; we also set $E_G(v) = E_G(\{v\})$ for any vertex v. If $E \subseteq E(G)$ then we denote as $V(E)$ the set of all the endpoints of the edges in E i.e. we set $V(E) = \cup_{e \in E} e$. The neighborhood of a vertex v in graph G is the set of vertices in G that are adjacent, in G, with v and we denote it as $N_G(v)$, i.e. $N_G(v) = V(E_G(v)) - \{v\}$. We call a path $(t_1, \ldots, t_r)$ of a graph G *poor* if $r \geq 2$, its internal vertices (if exist) have degree 2, and its first and last vertices have degree different than 2, i.e. if $\forall_{1<i<r}\ |N_G(t_i)| = 2$ and $d_G(t_1) = d_G(t_r) \neq 2$ (notice that

any edge with both endpoints of degree different than 2 is a poor path). Observe that two poor paths of a graph have never common internal vertices. Finally, if G is not a cycle, then the edges of its poor paths for a partition of $E(G)$.

Given a tree T, we will denote as $A(T)$ the set of leaves of T, i.e. the vertices of T that have degree 1. Given two vertices v_1, v_2 of T we will denote as $P_T(v_1, v_2)$ the unique path of T connecting v_1 and v_2 and as $T(v_1, v_2)$ the subtree of T created if, for $i = 1, 2$, we remove all the connected components of $T - v_i$ that do not contain v_{3-i}. We will use the term *spine* for the path $P_T(v_1, v_2)$ of $T(v_1, v_2)$. Finally, if T is a tree $v \in V(T)$ and $S \subset N_T(v)$ where $|S| = |N_T(v)| - 1$, we define as $T(v, S)$ as the subtree of T created if we remove all the connected components of $T - v$ that contain vertices in S.

Finally, whenever we deal with a function $\varphi : A \to B$ we will use the notation $\varphi(S)$ to denote the set $\{\varphi(\eta) \mid \eta \in S\}$ for any $S \subseteq A$. Also, if S is a subset of B we will use the notation $\varphi^{-1}(S) = \{\eta \mid \varphi(\eta) \in S\}$. Finally if $\langle a, b\rangle$ is a directed pair of elements of A we denote $\varphi(\langle a, b\rangle) = \langle \varphi(a), \varphi(b)\rangle$.

2.1 Treewidth

A *tree decomposition* of a graph G is a pair (X, U) where U is a tree whose vertices we will call *nodes* and $X = (\{X_i \mid i \in V(U)\})$ is a collection of subsets of $V(G)$ such that

1. $\bigcup_{i \in V(U)} X_i = V(G)$,
2. for each edge $\{v, w\} \in E(G)$, there is an $i \in V(I)$ such that $v, w \in X_i$, and
3. for each $v \in V(G)$ the set of nodes $\{i \mid v \in X_i\}$ forms a subtree of U.

The *width* of a tree decomposition $(\{X_i \mid i \in V(U)\}, U)$ equals $\max_{i \in V(U)}\{|X_i| - 1\}$. The *treewidth* of a graph G is the minimum width over all tree decompositions of G.

A *rooted* tree decomposition is a triple $D = (X, U, r)$ in which U is a tree rooted at r and (X, U) is a tree decomposition.

Let $D = (X, U, r)$ be a rooted tree decomposition of a graph G. For each node i of T, let U_i be the subtree of U, rooted at node i. We set $V_i = \cup_{v \in V(U_i)} X_v$ and let $G_i = G[V_i]$. Notice that if r is the root of U, then $G_r = G$. We call G_i the subgraph of G *rooted* at i. We finally set, for any $i \in V(U)$, $D_i = (X^i, U_i)$ where $X^i = \{X_v \mid v \in V(U_i)\}$. Observe that for each node $i \in V(U)$, D_i is a tree decomposition of G_i.

Let $D = (X, U, r)$ be a rooted tree decomposition of a graph G where $X = \{X_i \mid i \in V(U)\}$. D is called a *nice* tree decomposition if the following are satisfied

1. Every node of U has at most two children,
2. if a node i has two children j, h then $X_i = X_j = X_h$,
3. if a node i has one child, then either $|X_i| = |X_j| + 1$ and $X_j \subset X_i$ or $|X_i| = |X_j| - 1$ and $X_i \subset X_j$.

Notice that a nice tree decomposition is always a rooted tree decomposition. For the following, see e.g. [1].

Lemma 1. *For any constant $k \geq 1$, given a tree decomposition of a graph G of width $\leq k$ and $O(|V(G)|)$ nodes, there exists an algorithm that, in $O(|V(G)|)$ time, constructs a nice tree decomposition of G of width $\leq k$ and with at most $4|V(G)|$ nodes.*

We now observe that a nice tree decomposition $(\{X_i \mid i \in V(U)\}, U)$ contains nodes of the following four possible types. A node $i \in V(U)$ is called "*start*" if $i \in A(U)$, "*join*" if i has two children, "*forget*" if i has only one child j and $|X_i| < |X_j|$, "*introduce*" if i has only one child j and $|X_i| > |X_j|$. We may also assume that if i is a *start* node then $|X_i| = 2$: the effect of *start* nodes with $|X_i| > 2$ can be obtained by using a *start* node with a set containing 2 vertices, and then $|X_i| - 2$ *introduce* nodes, which add all the other vertices.

2.2 Carving-width

A *carving* of a graph G is a pair $\gamma = (Y, \theta)$ where Y is a tree with internal vertices of degree 3 and $|V(G)|$ leaves and $\theta : A(Y) \to V(G)$ is a bijection mapping the leaves of Y to the vertices of G.

Given a subset S of $A(Y)$, we define function $\alpha_{G,\gamma,S} : E(Y) \to V(G)$ such that $\alpha_{G,\gamma,S}(e) = E_G(V^{(1)}) \cap E_G(V^{(2)})$ where $V^{(i)} = \theta(S \cap A(Y^{(i)})), i = 1, 2$ and $Y^{(i)}, i = 1, 2$ are the connected components of $Y - e$. Moreover, we define the function $d_{G,\gamma}$ mapping any directed edge $\boldsymbol{e} = \langle t_1, t_2 \rangle \in \boldsymbol{E}(Y) = \{\langle t_1, t_2 \rangle \mid \{t_1, t_2\} \in E(Y)\}$ to the one element typical sequence $[|\alpha_{G,\gamma,A(Y)}(e)|]$, i.e. $d_{G,\gamma}(\boldsymbol{e}) = [|\alpha_{G,\gamma,A(Y)}(e)|]$. Notice that $d_{G,\gamma}(\langle t_1, t_2 \rangle) = d_{G,\gamma}(\langle t_2, t_1 \rangle)$. We insist on this somehow overloaded notation for reasons for consistency with terminology that will be introduced later.

The width of a carving $\gamma = (Y, \theta)$ of G is defined as $\max\{\max(d_{G,\gamma}(e)) \mid e \in E(Y)\}$ (i.e. the maximum number appearing in the sequences corresponding to the edges of Y). The carving-width of a graph is the minimum width over all of its carvings.

Lemma 2. *For any graph*[1] *G*, $\text{treewidth}(G) \leq 3 \cdot \text{carving-width}(G)$.

Proof. Let $\gamma = (Y, \theta)$ be a carving of a graph G where $\forall_{e \in E(Y)} \ \max(d_{G,\gamma}(e)) \leq k$. We construct a tree-decomposition $D = (\{X_t, t \in V(t)\}, Y')$ of G as follows. Let $Y' = Y[V(Y) - A(Y)]$ and, for any vertex $t \in Y'$ we set $X_t = V(\alpha_{G,\gamma,V(G)}(e_1) \cup \alpha_{G,\gamma,V(G)}(e_2) \cup \alpha_{G,\gamma,V(G)}(e_3))$ where e_1, e_2, e_3 are the edges of Y that share t as a common endpoint. Notice that any vertex that appears in X_t will belong to at least two of the sets $V(\alpha_{G,\gamma}(e_i)), i = 1, 2, 3$ and therefore $|X_t| \leq 3k$. It is easy to see that D is a tree decomposition of width $\leq 3k$. □

2.3 Sequences of integers

We denote as $\mathcal{S}$ the set of all the sequences of non-negative integers. For any sequence $A = [a_1, \ldots, a_{|A|}] \in \mathcal{S}$ and any integer $t \geq 0$ we set $A + t = [a_1 + t, \ldots, a_{|A|} + t]$. If $A, B \in \mathcal{S}$ and $A = [a_1, \ldots, a_{|A|}]$ we say that $A \sqsubseteq B$ if B is a subsequence of A obtained after applying a number of times (possibly none) the following operations

(i) If for some i, $1 \leq i \leq |A| - 1$ $a_i = a_{i+1}$, then set $A \leftarrow A(1, i) \oplus A(i + 2, |A|)$.
(ii) If the sequence contains two elements a_i and a_j such that $j - i \geq 2$ and $\forall_{i<k<j} \ a_i \leq a_k \leq a_j$ or $\forall_{i<k<j} \ a_i \geq a_k \geq a_j$, then set $A \leftarrow A(1, i) \oplus A(j, |A|)$.

[1] Actually, we are able to prove the more tight relation $\frac{1}{\Delta(G)}\text{carving-width}(G) - 1 \leq \text{treewidth}(G) \leq 2\cdot\text{carving-width}(G)$ where $\Delta(G)$ is the maximum degree of G. However, the relation of Lemma 2 is sufficient for the purposes of our paper.

We define $\tau(A)$ as the unique minimum length element of the set $\{B \mid B \sqsubseteq A\}$.

For example, $\tau([\mathbf{5},5,6,7,7,\mathbf{7},4,4,\mathbf{3},5,4,6,\mathbf{8},\mathbf{2},\mathbf{9},3,4,6,7,\mathbf{2},\mathbf{7},5,4,4,6,\mathbf{4}]) = [5,7,3,8,2,9,2,7,4]$. We call a sequence A *typical* if $A \in \mathcal{S}$ and $\tau(A) = A$ (i.e. if none of (i) or (ii) can be applied any more).

The following results has been proved in [2] (Lemma 3.5 and Lemma 3.3 respectively).

Lemma 3. *The number of different typical sequences consisting of integers in* $\{0, 1, \ldots, n\}$ *is at most* $\frac{8}{3}2^{2n}$.

Notice that $B = \tau(A)$ is a subsequence $[a_{i_1}, \ldots, a_{i_{|B|}}]$ of $A = [a_1, \ldots, a_{|A|}]$ such that for any j, $1 \le j \le |B| - 1$ either $a_{i_j} \le a_{i_j+1} \le \cdots \le a_{i_{j+1}-1} \le a_{i_{j+1}}$ or $a_{i_j} \ge a_{i_j+1} \le \cdots \ge a_{i_{j+1}-1} \ge a_{i_{j+1}}$. We can now define a function $\beta_A : \{1, \ldots, |\tau(A)|\} \to \{1, \ldots, |A|\}$ where $\beta_A(j) = i_j$ is one of the possible original positions in A of the j-th element in $\tau(A)$. Consider the sequence of the previous example

$$A = [\mathbf{5}, 5, 6, \underline{7, 7, \mathbf{7}, 7}, 4, 4, \mathbf{3}, 5, 4, 6, \mathbf{8}, \mathbf{2}, \mathbf{9}, 3, 4, 6, 7, \mathbf{2}, \mathbf{7}, 5, 4, 4, 6, \mathbf{4}],$$

then we have

$$\begin{array}{lll} \beta_A(1) = 1, & \beta_A(2) = 6 \text{ (or 4 or 5 or 7)}, & \beta_A(3) = 10, \\ \beta_A(4) = 14, & \beta_A(5) = 15, & \beta_A(6) = 16, \\ \beta_A(7) = 21, & \beta_A(8) = 22, & \beta_A(9) = 27. \end{array}$$

Given two typical sequences A, B and an integer j, $1 \le 1 \le |\tau(A \oplus B)|$, we define

$$\alpha(A, B, j) = \begin{cases} (0, \beta_{A \oplus B}(j)) & \text{if } \beta_{A \oplus B}(j) \le |A| \\ (1, \beta_{A \oplus B}(j) - |A|) & \text{otherwise} \end{cases}$$

As an example we have that if $A = [1, 3, 2]$ and $B = [8, 5, 9]$, we have that $\tau(A \oplus B) = [1, 9]$, $\alpha(A, B, 1) = (0, 1)$, and $\alpha(A, B, 2) = (1, 3)$

For any $A \in \mathcal{S}$ we define $\eta(A)$ in the same way as $\tau(A)$ with the difference that only operation (i) is considered. If now A is a sequence, we define the *set of extensions* of A as

$$\mathcal{E}(A) = \{\tilde{A} \in \mathcal{S} \mid \eta(\tilde{A}) = A\}.$$

Let $A = [a_1, \ldots, a_{r_1}]$ and $B = [b_1, \ldots, b_{r_2}]$ be two sequences in $\mathcal{S}$. We say that $A \le B$ if $r_1 = r_2$ and $\forall_{1 \le i \le r_1}\ a_i \le b_i$. In general, we say that $A \prec B$ if there exist extensions $\tilde{A} \in \mathcal{E}(A)$, and $\tilde{B} \in \mathcal{E}(B)$ such that $\tilde{A} \le \tilde{B}$. For example if $A = [1, 7, 2, 6, 4]$ and $B = [5, 7, 3, 8]$ then $A \prec B$ because $\tilde{B} = [5,7,7,7,4,8,8,8,8]$ is an extension of B, $\tilde{A} = [1,7,2,6,4,4,4,4,4]$ is an extension of A, and $\tilde{A} \le \tilde{B}$.

The following three lemmata are easy consequences of the definitions.

Lemma 4. *Given* $R \in \mathcal{S}$, *if we set* $A = \tau(R)$ *then, for any* $m, 1 \le m \le |A|$, *there exists a* i, $1 \le i \le |R|$ *such that* $A[1, m] = \tau(R[1, i])$ *and* $A[m, |R|] = \tau(R[i, |A|])$.

Lemma 5. *Let* A_1, A_2 *be two typical sequences where* $A_2 \prec A_1$. *Then, for any* $m_1, 1 \le m_1 \le |A_1|$, *there exists a* $m_2, 1 \le m_2 \le |A_2|$ *such that* $A_2[1, m_2] \prec A_1[1, m_1]$ *and* $A_2[m_2, |A_2|] \prec A_1[m_1, |A_1|]$.

Lemma 6. *Given* $R \in \mathcal{S}$, *if we set* $A = \tau(R)$ *then, for any* $r, 1 \le r \le |R|$, *there exists an integer* $i, 1 \le i \le |A|$ *such that* $A[1, i] \prec \tau(R[1, r])$ *and* $A[i, |R|] \prec \tau(R[r, |A|])$.

