



HAL
open science

On the recognition of C_4 -free and $1/2$ -hyperbolic graphs

David Coudert, Guillaume Ducoffe

► **To cite this version:**

David Coudert, Guillaume Ducoffe. On the recognition of C_4 -free and $1/2$ -hyperbolic graphs. [Research Report] RR-8458, 2014, pp.20. hal-00937935v1

HAL Id: hal-00937935

<https://inria.hal.science/hal-00937935v1>

Submitted on 29 Jan 2014 (v1), last revised 22 Jul 2014 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



On the recognition of C_4 -free and $1/2$ -hyperbolic graphs

David Coudert , Guillaume Ducoffe

**RESEARCH
REPORT**

N° 8458

January 2014

Project-Team COATI



On the recognition of C_4 -free and $1/2$ -hyperbolic graphs

David Coudert* †, Guillaume Ducoffe* † ‡

Project-Team COATI

Research Report n° 8458 — January 2014 — 20 pages

Abstract: The shortest-path metric d of a connected graph G is $\frac{1}{2}$ -hyperbolic if, and only if, it satisfies $d(u, v) + d(x, y) \leq \max\{d(u, x) + d(v, y), d(u, y) + d(v, x)\} + 1$, for every 4-tuple u, x, v, y of G . We show that the problem of deciding whether an unweighted graph is $\frac{1}{2}$ -hyperbolic is *subcubic equivalent* to the problem of determining whether there is a chordless cycle of length 4 in a graph. An improved algorithm is also given for both problems, taking advantage of fast rectangular matrix multiplication. In the worst case it runs in $O(n^{3.333953})$ time.

Key-words: Hyperbolicity, algorithm, graph, C_4 -free.

* Inria, France

† Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, 06900 Sophia Antipolis, France

‡ ENS Cachan, 61 Avenue du Président Wilson, 94230 Cachan, France

**RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE**

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Sur la reconnaissance des graphes sans C_4 et des graphes 1/2-hyperboliques

Résumé : La métrique d des plus courts chemins d'un graphe connexe G est $\frac{1}{2}$ -hyperbolique si et seulement si elle satisfait $d(u, v) + d(x, y) \leq \max\{d(u, x) + d(v, y), d(u, y) + d(v, x)\} + 1$ pour tout quadruplet u, x, v, y de G . Nous montrons que résoudre le problème de décider si un graphe est 1/2-hyperbolique revient à résoudre le problème de décider si un graphe contient un cycle sans corde de longueur 4, et inversement, en proposant une transformation en temps sous-cubique d'un problème vers l'autre. De plus, nous proposons un algorithme en temps $O(n^{3.333953})$, basé sur la multiplication de matrices rectangulaires, pour résoudre chacun de ces problèmes. Nous améliorons ainsi l'état de l'art.

Mots-clés : Hyperbolicité, algorithme, graphe, cycles induit.

1 Introduction

The primary aim of our work is to study hyperbolicity of simple unweighted graphs. This is a metric parameter, that was first introduced by Gromov in the context of automatic groups (see [22]), then extended to more general metric spaces [5]. Roughly, the hyperbolicity of a connected graph is a measure of how far is the shortest-path metric of the graph from a *tree metric*. One can deduce from this parameter tight bounds for the (worst) additive distortion of the distances in the graph when its vertices are embedded into a weighted tree [10]. Practical applications of hyperbolicity were proposed in the domains of routing [6], network security [26], bioinformatics [18], and in the spread of information in social networks [4].

So far, the best known algorithm for determining the hyperbolicity of a graph has a $O(n^{3.69})$ time complexity [20]. This is however prohibitive for graphs with tens of thousands of nodes such as the graph of the Autonomous Systems of the Internet, road maps, etc. An algorithm with good practical performances has been proposed in [11]. It improves the worst-case running time on certain graph classes, but it cannot be used on graphs with hyperbolicity less than one.

