UNIVERSITE D'ORLEANS
Faculté des Sciences
LIFO

Laboratoire d'Informatique Fondamentale d'Orléans
4, rue Léonard de Vinci, BP 6759
F-45067 Orléans Cedex 2
FRANCE

Rapport de Recherche

www : http://www.univ-orleans.fr/SCIENCES/LIFO/

Coloring powers of graphs of bounded clique-width

Ioan Todinca
LIFO, Université d'Orléans

Rapport N^o **2002-10**

Coloring powers of graphs of bounded clique-width

Ioan Todinca

LIFO - Université d'Orléans

Abstract. Given a graph G, the graph G^l has the same vertex set and two vertices are adjacent in G^l if and only if they are at distance at most l in G. The l-coloring problem consists in finding an optimal vertex coloring of the graph G^l, where G the input graph. We show that, for any fixed value of l, the l-coloring problem is polynomial when restricted to graphs of bounded clique-width, if an expression of the graph is also part of the input.

1 Introduction

The classical coloring problem on graphs consists in finding an assignement of colors to the vertices of the input graph such that any adjacent vertices use different colors and the number of colors used for the coloring is as small as possible.

Many versions of this problem were proposed as models for real-life applications. One of them is the radio-coloring problem (also called λ-coloring or $L(2,1)$-coloring), in which the vertices of the graph represent transmitters, and two vertices are adjacent if the two transmitters are very close. We want to assign to each vertex a integer, representing the frequency of the transmitter, in such a way that adjacent vertices (very close transmitters) get frequencies differing by at least two units, and vertices at distance two in the graph (close transmitters) get different frequencies. In the radio-coloring problem, we want to minimize the range of the used frequencies, i.e. we search for the minimum λ such that G admits a radio-coloring with frequencies between 0 and λ. Let us call $\lambda_{2,1}(G)$ this minimum. Another generalization of the classical coloring is the l-coloring problem. An l-coloring of a graph $G = (V, E)$ is an assignment of colors to the vertices of G such that any two vertices at distance at most l in G receive different colors. An l-coloring of G is optimal if it uses the minimum number of colors. Equivalently, if $G^l = (V, E^l)$ is the l^{th} power of G (two vertices are adjacent in G^l if and only if they are at distance at most l in G), then the l-colorings of G are exactly the classical colorings of G^l.

The radio-coloring and the l-coloring problems are related: the number $\lambda_{1,1}$ of colors necessary for a 2-coloring of G is at least $\lambda_{2,1}(G) + 1$, and conversely from an optimal 2-coloring of G we can easily obtain a radio-coloring of G with frequencies between 0 and $2\lambda_{1,1} - 1$. Hence, an algorithm solving the 2-coloring problem for a class of graphs also provides a 2-approximation for the radio-coloring problem.

It is well-known that all these problems are NP-hard for general graphs [1, 6]. The radio-coloring problem is polynomial for trees and cographs [3] and as far as I know it is open for partial k-trees and even for outerplanar graphs, which are partial 2-trees. The l-coloring problem is polynomial for partial k-trees, for any fixed k and l [8].

In this paper we study the l-coloring problem on graphs of bounded clique-width. The clique-width of a graph G, denoted by $\mathrm{cwd}(G)$, is the minimum number of labels needed to construct the graph G, using four operators: $\bullet$, $\oplus$, ρ and η. The operation $\bullet_i$ creates a graph with a single vertex labeled i. The binary operator $\oplus$ constructs the union of two disjoin graphs. The operation $\rho_{i \to j}$ renames all vertices labeled i with label j. The operation $\eta_{i,j}$, with $i \neq j$, adds all the edges between every vertex of label i and every vertex of label j. Clique-width bounded graphs are particulary interesting from an algorithmic point of view. Many NP-hard problems become polynomially solvable for graphs of bounded clique-width, if an expression of the graph is part of the input. The classical coloring problem is one of them [5, 7]. Let us note that the partial k-trees are a particular case of graphs of bounded clique-width, more precisely if G is a partial k-tree than $\mathrm{cwd}(G) \leq 2^{k+1}+1$ [4]. Several other classes of graphs (distance-hereditary graphs, P_4-sparse and P_4-tidy graphs, ...) are known to have bounded clique-width.

The initial motivation for this work was the radio-coloring problem for graphs of bounded clique-width, but it turns out that the problem is NP-complete even for graphs of clique-width at most 3 [2]. We show in this article that the l-coloring problem is polynomial when restricted to graphs with bounded clique-width, if an expression of the graph is part of the input. As mentioned above, this provides a 2-approximation for the radio-coloring problem on this class of graphs.

The paper is organized as follows. After introducing the basic notions in Section 2, we show in Section 3 that, for any graph G of clique-width k, the clique-width of G^2 is at most $k2^{k+1}$. We also state that the clique-width of G^l is at most $2kl^k$. This result directly implies that, for fixed k and l, the l-coloring problem is solvable in polynomial time when restricted to graphs of clique-width at most k if an expression of G is part of the input. But the time complexity is of type $\mathcal{O}(n^{2^{2kl^k+1}})$, where n is the number of vertices of the input graph.

In Section 4 we give an algorithm for the l-coloring problem with a $\mathcal{O}(n^{2^{2kl}+2^{kl+1}+1})$ time bound. Although the complexity remains extremely huge it is significantly better than the previous one. Moreover, this time bound is to be compared with the existant time bound for classical coloring problem on graphs with clique-width at most k ($\mathcal{O}(n^{2^{2k+1}})$, see [7]) or with the time-bound of the l-coloring problem on partial k-trees ($\mathcal{O}(n^{2^{2(k+1)(l+2)+1}})$, see [8]).

2 Basic definitions

Throughout this paper we consider simple, finite, undirected graphs. Given a graph $G=(V,E)$, we denote by n the number of vertices of G. An edge $\{x,y\}$ will be simply denoted by xy. A path $\mu=[x_1,\dots,x_l]$ is a sequence of vertices such that $x_ix_{i+1}\in E$, $\forall i, 1\leq i<l$. The length $|\mu|$ of this path is $l-1$.

The distance between two vertices x and y of G, denoted by $d_G(x,y)$, is the length of the shortest path from x to y. We define by $G^l=(V,E^l)$ the l^{th} power of G, i.e. $E^l=\{xy \mid d_G(x,y)\leq l\}$.

Let $[k]=\{1,\dots,k\}$ be the set of integers from 1 to k. A k-labeled graph $G=(V,E,\mathrm{lab})$ is a graph (V,E) whose vertices are labeled by some mapping $\mathrm{lab}: V\to[k]$. Given a labeled graph $G=(V,E,\mathrm{lab})$, we denote by $\mathrm{unlab}(G)$ the unlabeled graph (V,E).

Definition 1 (Clique-width [4]). *Let k be a positive integer. The class CW_k of k-labeled graphs is defined as follows.*

1. *For any $i\in[k]$, the graph $\bullet_i$ with a single vertex labeled i is in CW_k.*
2. *Given two disjoint graphs $G=(V_G,E_G,\mathrm{lab}_G)$ and $H=(V_H,E_H,\mathrm{lab}_H)$ in CW_k, their disjoint union $G\oplus H=(V',E',\mathrm{lab}')$ defined by $V'=V_G\cup V_H$, $E'=E_G\cup E_H$, and*

$$\mathrm{lab}(x)=\begin{cases}\mathrm{lab}_G(x) & \text{if } x\in V_G\\ \mathrm{lab}_H(x) & \text{if } x\in V_H\end{cases},\forall x\in V'$$

   is in CW_k.
3. *Given a labeled graph $G=(V,E,\mathrm{lab})\in CW_k$ and two distinct integers $i,j\in[k]$, the graphs*

(a) $\rho_{i\to j}(G) = (V, E, lab')$, *where*

$$lab'(x) = \begin{cases} lab(x) & \text{if } lab(x) \neq i \\ j & \text{if } lab(x) = i \end{cases}$$

and

(b) $\eta_{i,j}(G) = (V, E', lab)$ *where* $E' = E \cup \{xy \mid x, y \in V, lab(x) = i, lab(y) = j\}$ *are in* CW_k.

