Finding cut-vertices in the square roots of a graph

Guillaume Ducoffe

Finding cut-vertices in the square roots of a graph *

Guillaume Ducoffe[1]

[1]Université Côte d'Azur, Inria, CNRS, I3S, France

Abstract

The square of a given graph $H = (V, E)$ is obtained from H by adding an edge between every two vertices at distance two in H. Given a graph class $\mathcal{H}$, the $\mathcal{H}$-Square Root problem asks for the recognition of the squares of graphs in $\mathcal{H}$. In this paper, we answer positively to an open question of [Golovach et al., IWOCA'16] by showing that the squares of *cactus block graphs* can be recognized in polynomial time. Our proof is based on new relationships between the decomposition of a graph by cut-vertices and the decomposition of its square by clique cutsets. More precisely, we prove that the closed neighbourhoods of cut-vertices in H induce maximal prime complete subgraphs of $G = H^2$. Furthermore, based on this relationship, we introduce a quite complete method in order to compute from a given graph G the block-cut tree of a desired square root (if any). Although the latter tree is not uniquely defined, we show surprisingly that it can only differ marginally between two different roots. Our approach not only gives the first polynomial-time algorithm for the $\mathcal{H}$-Square Root problem in different graph classes $\mathcal{H}$, but it also provides a unifying framework for the recognition of the squares of trees, block graphs and cactus graphs — among others.

1 Introduction

This paper deals with the well-known concepts of *square* and *square root* in graph theory. Roughly, the square of a given graph is obtained by adding an edge between the pairs of vertices at distance two (technical definitions are postponed to Section 2). While a square root of a given graph G has G as its square. The reason for this terminology is that when encoding a graph as an adjacency matrix A (with $1's$ on the diagonal), its square has for adjacency matrix A^2 –obtained from A using Boolean matrix multiplication.

The squares of graphs appear, somewhat naturally, in the study of coloring problems: when it comes about modelling interferences at a bounded distance in a radio network [46]. Unsurprisingly, there is an important literature on the topic, with nice structural properties of these graphs being undercovered [2, 7, 16, 28, 31, 34]. In particular, an elegant characterization of the squares of graphs has been given in [36]. However, this does not lead to an efficient (polynomial-time) algorithm for recognizing these graphs. Our main focus in the paper is on the existence of such algorithms. They are, in fact, unlikely to exist since the problem has been proved NP-complete [35]. In light of this negative result, there has been a growing literature trying to identify the cases where the recognition of the squares of graphs remains tractable [10, 21, 24, 23, 25, 30, 37]. We are interested in the variant where the desired square root (if any) must belong to some specified graph class.

*This work is partially supported by ANR project Stint under reference ANR-13-BS02-0007 and ANR program "Investments for the Future" under reference ANR-11-LABX-0031-01.

1.1 Related work

There is a complete dichotomy result for the problem when it is parameterized by the *girth* of the square root. More precisely, the squares of graphs with girth at least six can be recognized in polynomial-time, and it is NP-complete to decide whether a graph has a square root with girth at most five [1, 14, 15]. One first motivation for our work was to obtain similar dichotomy results based on the *separators* in the square root. We are thus more interested in graph classes with nice separability properties, such as chordal graphs. Recognizing the squares of chordal graphs is already NP-complete [24]. However, it can be done in polynomial-time for many subclasses [24, 25, 26, 33, 38, 42].

The most relevant examples to explain our approach are the classes of trees [42], block graphs [26] and cacti [19]. The squares of all these graphs can be recognized in polynomial-time. Perhaps surprisingly, whereas the case of trees is a well-known success story for which many algorithmic improvements have been proposed over the years [9, 26, 30, 42], the polynomial-time recognition of the squares of cactus graphs has been proved only very recently. A common point to these three above classes of graphs is that they can be decomposed into very simple subgraphs by using *cut-vertices* (respectively, in edges for trees, in cliques for block graphs and in cycles for cactus graphs). This fact is exploited in the polynomial-time recognition algorithms for the squares of these graphs. We observe that more generally, cut-vertices play a discrete, but important role, in the complexity of the recognition of squares, even for general graphs. As an example, most hardness results rely on a gadget called a "tail", that is a particular case of cut-vertices in the square roots [15, 35]. Interestingly, this tail construction imposes for some vertex in the square to be a cut-vertex with the same closed neighbourhood in any square root (see Figure 1). It is thus natural to ask whether more general considerations on the cut-vertices can help to derive additional constraints on the closed neighbourhoods in these roots. Our results prove that it is indeed the case.

(a) Tail in H.

(b) Tail in H^2.

Figure 1: Tail in a graph and its square.

As stated before, we are not the first to study the properties of cut-vertices in the square roots. In this respect, the work in [19] has been a major source of inspiration for this paper. However, most of the results so far obtained are specific to some graph classes and they hardly generalize to more general graphs [19, 26]. Evidence of this fact is that whereas both the squares of block graphs and the squares of cacti can be recognized in polynomial-time, the techniques involved in these two cases do not apply to the slightly more general class of *cactus-block graphs* (graphs that can be decomposed by cut-vertices into cycles and cliques) [19]. In the end, the characterization of the cut-vertices in these roots is only partial – even for cactus roots –, with most of the technical work for the recognition algorithm being rather focused on the notion of *tree decompositions* (*e.g.*, clique-trees for chordal squares, or decomposition of the square into bounded-treewidth graphs). Informally, tree decompositions [41] aim at decomposing graphs into pieces, called *bags*, organized in a tree-like manner. As proved in [18], the decomposition of the square root of a graph G by its cut-vertices leads to a specific type of tree decompositions for G that are called "H-tree decompositions". Note that it is not known whether a H-tree decomposition can be computed in polynomial-time. In contrast, we use in this work another type of tree decompositions, called an *atom tree*, that generalizes the notion of clique-trees for every graph. It can be computed in polynomial-time [5].

1.2 Our contributions

Our work is based on new relationships between the cut-vertices in a given graph and the *clique-cutsets* of its square (separators inducing a clique). These results are presented in Section 3. In particular, we obtain a complete characterization of the *atoms* of a graph (maximal subgraphs with no clique cutset) based on the *blocks* of its square roots (maximal subgraphs with no cut-vertices).

Having a thought on the problem, the existence of such relationships may look unsurprising, or even trivial. The most difficult part is to show how to "reverse" these relationships: from the square back to its square root. We prove in Section 4 that it can be done to some extent. More precisely, in Section 4.1 we show that the "essential" cut-vertices of the square roots: with at least two connected components not fully contained in their closed neighbourhoods, are in some sense unique (independent of the root) and that they can be computed in polynomial-time, along with their closed neighbourhood in any square root. Indeed, structural properties of these vertices allow to reinterpret them as the cut-vertices of some incidence graphs that can be locally constructed from the intersection of the atoms in an atom tree (tree decomposition whose bags are exactly the atoms). Proving a similar characterization for non essential cut-vertices remains to be done. We give sufficient conditions and a complete characterization of the closed neighbourhoods of these vertices for a large class of graphs in Section 4.2.

Then, inspired from these above results, we introduce a novel framework in Section 5 for the recognition of squares. Assuming a square root exists, we can push further some ideas of Section 4 in order to compute, for every block in this root, a graph that is isomorphic to its square. We thus reduce the recognition of the squares to a stronger variant of the problem for the squares of biconnected graphs. This is further discussed in Section 5.1. Let us point out that this approach can be particularly beneficial when the blocks of the roots are assumed to be part of a well-structured graph class.

In Section 6, we finally answer positively to an open question of [19] by proving that the squares of cactus-block graphs can be recognized in polynomial-time. Our result is actually much more general, as it gives a unifying algorithm for many graph classes already known to be tractable (*e.g.*, trees, block graphs and cacti) and it provides the first polynomial-time recognition algorithm for the squares of related graph classes – such as Gallai trees [17]. In its full generality, the result applies to "*j-cactus-block graphs*": a generalization of cactus-block graphs where each block is either a complete graph or the k^{th}-power of a cycle, for some $1 \leq k \leq j$ (*i.e.*, graphs obtained from cycles by adding an edge between every two distinct vertices at distance at most k). The j-cactus-block graphs form an increasing hierarchy that spans the whole class of graphs whose blocks are *cycle-power graphs*. We call them trees of cycle-powers in what follows. Note that cycle-power graphs are related to the class of Harary graphs and they have already received some attention in the literature [29]. Furthermore, as expected this last result is obtained by using our novel framework. This application is not straightforward. Indeed, we need to show the existence of a square root that is a tree of cycle-powers and has some "good" properties in order for the framework to be applied.

We also need to show that a stronger variant of the recognition of squares (discussed in Section 5.1) can be solved in polynomial-time for j-cactus-block graphs when j is a fixed constant[1]. We do so by introducing classical techniques from the study of circular-arc graphs [45].

Definitions and preliminary results are given in Section 2. We conclude this paper in Section 7 with some open questions.

2 Preliminaries

We use standard graph terminology from [8]. All graphs in this study are finite, unweighted and simple (hence with neither loops nor multiple edges), unless stated otherwise. Given a graph $G = (V, E)$ and a set $S \subseteq V$, we will denote by $G[S]$ the subgraph of G that is induced by S. The open neighbourhood of S, denoted by $N_G(S)$, is the set of all vertices in $G[V \setminus S]$ that are adjacent to at least one vertex in S. Similarly, the closed neighbourhood of S is denoted by $N_G[S] = N_G(S) \cup S$.

For every $u, v \in V$, vertex v is *dominated* by u if $N_G[v] \subseteq N_G[u]$. In particular, if $N_G[u] = N_G[v]$ then we say u and v are *true twins*. If even more strongly, we have $N_G[w] \subseteq N_G[u]$ for every $w \in N_G[v]$, then u is a *maximum neighbour* of v.

2.1 Squares and powers of graphs

For every connected graph G and for every $u, v \in V$, the distance between u and v in G, denoted by $dist_G(u, v)$, is equal to the minimum length (number of edges) of a uv-path in G. The j^{th}-power of G is the graph $G^j = (V, E_j)$ with same vertex-set as G and an edge between every two distinct vertices at distance at most j in G.

In particular, the *square* of a graph $G = (V, E)$ is the graph $G^2 = (V, E_2)$ with same vertex-set V as G and an edge between every two distinct vertices $u, v \in V$ such that $N_G[u] \cap N_G[v] \neq \emptyset$. Conversely, if there exists a graph H such that G is isomorphic to H^2 then H is called a *square root* of G. On the one hand it is easy to see that not all graphs have a square root. For example, if G is a tree with at least three vertices then it does not have any square root. On the other hand, note that a graph can have more than one square root. As an example, the complete graph K_n with n-vertices is the square of any diameter two n-vertex graph.

In what follows, we will focus on the following recognition problem:

Problem 1 ($\mathcal{H}$-SQUARE ROOT)**.**

Input: *A graph* $G = (V, E)$*.*

Question: *Is* G *the square of a graph in* $\mathcal{H}$ *?*

Our proofs will make use of the notions of subgraphs, induced subgraphs and *isometric subgraphs*, the latter denoting a subgraph H of a connected graph G such that $dist_H(x, y) = dist_G(x, y)$ for every $x, y \in V(H)$.

Furthermore, since we extensively use it in our proofs, we formalize the somewhat natural relationship between the walks in a square graph and walks in its square roots. More precisely:

[1] In a previous version of this manuscript, we presented a proof of this above result when j is arbitrary. Unfortunately, there was a flaw in a lemma which kills a central argument in the correctness analysis.

Definition 1. Let $G = (V, E)$ be a graph, H be a square root of G, and $\mathcal{W} = (x_0, x_1, \ldots, x_l)$ be a walk in G. An H-*extension* of $\mathcal{W}$ is any walk $\mathcal{W}'$ of H that is obtained from $\mathcal{W}$ by adding, for every i such that x_i and x_{i+1} are nonadjacent in H, a common neighbour $y_i \in N_H(x_i) \cap N_H(x_{i+1})$ between x_i and x_{i+1}.