Related work Our work focuses on a decision version of the problem, namely the recognition of graphs with hyperbolicity (at most) $\frac{1}{2}$. Graphs with small hyperbolicity value have already received some attention, as a first characterization of $\frac{1}{2}$ -hyperbolic graphs was proposed in [1]. However, to the best of our knowledge, there was no known algorithmic application to it prior to this work. We are more interested in a reduction such as the recent one in [20], where the authors proved an equivalence between the problems of finding a 2-approximation for the hyperbolicity and the (max, min)-matrix multiplication. A similar line of research was followed in [32, 37], where they determined the subcubic equivalence between various combinatorial problems.

Our contribution We relate the recognition of graphs with hyperbolicity (at most) $\frac{1}{2}$ to the search of (induced) cycles of length 4, *e.g.* C_4 , in a graph. It actually follows from our work that either both problems are solvable in subcubic-time, or none of them is. We first present a linear-time reduction from the C_4 -free graph recognition problem to the recognition of $\frac{1}{2}$ -hyperbolic graphs (Section 3.1). Then we prove a new characterization of $\frac{1}{2}$ -hyperbolic graphs, which is based on graph powers [29], and from which it follows that, conversely, deciding whether a graph is $\frac{1}{2}$ -hyperbolic can be reduced in subcubic-time to the C_4 -free graph recognition problem (Section 3.2). In Section 4, we finally reduce both problems to the problem of the rectangular matrix product that was defined in [24]. This allows us to solve both of them in $O(n^{3.333953})$ time, which beats the previous records established in [20, 35].

We give the notations used in this paper in Section 2, along with definitions for graph hyperbolicity and C_4 -free graphs.

2 Definitions and Notations

A graph G is a pair (V, E) , whose n *vertices* are the elements of the set V , and whose m *edges* are the elements of E ; every edge is a set of two distinct vertices of G . The *neighborhood* $N(u)$ of a vertex $u \in V$, is the (possibly empty) set of vertices $v \in V$ such that $\{u, v\}$ is an edge. Alternatively, we say that the elements of $N(u)$ are *adjacent* to u . A *clique* is a set of pairwise adjacent vertices. Note that the adjacency relation is clearly symmetric; we also define the (symmetric) adjacency matrix $A = (\mathbb{I}_{\{u \in N(v)\}})_{u, v \in V}$, where \mathbb{I} denotes the Kronecker delta¹.

¹We use the symbol \mathbb{I} instead of the classical symbol δ for the Kronecker delta in order to prevent confusion with hyperbolicity.

Finally, an *induced subgraph* of G is a graph $G[X] = (X, F)$ such that $X \subseteq V$ and $F = \{\{u, v\} \in E : u, v \in X\}$. In particular if X is a clique, then it is called a *complete* subgraph. The induced subgraph is a *path of length* $l \geq 0$ if $|X| = l + 1$ and the vertices of X can be linearly ordered into a sequence (v_0, v_1, \dots, v_l) such that for every $0 \leq i, j \leq l$, the vertex v_i is adjacent to v_j if, and only if, $|j - i| = 1$. In such a case, the vertices v_0 and v_l are called the *endpoints* of the path, and the path is a v_0v_l -path. The graph G is *connected* if, for every pair $u, v \in V$, there exists a uv -path. Also, a cycle is a graph such that the deletion of any edge $\{v_0, v_l\} \in F$ yields a v_0v_l -path, and a tree is a connected graph which does not contain any cycle as a subgraph.

Further standard graph terminology can be found in [7, 14].

2.1 $\frac{1}{2}$ -hyperbolic graphs

Given a connected graph $G = (V, E)$, we define the distance $d_G(u, v)$ between two vertices $u, v \in V$ as the minimum length of a uv -path in the graph. This yields a (discrete) metric space (V, d_G) . For a survey of metric graph theory, the reader may refer to [2]. We define in the space (V, d_G) an interval $[u, v]$ between any two vertices $u, v \in V$, as the set of vertices “in between” u and v , e.g. $[u, v] = \{x \in V : d_G(u, v) = d_G(u, x) + d_G(x, v)\}$.