An unlabeled graph $G = (V, E)$ *is in* CW_k *if there is a labeling* lab *of* G *such that* $(V, E, \mathrm{lab}) \in CW_k$.

The clique-width *of a graph* $G = (V, E)$, *denoted* $\mathrm{cwd}(G)$, *is the smallest positive integer* k *such that* $G \in CW_k$.

To any graph $G \in CW_k$ we can associate an expression tree T using the operators $\bullet_i, \oplus, \rho_{i\to j}$ and $\eta_{i,j}$, with distinct integers $i, j \in [k]$. We say that T is a CW_k-expression tree and that G is the value $\mathrm{val}(T)$ of T.

3 Clique-width and powers of graphs

We show in this section that, for any a graph $G \in CW_k$, the clique-width of G^2 is at most $k2^{k+1}$. At the end of the section we discuss a similar bound for $\mathrm{cwd}(G^l)$.

Notice that, in the definition of the class CW_k, the operator $\eta_{i,j}$ uses *distinct* integers i and j. For technical reasons, we want to allow the operator $\eta_{i,i}$.

Definition 2. *Let* CW'_k *the class of* k*-labeled graphs defined like* CW_k, *but allowing the operation* $\eta_{i,i}$ *for* $i \in [k]$.

Clearly any graph $G \in CW_k$ is also in CW'_k, but the converse is not true: a clique having $n \geq 2$ vertices is in CW'_1 and in CW_2, but not in CW_1. Nevertheless, we have the following proposition:

Proposition 1. *For any positive integer* k, *if* $G \in CW'_k$ *then* $G \in CW_{2k}$.

The proof of Proposition 1 is given in Appendix A.

Theorem 1. *Consider a labeled graph* $G \in CW_k$ *and let* T *be a* CW_k*-expression tree of* G. *We can construct a* CW'*-expression tree* $f(T)$ *with labels in* $[k] \times \{0,1\}^k$ *such that* $\mathrm{unlab}(\mathrm{val}(f(T))) = (\mathrm{unlab}(G))^2$ *and, for each* $x \in V$, *the label* $(i, b_1, b_2, \ldots, b_k)$ *of* x *in* $\mathrm{val}(f(T))$ *has the following properties:*

1. *i is the label of x in G;*
2. *for any $j \in [k]$, $b_j = 1$ if and only if x has a neighbor with label j in G.*

Proof. The tree $f(T)$ is obtained by parsing the tree T, from bottom to top. In the construction of $f(T)$, we distinguish four cases, depending whether the root of T is a $\bullet$ node, a ρ node, a η node or a $\oplus$ node.

The root of T is a $\bullet$ node. If T has a unique node $\bullet_i$, then $f(T) = \bullet_{(i,0,0,\ldots,0)}$.

The root of T is a $\oplus$ node. If $T = T_1 \oplus T_2$, then $f(T) = f(T_1) \oplus f(T_2)$.

The root of T is a $\rho_{i\to j}$ node. Let $T=\rho_{i\to j}(T_1)$. In order to obtain $f(T)$ from $f(T_1)$, we transform each label $L=(p,b_1,\dots,b_k)$ of $f(T_1)$ into a label $L'=(p',b'_1,\dots,b'_q)$ such that:

1. $p'=\begin{cases} j & \text{if } p=i \\ p & \text{otherwise}\end{cases}$
2. $b'_i=0$, $b'_j=\max(b_i,b_j)$ and $b'_q=b_q$, $\forall q\in[k]-\{i,j\}$

The second condition expresses the fact that, for any vertex x of G, x has no neighbor with label i, and x has a neighbor with label j if and only if, before the $\rho_{i\to j}$ operation, x had a neighbor labeled i or a neighbor labeled j. Let $(L_1,L'_1),(L_2,L'_2),\dots,(L_q,L'_q)$ be the couple of distinct labels of $f(T_1)$ satisfying the two conditions above. Then

$$f(T)=\rho_{L_1\to L'_1}(\rho_{L_2\to L'_2}(\dots(\rho_{L_q\to L'_q}(f(T_1))))).$$

The root of T is a $\eta_{i,j}$ node. In this case, $T=\eta_{i,j}(T_1)$, and again we construct $f(T)$ from $f(T_1)$.

Let $G_1=\mathrm{val}(T_1)$. Let V_i be the set of vertices labeled i in G_1 and let N_i be the set of vertices of G_1 having at least one neighbor in V_i (*the neighborhood is taken in the graph* G_1). We define V_j and N_j in a similar way. If V_i or V_j are empty, then $G=G_1$, so $f(T)=f(T_1)$. Suppose now that V_i and V_j are not empty. Consider an edge xy of G^2 which is not an edge of G_1^2. If $\mathrm{dist}_G(x,y)=1$, then (x,y) is contained in $V_i\times V_j$ or $V_j\times V_i$. If $\mathrm{dist}_G(x,y)=2$, there is a path $[x,z,y]$ of length 2 in G which does not exist in G_1, so $z\in V_i\cup V_j$. Hence (x,y) or (y,x) is in one of the following sets: $V_i\times N_j$, $V_j\times N_i$, $V_i\times V_i$ or $V_j\times V_j$. Conversely, since V_i and V_j are not empty, for any couple (x,y) in one of these four sets, we observe that x and y are at distance at most 2 in G. We take all the couples of labels (L,L') of $f(T_1)$, $L=(p,b_1,\dots,b_k)$, $L'=(p',b'_1,\dots,b'_k)$ satisfying one of the following conditions: $(p,p')\in\{(i,j),(i,i),(j,j)\}$ or ($p=i$ and $b'_j=1$) or ($p=j$ and $b'_i=1$). Let $(L_1,L'_1),\dots(L_q,L'_q)$ be these couples, we take

$$T'_1=\eta_{L_1,L'_1}(\eta_{L_2,L'_2}(\dots(\eta_{L_p,L'_p}(f(T_1))))).$$

Then the unlabeled graph corresponding to the expression tree T'_1 is exactly G^2. Observe that T'_1 may contain nodes of type $\eta_{L,L'}$ with $L=L'$ – and this is the reason why $f(T)$ is a CW'-expression tree, but not necessarily a CW-expression tree.

It remains to update the labels of T'_1, in order to satisfy the conditions 1 and 2 of our proposition. Each vertex labeled i (resp. j) in G has at least one neighbor labeled j (resp. i). Thus, each label $L=(i,b_1,b_2,\dots,b_k)$ (resp. $L=(j,b_1,b_2,\dots,b_k)$) is transformed into $L'=(i,b_1,\dots,b_{j-1},1,b_{j+1},\dots,b_k)$ (resp. $L'=(j,b_1,\dots,b_{i-1},1,b_{i+1},\dots,b_k)$) and all the other labels remain unchanged. We denote by $(L_1,L'_1),\dots,(L_q,L'_q)$ the couples of distinct labels of $f(T_1)$ satisfying the conditions above. Then

$$f(T)=\rho_{L_1\to L'_1}(\rho_{L_2\to L'_2}(\dots(\rho_{L_q\to L'_q}(T'_1)))).$$

□

From Theorem 1 and Proposition 1 we deduce:

Theorem 2. *For any graph G, $\mathrm{cwd}(G^2)\le cwd(G)\times 2^{\mathrm{cwd}(G)+1}$.*

The result of Theorem 1 can be extended to arbitrary powers of G. From a CW_k-expression tree T of G, we can construct a CW'_{kl^k}-expression tree $f_l(T)$ of G^l, by generalizing the previous construction as follows. The labels of $f_l(T)$ are taken in the set $[k]\times[l]^k$. For any vertex $x\in V$ its label $(p,b_1,\dots,b_k)$ in $val(f_l(T))$ is such that $p=\mathrm{lab}_G(x)$ and $b_i=\min(l,\min\{y\in V\mid \mathrm{lab}_G(y)=i\})$ for all $i\in[k]-\{p\}$ (b_p remains unset). We do not give here the proof of this result, which is significantly more complicated that the one of Theorem 1, but the ingredients of the proof are similar to those used in the next section for the l-coloring problem.