2.2 Graph decompositions

A set $S \subseteq V$ is a *separator* in a graph $G = (V, E)$ if its removal increases the number of connected components. A *full component* in $G[V \setminus S]$ is any connected component C in $G[V \setminus S]$ satisfying that $N_G(C) = S$ (note that a full component might fail to exist). The set S is called a *minimal separator* in G if it is a separator and there are at least two full components in $G[V \setminus S]$.

Minimal separators are closely related to the notion of Robertson and Seymour's *tree decompositions* (*e.g.*, see [39]). Formally, a *tree-decomposition* $(T, \mathcal{X})$ of G is a pair consisting of a tree T and of a family $\mathcal{X} = (X_t)_{t \in V(T)}$ of subsets of V indexed by the nodes of T and satisfying:

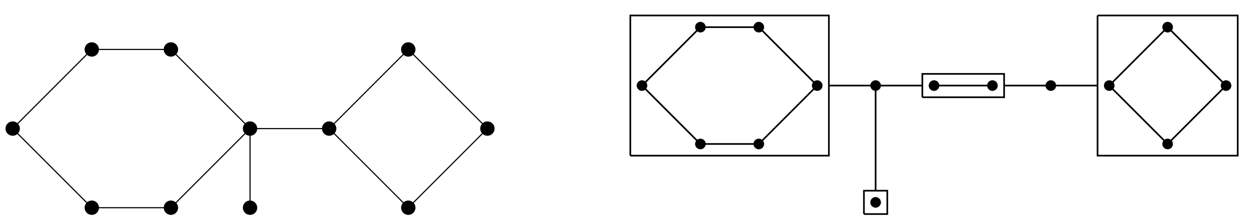
- $\bigcup_{t \in V(T)} X_t = V$;
- for any edge $e = \{u, v\} \in E$, there exists $t \in V(T)$ such that $u, v \in X_t$;
- for any $v \in V$, $\{t \in V(T) \mid v \in X_t\}$ induces a subtree, denoted by T_v, of T.

The sets X_t are called *the bags* of the decomposition.

In what follows, we will consider two main types of minimal separators.

Cut-vertices. If $S = \{v\}$ is a separator then it is a minimal one and we call it a *cut-vertex* of G. Following the terminology of [19], we name v an *essential* cut-vertex if there are at least two components C_1, C_2 of $G \setminus v$ such that $C_1 \not\subseteq N_G(v)$ and similarly $C_2 \not\subseteq N_G(v)$; otherwise, v is called a *non essential* cut-vertex[2].

Given a connected graph $G = (V, E)$, it is called *biconnected* if it does not have a cut-vertex. Examples of biconnected graphs are cycles and complete graphs. Furthermore, the *blocks* of G are the maximal biconnected subgraphs of G. It is well-known that the blocks and the cut-vertices of a connected graph G are the nodes of a tree, sometimes called the *block-cut tree*, that is obtained by adding an edge between every block B and every cut-vertex v such that $v \in B$ (see Figure 2 for an example). It can be computed in linear $\mathcal{O}(n + m)$-time [22].



(a) Graph G.

(b) Block-cut tree of G.

Figure 2: An example of block-cut-tree.

[2] The authors in [19] have rather focused on the stronger notion of *important* cut-vertices, that requires the existence of an additional third component C_3 of $G \setminus v$ such that $C_3 \not\subseteq N_G(v)$. We do not use this notion in our paper.

Observe that if G has a square root then it is biconnected. However, the following was proved in [18].

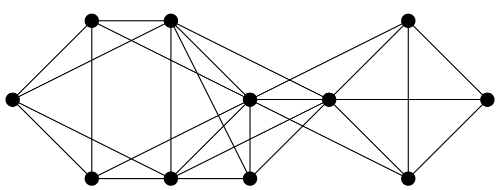
Lemma 2 ([18]). *Let H be a square root of a graph G. Let $\mathcal{C}$ and $\mathcal{B}$ be the sets of cut-vertices and blocks of H, respectively, and let T_H be the block-cut tree of H, with vertex set $V_T = \mathcal{C} \cup \mathcal{B}$. For every $u \in \mathcal{C}$, let $X_u = N_H[u]$ and for every $B \in \mathcal{B}$, let $X_B = V(B)$. Then, $(T_H, (X_t)_{t \in V_T})$ is a tree decomposition of G that is called the* H-tree decomposition *of G.*

Clique cutsets. More generally, if S is a minimal separator inducing a *clique* of $G = (V, E)$ then we call it a *clique cutset* of G. A connected graph $G = (V, E)$ is *prime* if it does not have a clique cutset. Cycles and complete graphs are again examples of prime graphs, and it can be observed more generally that every prime graph is biconnected.

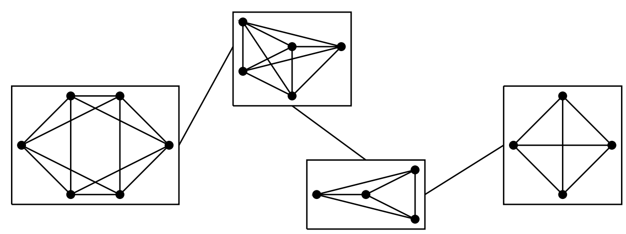
The *atoms* of G are the maximal prime subgraphs of G. They can be computed in polynomial-time [27, 44]. A *clique-atom* is an atom inducing a clique. Furthermore, a *simplicial vertex* is a vertex $v \in V$ such that $N_G[v]$ induces a clique. The following was proved in [13].

Lemma 3 ([13]). *Let $G = (V, E)$ be a graph and let $v \in V$. Then, v is simplicial if and only if $N_G[v]$ is a clique-atom, and it is the unique atom containing v.*

If the atoms of G are given, then the clique-atoms and the simplicial vertices of G can be computed in linear $\mathcal{O}(n + m)$-time [13].



(a) Graph G.



(b) Atom tree of G.

Figure 3: An example of atom tree.

Finally, it has been proved in [5] that the atoms of G are the bags of a tree decomposition of G, sometimes called an *atom tree*. We refer to Figure 3 for an illustration. An atom tree can be computed in $\mathcal{O}(nm)$-time, and it is not necessarily unique. In what follows, we often use in our analysis the following properties of an atom tree:

Lemma 4 ([6]). *Let $G = (V, E)$ be a graph and let A, A' be atoms of G. Then, $A \setminus A'$ is a connected subset and $A \cap A' \subseteq N_G(A \setminus A')$.*

Lemma 5 ([6]). *Let $G = (V, E)$ be a graph. For every atom tree $(T_G, \mathcal{A})$ of G, we have:*

$$\sum_{\{A, A'\} \in E(T_G)} |A \cap A'| = \mathcal{O}(n + m).$$

3 Basic properties of the atoms in a square

We start presenting relationships between the block-cut tree of a given graph and the decomposition of its square by clique cutsets (Theorem 8). These relationships are compared to some existing results in the literature for the $\mathcal{H}$-SQUARE ROOT problem.

More precisely, our approach in this paper is based on the following relationship between the clique cutsets in a graph G and the cut-vertices in its square-roots (if any).

Proposition 6. *Let $H = (V, E)$ be a graph. The closed neighbourhood of any cut-vertex in H is a clique-atom of $G = H^2$.*

Proof. Let $v \in V$ be a cut-vertex of H and let $A_v = N_H[v]$. It is clear that A_v is a clique of G and so, this set induces a prime subgraph of G. In particular, A_v must be contained in an atom A of G. Suppose for the sake of contradiction that $A \neq A_v$. Let $u \in A \setminus A_v$. This vertex u is contained in some connected component C_u of $H \setminus v$. Furthermore since v is a cut-vertex of H, there exists $w \in N_H(v) \setminus C_u$. We claim that $S = (C_u \cap N_H(v)) \cup \{v\}$ is an uw-clique separator of G. Indeed, let us consider any uw-path $\mathcal{P}$ in G. We name $\mathcal{Q} = (x_0 = u, x_1, \ldots, x_l = w)$ an arbitrary H-extension of $\mathcal{P}$. Since $\mathcal{Q}$ is an uw-walk in H, and u and w are in different connected components of $H \setminus v$, there exists an i such that $x_i \in C_u,\ x_{i+1} = v$. In particular, $x_i \in C_u \cap N_H(v) = S \setminus v$. Furthermore, by construction, for every two consecutive vertices x_i, x_{i+1} in the H-extension $\mathcal{Q}$, at least one of x_i or x_{i+1} belongs to $\mathcal{P}$. As a result, every uw-path in G intersects S, that proves the claim and so, that contradicts the fact that A is an atom of G. Therefore, $A = A_v$. Since A_v is a clique it is indeed a clique-atom of G. □

The above Proposition 6 unifies and generalizes some previous results that have been found only for specific graph classes [19, 26]. For example, it has been proved in [26] that for every block-graph H, the closed neighbourhoods of its cut-vertices are maximal cliques of its square. Our result shows that it holds for *any* square root H (not only block-graphs). Indeed, a clique-atom is always a maximal clique.

Furthermore by the previous Proposition 6, the closed neighbourhoods of cut-vertices in a given graph H are atoms of its square $G = H^2$. However, these may not be the only atoms of G. Our purpose with Theorem 8 is to give a partial characterization of the remaining atoms. Ideally, we would have liked them to correspond to the blocks of H. It turns out that this is not always the case. A counter-example is given in Figure 4. However, we prove in Lemma 7 the following weaker statement.

We first remind that given a graph $G = (V, E)$, $X \subseteq V$ is a *(monophonic) convex set* of G if, for every $x, y \in X$, every *induced* xy-path in G is contained in X (see [11]). For example, every atom is a monophonic convex set.

Lemma 7. *Let $G = (V, E)$ be a graph and let H be a square root of G. Every block of H is a monophonic convex set of G.*

Proof. Let B be a block of H, let $u, w \in B$ and let $\mathcal{P}$ be an induced uw-path in G. Since $\mathcal{P}$ is induced, any of its H-extensions $\mathcal{Q}$ must be a path (*i.e.*, with no repeated vertices). In particular, we claim that it implies $V(\mathcal{Q}) \subseteq B$. Indeed if it were not the case then $\mathcal{Q}$ should pass twice by a same cut-vertex v of H (in order to leave then to go back in B), thereby contradicting the fact that it is a path. As a result, $V(\mathcal{P}) \subseteq V(\mathcal{Q}) \subseteq B$, and so, B is a monophonic convex set of G. □

(a) Graph H. (b) Square H^2. Vertices x, y are simplicial in it.

Figure 4: A biconnected graph with a non prime square.

We finally present the main result in this section.

Theorem 8. *Let H be the square root of a given graph $G = (V, E)$. Then, the atoms of G are exactly:*

- *the cliques $A_v = N_H[v]$, for every cut-vertex v of H;*
- *and for every block B of H, the atoms A' of $H[B]^2$ that are not dominated in H by a cut-vertex.*

Proof. Let A be an atom of G. There are two cases to be distinguished.

1. Suppose that A is not contained in a block of H. Let B_1, B_2 be two different blocks of H intersecting A and let $u \in B_1 \setminus B_2,\ w \in B_2 \setminus B_1$ such that $u, w \in A$. By the hypothesis, there exists a cut-vertex v of H that intersect every uw-path of H. We claim that $A = A_v$. Indeed, suppose for the sake of contradiction that $A \neq A_v$. W.l.o.g., $u \notin A_v$. Let us name by C_u the connected component of $H \setminus v$ that contains u. Then, it is easy to prove (as we did for Proposition 6) that every uw-path in G must intersect the clique $S = (N_H(v) \cap C_u) \cup \{v\}$. The latter contradicts our assumption that the atom A has no clique cutset. Therefore, we obtain that $A \subseteq A_v$. Since A_v is an atom by Proposition 6, it follows as claimed that $A = A_v$.

