Efficient clique decomposition of a graph into its atom graph

Anne Berry, Romain Pogorelcnik, Geneviève Simonet

Efficient clique decomposition of a graph into its atom graph

Anne Berry[1] Romain Pogorelcnik[*]
Geneviève Simonet[4]

Research Report LIMOS/RR-10-07

10 mars 2010

[1] LIMOS UMR CNRS 6158, Ensemble Scientifique des Cézeaux, F-63 173 Aubière, France, berry@isima.fr

[4] LIRMM, 161 Rue Ada, F-34392 Montpellier, France, simonet@lirmm.fr

Abstract

We explain how to organize the atoms resulting from clique minimal separator decomposition into a metagraph which we call the atom graph, and give an efficient recursive algorithm to compute this graph at no extra cost than computing the atoms.

Keywords: Atom graph, clique separator decomposition

1 Introduction

Clique separator decomposition was introduced by Tarjan in [16] as useful for a divide-and-conquer approach for hard problems such as minimum fill-in, maximum clique, graph coloring and maximum independent sets. The process consists in repeatedly finding a clique separator S of a graph $G = (V, E)$ and decomposing G by copying S into the dif and only iferent connected components of $G - S$, obtaining in the end a set of induced subgraphs having no clique separator called 'atoms'. Tarjan presented an $O(nm)$ time algorithm to implement this process in $O(nm)$ time by repeatedly testing transitory neighborhoods of a minimal triangulation $H = (V, E + F)$ of G, as will be detailed further.

Tarjan left open the problem of unicity; this was solved by Leimer [12], who showed that using only clique separators which are also minimal separators will result in a unique set of atoms which are maximally large and in minimum number, as they are exactly the maximal induced connected subgraphs having no clique separator. Leimer modified Tarjan's algorithm to compute this unique decomposition, maintaining the same complexity.

Recently, this decomposition has been applied with good results to graph modelizations of data, for instance on microarray dataset in yeast, where the atoms were proved to provide very good gene groupings with respects to their function [9]. Good results have also been obtained in data from text mining, see *e.g.*[5]. One of the advantages of this decomposition is that the subgraphs obtained allow an overlap, so that the modelized objects (such as genes), can be represented as participating in several groups (such as gene functions).

In these applications, it turns out that not only do the atoms define 'clusters' which are highly significant, but that another interesting object is the 'atom graph' (introduced in [9]), which is a partial subgraph of the intersection graph of the atoms, with an edge between two atoms A and B if their intersection separates set $A \setminus B$ from set $B \setminus A$. This atom graph is a compact representation of large graphs which can yield structural insight: in [9], this atom graph turned out to be an interval graph which precisely described the four consecutive temporal classes of genes that coincide with the major biological processes of sporulation in yeast. The atom graph can also be used for visualization purposes. The corresponding intersection graph has been studied for chordal graphs as the *clique graph* [1] (the atoms in a chordal graph are the maximal cliques), but not for a general graph.

In view of these new applications to large graphs, we investigate how to construct the atom graph efficiently. Since there are less than n atoms, a straightforward approach will yield the atom graph in $O(n^2m)$. In this paper, we present a recursive $O(nm)$ time process, thus enabling the construction of the atom graph at no extra cost.

2 Preliminaries

Given a graph $G = (V, E)$, we denote $n = |V|$ and $m = |E|$. For any subset S of V, $G(S)$ denotes the subgraph of G induced by S. For any vertex v of G, $N_G(v)$ denotes the neighborhood of v in G. For a given set of vertices $X \subseteq V$, $N_G(X) = \cup_{v \in X} N_G(v) \setminus X$ and $N_G[X] = \cup_{v \in X} N_G(v) \cup X$. We will omit the subscripts when there is no ambiguity. For any graphs $G = (V, E)$ and $G' = (V, E')$, we say that $G \subseteq G'$ if $E \subseteq E'$. A set of vertices X is *complete* or a *clique* if its vertices are pairwise adjacent. We will say that we *saturate* a set of vertices X when we add to the graph all the edges necessary to make X into a clique.

Separation. In a connected graph $G = (V, E)$, a subset S of V is a *separator* of G if $G(V \setminus S)$ is disconnected. For any non-adjacent vertices a and b in $V \setminus S$, S is an *ab-separator*, or S *separates* a and b, if a and b are in dif and only iferent connected components of $G(V \setminus S)$, called *components of* S *in* G. S is a *minimal ab-separator* if it is an inclusion-minimal ab-separator, and a *minimal separator* if there is some pair $\{a, b\}$ such that S is a minimal ab-separator. Equivalently, S is a minimal separator if there are at least two components C_1 and C_2 of S in G such that $N(C_1) = N(C_2) = S$ (such components are called *full* components). (Minimal) AB-separators are defined in the same way as (minimal) ab-separators for any disjoint connected subsets A and B of $V \setminus S$. Given two graphs $G = (V, E)$ and $(H = V, E')$, and a subset S of V, if S has the same components in G as in H, and if, for each such component C, $N_G(C) = N_H(C)$, we say that S has *the same components and neighborhoods* in G and H.