As an example of Lemma 6 we consider the sequences

$$R = [\mathbf{2}, 6, 7, \mathbf{8}, 5, 4, 3, 5, \mathbf{2}, 4, \mathbf{6}, 4, \mathbf{4}] \text{ and } A = \tau(R) = [2, 8, 2, 6, 4].$$

If we choose $r = 7$ we have that $j = 2$, $k = 4$, and $l = 9$. Notice that,

$$\begin{aligned}
[2, 8] &= \tau([1, 6, 7, 8]), \\
[8, 2] &\prec \tau([8, 5, 4, 3]), \\
[2] &\prec \tau([3, 5, 2]), \\
[2, 6, 4] &= \tau[2, 4, 6, 4, 4], \\
[2, 8, 2] &\prec \tau([1, 6, 7, 8, 5, 4, 3]) \text{ and} \\
[2, 6, 4] &\prec \tau([3, 5, 2, 4, 6, 4, 4]).
\end{aligned}$$

Let two sequences A, B of $\mathcal{S}$ where $A = [a_1, \ldots, a_r]$, $B = [b_1, \ldots, b_r]$. We define $A+B = [a_1+b_1, \ldots, a_r+b_r]$ and we say that $A \sim B$ iff $\forall_{1 \le i < r}\ a_i \neq a_{i+1} \Leftrightarrow b_i = b_{i+1}$ (and, therefore, $b_i \neq b_{i+1} \Leftrightarrow a_i = a_{i+1}$). As an example we mention that

$$[1, 1, 8, 5, 5, 6, 7] \sim [3, 6, 6, 6, 9, 9, 9].$$

The *interleaving* $A \otimes B$ of two typical sequences A and B is a set of typical sequences defined as follows

$$A \otimes B = \{\tau(\tilde{A} + \tilde{B}) \mid \tilde{A} \in \mathcal{E}(A), \tilde{B} \in \mathcal{E}(B) \text{ and, } \tilde{A} \sim \tilde{B}\}.$$

Notice that the length of the resulting sequences is at most $|A| + |B| - 1$

For example if $A = [2, 9, 5]$ and $B = [3, 2, 6]$ we have that

$$\begin{aligned}
A \otimes B = \{&\tau([2, 9, 5, 5, 5] + [3, 3, 3, 2, 6]), \tau([2, 9, 9, 5, 5] + [3, 3, 2, 2, 6]), \\
&\tau([2, 9, 9, 9, 5] + [3, 3, 2, 6, 6]), \tau([2, 2, 9, 9, 5] + [3, 2, 2, 6, 6]), \\
&\tau([2, 2, 9, 5, 5] + [3, 2, 2, 2, 6]), \tau([2, 2, 2, 9, 5] + [3, 2, 6, 6, 6])\} \\
= \{&\tau([5, 12, 8, 7, 11]), \tau([5, 12, 11, 7, 11]), \tau([5, 12, 11, 15, 11]), \\
&\tau([5, 4, 11, 15, 11]), \tau([5, 4, 11, 7, 11]), \tau([5, 4, 8, 15, 11])\} \\
= \{&[5, 12, 7, 11], [5, 12, 7, 11], [5, 15, 11], [5, 4, 15, 11], [5, 4, 11]\}
\end{aligned}$$

We will need the following lemmata.

Lemma 7. *Let $B_i, C_i, i = 1, 2$ be sequences where $B_1 \sqsubseteq B_2$ and $C_1 \sqsubseteq C_2$ and $|B_i| = |C_i|, i = 1, 2$. Then $\tau(B_1 + C_1) = \tau(B_2 + C_2)$.*

Lemma 8. *Let $A_i, B_i, i = 1, 2$ be four typical sequences where $A_i \prec B_i, i = 1, 2$. Then $\tau(A_1 \oplus B_1) \prec \tau(A_2 \oplus B_2)$.*

Lemma 9. *Let A, B, C be sequences such that $|B| = |C|$ and $A = B + C$. Then there exists a sequence $A' \in \tau(B) \otimes \tau(C)$ such that $\tau(A') \prec \tau(A)$.*

Lemma 10. *Let A, B be two typical sequence and C a sequence such that $C \in A \otimes B$. Suppose also that A', B' be two typical sequence such that $A \prec A'$ and $B \prec B'$. Then there exists a sequence $C' \in A' \otimes B'$ such that $C \prec C'$.*

3 Characteristics and overview of the algorithm

In this section, we will describe the general structure of our algorithm along with their basic mathematical concepts. A key tool that has already been used in the bibliography for other parameters like pathwidth and treewidth in [2], linear-width in [4], branchwidth in [3], and cutwidth in [10] is the notion of a *set of characteristics*. In a few words, a characteristic serves as a mathematical tool that filters the data of the main structure of a parameter to its essential part, that is, the part able to reproduce it with respect to a node i of the tree decomposition. Moreover, as we will see, the information encoded by a characteristic depends on the width of this decomposition and, therefore, it is constant for graphs with bounded treewidth.

Our algorithms roughly work as follows. Given a tree decomposition of the input graph G, we transform it in a nice tree decomposition as it is indicated in Lemma 1. We correspond to any of the subgraphs G_p a set of characteristics that "represent" the carvings of G_p that have width $\leq k$. Our algorithms are based on a bottom up procedure that is able to compute a set of characteristics of G_p using the information of the set of characteristics corresponding to the children of p. The procedure starts from the leaf nodes of the tree decomposition. In the rest of this section we will specify these characteristics for the parameter of carving-width and we will demonstrate the main theorems supporting their use in our algorithms.

3.1 Characteristic pairs

We call *tree arrangement over* $\mathcal{O}$ any pair (Y, θ) where Y is a tree with all internal vertices of degree 3 and θ is a bijection mapping the leaves of Y to distinct elements of a set $\mathcal{O}$ (notice that if $\mathcal{O}$ is the vertex set of a graph G, then (Y, θ) is a carving of G).

Let $\gamma = (Y, \theta)$ be a tree arrangement over $\mathcal{O}$. If $e = \{t_1, t_2\}$ is any edge of Y and v is an element of $\mathcal{O} - \theta(A(Y))$ then we will denote as $\mathsf{Add}(\gamma, e, v)$ the tree arrangement $\gamma' = (Y', \theta')$ where Y' and θ' are defined as follows:

$$\begin{aligned} Y' = \ & (V(Y) \cup \{t, l\}, \\ & E(Y) \cup \{\{t_1, t\}, \{t_2, t\}, \{t_t, l\}\} - \{\{t_1, t_2\}\}) \\ & \text{(we assume that } t \text{ and } l \text{ are not vertices of } Y \text{ and that } e = \{t_1, t_2\}) \\ \text{and} \quad \theta' = \ & \theta \cup \{l, v\}. \end{aligned}$$

If $u \in \theta(A(Y))$ then we will denote as $\mathsf{Rem}(\gamma, u)$ the tree arrangement $\gamma' = (Y', \theta')$ where Y' and θ' are defined as follows:

$$\begin{aligned} Y' = \ & (V(Y) - \{l, t\}, \text{ where } l = \theta^{-1}(u), \\ & E(Y) \cup \{\{t_1, t_2\}\} - \{\{t, t_1\}, \{t, t_2\}\}) \\ \text{and} \quad \theta' = \ & \theta - \{l, \theta(l)\}. \\ & (t, \text{ is the unique vertex adjacent to } l \text{ and} \\ & \ t_1, \text{ and } t_2 \text{ are two vertices of } Y \text{ with distance 2 from } l) \end{aligned}$$

For an example of the operation of functions Rem and Add see Figure 1.

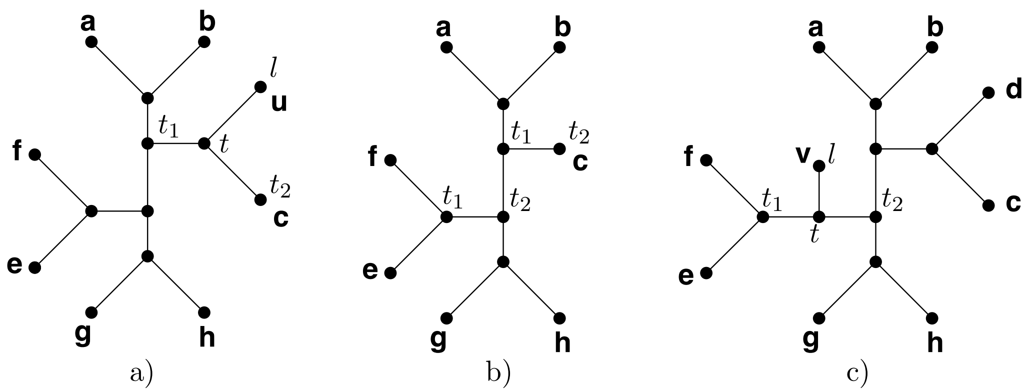

Fig. 1. a) The carving γ, b) The carving $\gamma' = \mathsf{Rem}(\gamma, \mathbf{u})$, c) The carving $\gamma'' = \mathsf{Add}(\gamma', \{t_1, t_2\}, \mathbf{v})$.

We call *characteristic pair* any pair (γ, δ) where $\gamma = (Y, \theta)$ as a tree arrangement over some set $\mathcal{O}$ and δ is a function mapping any directed edge $\boldsymbol{e} = \langle t_1, t_2 \rangle \in \boldsymbol{E}(Y)$ to some typical sequence such that $\delta(\langle t_1, t_2 \rangle) = (\delta(\langle t_2, t_1 \rangle))^r$. We stress out that, besides the fact that, in general, the edges of $E(Y)$ are denote as unordered pairs, in the case where an edge is viewed as the input of function δ, the output will be sensitive to the ordering that its endpoints are presented. For this reason, whenever we define the value of δ for some input $\langle t_1, t_2 \rangle$ we will automaticaly assume that the value $\delta(\langle t_2, t_1 \rangle)$ is defined to be $(\delta(\langle t_2, t_1 \rangle))^r$. As an example we mention that for the characteristic pair of Figure 2, $\delta(\langle l, t \rangle) = [6, 8, 2, 5]$ and $\delta(\langle t, l \rangle) = [5, 2, 8, 6]$. Finally, if $\boldsymbol{e} = \langle t_1, t_2 \rangle$ is a directed edge we will denote its unordered version by removing the " " symbol, i.e. $e = \{t_1, t_2\}$.

We define $\max(\gamma, \delta) = \max\{\max(\delta(\boldsymbol{e})) \mid \boldsymbol{e} \in \boldsymbol{E}(Y)\}$. Notice that for any graph G, any carving $\gamma = (Y, \theta)$ of $V(G)$ is a tree arrangement over $V(T)$ and the pair $(\gamma, d_{G,\gamma})$ is a characteristic pair.

The following procedure defines the *compression* of a characteristic pair relative to a subset S of $\mathcal{O}$ (for an example see Figure 2).

Procedure $\mathsf{Com}(\gamma, \delta, S)$.
Input: A characteristic pair $(\gamma = (Y, \theta), \delta)$ and a set S.
Output: A characteristic pair $(\hat{\gamma} = (\hat{Y}, \hat{\theta}), \hat{\delta})$.

1: Let $Y_{\text{trunk}} = Y[(V(Y) - A(Y)) \cup \theta^{-1}(S)]$, (i.e. Y_{trunk} is the tree obtained from Y after removing all the leaves that do not map through θ to vertices in S.)
2: While $A' = A(Y_{\text{trunk}}) - A(Y) \neq \emptyset$, set $Y_{\text{trunk}} = Y_{\text{trunk}}[V(Y_{\text{trunk}}) - A']$, (i.e. remove leaves of Y_{trunk} that are not leaves of Y as long as such leaves exist.)
3: Set $\hat{Y} = Y_{\text{trunk}}$ and $\hat{\delta} = \emptyset$.
4: Replace any maximal poor path $(t_1, \ldots, t_r), r \geq 2$ of Y_{trunk} with an edge $\{\hat{t}_1, \hat{t}_2\}$ and set $\hat{\delta} \leftarrow \hat{\delta} \cup \{(\langle \hat{t}_1, \hat{t}_2 \rangle, \tau(\delta(\langle t_1, t_2 \rangle) \oplus \cdots \oplus \delta(\langle t_{r-1}, t_r \rangle)))\}$.
5: Let $\gamma' = (Y', \theta')$, where θ' is the restriction of θ to the leaves of Y' (notice that $\theta'(A(Y')) = \theta(A(Y)) \cap S$).
6: Output (γ', δ').
7: End.

Notice that the result of the successive compressions of step **4** is independent of the order they are realized.

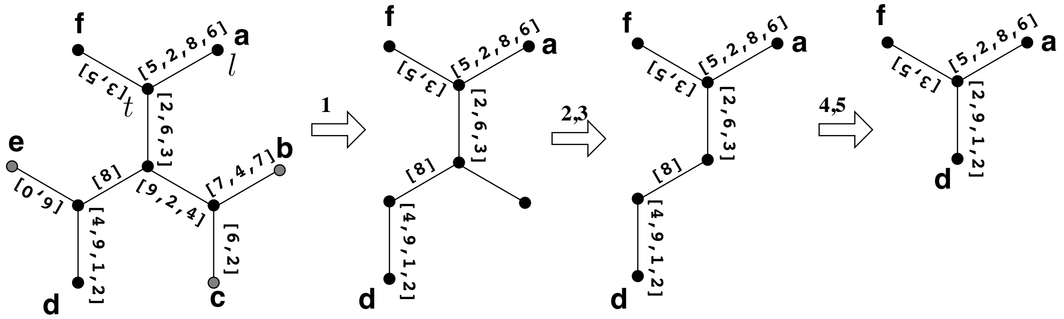

Fig. 2. An example of the compression of a characteristic pair relative to the set $\{\mathsf{f}, \mathsf{a}, \mathsf{d}\}$.

Given two characteristic pairs over $S \subseteq \mathcal{O}$ $(\hat{\gamma}_i, \hat{\delta}_i), i = 1, 2$ where $\hat{\gamma}_i = (\hat{Y}_i, \hat{\theta}_i), i = 1, 2$, we say that $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$ if $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$ is an isomorphic bijection between $\hat{Y}_1$ and $\hat{Y}_2$ whose restriction on $A(\hat{Y}_1)$ and $A(\hat{Y}_2)$ is the bijection $\hat{\theta}_1 \circ \hat{\theta}_2^{-1} : A(\hat{Y}_1) \to A(\hat{Y}_2)$. Suppose that $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$. We define the interleaving of $\hat{\delta}_1$ and $\hat{\delta}_2$ as follows.

$$\hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2 = \{\hat{\delta} \mid \forall_{\langle \hat{t}, \hat{t}' \rangle \in \boldsymbol{E}(\hat{Y}_1)} \ \hat{\delta}(\langle\!\langle \hat{t}, \hat{t}' \rangle\!\rangle) \in \hat{\delta}_1(\langle\!\langle \hat{t}, \hat{t}' \rangle\!\rangle) \otimes \hat{\delta}_2(\langle\!\langle \hat{\phi}(\hat{t}), \hat{\phi}(\hat{t}') \rangle\!\rangle)\}.$$

We also say that $(\hat{\gamma}_1, \hat{\delta}_1) \prec_{\hat{\phi}} (\hat{\gamma}_2, \hat{\delta}_2)$ when, for any $\langle \hat{t}, \hat{t}' \rangle \in \boldsymbol{E}(\hat{Y}_1)$, $\hat{\delta}_1(\langle\!\langle \hat{t}, \hat{t}' \rangle\!\rangle) \prec \hat{\delta}_2(\hat{\phi}(\langle\!\langle \hat{t}, \hat{t}' \rangle\!\rangle))$. Finally, $(\hat{\gamma}_1, \hat{\delta}_1) \prec (\hat{\gamma}_2, \hat{\delta}_2)$ if there exist an isomorphic bijection $\hat{\phi}$ such that $(\hat{\gamma}_1, \hat{\delta}_1) \prec_{\hat{\phi}} (\hat{\gamma}_2, \hat{\delta}_2)$.

The following lemma is a direct consequence of Lemma 10.