Theorem 3. *For any graph G and any integer $l \geq 2$,*

$$\mathrm{cwd}(G^l) \leq 2cwd(G) \times l^{\mathrm{cwd}(G)}.$$

Moreover, given a CW_k expression of G, a CW_{2kl^k} expression of G^l can be constructed in polynomial time.

These upper bounds for the clique-width of G^l might not be tight. Nevertheless, we conjecture that any upper bound for $\mathrm{cwd}(G^l)$ is exponential in $\mathrm{cwd}(G)$.

Conjecture 1. There is a constant $\alpha > 0$ such that, for any positive integer k, there is a graph $G_k \in CW_k$ such that $\mathrm{cwd}(G_k^2) \geq 2^{\alpha k}$.

More precisely, consider the graph $G(q+2)$, having q vertices $\{x_1, \ldots, x_q\}$ and 2^q vertices $\{y_Z \mid Z \subseteq [q]\}$. The neighborhood of each vertex y_Z is $\{x_i \mid i \in Z\}$ and G contains no other edges. It is easy to see that G_{q+2} is of clique-width at most $q+2$ and we conjecture that the clique-width of the square of G_{q+2} is exponential in q.

4 The l-coloring problem for graphs of bounded clique-width

Given a graph $G \in CW_k$ and a k-expression of G, one can compute a coloring of G using the algorithm of Kobler and Rotics in $\mathcal{O}(2^{3k+1}k^2n^{2^{2k+1}+1})$ time [7]. A different solution of the problem is given in [5]. We conclude that, for graphs of bounded clique-width and for bounded l, the chromatic number of G^l can be computed in polynomial time (if the CW-expression tree of G is also given). But the time bounds are extremely huge, we obtain a complexity of type $\mathcal{O}(n^{2^{2kl^k+1}+1})$.

In this section, we give an algorithm for the l-coloring problem, with a $\mathcal{O}(n^{2^{2kl}+2^{kl+1}+1})$ time bound. The technique used here generalizes the coloring algorithm for graphs of bounded clique-width proposed by Kobler and Rotics. The data structures and the general framework of the algorithm for the l-coloring problem (Subsections 4.1 to 4.3) are similar to the ones used in [7], but the treatment of the $\oplus$ operation (Subsections 4.5 and 4.6) is completely different in our case.

4.1 A sketch of the algorithm

For computing a l-coloring of G, we parse the expression tree T of G from bottom to top. We may assume that T has at most k^2n nodes (see [7]). For each node u of T, let $T[u]$ denote the subtree of T rooted in u and let $G[u] = (V[u], E[u], \mathrm{lab}_u)$ be the graph $\mathrm{val}(T[u])$. We compute for each node u an information $F(u)$, representing some colorings of $G[u]$. More precisely, let us say that a coloring c of $G[u]$ is *admissible* with respect to G if, for any $x, y \in V[u]$ such that $\mathrm{dist}_G(x,y) \leq l$, the vertices x and y have different colors. Let us point out the that the distance between x and y is taken in G and *not* in $G[u]$.The information $F(u)$ kept in the node u represents all the admissible colorings of $G[u]$. In particular, the root r of T will contain in $F(r)$ the information about the optimal l-colorings of G.

Subsection 4.2 gives a complete description of $F(u)$ and shows that $F(u)$ has polynomial size. In Subsection 4.3, we show how to compute this information in polynomial time.

4.2 Encoding of $\mathbf{F(u)}$

Let us define now the information $F(u)$.

We denote by $\langle l \rangle$ the set of integers $\{0, 1, \ldots, l-1\}$. Given a label $i \in [k]$ and a number $d \in \langle l \rangle$, let

$$V_{i,d}[u] = \{x \in V[u] \mid \exists y \in V[u] \text{ such that } lab_u(y) = i \text{ and } \operatorname{dist}_G(x, y) \leq d\}$$

So $V_{i,0}[u] = V_i[u]$ denotes the set of vertices having label i in $G[u]$, and $V_{i,d}[u]$ is the set of vertices at distance at most d from $V_i[u]$, where the distance is taken in the graph G and not $G[u]$.

Consider a graph H and a coloring c of H. Let K be a finite set. For each $a \in K$ let X_a be a subset of $V(H)$ and let $\mathcal{X}$ be the family of not necessarily distinct sets $\{X_a \mid a \in K\}$. We suppose that the family covers $V(H)$, i.e. each vertex of $V(H)$ is in some set X_a of $\mathcal{X}$. Consider a subset $\mathcal{B}$ of K, we denote by $\mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$ the set of colors appearing in $c(X_a)$ for each $a \in \mathcal{B}$ and not appearing in $c(X_b)$ for each $b \notin \mathcal{B}$. We associate to c an array $N^c_{\mathcal{X}}$ with $2^{|K|}$ elements, defined by $N^c_{\mathcal{X}}(\mathcal{B}) = |\mathcal{C}^c_{\mathcal{X}}(\mathcal{B})|$ for all subsets $\mathcal{B}$ of K. Observe that the sets $\mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$ form a partition of the colors of c. Indeed, for any color *col* used by c, there is a unique subset $\mathcal{B}_{col}$ of K such that $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B}_{col})$. More precisely, $\mathcal{B}_{col} = \{a \in K \mid col \in c(X_a)\}$. Therefore, the number of colors used by c is

$$\sum_{\mathcal{B} \subseteq K} N^c_{\mathcal{X}}(\mathcal{B}).$$

Given a node u of the tree T, we use the above definitions with $H = G[u]$ and $K = [k] \times \langle l \rangle$. For each $(i, d) \in [k] \times \langle l \rangle$, $X_{(i,d)} = V_{i,d}[u]$, so the family $\mathcal{X}[u]$ is the family of sets $\{V_{i,d} \mid (i, d) \in [k] \times \langle l \rangle\}$. Clearly, this family covers $V[u]$. To simplify the notation, for any coloring c of $G[u]$ and any $\mathcal{B} \subseteq [k] \times \langle l \rangle$ we denote by $\mathcal{C}^c_u(\mathcal{B})$ the set $\mathcal{C}^c_{\mathcal{X}[u]}[\mathcal{B}]$. Let $N^c_u(\mathcal{B})$ be the cardinality of $\mathcal{C}^c_u(\mathcal{B})$. A matrix N *represents a coloring c of $G[u]$ in the node u* if $N = N^c_u$. A matrix may represent several colorings, but all these colorings use the same number of colors. If c is an admissible coloring with respect to G, then we say that $N = N^c_u$ is admissible. We will see later that, given a matrix N representing some colorings in the node u, either all these colorings are admissible, or they are all non admissible w.r.t. G.

The information $F(u)$ related to the node u is precisely the set of all the admissible matrices w.r.t. u. Since each admissible matrix is an array with 2^{kl} elements in $\{0, \ldots, |V[u]|\}$, the size of $F(u)$ is $\mathcal{O}((|V[u]| + 1)^{2^{kl}})$.

4.3 Computation of $\boldsymbol{F(u)}$

First case: u is a $\bullet$ node. When u is a leaf node $\bullet_i$, let $\mathcal{B}_0 = \{(i, d) \mid 0 \leq d \leq l-1\}$. Clearly, the only admissible matrix N satisfies $N(\mathcal{B}_0) = 1$ and $N(\mathcal{B}) = 0$ for all subsets $\mathcal{B} \neq \mathcal{B}_0$ of $[k] \times \langle l \rangle$.

Second case: u is a ρ node. Let v be the son of u in T. The proof of the following lemma is straightforward.