2. Else, $A \subseteq B$, for some block B of H. Observe that B induces an isometric subgraph of H, hence $H[B]^2$ and $G[V(B)]$ are isomorphic. As a result, A is an atom of G implies that A is also an atom of $H[B]^2$. Conversely, if A' is an atom of $H[B]^2$ then A' induces a prime subgraph of G. In particular, it is an atom of G if and only if it is inclusion wise maximal w.r.t. this property. Finally, we note that the only atoms that can possibly contain A' are the sets A_v, for any cut-vertex v of H that is contained in B. Altogether combined, we have in this situation that A is an atom of $H[B]^2$ such that, for every cut-vertex v of H that is contained in B, $A' \nsubseteq A_v$.

□

From Theorem 8, we deduce that the atom trees of a given square graph G are related to its *H-tree decompositions*, as introduced in Lemma 2. In fact, for any square root H of G, the atom trees of G are refinements of its H-tree decomposition, in the sense that every atom is included in a bag of the H-tree decomposition of G.

Furthermore, the atoms of G coincide with the bags of its H-tree decomposition if and only if: every block of H has a prime square, and there is no block of H included in the closed neighbourhood of a cut-vertex. In this situation, the H-tree decomposition of G is an atom tree. However, it may be the case that G has other atom trees. We refer to [43] for counting the number of atom trees for a given graph.

4 Computation of the cut-vertices from the square

Given a square graph $G = (V, E)$, we aim at computing all the cut-vertices in some square root H of G. More precisely, given two square roots H_1 and H_2 of G, we say that H_1 is "finer" than H_2, denoted by $H_1 \preceq H_2$, if every block of H_1 is contained in a block of H_2. The latter defines a partial ordering over the square roots of G, of which we call *maxblock square roots* its minimal elements. This notion is related to, but different than, the notion of minimal square root studied in [19].

Let $H_{\max}$ be a maxblock square root of G. The following section is based on Proposition 6, that gives a necessary condition for a vertex to be a cut-vertex in $H_{\max}$. Indeed, it follows from this Proposition 6 that there is a mapping from the cut-vertices of $H_{\max}$ to the clique-atoms of its square $G = H_{\max}^2$. We can observe that the mapping is injective: indeed, it follows from the existence of the block-cut tree that every two distinct cut-vertices v, v' of $H_{\max}$ can be contained in at most one common block; since every cut-vertex is contained in at least two blocks, we get $N_{H_{\max}}[v] \neq N_{H_{\max}}[v']$. However, the mapping is not surjective in general.

In what follows, we present sufficient conditions for a clique-atom of G to be the closed neighbourhood of a cut-vertex in *any* maxblock square root of G. In particular, we obtain a complete characterization for the essential cut-vertices.

4.1 Recognition of the essential cut-vertices

We recall that a cut-vertex v of $H_{\max}$ is called essential if there are two vertices in different connected components of $H_{\max} \setminus v$ that are both at distance two from v in $H_{\max}$. The remaining of the section is devoted to prove the following result.

Theorem 9. *Let $G = (V, E)$ be a square graph. Every maxblock square root of G has the same set $\mathcal{C}$ of essential cut-vertices. Furthermore, every vertex $v \in \mathcal{C}$ has the same neighbourhood A_v in any maxblock square root of G. All the vertices $v \in \mathcal{C}$ and their neighbourhood A_v can be computed in $\mathcal{O}(n + m)$-time if an atom tree of G is given.*

Algorithm 1 Computation of the essential cut-vertices.

Require: A graph $G = (V, E)$; an atom tree $(T_G, \mathcal{A})$ of G.
Ensure: Returns (if G is a square) the set $\mathcal{C}$ of essential cut-vertices, and for every $v \in \mathcal{C}$ its neighbourhood A_v, in any maxblock square root of G.
1: $\mathcal{C} \leftarrow \emptyset$.
2: **for all** clique-atom $A \in \mathcal{A}$ **do**
3: Compute the incidence graph $I_A = Inc(\Omega(A), A)$, with $\Omega(A)$ being the multiset of neighbourhoods of the connected components of $G \setminus A$.
4: **if** $\bigcap_{S \in \Omega(A)} S = \{v\}$ **and** v is a cut-vertex of I_A **then**
5: $\mathcal{C} \leftarrow \mathcal{C} \cup \{v\}$; $A_v \leftarrow A$.

The proof of Theorem 9 mainly follows from the correctness proof and the complexity analysis of Algorithm 1. Its basic idea is that the essential cut-vertices in any maxblock square root of G are exactly the cut-vertices in some "incidence graphs", that are locally constructed from the neighbourhoods of each clique-atom in the atom tree.

Formally, for every clique-atom A of G, let $\Omega(A)$ contain $N_G(C)$ for every connected component C of $G \backslash A$ (note that $\Omega(A)$ is a multiset, with its cardinality being equal to the number of connected components in $G \backslash A$). The incidence graph $I_A = Inc(\Omega(A), A)$ is the bipartite graph with respective sides $\Omega(A)$ and A and an edge between every $S \in \Omega(A)$ and every $u \in S$. Note that I_A may contain isolated vertices (*i.e.*, simplicial vertices in A that are not contained in the neighbourhood of any component), and so, disconnected. We first make the following useful observation:

Fact 10. *For every $v \in A$, v is a cut-vertex of I_A if and only if there is a bipartition P, Q of the connected components of $G \setminus A$ such that $N_G(P) \cap N_G(Q) = \{v\}$.*

We now subdivide the correctness proof of Algorithm 1 in the two following lemmas.

Lemma 11. *Let H be the square root of a given graph $G = (V, E)$, let $v \in V$ be an essential cut-vertex of H and let $A_v = N_H[v]$. Then, v has a neighbour in G in every connected component of $G \setminus A_v$. Furthermore, there is a bipartition P, Q of the connected components of $G \setminus A_v$ such that $N_G(P) \cap N_G(Q) = \{v\}$.*

Proof. We prove the two statements of the lemma separately. First, observe that for every connected component D of $G \backslash A_v$, we have that $N_H(D) \cap A_v \neq \emptyset$. Since $A_v = N_H[v]$, it follows that $v \in N_G(D)$. In particular, v has a neighbour in G in every connected component of $G \setminus A_v$.

Second, let $C_1, C_2, \ldots, C_k$ be all the connected components of $H \backslash v$ such that $C_i \not\subseteq A_v$. Note that $k \geq 2$ by the hypothesis. Furthermore, since for every $i \neq j$ and for every $u_i \in C_i \setminus A_v$, $u_j \in C_j \setminus A_v$, we have $dist_H(u_i, u_j) = dist_H(u_i, v) + dist_H(u_j, v) \geq 4$, there can be no edge between $C_i \setminus A_v$ and $C_j \setminus A_v$ in G. It implies that for every component D of $G \setminus A_v$, there is an $1 \leq i \leq k$ such that $D \subseteq C_i \setminus A_v$. So, let us group the components of $G \setminus A_v$ in order to obtain the sets $C_i \setminus A_v$, $1 \leq i \leq k$. For every $1 \leq i \leq k$, we have $\{v\} \subseteq N_G(C_i \setminus A_v) \subseteq (N_H(v) \cap C_i) \cup \{v\}$. In particular, for every $i \neq j$, we obtain $N_G(C_i \setminus A_v) \cap N_G(C_j \setminus A_v) = \{v\}$. Hence, let us bipartition the sets $C_i \setminus A_v$ into two nonempty supersets P and Q; by construction we have $N_G(P) \cap N_G(Q) = \{v\}$. □

It turns out that conversely, Lemma 11 also provides a sufficient condition for a vertex v to be an essential cut-vertex in some square root of G (and in particular, in any maxblock square root). We formalize this next.

Lemma 12. *Let $H_{\max}$ be a maxblock square root of a given graph $G = (V, E)$, and let $v \in V$. Suppose there is a clique-atom A_v of G and a bipartition P, Q of the connected components of $G \setminus A_v$ such that $N_G(P) \cap N_G(Q) = \{v\}$. Then, for every square root H of G, we have $N_H(P) \cup N_H(Q) \subseteq N_H(v) \subseteq A_v$. In particular, v is an essential cut-vertex of $H_{\max}$ and $N_{H_{\max}}[v] = A_v$.*

Proof. Let H be any square root of G. By the hypothesis, there can be no edge in G between P and Q, and so, $N_H(P) \cap N_H(Q) = \emptyset$. Furthermore, if it were the case that $u \in N_H(P)$ and $w \in N_H(Q)$ are adjacent in H, then it would imply that $u, w \in N_G(P) \cap N_G(Q)$, that would contradict the hypothesis that $N_G(P) \cap N_G(Q) = \{v\}$. As a result, since A is a clique of G we have that for every $u \in N_H(P)$ and $w \in N_H(Q)$, there exists a common neighbour $v' \in N_H(u) \cap N_H(w)$, and necessarily $v' \notin N_H(P) \cup N_H(Q)$ (otherwise it would imply the existence of an edge in H between

$N_H(P)$ and $N_H(Q)$). Since in addition $v' \in N_G(P) \cap N_G(Q)$ then it must be the case that $v' = v$. Altogether combined, it implies that $N_H(P) \cup N_H(Q) \subseteq N_H(v) \subseteq A_v$.

Then, suppose for the sake of contradiction that v is not a cut-vertex of $H_{\max}$. We claim that A_v must be contained in a block of $H_{\max}$. Indeed, observe that since v is the unique vertex such that $N_{H_{\max}}(P) \cup N_{H_{\max}}(Q) \subseteq N_{H_{\max}}(v)$, there can be no cut-vertex of $H_{\max}$ whose closed neighbourhood is A_v. In this situation, by Theorem 8 A_v is an atom in the square of some block B of $H_{\max}$. It implies as claimed that A_v must be contained in B.

Let H_0 be obtained from $H_{\max}$ as follows. We first remove all the edges in $H_{\max}$ between every vertex of $N_G(P) \setminus N_{H_{\max}}(P)$ and every vertex of $N_G(Q) \setminus N_{H_{\max}}(Q)$. Note that this operation does not change the neighbourhood at distance two of P and Q. Then, we make every vertex of A_v adjacent to v. By doing so, we make of vertex v a cut-vertex in H_0, and we so increase the number of blocks. Furthermore, A_v is still a clique of H_0^2, and since $N_{H_{\max}}(P) \cup N_{H_{\max}}(Q) \subseteq N_{H_{\max}}(v)$, the adjacency relations between the vertices of $P \cup Q$ and the vertices of A_v are the same in H_0^2 as in G. Hence, by construction H_0 is a square root of G which is finer than $H_{\max}$, that contradicts the fact that $H_{\max}$ is a maxblock square root.

Therefore, v is a cut-vertex of $H_{\max}$. By using the same arguments as above, it can also be proved that the vertices in P are in different connected components of $H_{\max} \setminus v$ than the vertices in Q. Hence, v is an essential cut-vertex of $H_{\max}$. Finally, since $N_{H_{\max}}[v] \subseteq A_v$ and $N_{H_{\max}}[v]$ is a clique-atom of G by Proposition 6, we have $N_{H_{\max}}[v] = A_v$. □

Proof of Theorem 9. The unicity of $\mathcal{C}$ and of the sets $A_v,\ v \in \mathcal{C}$ follows from Lemmas 11 and 12. Indeed, on the one hand let $v \in V$ be an essential cut-vertex in some square root H of G and let $A_v = N_H[v]$. Let $I_{A_v} = Inc(\Omega(A_v), A_v)$ be the incidence graph as defined above. By Lemma 11 we have $v \in \bigcap_{S \in \Omega(A)} S$. Furthermore, since v is a cut-vertex of I_A, it is the unique vertex in this common intersection $\bigcap_{S \in \Omega(A)} S$. As a result, vertex v passes the test of Algorithm 1 for $A = A_v$. Conversely, if for some clique-atom A the (unique) vertex v tested passes the test of Algorithm 1, then by Lemma 12 v is an essential cut-vertex with closed neighbourhood being equal to A in any maxblock square root of G.

Let us finally prove that given an atom tree $(T_G, \mathcal{A})$ of G, the set $\mathcal{C}$ and the closed neighbourhood $A_v,\ v \in \mathcal{C}$ can be computed in linear time. In order to prove it, it suffices to prove that Algorithm 1 can be implemented to run in linear time. We first remind that the set of clique-atoms of G can be computed from $(T_G, \mathcal{A})$ in linear time [13].