Lemma 11. *Let, for $i = 1, 2$, $(\hat{\gamma}_i, \hat{\delta}_i)$ and $(\hat{\gamma}_i^*, \hat{\delta}_i^*)$ be characteristic pairs and an isomorphism $\psi_i : V(\hat{Y}^*) \to V(\hat{Y})$ where $\hat{\gamma}_i \equiv_{\psi_i} \hat{\gamma}_i^*$ and $(\hat{\gamma}_i^*, \hat{\delta}_i^*) \prec_{\psi_i} (\hat{\gamma}_i, \hat{\delta}_i)$. Let also $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$ for some isomorphism $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$. Then for any $\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2$ there exists a characterisic pair $(\hat{\gamma}_1, \hat{\delta}'^*)$ and two isomorphisms $\hat{\phi}^* : V(\hat{Y}_1^*) \to V(\hat{Y}_2^*)$ and $\psi : V(\hat{Y}^*) \to V(\hat{Y})$ such that*

1. *$\hat{\gamma}_1^* \equiv_{\hat{\phi}^*} \hat{\gamma}_2^*$.*
2. *$\hat{\gamma}_1^* \equiv_{\psi} \hat{\gamma}_1$.*
3. *$\hat{\delta}'^* \in \hat{\delta}_1^* \bigotimes_{\hat{\phi}^*} \hat{\delta}_2^*$*
4. *$(\hat{\gamma}_1^*, \hat{\delta}'^*) \prec_{\psi} (\hat{\gamma}_1, \hat{\delta}')$.*

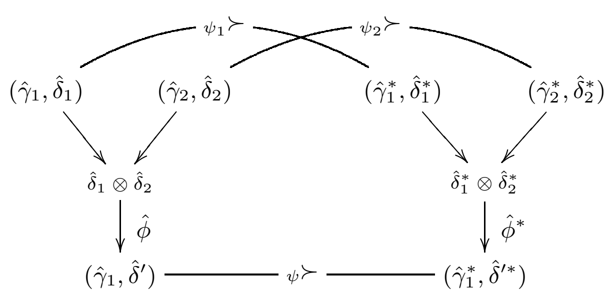

Fig. 3. The structure of Lemma 11.

3.2 Characteristic of a carving of a graph

Let us start by definig a *characteristic* of a carving of a graph.

Given a graph G, a carving $\gamma = (Y, \theta)$ of G and a set $S \subseteq V(G)$, the S-*characteristic* of γ is $C_S(G, \gamma) = \mathsf{Com}(\gamma, d_{G,\gamma}, S)$. Notice that, from the definition we have that the $V(G)$-characteristic of carving γ is equal to $(\gamma, d_{G,\gamma})$, i.e. $C_{V(G)}(G, \gamma) = (\gamma, d_{G,\gamma})$ (clearly, $\mathsf{Com}(\gamma, d_{G,\gamma}, V(G)) = (\gamma, d_{G,\gamma})$).

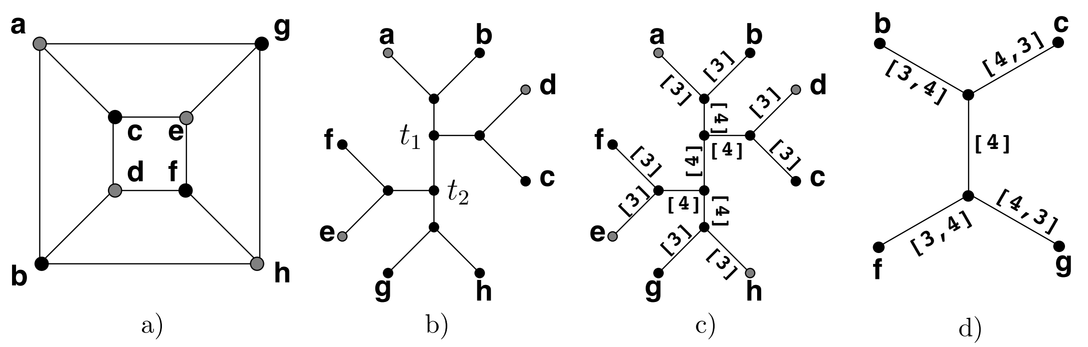

Fig. 4. a) A graph G, b) a carving γ of G, c) the characteristic $C_{V(G)}(G, \gamma)$, and d) the characteristic $C_{\{\mathbf{b},\mathbf{c},\mathbf{g},\mathbf{f}\}}(G, \gamma)$.

As an example, in Figure 4, we present $C_{V(G)}(G, \gamma)$ and $C_{\{b,c,g,f\}}(G, \gamma)$, for the carving γ of some graph G.

From now on we will denote any object refering to an S-characteritic with "hatted" symbols. We will use "unhatted" symbols exclusively in the cases where we refere to a $V(G)$-characteristic. Moreover we will keep on using unhatted symbols when we refer to characteristics pairs in general.

Notice that if $(\hat{\gamma}, \hat{\delta})$ is an S-characteristic of some carving γ of G, then $\hat{\gamma}$ is a carving of $G[S]$.

The following lemma is a direct consequence of Procedure $\mathsf{Com}(\gamma, \delta, S)$.

Lemma 12. *Let γ be a carving of a graph G, $S \subseteq V(G)$, and set $\delta = d_{G,\gamma}$. Let also $(\hat{\gamma}, \hat{\delta})$ be a characteristic pair on $V(G)$. Then $(\hat{\gamma}, \hat{\delta}) = \mathsf{Com}(\gamma, \delta, S)$ iff there exists a function $\chi : V(\hat{Y}) \to V(Y)$ such that*

1. *$\chi(A(\hat{Y})) = \theta^{-1}(S)$ (the leaves of $\hat{Y}$ map to the leaves of Y corresponding to the vertices of S).*
2. *The set $\{E(T(\chi(\hat{v}), \chi(\hat{v}))) \mid \{\hat{v}, \hat{u}\} \in E(\hat{Y})\}$ forms a partition of $E(Y)$.*
3. *$\forall_{\langle \hat{v}, \hat{u} \rangle \in \boldsymbol{E}(\hat{Y})}\ \hat{\delta}(\hat{\boldsymbol{e}}) = \tau(\delta(\langle\langle t_1, t_2 \rangle\rangle \oplus \cdots \oplus \delta(\langle\langle t_{\rho-1}, t_\rho \rangle\rangle))$ where $(t_1, \ldots, t_\rho) = P_Y(\chi(\hat{v}), \chi(\hat{u}))$.*

Given a carving γ of a graph G and a set $S \subseteq V(G)$ we call $\chi_{\gamma,G}$ tyhe function uniquely defined by the above lemma. For an example, see Figure 5.

3.3 Analysis of the algorithm

Using now Lemma 3 and Lemma 2 and working in a similar way as in the proof of Lemma 3.1 in [2], we can prove the following lemma.

Lemma 13. *Let $D = (X, U, r)$ be a nice tree decomposition of G, rooted at r, with width at most w. Let X_p be some node in X. The number of different X_p-characteristics of all possible carvings of G_p with width at most k, is bounded by a function depending only on k and w.*

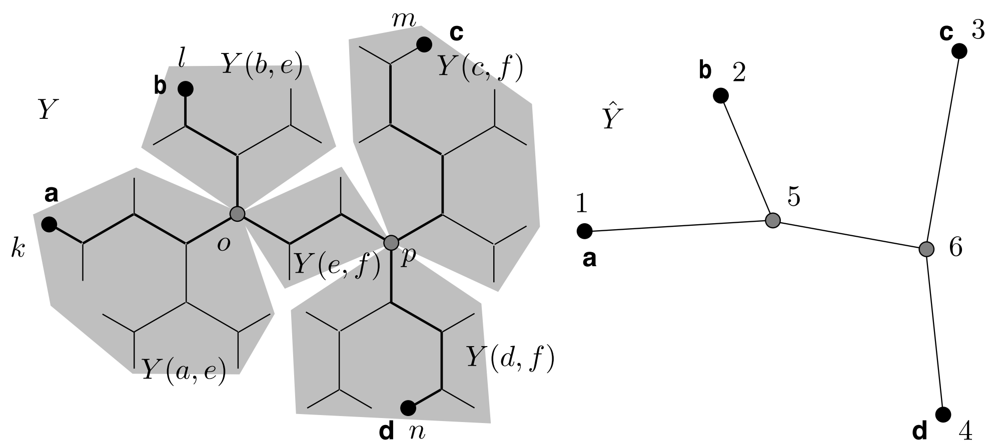

Fig. 5. Example of function $\chi_{\gamma,S}$ when $\gamma = (Y, \theta)$, $S = \{a, b, c, d\}$, and where $\chi(1) = k, \chi(2) = l, \chi(3) = m, \chi(4) = n, \chi(5) = o$, and $\chi(6) = p$.

Lemma 13 is important for the linear time of our algorithms as it ensures that the time cost of algorithmic steps concerning sets of characteristics is *independent* of the size of the graph G_p they refer to.

A set $FS(p)$ of X_p-characteristics of carvings of a graph G_p with width at most k is called a *full set of characteristics of carvings* for G_p if for each carving γ of G_p with width at most k, there is a carving γ' of G_p such that $C_{X_p}(G_p, \gamma') \prec C_{X_p}(G_p, \gamma)$ and $C_{X_p}(G_p, \gamma') \in FS(p)$, i.e. the X_p-characteristic of γ' is in $FS(p)$.

The following lemma can be derived directly from the definitions.

Lemma 14. *A full set of characteristics of carvings for G_p is non-empty if and only if the carving-width of G_p is at most k. If some full set of characteristics of carvings for G_p is non-empty, then any full set of characteristics of carvings for G_p is non-empty.*

An important consequence of Lemma 14 is that the carving-width of G is at most k, if and only if any full set of characteristics of carvings for $G_r = G$ is non-empty.

In what follows, it remains to show how to compute a full set of characteristics of carvings at a node i in $O(1)$ time, when a full set of characteristic of vertex carvings for the children of i is given. This will be demonstrated in section 4 and will make it possible, given any pair of integers constants k, w, to construct an algorithm that given a graph and a nice tree decomposition of G of width at most w, to decide whether G has carving-width at most k.

4 Basic subroutines for carving-width

4.1 A full set for a *start* node

Let X_p be a *start* node. Clearly, $V(G_p) = X_p = \{x, x'\}$ and

$$FS(p) = \{(((\{t, t'\}, \{\{t, t'\}\}), \{(t, x), (t', x')\}), \{(\langle t, t' \rangle, [|E(G[\{x, x'\}])|])\})\}.$$

4.2 A full set for an *introduce* node

We will now consider the case where X_p is an *introduce* node. Let q be the unique child of p in U. The following procedure is the basis to compute a characteistic after the insertion of an introduced vertex and the additional edges that appear in G_p.

Procedure $\mathsf{Ins}(G, u, S, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m)$.

Input: A graph G, a vertex $u \notin V(G)$, two sets S, N where $N \subseteq S \subseteq V(G)$, an S-characteristic $(\hat{\gamma}, \hat{\delta})$ of some carving γ of G where $\hat{\gamma} = (\hat{Y}, \hat{\theta})$ and $\gamma = (Y, \theta)$, a directed edge $\hat{\boldsymbol{e}} = \langle \hat{t}_{\text{left}}, \hat{t}_{\text{right}} \rangle \in \boldsymbol{E}(\hat{Y})$, and an integer $m, 1 \leq m \leq |\hat{\delta}(\hat{e})|$.

Output: An $(S \cup \{u\})$-characteristic $(\hat{\gamma}', \hat{\delta}')$ $(\hat{\gamma}' = (\hat{Y}', \hat{\theta}'))$ of the carving $\gamma' = \mathsf{Add}(\gamma, e, u)$ of $G' = G \overset{u}{+} N$ for some $e \in E(Y)$.

1: Let $\{\hat{t}_1, \ldots, \hat{t}_\sigma\} = \hat{\theta}^{-1}(N_G(u))$.

2: (insertion of u)

Set $(\hat{Y}', \hat{\theta}') = \mathsf{Add}(\hat{\gamma}, \hat{e}, u)$.

Use the notation $V(\hat{Y}') - V(\hat{Y}) = \{\hat{l}, \hat{t}\}$ where $\hat{l} \in A(\hat{Y}')$.

Set $\hat{\theta}' = \hat{\theta} \cup \{(\hat{l}, u)\}$, and

$$\hat{\delta}' = \hat{\delta} \cup \{(\langle \hat{t}_1, \hat{t} \rangle, \hat{\delta}(\hat{e})[1, m]), (\langle \hat{t}_2, \hat{t} \rangle, \hat{\delta}(\hat{e})[m, |\hat{\delta}(\hat{e})|]), (\langle \hat{t}, \hat{l} \rangle, [0])\} - \{(\hat{\boldsymbol{e}}, \hat{\delta}(\hat{\boldsymbol{e}}))\}.$$

3: (insertion of edges in $E_G(u)$)

For $h = 1, \ldots, \sigma$,

For any edge $\hat{f}$ in $P_{\hat{Y}'}(\hat{t}_h, \hat{l})$,

Set $\hat{\delta}'(\hat{f}) \leftarrow \hat{\delta}'(\hat{f}) + 1$.

4: Output $(\hat{\gamma}', \hat{\delta}')$ (where $\hat{\gamma}' = (\hat{Y}', \hat{\theta}')$).

5: End.

The following lemma is an easy consequence of Procedure Ins and function Add (as an example, compare Figures 4.c and 6.c).

Lemma 15. *Let G be a graph, $\gamma = (Y, \theta)$ be a carving of G, N a subset of $V(G)$ and $\boldsymbol{e} \in \boldsymbol{E}(Y)$. We set $\gamma' = \mathsf{Add}(\gamma, e, u)$ and we define G' as the graph obtained from G after introducing a new node u and connecting it to the vertices of N. Then $\mathsf{Ins}(G, u, V(G), N, \gamma, d_{G,\gamma}, \boldsymbol{e}, 1)$ is the $V(G')$-characteristic of γ', i.e $\mathsf{Ins}(G, u, V(G), N, \gamma, d_{G,\gamma}, \boldsymbol{e}, 1) = (\gamma', d_{G',\gamma'})$.*

Lemma 16. *Let G be a graph, $\gamma = (Y, \theta)$ a carving of G, and $S \subseteq V(G)$. We also set $G' = G \overset{u}{+} N$ for some $N \subseteq S$ and some new vertex $v \notin V(G)$ and we use the notation $\delta = d_{\gamma,G}$ and $(\hat{\gamma}, \hat{\delta}) = C_S(G, \gamma)$ where $\hat{\gamma} = (\hat{Y}, \hat{\theta})$. Then the following hold.*

(i) For any $\hat{\boldsymbol{e}} \in \boldsymbol{E}(\hat{Y})$ and any $m, 1 \leq m \leq |\hat{\delta}(\hat{\boldsymbol{e}})|$, there exists an edge $\boldsymbol{e} \in \boldsymbol{E}(Y)$ such that $\mathsf{Ins}(G, u, S, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) = \mathsf{Com}(\mathsf{Ins}(G, u, V(G), N, \gamma, \delta, \boldsymbol{e}, 1), S \cup \{u\})$.

(ii) For any $\boldsymbol{e} \in \boldsymbol{E}(Y)$ there exist an edge $\hat{\boldsymbol{e}} \in \boldsymbol{E}(\hat{Y})$ and an integer $m, 1 \leq m \leq |\hat{\delta}(\hat{\boldsymbol{e}})|$,such that $\mathsf{Ins}(G, u, S, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) \prec \mathsf{Com}(\mathsf{Ins}(G, u, V(G), N, \gamma, \delta, \boldsymbol{e}, 1), S \cup \{u\})$.