Chordal graphs. A graph is *chordal*, or *triangulated*, if it contains no chordless cycle of length ≥ 4. A connected graph is chordal if and only if all its minimal separators are cliques. A chordal graph $H = (V, E + F)$ is called a *triangulation* of $G = (V, E)$. The set F of edges, of size $|F| = f$, which are added to obtain a triangulation is called a *fill*. A triangulation H is *minimal* if no strict subset of F can be added to G to obtain a triangulation.

Property 2.1 *[2, 14] Let G be a connected graph, let H be a minimal triangulation of G.*
a) Every clique minimal separator of G is a minimal separator of H.
b) Every minimal separator of H is a minimal separator of G, with the same components and neighborhoods.

Property 2.2 *(Unique Chord Property) [15] Let $G = (V, E)$ be a graph, and let $H = (V, E + F)$ be a triangulation of G. H is a minimal triangulation of G if and only if each edge in F is the unique chord of a cycle in H of length 4.*

Given a graph G and an ordering π of the vertices of G, a triangulation of G called G^+_π is obtained from G by repeatedly, saturating the neighborhood in the current graph G' of the next vertex x in ordering π and remove x from G'. An ordering π is a *minimal elimination ordering (meo)* if there is no ordering π' such that $G^+_{\pi'} \subset G^+_\pi$; in that case G^+_π is a minimal triangulation of G. Given a graph $G = (V, E)$ and an ordering π of V, $\pi(i)$ will denote the vertex of number i, and $\pi^{-1}(x)$ will denote the number of vertex x; the *upper neighborhood* of a vertex x will be $Madj(x) = \{y \in V | \pi^{-1}(y) > \pi^{-1}(x)\}$.

3 Construction of the atom graph

In this section, we will present and prove an $O(nm)$ time algorithm to construct the atom graph.

Let us first give some useful definitions and properties for atoms.

Definition 3.1 *[12] An atom of a connected graph $G = (V, E)$ is an inclusion-maximal subset of V inducing a connected graph having no clique separator.*

Property 3.2 *[12] The intersection of two distinct atoms is a clique.*

Characterization 3.3 *[12] A connected graph is chordal if and only if all its atoms are cliques.*

Property 3.4 *[12] For any connected graph G, the graph obtained from G by saturating its atoms, denoted by G^*, is chordal.*

Below, we recall how the clique minimal separator decomposition can be computed on a graph G [16, 12, 4]:

- Compute a meo π and the associated minimal triangulation $H = G^+_\pi$ of G, and initialize V' as V.
- For $i = 1$ to n
  If $S_i = Madj_H(\pi(i))$ is a clique minimal separator of G, then
  - Find the component C of S_i in $G(V')$ containing $\pi(i)$;
  - Decompose the current vertex set V' into $C \cup S_i$, which is an atom of G, and $V' \setminus C$, which becomes the current set V'.

This process defines a binary decomposition tree, which we call the *atom tree*:

Definition 3.5 *An atom tree of a connected graph $G = (V, E)$ is a labeled binary tree T recursively defined as follows: if G has no clique separator then T is reduced to a node labeled with V, otherwise the root of T is labeled with a clique minimal separator S of G, its left child is a leaf labeled with a subset A of V in the form $C \cup S$, where C is a full component of S in G, such that $G(A)$ has no clique separator, and the subtree T' of T rooted at the right child of its root is an atom tree of $G(V \setminus C)$. In that case we say that T is defined by (A, S, T').*

Property 3.6 *[12] Let G be a connected graph, and T be an atom tree of G. Each leaf of T is labeled with an atom of G, and each atom of G appears once as a leaf label in T.*

Example 3.7 *Figure 1 gives a LEX M ordering of a graph G, with the corresponding atom tree.*

We now define the metagraph which we call the atomgraph:

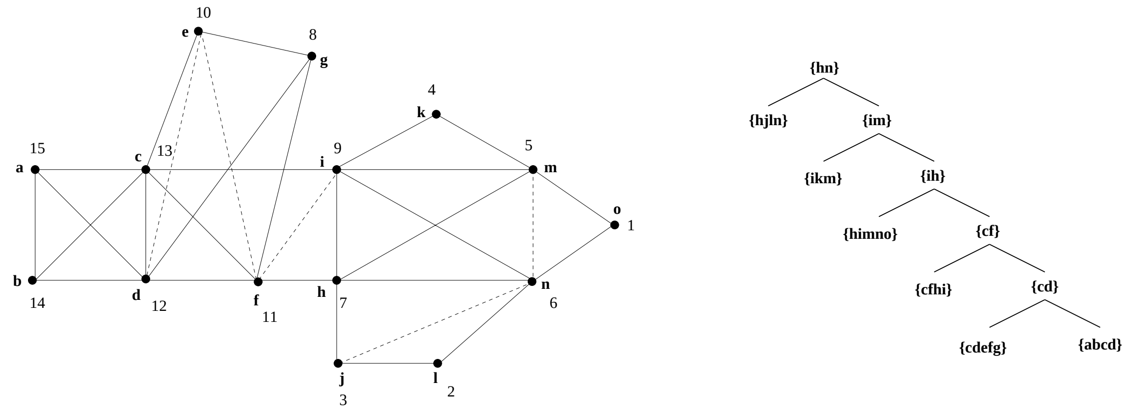

Figure 1: Graph G with an LEX M ordering and the corresponding atom tree. On G, the fill edges are represented by dashed lines.

Definition 3.8 *The atom graph of a connected graph G, denoted by $AG(G)$, is defined as follows: its vertices are the atoms of G, and its edges are the pairs $\{A, B\}$ of atoms of G such that $A \cap B$ is a clique minimal $(A \setminus B)(B \setminus A)$-separator.*

In turns out that if two atoms are separated, then they are minimally separated:

Property 3.9 *Two atoms A and B are adjacent in the atom graph of G if and only if $A \cap B$ separates $A \setminus B$ and $B \setminus A$ in G.*

In order to prove Property 3.9, we use the following lemma:

Lemma 3.10 *Let G be a connected graph, let S be a clique of G and let B be an atom of G. Then $B = S$ or there is a component D of S in G such that $B \subseteq N_G[D]$.*

Proof: If $B \subseteq S$ then $B = S$ since S has no clique separator. Otherwise let D be a component of S in G such that $B \cap D \neq \emptyset$. $B \subseteq N_G[D]$ because otherwise $B \cap N_G(D)$ would be a clique (as a subset of clique S) separating any vertex of $B \cap D$ and any vertex of $B \setminus N_G[D]$. □

Proof: (of Property 3.9) The condition is obviously necessary. Let us show that it is sufficient. We assume that $A \cap B$ separates $A \setminus B$ from $B \setminus A$. By Property 3.2 $A \cap B$ is a clique. $A \setminus B$ and $B \setminus A$ are connected, since otherwise $A \cap B$ would be a clique separator in $G(A)$ or in $G(B)$. Let C_1 and C_2 be the components of $A \cap B$ containing $A \setminus B$ and $B \setminus A$ respectively. By Lemma 3.10 $A \subseteq N_G[C_1]$, so $A \cap B \subseteq N_G(C_1)$ and therefore C_1 is a full component of $A \cap B$. Similarly, C_2 is full, and is distinct from C_1 by assumption. Hence $A \cap B$ is a clique minimal $(A \setminus B)(B \setminus A)$-separator. □

Example 3.11 *Figure 2 gives the atom graph of graph G from Figure 1.*

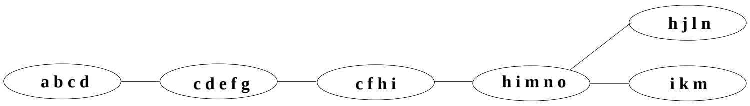

Figure 2: The atom graph of graph G from Figure 1

.

A graph may have several atom trees, depending on the order in which clique minimal separators and atoms are chosen. However, in any atom tree of any connected graph G, the leaf labels are exactly the atoms of G (Property 3.6), and we will show that the internal node labels are exactly the clique minimal separators of G, each clique minimal separator appearing as an internal node label as many times as it has full components in G, minus one (Property 3.12). Thus the labels are the same in all atom trees of G, with the same number of occurrences of each label.

Property 3.12 *Let G be a connected graph, and T be an atom tree of G. Each internal node of T is labeled with a clique minimal separator of G, and each clique minimal separator of G appears as an internal node of T as many times as it has full components minus one in G.*

In order to prove Property 3.12, we use the following lemma:

Lemma 3.13 *Let $G = (V, E)$ be a graph, let S be a clique of G, let C be a component of S in G, let $G' = G(V \setminus C)$, let R be a subset of $V \setminus C$ and let C_1 be the component of R in G containing C. Then the components of R in G' are obtained from its components in G by replacing C_1 by $C_1 \setminus C$ if $C \subset C_1$ and removing C_1 otherwise, i.e. if $C = C_1$, and the components of R have the same neighborhoods in G and in G'.*