Furthermore, let $A \in \mathcal{A}$ be a fixed clique-atom, and let $\Omega^*(A) = \{A' \cap A \mid \{A', A\} \in E(T_G)\}$. By the properties of atom trees (Lemma 4), $\Omega^*(A)$ is a subset of $\Omega(A)$. In addition (by the properties of tree decompositions) we have that every set $S \in \Omega(A)$ is included in some set $S^* \in \Omega^*(A)$. So, instead of computing the incidence graph I_A, let us compute the "reduced incidence graph" $I_A^* = Inc(\Omega^*(A), A)$. Let us replace I_A by I_A^* for the test of Algorithm 1. By doing so, this test can be performed in $\mathcal{O}(|E(I_A^*)|) = \mathcal{O}(\sum_{S^* \in \Omega^*(A)} |S^*|)$.

Let us prove that this modification of the test does not change the output of the algorithm. On the one hand, if $v \in A$ passes the modified test then by Lemma 12, $v \in \mathcal{C}$ and $A_v = A$. Conversely, let $v \in \mathcal{C}$ be arbitrary and let $A = A_v$. Note that since by Lemma 11 we have $v \in \bigcap_{S \in \Omega(A)} S$, we also have $v \in \bigcap_{S^* \in \Omega^*(A)} S^*$. In addition, since G is assumed to be a square, it is biconnected and so,

$|S| \geq 2$ for every $S \in \Omega(A)$; in particular, every two $S, S^* \in \Omega(A)$ such that $S \subseteq S^*$ are in the same block of I_A (since they have two common neighbours in I_A), that implies that v keeps the property to be a cut-vertex of I_A^*. The latter implies as before that $\bigcap_{S^* \in \Omega^*(A)} S^* = \{v\}$. It thus follows that every vertex $v \in \mathcal{C}$ passes the modified test.

Overall, this above implementation of Algorithm 1 runs in time:

$$\mathcal{O}(\sum_{A \in \mathcal{A}} \sum_{S^* \in \Omega^*(A)} |S^*|) = \mathcal{O}(\sum_{\{A, A'\} \in E(T_G)} |A \cap A'|).$$

Since $(T, \mathcal{A})$ is an atom tree, it is $\mathcal{O}(n+m)$ by Lemma 5. □

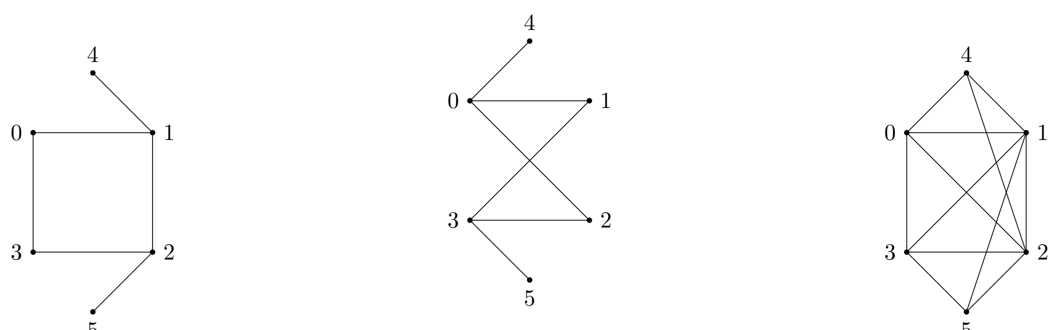
4.2 Sufficient conditions for non essential cut-vertices

We let open whether a good characterization of non essential cut vertices can be found. The remaining of this section is devoted to partial results in this direction.

It can be observed that in general, not all the maxblock square roots of a graph have the same set of non essential cut-vertices. This is due to the fact that non essential cut-vertices can have a true twin in the square graph (*e.g.*, see Figure 5). More precisely, the following observation can be of independent interest:

Fact 13. *Let H be the square root of a graph $G = (V, E)$ and let $u, v \in V$ be true twins in G. Then, $H_{u \leftrightarrow v}$: obtained from H by exchanging the neighbours of u and v, is also a square root of G* [3].

Our main result in this section is a complete characterization of the closed neighbourhoods of non essential cut-vertices in any maxblock square root — under additional assumptions on the properties of its blocks (Theorem 17). Admittedly, these additional conditions are a bit technical. However, we show in the next sections that they are satisfied by many interesting graph classes.



(a) Square root H_1. (b) Square root H_2 (isomorphic to H_1). (c) Square G.

Figure 5: Two square roots where the cut-vertices are different.

Observe that if v is a non essential cut-vertex in some square root H of a graph G, there is at most one component of $H \setminus v$ that is not fully contained in $N_H[v]$. Thus, we can make the following easy observation:

[3]Of course, if u and v are (non)adjacent in H then they remain so in $H_{u \leftrightarrow v}$. In fact, this is just a relabeling of u and v by v and u, respectively.

Fact 14. *Let $H_{\max}$ be a maxblock square root of a graph $G = (V, E)$ and let $v \in V$ be a non essential cut-vertex of $H_{\max}$. All but at most one components of $H_{\max} \setminus v$ are reduced to an edge between v and a pending vertex.*

Non essential cut-vertices in a maxblock square root are strongly related to simplicial vertices in the square. Indeed, let $H_{\max}$ be a maxblock square root of a graph $G = (V, E)$. Let v be a non essential cut-vertex of $H_{\max}$ and u be a pending vertex adjacent to v in $H_{\max}$. Then, u becomes a simplicial vertex of G such that $N_G[u] = N_H[v]$.

In general, if a clique-atom of G contains a simplicial vertex then it may not necessarily represent the closed neighbourhood of a cut-vertex in $H_{\max}$. However, we are able to prove the following weaker statement:

Lemma 15. *Let $G = (V, E)$ be a graph, and let A be a clique-atom of G that contains a nonempty set K of simplicial vertices. If G is a square then it has a square root H such that $H \setminus K$ is a monophonic convex subgraph of H.*

Proof. Let H' be any square root of G. Let $S_1 = N_{H'}(V \setminus A)$, $S_2 = N_{H'}(S_1) \cap A$. Note that $N_{H'}(K) \subseteq S_2$. Furthermore, let H be obtained from H' by adding an edge between every two nonadjacent vertices of S_2. By construction, H is still a square root of G, and since S_2 is a clique we have that $H \setminus K$ is a monophonic convex subgraph of H. $\square$

The above construction of Lemma 15 could be refined if we were able to decide when a vertex of S_2 can be made adjacent to every vertex of S_1. The following are, so far, the most general cases where we are able to do so. We first recall that a vertex is called *simple* if it is simplicial and the closed neighbourhoods of its neighbours can be linearly ordered by inclusion.

Lemma 16. *Let $H_{\max}$ be a maxblock square root of a graph $G = (V, E)$. If there exists a simple vertex u in G then it has a neighbour $v \in N_G(u)$ that is a non essential cut-vertex of $H_{\max}$. Furthermore, $N_{H_{\max}}[v] = N_G[u]$.*

Proof. Let $A = N_G[u]$. Since u is simple, by Lemma 3 A is the unique atom of G containing u. Furthermore, we claim that A is the closed neighbourhood of a cut-vertex in $H_{\max}$. By contradiction, suppose that it is not the case. By Theorem 8, A is an atom in the square of some block of $H_{\max}$. Hence, A is contained in a block of $H_{\max}$. We will transform $H_{\max}$ so that it is no longer the case, that will arise a contradiction. In order to do so, let $S = S_1 \cup S_2$ with $S_1 = N_{H_{\max}}(V \setminus A)$, $S_2 = N_{H_{\max}}(S_1) \cap A$ and let $K = A \setminus S$. Observe that $K \neq \emptyset$ (since $u \in K$). Thus, since $N_{H_{\max}}(K) \subseteq S_2$ and A is a clique-atom of G, we have that $N_{H_{\max}}(w) \cap S_2 \neq \emptyset$ for every $w \in S_1$. In particular, $\bigcup_{v' \in S_2} N_G(v') \setminus A = \bigcup_{w \in S_1} N_{H_{\max}}(w) \setminus A = N_{H_{\max}}(A)$.

Let $v \in S_2$ maximize her degree in G. Note that since u is assumed to be simple by the hypothesis, $N_G[v'] \subseteq N_G[v]$ for every $v' \in S_2$. So, let H_0 be obtained from $H_{\max}$ by removing all the edges incident to a vertex in K, then making vertex v universal in A. By construction, all the vertices in K are now pending vertices adjacent to v. Furthermore, H_0 has strictly more cut-vertices and strictly more blocks than $H_{\max}$. We claim that H_0 is a square root of G, that will arise a contradiction. Indeed, on the one hand A is a clique of H_0^2. In addition, for every $v' \in S \setminus v$ we have $N_{H_0^2}(v') \setminus A = N_G(v') \setminus A$ because $N_{H_{\max}}[v'] \cap S_1 = N_{H_0}[v'] \cap S_1$. On the other hand, $N_{H_0^2}(v) \setminus A = N_{H_{\max}}(S_1) \setminus A = \left(\bigcup_{v' \in S_2} N_G(v')\right) \setminus A = N_G(v) \setminus A$. As a result, H_0 is a square root of G which is finer than $H_{\max}$, thereby contradicting the fact that $H_{\max}$ is a maxblock square root of G. This implies as claimed that A is the closed neighbourhood of a cut-vertex v of $H_{\max}$.

In order to complete the proof of the lemma, it suffices to prove that v cannot be an essential cut-vertex. Indeed, let us totally order the vertices $w_1, w_2, \ldots, w_k \in S$ such that $i < j \implies N_G[w_i] \subseteq N_G[w_j]$ (it can be done since u is simple). In this situation, if for some component C of $G \setminus A$, we have $w_i \in N_G(C)$, then we also have $w_i, w_{i+1}, \ldots, w_k \in N_G(C)$. In particular, since G is biconnected, it implies that w_{k-1}, w_k have a neighbour in every component of $G \setminus A$. As a result, there can be no partition P, Q of the connected components of $G \setminus A$ such that $N_G(P) \cap N_G(Q)$ is reduced to a singleton. It thus follows from Lemma 11 that A cannot be the closed neighbourhood of an essential cut-vertex. $\square$

A main drawback of Lemma 16 is that it does not say how to compute the desired cut-vertex v. We show how to overcome this difficulty in the next sections. Before concluding this section, we now state its main result.

Theorem 17. *Let $G = (V, E)$ be a connected graph that is not a clique, and let $H_{\max}$ be a* finest *square root of G such that, for every block B of $H_{\max}$, we have:*

- *$H_{\max}[B]$ has no dominated vertex, unless it is a clique;*
- *and it has a prime square.*

Then a clique-atom A of G is the closed neighbourhood of a non essential cut-vertex in $H_{\max}$ if and only if *it satisfies the following condition: there exists a clique cutset $S \subseteq A$ such that, for every component C of $G \setminus A$, we have $N_G(C) \subseteq S$*[4]*. Furthermore, all the clique-atoms satisfying this above property can be computed in $\mathcal{O}(n + m)$-time if an atom tree of G is given.*

The first assumption on the blocks may look a bit artificial. However, we emphasize that it holds for every regular graph [3]. Furthermore, we note that the condition of Theorem 17 is related to the notion of simplicial moplex (see [4]). In fact, the atoms satisfying the condition of the theorem are exactly the closed neighbourhoods of simplicial moplex.

Proof. Let $(T_G, \mathcal{A})$ be an atom tree of G. Let $A \in \mathcal{A}$ be an arbitrary clique-atom.

Suppose that $v \in A$ is a non essential cut-vertex of $H_{\max}$ such that $N_{H_{\max}}[v] = A$. Observe that since G is not a clique, we have $V \neq A$. Furthermore, since v is non essential, there is a unique connected component D of $H_{\max} \setminus v$ such that $D \not\subseteq A$. So, let $S = (N_{H_{\max}}(v) \cap D) \cup \{v\}$. By construction, S induces a clique and we have for every component C of $G \setminus A$, $N_G(C) \subseteq S$. We claim that S is a clique-cutset of G. Indeed, we first need to observe that for every $u, w \in S$ non-adjacent, there exist two uw-paths in $H_{\max}$ that are internally vertex-disjoint (namely, (u, v, w) and any uw-path in D). Hence, there exists a block B of $H_{\max}$ such that $S \subseteq B$.