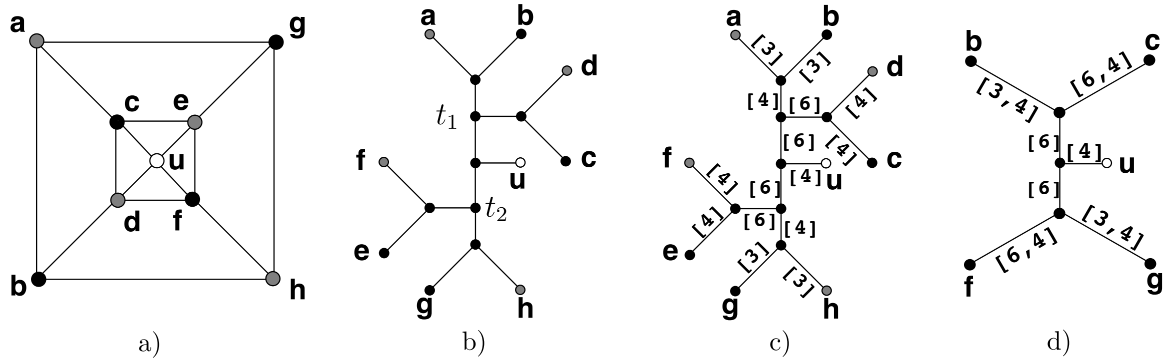

Fig. 6. a) The graph G', b) the carving $\gamma' = \mathsf{Add}(\gamma, \{t_1, t_2\}, \mathbf{u})$ of G', c) the characteristic $C_{V(G)}(G', \gamma')$, and d) the characteristic $C_{\{\mathbf{b},\mathbf{c},\mathbf{g},\mathbf{f},\mathbf{u}\}}(G', \gamma')$.

Proof. First we prove part (i). As $(\hat{\gamma}, \hat{\delta}) = \mathsf{Com}(\gamma, \delta, S)$, there exists a function χ satisfying the conditions of Lemma 12. Let $P_Y(\chi(\hat{t}_1), \chi(\hat{t}_2)) = (t_1, \ldots, t_r), r \geq 2$. As $(\hat{\gamma}, \hat{\delta}) = \mathsf{Com}((\gamma, \delta), S)$, there exists a function χ satisfying the conditions of Lemma 12. Therefore, we get that

$$\hat{\delta}(\hat{\boldsymbol{e}}) = \tau(\delta(\langle\langle t_1, t_2\rangle\rangle) \oplus \cdots \oplus \delta(\langle\langle t_{r-1}, t_r\rangle\rangle)) \tag{1}$$

Applying now Lemma 4 on (1), we have that there exists $h, 1 \leq h < r$ such that

$$\hat{\delta}(\hat{\boldsymbol{e}})[1, m] = \tau(\delta(\langle\langle t_1, t_2\rangle\rangle) \oplus \ldots \oplus \delta(\langle\langle t_h, t_{h+1}\rangle\rangle)) \tag{2}$$

$$\hat{\delta}(\hat{\boldsymbol{e}})[m, |\hat{\delta}(\hat{\boldsymbol{e}})|] = \tau(\delta(\langle\langle t_h, t_{h+1}\rangle\rangle) \oplus \ldots \oplus \delta(\langle\langle t_{r-1}, t_r\rangle\rangle)) \tag{3}$$

We set $\boldsymbol{e} = \langle t_h, t_{h+1}\rangle$. Let (γ', δ') and $(\hat{\gamma}', \hat{\delta}')$ be the characteristic pairs constructed by step **2** of Procedures $\mathsf{Ins}(G, u, S, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m)$ and $\mathsf{Ins}(G, u, S, N, \gamma, \delta, \boldsymbol{e}, 1)$ respectively. We set $l = V(Y) - V(Y')$, $\hat{l} = V(\hat{Y}) - V(\hat{Y}')$ and we denote as t $(\hat{t})$ the unique vertex of Y' $(\hat{Y}')$ adjacent to l $(\hat{l})$.

We will prove now that $(\hat{\gamma}', \hat{\delta}') = \mathsf{Com}((\gamma', \delta'), S \cup \{u\})$. Clearly, it is enough to find a function χ satisfying the condition of Lemma 12. We claim that this function is $\chi' = \chi \cup \{(\hat{t}, t), (\hat{l}, l)\}$. Condition 1 is a direct consequence of the definition of $\chi_{\gamma', S \cup \{u\}}$ and condition 1 for χ. For Condition 2, we first recall that from the same condition for χ we have that $\mathcal{P}_{\text{old}} = \{E(T(\chi(\hat{v}), \chi(\hat{v}))) \mid \{\hat{v}, \hat{u}\} \in E(\hat{Y}) - \{\{\hat{t}_1, \hat{t}_2\}\}\}$ is a partition of $E(Y) - E(T(t_1, t_r))$ which can also be seen as a partition of $E(Y') - (E(Y'(t, t_1)) \cup E(Y'(t, t_r)) \cup E(Y'(t, l)))$. As $(Y', \theta') = \mathsf{Add}(\hat{\gamma}, \hat{e}, u)$, $\mathcal{P}_{\text{new}} = \{E(Y'(\{\chi(\hat{t}), \chi(\hat{t}_1)\})), E(Y'(\{\chi(\hat{t}), \chi(\hat{t}_2)\})), E(Y'(\{\chi(\hat{t}), \chi(\hat{l})\}))\}$ is a partition of $E(Y'(t, t_1)) \cup E(Y'(t, t_r)) \cup E(Y'(t, l))$. Notice now that $\mathcal{P}_{\text{old}} \cup \mathcal{P}_{\text{new}}$ forms a partition of $E(Y')$ and condition 2 holds for χ'. Observe now that because of Condition 3 for χ, the same condition holds also for χ' as far as it concerns the edges of $\hat{Y}'$ that also edges of $\hat{Y}$, i.e. edges not in $E_{\text{new}} = \{\{\hat{t}, \hat{t}_1\}, \{\hat{t}, \hat{t}_2\}, \{\hat{t}, \hat{l}\}\}$. It remains to see that it holds for edges in E_{new} as well. This is obvious for $\{\hat{t}, \hat{l}\}$. Finally, it holds for $\{\hat{t}, \hat{t}_1\}$ and $\{\hat{t}, \hat{t}_2\}$, as a result of 2 and 3.

So far, we have seen that $(\gamma', \delta') = \mathsf{Com}((\hat{\gamma}', \hat{\delta}'), S \cup \{u\})$. In what follows we will prove that this relation is invariant under the transformations applied on $(\hat{\gamma}', \hat{\delta}')$ and (γ', δ') during the loops of step **3** of the computation of of $\mathsf{Ins}(G, u, S, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m)$ and $\mathsf{Ins}(G, u, S, N, \gamma, \delta, \boldsymbol{e}, 1)$ respectively.

Notice that, during the execution of step **3**, no vertex is introduced or removed and therefore the carvings $\hat{\gamma}'$ and γ' remain the same. The only changes to $(\hat{\gamma}', \hat{\delta}')$ and (γ', δ') concern the values of $\hat{\delta}$ and δ.

We will use the notation $(\hat{\gamma}', \hat{\delta}^{(h)})$ and $(\gamma', \delta^{(h)})$ for the results of the h-th execution of the loop of step **3** and, for convenience, we denote $(\hat{\gamma}', \hat{\delta}')$ as $(\hat{\gamma}', \hat{\delta}^{(0)})$, and (γ', δ') as $(\gamma', \delta^{(0)})$. Our proof is by induction.

Suppose that $(\hat{\gamma}', \hat{\delta}^{(h)}) = \mathsf{Com}(\gamma', \delta^{(h)}, S \cup \{u\})$ for any h, $0 < h < \xi$. It remains to prove that $(\hat{\gamma}', \hat{\delta}^{(\xi)}) = \mathsf{Com}(\gamma', \delta^{(\xi)}, S \cup \{u\})$. Notice first that Conditions 1 and 2 of Lemma 12 remain invariant as they concern $\hat{\gamma}$ and γ. Moreover, condition 3 is also invariant for all the edges of $\hat{Y}$ that are not in path $P_{\hat{Y}}(\hat{t}_h, \hat{l})$. Let now $\hat{\boldsymbol{e}} = \langle \hat{v}, \hat{u} \rangle$ be an edge in $P_{\hat{Y}}(\hat{t}_h, \hat{l})$. From the induction hypothesis,

$$\hat{\delta}^{(\xi-1)}(\hat{\boldsymbol{e}}) = \tau(\delta^{(\xi-1)}(\langle\!\langle z_1, z_2 \rangle\!\rangle) \oplus \cdots \oplus \delta^{(\xi-1)}(\langle\!\langle z_{\rho-1}, z_\rho \rangle\!\rangle)) \tag{4}$$

where $(z_1, \ldots, z_\rho) = P_Y(\chi(\hat{v}), \chi(\hat{u}))$. An easy consequence of Condition 2 is that the edges in $P_Y(\chi(\hat{v}), \chi(\hat{u}))$ are also edges of $P_Y(\chi(\hat{t}_h), \chi(\hat{l}))$. Therefore,

$$\forall_{i=1,\ldots\rho-1}\ \delta^{(\xi)}(\langle\!\langle z_i, z_{i+1} \rangle\!\rangle) = \delta^{(\xi-1)}(\langle\!\langle z_i, z_{i+1} \rangle\!\rangle) + 1. \tag{5}$$

As $\hat{\delta}^{\xi}(\langle\!\langle \hat{v}, \hat{u} \rangle\!\rangle) = \hat{\delta}^{\xi-1}(\langle\!\langle \hat{v}, \hat{u} \rangle\!\rangle) + 1$, 4 and 5 along with the simple observation that for any sequence A $\tau(A)+1 = \tau(A+1)$, implies that 4 holds for any $\hat{\boldsymbol{e}} \in \boldsymbol{E}(\hat{Y})$ and condition 3 is satisfied.This completes the proof of (i)

The flow of the proof for (ii) is exactly the same as in the proof of (i) with the difference that now weaker versions of (2), (3). In particular, let ξ be a function satisfying the conditions 1–3 of Lemma 12 and we define $\hat{e} = \{\hat{v}, \hat{u}\}$ as the unique edge of $\hat{Y}$ where $P_Y(\hat{v}, \hat{u})$ contains e. Respecting the orientation of e in $P_Y(\hat{v}, \hat{u})$, we can assume that $\hat{\boldsymbol{e}} = \langle \hat{v}, \hat{u} \rangle$, $\boldsymbol{e} = \langle t_\omega, t_{\omega+1} \rangle$, $1 \leq \omega \leq |P_Y(\hat{v}, \hat{u})|$ and that

$$\hat{\delta}(\hat{\boldsymbol{e}}) = \tau(\delta(\langle\!\langle t_1, t_2 \rangle\!\rangle) \oplus \cdots \oplus \delta(\langle\!\langle t_\omega, t_{\omega+1} \rangle\!\rangle) \oplus \cdots \oplus \delta(\langle\!\langle t_{r-1}, t_r \rangle\!\rangle)) \tag{6}$$

From Lemma 6 and 6there exists an integer $m, 1 \leq m \leq |\hat{\delta}(\hat{\boldsymbol{e}})|$ where

$$\hat{\delta}(\hat{\boldsymbol{e}})[1, m] \prec \tau(\langle\!\langle t_1, t_2 \rangle\!\rangle) \oplus \cdots \oplus \delta(\langle\!\langle t_\omega, t_{\omega+1} \rangle\!\rangle)) \tag{7}$$

$$\hat{\delta}(\hat{\boldsymbol{e}})[m, |\hat{\delta}(\hat{\boldsymbol{e}})|] \prec \tau(\delta(\langle\!\langle t_\omega, t_{\omega+1} \rangle\!\rangle) \oplus \cdots \oplus \delta(\langle\!\langle t_{r-1}, t_r \rangle\!\rangle)) \tag{8}$$

Notice that (7), (8), are the same as (2), (3), with the difference that "=" has been replaced by "$\prec$".

It is now easy to check that repeating the steps of the proof of case (i) it follows that $\mathsf{Ins}(G, u, S, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) \prec \mathsf{Com}(\mathsf{Ins}(G, u, V(G), N, \gamma, \delta, \boldsymbol{e}, 1), S \cup \{u\})$. □

Lemma 17. *Let $(\hat{\gamma}_i, \hat{\delta}_i)$ be two characteristics of a graph G where $(\hat{\gamma}_1, \hat{\delta}_1) \prec_\psi (\hat{\gamma}_2, \hat{\delta}_2)$ for some bijection $\psi : V(\hat{T}_1) \to V(\hat{T}_2)$ (we use the notation $\hat{\gamma}_i = (\hat{T}_i, \hat{\theta}_i), i = 1, 2$). Let also $N \subseteq S$ be subsets of $V(G)$, and $G' = G \overset{u}{+} N$ for some new vertex $u \notin V(G)$. We use the notation $\hat{\gamma}_i = (\hat{Y}_i, \hat{\theta}_i), i = 1, 2$. Then for any $\hat{\boldsymbol{e}} \in \boldsymbol{E}(\hat{Y}_1)$ and any $m_1, 1 \leq m_1 \leq |\hat{\delta}_1(\hat{\boldsymbol{e}})|$ there exists an $m_2, 1 \leq m_2 \leq |\hat{\delta}_2(\psi(\hat{\boldsymbol{e}}))|$ and a function ψ' such that*

$$\mathsf{Ins}(G, u, S, N, \hat{\gamma}_1, \hat{\delta}_1, \hat{\boldsymbol{e}}_1, m_1) \prec_{\psi'} \mathsf{Ins}(G, u, S, N, \hat{\gamma}_2, \hat{\delta}_2, \hat{\phi}(\hat{\boldsymbol{e}}), m_2).$$

Proof. We apply Lemma 5 on the typical sequence $\hat{\delta}_1(\hat{\boldsymbol{e}})$ and $\hat{\delta}_2(\psi(\hat{\boldsymbol{e}}))$ and we have that there exists an $m_2, 1 \leq m_2 \leq |\hat{\delta}_2(\psi(\hat{\boldsymbol{e}}))|$ such that

$$\hat{\delta}_1(\hat{\boldsymbol{e}})[1, m_1] \prec \hat{\delta}_1(\hat{\boldsymbol{e}})[1, m_2] \tag{9}$$

$$\hat{\delta}_1(\hat{\boldsymbol{e}})[m_1, |\hat{\delta}_1(\hat{\boldsymbol{e}})|] \prec \hat{\delta}_1(\hat{\boldsymbol{e}})[m_2, |\hat{\delta}_2(\psi(\hat{\boldsymbol{e}}))|] \tag{10}$$

We run, in parallel $\mathsf{Ins}(G,u,S,N,\hat{\gamma}_1,\hat{\delta}_1,\hat{\boldsymbol{e}}_1,m_1)$ and $\mathsf{Ins}(G,u,S,N,\hat{\gamma}_2,\hat{\delta}_2,\hat{\phi}(\hat{\boldsymbol{e}}),m_2)$ and we assume that, for $i=1,2$, during the computation of $(\hat{Y}_i',\hat{\theta}_i')=\mathsf{Add}(\hat{\gamma}_i,\hat{e}_i,u)$ in step **2** the notation used is $V(\hat{Y}_i')-V(\hat{Y}_i)=\{l_i,t_i\}$. We define $\psi'=\psi\cup\{(l_1,l_2),(t_1,t_2)\}$. We also assume that $\hat{\boldsymbol{e}}=\langle\hat{t}_1,\hat{t}_2\rangle$. Using now the fact that $\hat{\gamma}_1\equiv_\psi\hat{\gamma}_2$, it is easy to see that $\hat{\gamma}_1'\equiv_\psi\hat{\gamma}_2'$.