Lemma 1. *Suppose that the node u is labeled $\rho_{i \to j}$ and let v be the son of u in T. Then, for each $d \in \langle l \rangle$, we have that $V_{i,d}[u] = \emptyset$, $V_{j,d}[u] = V_{j,d}[v] \cup V_{i,d}[v]$ and $V_{q,d}[u] = V_{q,d}[v]$, $\forall q \in [k] - \{i, j\}$.*

For computing the admissible matrices of $F(u)$ from the admissible matrices of $N(v)$ we use the following lemma – which is also stated, in different terms, in [7]:

Lemma 2 (See also [7]). *Let H be a graph, c a coloring of H and $\mathcal{X} = \{X_q \mid q \in K\}$ a family of subsets of $V(H)$, covering $V(H)$. Consider two distinct elements $a, b \in K$ and a new family $\mathcal{Y} = \{Y_q \mid q \in K\}$ defined by $Y_a = \emptyset$, $Y_b = X_a \cup X_b$ and $Y_q = X_q$ for any $q \in K - \{a, b\}$.*

Then, for any $\mathcal{B} \subseteq K$,

1. $N^c_{\mathcal{Y}}(\mathcal{B}) = 0$ *if* $a \in \mathcal{B}$.
2. $N^c_{\mathcal{Y}}(\mathcal{B}) = N^c_{\mathcal{X}}(\mathcal{B})$ *if a and b are not in* $\mathcal{B}$.
3. $N^c_{\mathcal{Y}}(\mathcal{B}) = N^c_{\mathcal{X}}(\mathcal{B}) + N^c_{\mathcal{X}}(\mathcal{B} \cup \{a\}) + N^c_{\mathcal{X}}(\mathcal{B} - \{b\} \cup \{a\})$ *if $a \notin \mathcal{B}$ and $b \in \mathcal{B}$.*

Proof. Let $\mathcal{B} \subseteq K$. A color col used by c is in $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$ if and only if $\mathcal{B} = \{q \in K \mid col \in c(Y_q)\}$.

If $a \in \mathcal{B}$, since Y_a is empty we have that $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$ is empty, so $N^c_{\mathcal{Y}}(\mathcal{B}) = 0$.

If $a, b \notin \mathcal{B}$, we show that $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B}) = \mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$. A color $col \in \mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$ does not belong to $c(Y_b)$, so $col \notin c(X_a) \cup c(X_b)$. For any $q \in K - \{a, b\}$, $col \in c(X_q)$ iff $q \in \mathcal{B}$. Hence, $\mathcal{B} = \{q \in K \mid col \in c(X_q)\}$, so $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$. Conversely, let $col' \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$, so col' does not appear in $c(X_a)$ or $c(X_b)$ and thus $col' \notin c(Y_b)$. Clearly $col' \notin Y_a$ and $\forall q \in K - \{a, b\}$, $col' \in c(Y_q)$ iff $q \in \mathcal{B}$. It follows that $\mathcal{B} = \{q \in K \mid col' \in Y_q\}$ and thus $col' \in \mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$. We conclude that, if $a, b \notin \mathcal{B}$ then $N^c_{\mathcal{Y}}(\mathcal{B}) = N^c_{\mathcal{X}}(\mathcal{B})$.

Consider the case $a \notin \mathcal{B}$, $b \in \mathcal{B}$. For any color $col \in \mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$, $col \in c(Y_b)$ so $col \in c(X_a \cup X_b) = c(X_a) \cup c(X_b)$. Thus, col is in exactly one of the three sets $c(X_a) - c(X_b)$, $c(X_a) \cap c(X_b)$ or $c(X_b) - c(X_a)$. In the first case, $\forall q \in K$, $col \in X_q$ iff $q = a$ or $q \in \mathcal{B} - \{b\}$, so $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} - \{b\} \cup \{a\})$. For similar reasons, $col \in c(X_a) \cap c(X_b)$ implies that $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} \cup \{a\})$ and $col \in c(X_b) - c(X_a)$ leads to $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$. Hence, $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B}) \subseteq \mathcal{C}^c_{\mathcal{X}}(\mathcal{B}) \cup \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} \cup \{a\}) \cup \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} - \{b\} \cup \{a\})$. Conversely, for any color $col' \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B}')$ with $\mathcal{B}' \in \{\mathcal{B}, \mathcal{B} \cup \{a\}, \mathcal{B} \cup \{a\} - \{b\}\}$, we have that $col' \in c(X_a \cup X_b)$ and, for any $q \in K - \{a, b\}$, $col' \in c(X_q) = c(Y_q)$ iff $q \in \mathcal{B} - \{b\}$. We deduce that $col' \in \mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$. We have proved that $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B}) = \mathcal{C}^c_{\mathcal{X}}(\mathcal{B}) \cup \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} \cup \{a\}) \cup \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} - \{b\} \cup \{a\})$. The three sets of the right member of the equality are disjoints, which directly implies the third condition of the theorem. □

Clearly, $\text{unlab}(G[u]) = \text{unlab}(G[v])$, so a coloring c of $G[u]$ is admissible with respect to u if and only if the same coloring is admissible with respect to v. Given an admissible coloring c of $G[v]$, we have to compute the matrix N^c_u from N^c_v. We consider the family $\mathcal{X} = \{V_{i',d'}[v] \mid (i', d') \in [k] \times \langle l \rangle\}$, so $N^c_{\mathcal{X}} = N^c_v$. For each $d \in \langle l \rangle$ we replace in $\mathcal{X}$ the set $V_{i,d}[v]$ by $V_{i,d}[u] = \emptyset$ and the set $V_{j,d}[v]$ by $V_{j,d}[v] = V_{i,d}[v] \cup V_{j,d}[v]$ and we compute the matrix $N^c_{\mathcal{X}}$ for this new family using Lemma 2. According to Lemma 1, at the end of this loop we obtain the matrix N^c_u.

Third case: u is a η node If v is the son of u in T, clearly $V[u] = V[v]$ and the labels of any vertex $x \in V[u]$ are the same in $G[u]$ and in $G[v]$. Since the admissible colorings of $G[u]$ are exactly the admissible colorings of $G[v]$, we have $F(u) = F(v)$.

Fourth case: u is a $\oplus$ node. Let v and w be the two children of u in T. The algorithm for computing $F(u)$ proceeds by considering each pair of matrices $(N_1, N_2) \in F(v) \times F(w)$ and by computing all the matrices $N \in F(u)$ such that N represents an admissible coloring c of $G[u]$ whose restriction to $G[v]$ (resp. $G[w]$) is represented by N_1 (resp. N_2).

Consider a coloring c of $G[u]$ and let e (resp. f) be the restriction of c to $G[v]$ (resp $G[w]$). Let $N_1 = N^e_v$ (resp. $N_2 = N^f_w$) be the matrices representing e (resp. f) in the node v (resp. w). We associate to c an array S defined as follows. For any couple of non-empty sets $\mathcal{B}_1, \mathcal{B}_2 \in [k] \times \langle l \rangle$, let $S(\mathcal{B}_1, \mathcal{B}_2)$ be the number of colors occuring both in $\mathcal{C}^e_v(\mathcal{B}_1)$ and $\mathcal{C}^f_w(\mathcal{B}_2)$. Let $S(\mathcal{B}_1, \emptyset)$ (resp. $S(\emptyset, \mathcal{B}_2)$) be the number of colors appearing in $\mathcal{C}^e_v(\mathcal{B}_1)$ (resp. $\mathcal{C}^f_w(\mathcal{B}_2)$) and not appearing in $f(G[w])$ (resp. $e(G[v])$). We put $S(\emptyset, \emptyset) = 0$. We say that the matrices S, N_1 and N_2 *represent* *the coloring c of $G[u]$ in the nodes (v, w).* Indeed, in subsection 4.6 we will see that the matrix N_u^c only depends on N_1, N_2 and S and we give an explicit algorithm for computing this matrix.

The following proposition is also given in a slightly different form in [7]. Nevertheless, we give here its proof for a better understanding of the algorithm.