Proof: The result is evident if $C = C_1$. We assume now that $C \subset C_1$. We have to show that $G(C_1 \setminus C)$ is connected and that $N_G(C_1) = N_{G'}(C_1 \setminus C)$. $G(C_1 \setminus C)$ is connected since a shortest path in $G(C_1)$ between two vertices of $C_1 \setminus C$ is also a path in $G(C_1 \setminus C)$ (otherwise this path would contain two non-consecutive vertices of $N_G(C) \subseteq S$ and therefore would have a chord). $N_{G'}(C_1 \setminus C) \subseteq N_G(C_1)$ since $G' = G(V \setminus C)$. Let us show that $N_G(C_1) \subseteq N_{G'}(C_1 \setminus C)$. Let $x \in N_G(C_1)$. Let us show that $x \in N_{G'}(C_1 \setminus C)$. $x \in R$, so x is a vertex of G'. It is sufficient to show that $(C_1 \setminus C) \cap N_G(x) \neq \emptyset$. Let $y \in C_1 \cap N_G(x)$. If $y \notin C$ then $y \in (C_1 \setminus C) \cap N_G(x)$ and we are done. Now if $y \in C$ then $x \in N_G(C) \subseteq S$, and as $C \subset C_1$, $N_G(C) \cap C_1 \neq \emptyset$. Let $z \in N_G(C) \cap C_1$. As x and z are both in clique S and are distinct (since $z \in C_1$ and $x \notin C_1$), $z \in N_G(x)$, and therefore $z \in (C_1 \setminus C) \cap N_G(x)$. □

Proof: (of Property 3.12) Let us show this by induction on the number k of atoms of G. If $k = 1$ then G has no clique minimal separator and T has no internal node, so the property holds. We suppose that it holds if G has k atoms. Let us show that it holds if G has $k + 1$ atoms. Let T be defined by (A, S, T'), let $C = A \setminus S$ and let $G' = G(V \setminus C)$. We first show that any clique minimal separator of G is in G'. We suppose for contradiction that there is a clique minimal separator R of G such that $R \cap C \neq \emptyset$. As R is a clique $R \subseteq N_G[C] = A$. Let C_1, ..., C_p be the components of S in G'. $G(A)$ is obtained from G by successively removing the vertices of C_i for i from 1 to p. As $R \neq N(C_i)$ for each i from 1 to p, (since $N(C_i) \subseteq S$ and $R \cap C \neq \emptyset$), it follows from Lemma 3.13 applied p times that R has as many full components in $G(A)$ as in G. Hence R is a clique minimal separator of $G(A)$, a contradiction. Thus any clique minimal separator R of G is a clique of G', and therefore by Lemma 3.13 has as many full components in G' as in G if $R \neq S$, and one less if $R = S$. We conclude with the induction hypothesis, since G' has k atoms and the internal node labels in T' are the same as in T, except that S has one occurrence less. □

To compute the atom graph of G, we will apply our recursive Definition 3.5. We will use atom tree T defined by (A, S, T'). Recall that S is a clique minimal separator, A is the first atom found for T, and T' is the remaining subtree. We will recursively compute the atom graph of $G' = G(V \setminus (A \setminus S))$ and add atom A and the edges incident to A, which are defined as follows:

Property 3.14 *Given an atom tree of G defined by (A, S, T'), the atom graph of G is obtained from the atom graph of $G' = G(V \setminus (A \setminus S))$ by adding vertex A and edge AB for each atom B of G' such that $N_{G'}(D) \subseteq B$, where D is the unique component of S in G' such that $B \subseteq N_{G'}[D]$.*

Property 3.14 follows directly from the two lemmas below:

Lemma 3.15 *Let $G = (V, E)$ be a connected graph, let $\mathscr{A}$ be the set of atoms of G, let T be an atom tree of G defined by (A, S, T'), and let $G' = G(V \setminus (A \setminus S))$. Then $AG(G') = AG(G)(\mathscr{A} \setminus \{A\})$.*

Proof: By Property 3.6, the set of atoms of G' is the set of leaf labels of T', i.e. $\mathscr{A} \setminus \{A\}$. It follows from Lemma 3.13 that for any atoms B and B' of G', $B \cap B'$ separates $B \setminus B'$ and $B' \setminus B$ in G' if and only if it separates them in G. Hence by Property 3.9 $AG(G')$ is an induced subgraph of $AG(G)$. □

Lemma 3.16 *Let $G = (V, E)$ be a connected graph, let T be an atom tree of G defined by (A, S, T'), let $G' = G(V \setminus (A \setminus S))$, and let B be an atom of G'. Then there is a component D of S in G' such that $B \cap D \neq \emptyset$, $B \subseteq N_{G'}[D]$ and the following equivalence holds: AB is an edge of $AG(G)$ if and only if $N_{G'}(D) \subseteq B$.*