In the sequel, we will call a uv -path of minimum length a *uv-shortest path*, and we will denote the distance function by d instead of d_G whenever G is clear from the context. The graph hyperbolicity of G can now be defined as follows:

Definition 1 (4-points Condition, [22]). Let G be a connected graph.

For every 4-tuple u, x, v, y of G , we define $\delta(u, v, x, y)$ as half of the difference between the two largest sums amongst

$$S_1 = d(u, v) + d(x, y),$$

$$S_2 = d(u, x) + d(v, y), \text{ and}$$

$$S_3 = d(u, y) + d(v, x).$$

The graph hyperbolicity, denoted $\delta(G)$, is equal to $\max_{u, x, v, y} \delta(u, v, x, y)$.

Moreover, we say that G is δ -hyperbolic, for every $\delta \geq \delta(G)$.

Unlike many well-known graph properties, it is very important to note that the hyperbolicity of an *induced subgraph* of G does not yield any information in general about the hyperbolicity of G . For example, the wheel W_n is 1-hyperbolic, whereas it contains as an induced subgraph the cycle C_n whose hyperbolicity grows linearly with n [11]. A way to deal with this difficulty is to constrain ourselves to distance-preserving, or *isometric* subgraphs. Formally, an induced subgraph H of G is isometric if, and only if, it is connected and for every pair of vertices $u, v \in H$, we have that $d_H(u, v) = d_G(u, v)$. We also say that a subgraph which is not isometric is a *bridged* subgraph.

Lower and upper bounds on the hyperbolicity can be deduced from classical parameters such as the girth [31], the circumference [8], the domination number [33], and the chordality [28, 38]. In particular, we have that $\delta(G) \leq \lfloor \frac{\text{diam}(G)}{2} \rfloor$, where $\text{diam}(G) = \max_{u, v \in V} d_G(u, v)$ is the *diameter* of the graph [11, 38].

We inform the reader that Definition 1 is not a universal definition for the hyperbolicity of a graph. Some authors actually proposed and studied other definitions (see, for instance [13, 22]). Though the value of $\delta(G)$ may vary depending on the choice of the definition, any δ -hyperbolic graph with respect to (w.r.t.) any of the definitions is $f(\delta)$ -hyperbolic w.r.t. any other definition of the hyperbolicity². Moreover, the class of trees is always contained into the class of 0-hyperbolic graphs, which makes the graph hyperbolicity a tree-likeness parameter.

²The function f is linear in δ in most cases.

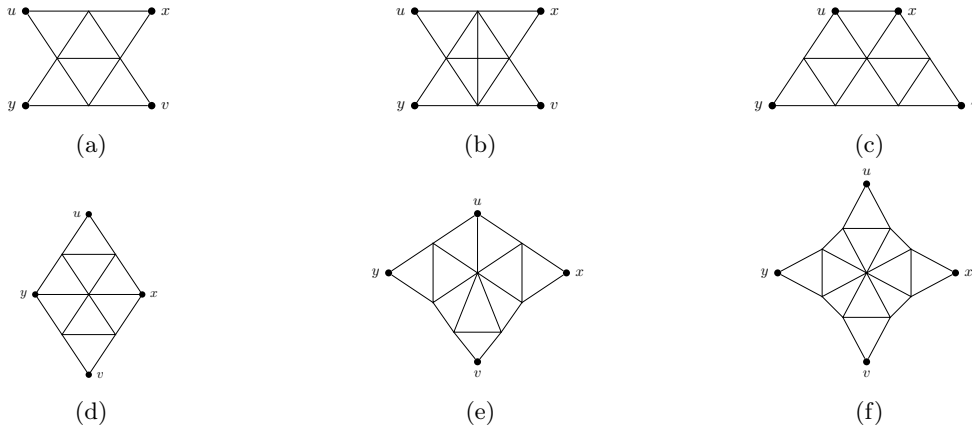


Figure 1: The six forbidden isometric subgraphs.