We now claim that if $(\hat{\gamma}_i',\hat{\delta}_i'), i=1,2$ are the characteristics constructed after step **2** of , $i=1,2$ then

$$(\hat{\gamma}_1',\hat{\delta}_1')\prec_{\psi'}(\hat{\gamma}_2',\hat{\delta}_2') \tag{11}$$

For this, we need to show that $\forall_{\hat{\boldsymbol{e}}\in\boldsymbol{E}(\hat{Y}')}\ \hat{\delta}_1'(\hat{\boldsymbol{e}})\prec\hat{\delta}_2'(\psi'(\hat{\boldsymbol{e}}))$ Clearly, this is obvious if $\hat{e}$ is an edge in $E(\hat{Y}')\cap E(\hat{Y})=E(\hat{Y})-\{\{\hat{t},\hat{t}_1\},\{\hat{t},\hat{t}_2\},\{\hat{t},\hat{l}\}\}$. In the cases where $\hat{e}=\{\hat{t},\hat{t}_1\}$ and $\hat{e}=\{\hat{t},\hat{t}_2\}$, the required is a direct consequence of 9 and 10 respectively. The case where $\hat{e}=\{\hat{t},\hat{l}\}$ is trivial.

It now remains to prove that 11 holds after the application of step **3** as well. Clearly, we have to fix our attention to the modifications applied to $\hat{\delta}_1'$ and $\hat{\delta}_2'$ as $\hat{\gamma}_1'$ and $\hat{\gamma}_2'$ do not change during step **3**.

We will denote $\hat{\delta}_i'$ as $\hat{\delta}_i^{(0)}, i=1,2$ and we will use the notation $\hat{\delta}^{(h)}$ for the results of the h-th execution of the loop in step **3** during the computation of $\mathsf{Ins}(G,u,S,N,\hat{\gamma}_i,\hat{\delta}_i,\hat{\phi}(\hat{\boldsymbol{e}}),m_i),\ i=1,2$. Suppose that $(\hat{\gamma}_1',\hat{\delta}_1^{(h)})\prec_\psi(\hat{\gamma}_2',\hat{\delta}_2^{(h)}$ for any h, $0<h<\xi$. It remains to prove that $(\hat{\gamma}_1',\hat{\delta}_1^{(\xi)})\prec(\hat{\gamma}_2',\hat{\delta}_2^{(\xi)})$ or, equivalently, that $\forall_{\hat{\boldsymbol{e}}\in\boldsymbol{E}(\hat{Y}_1')}\ \hat{\delta}_1^{(\xi)}(\hat{\boldsymbol{e}})\prec\hat{\delta}_2^{(\xi)}(\psi'(\hat{\boldsymbol{e}}))$. Observe that $\psi'(P_{\hat{Y}_1'}(\hat{t}_\xi,\hat{l}))=P_{\hat{Y}_2'}(\psi'(\hat{t}_\xi),\psi'(\hat{l}))$. This means that, for i=1,2, $\hat{\delta}_i^{(\xi)}$ is different than $\hat{\delta}_i^{(\xi-1)}$ only for the edges of paths $P_{\hat{Y}_1'}(\hat{t}_\xi,\hat{l})$ and $P_{\hat{Y}_2'}(\psi'(\hat{t}_\xi),\psi'(\hat{l}))$ respectively. Therefore it is enough to examine the case where $\hat{\boldsymbol{e}}$ is an edge of $P_{\hat{Y}_1'}(\hat{t}_\xi,\hat{l})$. Clearly, $\hat{\delta}_1^{(\xi)}(\hat{\boldsymbol{e}})=\hat{\delta}_1^{(\xi-1)}(\hat{\boldsymbol{e}})+1$ and $\hat{\delta}_2^{(\xi)}(\psi'(\hat{\boldsymbol{e}}))=\hat{\delta}_2^{(\xi-1)}(\psi'(\hat{\boldsymbol{e}}))+1$. Combining the last two relations with the fact that $\hat{\delta}_1^{(\xi-1)}(\hat{\boldsymbol{e}})\prec\hat{\delta}_2^{(\xi-1)}(\psi'(\hat{\boldsymbol{e}}))$, gives $\hat{\delta}_1^{(\xi)}(\hat{\boldsymbol{e}})\prec\hat{\delta}_2^{(\xi)}(\psi'(\hat{\boldsymbol{e}}))$ and this completes the proof of the lemma. □

We now give an algorithm that, if X_p is an *introduce* node, computes a full set of characteristics $FS(p)$ for G_p, given a full set of characteristics $FS(q)$ of carvings for G_q where q is the (unique) child of p in U.

Algorithm Introduce-Node

Input: A full set of characteristics $FS(q)$ of carvings for G_q.

Output: A full set of characteristics $FS(p)$ of carvings for G_p.

1: Initialize $FS(p)=\emptyset$ and set $N=N_{G_p}(u)$ where $\{u\}=X_p-X_q$.
2: For any X_q-characteristic $(\hat{\gamma},\hat{\delta})\in FS(q)$ where $\hat{\gamma}=(\hat{Y},\hat{\theta})$ apply step **3**.
3: For any edge $\hat{\boldsymbol{e}}\in\boldsymbol{E}(\hat{Y})$ apply step **4**.
4: For any $m=1,\dots,|\delta(\hat{\boldsymbol{e}})|$, apply step **5**.
5: Let $(\hat{\gamma}',\hat{\delta}')=\mathsf{Ins}(G_q,u,X_q,N,\hat{\gamma},\hat{\delta},\hat{\boldsymbol{e}},m)$ and
if $\max(\hat{\gamma}',\hat{\delta}')\le k$, then set $FS(p)\leftarrow FS(p)\cup\{(\hat{\gamma}',\hat{\delta}')\}$.
6: Output $FS(p)$.
7: End.

Lemma 18. *If $FS(q)$ is a full set of characteristics of carvings for G_q and p is an* introduce *node with child q, then the set $FS(p)$ constructed by* Algorithm Introduce-Node *is a full set of characteristics of carvings for G_p.*

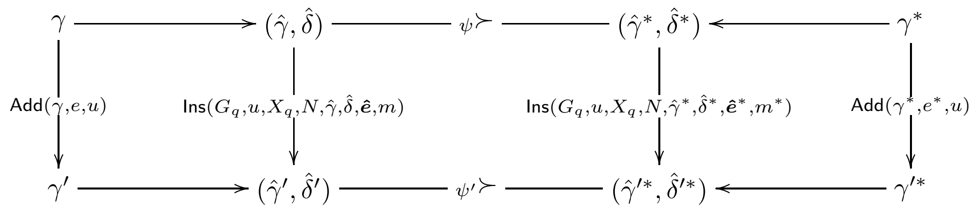

Fig. 7. The structure of Lemma 18.

Proof. We will prove first that the set $FS(p)$ computed by the algorithm Introduce-Node is a set of X_p-characteristics of carvings for G_i. To avoid overloaded expressions, whenever we refer to a carving or to a characteristic, we will insist that its width is bounded by k. We will show that for any $(\hat{\gamma}', \hat{\delta}') \in FS(q)$ there exists a carving γ' of G_p where $(\hat{\gamma}', \hat{\delta}') = C_{X_p}(G_p, \gamma')$. Clearly, as $(\hat{\gamma}', \hat{\delta}')$ was constructed by Algorithm Introduce-Node, there must be a characteristic $(\hat{\gamma}, \hat{\delta}) \in FS(q)$ an edge $\hat{\boldsymbol{e}} = \boldsymbol{E}(\hat{Y})$ (we denote $\hat{\gamma} = (\hat{Y}, \hat{\theta})$) and an integer $m, 1 \leq m \leq |\hat{\delta}(\hat{\boldsymbol{e}})|$ such that

$$(\hat{\gamma}', \hat{\delta}') = \mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) \tag{12}$$

As $(\hat{\gamma}, \hat{\delta}) \in FS(q)$, it will be a X_q-characteristic for G_q and therefore, there exists a carving $\gamma = (Y, \theta)$ of G_q where $(\hat{\gamma}, \hat{\delta}) = C_{X_q}(G_q, \gamma)$. From Lemma 16(i) we have that there exists an edge $\boldsymbol{e} \in \boldsymbol{E}(Y)$ such that

$$\mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) = \mathsf{Com}(\mathsf{Ins}(G_q, u, V(G_q), N, \gamma, d_{G_q,\gamma}, \boldsymbol{e}, 1), X_q \cup \{u\}) \tag{13}$$

From Lemma 15, we have that $\mathsf{Ins}(G_q, u, V(G_q), N, \gamma, d_{G_q,\gamma}, \gamma, 1)$ is the $V(G_p)$-characteristic of $\gamma' = \mathsf{Add}(\gamma, e, u)$ and therefore,

$$\mathsf{Com}(\mathsf{Ins}(G_q, u, V(G_q), N, \gamma, d_{G_q,\gamma}, \boldsymbol{e}, 1), X_p) = \mathsf{Com}(\gamma', d_{G',\gamma'}, X_p) \tag{14}$$

Combining now (12), (13), and (14), we have that $(\hat{\gamma}', \hat{\delta}') = \mathsf{Com}(\gamma', d_{G',\gamma'}, X_p) = C_{X_p}(G_p, \gamma')$.

It remains now to prove that $FS(p)$ is a full set of characteristics of carvings for G_p. Let $\gamma' = (Y', \theta')$ be a carving of G_p. We will show that there exists a carving γ'^* of G_p such that

$$C_{X_p}(G_p, \gamma'^*) \prec C_{X_p}(G_i, l') \text{ and } C_{X_p}(G_i, \gamma'^*) \in FS(p).$$

Set now $\gamma = \mathsf{Del}(\gamma', \theta'^{-1}(u))$ and let e be the unique edge of Y' that contains $\theta'^{-1}(u)$ as an endpoint. From Lemma 15, we have that

$$\mathsf{Ins}(G_q, u, V(G_q), N, \gamma, d_{G_q,\gamma}, \boldsymbol{e}, 1) = (\gamma', d_{G_p,\gamma'})$$

and therefore,

$$\mathsf{Com}(\mathsf{Ins}(G_q, u, V(G_q), N, \gamma, d_{G_q,\gamma}, e, 1), X_p) = \mathsf{Com}(\gamma', d_{G_p,\gamma'}, X_p) = C_{X_p}(G_p, \gamma') \tag{15}$$

Set now $(\hat{\gamma}, \hat{\delta}) = C_{X_q}(G_q, \gamma)$ where $\hat{\gamma} = (\hat{Y}, \hat{\theta})$. From Lemma 16.ii we have that there exists an edge $\hat{\boldsymbol{e}} \in \boldsymbol{E}(\hat{Y})$ and and integer m, $1 \leq m \leq |\hat{\delta}(\hat{\boldsymbol{e}})|$ such that

$$\mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) \prec \mathsf{Com}(\mathsf{Ins}(G_q, u, V(G_q), N, \gamma, d_{G_q,\gamma}, \boldsymbol{e}, 1), X_q \cup \{u\}) \tag{16}$$

As $FS(q)$ is a full set of characteristics, we have that there exists a carving γ^* of $V(G_q)$ such that

$$C_{X_q}(G_q, \gamma^*) \prec C_{X_q}(G_q, \gamma) \text{ and } C_{X_q}(G_q, l^*) \in FS(q).$$

Let $C_{X_q}(G_q, \gamma^*) = (\hat{\gamma}^*, \hat{\delta}^*)$. We use the notation $\hat{\gamma}^* = (\hat{Y}^*, \hat{\theta}^*)$, notice that ψ maps the vertices of $\hat{Y}$ to the vertices of $\hat{Y}^*$. From Lemma 17 we have that there exists a bijection ψ' and an integer $m^*, 1 \leq m^* \leq |\hat{\delta}(\hat{\boldsymbol{e}}^*)|$ (we set $\hat{\boldsymbol{e}}^* = \hat{\delta}(\psi(\hat{\boldsymbol{e}}))$) such that

$$\mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}_*, \hat{\delta}_*, \hat{\boldsymbol{e}}^*, m^*) \prec_{\psi'} \mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}, \hat{\delta}, \hat{\boldsymbol{e}}, m) \tag{17}$$

From Lemma 16.i, we have that there exists an edge $\boldsymbol{e}^* \in \boldsymbol{E}(Y^*)$ such that

$$\mathsf{Com}(\mathsf{Ins}(G_q, u, V(G_q), N, \gamma^*, d_{G_q,\gamma^*}, \boldsymbol{e}^*, 1), X_p) = \mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}^*, \hat{\delta}^*, \hat{\boldsymbol{e}}^*, m^*) \tag{18}$$

We set $\gamma'^* = \mathsf{Add}(\gamma^*, e^*, u)$ and after applying Lemma 15 we have that

$$(\gamma'^*, d_{G_p,\gamma'^*}) = \mathsf{Ins}(G_q, u, V(G_q), N, l^*, d_{G_q,l^*}, \hat{e}^*, 1)$$

and therefore,

$$C_{X_p}(G_p, \gamma'^*) = \mathsf{Com}(\gamma'^*, d_{G_p,\gamma'^*}, X_p) = \mathsf{Com}(\mathsf{Ins}(G_q, u, V(G_q), N, \gamma^*, d_{G_q,\gamma^*}, \hat{e}^*, 1), X_p) \tag{19}$$

From (18) and (19) we have that $C_{X_p}(G_p, \gamma'^*) = \mathsf{Ins}(G_q, u, X_q, N, \hat{\gamma}^*, \hat{\delta}^*, \boldsymbol{e}^*, m^*)$. Since $(d_{G_q,\gamma^*}) \in FS(q)$ algorithm Introduce-Node makes that $C_{X_p}(G_p, \gamma'^*) \in FS(p)$. Moreover, combining (15)–(19) we conclude that $C_{X_p}(G_p, \gamma'^*) \prec C_{X_p}(G_p, \gamma')$. (For a diagram depicting the structure of the proof see Figure 7) □

4.3 A full set for a *forget* node

We will now consider the case where X_p is a *forget* node. We will provide an algorithm that given a full set of characteristics $FS(q)$ for X_q, computes a full set of characteristics $FS(p)$ for X_p. We start by defining a deletion procedure that, when applied to characteristics of carvings, operates inversely to procedure Ins.

Procedure $\mathsf{Del}(\gamma, \delta, u)$.
Input: A characteristic pair (γ, δ) where $\gamma = (Y, \theta)$ and an element $u \in \theta(A(Y))$.
Output: A characteristic pair (γ', δ').

1: $\gamma' = \mathsf{Rem}(\gamma, u)$.
2: $\delta' = \delta \cup \{(\langle t_1, t_2\rangle, \tau(\delta(\langle t, t_1\rangle) \oplus \delta(\langle t, t_2\rangle)))\} -$
$\{(\langle t, l\rangle, \delta(\langle t, l\rangle)),$
$(\langle t, t_1\rangle, \delta(\langle t, t_1\rangle)),$
$(\langle t, t_2\rangle, \delta(\langle t, t_2\rangle)\}.$
3: Output (γ', δ').
4: End.

The following lemma is a direct consequence of the definitions of the procedures Com and Del.

Lemma 19. *Let (γ, δ) be a characteristic pair of a given set $\mathcal{O}$ where $\gamma = (Y, \theta)$ and let $V \subseteq \theta^{-1}(A(Y))$. Then, for any $v \in V$ the following holds.*

$$\mathsf{Com}(\gamma, \delta, V - \{v\}) = \mathsf{Del}(\mathsf{Com}(\gamma, \delta, V), v)$$

Observe that Lemma 19 can provide an alternative, recursive definition of procedure Com, based on procedure Del.

The following monotonicity result is a direct consequence of Lemma 8

Lemma 20. *Let $(\gamma_i, \delta_i), i = 1, 2$ be two characteristic pairs of a given graph G. If $(\gamma_2, \delta_2) \prec (\gamma_1, \delta_1)$, then for any $u \in V(\gamma_1)$, $\mathsf{Del}(\gamma_2, \delta_2, u) \prec \mathsf{Del}(\gamma_1, \delta_1, u)$.*

Now we can give an algorithm that, given a tree decomposition D of the graph G, for any forget node X_p, computes a full set of characteristics $FS(p)$ for the graph G_p, given a full set of characteristics $FS(q)$ for the graph G_q.