Proof: By Lemma 3.15 B is also an atom of G. There is a component D of S in G' such that $B \cap D \neq \emptyset$, because otherwise B would be a subset of S and therefore a strict subset of atom A in G, which contradicts the maximality condition in the definition of an atom. By Lemma 3.10 $B \subseteq N_{G'}[D]$. Let us show that AB is an edge of $AG(G)$ if and only if $N_{G'}(D) \subseteq B$, i.e., by Property 3.9, that $N_{G'}(D) \subseteq B$ if and only if $A \cap B$ separates $A \setminus B$ and $B \setminus A$. As $B \subseteq N_{G'}[D]$, $A \cap B = B \cap N_{G'}(D)$ and $B \setminus A = B \cap D$. If $N_{G'}(D) \subseteq B$

then $A \cap B = N_{G'}(D)$. In that case, as D is a component of $N_{G'}(D)$ in G containing $B \cap D$ but not $A \setminus B$, $A \cap B$ separates $A \setminus B$ and $B \setminus A$. We assume now that $N_{G'}(D) \not\subseteq B$. Let $x \in N_{G'}(D) \setminus B$. As x is in $A \setminus B$ and is adjacent to some vertex in D, the component of $A \cap B$ containing $A \setminus B$ also contains D, and therefore $B \cap D = B \setminus A$. Hence $A \cap B$ does not separate $A \setminus B$ and $B \setminus A$. □

Property 3.14 yields a recursive algorithm to compute the atom graph:

```
Algorithm AtomGraph
input  : A graph $G = (V, E)$ and an atom tree $T$ of $G$.
output: The atom graph $AG(G)$ of $G$.

if $T$ has a unique node then
    $(\mathscr{A}, F) \leftarrow (\{V\}, \emptyset)$
else
    Let $S$ be the label of the root of $T$;
    Let $A$ be the label of its left child;
    Let $T'$ be the subtree of $T$ rooted at its right child;
    $G' \leftarrow G(V \setminus (A \setminus S))$;
    Compute the components $C_1, C_2, ..., C_p$ of $S$ in $G'$
    as well as the integers $|N_{G'}(C_i)|$ for $i$ from 1 to $p$
    and $num(x)$ for each $x$ in $V \setminus A$, with $num(x) = i$ if and only if $x \in C_i$;
    $(\mathscr{A}, F) \leftarrow AtomGraph(G', T')$;
    foreach $B$ in $\mathscr{A}$ do
        $cpt \leftarrow 0$;
        foreach $x$ in $B$ do
            if $x \in S$ then
                $cpt \leftarrow cpt + 1$
            else
                $i \leftarrow num(x)$
        if $cpt = |N_{G'}(C_i)|$ then
            $F \leftarrow F \cup \{AB\}$
    $\mathscr{A} \leftarrow \mathscr{A} \cup \{A\}$
$AG(G) \leftarrow (\mathscr{A}, F)$;
```

Complexity of Algorithm AtomGraph.

Theorem 3.17 *The atom graph of a connected graph G can be computed in $O(nm)$ time using Algorithm AtomGraph.*

To prove Theorem 3.17, we need to evaluate the sum of the sizes of the atoms as follows:

Lemma 3.18 *Let $G = (V, E)$ be a connected graph, let $\mathscr{A}$ be the set of atoms of G. Then $\Sigma_{B \in \mathscr{A}} |B| \leq n + m$.*

Proof: We prove this by induction on the number k of atoms of G. If $k = 1$ then the property trivially holds. We assume that it holds if $|\mathscr{A}| = k$. Let us show that it holds if $|\mathscr{A}| = k + 1$. Let T be an atom tree of G defined by (A, S, T'), let $C = A \setminus S$, let $G' = G(V \setminus C)$, and let n', m' and $\mathscr{A}'$ denote the numbers of vertices and edges and the set of atoms of G' respectively. By Property 3.6 and induction hypothesis, $\Sigma_{B \in \mathscr{A}} |B| = \Sigma_{B \in \mathscr{A}'} |B| + |A| \leq n' + m' + |A|$. As $G(C)$ is connected, it has at least $|C| - 1$ edges. As C is a full component of S in G, there are at least $|S|$ edges in G having an endpoint in S and the other endpoint in C. It follows that $m \geq m' + (|C| - 1) + |S| = m' + |A| - 1$, so $m' + |A| \leq m + 1$ and $\Sigma_{B \in \mathscr{A}} |B| \leq n' + m + 1 \leq n + m$. □

Proof: (of Lemma 3.17) An atom tree T of G can be computed in $O(nm)$ time [12]. Let us show that Algorithm AtomGraph with input G and T can output $AG(G)$ in $O(nm)$ time too. Components C_i and integers $|N_{G'}(C_i)|$ and $num(x)$ can be computed in $O(m)$ time by a search in G', and scanning all atoms of G' also costs $O(m)$ time by Lemma 3.18. The number of recursive calls is bounded by n since the number of vertices of the current graph strictly decreases at each call. Hence Algorithm AtomGraph runs in $O(nm)$ time. □

4 Some atom graph properties.

We will end this paper by proving several properties of the atom graph.