We here restrict our study to Definition 1, as it has algorithmic applications. Indeed, it is straightforward by using Definition 1 to compute the graph hyperbolicity $\delta(G)$ in $\Theta(n^4)$ time (see [11] and [20] for practical and theoretical improvements of the complexity). Also, note that $\delta(G)$ is always a half-integer (w.r.t. Definition 1). Our work focuses on graphs with small hyperbolicity, that is hyperbolicity at most $\frac{1}{2}$. Those graphs thus satisfy either $\delta(G) = 0$ or $\delta(G) = \frac{1}{2}$. We address the problem of recognizing those graphs, that we formulate as follows.

Problem 1. *Given a connected graph G , is G a $\frac{1}{2}$ -hyperbolic graph ?*

In [1], Bandelt and Chepoi characterized the $\frac{1}{2}$ -hyperbolic graphs as the connected graphs that simultaneously satisfy the three following conditions:

Condition 1. *Every cycle of length at least 6 in G is bridged.*

Condition 2. *For every pair $u, v \in G$, $N(u) \cap [u, v]$ is a clique.*

Condition 3. *No graph in Figure 1 is an isometric subgraph of G .*

A simpler characterization was previously given for 0-hyperbolic graphs [3, 23]. In fact, 0-hyperbolic graphs are *block-graphs*, that are graphs in which every biconnected component (block) is a clique (possibly reduced to a single vertex). This class includes cliques and trees, and a block-graph can be recognized in $O(n + m)$ time.

2.2 C_4 -free graphs

The C_4 -free graph recognition problem asks whether a given graph G contains an induced cycle of length 4. In the sequel, such a cycle, if any, is called a C_4 , or a *quadrangle*. A graph G which does not contain any C_4 as an induced subgraph is a C_4 -free graph. Let us define our decision problem in the following way.

Problem 2. *Given a graph G , does G contain a C_4 as an induced subgraph ?*

We now remind a well-known, local characterization of those graphs:

Lemma 2. *A graph $G = (V, E)$ is C_4 -free if, and only if, for every pair of non-adjacent vertices u, v , the set $N(u) \cap N(v)$ is a (possibly empty) clique.*

Proof. By the contrapositive, G contains an induced C_4 if, and only if, there exist $u, v, x, y \in V$ such that $\{u, x\}, \{u, y\}, \{v, x\}, \{v, y\} \in E$ and $\{u, v\}, \{x, y\} \notin E$. Equivalently, there exists a pair u, v of non-adjacent vertices such that there is a pair x, y of non-adjacent vertices in the set $N(u) \cap N(v)$. So, there is an induced quadrangle in G if, and only if, there exists a pair u, v of non-adjacent vertices such that the set $N(u) \cap N(v)$ is not a clique. \square

To see the relation between Problem 2 and Problem 1, one can observe that the condition of Lemma 2 is equivalent to Condition 2 when considering vertices u, v at distance 2. As a consequence, every $\frac{1}{2}$ -hyperbolic graph is also C_4 -free. In fact, a more direct way to see this is to note that every induced subgraph which is a quadrangle is *isometric* (this comes from the fact that a C_4 is connected and it has diameter 2). Since one can easily check that $\delta(C_4) = 1$, then it indeed follows that a $\frac{1}{2}$ -hyperbolic graph cannot contain a quadrangle as an induced subgraph.

So far, the best-known algorithm we are aware of to detect an induced C_4 in a graph has $O(n^{\omega(1)+1}) = O(n^{3.3727})$ time complexity [35], with $O(n^{\omega(1)})$ being the complexity of multiplying two $n \times n$ matrices (see Section 4.1 for details). We will improve this result in Section 4.

3 The subcubic equivalence

It is straightforward by the definitions that both the $\frac{1}{2}$ -hyperbolic graph recognition problem (Problem 1), and the C_4 -free graph recognition problem (Problem 2), are polynomial-time solvable [20, 35]. On the other hand, their best-known time complexity is strictly more than cubic. Thus it motivates the search for *subcubic reductions* between these problems, as they are defined in [37]. Formally, a subcubic reduction from a problem A to a problem B is a subcubic-time Turing reduction, which verifies the following additional properties on the oracle access to problem B. For every positive real μ , there has to exist a positive real ε such that:

- the reduction runs in $\tilde{O}(n^{3-\varepsilon})$ time³, where n denotes the size of the input;
- and given an instance of size n of problem A, $\sum_i \tilde{O}(n_i^{3-\mu}) = \tilde{O}(n^{3-\varepsilon})$, where n_i denotes the size of the i^{th} oracle call to problem B in the reduction.