Proposition 2 (See also [7]). *Let c be a coloring of $G[u]$ and let S, N_1 and N_2 be the three matrices representing c in the nodes (v, w). Then:*

1. *$S(\emptyset, \emptyset) = 0$.*
2. *For each non-empty set $B_1 \subseteq [k] \times \langle l \rangle$, we have*
$$\sum_{\mathcal{B} \subseteq [k] \times \langle l \rangle} S(\mathcal{B}_1, \mathcal{B}) = N_1(\mathcal{B}_1).$$
3. *For each non-empty set $B_2 \subseteq [k] \times \langle l \rangle$, we have*
$$\sum_{\mathcal{B} \subseteq [k] \times \langle l \rangle} S(\mathcal{B}, \mathcal{B}_2) = N_2(\mathcal{B}_2).$$

Conversely, let N_1 be a matrix representing a coloring of $G[v]$ in the node v, N_2 be a matrix representing a coloring of $G[w]$ in the node w and S be a matrix of positive integers satisfying the above conditions. There is a coloring c of $G[u]$ such that c is represented in the nodes (v, w) by S, N_1 and N_2.

Proof. Let us show that if c is a coloring of $G[u]$, then S, N_1 and N_2 satisfy the three conditions. Let e (resp. f) be the restriction of c to $G[v]$ (resp. $G[u]$). Clearly, $S(\emptyset, \emptyset) = 0$ by definition of S. Let $\mathcal{B}_1$ be a non-empty subset of $[k] \times \langle l \rangle$. For each color $col \in \mathcal{C}_u^e(\mathcal{B}_1)$, let $\mathcal{B}_{col} = \{(i, d) \in [k] \times \langle l \rangle | col \in f(V_{i,d}[w])\}$. In particular, if $col \notin f(V[w])$, then $\mathcal{B}_{col} = \emptyset$. Then $S(\mathcal{B}_1, \mathcal{B}) = |\{col \mid \mathcal{B} = \mathcal{B}_{col}\}|$. In the sum $\sum_{\mathcal{B} \subseteq [k] \times \langle l \rangle} S(\mathcal{B}_1, \mathcal{B})$, each color $col \in \mathcal{C}_u^e(\mathcal{B}_1)$ is counted in exactly once, when $\mathcal{B} = \mathcal{B}_{col}$. Hence, this sum is equal to $N_1(\mathcal{B}_1)$. The third condition is similar to the second one.

Conversely, let S, N_1 and N_2 satisfying the three conditions above. Consider a coloring e of $G[v]$ (resp. a coloring f of $G[w]$) whose representation in the node v (resp. w) is N_1 (resp. N_2). We may suppose w.l.o.g. that e and f use different colors, i.e. $e(V[v]) \cap f(V[w]) = \emptyset$ (otherwise we can rename the colors of f such that they become distinct from those of e, and this operation does not change the matrix representing f in w). We will proceed by *merging* colors from e and f, i.e. we will consider some colors $col \in e(V[v])$ and $col' \in f(V[w])$ and we will rename the two colors by a new color $mcol$. The former color is called a *merged color*. More precisely, we consider each couple of non empty sets $\mathcal{B}_1, \mathcal{B}_2 \subset [k] \times \langle l \rangle$, we take $S(\mathcal{B}_1, \mathcal{B}_2)$ couples of non-merged colors from $\mathcal{C}_v^e(\mathcal{B}_1) \times \mathcal{C}_w^f(\mathcal{B}_2)$ and, for each of these couples (col, col'), we merge these two colors into a new one. Conditions 2 and 3 ensure us that we always have enough non-merged colors in $\mathcal{C}_v^e(\mathcal{B}_1)$ and in $\mathcal{C}_w^f(\mathcal{B}_2)$ for this operation. Moreover, it will remain in $\mathcal{C}_v^e(\mathcal{B}_1)$ (resp. $\mathcal{C}_w^f(\mathcal{B}_2)$) exactly $S(\mathcal{B}_1, \emptyset) \geq 0$ (resp. $S(\emptyset, \mathcal{B}_2) \geq 0$) non-merged colors. Then let c be the coloring of $G[u]$ whose restriction on $G[v]$ (resp. $G[w]$) is e (resp. f). By construction of c, it is represented in the nodes (v, w) by the matrices S, N_1 and N_2. □

We want to compute all the matrices of $F(u)$, representing in the node u the admissible colorings of $G[u]$. If c is an admissible coloring of $G[u]$, then its restrictions to $G[v]$ and $G[w]$ are also admissible. Hence, if S, N_1 and N_2 are the matrices representing c in the nodes (v, w), we have $N_1 \in F(v)$ and $N_2 \in F(w)$.

For computing $F(u)$ we proceed as follows. For each couple of matrices $N_1 \in F(v)$ and $N_2 \in F(w)$, we generate all the $2^{kl} \times 2^{kl}$ matrices S with elements in $\{0, \dots, n\}$. For each matrix S, we

1. Check in that S satisfies the conditions of Proposition 2.
2. Check if S, N_1 and N_2 represent an admissible coloring of $G[u]$, using Theorem 5 of Subsection 4.5. Indeed, for any coloring c of $G[u]$ represented in the nodes (v, w) by the three matrices, the admissibility of c only depends on the matrices.
3. Compute the matrix $N(S, N_1, N_2)$, representing in the node u the same coloring as S, N_1 and N_2 in the nodes (v, w), using Theorem 6 of Subsection 4.6. Indeed, for any coloring c of $G[u]$ represented in the nodes (v, w) by the three matrices, the matrix N_u^c is the same.

The second and the third point of the algorithm will be postponed to the next subsections.

4.4 Complexity of the algorithm

We deduce that the information $F(u)$ can be computed for each node u of T, and in particular we know how to obtain, in the root r of T, the matrix N_r^c representing a l-coloring c of G with minimum number of colors. By keeping the matrices S used for the merging operations, we can also compute an optimal l-coloring of G.

Theorem 4. *The l-coloring problem can be solved in $\mathcal{O}(k^3 l^3 2^{3kl} n^{2^{2kl}+2^{kl+1}+1})$ time on the class of graphs of clique-width at most k, if a CW_k-expression of the input graph is also given.*

Proof. It remains to discuss the complexity of the algorithm.

As mentioned before, T has at most $k^2 n$ nodes. For each node u of the expression-tree, $F(u)$ is a set of matrices with 2^{kl} elements in $\{0, \dots, |V[u]| + 1\}$, so $F(u)$ contains $\mathcal{O}(|V[u] + 1|^{2^{kl}})$ such matrices. The cost of the ρ, η and $\bullet$ operations are small compared with the cost of the $\oplus$ operations, which gives the complexity of the algorithm.

Let u be a $\oplus$ node and v and w be its descendants. Then $|V[v]| \leq n - 1$ and $|V[w]| \leq n - 1$, so $F(v)$ and $F(u)$ have at most $n^{2^{kl}}$ matrices. We consider each couple of matrices $(N_1, N_2) \in F(v) \times F(w)$, we generate all the $n^{2^{2kl}}$ matrices S of size $2^{kl} \times 2^{kl}$ with elements in $\{0, \dots, n-1\}$ and we use Proposition 2, Theorem 5 and Theorem 6 for computing (or rejecting) the matrix $N(S, N_1, N_2)$. Therefore, the cost of $F(u)$ is $\mathcal{O}(k^3 l^3 2^{3kl} n^{2^{2kl}+2^{kl+1}})$, and since there are at most n nodes of type $\oplus$ the theorem follows. □

4.5 Checking if S, N_1 and N_2 represent an admissible coloring of $G[u]$

The aim of this part is to prove the following theorem.

Theorem 5. *Given three matrices $S, N_1 \in F(v)$ and $N_2 \in F(w)$ representing a coloring c of $G[u]$ in the nodes (v, w), we can verify if c is admissible in $\mathcal{O}(k^2 l^2 2^{2kl})$ time.*

We denote by $\mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$ the length of the shortest path μ of G such that μ goes from a vertex labeled i in $G[v]$ to a vertex labeled j in $G[w]$ and μ does not intersect $V[v]$ or $V[w]$, except in its extremities. We may suppose that, for each $\oplus$ node u of the expression-tree T, the distances $\mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$ are calculated, for all couples $(i, j) \in [k]^2$, in a preprocessing phase. This calculation is made, for each $\oplus$ node, by an all-pairs shortest paths algorithm. There are at most n nodes of type $\oplus$, so this preprocessing step costs $\mathcal{O}(n^4)$ time.

