Laboratoire de l'Informatique du Parallélisme

École Normale Supérieure de Lyon
Unité de recherche associée au CNRS n° 1398

Minimal Triangulations for Graphs with "Few" Minimal Separators

Vincent Bouchitté
Ioan Todinca

May 1998

Research Report N° 98-24

École Normale Supérieure de Lyon
46 Allée d'Italie, 69364 Lyon Cedex 07, France
Téléphone : +33(0)4.72.72.80.00
Télécopieur : +33(0)4.72.72.80.80
Adresse électronique : `lip@ens-lyon.fr`

Minimal Triangulations for Graphs with “Few” Minimal Separators

Vincent Bouchitté
Ioan Todinca

May 1998

Abstract

We give a characterization of minimal triangulation of graphs using the notion of “maximal set of neighbor separators”. We prove that if all the maximal sets of neighbor separators of some graphs can be computed in polynomial time, the treewidth of those graphs can be computed in polynomial time. This notion also unifies the already known algorithms computing the treewidth of several classes of graphs.

Keywords: treewidth, chordal graphs, minimal separators, minimal triangulations

Résumé

Nous donnons une caractérisation des triangulations minimales des graphes en utilisant la notion de “ensemble maximal de séparateurs voisins”. Nous prouvons que si tous les ensembles maximaux de séparateurs voisins d'une classe de graphes sont calculables en temps polynomial, le treewidth de ces graphes se calcule en temps polynomial. Cette notion permet d'unifier les algorithmes calculant le treewidth pour plusieures classes de graphes.

Mots-clés: treewidth, graphes triangulés, séparateurs minimaux, triangulations minimales

1 Introduction

The *treewidth* of graphs, introduced by Robertson and Seymour [12], has been intensively studied in the last years, mainly because many NP-hard problems become solvable in polynomial and even linear time when restricted to graphs with small treewidth. These algorithms use a tree-decomposition of small width of the graph. A tree-decomposition or a *triangulation* of a graph is a chordal supergraph (i.e. all the cycles of the supergraph of length more than three have a chord). Computing the treewidth of a graph corresponds to finding a triangulation with the smallest cliquesize. In particular, we can restrict ourselves to triangulations minimal by inclusion, that we call *minimal triangulations*.

Computing the treewidth of arbitrary graphs is NP-hard. Nevertheless, the treewidth can be computed in polynomial time for several well-known classes of graphs, for example the chordal bipartite graphs [9], the circle and circular-arc graphs [5] [14] and permutation graphs [2]. All these algorithms use the *minimal separators* of the graph and the fact that these classes of graphs have "few" minimal separators, in the sense that the number of the separators is polynomially bounded in the size of the graph. All of them compute minimal triangulations. Different algorithms and different proofs are given for each of these classes of graphs. This paper gives a unified version of the cited algorithms.

Characterizations of the minimal triangulations of a graph by the minimal separators have already been given in [2], [11]. Their approach is a global vision of all the minimal separators of a graph and therefore they do not yield an algorithmic construction of minimal triangulations with small cliquesize. A more local view of the minimal triangulations has been tried in [6], but it was unfortunately not correct [7]. We will give a new local characterization of minimal triangulations that we will use to compute the treewidth for particular classes of graphs, including those previously mentioned.

In section 2 we present the relation between the minimal triangulations of a graph and its separators. This leads to a *global* characterization of the minimal triangulations by maximal sets of pairwise parallel separators [11].

In section 3 we give a *local* characterization of minimal triangulations by means of "maximal sets of neighbor separators". We show the relation between the maximal sets of neighbor separators and the maximal cliques of any minimal triangulation of a graph.

In section 4 we prove that if all the maximal sets of neighbor separators of some classes of graphs can be computed in polynomial time, then the treewidth of those graphs can be computed in polynomial time. We give an algorithm performing this computation.

In section 5 we show that for all classes of graphs mentioned above, for which it is known how to compute the treewidth, it is in fact easy to compute the maximal sets of neighbor separators (and this is precisely what the already known algorithms do). This allows us to prove in a simple manner all these algorithms.

2 Chordal graphs and minimal separators

Throughout this paper we consider connected, simple, finite, undirected graphs.

A graph H is *chordal* (or *triangulated*) if every cycle of length at least four has a chord. A *triangulation* of a graph $G = (V, E)$ is a chordal graph $H = (V, E')$ such that $E \subseteq E'$. H is a *minimal triangulation* if for any intermediate set E'' with $E \subseteq E'' \subset E'$, the graph (V, E'') is not triangulated.

We will use the representation of the chordal graphs provided by *clique trees*. For an extensive survey of these notions, see [1], [3]. A *clique* is a complete subgraph of G. Consider now the set $\mathcal{K}_G = \{\Omega_1, \ldots, \Omega_p\}$ of maximal cliques of G. Let $\mathcal{T}$ be a tree on $\mathcal{K}_G$, i.e. every maximal clique $\Omega \in \mathcal{K}_G$ corresponds to exactly one node of $\mathcal{T}$ (we also say that the nodes of $\mathcal{T}$ are *labeled* by the cliques of $\mathcal{K}_G$). We say that $\mathcal{T}$ is a *clique tree* of G if it satisfies the *clique-intersection property*: for every pair of distinct cliques $\Omega, \Omega' \in \mathcal{K}$, the set $\Omega \cap \Omega'$ is contained in every clique on the path connecting Ω and Ω' in the tree $\mathcal{T}$. It is well known (see [1] for a proof) that a graph G is chordal if and only if it has a clique tree.