In particular, a linear-time reduction is a subcubic reduction. More generally, any subcubic-time reduction which satisfies $\sum_i n_i = \tilde{O}(n)$ is also a subcubic reduction. We will only consider subcubic reductions of this kind in the sequel.

Subcubic reductions are of specific interest in the study of subcubic-time algorithms, as if there exists a subcubic reduction from problem A to problem B, and there is a subcubic-time algorithm which solves problem B, then there also exists a subcubic-time algorithm which solves problem A. In particular, if problems A and B are *subcubic equivalent*, then either both of them are solvable in subcubic-time, or none of them is. In this section, we will show that Problem 1 and Problem 2 are subcubic equivalent. We first present a linear-time reduction from Problem 2 to Problem 1 in Section 3.1 (Proposition 3). Then we present a subcubic reduction from Problem 1 to Problem 2 in Section 3.2 (Theorem 14).

³The notation $\tilde{O}(f(n))$ is for a complexity $O(f(n) \log^{O(1)} n)$.

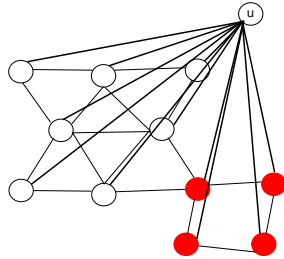


Figure 2: An illustration of the linear-time reduction of Proposition 3.

3.1 Reducing the detection of a C_4 to the recognition of a $\frac{1}{2}$ -hyperbolic graph

Proposition 3. *There is a linear-time reduction from the C_4 -free graph recognition problem to the problem of deciding whether a graph is $\frac{1}{2}$ -hyperbolic.*

Proof. Let $G = (V, E)$ be an instance of the C_4 -free graph recognition problem. Let $u \notin V$, and let $G' = (V \cup \{u\}, E \cup \{\{u, v\} : v \in V\})$. By construction, G' is connected, and it has diameter at most 2. Thus, we have that $\delta(G') \leq 1$ (e.g. see [11]), and all of its isometric subgraphs also have diameter at most 2.

Recall that a cycle of length l has diameter $\lfloor \frac{l}{2} \rfloor$. In particular, every cycle of length at least 6 is a graph of diameter at least 3 and as such, it cannot be an isometric cycle of G' . Consequently, G' always satisfies Condition 1. We can prove that it always satisfies Condition 3 in the same way.

Finally, since Condition 2 is satisfied for every pair $u, v \in V$ of *adjacent* vertices, then G' satisfies Condition 2 if, and only if, for every pair $u, v \in V$ of *non-adjacent* vertices, we have that $N_{G'}(u) \cap [u, v]$ is a clique; since $d_{G'}(u, v) = 2$ in such a case, then it is equivalent to have that $N_{G'}(u) \cap N_{G'}(v)$ is a clique, e.g. the graph G' is C_4 -free by Lemma 2.

Moreover, we have by construction that any induced C_4 in G' is an induced quadrangle in G , and vice-versa. Consequently, G is C_4 -free if, and only if, the graph G' is $\frac{1}{2}$ -hyperbolic. \square

We want to highlight that our reduction from Problem 2 to Problem 1 here is linear-time, whereas the converse reduction from Problem 1 to Problem 2, presented in Section 3.2, is super-linear. It might be of interest to determine whether a linear-time reduction from Problem 1 to Problem 2 exists.