Algorithm Forget-Node

Input: A full set of characteristics $FS(q)$ for G_q.
Output: A full set of characteristics $FS(p)$ for G_p.

```
Initialize FS(p) = ∅ and let u be the forget vertex of G_p.
For any (γ̂, δ̂) ∈ FS(q) do
      FS(p) ← FS(q) ∪ {Del(γ̂, δ̂, u)}.
Output FS(p).
End.
```

Lemma 21. *If $FS(q)$ is a full set of characteristics of carvings for G_p and q is a* forget *node with child q, then the set $FS(p)$ constructed by* Algorithm Forget-Node *is a full set of characteristics of carvings for G_p.*

Proof. As $G_p = G_q$ we will use the notation G for both of them. We will also denote as u the forget vertex of G_p. We will prove first that $FS(p)$ is a set of X_p-characteristics for G. We need to prove that there exists a carving γ of G where

$$C_{X_p}(\gamma, G) = \mathsf{Com}(\gamma, d_{G,\gamma}, X_p) = (\hat{\gamma}', \hat{\delta}')$$

for any $(\hat{\gamma}', \hat{\delta}') \in FS(p)$. As $(\hat{\gamma}', \hat{\delta}')$ has been constructed by procedure Forget-Node there must exists a X_q-characteristic $(\hat{\gamma}, \hat{\delta}) \in FS(q)$ such that

$$(\hat{\gamma}', \hat{\delta}') = \mathsf{Del}(\hat{\gamma}, \hat{\delta}, u). \tag{20}$$

As $(\hat{\gamma}, \hat{\delta}) \in FS(q)$, there exists a carving γ of G such that

$$(\hat{\gamma}, \hat{\delta}) = \mathsf{Com}(\gamma, d_{G,\gamma}, X_q) \tag{21}$$

and therefore, from (20) and (21) we have

$$(\hat{\gamma}', \hat{\delta}') = \mathsf{Del}(\mathsf{Com}(\gamma, d_{G,\gamma}, X_q), u) \tag{22}$$

and using (22) and Lemma 19 we have that $C_{X_p}(\gamma, G) = \mathsf{Com}(\gamma, d_{G,\gamma}, X_p) = (\gamma', \delta')$.

We will now prove that $FS(i)$ is a full set of X_p-characteristics for G. Let γ be a carving of G of cutwidth at most k. We will show that there exists a carving γ_* of G such that

$$C_{X_p}(G,\gamma_*) \prec C_{X_p}(G,\gamma) \text{ and } C_{X_p}(G,\gamma_*) \in FS(p).$$

From Lemma 19 we have that

$$C_{X_p}(G,\gamma) = \mathsf{Com}(\gamma, d_{G,\gamma}, X_p) = \mathsf{Del}(\mathsf{Com}(\gamma, d_{G,\gamma}, X_q), u) \tag{23}$$

As $FS(q)$ is a full set of characteristics, there exists a carving γ_* of $V(G)$ such that $C_{X_q}(G,\gamma_*) \in FS(q)$ and $C_{X_q}(G,\gamma_*) \prec C_{X_q}(G,\gamma)$ or, equivalently,

$$\mathsf{Com}(\gamma_*, d_{G,\gamma_*}, X_q) \prec \mathsf{Com}(\gamma, d_{G,\gamma}, X_q) \tag{24}$$

Using now Lemma 20 we can rewrite (24) as follows.

$$\mathsf{Del}(\mathsf{Com}(\gamma_*, d_{G,\gamma_*}, X_q), u) \prec \mathsf{Del}(\mathsf{Com}(\gamma, d_{G,\gamma}, X_q), u) \tag{25}$$

Applying again Lemma 19 we have that

$$C_{X_p}(G,\gamma_*) = \mathsf{Com}(\gamma_*, d_{G,\gamma_*}, X_p) = \mathsf{Del}(\mathsf{Com}(\gamma_*, d_{G,\gamma_*}, X_q), u) \tag{26}$$

Combining now (23), (25), and (26), we have that $C_{X_p}(G,\gamma_*) \prec C_{X_p}(G,\gamma)$. Finally as

$$C_{X_q}(G,\gamma_*) = \mathsf{Com}(\gamma_*, d_{G,\gamma_*}, X_q) \in FS(q),$$

the output of $\mathsf{Del}(\mathsf{Com}(\gamma_*, d_{G,\gamma_*}, X_q), u)$ will be one of the characteristics included in $FS(p)$. Therefore, $C_{X_p}(G,\gamma_*) \in FS(p)$ and this completes the proof of the lemma.

4.4 A full set for a *join* node

We will now consider the case where X_p is a *join* node and $q_i, i = 1,2$ are the two children of p in U. We observe that $V(G_{q_1}) \cap V(G_{q_2}) = X_p$, $G_{q_1} \cup G_{q_2} = G_p$ and we may assume that $E(G_{q_1}) \cap E(G_{q_2}) = \emptyset$. Given a full set of characteristics $FS(q_1)$ for X_{q_1} and a full set of characteristics F_{q_2} for X_{q_2}, the following algorithm computes a full set of characteristics $FS(p)$ for X_p.

Algorithm Join-Node

Input: A full set of characteristics $FS(q_1)$ of carvings for G_{q_1} and a full set of characteristics $FS(q_2)$ of carvings for G_{q_2}.
Output: A full set of characteristics $FS(p)$ of carvings for G_p.

1: Initialize $FS(p) = \emptyset$.
2: For any pair of X_{q_i}-characteristics $(\hat{\gamma}_i, \hat{\delta}_i) \in FS(q_i), i = 1,2$, apply step (**3**).
(we use the notation $\hat{\gamma}_i = (\hat{Y}_i, \hat{\theta}_i)$)
3: For any bijection $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$ where $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$ apply step (**4**).
4: For any $\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2$ apply step (**5**).
5: If $\max(\hat{\gamma}_1, \hat{\delta}') \leq k$, set $FS(p) \leftarrow FS(p) \cup \{(\hat{\gamma}_1, \hat{\delta}')\}$.
6: Output $FS(p)$.
7: End.

Procedure Construct-Join-Carving($G_1, G_2, S, \gamma_1, \gamma_2, \hat{\phi}', \hat{\delta}'$).

Input: Two graphs G_1, G_2 and a set S where $G_1 \cap G_2 = (S, \emptyset)$.

Two carvings γ_1 and γ_2, and a bijection $\hat{\phi}$ where $C_S(G_1, \gamma_1) \equiv_{\hat{\phi}} C_S(G_2, \gamma_2)$.

A function $\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2$ where $(\hat{\gamma}_i, \hat{\delta}_i) = C_S(G_i, \gamma_i), i = 1, 2$.

Output: A $V(G)$-characteristic (γ, δ) of $G = G_1 \cup G_2$ where $C_S(G, \gamma) = (\hat{\gamma}_1, \hat{\delta}')$.

(Assume the notations $\delta_i = d_{G,\gamma_i}, \gamma_i = (Y_i, \theta_i)$ and $\hat{\gamma}_i = (\hat{Y}_i, \hat{\theta}_i), i = 1, 2$.)

1: Define $\sigma_i : V(\hat{Y}_1) \to V(Y_i)), i = 1, 2$ such that $\sigma_1 = \chi_{\gamma_1, S}$ and $\sigma_2 = \chi_{\gamma_2, S} \circ \hat{\phi}$.

2: For any $\hat{\boldsymbol{e}} = \langle \hat{v}, \hat{u} \rangle \in \hat{E}(\hat{Y}_i)$

(a) For $i = 1, 2$

Set $(t^i_1, \ldots, t^i_{r^i_{\hat{e}}}) = P_{Y_i}(\sigma_i(\hat{v}), \sigma_i(\hat{u}))$.

Set $A_i = [\delta_i(\langle\!\langle t^i_1, t^i_2 \rangle\!\rangle), \ldots, \delta_i(\langle\!\langle t^i_{r^i_{\hat{e}}-1}, t^i_{r^i_{\hat{e}}} \rangle\!\rangle)]$. (Notice that $\tau(A_i) = \hat{\delta}_i(\hat{\boldsymbol{e}})$).

(b) Let $\hat{\delta}'(\hat{\boldsymbol{e}}) = \tau(\tilde{A}_1 + \tilde{A}_2)$ where $\tilde{A}_1 \sim \tilde{A}_2$, and $\tilde{A}_i \in \mathcal{E}(\hat{\delta}_i(\hat{\boldsymbol{e}})), i = 1, 2$.

(c) For $i = 1, 2$

For $j = 1, \ldots, |\hat{\delta}_i(\hat{\boldsymbol{e}})| - 1$

Set $s^i_j = \beta_{A_i}(j)$, $f^i_j = \beta_{A_i}(j+1)$, and $\pi^i_j = A_i[s^i_j + 1, f^i_j - 1]$.

Let $\tilde{A}_i = [\tilde{a}^i_1, \ldots, \tilde{a}^i_\nu]$ where $\nu = |\hat{\delta}_1(\hat{\boldsymbol{e}})| + |\hat{\delta}_2(\hat{\boldsymbol{e}})| - 1$.

(d) For $i = 1, 2$

For $j = 1, \ldots, \nu - 1$

If $\tilde{a}^i_j = \tilde{a}^i_{j+1}$ and $\tilde{a}^{3-i}_j, \tilde{a}^{3-i}_{j+1}$ is the h-th number change in $\tilde{A}_{3-i}$ then

set $x^i_{j,j+1} = |\pi^{3-i}_h| \times [\tilde{a}^i_j]$, $Y_{j,j+1} = T^{3-i}_Y(t^{3-i}_{s^{3-i}_h}, t^{3-i}_{f^{3-i}_h})$,

$e^s_j = \{t^{3-i}_{s^{3-i}_j}, t^{3-i}_{s^{3-i}_j+1}\}$, and $e^l_j = \{t^{3-i}_{f^{3-i}_j}, t^{3-i}_{f^{3-i}_j+1}\}$.

If $\tilde{a}^i_j \neq \tilde{a}^i_{j+1}$ and $\tilde{a}^i_j, \tilde{a}^i_{j+1}$ is the h-th number change in $\tilde{A}_i$ then

set $x^i_{j,j+1} = \pi^i_h$, $Y_{j,j+1} = T^i_Y(t^i_{s^i_h}, t^i_{f^i_h})$,

$e^s_j = \{t^i_{s^i_j}, t^i_{s^i_j+1}\}$, and $e^l_j = \{t^i_{f^i_j}, t^i_{f^i_j+1}\}$.

Set $A'_i = [\tilde{a}^i_1] \oplus x^i_{1,2} \oplus [\tilde{a}^i_2] \oplus \cdots \oplus [\tilde{a}^i_{\nu-1}] \oplus x^i_{\nu-1,\nu} \oplus [\tilde{a}^i_\nu]$.

(e) Set $A'_{\hat{e}} = A'_1 + A'_2$ and set $Y_{\hat{e}} = \bigcup_{j=1,\ldots,\nu-1} Y_{j,j+1}$.

(f) For $j = 1, \ldots, \nu - 1$, identify e^l_j and e^s_{j+1} in $Y_{\hat{e}}$.

(g) Let $t^{\hat{e}}_{\text{start}}$ and $t^{\hat{e}}_{\text{fin}}$ be the endpoints of e^s_1 and $e^l_{\nu-1}$ that are leaves of $Y_{\hat{e}}$.

(Notice that $t^{\hat{e}}_{\text{start}} = \sigma_{1 \text{ or } 2}(\hat{v})$ and $t^{\hat{e}}_{\text{fin}} = \sigma_{1 \text{ or } 2}(\hat{u})$)

(h) For any edge $\boldsymbol{e} \in \boldsymbol{E}(Y_{\hat{e}})$

if $\boldsymbol{e}$ is the j-th edge of $P_{Y_{\hat{e}}}(t^{\hat{e}}_{\text{start}}, t^{\hat{e}}_{\text{fin}})$ then $\delta_{\hat{e}}(\boldsymbol{e}) = [A'(j)]$,

otherwise, if $\boldsymbol{e} \in \boldsymbol{E}(T_1)$ then $\delta_{\hat{e}}(\boldsymbol{e}) = \delta_1(\boldsymbol{e})$,

otherwise, $\delta_{\hat{e}}(\boldsymbol{e}) = \delta_2(\boldsymbol{e})$.

3: Set $\delta = \bigcup_{\hat{e} \in E(\hat{Y}_1)} \delta_{\hat{e}}$, $\theta = \{(v, (\theta_1 \cup \theta_2)(v)) \mid v \in A(Y)\}$, and $Y = \bigcup_{\hat{e} \in E(\hat{Y}_1)} Y_{\hat{e}}$.

4: For any $\hat{v} \in A(\hat{Y}_1)$,

Let $\hat{e}_1, \hat{e}_2$, and $\hat{e}_3$ be the edges of $\hat{Y}_1$ incident to $\hat{v}$.

For $h = 1, 2, 3$, let $t^{\hat{e}_h}_{\text{start or fin}}$ be the leaf of $Y_{\hat{e}_h}$ where $t^{\hat{e}_h}_{\text{start or fin}} = \sigma_{1 \text{ or } 2}(\hat{v})$.

Identify in Y vertices $t^{\hat{e}_h}_{\text{start or fin}}, h = 1, 2, 3$.

5: Output (γ, δ) where $\gamma = (Y, \theta)$

6: End.

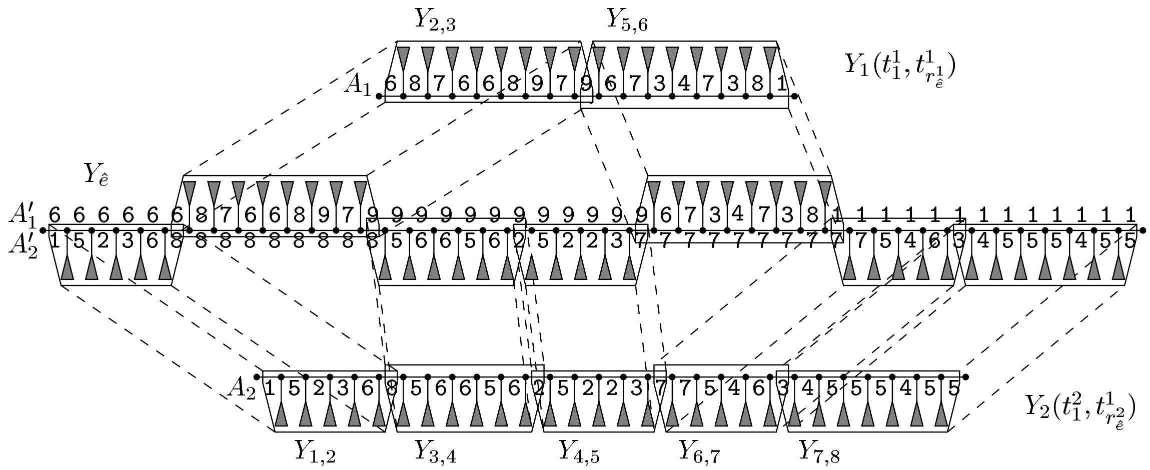

Fig. 8. Example of the operation of the loop of step 2. The first and the third tree are $Y_1(t_1^1, t_{r_{\hat{e}}^1}^1)$ and $Y_2(t_1^2, t_{r_{\hat{e}}^2}^2)$ respectively. The second tree is $Y_{\hat{e}}$.