A subset $S \subseteq V$ is an *a,b-separator* for two nonadjacent vertices $a, b \in V$ if the removal of S from the graph separates a and b in different connected components. S is a *minimal a,b-separator* if no proper subset of S separates a and b. We say that S is a *minimal separator* of G if there are two vertices a and b such that S is a minimal a,b separator (notice that a minimal separator can be strictly included in another). We denote by Δ_G is the set of all minimal separators of G.

The following crucial property is proved in [1] :

Proposition 1 *Let H be a chordal graph and $\mathcal{T}$ be any clique tree of H. A set S is a minimal separator if and only if $S = \Omega \cap \Omega'$ for some maximal cliques Ω and Ω' of H adjacent in the clique tree $\mathcal{T}$.*

Let G be a graph and S a minimal separator of G. We note $\mathcal{C}_G(S)$ the set of connected components of $G - S$. A component $C \in \mathcal{C}_G(S)$ is *full* if every vertex of S is adjacent to some vertex of C. We denote by $\mathcal{C}^*_G(S)$ the set of all full components of $G - S$. We remark that every minimal separator has at least two full components and it is a minimal a,b-separator for every couple of vertices a and b which are in different full components of S. If $C \in \mathcal{C}(S)$, we say that $(S, C) = S \cup C$ is a *block* of S. A block (S, C) is called *full* if C is a full component of S.

Definition 1 *Two separators S and T* cross, *denoted by $S \sharp T$, if there are some distinct components C and D of $G - T$ such that S intersects both of them. If S and T do not cross, they are called* parallel, *denoted by $S \| T$.*

It is easy prove that these relations are symmetric (see [10]). Remark that for any couple of parallel separators S and T, T is contained in some block (S, C) of S. The parallelism of the separators has a certain "pseudo-transitivity". Indeed, if $T \| S$, $S \| U$ and T and U are contained in different blocks of S, then then $T \| U$. In particular, S separates any vertex of $T - S$ from any vertex of $U - S$. In this case we say that S *separates* T and U.

Using the fact that a separator cannot separate two adjacent vertices and the proposition 1, we deduce the following lemma.

Lemma 1 *Let G be a graph, S a minimal separator and Ω a clique of G. Then Ω is included in some block of S. In particular, the minimal separators of a chordal graph are pairwise parallel.*

Definition 2 *Let G be a graph. The* treewidth *of G is the minimum, over all triangulations H of G, of $\omega(H)-1$, where $\omega(H)$ is the the maximum cliquesize of H.*

In other words, computing the treewidth of G means finding a triangulation with smallest cliquesize, so we can restrict our work to minimal triangulations.

Let $S \in \Delta_G$ be a minimal separator. We denote by G_S the graph obtained from G by completing S, i.e. by adding an edge between every pair of vertices $x, y \in S$. If $\Gamma \subseteq \Delta_G$ is a set of separators of G, G_Γ is the graph obtained by completing all the separators of Γ. The results of [2], concluded in [11], establish a strong relation between the minimal triangulations of a graph and its separator graph.

Theorem 1

- *Let $\Gamma \in \Delta_G$ be a maximal set of pairwise parallel separators of G. Then $H = G_\Gamma$ is a minimal triangulation of G and $\Delta_H = \Gamma$.*
- *Let H be a minimal triangulation of a graph G. Then Δ_H is a maximal set of pairwise parallel separators of G and $H = G_{\Delta_H}$.*

In other terms, every minimal triangulation of a graph G is obtained by considering a maximal set Γ of pairwise parallel separators of G and completing the separators of Γ. The minimal separators of the triangulation are exactly the elements of Γ.

It is important to know that the elements of Γ, who become the separators of H, have strictly the same behavior in H as in G. Indeed, the connected components of $H-S$ are exactly the same in $G-S$, for every $S \in \Gamma$. Moreover, the full components are the same in the two graph: $\mathcal{C}^*_H(S) = \mathcal{C}^*_G(S)$.

3 Maximal sets of neighbor separators

The previous theorem gives a characterization of the minimal triangulations of a graph by means of minimal separators, but it needs a global look over the set of minimal separators. Therefore, it gives no algorithmic information about how we should construct a minimal triangulation in order to optimize its cliquesize. In this section we give a local characterization of minimal triangulations. We introduce the notion of "maximal set of neighbor separators" and we show how these sets are related to the maximal cliques of any triangulation of a graph. We also give a tool for recognizing the maximal sets of neighbor separators.

Definition 3 *Let H be a triangulated graph. We say that $\mathcal{S} \subseteq \Delta_H$ is a* maximal set of neighbor separators *if there is a maximal clique Ω of H such that $\mathcal{S} = \{S \in \Delta_H | S \subset \Omega\}$. We also say that $\mathcal{S}$* borders *Ω in H.*

Definition 4 *Let G be a graph and $\mathcal{S} \subseteq \Delta_G$ a set of pairwise parallel separators such that for any $S \in \mathcal{S}$, there is a block $(S, C(S))$ containing all the separators of $\mathcal{S}$.*

We define the piece between *the elements of $\mathcal{S}$, denoted by $P(\mathcal{S})$, as*

- *U, if there is some $U \in \mathcal{S}$ that contains every $S \in \mathcal{S}$.*

- $\bigcap_{S \in \mathcal{S}} (S, C(S))$ *otherwise.*

Notice that if we are in the second case, for any $S \in \mathcal{S}$ the block of S containing all the separators of $\mathcal{S}$ is unique : if $T \in \mathcal{S}$ is not included in S, there is a unique connected component of S containing $T - S$.