We prove as a subclaim that $B \subseteq A'$ where $A' \neq A$ is an atom of G. There are two subcases to be distinguished.

- Suppose that B is a clique of $H_{\max}$. In this situation, $B = S$. Since $B \subseteq D \not\subseteq A$, there exists a cut-vertex v' of $H_{\max}$ such that $v' \in D$ and $B \subseteq A_{v'}$. So, we can choose $A' = A_{v'}$.
- Otherwise, B is not a clique of $H_{\max}$. Let $H_B = H_{\max}[B]$. By the hypothesis, H_B has no dominated vertex. Hence, B is an atom of G by Theorem 8, and so, we can choose $A' = B$.

[4]Since clique-cutsets are in bijective correspondance with the edges in an atom tree, this property is equivalent to have A being a leaf in some atom tree.

The latter proves the subclaim. Then, by Lemma 4, there exists a connected component C_S of $G \setminus A$ such that $A' \setminus S \subseteq C_S$ and $S \subseteq N_G(A' \setminus S) \subseteq N_G(C_S)$. It implies $N_G(C_S) = S$. Overall, this proves as claimed that S is a clique-cutset of G, and so, the clique-atom A indeed satisfies the condition of the theorem.

Conversely, suppose that A satisfies this condition, and let us prove that it is the closed neighbourhood of a non essential cut-vertex of $H_{\max}$. Let $S \subseteq A$ be the clique cutset as defined by the condition of the theorem.

We first prove as an intermediary claim that A cannot be the closed neighbourhood of an essential cut-vertex. Indeed, since S is a clique cutset, and so, a minimal separator, there are at least two full components of $G \setminus S$. The latter implies the existence of a connected component C_S of $G \setminus A$ such that $N_G(C_S) = S$. However in this situation, let P, Q be any bipartition of the connected components of $G \setminus A$ such that $C_S \in P$. Since A satisfies the condition of the theorem, we have $N_G(Q) \subseteq N_G(P) = S$. Hence $N_G(P) \cap N_G(Q) = N_G(Q)$ cannot be reduced to a singleton because G is biconnected. By Lemma 11, this implies as claimed that A cannot be the closed neighbourhood of an essential cut-vertex.

Finally, suppose for the sake of contradiction that A is not the closed neighbourhood of a cut-vertex of $H_{\max}$. By Theorem 8, A is an atom in the square of some block of $H_{\max}$. By the hypothesis, it implies that A coincides with a block of $H_{\max}$, and this block cannot be a clique (otherwise, A would be contained in the closed neighbourhood of a cut-vertex in $H_{\max}$). Therefore, $H_A = H[A]$ has no dominated vertex. We prove as a new intermediary claim that A contains a unique cut-vertex of $H_{\max}$. Indeed, the set of neighbourhoods of the connected components of $G \setminus A$ is exactly the set of intersections $A_v \cap A = N_{H_A}[v],\ A_v = N_{H_{\max}}[v]$, for every cut-vertex v of $H_{\max}$ contained in A. In particular, $A_v \cap A = N_{H_A}[v] \subseteq S$. Furthermore, recall that there exists a connected component C_S of $G \setminus A$ such that $N_G(C_S) = S$. Hence, there exists $v_0 \in C_S$ such that v_0 is a cut-vertex of $H_{\max}$ and $A_{v_0} \cap A = N_{H_A}[v_0] = S$. Overall, since H_A has no dominated vertex, it implies as claimed that v_0 is the unique cut-vertex of $H_{\max}$ contained in A.

However, in this situation, let $v \in N_{H_A}(v_0)$ be arbitrary. Let H_0 be obtained from $H_{\max}$ by transforming S into a clique and making every vertex of $A \setminus S$ a pending vertex adjacent to v. By construction, H_0 is a square root of G which is finer than $H_{\max}$. Since the blocks of H_0 still satisfy the two assumptions of the theorem, it contradicts the minimality of $H_{\max}$. As a result, we obtain that A is the closed neighbourhood of some cut-vertex of $H_{\max}$ (necessarily, non essential).

In order to complete the proof, we are left to prove the linear time bound in order to compute all the clique-atoms that satisfy the desired property. We recall that given $(T_G, \mathcal{A})$, all the clique-atoms can be computed in $\mathcal{O}(n+m)$-time [13]. Furthermore, for every clique-atom A, let $\Omega^*(A) = \{A' \cap A \mid \{A', A\} \in E(T_G)\}$ and let $S_A \in \Omega^*(A)$ be of maximum size. By the combination of Lemma 4 with usual properties of tree decompositions, we have that if A satisfies the condition of the theorem then it does so with $S = S_A$. It can be checked in time $\mathcal{O}(\sum_{S^* \in \Omega^*(A)} |S^*|)$. Overall, all the clique-atoms that satisfy the condition of the theorem can be retrieved in time:

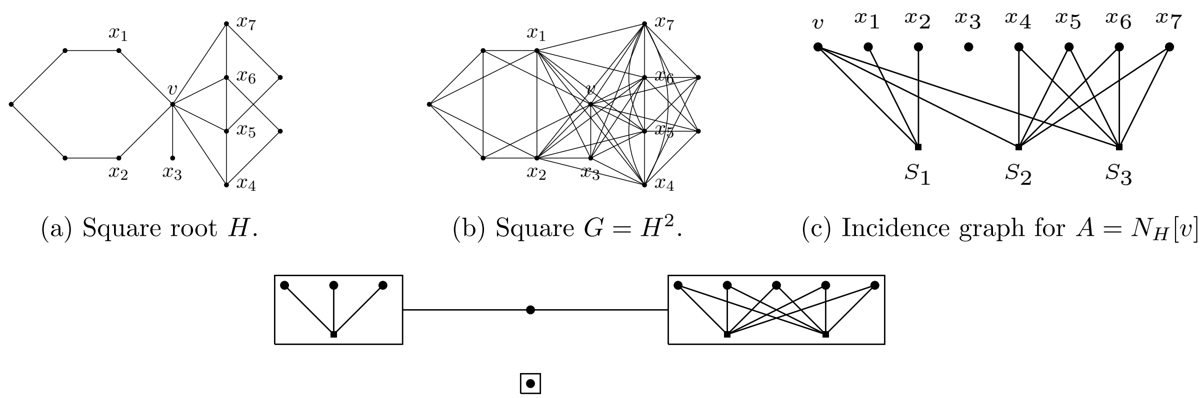
$$\mathcal{O}(\sum_{A \in \mathcal{A}} \sum_{S^* \in \Omega^*(A)} |S^*|) = \mathcal{O}(\sum_{\{A, A'\} \in E(T_G)} |A \cap A'|),$$

that is $\mathcal{O}(n+m)$ by Lemma 5. $\square$

5 Reconstructing the block-cut tree of a square root

Given a graph $G = (V, E)$, we propose a generic approach in order to compute the block-cut tree of one of its square-roots (if any). More precisely, we remind that a square root $H_{\max}$ of G is called a *maxblock square root* if there does not exist any other square root $H \neq H_{\max}$ of G with all its blocks being contained in the blocks of $H_{\max}$. We suppose we are given the closed neighbourhoods of all the cut-vertices in some maxblock square root $H_{\max}$ of G (the cut-vertices may not be part of the input). Based on this information, we show how to compute for every block of $H_{\max}$ a graph that is isomorphic to its square (Theorem 18). However, in doing so, the correspondance between the nodes in these graphs and the nodes in G is partly lost. Hence for each block, we need to solve a stronger version of the $\mathcal{H}$-SQUARE ROOT problem in order to obtain a global solution for G. This is discussed in Section 5.1.

Theorem 18. *Let $H_{\max}$ be a maxblock square root of a graph $G = (V, E)$, and let $A_1, A_2, \ldots, A_k$ be the closed neighbourhoods of every cut-vertex in $H_{\max}$. For every block B of $H_{\max}$, we can compute a graph G_B that is isomorphic to its square. Furthermore, we can also compute the mapping from $V(G_B)$ to B, unless B is reduced to a pending vertex and a non essential cut-vertex of $H_{\max}$. It can be done in $\mathcal{O}(n + m)$-time in total if an atom tree of G is given.*



(a) Square root H. (b) Square $G = H^2$. (c) Incidence graph for $A = N_H[v]$.

(d) Block-cut tree of I_A.

Figure 6: Computation of the connected components in a square root.

The remaining of this section is devoted to the proof of Theorem 18. It is based on the incidence graphs presented in Section 4.1. More precisely, we recall that for every clique-atom A of G, we define $\Omega(A)$ as the multiset containing $N_G(C)$ for every connected component C of $G \setminus A$. The incidence graph $I_A = Inc(\Omega(A), A)$ is the bipartite graph with respective sides $\Omega(A)$ and A and an edge between every $S \in \Omega(A)$ and every $u \in S$.

We need the following technical lemma.

Lemma 19. *Let $H_{\max}$ be a maxblock square root of a graph $G = (V, E)$. A vertex has a maximum neighbour in $H_{\max}$ if and only if it is a pending vertex.*

Proof. On the one hand, let $v \in V$ be a pending vertex of $H_{\max}$ with u being its unique neighbour in the square root. Clearly, u is a maximum neighbour of v in $H_{\max}$. Conversely, let $u, v \in V$ be such that u is a maximum neighbour of v in $H_{\max}$. We have that $N_G[v] = N_{H_{\max}}[u]$. In particular, let H_0 obtained from $H_{\max} \setminus v$ by adding vertex v and the edge $\{u, v\}$. Since u is a maximum neighbour of v, we have that H_0 keeps the property to be a square root of G. Furthermore, since $H_{\max}$ is a maxblock square root, we obtain that $H_0 = H_{\max}$. Hence, v is a pending vertex. $\square$

Given a clique-atom A and its incidence graph I_A, we can compute the blocks of I_A. Then, let us define the following equivalence relation over the connected components of $G \setminus A$: $C \sim_A C'$ if and only if $N_G(C)$ and $N_G(C')$ (taken as elements of $\Omega(A)$) are in the same block of I_A. The latter relation naturally extends to an equivalence relation over $V \setminus A$: for every components C, C' of $G \setminus A$ and for every $u \in C,\ u' \in C',\ u \equiv_A u'$ if and only if $C \sim_A C'$. In doing so, the equivalence classes of $\equiv_A$ partition the set $V \setminus A$. We refer to Figure 6 for an illustration of the procedure.

Lemma 20. *Let $G = (V, E)$ be a graph, let $H_{\max}$ be a maxblock square root of G, and let A be a clique-atom of G. Suppose that A is the closed neighbourhood of a cut-vertex v in $H_{\max}$ and let $C_1, C_2, \ldots, C_l$ be the nontrivial connected components of $H_{\max} \setminus v$. Then the equivalence classes of $\equiv_A$ are exactly the sets $C_i \setminus A,\ 1 \leq i \leq l$.*

Proof. This follows from a more careful use of the arguments in the proofs of Lemmas 11 and 12. On the one direction, let D be a connected component of $G \setminus A$. As observed in the proof of Lemma 11, it intersects exactly one of the sets $C_i \setminus A$. Furthermore (still by the proof of Lemma 11), for every $i \neq j$ we have that if $D \subseteq C_i \setminus A$ and $D' \subseteq C_j \setminus A$ then $N_G(D)$ and $N_G(D')$ are in different connected components of $I_A \setminus v$. Hence, each equivalence class of $\equiv_A$ intersects exactly one of the sets $C_i \setminus A$.