The algorithm starts setting up (step 1) the correspondences σ_1 and σ_2 of the vertices in $\hat{Y}_1$ to the vertices of Y_1 and Y_2 respectively. The main part of the algorithm is the loop applied in step 2 for any edge $\hat{e} = \{\hat{v}, \hat{u}\}$ of $\hat{Y}_1$ (an example is displayed in Figure 8). The integers corresponding to the spines of these trees form the sequences A_1 and A_2 computed in loop 2(a) and make it possible to calculate in step 2(b) the sequences $\tilde{A}_i, i = 1, 2$. Each of $T_{Y_i}(t_1^i, t_{r_{\hat{e}}^i}^i), i = 1, 2$ is decomposed into parts delimited by the portions of A_i that need to be removed towards extracting $\tau(A_i)$. This information is stored to the sequences $[s_1^i, \ldots, s_{|\hat{\delta}_i(\hat{e})|}^i]$ and $[f_1^i, \ldots, f_{|\hat{\delta}_i(\hat{e})|}^i]$ computed in loop 2(c). The crucial part of the algorithm is step 2:(d) where these portions are interleaved following the way $\hat{\delta}_i(\hat{e}), i = 1, 2$ are expanded in order to create $\tilde{A}_i, i = 1, 2$ respectively. This interlieving is represented by an arrangement of all these portions into a sequence $\boldsymbol{Y} = [Y_{1,2}, \ldots, Y_{|\tilde{A}_i - 1|, |\tilde{A}_i|}]$ and by the definition of sequences $A'_i, i = 1, 2$ where $|A'_1| = |A'_2|$. The union of the graphs in $\boldsymbol{Y}$ and the identification of the extreme edges of the spines of its consecutive members is accomplished in steps 2(e) and 2(f). Moreover the extremes of the spine of the constructed tree are distinguished in step 2(g). The calculation of the values of δ for the edges of this graph is done in step 2(h) where the values of the non spine edges remain the same and the the values of the spine edges are defined by the sum of A'_1 and A'_2. After the calculation of the values of δ and the subtrees of Y corresponding to each of the edges $\hat{e}$ of $\hat{Y}_1$ the algorithm assembles all of them (step 3,4) to the output pair (γ, δ) using the information calculated in step 2(g). The correctness of the algorithm is supported by the following lemma.

Lemma 22. *Let G, G_1 and G_2 be graphs such that $G_1 \cup G_2 = G$ and $G_1 \cap G_2 = (S, \emptyset)$. For $i = 1, 2$, let γ_i be a carving of G_i and let $(\hat{\gamma}_i, \hat{\delta}_i) = C_S(G_i, \gamma_i)$. Then, for any bijection $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$ where $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$ (we assume that $\hat{\gamma}_i = (\hat{Y}_i, \hat{\theta}_i), i = 1, 2$) and any $\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2$ there exists a carving γ of G such that $C_S(G, \gamma) = (\hat{\gamma}_1, \hat{\delta}')$.*

Proof. It is enough to prove that if $(\hat{\gamma}_1, \hat{\delta}')$ is the output of the procedure Construct-Join-Carving then $C_S(G, \gamma) = (\hat{\gamma}_1, \hat{\delta}')$. In order to prove this it is crucial to show first that $\delta = d_{G,\gamma}$. For this, it is enough to show that if $\boldsymbol{e} = \langle t, t' \rangle$ is an edge of $\boldsymbol{E}(Y)$ then $\delta(\boldsymbol{e}) = |\alpha_{G,\gamma,V(G)}(\boldsymbol{e})|$.

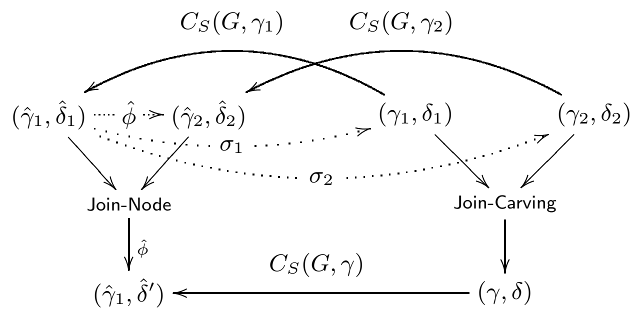

Fig. 9. The structure of Lemma 22.

In the case where e is not an edge of some spine $P_{Y_{\hat e}}(t^{\hat e}_{\text{start}}, t^{\hat e}_{\text{fin}})$ we have, from step 2(h), that either $e \in E(Y_1)$ or $e \in E(Y_2)$. W.l.o.g. we assume that $e \in E(Y_1)$. Notice that in this case, one, say $Y^{(1)}$, of the two connected components $Y^{(1)}$ and $Y^{(2)}$, of $Y - e$ is exactly the same as one, say $Y_1^{(1)}$, of the two connected components $Y_1^{(1)}$ and $Y_1^{(2)}$ of $Y_1 - e$. From the way θ is defined during step 3, the preimages of θ that are leaves of $Y^{(1)}$ are exactly the same as the preimages of θ_1 that are leaves of $Y_1^{(1)}$. Therefore, $\theta(A(Y) \cap A(Y^{(1)})) = \theta(A(Y_1) \cap A(Y_1^{(1)}))$. Combining this with the fact that $E(G_1) \cap E(G_2) = \emptyset$, we conclude that $\alpha_{G,\gamma,S}(e) = \alpha_{G_1,\gamma_1,S}(e)$. As $\delta(e) = |\alpha_{G_1,\gamma_1,S}(e)|$, we have the required.

What now remains is the case where e is the edge of some spine $P = P_{Y_{\hat e}}(t^{\hat e}_{\text{start}}, t^{\hat e}_{\text{fin}})$. From the way Y is assembled during steps 2(e) and 2(f), we have that either e is an edge of Y_1, or an edge of Y_2 or the result of the identification of two edges e^l_j, $e^s_{j+1} \in E(T_1) \cup E(T_2)$.

We examine first the case where, for some $i = 1, 2$, e_1 is an edge of Y_i or the result of the the identification of two edges $e^l_j, e^s_{j+1} \in E(T_i)$. W.l.o.g. we will only examine the case where $i = 2$. Notice that the removal of e from spine P brakes it into two parts each containing at least one edge. In each of these parts we detect the edges $e_{\text{left}}, e_{\text{right}}$ that are closest (in each direction) to e and with the property that are products of some of the the identifications of step 2(f). In particular steps 2(d)–d(f) indicate that, e_{left} and e_{right} are the results of the identification of the pairs (e^l_{j-1}, e^s_j) and $(e^l_{j'}, e^s_{j'+1})$ respectively, where $2 \le j \le j' \le |\hat\delta_1(\hat{\boldsymbol e})| + |\hat\delta_2(\hat{\boldsymbol e})| - 2$. Notice now that e^l_{j-1} and e^s_{j+1} are the same edge of Y_1. From now on we will call this edge e_1.

Let now $Y^{(1)}$ ($Y_i^{(1)}, i = 1, 2$) and $Y^{(2)}$ ($Y_i^{(2)}, i = 1, 2$) be the connected components of $Y - e$ ($Y_i - e_i, i = 1, 2$) where the indices are assigned with respect to the sence of direction defined by edges e, e_1, and e_2. Observe that, for $j = 1, 2$,

$$A(Y^{(j)}) = (A(Y_1) \cap A(Y_1^{(j)})) \cup (A(Y_2) \cap A(Y_2^{(j)})) \text{ and} \tag{27}$$

$$\theta(A(Y) \cap A(Y^{(j)})) = \theta(A(Y_1) \cap A(Y_1^{(j)})) \cup \theta(A(Y_2) \cap A(Y_2^{(j)})) \tag{28}$$

Using now (27), (28), and the fact that $E(G_1) \cap E(G_2) = \emptyset$ we observe that

$$\alpha_{G_1,\gamma_1,S}(e_1) \cup \alpha_{G_2,\gamma_2,S}(e_2) = \alpha_{G,\gamma,S}(e) \tag{29}$$

$$\alpha_{G_1,\gamma_1,S}(e_1) \cap \alpha_{G_2,\gamma_2,S}(e_2) = \emptyset \tag{30}$$

Using (29) and (30) we conclude that $|\alpha_{G,\gamma,S}(e)| = |\alpha_{G_1,\gamma_1,S}(e_1)| + |\alpha_{G_1,\gamma_2,S}(e_2)| = d_{G_1,\gamma_1,S}(\boldsymbol{e}_1) + d_{G_2,\gamma_2,S}(\boldsymbol{e}_2) = \delta_1(\boldsymbol{e}_1) + \delta_2(\boldsymbol{e}_2) = \delta(\boldsymbol{e})$.

$$
\begin{aligned}
\tilde{A}_1 &= [\;\overline{6\;6}\;\overline{9\;9\;9}\;\overline{1\;1\;1}\;] \quad \tilde{\delta}_1(\hat{e}) = [\;6\;9\;1\;] \\
\tilde{A}_2 &= [\;\underline{1}\;\underline{8\;8}\;\underline{2}\;\underline{7\;7}\;\underline{3\;5}\;] \quad \tilde{\delta}_2(\hat{e}) = [\;1\;8\;2\;7\;3\;5\;]
\end{aligned}
$$

$$
\begin{aligned}
\tilde{A}_1 &\in \mathcal{E}(\tilde{\delta}_1(\hat{e})) \qquad \tilde{A}_1 \sim \tilde{A}_1 \\
\tilde{A}_2 &\in \mathcal{E}(\tilde{\delta}_2(\hat{e})) \qquad \hat{\delta}'(\hat{e}) = \tau(\tilde{A}_1 + \tilde{A}_2) = \tau([\;7,14,17,11,16,8,4,6\;]) = [\;7,17,6\;] \in \hat{\delta}_1(\hat{e}) \bigotimes \hat{\delta}_2(\hat{e})
\end{aligned}
$$

$$
\begin{aligned}
A_1 &= [\;6\;\overbrace{8\;7\;6\;6\;8\;9\;7}^{\pi_1^1}\;9\;\overbrace{6\;7\;3\;4\;7\;3\;8}^{\pi_2^1}\;1\;] \\
A_2 &= [\;1\;\underbrace{5\;2\;3\;6}_{\pi_1^2}\;8\;\underbrace{5\;6\;6\;5\;6}_{\pi_2^2}\;2\;\underbrace{5\;2\;2\;3}_{\pi_3^2}\;7\;\underbrace{7\;5\;4\;6}_{\pi_4^2}\;3\;\underbrace{4\;5\;5\;5\;4\;5}_{\pi_5^2}\;5\;]
\end{aligned}
$$

$$
\begin{aligned}
A'_1 &= [\;6\;\overbrace{6\;6\;6\;6}^{|\pi_1^2|\times[6]}\;6\;\overbrace{8\;7\;6\;6\;8\;9\;7}^{\pi_1^1}\;9\;\overbrace{9\;9\;9\;9\;9}^{|\pi_2^2|\times[9]}\;9\;\overbrace{9\;9\;9\;9}^{|\pi_3^2|\times[9]}\;9\;\overbrace{6\;7\;3\;4\;7\;3\;8}^{\pi_2^1}\;1\;\overbrace{1\;1\;1\;1}^{|\pi_4^2|\times[1]}\;1\;\overbrace{1\;1\;1\;1\;1\;1}^{|\pi_5^2|\times[1]}\;1\;] \\
A'_2 &= [\;1\;\underbrace{5\;2\;3\;6}_{\pi_1^2}\;8\;\underbrace{8\;8\;8\;8\;8\;8\;8}_{|\pi_1^1|\times[8]}\;8\;\underbrace{5\;6\;6\;5\;6}_{\pi_2^2}\;2\;\underbrace{5\;2\;2\;3}_{\pi_3^2}\;7\;\underbrace{7\;7\;7\;7\;7\;7\;7}_{|\pi_2^1|\times[7]}\;7\;\underbrace{7\;5\;4\;6}_{\pi_4^2}\;3\;\underbrace{4\;5\;5\;5\;4\;5}_{\pi_5^2}\;5\;]
\end{aligned}
$$

$$
\begin{aligned}
A'_{\hat{e}} = [\;7,\; &\underbrace{11,8,9,12}_{\pi_1^2+(|\pi_1^2|\times[6])},\;14,\;\underbrace{16,15,14,14,16,17,15}_{\pi_1^1+(|\pi_1^1|\times[8])},\;17,\;\underbrace{14,15,15,14,15}_{\pi_2^2+(|\pi_2^2|\times[9])},\;11,\;\underbrace{14,11,11,12}_{\pi_3^2+(|\pi_3^2|\times[9])}, \\
&16,\;\underbrace{13,14,10,11,14,10,15}_{\pi_2^1+(|\pi_2^1|\times[7])},\;8,\;\underbrace{8,6,5,7}_{\pi_4^2+(|\pi_4^2|\times[1])},\;4,\;\underbrace{5,6,6,6,5,6}_{\pi_5^2+(|\pi_5^2|\times[1])},\;6\;].
\end{aligned}
$$

$$
\tau(A') = [\;7,17,6\;]
$$

Fig. 10. Example of the operation on sequences during the loop of step 2.

The remaining case is the case where e is the result of the identification of two edges $e_j^l \in E(T_1)$ and $e_{j+1}^s \in E(T_2)$. This case is exactly the same as the previous one if we define $e_1 = e_j^l$ and $e_2 = e_{j+1}^s$. This completes the proof of the fact that $\delta = d_{G,\gamma}$.

We return now to the proof of $C_S(G,\gamma) = (\hat{\gamma}_1, \hat{\delta}')$. Clearly, if we insist that the identifications of step 4 are all favorable to vertices that were vertices of $\hat{\gamma}$, the only remaining is to show that for any edge $\hat{\boldsymbol{e}} \in \boldsymbol{E}(\hat{Y}_1)$, $\hat{\delta}'(\hat{\boldsymbol{e}}) = \tau(A'_{\hat{e}})$. Notice that

$$A'_{\hat{e}} = A'_1 + A'_2, \tag{31}$$

$$\hat{\delta}'(\hat{\boldsymbol{e}}) = \tau(\tilde{A}_1 + \tilde{A}_2), \tag{32}$$

$$\tilde{A}_i \sqsubseteq A'_i,\; i = 1,2. \tag{33}$$

(31) follows from step 2(e), (32) follows from step 2(b), and (33) follows from the last line of step 2(d) and fact that for any $i = 1,2$ and any $j = 1,\ldots,|\hat{\delta}_i(\hat{\boldsymbol{e}})| - 1$, either $[A_i(s_j^i)] \prec \pi_j^i \prec [A_i(f_j^i)]$ or $[A_i(s_j^i)] \succ \pi_j^i \succ [A_i(f_j^i)]$.

(31)–(33) along with Lemma 7 imply that $\hat{\delta}'(\hat{\boldsymbol{e}}) = \tau(A'_{\hat{e}})$ and this completes the proof of the lemma. □

Lemma 23. *Let G, G_1, and G_2 be graphs where $G_1 \cup G_2 = G$ and $G_1 \cap G_2 = (S, \emptyset)$ and and let γ be a carving of G. We set $(\gamma_i, \delta_i^\star) = C_{V(G_i)}(G,\gamma), i = 1,2$, $(\hat{\gamma}_i, \hat{\delta}_i) = C_S(G_i, \gamma_i), i = 1,2$, and $C_S(G,\gamma) = (\bar{\gamma}, \bar{\delta})$.*

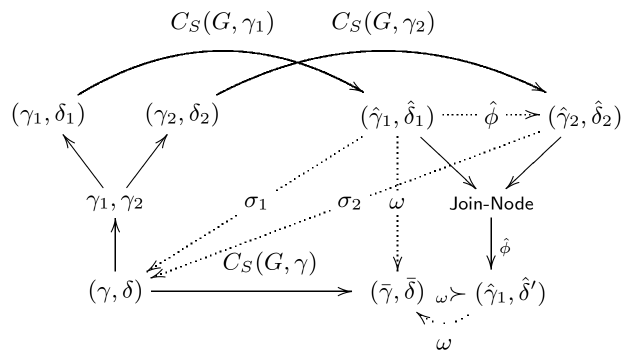

Fig. 11. The structure of Lemma 23.