Lemma 2 *Let $\mathcal{S} \subseteq \Delta_H$ be a maximal set of neighbor separators. Then $P(\mathcal{S})$ exists.*

Proof. Let Ω be a clique bordered by $\mathcal{S}$. For any $S \in \mathcal{S}$, the clique Ω is contained in some block (S, C) (lemma 1). So all the elements of $\mathcal{S}$ are contained in a same block of S. $\diamond$

We are now able to establish the relation between a maximal set $\mathcal{S}$ of neighbor separators of a chordal graph H and a maximal clique of H bordered by $\mathcal{S}$.

Theorem 2 *Let $\mathcal{S}$ be the maximal set of neighbor separators bordering a maximal clique Ω of a chordal graph H. If all the separators of $\mathcal{S}$ are included in some $S \in \mathcal{S}$, then Ω is a block of S. Otherwise, $\Omega = P(\mathcal{S})$.*

Proof. Suppose that $\bigcup_{T \in \mathcal{S}} T = S$ with $S \in \mathcal{S}$. Ω is included in some block (S, C) of S. If $x \in C$ is not in Ω, we take in a clique tree of H the shortest path $\Omega, \Omega', \dots, \Omega^{(i)}$ of adjacent cliques such that $x \in \Omega^{(i)}$. Then $T = \Omega \cap \Omega'$ is a minimal separator of H included in Ω, so it must be an element of $\mathcal{S}$. In particular, T is included in S. T separates x and any vertex of $\Omega - T$. Since $T \subseteq S$, clearly S separates x and any vertex of $\Omega - S$, contradicting the fact that x and Ω are in the same block of S. It remains that $(S, C) = \Omega$.

Suppose now that no separator $S \in \mathcal{S}$ contains all the others. According to the lemma 2, $P(\mathcal{S})$ exists. On one hand $\Omega \subseteq P(\mathcal{S})$. Indeed, for any $S \in \mathcal{S}$, Ω is in the block $(S, C(S))$ containing all the separators of $\mathcal{S}$. On the other hand, consider $y \in P(\mathcal{S}) - \Omega$. In a clique tree of H, we consider the shortest path $\Omega, \Omega', \dots, \Omega^{(i)}$ of adjacent cliques such that $y \in \Omega^{(i)}$. Then $S = \Omega \cap \Omega'$ is a minimal separator that belongs to $\mathcal{S}$ and S separates y from every vertex of $\Omega - S$. Let T be a separator of $\mathcal{S}$, not included in S. Then S separates y and a vertex of $T - S$, so $y \notin (S, C_{\mathcal{S}}(S))$, contradicting our choice. So we have $\Omega = P(\mathcal{S})$. $\diamond$

Definition 5 *For an arbitrary graph G, $\mathcal{S} \subseteq \Delta_G$ is a* maximal set of neighbor separators *if there is a minimal triangulation H of G such that $\mathcal{S}$ is a maximal set of neighbor separators in H.*

We now establish the "reverse" of the theorem 2, which will allow us to "recognize" a set of neighbor separators. For this we need some easy lemmas.

Lemma 3 *Let H be a chordal graph, Ω a maximal clique and S a minimal separator of H intersecting Ω. Then $S \cap \Omega$ is included in some minimal separator T with $T \subset \Omega$.*

Proof. Consider a clique tree $\mathcal{T}$ of H. We can write $S = \Omega_1 \cap \Omega_2$ for some maximal cliques Ω_1, Ω_2 of H, adjacent in $\mathcal{T}$ (proposition 1). Let us say that Ω_1

is closer to Ω in the clique tree than Ω_2. Let Ω' be the clique next to Ω on the path from Ω to Ω_2 in $\mathcal{T}$, i.e. the path is $\Omega, \Omega', \ldots, \Omega_1, \Omega_2$.

Then $\Omega_1 \cap \Omega \subset \Omega'$ and $\Omega_2 \cap \Omega \subset \Omega'$ by definition of a clique tree. Since $S = \Omega_1 \cap \Omega_2$, we deduce that $S \cap \Omega \subset \Omega'$ and in particular $S \cap \Omega$ is contained in $\Omega \cap \Omega'$, which is a minimal separator by property 1. $\diamond$

Lemma 4 *Let H be a minimal triangulation of a graph G and T be a minimal separator of G such that $H[T]$ is a clique. Then T is also a minimal separator of H.*

Proof. We know that $H = G_\Gamma$ for a maximal set of pairwise parallel separators $\Gamma \subseteq \Delta_G$ and $\Delta_H = \Gamma$ (theorem 1). Let $S \in \Gamma$ be any minimal separator of H and G. By lemma 1, the clique T will be in some block (S, C) of H. Since the blocks of S in G are the same as in H (theorem 1 and the related remarks) we deduce that S and T are parallel in G. Since Γ is a *maximal* set of pairwise parallel separators, T must be in $\Gamma = \Delta_H$. $\diamond$

Corollary 1 *In particular, lemma 4 asserts that if T is a minimal separator of G contained in S with $S \in \Delta_H$, then $T \in \Delta_H$.*

Lemma 5 *Let (S, C) be a block in G such that $G_S[S \cup C]$ is a clique. Then every minimal separator $T \subseteq (S, C)$ is included in S.*