Conversely, let i be fixed, $1 \leq i \leq l$, and let $D, D' \subseteq C_i \setminus A$ be two connected components of $G \setminus A$. First note that since $N_{H_{\max}}(D) \cup N_{H_{\max}}(D') \subseteq A = N_{H_{\max}}[v]$, therefore $v \in N_G(D) \cap N_G(D')$. In addition, and as noted in the proof of Lemma 12, if there exists a bipartition of the connected components of $G \setminus A$, in P and Q say, such that $N_G(P) \cap N_G(Q) = \{v\}$, then the vertices in P are in different components of $H_{\max} \setminus v$ than the vertices in Q. In particular, for any such a bipartition we must have either $D, D' \in P$ or $D, D' \in Q$. Consequently, $N_G(D)$ and $N_G(D')$ (taken as vertices of I_A) cannot be disconnected by v. It implies the existence of two internally vertex-disjoint $N_G(D)N_G(D')$-paths of I_A (namely, $(N_G(D), v, N_G(D'))$ and any $N_G(D)N_G(D')$-path in $I_A \setminus v$. Altogether combined, $N_G(D), N_G(D') \in \Omega(A)$ are in a common block of I_A. In particular, all the components that intersect the set $C_i \setminus A$ are in the same equivalence class of $\equiv_A$. $\square$

We are now ready to prove Theorem 18.

Proof of Theorem 18. Our approach mimics the following "naive" algorithm for computing the blocks of $H_{\max}$. Let $v \in V$ be an arbitrary cut-vertex. We compute the connected components $C_1, \ldots, C_l$ of $H_{\max} \setminus v$. Then, it follows from the existence of the block-cut tree that the blocks of $H_{\max}$ are exactly the blocks contained in the subgraphs $H_{\max}[C_1 \cup \{v\}], \ldots, H_{\max}[C_l \cup \{v\}]$. Each of the subgraphs can thus be considered separately.

More formally, we consider the cut-vertices $v_1, \ldots, v_k$ of $H_{\max}$ sequentially. At each step i we maintain a T-decomposition of $H_{\max}$, that is, a collection of subgraphs $\mathcal{T}_i$ with the property that every block is contained in a unique subgraph of the collection. Initially, $\mathcal{T}_1 = \{H_{\max}\}$. We ensure that each cut-vertex $v_i, v_{i+1} \ldots, v_k$ is contained in a unique subgraph of $\mathcal{T}_i$. In particular, we consider the unique subgraph H_i that contains v_i and we compute the components $C_1, \ldots, C_l$ of $H_i \setminus v_i$. We finally construct the collection $\mathcal{T}_{i+1}$ from $\mathcal{T}_i \setminus \{H_i\}$ by adding, for every $1 \leq j \leq l$, the subgraph $H_i[C_j \cup \{v_i\}]$ to the current collection. Note that the graphs in the final set $\mathcal{T}_{k+1}$ are exactly the atoms of $H_{\max}$.

Back to the square G, we consider the sets $A_1, A_2, \ldots, A_k$ sequentially. Our purpose for proving Theorem 18 is to maintain an "atom forest" $\mathcal{F}_i$: with an atom tree for the square of every subgraph in $\mathcal{T}_i$. This cannot be done in general, as we may not know the cut-vertices of $H_{\max}$ (we know for sure their closed neighbourhoods). However, we prove that it can be done for all the subgraphs of $\mathcal{T}_i$ that are not reduced to a pending vertex and its (unique) neighbour in $H_{\max}$.

We set initially $\mathcal{F}_1 = \{(T_G, \mathcal{A})\}$ with $(T_G, \mathcal{A})$ being an atom tree of G. Furthermore, at each step i we ensure that each set $A_i, A_{i+1}, \ldots, A_k$ is a bag in a unique atom tree in $\mathcal{F}_i$. For every $1 \leq i \leq k$, we consider the unique $(T_i, \mathcal{A}_i) \in \mathcal{F}_i$ such that $A_i \in \mathcal{A}_i$. It represents an atom tree of $G_i = H_i^2$. Let $\Omega^*(A_i) = \{A_i \cap A' \mid \{A_i, A'\} \in E(T_G)\}$.

- First we compute the number s_i of vertices of G_i that are uniquely contained in A_i. It can be computed in time $\mathcal{O}(|A_i| + \sum_{S^* \in \Omega^*(A_i)} |S^*|)$. By Lemma 3, all these vertices are simplicial in G, with their closed neighbourhood in G being A_i. Since v_i is a maximum neighbour (in the square root) for all the simplicial vertices in A_i, by Lemma 19 there are exactly s_i trivial connected components of $H_{\max} \setminus v_i$. Note that each such component induces a block isomorphic to K_2.

- We then compute the blocks of the incidence graph $I_{A_i} = Inc(\Omega(A_i), A_i)$. Note that each block represents an equivalence class of $\equiv_{A_i}$ in G (not in G_i). Furthermore, as explained in the proof of Theorem 9, it can be done using the reduced incidence graph $I^*_{A_i} = Inc(\Omega^*(A_i), A_i)$. So, it can be done in time $\mathcal{O}(\sum_{S^* \in \Omega^*(A_i)} |S^*|)$.

  Let $C_1, C_2, \ldots, C_q$ be the remaining $q = l - s_i$ connected components of $H_i \setminus v$ (we remove from the list the trivial components that correspond to the s_i simplicial vertices in A_i). There is a bijection between these components and the nontrivial components $C'_1, C'_2, \ldots, C'_q$ of $H_{\max} \setminus v$. Moreover by Lemma 20, the set of equivalence classes of $\equiv_{A_i}$ is exactly $\{C'_j \setminus A_i \mid 1 \leq j \leq q\}$. Since by Lemma 19 vertex v_i cannot be a maximum neighbour of the vertices in these components, we can thus obtain the sets $C_j \cup \{v_i\}$ from $N_G[C'_j \setminus A_i] \cap V(H_i)$, for every $1 \leq i \leq q$.

- By the properties of tree decompositions, each set $C_j \cup \{v\}$ corresponds to a collection $\mathcal{C}_j$ of the subtrees in $T_i \setminus A_i$. Namely, these subtrees are obtained from the blocks of $I^*_{A_i}$, by grouping the subtrees whose respective intersections with A_i are in the same block of $I^*_{A_i}$. Let us remove all the subtrees of $T_i \setminus A_i$ that are not part of $\mathcal{C}_j$. Then, let us replace A_i by $A_i \cap (C_j \cup \{v\})$. We so obtain a tree decomposition $(T_{i,j}, \mathcal{A}_{i,j})$ for the square of $H_i[C_j \cup \{v\}]$. Finally, in order to obtain $\mathcal{F}_{i+1}$ from $\mathcal{F}_i \setminus \{(T_i, \mathcal{A}_i)\}$, we add all the $(T_{i,j}, \mathcal{A}_{i,j})$ in the collection, for every $1 \leq j \leq q$.

  Formally, the tree decompositions $(T_{i,j}, \mathcal{A}_{i,j})$ are not exactly atom trees since they may contain non maximal prime subgraphs (*e.g.*, the bag $A_i \cap (C_j \cup \{v\})$). However, it is necessary to keep these additional bags for the correctness of the algorithm. Indeed, consider the special case of a block B that is dominated by at least two cut-vertices v_{i_1}, v_{i_2}. At the i_1^{th} step, we replace A_{i_1} by B in the unique tree-decomposition $(T_{i_1,j}, \mathcal{A}_j)$ containing A_{i_2}. This is clearly not an atom since $B \subseteq A_{i_2}$. However, deleting this bag could result in missing B. This is also why we need to compute $\equiv_{A_i}$ in G rather than in the subgraph G_i.

  Overall, the time complexity is in:

$$\mathcal{O}(\sum_{A \in \mathcal{A}} \sum_{S^* \in \Omega^*(A)} |S^*|) = \mathcal{O}(\sum_{\{A,A'\} \in E(T_G)} |A \cap A'|),$$

that is $\mathcal{O}(n+m)$ by Lemma 5.

$\square$

Having a closer look at our reconstruction method, it computes for *any* square root (not necessarily maxblock) a collection of subsets of its blocks. We need Lemmas 19 and 20 in order to ensure that no subset can be proper, *i.e.*, the subsets computed are exactly the blocks of the root.

5.1 Discussion: obtaining a global solution from the blocks

By Theorem 18, we can compute, for every block of the desired square root, a graph that is isomorphic to its square. Then, we wish to solve the $\mathcal{H}$-SQUARE ROOT problem for each output graph separately, that can be done assuming the blocks are part of a class $\mathcal{H}$ for which the problem is trivial, or at least tractable. However, doing so, we may not be able to reconstruct a square root for the original graph. Indeed, the closed neighbourhoods of cut-vertices are imposed, and these additional constraints may be violated by the partial solutions. We thus need to solve a stronger version of the problem.

Problem 2 ($\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS)**.**

Input: *A graph $G=(V,E)$; a list $\mathcal{N}_F$ of pairs $\langle v, N_v\rangle$ with $v \in V,\ N_v \subseteq V$; a list $\mathcal{N}_A$ of subsets $N_i \subseteq V,\ 1 \leq i \leq k$.*

Question: *Are there a graph $H \in \mathcal{H}$ and a collection $v_1, v_2, \ldots, v_k \in V$ of pairwise disjoint vertices such that H is a square root of G, and:*

- *$\forall \langle v, N_v\rangle \in \mathcal{N}_F$, we have $N_H[v] = N_v$*
- *$\forall 1 \leq i \leq k$, we have $N_H[v_i] = N_i$; furthermore, $\langle v_i, N_i\rangle \notin \mathcal{N}_F$?*

Intuitively, the list $\mathcal{N}_F$ represents the *essential* cut-vertices and their closed neighbourhoods in the block. The list $\mathcal{N}_A$ represents the closed neighbourhoods of non essential cut-vertices. Let us point out that these two lists can be computed as a byproduct of the computation of the blocks.

Furthermore, non essential cut-vertices correspond to the vertices $v_1, \ldots, v_k$ to be computed. Notice that we need to ensure the pairwise disjointness of the vertices v_i in case there may be true twins in the square root. We also need to ensure that $\langle v_i, N_i\rangle \notin \mathcal{N}_F$ for the same reason. Finally, let us point out that if we can solve $\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS for each square of block separately, then a square root for the original graph G can be found by connected some pending vertices to the cut-vertices. The latter correspond to the simplicial vertices of G that are contained in the closed neighbourhood of a cut-vertex in its root.

We note that in the special case when $\mathcal{N}_A = \emptyset$ (*i.e.*, all the cut-vertices are known), our problem reduces to a particular case of $\mathcal{H}$-SQUARE ROOT WITH LABELS; this other variant of the problem has already received some attention in the literature [18].

6 Application to trees of cycle-powers

A cycle-power graph is any j^{th}-power C_n^j of the n-node cycle C_n, for some $j \geq 1$. In particular, if $j=1$ then C_n^1 is exactly the cycle C_n. If $2 \leq j \leq \lfloor n/2 \rfloor - 1$ then C_n^j is a $2j$-regular graph. Otherwise, $j \geq \lfloor n/2 \rfloor$ and the graph C_n^j is isomorphic to the complete graph K_n.

Definition 21. A tree of cycle-powers is a graph whose blocks are cycle-power graphs.

In particular, a j-cactus-block graph is a graph whose blocks are complete graphs or k^{th}-powers of cycles, for any $1 \leq k \leq j$.

Definition 21 generalizes the classes of trees, block graphs and cacti: where all the blocks are edges, complete subgraphs and cycles, respectively. Other relevant examples are the class of *cactus-block graphs* (*a.k.a.*, 1-cactus-block graphs with our terminology): where all the blocks are either cycles or complete subgraphs [40]; and the *Gallai trees*, that are the cactus-block graphs with no block being isomorphic to an even cycle [17].