- We investigate which graphs are atom graphs, and have the following partial result:

  Property 4.1 *A graph is an atom graph if and only if it is the atom graph of a chordal graph.*

  Property 4.1 directly follows from Property 3.4 and Lemma 4.2 below:.

  Lemma 4.2 *For any connected graph G, G and G^* have the same clique minimal separators with the same components and neighborhoods, the same atom trees and the same atom graph.*

  In order to prove Lemma 4.2, we need the following lemmas.

  Lemma 4.3 *Let G and H be two graphs with the same vertices, let S be a clique minimal separator of G, let C be a full component of S in G, let $V' = V \setminus C$ (resp. $C \cup S$), let $G' = G(V')$ and $H' = H(V')$. If every clique minimal separator of G is a clique minimal separator of H, with the same components and neighborhoods then every clique minimal separator of G' is a clique minimal separator of H', with the same components and neighborhoods.*

Proof: We first prove it for $V' = V \setminus C$. Let R be a clique minimal separator of G'. By Lemma 3.13 R is also a clique minimal separator of G and its components in G are obtained from its components in G' by adding the component C if $S \subseteq R$ and by replacing the component C_1 containing $S \setminus R$ by $C_1 \cup C$ otherwise, with the same neighborhoods. Hence R is a clique minimal separator of H and has the same components in H as in G, with the same neighborhoods. By Lemma 3.13 again on H and H', which holds since S is also a clique minimal separator of H and C is also a full component of S in H, R has the same components in H' as in G', with the same neighborhoods, and therefore is a clique minimal separator of H'.
We now prove it for $V' = C \cup S$. Let C_1, ..., C_p be the components of S in $G(V \setminus C)$ and $H(V \setminus C)$. G' (resp. H') is obtained from G (resp. H) by successively removing the vertices of C_i for i from 1 to p. Thus it is sufficient to apply the preceding result (for $V' = V \setminus C$) p times. □

Lemma 4.4 *Let G and H be two connected graphs with the same vertices. If G and H satisfy the two following conditions,*
(a) every clique minimal separator of G is a clique minimal separator of H, with the same components and neighborhoods,
(b) for every atom A of G, $H(A)$ has no clique separator,
then they have the same clique minimal separators.

Proof: By Property 3.12 it is sufficient to show that any atom tree of G is also an atom tree of H. We prove this by induction on the number k of atoms of G. If $k = 1$ then by condition (b) H has also a unique atom and the property holds. We suppose that it holds if G has k atoms. Let us show that it holds if G has $k+1$ atoms. Let T be an atom tree of G defined by (A, S, T'). Let us show that T is also an atom tree of H. Let $C = A \setminus S$, let $G' = G(V \setminus C)$ and let $H' = H(V \setminus C)$. T' is an atom tree of G', S is also a clique minimal separator of H and C is also a full component of S in H. By Lemma 4.3 G' and H' also satisfy condition (a). They also satisfy condition (b) since every atom A of G' is also an atom of G by Property 3.6, with $H(A) = H'(A)$. By condition (b) $H(A)$ has no clique separator. Hence T is also an atom tree of H, which completes the proof by induction. □

Lemma 4.5 *Let G and H be two graphs with the same vertices. If G and H have the same clique minimal separators with the same components and neighborhoods then they have the same atom trees and atom graph.*

Proof: Let us show this by induction on the number k of atoms of G. If $k = 1$ then H has also a unique atom since it has no clique minimal separator (and therefore no clique separator) and the property holds. We suppose that it holds if G has k atoms. Let us show that it holds if G has $k+1$ atoms. Let T be an atom tree of G defined by (A, S, T'). Let us show that T is also an atom tree of H and that G and H have the same atom graph. Let $C = A \setminus S$, let $G' = G(V \setminus C)$ and let $H' = H(V \setminus C)$. T' is an atom tree of G', S is also a clique minimal separator of H and C is also a full component of S in H. By Lemma 4.3 G' and H' have the same clique minimal separators, with the same components and neighborhoods. As G' has k atoms, by induction hypothesis T' is also an atom tree of H', and G' and H' have the same atom graph. As A is an atom of G, $G(A)$ has no clique minimal separator. By Lemma 4.3, $H(A)$ has no clique minimal separator either. Hence T is also an atom tree of H. To prove that G and H have the same atom graph, it remains to show that A is adjacent to the same atoms in both atom graphs. This follows from Lemma 3.16 and the fact that S has the sames components in G and in H, with the same neighborhoods. By symmetry, any atom tree of H is also an atom tree of G, which completes the proof by induction. □