Proof. Clearly $T \| S$. Consider a minimal triangulation H of G such that $S, T \in \Delta_H$. Then $S \cup C$ is a clique of H. Since C is a connected component of $H - S$, for any vertex $x \in C$, all the vertices adjacent to x in H are contained in $S \cup C$. Now if $x \in T \cap C$, consider two full components of $H - T$. Let a, respectively b be adjacent to x in each one of these components. Then T separates a and b in H, contradicting the fact that $a, b \in S \cup C$ are adjacent in H. It remains that $T \subseteq S$. $\diamond$

Theorem 3 *Let G be a graph and $\mathcal{S} \subseteq \Delta_G$ a set of separators having a maximum element S (S contains every $T \in \mathcal{S}$). Then $\mathcal{S}$ is a maximal set of neighbor separators of G if and only if:*

1. *there is a block (S, C) such that $G_S[S \cup C]$ is a clique.*
2. *$\forall T \in \Delta_G$, if $T \subseteq S$, then $T \in \mathcal{S}$.*

Proof.

"$\Rightarrow$" Let $H = G_\Gamma$ be a minimal triangulation such that $\mathcal{S}$ borders a clique Ω. According to theorem 2, there is a block (S, C) such that $H[S \cup C] = \Omega$. We prove that $S \cup C$ is a clique in G_S. Indeed, every edge of $H[S \cup C]$ is either an edge of G, or its endpoints x and y are on a same separator $T \in \Gamma$. In the last case, by lemma 3 we have a separator T' of H such that $T' \subseteq \Omega$ and T' contains x and y. Necessarily $T' \in \mathcal{S}$ (cf. the definition of maximal set of neighbor separators), so $T' \subseteq S$. It remains that x and y are adjacent in $G_S[S \cup C]$. We conclude that $G_S[S \cup C]$ is a clique.

The second statement is a direct consequence of corollary 1.

"$\Leftarrow$" Notice that the separators of $\mathcal{S}$ are pairwise parallel. Let $H = G_\Gamma$ be a minimal triangulation extending $\mathcal{S}$, i.e. with $\mathcal{S} \subseteq \Gamma$. We prove that $S \cup C$ is a maximal clique of H, bordered by $\mathcal{S}$. The clique $S \cup C$ of H is maximal.

Indeed, if Ω is a clique of H including $S \cup C$, Ω must be in some block of S (lemma 1) and the only choice is $\Omega \subseteq (S, C)$. We denote by $\mathcal{S}'$ the maximal set of neighbor separators bordering Ω. We have $\mathcal{S} \subseteq \mathcal{S}'$. Moreover, every element of $\mathcal{S}'$ is a separator of H, and also a separator of G. By the lemma 5 such a separator is included in S, and by the second statement it is an element of $\mathcal{S}$. It remains that $\mathcal{S} = \mathcal{S}'$. $\diamond$

Theorem 4 *Let G be a graph and $\mathcal{S} \subset \Delta_G$ a set of separators not having a greatest element. Then $\mathcal{S}$ is a maximal set of neighbor separators of G if and only if:*

1. *$P(\mathcal{S})$ is non empty and $G_{\mathcal{S}}[P(\mathcal{S})]$ is a clique.*
2. *$\forall S \in \Delta_G$, if $S \subset P(\mathcal{S})$ then $S \in \mathcal{S}$.*

Proof. The proof is almost the same as the one of the previous theorem.

"$\Rightarrow$" We consider a minimal triangulation H of G such that $\mathcal{S}$ is a maximal set of neighbor separators in H, bordering a clique Ω. According to theorem 2, $\Omega = P(\mathcal{S})$. We show now that $H[P(\mathcal{S})]$ is exactly $G_{\mathcal{S}}[P(\mathcal{S})]$. For every x and $y \in \Omega$, either x and y are adjacent in G or they belong to a same separator of H. The later implies that x and y belong to a same separator T included in Ω by lemma 3, and by definition of a maximal set of neighbor separators we get $T \in \mathcal{S}$. In both cases x and y are adjacent in $G_{\mathcal{S}}[P(\mathcal{S})]$. This proves that $G_{\mathcal{S}}[P(\mathcal{S})]$ is a clique.

Let T be a separator of G included in $P(\mathcal{S})$. $H[T]$ is a clique, so $T \in \Gamma$ by lemma 4 and necessarily $T \in \mathcal{S}$, which proves the second statement.

"$\Leftarrow$" As in the previous theorem, we prove that $\mathcal{S}$ borders the clique $G_{\mathcal{S}}[P(\mathcal{S})]$ in any triangulation extending $\mathcal{S}$.

The first statement implies that the separators of $\mathcal{S}$ are pairwise parallel. We consider a maximal set of pairwise parallel separators $\Gamma \supseteq \mathcal{S}$ and we take $H = G_\Gamma$. We consider a maximal clique Ω of H containing $P(\mathcal{S})$. We want to prove that $\Omega = P(\mathcal{S})$. Let x be a vertex of Ω. For any $S \in \mathcal{S}$, x must be in the block $(S, C_\Omega(S))$ containing Ω. So $x \in P(\mathcal{S})$, which proves that $\Omega = P(\mathcal{S})$. Let now $\mathcal{S}'$ be the maximal set of neighbor separators bordering Ω. Clearly $\mathcal{S} \subseteq \mathcal{S}'$ and by the second statement $\mathcal{S}'$ is a subset of $\mathcal{S}$. It remains that $\mathcal{S} = \mathcal{S}'$ is a maximal set of neighbor separators of G. $\diamond$

So, given a set of separators $\mathcal{S}$, one can easily verify if it is a maximal set of neighbor separators.