Let us start on an easy remark on $\mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$.

Lemma 3. *Let $x \in V_i[v]$ and $y \in V_j[w]$. Then* $\mathrm{dist}_G(x, y) \leq \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$.

Proof. Let $\mu = [x_1, x_2, \ldots, x_p]$ be a shortest path from $V_i[v]$ to $V_j[w]$ in G, such that the interior of μ does not meet $V_i[v]$ or $V_j[w]$. Then $x_1 \in V_i[v]$ and $x_p \in V_j[w]$. The second vertex x_2 of μ is not in $V[v]$, and since it is adjacent to x_1 in G it also adjacent to all vertices having the same label as x_1 in $G[v]$, by definition of the class CW_k. In particular, x_2 is adjacent to x. For similar reasons, x_{p-1} is adjacent to y in G. So $\mu' = [x'_1 = x, x_2, \ldots, x_{p-1}, x'_p = y]$ is a path of G. Hence, $\mathrm{dist}_G(x, y) \leq |\mu'| = \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$. □

Lemma 4. *Let $x \in V[v]$ and $y \in V[w]$. Then $\mathrm{dist}_G(x, y) \leq l$ if and only if there are $i, j \in [k]$, $d, d' \in \langle l \rangle$ such that $x \in V_{i,d}[v]$, $y \in V_{j,d'}[w]$ and $d + d' + \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j) \leq l$.*

Proof. Suppose there are i, j, d and d' such that $x \in V_{i,d}[v]$, $y \in V_{j,d'}[w]$ and $\mathrm{dist}_G(v, i; w, j) + d + d' \leq l$, we show that $\mathrm{dist}_G(x, y) \leq l$. Since $x \in V_{i,d}[v]$, there is a vertex $x' \in V_i[v]$ such that $\mathrm{dist}_G(x, x') \leq d$. For similar reasons, there is a vertex $y' \in V_j[w]$ such that $\mathrm{dist}_G(y, y') \leq d'$. By Lemma 3, $\mathrm{dist}_G(x', y') \leq \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$. Hence, $\mathrm{dist}_G(x, y) \leq \mathrm{dist}_G(x, x') + \mathrm{dist}_G(x', y') + \mathrm{dist}_G(y', y) \leq d + d' + \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j) \leq l$.

Conversely, we show that if $\mathrm{dist}_G(x, y) \leq l$, then there are i, j, d and d' satisfying the conditions of the lemma. Consider a shortest path $\mu = [x = x_1, \ldots, x_p = y]$ from x to y in G. Let $s, 1 < s \leq p$ be smallest index such that $x_s \in V[w]$ and $q, 1 \leq q < s$ be the highest index such that $x_q \in V[v]$. Let i (resp. j) be the label of x_q (resp. x_r) in $G[v]$ (resp. $G[w]$). The sub-path $\mu' = [x_q, x_{q+1}, \ldots, x_s]$ of μ intersects $V[v]$ and $V[w]$ only in its ends. Thus, by definition of $\mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j)$, we have that $\mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j) \leq |\mu'|$. Let d (resp. d') be the length of the sub-path from x to x_q (resp. from x_q to y) of μ. Clearly, $x \in V_{i,d}[v]$, $y \in V_{j,d'}[w]$ and the length of μ is $d + d' + |\mu'|$. We conclude that $d + d' + \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j) \leq l$. □

Proposition 3. *Let S, $N_1 \in F(v)$ and $N_2 \in F(w)$ be three matrices representing a coloring c of $G[u]$. Then c is admissible w.r.t. G if and only if, for all $(i, d), (j, d') \in [k] \times \langle l \rangle$ such that $d + d' + \mathrm{dist}_G(v, i; w, j) \leq l$ and for all sets $\mathcal{B}_1, \mathcal{B}_2 \in [k] \times \langle l \rangle$ with $(i, d) \in \mathcal{B}_1$ and $(j, d') \in \mathcal{B}_2$ we have that $S(\mathcal{B}_1, \mathcal{B}_2) = 0$.*

Proof. Let e (resp. f) be the restrictions of c to $G[v]$ (resp. $G[w]$).

Suppose that the matrix S does not satisfy the condition of the proposition, so there are two non-empty sets $\mathcal{B}_1, \mathcal{B}_2 \subset [k] \times \langle l \rangle$ with $S(\mathcal{B}_1, \mathcal{B}_2) > 0$ and two couples $(i, d) \in \mathcal{B}_1$ and $(j, d') \in \mathcal{B}_2$ such that $d + d' + \mathrm{dist}_G(v, i; w, j) \leq l$. Since $S(\mathcal{B}_1, \mathcal{B}_2) > 0$, there is at least one color of c appearing in $\mathcal{C}_v^e(\mathcal{B}_1)$ and in $\mathcal{C}_w^f(\mathcal{B}_2)$. In particular, this colors occurs in $e(V_{i,d}[v])$ and in $f(V_{j,d'}[w])$. Hence, there are two vertices $x \in V_{i,d}[v]$ and $y \in V_{j,d'}[w]$ such that $e(x) = f(y)$, so $c(x) = c(y)$. By Lemma 4, the inequality $d + d' + \mathrm{dist}_G(v, i; w, j) \leq l$ implies $\mathrm{dist}_G(x, y) \leq l$, and therefore the coloring c is not an admissible w.r.t. G.

Conversely, suppose that c is not admissible w.r.t. G, we show that S does not satisfiy the condition of the proposition. The coloring e of $G[v]$ is represented in the node v by $N_1 \in F(v)$ so, by construction of $F(v)$, e is admissible w.r.t. G. For similar reasons, f is also admissible w.r.t. G. Since c is not admissible, the only possibility is the existence of two vertices $x \in V[v]$ and $y \in V[w]$ such that $\mathrm{dist}_G(x, y) \leq l$ and $c(x) = c(y)$ (or, equivalently, $e(x) = f(y)$). By Lemma 4, there are two couples $(i, d), (j, d') \in [k] \times \langle l \rangle$ such that $x \in V_{i,d}[v]$, $y \in V_{j,d'}[w]$ and $d + d' + \mathrm{dist}_G^{\mathrm{ext}}(v, i; w, j) \leq l$. Let $\mathcal{B}_1 = \{(i'', d'') \in [k] \times \langle l \rangle \mid e(x) \in e(V_{i'',d''}[v])\}$ and $\mathcal{B}_2 = \{(i'', d'') \in [k] \times \langle l \rangle \mid f(x) \in f(V_{i'',d''}[w])\}$. In particular, $(i, d) \in \mathcal{B}_1$ and $(j, d') \in \mathcal{B}_2$. By definition of $\mathcal{B}_1$ and $\mathcal{B}_2$, $e(x) = f(y)$ occurs in $\mathcal{C}_v^e(\mathcal{B}_1)$ and in $\mathcal{C}_w^f(\mathcal{B}_2)$. Hence, $S(\mathcal{B}_1, \mathcal{B}_2) > 0$, so S does not satisfy the conditions of the proposition. □

Given three matrices S, $N_1 \in F(v)$ and $N_2 \in F(w)$ representing a coloring c of $F(v)$, we can check if c is admissible w.r.t. G in $\mathcal{O}(k^2 l^2 2^{2kl})$ time, if we assume that the distances $\mathrm{dist}_G(v,i;w,j)$ were computed in a preprocessing step. Indeed, we take all the tuples $(i,j,d,d') \in [k] \times [k] \times \langle l \rangle \times \langle l \rangle$, and if $d + d' + \mathrm{dist}_G(v,i;w,j) \leq l$, we can check if S satisfies the conditions of Proposition 3. There are $k^2 l^2$ tuples (i,j,d,d'), and for each of tuple we need $\mathcal{O}(2^{2kl})$ time for verifying the matrix S. This achieves the proof of Theorem 5.