Proof: (of Lemma 4.2) By Lemmas 4.5 and 4.4 it is sufficient to show that G and G^* satisfy conditions (a) and (b) of Lemma 4.4. They satisfy condition (b) since every atoms of G is a clique of G^*. Let us show that they satisfy condition (a). Let R be a clique minimal separator of G. By Lemma 3.10 for any atom B of G there is a component D of S in G such that $B \subseteq N_G[D]$. It follows that saturating the atoms of G does not modify the components of R and their neighborhoods. Hence R is also a minimal separator of G^*, with the same components and neighborhoods as in G, and it is a clique of G^* since $G \subseteq G^*$. □

- We investigate which graphs are “equivalent” regarding atoms and atom graphs. Recall that G^* denotes the graph obtained from G by saturating its atoms and is chordal (Property 3.4).

Property 4.6 *Let G and H be two connected graphs with the same vertices. The following propositions are equivalent.*
1) G and H have the same atom trees,
2) G and H have a common atom tree,
3) G and H have the same atom graph,
4) G and H have the same atoms,
5) $G^* = H^*$,
6) (a) every clique minimal separator of G is a clique minimal separator of H with the same components and neighborhoods, and (b) for every atom A of G, $H(A)$ has no clique separator,
7) G and H have the same clique minimal separators with the same components and neighborhoods.
If G and H satisfy these propositions, we say that they are atom-equivalent.

Proof: 1) ⇒ 2) follows from the fact that each connected graph has at least one atom tree.
2) ⇒ 4) and 3) ⇒ 4) are evident.
Let us show that 4) ⇒ 5). Every edge e of G is contained in an atom of G because at

each step of the decomposition represented by an atom tree of G, e is contained either in atom $A = C \cup S$ or in $G(V' \setminus C)$, where V' is the vertex set of the current subgraph of G. Hence G^* is obtained from the graph $(V, \emptyset)$ by saturating the atoms of G, and so is H^*.
5) $\Rightarrow$ 6) follows from Lemma 4.2 6) $\Rightarrow$ 7) is Lemma 4.4, and Lemma 4.5 is 7) $\Rightarrow$ (1) $\wedge$ 3)), which completes the proof of the equivalence of propositions 1) to 7). □

From 4.6, We can deduce that every graph H that is atom-equivalent to G is contained in G^* since $H \subseteq H^* = G^*$. However a graph G_1 such that $G \subseteq G_1 \subseteq G^*$ is not necessarily atom-equivalent to G, and also that two chordal atom-equivalent graphs G and H are necessarily equal, since in that case, by Characterization 3.3, $G = G^* = H^* = H$.

- [7] showed that some clique graphs are not chordal. This also holds for atom graphs, as shown by the counterexample below:

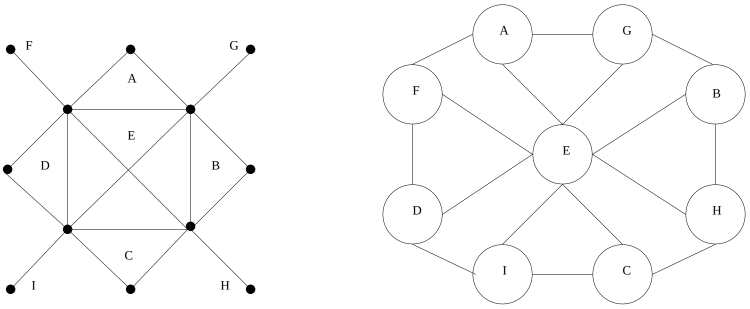

Figure 3: A graph G and its non-chordal atom graph.

We do not know how to characterize chordal atom graphs, but we have the following partial result.

Property 4.7 *Let G be a connected graph. If each component of each clique minimal separator of G is full then the atom graph of G is chordal.*

Proof: We assume for contradiction that each component of each clique minimal separator of G is full but that the atom graph of G is not chordal. Let $H = G^*$. By Lemma 4.2 each component of each clique minimal separator of H is full and the atom graph of H is not chordal. Let μ be a chordless cycle in the atom graph of H of length at least 4, let T be an atom tree of H, and let A be the vertex of μ (which is an atom of H) that comes first in T, i.e. such that the node of T labeled with A is at the shortest distance from the root of T. Let T' be the subtree of T rooted at the parent in T of the node labeled with A, let $H' = (V', E')$ be the subgraph of H such that T' is an atom tree of H', let S and T" such that T' is defined by $(A, S, T")$, let $C = A \setminus S$, and let $H" = H'(V' \setminus C)$. By Lemma 3.13 every clique minimal separator of H' is a clique minimal separator of H, so its components in H are full, and by

Lemma 3.13 again its components in H' are full too. Moreover by Lemma 3.15 μ is also a chordless cycle in the atom graph of H'.