For a maximal set of neighbor separators $\mathcal{S}$, we can define a clique $\Omega_{\mathcal{S}}$ bordered by $\mathcal{S}$ in $G_{\mathcal{S}}$ as a clique (S, C) of $G_{\mathcal{S}}$ if S is maximum in $\mathcal{S}$, $P(\mathcal{S})$ otherwise (respectively the cliques that appear in the two previous theorems). Remark that if we are under the condition of the theorem 4 the clique $\Omega_{\mathcal{S}}$ is unique, but that might not be the case if we are under the condition of theorem 3. By the proofs of the two previous theorems we deduce that:

Corollary 2 *Let $\mathcal{S}$ be a maximal set of neighbor separators of G and $\Gamma \supseteq \mathcal{S}$ a set of pairwise parallel separators. Then a clique bordered by $\mathcal{S}$ in $G_{\mathcal{S}}$ is a maximal clique in G_Γ.*

In the future, when speaking about a clique bordered by a maximal set of neighbor separators $\mathcal{S}$ we will mean a clique of $G_{\mathcal{S}}$.

Let S be a minimal separator and let (S, C) be a full block of S. A maximal set of neighbor separators $\mathcal{S}$ is said to be *associated* to the block (S, C) if $S \in \mathcal{S}$ and some clique $\Omega_{\mathcal{S}}$ bordered by $\mathcal{S}$ is contained in (S, C).

4 Computing the treewidth

In this section we show how to compute in polynomial time the treewidth of a graph if we have all its maximal sets of neighbor separators.

Let $\mathcal{F} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_p\}$ be a family of maximal sets of neighbor separators and $\Gamma_{\mathcal{F}} = \{S | \exists \mathcal{S} \in \mathcal{F} \text{ s.t. } S \in \mathcal{S}\}$. We say that the family $\mathcal{F}$ is a *complete family of separators* if for any separator $S \in \Gamma_{\mathcal{F}}$ and for any full block (S, C) of S, there is a maximal set of neighbor separators $\mathcal{S} \in \mathcal{F}$ associated to the block (S, C).

For example, if H is a triangulation of a graph G, the family $\mathcal{F} = \{\mathcal{S} | \mathcal{S} \subseteq \Delta_H$ is a maximal set of neighbor separators of $H\}$ is a complete family of separators of G. Remark that if H has treewidth at most k, for any maximal set of neighbor separators $\mathcal{S}$ of H, all the cliques bordered by $\mathcal{S}$ have at most $k+1$ vertices.

We will first compute the biggest complete family of separators of G, denoted by $\mathcal{F} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_p\}$ such that all the maximal cliques bordered by any $\mathcal{S} \in \mathcal{F}$ have at most $k+1$ elements. This family is empty if and only if G has treewidth greater than k or G is a clique.

Elimination algorithm

1. Compute all the maximal sets $\mathcal{S}$ of neighbor separators such that any clique bordered by $\mathcal{S}$ has at most $k+1$ vertices. Let $\mathcal{F}$ be the family of these separators and $\Gamma_{\mathcal{F}}$ the union of the sets $\mathcal{S}$.

2. Extract from $\mathcal{F}$ the maximum complete family of separators :
   `do`

   - choose if possible an $S \in \Gamma_{\mathcal{F}}$ such that for some full bloc (S, C), there is no maximal set $\mathcal{S}$ of neighbor separators associated to (S, C) with $\mathcal{S} \in \mathcal{F}$.
   - $\mathcal{F} = \mathcal{F} - \{\mathcal{S} | S \in \mathcal{S}\}$;
     $\Gamma_{\mathcal{F}} = \bigcup_{\mathcal{S} \in \mathcal{F}} \mathcal{S}$.

   `while` such a separator S exists.

3. `return` $\mathcal{F}$.

Clearly, $\mathcal{F}$ is a complete family of separators.

Lemma 6 *Let G be a graph and H be any minimal triangulation of treewidth at most k. Then, at the end of the previous algorithm, any maximal set $\mathcal{S} \subseteq \Delta_H$ of neighbor separators of H is in $\mathcal{F}$.*

Proof. Let $\mathcal{F}' = \{\mathcal{S}'_1, \mathcal{S}'_2, \ldots, \mathcal{S}'_q\}$ be the maximum sets of neighbor separators of H. For any $\mathcal{S} \in \mathcal{F}'$, any clique $\Omega_{\mathcal{S}}$ bordered by $\mathcal{S}$ has at most $k+1$ vertices, so $\mathcal{S}$ will be in $\mathcal{F}$ after the first step of the algorithm.

Moreover, let $S \in \Delta_H$ be a minimal separator of H and G. For any full block (S, C) of S, consider a maximal clique Ω of H included in (S, C) such that $S \subset \Omega$. Then the maximal set of neighbor separators $\mathcal{S}'$ that borders Ω is an element of $\mathcal{F}'$, so an element of $\mathcal{F}$. It remains that no $\mathcal{S}' \in \mathcal{F}'$ will be eliminated from $\mathcal{F}$ during the second step. $\diamond$

The next step is to obtain from any non-empty complete family of separators $\mathcal{F}$ a minimal triangulation H of G such that the maximal sets of neighbor separators of H are elements of $\mathcal{F}$.