3.2 Finding quadrangles to recognize non- $\frac{1}{2}$ -hyperbolic graphs

Our aim is now to prove that there exists a subcubic reduction from Problem 1 to Problem 2 (Theorem 14). Ideally, we would ask for a subcubic-time routine for checking whether Conditions 1, 2 and 3 are satisfied. Our reduction is however more complex, as we actually introduce and verify different conditions in subsequent sections. Thus we have to prove that these conditions imply Conditions 1, 2 and 3, and also that they are satisfied by $\frac{1}{2}$ -hyperbolic graphs.

3.2.1 Quickly excluding long isometric cycles

Let us first deal for our reduction with a tool that we will use later to verify whether Condition 1 is satisfied. We recall that Condition 1 requires that every cycle of length at least 6 in G is a

bridged subgraph. A first naive approach to deal with this condition is to compute the length of a longest isometric cycle of G . This can be done in polynomial-time, but the best-known time complexity for this is $O(n^{2\omega(1)} \log n) = O(n^{4.752} \log n)$, which is super-cubic [30]. Instead, we propose in this section a way to weaken Condition 1, by only having to consider isometric cycles of *polylogarithmically-bounded* length.

Given a connected graph G , we recall that a c -factor approximation of the hyperbolicity of G is a half-integer $\delta_c(G)$ such that $\delta(G) \leq \delta_c(G) \leq c \cdot \delta(G)$. In this section, we will assume we have a subcubic-time algorithm computing a c -factor approximation of the hyperbolicity, for some fixed choice of $c = O(\log^{O(1)} n)$. Below, we remind possible ways to achieve such a result:

Lemma 4 ([9, 20]). *Let $G = (V, E)$ be a connected graph.*

- *There exists an algorithm computing a 2-factor approximation of the hyperbolicity in $O(n^{\frac{3-\omega(1)}{2}}) = O(n^{2.69})$ time [20].*
- *There exists an algorithm computing a 6272-factor approximation of the hyperbolicity in $O(\min\{nm, n^{\omega(1)} \log n\} + n^2 \log \delta(G)) = \tilde{O}(n^{2.3727})$ time⁴ [9].*

Combining results from [10, 15, 20], there also exists a $\theta(\log n)$ -factor approximation algorithm of the hyperbolicity in $\tilde{O}(n^2)$ time⁵.

In addition, one can deduce from the results from [16, 17, 21] an algorithm which computes a $\theta(\log^2 n)$ -factor approximation of the hyperbolicity in $O(m \log n)$ time.

It is well-known that the hyperbolicity of the cycle C_n grows linearly with n . Formally:

Lemma 5 ([11, 38]). *Cycles of order $4p + \varepsilon \geq 3$, with $p \geq 0$ and $\varepsilon \in \{0, 1, 2, 3\}$, are $(p - 1/2)$ -hyperbolic when $\varepsilon = 1$, and p -hyperbolic otherwise.*

As a result, it follows from Lemma 5 that a (polylogarithmic) upper-bound on the hyperbolicity of G yields a (polylogarithmic) upper-bound on the length of a longest isometric cycle of G ; more accurately:

Corollary 6. *Let $G = (V, E)$ be a connected graph. Then all the isometric cycles of G have length upper-bounded by $4\delta(G) + 3$.*

Proof. First assume $\delta(G)$ is an integer. By Lemma 5, the longest $\delta(G)$ -hyperbolic cycle has length $4\delta(G) + 3$. Otherwise, $\delta(G)$ is a half-integer by Definition 1 and so, again by Lemma 5, the longest $\delta(G)$ -hyperbolic cycle has length $4(\delta(G) + \frac{1}{2}) + 1 = 4\delta(G) + 3$. \square

Since we are interested in $\frac{1}{2}$ -hyperbolic graphs, then Corollary 6 implies that every isometric cycle must have length upper-bounded by 5, *i.e.* Condition 1. We introduce in next section a second tool so that we can detect isometric cycles of polylogarithmically-bounded length.

⁴The term $\min\{nm, n^{\omega(1)} \log n\}$ in the complexity comes from the computation of all the distances in the graph (see [34] for an algorithm in $\tilde{O}(n^{\omega(1)})$ time for the problem).