The remaining of the section is devoted to prove the following result:

Theorem 22. *For every fixed $j \geq 1$, the squares of j-cactus-block graphs can be recognized in polynomial-time.*

Up to conceptually simple changes, the proof of Theorem 22 also applies to all the subclasses mentioned above. This solves for the first time the complexity of the $\mathcal{H}$-SQUARE ROOT problem when $\mathcal{H}$ is the class of cactus-block graphs or the class of Gallai trees:

Theorem 23. *The squares of cactus-block graphs, resp. the squares of Gallai trees, can be recognized in polynomial-time.*

On the complexity point of view, our algorithm runs in $\mathcal{O}(nm)$-time. This improves upon the $\mathcal{O}(n^4)$-time algorithm of [19] for the recognition of squares of cacti. Furthermore, the main bottleneck is the computation of an atom tree for the graph, that requires $\mathcal{O}(nm)$-time [5]. Indeed, all the other operations can be performed in linear-time. It implies that for graphs where an atom tree is easier to compute, such as chordal squares, we achieve a better time complexity. In particular, we retrieve as particular cases the linear-time algorithms for the recognition of squares of trees and block graphs of [26].

The remaining of the section is divided as follows. We first prove in Lemma 25 that if a graph has a square root that is a tree of cycle-powers then in particular, there is one such square root whose cut-vertices can be computed by using the constructions of Theorems 9 and 17. Then, we show with Lemma 26 that for each graph output by the algorithm of Theorem 18 (isomorphic to the square of a block), the $\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS problem can be solved in linear time.

6.1 Existence of a nice square root

We start with a basic property of cycle-power graphs.

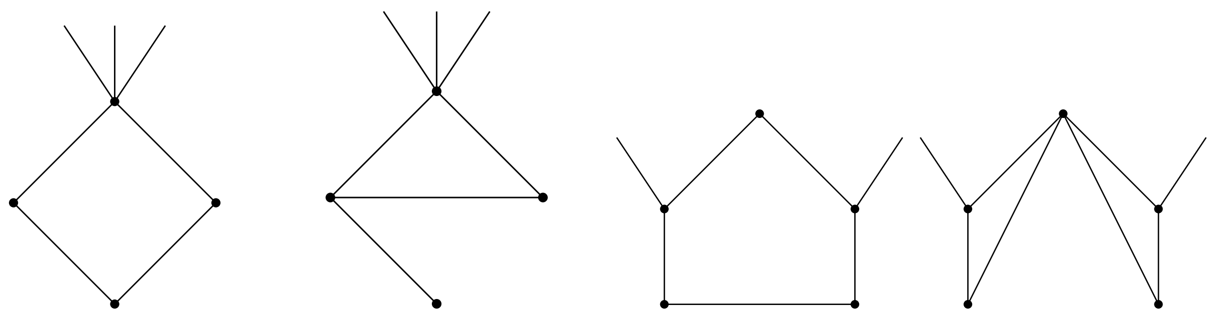
Lemma 24. *For every $j, n \geq 1$, the cycle-power graph C_n^j is prime.*

Proof. This is clear if $j \geq \lfloor n/2 \rfloor$ for then C_n^j is isomorphic to the complete graph K_n. Thus from now on, assume $1 \leq j < \lfloor n/2 \rfloor$. Let us label the vertices of C_n^j by $\mathbb{Z}_n$, in such a way that vertex i is adjacent to the vertices $i \pm k$, $1 \leq k \leq j$. It has been noted in [12] that every minimal separator S of C_n^j is the union of two disjoint "circular intervals", *i.e.*, $S = \{i_1, i_1+1, \ldots, i_1+j-1\} \cup \{i_2, i_2+1, \ldots, i_2+j-1\}$ for some i_1, i_2. Furthermore, the only two (full) components of $C_n^j \setminus S$ have respective vertex-sets $\{i_1+j, i_1+j+1, \ldots, i_2-1\}$ and $\{i_2+j, i_2+j+1, \ldots, i_1-1\}$ (indices are taken modulo n). In this situation, i_1 and i_2 are at distance at least $j+1$ in C_n. So, they cannot be adjacent in C_n^j. As a result, no minimal separator of C_n^j can be a clique, hence C_n^j is prime. $\square$

In order to prove the next result, we use classical techniques in the study of circular-arc graphs (intersection graphs of intervals on the cycle) [45]. Indeed, every cycle-power graph is a circular-arc graph [29].

Lemma 25. *Let $G = (V, E)$ be a graph that is not a clique. Let $H_{\max}$ be a* finest *square root of G that is a j-cactus-block graph. The set of closed neighbourhoods of the cut-vertices in $H_{\max}$ can be computed in $\mathcal{O}(n + m)$-time if an atom tree of G is given.*

Proof. Note that since every cycle-power graph is regular, and it has a prime square by Lemma 24, the blocks of $H_{\max}$ satisfy the two assumptions of Theorem 17. Furthermore, the characterization of this theorem allows to compute all the clique-atoms that can be the closed neighbourhood of a non essential cut-vertex in $H_{\max}$. In order to prove that they are exactly the set of these closed neighbourhoods, it suffices to prove that given $H_{\max}$, the graph modification used in Theorem 17 (in order to increase the number of non essential cut-vertices) outputs a j-cactus-block graph. Indeed, this modification takes a block B of the square root that has diameter two and contains a unique cut-vertex v; it makes of $S = N_{H_{\max}[B]}[v]$ a clique and it connects all the remaining vertices of $B \setminus S$ to an arbitrary vertex of $S \setminus v$. In the special case where $H_{\max}[B] = C_n^k$, $\lceil \lfloor n/2 \rfloor /2 \rceil \leq k \leq \min\{j, \lfloor n/2 \rfloor - 1\}$, this modification results in a clique S of size $2k+1$ (that induces the complete graph C_{2k+1}^k) and pending vertices. See Figures 7a and 7b for an illustration. Therefore, by maximality of the number of blocks in $H_{\max}$, the closed neighbourhoods of non essential cut-vertices in $H_{\max}$ are exactly the clique-atoms that satisfy the condition of Theorem 17. They can be computed in linear-time if an atom tree of G is given.



(a) A pending block. (b) Non-essential cut-vertex. (c) A "splittable" block. (d) Essential cut-vertex.

Figure 7: Local modifications of the blocks.

Similarly, we have by Lemma 11 that the essential cut-vertices of $H_{\max}$ are a subset of the set $\mathcal{C}$ that is computed with Algorithm 1. In order to prove that the set of essential cut-vertices of $H_{\max}$ is exactly $\mathcal{C}$, it suffices to prove that given $H_{\max}$, an appropriate variant of the graph modification of Lemma 12 (in order to increase the number of essential cut-vertices) outputs a j-cactus-block graph. As before, this operation takes a block B of the square root that has diameter two (*i.e.*, B is a clique-atom of G that is not contained in the closed neighbourhood of any cut-vertex of $H_{\max}$). We can thus write $H_{\max}[B] = C_n^k$, $\lceil \lfloor n/2 \rfloor /2 \rceil \leq k \leq \min\{j, \lfloor n/2 \rfloor - 1\}$. Furthermore, let $v_1, v_2, \ldots, v_c$, $c \geq 2$ be the cut-vertices that are contained in B.

First we prove as an intermediary claim that $c = 2$, *i.e.*, B only contains two cut-vertices. Furthermore, these two vertices v_1, v_2 have a unique common neighbour in B. In order to prove the claim, we see each of the $N_{C_n^k}[v_i]$ as a circular interval of $2k+1$ consecutive vertices on the cycle C_n. Observe that since B satisfies the assumptions of Lemma 12, the sets $N_{H_{\max}}[v_i] \setminus B$ can be bipartitioned into two nonempty parts P, Q such that $N_G(P) \cap N_G(Q) = \{v\}$. We derive the following consequences from this observation:

- The sets $N_{C_n^k}[v_i]$ cannot span all the edges of C_n (the underlying cycle of the block). Indeed, suppose by contradiction that every two consecutive vertices on this cycle are contained in one of the corresponding intervals of length $2k$. Then, any bipartition of the sets $N_{H_{\max}}[v_i] \setminus B$ into two non empty parts P and Q would result in $|N_G(P) \cap N_G(Q)| \geq 2$, that is a contradiction.
- In this situation, the edge-set $\bigcup_{i=1}^{c} N_{C_n^k}[v_i]$ induces an interval subgraph of C_n^k, and we can see the sets $N_{H_{\max}}[v_i] \setminus B$ as intervals on the infinite line. Since B is a clique of G, these intervals are pairwise intersecting. So, by the Helly property, they have a nonempty common intersection. This intersection must be reduced to v since otherwise, there could not exist a bipartition P, Q of the sets $N_{H_{\max}}[v_i] \setminus B$ such that $N_G(P) \cap N_G(Q) = \{v\}$.
- Finally, since all the intervals that represent the sets $N_{C_n^k}[v_i]$ on the line have the same length (equal to the degree $2k$) and a nonempty common intersection, the only possibility so that they can be bipartitioned into two sets that only intersect in v is that there are only two such intervals.

Altogether combined, this proves as claimed that B only contains two cut-vertices v_1, v_2 and that they have a unique common neighbour v in B.

In C_n^k, these three vertices thus correspond to some triple $(v_1, v, v_2) = (i, i+k, i+2k)$ for some $i \in \mathbb{Z}_n$ (indices are taken modulo n). Let us split the underlying cycle C_n of B into the three intervals $I_1 = N_{C_n^k}[v_1] = \{i-k, i-k+1, \ldots, i = v_1, i+1, \ldots, i+k = v\}$, $I_2 = N_{C_n^k}[v_2] = \{i+k = v, i+k+1, \ldots, i+2k = v_2, i+2k+1, \ldots, i+3k\}$ and $I_3 = B \setminus (I_1 \cup I_2)$ (with possibly I_3 being empty). By construction, $I_1 \cap I_2 = \{v\}$ and $I_1 \cap I_3 = I_2 \cap I_3 = \emptyset$. In order to make of vertex v a cut-vertex, we finally replace C_n^k by two cliques with respective vertex-sets I_1, I_2 (both isomorphic to C_{2k+1}^k) and pending vertices $i_3 \in I_3$ only adjacent to $i = v$. See Figures 7c and 7d for an illustration. This operation preserves the property for the square root to be a j-cactus-block graph. Therefore, by maximality of the number of blocks in $H_{\max}$, the essential cut-vertices of $H_{\max}$ and their closed neighbourhood are exactly those computed by Algorithm 1. By Theorem 9, they can be computed in linear-time if an atom tree of G is given. □

6.2 Solving a problem on circular intervals

Our proof of the previous Lemma 25 uses classical techniques in the study of circular-arc graphs (intersection graphs of intervals on the cycle) [45]. A circular-arc model is a mapping between the nodes of a circular-arc graph and some arcs/intervals on a cycle so that two vertices are adjacent if and only if the corresponding arcs intersect. We further exploit the relationship with this class of graphs and cycle-power graphs in order to prove Lemma 26.

Lemma 26. *Let $j \geq 1$ be a fixed constant. $\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS can be solved in linear-time when $\mathcal{H}$ is the class of j-cactus-block graphs.*

Proof. Let $G = (V, E)$; $\mathcal{N}_F$; $\mathcal{N}_A$ be an instance of the problem. There are two cases to be considered.

- Suppose that G is a clique. If there is no further constraint (*i.e.*, $\mathcal{N}_A = \mathcal{N}_F = \emptyset$) then G; $\mathcal{N}_F$; $\mathcal{N}_A$ is trivially a yes-instance. This is also the case if for every $\langle v, N_v \rangle \in \mathcal{N}_F$ we have $|N_v| = n$, and similarly for every $N_i \in \mathcal{N}_A$ we have $|N_i| = n$. Let us now assume that all these sets are pairwise different and they have equal odd size $2k+1$ (supposedly representing the closed neighbourhood of some vertex in C_n^k) for some $1 \leq k \leq j$. In this situation, since G is a clique we have $n \leq 4k+1 \leq 4j+1$. Furthermore since j is assumed to be a constant by the hypothesis, it implies that the graph G has bounded size, and so, the $\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS can be solved in constant-time by brute force[5].