We will construct a tree $\mathcal{T} = (I, E_T)$ and we label each of its nodes by a subset of $V(G)$. The nodes will be of two types :

- "treated", in which case the label will be some maximal clique Ω bordered by some set of separators $\mathcal{S} \in \mathcal{F}$.
- "untreated", in which case the node is a leaf of the tree and the label of the node is a full block (S, C).

$\mathcal{T}$ will be a clique tree for a minimal triangulation of G.

Extraction algorithm

1. `Initialization step`
   - Chose a maximal set of neighbor separators $\mathcal{S} \in \mathcal{F}$. Chose a clique Ω bordered by $\mathcal{S}$.
   - Create the root of $\mathcal{T}$, labeled with Ω. Mark the root as "treated".
   - for all full blocks (S, C) such that $S \in \mathcal{S}$ and $\Omega \not\subseteq (S, C)$, create a leaf as son of the root and label it with (S, C). Mark the leaf as "untreated".
   - put $\mathcal{F}' = \{\mathcal{S}\}$.
2. `Expansion step`
   - Choose an untreated leaf N, labeled (S, C). Choose a maximal set of neighbor separators $\mathcal{S} \in \mathcal{F}$ associated to (S, C). Let Ω be the clique bordered by $\mathcal{S}$ included in (S, C).
   - Label the node with Ω and mark it as treated.
   - for all the full blocks (S', C') such that $S' \in \mathcal{S}$, $S' \not\subseteq S$ and $\Omega \not\subseteq (S', C')$, create a leaf node as son of N, mark it as "untreated" and label it by (S', C').
   - put $\mathcal{F}' = \mathcal{F}' \cup \{\mathcal{S}\}$.
3. `repeat` the expansion step `until` all the nodes of the tree become "treated".
4. `return` $H = G_{\Gamma_{\mathcal{F}'}}$.

Notice first that when we expand a node labeled (S, C), the labels (S_i, C_i) of its sons will be strictly included in (S, C), so the algorithm terminates.

Lemma 7 *At any moment of the execution, all the vertices of G are in some label of $\mathcal{T}$.*

Proof. Remark that if S is a minimal separator of a graph G and $C \in \mathcal{C}(S)$ is a non-full connected component of $G - S$, then the set $S' = \{x \in S | x$ is adjacent to some vertex of $C\}$ is a minimal separator of G and C is full component of S'. Indeed, consider a vertex $a \in C$ and a vertex b in a full component D of S. Clearly S' separates a and b. Moreover, for any $x \in S'$, there is a path from a to x included in $C \cup \{x\}$ and a path from x to b in $D \cup \{x\}$. This proves that $S' - \{x\}$ is not an a, b separator, so we conclude that S' is a minimal a, b separator.

Consider now a maximal set of neighbor separators $\mathcal{S}$ of G. If (S, C) is a non-full block of some $S \in \mathcal{S}$, we have proved that there is a minimal separator $S' \subset S$ such that (S', C) is a full block of S'. But we have $S' \in \mathcal{S}$ by theorems 3 and 4. Consequently, if $\mathcal{S}$ is a maximal set of separators of G, any vertex x is in some full block (S, C) with $S \in \mathcal{S}$. We deduce that at any moment of the execution of the extraction algorithm, any vertex of G is in some label of the tree $\mathcal{T}$. $\diamond$

Lemma 8 *At any time of the algorithm, for any untreated leaf labeled (S, C), C does not intersect any other label of the tree $\mathcal{T}$.*

Proof. Let us prove the property after the initialization step. Let (S_1, C_1) be the label of a leaf. Let Ω be the label of the root. Clearly $C_1 \cap \Omega = \emptyset$ by construction. Now let (S_2, C_2) be the label of another leaf and suppose that C_1 intersects (S_2, C_2). Since $S_2 \subset \Omega$, we must have $C_1 \cap C_2 \neq \emptyset$. Notice that C_1 and C_2 are connected components of $G - \Omega$. Consequently, $C_1 = C_2$ contradicting the fact that the labels (S_1, C_1) and (S_2, C_2) are different.

Consider now an expansion step. Let N be an untreated node labeled (S, C); C does not intersect any other label of $\mathcal{T}$. Let (S_1, C_1) be the label of a new leaf, obtained by the expansion of N. Like for the initialization step, C_1 does not intersect the new label Ω of N or the label (S_2, C_2) of a new created leaf. Moreover, $C_1 \cap S = \emptyset$ and we have $C_1 \subseteq C$. Since C does not intersect any other label of the tree, C_1 cannot intersect any other label. $\diamond$

At any moment of the execution, let $\mathcal{T}'$ the subtree of $\mathcal{T}$ formed by the treated nodes. We take $\Gamma = \{S \in \Delta_G | S$ is in the label of some node of $\mathcal{T}'\}$, $V' = \{x \in V | x$ is in the label of some node of $\mathcal{T}'\}$ and $H = G_\Gamma[V']$.

Lemma 9 *At any moment of the execution, Γ is a set of pairwise parallel separators and $\mathcal{T}'$ is a clique tree of $H = G_\Gamma[V']$.*

Proof. The property is obvious after the initialization step.