⁵The authors in [20] actually claim a $O(n^2)$ time complexity for their algorithm, but they do not take into account the computation of all the distances in the graph. This can be corrected by using a $\theta(\log n)$ -additive approximation of all the distances, which can be computed in $\tilde{O}(n^2)$ time (see [15]). Also, their algorithm requires to compute all the distances in an intermediate tree of quadratic size, but this can be replaced with a tree of linear size using [10].

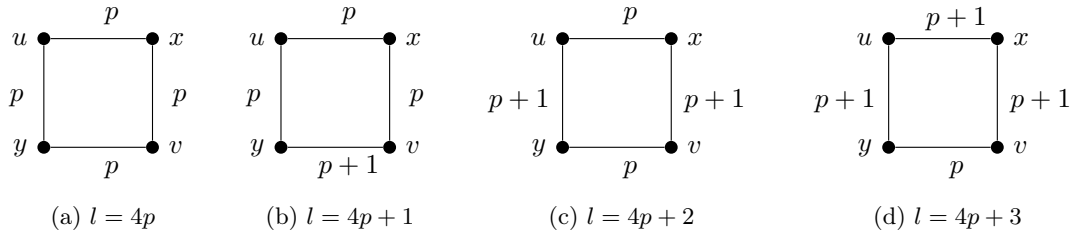


Figure 3: Shrinking long isometric cycles into quadrangles.

3.2.2 Using graph powers

Let us now present the main tool for our reduction, namely graph powers.

Definition 7. Given a connected graph $G = (V, E)$, let i be a positive integer. The i^{th} -power of G , denoted $G^i = (V, E_i)$, is a graph whose set of vertices is the same as for G ; two vertices $u, v \in V$ are adjacent in G^i if, and only if, there exists a uv -path of length at most i in G . Formally, $E_i = \{\{u, v\} : 0 < d_G(u, v) \leq i\}$.

In particular, the graph power G^1 is G , and the graph power $G^{\text{diam}(G)}$ is the complete graph K_n , where $\text{diam}(G) = \max_{u, v \in V} d(u, v)$ is the diameter of G . It is folklore that the graph power G^i can be computed in $O(n^{\omega(1)} \log i)$ time, using fast square matrix multiplication [34].

Recall that we said in Section 2.2 that every $\frac{1}{2}$ -hyperbolic graph G is C_4 -free. Roughly, most of our reduction will consist in checking whether a *polylogarithmic* number of graph powers of G are C_4 -free as well. This is a necessary condition so that G is $\frac{1}{2}$ -hyperbolic, as stated below.

Lemma 8. *If G is a $\frac{1}{2}$ -hyperbolic graph, then for every positive integer i , the graph G^i is C_4 -free.*

Proof. By contradiction, let u, v, x, y be the vertices of an induced quadrangle in G^i , for some positive integer $i \geq 1$. Without loss of generality, assume that $x, y \in N_{G^i}(u) \cap N_{G^i}(v)$. It follows by Definition 7 that:

$$\begin{aligned} \max\{d_G(u, x), d_G(u, y), d_G(v, x), d_G(v, y)\} &\leq i; \\ \text{and } \min\{d_G(u, v), d_G(x, y)\} &\geq i + 1. \end{aligned}$$

As a consequence, we have by Definition 1 that:

$$\begin{aligned} \delta(u, v, x, y) &= \frac{1}{2} [(d_G(u, v) + d_G(x, y)) - \max\{d_G(u, x) + d_G(v, y), d_G(u, y) + d_G(v, x)\}] \\ &\geq \frac{1}{2} [2(i + 1) - 2i] \\ &\geq 1 \end{aligned}$$

which contradicts the fact that G is $\frac{1}{2}$ -hyperbolic. □

Intuitively, the existence of isometric cycles of length l in G yields the existence of induced quadrangles in some graph power $G^{\theta(l)}$; this shrinking effect is illustrated with Figure 3. As a result, it may be more efficient to search for induced cycles of length 4 in the graph powers rather than computing the length of a longest isometric cycle of G directly.