- Else, G is not a clique. Since G is assumed to be the square of a cycle-power graph, there must exist some $1 \leq k \leq j$ such that: $k < \lceil \lfloor n/2 \rfloor /2 \rceil$; G is isomorphic to C_n^{2k}; the sets N_v, for every $\langle v, N_v \rangle \in \mathcal{N}_F$, and the sets $N_i \in \mathcal{N}_A$, are pairwise different and they have an equal odd size $2k+1$ (supposedly representing the closed neighbourhood of some vertex in C_n^k). It can be verified in linear-time [29]. Furthermore, in such situation, we are left for proving the existence of a one-to-one mapping $\varphi : V \to \mathbb{Z}_n$ such that: φ is an isomorphism of G and C_n^{2k}; for every $\langle v, N_v \rangle \in \mathcal{N}_F$, the vertices in N_v are consecutive with v being in the middle of the interval; each set $N_i \in \mathcal{N}_A$, is a "circular interval" of consecutive points on the cycle C_n.

  Given φ, we can construct a circular-arc model for $G = C_n^{2k}$ in the natural way with $\varphi(V) = \mathbb{Z}_n$ representing the extremal points of the arcs. It follows from [20] that a cycle-power graph that is not a clique has a unique such model up to rotation and reflection. So, let us compute this model. It takes linear-time [32]. Moreover, as proved in [20], we can obtain the desired mapping φ by labeling the vertices of G following a circular ordering of the starting points in this model. In such a situation, we are left to check whether all the neighbours constraints of $\mathcal{N}_F$ and $\mathcal{N}_A$ are satisfied by φ. It can be done in linear-time.

□

Altogether combined, we can finally prove Theorem 22.

Proof of Theorem 22. Given a graph $G = (V, E)$, the problem is trivial when G is a clique. So, we assume from now on that G is not a clique. We compute an atom tree of G. It can be done in $\mathcal{O}(nm)$-time [5]. Our objective is to compute a finest square root H of G that is a j-cactus-block graph. Assuming H exists, by Lemma 25 all the closed neighbourhood of the cut-vertices in H can be computed in $\mathcal{O}(n+m)$-time.

Then, we aim at computing the squares of the blocks of H, using Theorem 18. For this purpose, we only need to prove that the results of Lemmas 19 and 20 hold for H. First we claim that Lemma 19 holds for H. Indeed, observe that since in any cycle-power graph that is not a clique, there cannot be a dominated vertex, therefore there cannot be any vertex with a maximum neighbour either. It follows from this observation that, provided H exists, a vertex v of H can be a maximum neighbour of some other vertex w only if v and w are contained in a block B of H that induces a clique. Furthermore, v is the only possible cut-vertex contained in B (otherwise, v could not possibly be a maximum neighbour of w). In such situation, the clique B can be "splitted" into pending vertices adjacent to v. Hence, as claimed, the result of Lemma 19 holds for any *finest*

[5]Note, perhaps surprisingly, that this first case is the most difficult to solve with our methods. Indeed, solving the $\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS when G is a clique and j is arbitrary would directly lead to a polynomial-time recognition algorithm for the squares of trees of cycle-powers.

square root H that is a j-cactus-block graph. Then, we claim that Lemma 20 also holds for H. For proving the claim, let A be the closed neighbourhood of a cut-vertex v in H. As noted in Lemma 11 we have that every equivalence class of $\equiv_A$ intersects a unique connected component of $H \setminus v$. Conversely, we claim that for every nontrivial component C_i of $H \setminus v$, the subset $C_i \setminus A$ is contained in a unique equivalence class of $\equiv_A$. Indeed, either $C_i \setminus A$ is a connected subset of G (and so, we are done); or then, since by Lemma 24 the squares of the blocks of H are prime, we must have $B = A \cap (C_i \cup \{v\})$ is a block of H that disconnects $H[C_i \cup \{v\}]$. In the latter case, B is a clique of H since it contains a dominated vertex of H. As a result, all the connected components of $G[C_i \setminus A]$ are adjacent to every vertex of B (since they share with B a cut-vertex of H), hence $C_i \setminus A$ is in a unique equivalence class of $\equiv_A$. So, the claim is proved. Altogether combined, the latter is enough so that we can reuse the algorithm of Theorem 18 in order to compute, for every block of H, a graph isomorphic to its square. It can be done in $\mathcal{O}(n+m)$-time.

We finally need to solve the $\mathcal{H}$-SQUARE ROOT WITH NEIGHBOURS CONSTRAINTS problem for each of these output graphs separately. By Lemma 26, this last part of the algorithm can be performed in $\mathcal{O}(n+m)$-time in total. □

7 Conclusion

We intend the framework introduced in this paper to be applied for solving the $\mathcal{H}$-SQUARE ROOT problem in other graph classes — where the structure of the blocks is well-understood. We note by passing that a full characterization of non essential cut-vertices in the square roots would improve this framework and make it a bit less technical. In particular, this would leave us with solving a simpler variant of the $\mathcal{H}$-SQUARE ROOT WITH LABELS problem on the blocks. This is left as an interesting open question.

More generally, we aim at better understanding the relationships between small-size separators in a graph and small-diameter separators in its square. For instance, we believe that by studying the relationships between edge-separators in a graph and quasi-clique cutsets in its square (clique with one edge removed), we might be able to extend our framework in order to recognize the squares of outerplanar graphs. Let us mention that the complexity of recognizing the squares of planar graphs is still open.

References

[1] A. Adamaszek and M. Adamaszek. Uniqueness of graph square roots of girth six. *The Electronic Journal of Combinatorics*, 18(P139):1, 2011.

[2] G. Agnarsson and M. M. Halldórsson. Coloring powers of planar graphs. *SIAM Journal on Discrete Mathematics*, 16(4):651–662, 2003.

[3] M. Aigner and M. Fromme. A game of cops and robbers. *Discrete Applied Mathematics*, 8(1):1–12, 1984.

[4] A. Berry and J.-P. Bordat. Moplex elimination orderings. *Electronic Notes in Discrete Mathematics*, 8:6–9, 2001.

[5] A. Berry, R. Pogorelcnik, and G. Simonet. Organizing the atoms of the clique separator decomposition into an atom tree. *Discrete Applied Mathematics*, 177:1–13, 2014.

[6] A. Berry and G. Simonet. Computing the atom graph of a graph and the union join graph of a hypergraph. Technical Report arXiv:1607.02911, arXiv, 2016.

[7] M. Bonamy, B. Lévêque, and A. Pinlou. 2-distance coloring of sparse graphs. *Journal of Graph Theory*, 77(3):190–218, 2014.

[8] J. A. Bondy and U. S. R. Murty. *Graph theory.* Grad. Texts in Math., 2008.

[9] M.-S. Chang, M.-T. Ko, and H.-I. Lu. Linear-time algorithms for tree root problems. In *SWAT*, pages 411–422. Springer, 2006.

[10] M. Cochefert, J.-F. Couturier, P. A. Golovach, D. Kratsch, and D. Paulusma. Parameterized algorithms for finding square roots. *Algorithmica*, 74(2):602–629, 2016.

[11] M. C. Dourado, F. Protti, and J. L. Szwarcfiter. Complexity results related to monophonic convexity. *Discrete Applied Mathematics*, 158(12):1268–1274, 2010.

[12] P. Duchet, M. Las Vergnas, and H. Meyniel. Connected cutsets of a graph and triangle bases of the cycle space. *Discrete mathematics*, 62(2):145–154, 1986.

[13] G. Ducoffe and D. Coudert. Clique-decomposition revisited. In revision (Research Report on HAL, hal-01266147), 2017.

[14] B. Farzad and M. Karimi. Square-root finding problem in graphs, a complete dichotomy theorem. Technical Report arXiv:1210.7684, arXiv, 2012.

[15] B. Farzad, L. C. Lau, and N. N. Tuy. Complexity of finding graph roots with girth conditions. *Algorithmica*, 62(1-2):38–53, 2012.

[16] H. Fleischner. The square of every two-connected graph is hamiltonian. *Journal of Combinatorial Theory, Series B*, 16(1):29–34, 1974.

[17] T. Gallai. Graphen mit triangulierbaren ungeraden vielecken. *Magyar Tud. Akad. Mat. Kutató Int. Közl*, 7:3–36, 1962.

[18] P. Golovach, D. Kratsch, D. Paulusma, and A. Stewart. A linear kernel for finding square roots of almost planar graphs. In *SWAT*, pages 4:1–4:14, 2016.

[19] P. A. Golovach, D. Kratsch, D. Paulusma, and A. Stewart. Finding cactus roots in polynomial time. In *IWOCA*, pages 361–372. Springer, 2016.

[20] M. C. Golumbic and P. L. Hammer. Stability in circular arc graphs. *Journal of Algorithms*, 9(3):314–320, 1988.

[21] F. Harary, R. M. Karp, and W. T. Tutte. A criterion for planarity of the square of a graph. *Journal of Combinatorial Theory*, 2(4):395–405, 1967.

[22] J. Hopcroft and R. Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Communications of the ACM*, 16(6):372–378, 1973.

[23] L. C. Lau. Bipartite roots of graphs. *ACM Transactions on Algorithms (TALG)*, 2(2):178–208, 2006.

[24] L. C. Lau and D. G. Corneil. Recognizing powers of proper interval, split, and chordal graphs. *SIAM Journal on Discrete Mathematics*, 18(1):83–102, 2004.

[25] V. B. Le and N. T. Nguyen. A good characterization of squares of strongly chordal split graphs. *Information Processing Letters*, 111(3):120 – 123, 2011.

[26] V. B. Le and N. N. Tuy. The square of a block graph. *Discrete Mathematics*, 310(4):734 – 741, 2010.

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

[28] K.-W. Lih, W.-F. Wang, and X. Zhu. Coloring the square of a K_4-minor free graph. *Discrete Mathematics*, 269(1):303–309, 2003.

[29] M. C. Lin, D. Rautenbach, F. J. Soulignac, and J. L. Szwarcfiter. Powers of cycles, powers of paths, and distance graphs. *Discrete Applied Mathematics*, 159(7):621–627, 2011.

[30] Y.-L. Lin and S. S. Skiena. Algorithms for square roots of graphs. *SIAM Journal on Discrete Mathematics*, 8(1):99–118, 1995.

[31] E. Lloyd and S. Ramanathan. On the complexity of distance-2 coloring. In *ICCI*, pages 71–74. IEEE, 1992.

[32] R. M. McConnell. Linear-time recognition of circular-arc graphs. In *FOCS'01*, pages 386–394. IEEE.

[33] M. Milanič and O. Schaudt. Computing square roots of trivially perfect and threshold graphs. *Discrete Applied Mathematics*, 161(10):1538–1545, 2013.

[34] M. Molloy and M. R. Salavatipour. A bound on the chromatic number of the square of a planar graph. *Journal of Combinatorial Theory, Series B*, 94(2):189–213, 2005.

[35] R. Motwani and M. Sudan. Computing roots of graphs is hard. *Discrete Applied Mathematics*, 54(1):81–88, 1994.

[36] A. Mukhopadhyay. The square root of a graph. *Journal of Combinatorial Theory*, 2(3):290–295, 1967.

[37] N. V. Nestoridis and D. M. Thilikos. Square roots of minor closed graph classes. *Discrete Applied Mathematics*, 168:34–39, 2014.

[38] A. Oversberg and O. Schaudt. Polynomial time recognition of squares of ptolemaic graphs and 3-sun-free split graphs. In *WG*, pages 360–371. Springer, 2014.

[39] A. Parra and P. Scheffler. Characterizations and algorithmic applications of chordal graph embeddings. *Discrete Applied Mathematics*, 79(1-3):171–188, 1997.

[40] B. Randerath and L. Volkmann. A characterization of well covered block-cactus graphs. *Australas. J. Combin*, 9:307–314, 1994.

[41] N. Robertson and P. Seymour. Graph minors. II. algorithmic aspects of tree-width. *Journal of algorithms*, 7(3):309–322, 1986.

[42] I. C. Ross and F. Harary. The square of a tree. *The Bell System Technical Journal*, 39(3):641–647, 1960.

[43] Y. Shibata. On the tree representation of chordal graphs. *Journal of Graph Theory*, 12(3):421–428, 1988.

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

[45] A. Tucker. Characterizing circular-arc graphs. *Bulletin of the American Mathematical Society*, 76(6):1257–1260, 1970.

[46] G. Wegner. Graphs with given diameter and a coloring problem. Technical report, Univ. Dortmund, 1977.