4.6 Computing the matrix N

We show in Lemma 5 that for any couple $(i,d) \in [k] \times \langle l \rangle$, the set $V_{i,d}[u]$ is the union of some sets $V_{i',d'}[v]$ and $V_{i'',d''}[w]$. We then use this observation in Theorem 6, for computing the matrix N from S, N_1 and N_2.

We denote by $\mathrm{dist}_G(u;i,j) = \mathrm{dist}_G(V_i[u], V_j[u])$. Again we suppose that, for each $\oplus$ node u of T, the distances $\mathrm{dist}_G(u;i,j)$ are computed in a preprocessing step, which takes $\mathcal{O}(n^4)$ time.

Lemma 5. *Let $x \in V[v]$, $i \in [k]$ and $d \in \langle l \rangle$. Then $x \in V_{i,d}[u]$ if and only if one of the following holds:*

1. *$x \in V_{i,d}[v]$;*
2. *there exist $j, j' \in [k]$ and $d' \in \langle l \rangle$ such that $x \in V_{j,d'}[v]$ and*

$$d' + \mathrm{dist}_G^{\mathrm{ext}}(v,j;w,j') + \mathrm{dist}_G(w;j',i) \leq d$$

Proof. We show that if one of the two conditions holds, then $x \in V_{i,d}[u]$. Clearly, if $x \in V_{i,d}[v]$ then $x \in V_{i,d}[u]$. Suppose now that x satisfies the second condition. Let $x' \in V_j[v]$ such that $\mathrm{dist}_G(x,x') \leq d'$. Consider $y' \in V_{j'}[w]$ and $y \in V_i[w]$ such that $\mathrm{dist}_G(y',y) = \mathrm{dist}_G(w;j',i)$ (notice that if $d' = 0$ then $x = x'$, and if $j' = i$ then $y = y'$). According to Lemma 3, $\mathrm{dist}_G(x',y') \leq \mathrm{dist}_G^{\mathrm{ext}}(v,j;w,j')$. We deduce that $\mathrm{dist}_G(x,y) \leq \mathrm{dist}_G(x,x') + \mathrm{dist}_G(x',y') + \mathrm{dist}_G(y',y) \leq d$. Since y has label i in $G[u]$, we conclude that $x \in V_{i,d}[u]$.

Conversely, suppose there is a vertex y labeled i in $G[u]$ such that $\mathrm{dist}_G(x,y) \leq d$. We show that one of the two conditions holds. If $y \in V[v]$ then the first condition is true by definition of $V_{i,d}[v]$. Otherwise, take $y \in V_i[w]$ such that $\mathrm{dist}_G(x,y)$ be minimum. Let $\mu = [x = x_1, \ldots, x_p = y]$ be a shortest path from x to y in G. Let $y' = x_s$ be the first vertex of $\mu \cap V[w]$. We denote by $x' = x_r$ the last vertex of $\mu \cap V[v]$, encountered before y (so $r < s$). Then x' and y' split μ into three paths: an x,x'-path μ_1, an x',y'-path μ_2 and an y',y-path μ_3. Let j be the label of x' in $G[v]$, j' be the label of y' in $G[w]$ and d' be the length of μ_1. Observe that μ_2 only intersects $V[u]$ in x' and y', so $\mathrm{dist}_G^{\mathrm{ext}}(v,j;w,j') \leq |\mu_2|$. Clearly, $\mathrm{dist}_G(w;j',i) \leq |\mu_3|$. Then $d' + \mathrm{dist}_G^{\mathrm{ext}}(v,j;w,j') + \mathrm{dist}_G(w;j',i) \leq |\mu_1| + |\mu_2| + |\mu_3| = |\mu| \leq d$. Therefore, d', j and j' satisfy the second condition of our theorem. □

Lemma 6. *Let H be a graph, c be a coloring of H and $\mathcal{X} = \{X_q \mid q \in K\}$ a family of subsets of $V(H)$, covering $V(H)$. Consider two distinct elements $a, b \in K$ and a new family $\mathcal{Y} = \{Y_q \mid q \in K\}$ defined by $Y_b = X_a \cup X_b$ and $Y_q = X_q$ for any $q \in K - \{b\}$.*

Then, for any $\mathcal{B} \subseteq K$,

1. *$N^c_{\mathcal{Y}}(\mathcal{B}) = N^c_{\mathcal{X}}(\mathcal{B})$ if a and b are not in $\mathcal{B}$.*
2. *$N^c_{\mathcal{Y}}(\mathcal{B}) = 0$ if $a \in \mathcal{B}$ and $b \notin B$.*
3. *$N^c_{\mathcal{Y}}(\mathcal{B}) = N^c_{\mathcal{X}}(\mathcal{B}) + N^c_{\mathcal{X}}(\mathcal{B} - \{b\})$ if $a, b \in \mathcal{B}$.*

Proof. The proof is very similar to the one of Lemma 2. If $\mathcal{B}$ does not contain a or b, nothing changed between the colors of $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$ and $\mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$.

If $\mathcal{B}$ contains a but not b, a color of c can not be in Y_a without being in Y_b, so $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$ is empty.

If $a, b \in \mathcal{B}$, any color *col* of $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$ is in $Y_a = X_a$. Observe that *col* is either in $c(X_a) \cap c(X_b)$ or in $c(X_b) - c(X_a)$. In the first case, $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$, in the second case $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} - \{b\})$. Conversely, any color of $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$ or $col \in \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} - \{b\})$ is in $c(Y_a)$ and in $c(Y_b)$, so in $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B})$. Hence $\mathcal{C}^c_{\mathcal{Y}}(\mathcal{B}) = \mathcal{C}^c_{\mathcal{X}}(\mathcal{B}) \cup \mathcal{C}^c_{\mathcal{X}}(\mathcal{B} - \{b\})$ and the third condition follows directly. □

Theorem 6. *Given three matrices S, N_1, N_2 representing a coloring c in the nodes (v, w), the matrix N representing c in the node u is computable in $\mathcal{O}(k^3 l^3 2^{3kl})$ time.*

Proof. Let $K = [k] \times \langle l \rangle \times \{v, w, u\}$. We construct a family $\mathcal{X} = \{X_a \mid a \in K\}$ of subsets of $V[u]$, covering $V[u]$, as follows: for any element $a = (i, d, t)$ of K, we put $X_a = V_{i,d}[t]$ if $t \in \{v, w\}$, and $X_a = \emptyset$ if $t = u$. We compute first the matrix $N^c_{\mathcal{X}}$. Let $\mathcal{B}$ be a subset of K and, for each $t \in \{v, u, w\}$, let $\mathcal{B}_t = \{(i, d) \mid (i, d, t) \in K\}$. We show that $N^c_{\mathcal{X}}(\mathcal{B}) = \begin{cases} 0 & \text{if } \mathcal{B}_u \neq \emptyset \\ S(\mathcal{B}_v, \mathcal{B}_w) & \text{otherwise} \end{cases}$

If $\mathcal{B}_v$ contains a couple (i, d), since $X_{(i,d,u)}$ is empty and $(i, d, u) \in \mathcal{B}$, we have $N^c_{\mathcal{X}}(\mathcal{B}) = 0$. If $\mathcal{B}_u = \emptyset$, then by definition of $\mathcal{X}$ the set $\mathcal{C}^c_{\mathcal{X}}(\mathcal{B})$ is formed by the colors of c

- appearing in $c(V_{i,d}[v])$ for each $(i, d) \in \mathcal{B}_v$ and not appearing in $c(V_{i',d'}[v])$ for each $(i', d') \notin \mathcal{B}_v$;
- appearing in $c(V_{i,d}[w])$ for each $(i, d) \in \mathcal{B}_w$ and not appearing in $c(V_{i',d'}[v])$ for each $(i', d') \notin \mathcal{B}_w$;

Hence $N^c_{\mathcal{X}}(\mathcal{B}) = S(\mathcal{B}_v, \mathcal{B}_w)$. Thus, computing the matrix $N^c_{\mathcal{X}}$ takes $\mathcal{O}(kl 2^{3kl})$ time (each of the 2^{3kl} is identified by the sets $\mathcal{B}_u, \mathcal{B}_v$ and $\mathcal{B}_w$ of size kl, and for a given couple $\mathcal{B}_u, \mathcal{B}_v$ we access $S(\mathcal{B}_u, B_v)$ in $\mathcal{O}(kl)$ time).