Suppose that the property is true at a certain moment of the processing and we expand a node N. Let (S, C) be the label of N. Let $\mathcal{S} \in \mathcal{F}$ be the maximal set of neighbor separators associated to (S, C) by the algorithm and Ω the clique bordered by $\mathcal{S}$ in (S, C). We denote by Γ_e, V'_e, $\mathcal{T}'_e$, H_e the new parameters, after the expansion step.

First, we prove that the separators of Γ_e are pairwise parallel. Clearly we have $\Gamma_e = \Gamma \cup \mathcal{S}$. It is sufficient to show that if $T \in \Gamma$ and $U \in S$ we have $T || U$. Because we have $T \subseteq V'$ and $U \subseteq (S, C)$ and $C \cap V' = \emptyset$ (lemma 8), S separates T and U so $T || U$.

Now, we prove that $\mathcal{T}'_e$ is a clique tree of $H_e = G_{\Gamma_e}[V'_e]$. Notice that $V'_e = V' \cup \Omega$. By corollary 2, Ω is a maximal clique of H_e. Moreover, H_e has exactly one more maximal clique than H, which is Ω, since the edges in H_e are either in H or in $H_e[\Omega]$. So the labels of $\mathcal{T}_e'$ are exactly the maximal cliques of H_e. It remains to show that for any label $\Omega' \neq \Omega$ of $\mathcal{T}_e'$, the set $\Omega \cap \Omega'$ is contained in every clique on the path connecting Ω and Ω' in $\mathcal{T}_e'$. Let Ω_0 be the neighbor of Ω in $\mathcal{T}_e'$. All we have to prove is that $\Omega \cap \Omega' \subseteq \Omega_0$. By lemma 8, $\Omega \cap V' = S$. But clearly $\Omega' \subseteq V'$ and $S \subset \Omega_0$, so $\Omega \cap \Omega' = S \cap \Omega' \subset \Omega_0$. $\diamond$

We deduce directly from lemmas 7 and 9:

Theorem 5 *Let G be a graph, $\mathcal{F}$ be a complete family of minimal separators of G. If $\mathcal{F}$ is not empty, we can construct a minimal triangulation H of G such that any maximal set of neighbor separators of H is an element of $\mathcal{F}$.*

Corollary 3 *Let $\mathcal{G}$ be a class of graphs having a polynomial time algorithm for computing the family of all maximal sets of neighbor separators for every $G \in \mathcal{G}$. Then for any graph in $\mathcal{G}$ the treewidth is polynomially tractable.*

Proof. To decide if the treewidth of any $G \in \mathcal{G}$ is at most a fixed k, it is sufficient to :

1. apply the elimination algorithm to obtain a complete set of separators $\mathcal{F}$.
2. `if` $\mathcal{F}$ is empty `then` the treewidth of G is greater then k, unless G is a clique with at most $k+1$ vertices.
3. `else` apply the extraction algorithm on $\mathcal{F}$ to get a triangulation H of G with treewidth at most k.

By lemma 6, if the treewidth of G is at most k the elimination algorithm will produce a non-empty family of separators $\mathcal{F}$. Conversely, if the family $\mathcal{F}$ returned by the elimination algorithm is non-empty, by theorem 5 the extraction algorithm will produce a triangulation of treewidth at most k.

Concerning the complexity of this algorithm, remark that if the calculus of all the maximal sets of neighbor separators is done in polynomial time, both the elimination algorithm and the extraction algorithm work in polynomial time. $\diamond$

5 Application to some classes of graphs

Several classes of graphs have "few" minimal separators, in the sense that the number of minimal separators of such a graph is polynomially bounded in the size of the graph. Moreover, an algorithm given in [8] computes all the minimal separators of these graphs.

For these classes of graphs we also have algorithms computing the treewidth in polynomial time, using the minimal separators (cf. [9], [2], [5], [14], [10]). Different proofs have been given for each of these algorithms. We have remarked that, for computing the treewith of a graph, all these algorithms compute all the maximal sets of neighbor separators of the graph. Therefore, our approach unifies the cited algorithms.

AT-free graphs are graphs with no asteroidal triple where three vertices constitute an asteroidal triple if between every two of them there exists a path avoiding the neighborhood of the third. One can prove (see for example [10]) that a graph G is AT-free if and only if for any three pairwise separators $S, T, U \in$

Δ_G, one of them separates the two others. It follows that if $\mathcal{S}$ is a maximal set of neighbor separators of an AT-free graph, $\mathcal{S}$ ordered by inclusion has at most two maximal elements. Therefore, it is easy to compute the maximal sets of neighbor separators for any AT-free graph with "few" separators. The class of AT-free graphs includes the *cocomparability graphs* and in particular the *permutation graphs* (which have $\mathcal{O}(n^2)$ minimal separators for graphs with n vertices [2]) or the *d-trapezoid graphs*, which also have a polynomial number of separators for any fixed d.

Circle and circular arc graphs are obtained from an intersection model. A graph is a *circle graph* if we can associate every vertex of the graph to a chord of a circle such that two vertices are adjacent in the graph if and only if the corresponding chords cross. The circle and its chords are the *circle model* of the graph. In the same manner, a graph is a *circular arc graph* if the vertices can be associated to some arcs of a circle and two vertices cross if and only if the corresponding arcs overlap. For these "geometrical" classes of graphs, the minimal separators can be modeled by *scan-lines*. In the circle model of the circle graphs, we can add a *scan-point* between every two consecutive endpoints of the chords. For the circular arc graphs, the scan-points will be the middle of every circular arc. The scan-lines are the straight line segments between two scan-points. Using the results of [5] and [14], we deduce that the maximal sets of neighbor separators of these graphs correspond to some triangles of scanlines. Therefore, all the maximal sets of neighbor separators of circle and circular arc graphs can be computed in polynomial time.