Let B and B' be the neighbors of A in μ. By Property 3.14 there are components D and D' of S in H" such that $B \subseteq N_{H"}[D]$, $B' \subseteq N_{H"}[D']$, $N_{H"}(D) \subseteq B$ and $N_{H"}(D') \subseteq B'$. As all components of S in H', and therefore in H", are full, $N_{H"}(D) = N_{H"}(D') = S \subseteq B \cap B'$. It follows that $D = D'$, since otherwise $B \cap B'$ would be equal to S and would separate $B \setminus B'$ and $B' \setminus B$ in H', so that BB' would be a chord of μ in the atom graph of H'. As $B \subset B \cup B'$ and $B \cup B'$ is connected, $B \cup B'$ has a clique separator and therefore is not a clique of H'. Let a and b be two vertices of $B \cup B'$ that are not adjacent in H', and let R be a minimal ab-separator of H'. As B and B' are cliques of H', one of a and b is in $B \setminus B'$ and the other is in $B' \setminus B$, and $B \cap B' \subseteq R$. As $B \cap B'$ does not separate $B \setminus B'$ and $B' \setminus B$ in H', $B \cap B' \subset R$. As H' is chordal, R is a clique, so R is a clique minimal separator of H' with C as a non-full component, since $N_{H'}(C) = S \subseteq B \cap B' \subset R$, a contradiction. □

This is the case in particular for $K_{1,3} - free$ graphs.

5 Conclusion

We have presented a recursive algorithm to compute the atom graph at no extra cost than computing the atoms, as well as some properties of this atom graph.

In [7], it is shown that clique graph recognition is NP-complete. We leave open the question of deciding whether a graph is a clique graph or not.

References

[1] L. Alcón, L. Faria, C. M. H. De Figueiredo, and M. Gutierez. The complexity of clique graph recognition. *Theor. Comput. Sci.*, 410(21-23):2072–2083, 2009.

[2] A. Berry, J.-P. Bordat, P. Heggernes, G. Simonet, Y. Villanger Villanger. A wide-range algorithm for minimal triangulation from an arbitrary ordering. *J. Algorithms*, 58(1):33–66, 2006.

[3] A. Berry and R. Pogorelcnik. A simple algorithm to generate the minimal separators and the maximal cliques of a chordal graph. Technical report, 2010.

[4] A. Berry, R. Pogorelcnik, and G. Simonet. An introduction to clique minimal separator decomposition. Technical report, 2010.

[5] M. Didi Biha, B. Kaba, M.-J. Meurs, and E. SanJuan. Graph decomposition approaches for terminology graphs. In *MICAI*, pages 883–893, 2007.

[6] J. R. S. Blair, P. Heggernes, and J. A. Telle. A practical algorithm for making filled graphs minimal. *Theor. Comput. Sci.*, 250(1-2):125–141, 2001.

[7] J. R. S. Blair and B. W. Peyton. An introduction to chordal graphs and clique trees. WG 06, LNCS, vol. 4271, pp. 269-277.

[8] P. Heggernes, J. A. Telle, and Y. Villanger. Computing minimal triangulations in time $(n^{\alpha} logn) = n^{2.376}$.

[9] B. Kaba, N. Pinet, G. Lelandais, and A. Berry. Clustering gene expression data using graph separators. *In Silico Biology*, 7(0031), 2007.

[10] Dieter Kratsch and Jeremy Spinrad. Minimal fill in $o(n^{2.69})$ time. *Discrete Mathematics*, 306(3):366–371, 2006.

[11] P. Sreenivasa Kumar and C. E. Veni Madhavan. Minimal vertex separators of chordal graphs. *Discrete Applied Mathematics*, 89(1-3):155–168, 1998.

[12] H.-G. Leimer. Optimal decomposition by clique separators. *Discrete Mathematics*, 113(1-3):99–123, 1993.

[13] D. Meister. Recognition and computation of minimal triangulations for at-free claw-free and co-comparability graphs. *Discrete Applied Mathematics*, 146(3):193–218, 2005.

[14] A. Parra and P. Scheffler. Characterizations and algorithmic applications of chordal graph embeddings. *Discrete Applied Mathematics*, 79.

[15] D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM J. Comput.*, 5(2):266–283, 1976.

[16] R. E. Tarjan. Decomposition by clique separators. *Discrete Mathematics*, 55(2):221–232, 1985.