Let now $\mathcal{Y} = \{Y_a \mid a \in K\}$ be defined by $Y_{(i,d,t)} = V_{i,d}[t]$ for each $(i, d, t) \in [k] \times \langle l \rangle \times \{v, w, u\}$. We compute the matrix $N^c_{\mathcal{Y}}$ as follows. For each $(i, d) \in [k] \times \langle l \rangle$, replace in $\mathcal{X}$ the empty set $X_{(i,d,u)}$ with the set $Y_{(i,d,u)} = V_{i,d}[u]$. According to Lemma 5, $Y_{(i,d,u)}$ is the union of certain elements of $\mathcal{X}$. Indeed, $Y_{(i,d,u)} = V_{i,d}[u]$ is the union of some sets $V_{i',d'}[v] = X_{(i',d',v)}$ and $V_{i'',d'',w} = X_{(i'',d'',w)}$. Therefore, after each replacement we can update the matrix $N^c_{\mathcal{X}}$ by applying Lemma 6 at most $2kl$ times. This costs $\mathcal{O}(k^2 l^2 2^{3kl})$ time by replacement. At the end of this loop, $\mathcal{X} = \mathcal{Y}$ and we have computed $N^c_{\mathcal{Y}}$. This computation takes $\mathcal{O}(k^3 l^3 2^{3kl})$ time.

It remains to calculate N^c_u. Consider three subsets $\mathcal{B}_v, \mathcal{B}_w$ and B_u of $[k] \times \langle l \rangle$, and let $\mathcal{B}_v \oplus \mathcal{B}_w \oplus \mathcal{B}_u = \{(i, d, t) \in K \mid (i, d) \in B_t\}$. For any $\mathcal{B}_u \subseteq [k] \times \langle l \rangle$, $N^c_u(\mathcal{B}_u) = \sum_{\mathcal{B}_v, \mathcal{B}_w \subseteq [k] \times \langle l \rangle} N^c_{\mathcal{Y}}(\mathcal{B}_v \oplus \mathcal{B}_w \oplus \mathcal{B}_u)$.

The last step costs $\mathcal{O}(kl 2^{3kl})$ time. So the matrix N^c_u can be computed from $N^c_{\mathcal{Y}}$ in $\mathcal{O}(k^3 l^3 2^{3kl})$ time, which concludes the proof of Theorem 6. □

References

1. H. Bodlaender, T. Kloks, R. Tan, and J. van Leeuwen. λ-colorings of graphs. In *Proceedings 17th Annual Symposium on Theoretical Aspects of Computer Science (STACS 2000)*, volume 1770 of *Lecture Notes in Computer Science*, pages 395–406. Springer-Verlag, 2000.
2. H. Bodlaender and D. Kratsch, 2002. Private communication.
3. G. Chang and D. Kuo. The $L(2,1)$-labeling problem on graphs. *SIAM J. on Discrete Mathematics*, 9:309–316, 1996.
4. B. Courcelle and S. Olariu. Upper bounds to the clique-width of graphs. *Discrete Applied Mathematics*, 101:77–114, 2000.

5. W. Espelage, F. Gurski, and E. Wanke. How to solve NP-hard graph problems on clique-width bounded graphs in polynomial time. In *Workshop on Graph-theoretic Concepts in Computer Science (WG 2001)*, volume 2204 of *Lecture Notes in Computer Science*, pages 117–128. Springer, 2001.
6. M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completness.* Freeman, 1979.
7. D. Kobler and U. Rotics. Polynomial algorithms for partitioning problems on graphs with fixed clique-width (extended abstract). In *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms (SODA 2001)*, pages 468–476, 2001.
8. X. Zhou, Y. Kanari, and T. Nishizeki. Generalized vertex-colorings for partial k-trees. *IEICE Trans. Fundamentals*, E83-A(4):671 – 677, april 2000.

A Proof of Proposition 1

Let $G = (V, E)$ be a unlabeled graph in CW'_k and T be a CW'_k-expression tree corresponding to G. We construct a CW_{2k}-expression tree T' of G.

Let u be a node of T and u' be the lowest $\oplus$ ancestor of u. If u is in the left subtree of u' or u' does not exist, we say that u is a left-hand side node. Otherwise u is a right-hand side node of T. The labels of the CW_{2k} expression of G will be taken in the set $\{l, r\} \times [k]$. For each left-hand (resp. right-hand) side node u of T and for each $i \in [k]$, the label i will be transformed in (l, i) (resp (r, i)). Then each $\eta_{i,i}$ node of T will be replaced by $\eta_{i,j}$ nodes with $i \neq j$.

1. Each left-hand side node $\bullet_i$ (resp. $\rho_{i\to j}$, $\eta_{i,j}$ with $i \neq j$) is replaced in T' by $\bullet_{(l,i)}$ (resp. $\rho_{(l,i)\to(l,j)}$, $\eta_{(l,i),(l,j)}$).
2. Each right-hand side node $\bullet_i$ (resp. $\rho_{i\to j}$, $\eta_{i,j}$ with $i \neq j$) is replaced by $\bullet_{(r,i)}$ (resp. $\rho_{(r,i)\to(r,j)}$, $\eta_{(r,i),(r,j)}$).
3. Each $\oplus$ node u of T will be replaced by a path μ_u constructed as follows.
   (a) If u is a left-hand side node, then the path starts with $\rho_{(r,1)\to(l,1)}, \rho_{(r,2)\to(l,2)}, \dots, \rho_{(r,k)\to(l,k)}$. Otherwise, u is a right-hand side node and the path starts with $\rho_{(l,1)\to(r,1)}, \rho_{(l,2)\to(r,2)}, \dots, \rho_{(l,k)\to(r,k)}$.
   (b) For each ancestor u' of u of type $\eta_{i,i}$, let $Comp(u, u')$ be the set of labels $j \in [k]$ such that each vertex labeled j in $\mathrm{val}(T[u])$ will be labeled i in $\mathrm{val}(T[u'])$. Clearly, for any couple of vertices x, y of $\mathrm{val}(T[u])$ such that $\mathrm{lab}_{\mathrm{val}(T[u])}(x), \mathrm{lab}_{\mathrm{val}(T[u])}(y) \in Comp(u, u')$, x and y are adjacent in G because the edge xy is added by the node u'. We want to take into account these adjacences. For each couple $(j, j') \in Comp(u, u') \times Comp(u, u')$, add to the path μ_u a node labeled $\eta_{(l,j),(r,j')}$.
   (c) The path μ_u ends with a node labeled $\oplus$.
4. All the $\eta_{i,i}$ nodes are omitted.

Let $G' = (V, E')$ be the graph corresponding to the CW_{2k}-tree T' constructed above, it remains to check that $G = G'$. Clearly $E' \subseteq E$. Conversely, we show that $E \subseteq E'$. Suppose there is an edge $xy \in E - E'$. This edge was added to G during a $\eta_{i,i}$ operation in some node u' of T. Let u be the lowest common ancestor of the leaves of T corresponding to the vertices x and y. Clearly u is a $\oplus$ node. Since the label of x and y in $\mathrm{val}(T[u'])$ is i, it follows that the labels of x and y in $\mathrm{val}(T[u])$ are contained in $Comp(u, u')$, by definition of this set. Hence the edge xy is in G' (added during step 3b), so $G = G'$ which achieves our proof.