We also denote $\bar{\gamma} = (\bar{Y}, \bar{\theta})$ *and* $(\hat{\gamma}_1, \hat{\delta}_i) = (\hat{Y}_i, \hat{\theta}_i), i = 1, 2$. *Then there exist two bijections* $\omega : V(\hat{Y}_1) \to V(\bar{Y})$, $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$ *and a characteristic pair* $(\hat{\gamma}_1, \hat{\delta}')$ *where*

1. $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$,
2. $\hat{\gamma}_1 \equiv_\omega \bar{\gamma}$,
3. $\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2$, *and*
4. $(\hat{\gamma}_1, \hat{\delta}') \prec_\omega (\bar{\gamma}, \bar{\delta})$.

Proof. We use the notations $C_S(G, \gamma) = (\bar{\gamma}, \bar{\delta})$, $\delta = d_{G,\gamma}$, $\delta_i = d_{G_i,\gamma_i}, i = 1, 2$, $\gamma = (Y, \theta)$ $\gamma_i = (Y_i, \theta_i), i = 1, 2$, and $\hat{\gamma}_i = (\hat{Y}_i, \hat{\theta}_i), i = 1, 2$. Notice first that, from procedure Com, $\hat{Y}_i, i = 1, 2$ is isomorphic to the minor of Y obtained if we first remove leaves not in S until no such leaves exist any more and then replace any poor path with an edge. This means that there is an isomorphic bijection $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$ between $\hat{Y}_1$ and $\hat{Y}_2$. Observing now the way $\theta_i, i = 1, 2$ are created, it is easy to see that if $\hat{v}_i \in A(\hat{Y}_i), i = 1, 2$ and $\hat{v}_2 = \hat{\phi}(\hat{v}_1)$, then $\hat{v}_1$ and $\hat{v}_2$ map, via $\hat{\theta}_1$ and $\hat{\theta}_2$ respectively, to the same vertex of S. Therefore, $\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2$. Define $\sigma_i : V(\hat{Y}_i) \to V(Y), i = 1, 2$ such that $\sigma_i = \chi_{\gamma,V(G_i)} \circ \chi_{\gamma_i,S}, i = 1, 2$ and notice that $\hat{\phi} = \sigma_2^{-1} \circ \sigma_1$. Similarly, one can see that $\omega = \chi^{-1}_{\bar{\gamma},S} \circ \sigma_1$ is a bijection mapping the vertices of $\hat{Y}_1$ to $\bar{Y}$ such that $\hat{\gamma}_1 \equiv_\omega \bar{\gamma}$.

We now fix an edge $\hat{\boldsymbol{e}}_1 = \langle \hat{u}_1, \hat{v}_1 \rangle$ of $\hat{Y}_1$ and set $\hat{\boldsymbol{e}}_2 = \langle \hat{u}_2, \hat{v}_2 \rangle = \langle \hat{\phi}(\hat{u}_1), \hat{\phi}(\hat{v}_1) \rangle$ and $(\bar{e}) = \langle \bar{u}, \bar{v} \rangle = \langle \omega(\hat{u}_1), \omega(\hat{v}_1) \rangle$. Let $\sigma_1(\hat{v}_1) = t_1$ and $\sigma_1(\hat{v}_1) = t_r$ where $(t_1, \ldots, t_{r_{\hat{e}_1,\hat{e}_2}}) = P_Y(t_1, t_r)$. We define $A = [\delta(\langle\!\langle t_1, t_2 \rangle\!\rangle), \ldots, \delta(\langle\!\langle t_{r_{\hat{e}_1,\hat{e}_2}-1}, t_{r_{\hat{e}_1,\hat{e}_2}} \rangle\!\rangle)]$ and we notice that

$$\tau(A) = \bar{\delta}(\bar{\boldsymbol{e}}). \tag{34}$$

We also set $A_i = [a^i_1, \ldots, a^i_{r-1}]$ where $a^i_j = |\alpha_{G,\gamma,V(G_i)}(\{t_j, t_{j+1}\})|, 1 \le j < r$. It is easy to see that

$$\tau(A_i) = \hat{\delta}_i(\hat{\boldsymbol{e}}_i), i = 1, 2. \tag{35}$$

Notice that the definitions of $\alpha_{G,\gamma,S}$ and $\delta = d_{G,\gamma}$ along with the fact that $(E(G_1), E(G_2))$ is a partition of $E(G)$, implies that for any $j, 1 \le j < r$,

$$(\alpha_{G,\gamma,V(G_1)}(\{t_j, t_{j+1}\}), \alpha_{G,\gamma,V(G_2)}(\{t_j, t_{j+1}\}))$$

is a partition of $\delta(\langle\!\langle t_j, t_{j+1} \rangle\!\rangle) = \alpha_{G,\gamma,V(G)}(\{t_j, t_{j+1}\})$. Therefore, we have that

$$A = A_1 + A_2. \tag{36}$$

From (34)–(36) and Lemma 9, we have that there exists a sequence $\hat{A}' \in \tau(A_1) \otimes \tau(A_2) = \hat{\delta}_1(\hat{\boldsymbol{e}}_1) \otimes \hat{\delta}_2(\hat{\boldsymbol{e}}_2)$ such that $\hat{A}' \prec \tau(A) = \bar{\delta}(\bar{\boldsymbol{e}}_1)$.

It is now straightforward that by defining $\hat{\delta}'$ so that for any edge $\hat{\boldsymbol{e}}_1$, $\hat{\delta}'$ is equal to the corresponding sequence $\hat{A}'$ we have a characteristic pair $(\hat{\gamma}_1, \hat{\delta}')$ where $\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2$ and $(\hat{\gamma}_1, \hat{\delta}') \prec_{\bar{\sigma}} (\bar{\gamma}, \bar{\delta})$ (a diagram illustrating the proof is depicted in Figure 11). □

Lemma 24. *If, for $i = 1, 2$, $FS(q_i)$ is a full set of characteristics of carvings for G_{q_i} and p is a* join *node with children $q_i, i = 1, 2$, then the set $FS(p)$ constructed by* Algorithm Join-Node *is a full set of characteristics of carvings for G_p.*

Proof. We will prove first that $FS(p)$ is a set of characteristics. To avoid overloaded expressions, whenever we refer to a carving or to a characteristic, we will insist that its width is bounded by k. For this, it is enough to show that for any $(\hat{\gamma}_1, \hat{\delta}') \in FS(p)$, there exists a carving γ of G such that $C_{X_p}(G, \gamma) = (\hat{\gamma}_1, \hat{\delta}')$.

By algorithm Join Node we can assume that, for $i = 1, 2$, there exist a X_{q_i}-characteristic $(\hat{\gamma}_i, \hat{\delta}_i) \in FS(q_i)$ of some carving γ_i for $G_{X_{q_i}}$ and a bijection $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$ where

$$\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2 \text{ and} \tag{37}$$

$$\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2. \tag{38}$$

Clearly,

$$(\hat{\gamma}_i, \hat{\delta}_i) = G_{X_{q_i}}(G_i, \gamma_i), i = 1, 2. \tag{39}$$

Using now (37)–(39), we can apply Lemma 22 and conclude that there exists a carving γ of G_p such that $C_{X_p}(G, \gamma) = (\hat{\gamma}_1, \hat{\delta}')$. Therefore, $FS(p)$ is a set of characteristics.

It remains now to prove that $FS(p)$ is a full set of characteristics. To prove this we have to show that, for any carving γ of G_p there exists a carving γ^* of G_p such that $C_{X_p}(G_p, \gamma^*) \in FS(p)$ and $C_{X_p}(G_p, \gamma) \prec C_{X_p}(G_p, \gamma^*)$. (For a diagram of the structure of the proof that follows, see Figure 12.)

Let $(\gamma_i, \delta_i^\star) = C_{V(G_{q_i})}(G_p, \gamma), i = 1, 2$ and notice that $\gamma_i = (Y_i, \theta_i)$ is a carving of $G_{q_i}, i = 1, 2$. We now set $(\hat{\gamma}_i, \hat{\delta}_i) = C_{X_{q_i}}(G_{q_i}, \gamma_i), i = 1, 2$ and $(\bar{\gamma}, \bar{\delta}) = C_{X_p}(G_p, \gamma)$. From Lemma 23, there exists two bijections $\hat{\phi} : V(\hat{Y}_1) \to V(\hat{Y}_2)$, $\hat{\phi} : V(\hat{Y}_1) \to V(\bar{Y})$, and a characteristic pair $(\hat{\gamma}_1, \hat{\delta}')$ where

$$\hat{\gamma}_1 \equiv_{\hat{\phi}} \hat{\gamma}_2, \tag{40}$$

$$\hat{\gamma}_1 \equiv_{\omega} \bar{\gamma}, \tag{41}$$

$$\hat{\delta}' \in \hat{\delta}_1 \bigotimes_{\hat{\phi}} \hat{\delta}_2, \text{ and} \tag{42}$$

$$(\hat{\gamma}_1, \hat{\delta}') \prec_{\omega} (\bar{\gamma}, \bar{\delta}). \tag{43}$$

Recall now that, for $i = 1, 2$, that $FS(q_i)$ is a full set of characteristics of carvings for G_{q_i} and therefore there exists a carving γ_i^* of G_{q_i} where

$$C_{X_{q_i}}(G_{q_i}, \gamma_i^*) \in FS(q_i) \text{ and} \tag{44}$$

$$C_{X_{q_i}}(G_{q_i}, \gamma_i^*) \prec C_{X_{q_i}}(G_{q_i}, \gamma_i). \tag{45}$$

Let $(\hat{\gamma}_i^*, \hat{\delta}_i^*) = C_{X_{q_i}}(G_{q_i}, \gamma_i^*), i = 1,2$. We can now assume that, for $i = 1,2$, there exists a function $\psi_i : V(\hat{Y}_i^*) \to V(\hat{Y}_i)$ such that (45) can be rewritten as follows.

$$(\hat{\gamma}_i^*, \hat{\delta}_i^*) \prec_{\psi_i} (\hat{\gamma}_i, \hat{\delta}_i). \tag{46}$$

(40), (42), (46), and Lemma 11, imply that there exists a characteristic pair $(\hat{\gamma}_1^*, \hat{\delta}'^*)$ and two bijections $\hat{\phi}^* : V(\hat{T}_1^*) \to V(\hat{T}_2^*)$ and $\psi : V(\hat{Y}^*) \to V(\hat{Y})$ such that

$$\hat{\gamma}_1^* \equiv_{\hat{\phi}^*} \hat{\gamma}_2^*, \tag{47}$$

$$\hat{\gamma}_1^* \equiv_{\psi} \hat{\gamma}_1, \tag{48}$$

$$\hat{\delta}'^* \in \hat{\delta}_1^* \bigotimes_{\hat{\phi}^*} \hat{\delta}_2^* \text{ and} \tag{49}$$

$$(\hat{\gamma}_1^*, \hat{\delta}'^*) \prec_{\psi} (\hat{\gamma}_1, \hat{\delta}'). \tag{50}$$

Notice now that, from (47), (49), and Lemma 22, there exists a carving γ^* of G such that $C_{X_p}(G_p, \gamma^*) = (\hat{\gamma}_1^*, \hat{\delta}'^*)$. The fact that $C_{X_p}(G_p, \gamma^*) \in FS(p)$ follows from (44), (49), (47) and Algorithm Join-Node. Finally, (41), (43), (48), and (50) imply that $C_{X_p}(G_p, \gamma^*) = (\hat{\gamma}_1^*, \hat{\delta}'^*) \prec_{\omega \circ \psi} (\bar{\gamma}, \bar{\delta}) = C_{X_p}(G_p, \gamma)$ and this completes the proof of the lemma. □

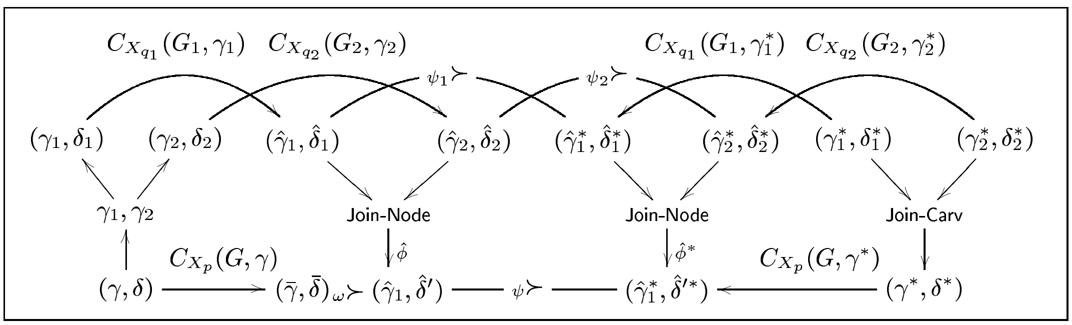

Fig. 12. The structure of Lemma 24.

5 Conclusions

Notice that, because of Lemma 3, both versions of the algorithms Introduce-node and Forget-node run in $O(1)$ time when k and w are fixed.

We resume the results of sections 3–4 in the following.

Theorem 1. *For all $k, w \geq 1$ there exists an algorithm that, given a graph G and a m-node tree decomposition X of G with width at most w, computes whether the carving-width of G is at most k and, if so, constructs a carving of G with width at most k and that uses $O(V(G) + m)$ time.*

Lemma 2 and Theorem 1, along with the results in [2] and [1], yield our central result:

Theorem 2. *For all k, there exists an algorithm, that given a graph G, computes whether the carving-width of G is at most k, and if so, constructs a carving of G with minimum width in $O(|V(G)|)$ time.*

References

1. H. L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25(6):1305–1317, 1996.
2. H. L. Bodlaender and T. Kloks. Efficient and constructive algorithms for the pathwidth and treewidth of graphs. *J. Algorithms*, 21:358–402, 1996.
3. H. L. Bodlaender and D. M. Thilikos. Constructive linear time algorithms for branchwidth. In P. Degano, R. Gorrieri, and A. Marchetti-Spaccamela, editors, *Proceedings 24th International Colloquium on Automata, Languages, and Programming*, pages 627–637. Springer Verlag, Lecture Notes in Computer Science, vol. 1256, 1997.
4. H. L. Bodlaender and D. M. Thilikos. Computing small search numbers in linear time. Technical Report UU-CS-1998-05, Dept. of Computer Science, Utrecht University, 1998.
5. M. R. Fellows and M. A. Langston. On well-partial-order theory and its application to combinatorial problems of VLSI design. *SIAM J. Disc. Math.*, 5:117–126, 1992.
6. S. Khuller, B. Raghavachari, and N. Young. Designing multicommodity flow trees. *Information Processing Letters*, 50:49–55, 1994.
7. N. Robertson and P. D. Seymour. Graph minors. XXII. The Nash-Wiliams immersion conjecture. To appear in *J. Combin. Theory Ser. B.*
8. N. Robertson and P. D. Seymour. Graph minors. XIII. The disjoint paths problem. *J. Comb. Theory Series B*, 63:65–110, 1995.
9. P. D. Seymour and R. Thomas. Call routing and the ratcatcher. *Combinatorica*, 14(2):217–241, 1994.
10. D. M. Thilikos, M. J. Serna, and H. L. Bodlaender. A constructive linear time algorithm for small cutwidth. Technical Report LSI-00-48-R, Departament de Llenguatges i Sistemes Informatics, Universitat Politecnica de Catalunya, Barcelona, Spain, 2000.