Chordal bipartite graphs are bipartite graphs for which every cycle of length at least 6 has a chord. It was shown in [9] that for any bipartite chordal graph G, any maximal set of neighbor separators is characterized by two maximal bipartite subgraphs. Since any chordal bipartite graph G with e edges has $\mathcal{O}(e)$ complete bipartite subgraphs, one can compute in polynomial time all the maximal sets of neighbor separators of G.

Weakly triangulated graphs are graphs G such that neither G, nor its complement $\overline{G}$ have induced cycle of more than four vertices. They were introduced in [4] and they contain the chordal bipartite graphs, the chordal graphs and the distance hereditary graphs. No algorithm computing the treewidth of these graphs was known up to now, although a polynomial bound for the number of the minimal separators was given in [6]. Using the recognition algorithm of weakly triangulated graphs given in [13], we could prove that a weakly triangulated graph G has at most $\overline{e}$ minimal separators, where $\overline{e}$ is the number of edges of $\overline{G}$. Moreover, we proved that G has at most $2\overline{e}$ maximal sets of neighbor separators and they are easily tractable in polynomial time. We obtain a polynomial algorithm computing the treewidth of weakly triangulated graphs. The proofs of these facts will be given in the full version of the paper.

6 Conclusion

In this paper we have given a characterization of minimal triangulation of graphs in terms of maximal sets of neighbor separators. This allows us to unify several algorithms computing the treewidth of some classes of graphs. The same techniques can also be used for computing the minimum fill-in of these classes.

It would be interesting to know more about the maximal sets of neighbor separators. Some questions arise naturally: can we compute in polynomial time all the maximal sets of neighbor separators of any graph with "few" minimal separators ? Is the number of maximal sets of neighbor separators of a graph polynomially bounded in the number of minimal separators ? Finally, for classes of graphs with "many" maximal sets of neighbor separators, it would be important to know if "few" well chosen maximal sets of neighbor separators can be sufficient to compute or to approximate the treewidth.

References

[1] J. R. S. Blair and B. Peyton. An introduction to chordal graphs and clique trees. In A. George, J. R. Gilbert, and J. H. U. Liu, editors, *Graph Theory and Sparse Matrix Computations*, pages 1–29. Springer, 1993.

[2] H.L. Bodlaender, T. Kloks, and D. Kratsch. Treewidth and pathwidth of permutation graphs. In *Proceedings of the 20th International Colloquium on Automata, Languages and Programming (ICALP'93)*, volume 700 of *Lecture Notes in Computer Science*, pages 114–125. Springer-Verlag, 1993.

[3] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[4] R. Hayward. Weakly triangulated graphs. *J. Combin. Theory ser. B*, 39:200–208, 1985.

[5] T. Kloks. Treewidth of circle graphs. In *Proceedings 4th Annual International Symposium on Algorithms and Computation (ISAAC'93)*, volume 762 of *Lecture Notes in Computer Science*, pages 108–117. Springer-Verlag, 1993.

[6] T. Kloks, H.L. Bodlaender, H. Mller, and D. Kratsch. Computing treewidth and minimum fill-in: All you need are the minimal separators. In *Proceedings First Annual European Symposium on Algorithms (ESA'93)*, volume 726 of *Lecture Notes in Computer Science*, pages 260–271. Springer-Verlag, 1993.

[7] T. Kloks, H.L. Bodlaender, H. Mller, and D. Kratsch. Erratum to the ESA'93 proceedings. In *Proceedings Second Annual European Symposium on Algorithms (ESA'94)*, volume 855 of *Lecture Notes in Computer Science*, page 508. Springer-Verlag, 1994.

[8] T. Kloks and D. Kratsch. Finding all minimal separators of a graph. In *Proceedings 11th Annual Symposium on Theoretical Aspects of Computer Science (STACS'94)*, volume 775 of *Lecture Notes in Computer Science*, pages 759–768. Springer-Verlag, 1994.

[9] T. Kloks and D. Kratsch. Treewidth of chordal bipartite graphs. In *Proceedings 11th Annual Symposium on Theoretical Aspects of Computer Science (STACS'94)*, volume 775 of *Lecture Notes in Computer Science*, pages 759–768. Springer-Verlag, 1994.

[10] A. Parra. *Structural and Algorithmic Aspects of Chordal Graph Embeddings.* PhD thesis, Technische Universität Berlin, 1996.

[11] A. Parra and P. Scheffler. How to use the minimal separators of a graph for its chordal triangulation. In *Proceedings of the 22nd International Colloquium on Automata, Languages and Programming (ICALP'95)*, volume 994 of *Lecture Notes in Computer Science*, pages 123–134. Springer-Verlag, 1995.

[12] N. Robertson and P. Seymour. Graphs minors. II. Algorithmic aspects of tree-width. *J. of Algorithms*, 7:309–322, 1986.

[13] J. Spinrad and R. Sritharan. Algorithms for weakly triangulated graphs. *Discrete Applied Mathematics*, 59:181–191, 1995.

[14] R. Sundaram, K. Sher Singh, and C. Pandu Rangan. Treewidth of circular-arc graphs. *SIAM J. Discrete Math.*, 7:647–655, 1994.