We emphasize that the converse does not hold: not all the induced quadrangles in G^i yield an isometric cycle in the original graph G . For instance, the square graph H_1^2 of the graph H_1 in

Figure 1a contains a quadrangle as an induced subgraph (the vertices of which are u, x, v, y), yet H_1 does not contain an isometric cycle of length more than 3. However, we remind that every graph power has to be C_4 -free by Lemma 8 and so, *any* induced quadrangle that we detect in some graph power is a certificate to prove that the graph is not $\frac{1}{2}$ -hyperbolic.

We formalize it as follows.

Lemma 9. *Let $G = (V, E)$ be a connected graph, and let C_l be an isometric cycle of length $l = 4p + \varepsilon$, $p \geq 1$, $\varepsilon \in \{0, 1, 2, 3\}$ and $l \neq 5$. There is an integer $i \in [p, 2p]$ such that the graph power G^i contains a C_4 as an induced subgraph.*

Proof. Let us fix an arbitrary orientation for C_l , and choose a 4-tuple u, x, v, y (in clockwise order) such that:

- if $\varepsilon = 0$: $d_G(u, x) = d_G(u, y) = d_G(v, x) = d_G(v, y) = p$;
- if $\varepsilon = 1$: $d_G(u, x) = d_G(u, y) = d_G(v, x) = p$, $d_G(v, y) = p + 1$;
- if $\varepsilon = 2$: $d_G(u, x) = d_G(v, y) = p$, $d_G(u, y) = d_G(v, x) = p + 1$;
- if $\varepsilon = 3$: $d_G(u, x) = d_G(u, y) = d_G(v, x) = p + 1$, $d_G(v, y) = p$.

An example of such a choice is given in Figure 3. Note that all the above distances are upper-bounded by $p + \lceil \frac{\varepsilon}{4} \rceil$. Furthermore, recall that C_l is an isometric cycle by the hypothesis. So, we have:

$$d_G(u, v) = d_G(x, y) = \begin{cases} 2p & \text{when } \varepsilon \in \{0, 1\} \\ 2p + 1 & \text{when } \varepsilon \in \{2, 3\} \end{cases}$$

Equivalently, we have $d_G(u, v) = d_G(x, y) = 2p + \lfloor \frac{\varepsilon}{2} \rfloor$. As a consequence, by Definition 7, we have that $G^i[\{u, x, v, y\}]$ is an induced C_4 , for every $p + \lceil \frac{\varepsilon}{4} \rceil \leq i \leq 2p + \lfloor \frac{\varepsilon}{2} \rfloor - 1$. To prove that such a value of i always exists, it now remains to prove that:

$$p + \lceil \frac{\varepsilon}{4} \rceil \leq 2p + \lfloor \frac{\varepsilon}{2} \rfloor - 1,$$

that is:

$$\left(2p + \lfloor \frac{\varepsilon}{2} \rfloor - 1\right) - \left(p + \lceil \frac{\varepsilon}{4} \rceil\right) = p + \left(\lfloor \frac{\varepsilon}{2} \rfloor - \lceil \frac{\varepsilon}{4} \rceil\right) - 1 \geq 0.$$

A straightforward calculation shows that:

$$\lfloor \frac{\varepsilon}{2} \rfloor - \lceil \frac{\varepsilon}{4} \rceil = \begin{cases} 0 & \text{if } \varepsilon \neq 1 \\ -1 & \text{otherwise} \end{cases}$$

As a consequence, if $\varepsilon \neq 1$ then we are done. Otherwise, since $l = 4p + 1 > 5$ by the hypothesis, then $p \geq 2$ and so:

$$p + \left(\lfloor \frac{\varepsilon}{2} \rfloor - \lceil \frac{\varepsilon}{4} \rceil\right) - 1 \geq 2 - 1 - 1 \geq 0.$$

□

Note that the only two possible lengths for an isometric cycle of the graph that Lemma 9 does not take into account are 3 and 5. This is not a coincidence, as C_3 and C_5 are the only cycles that are $\frac{1}{2}$ -hyperbolic by Lemma 5.

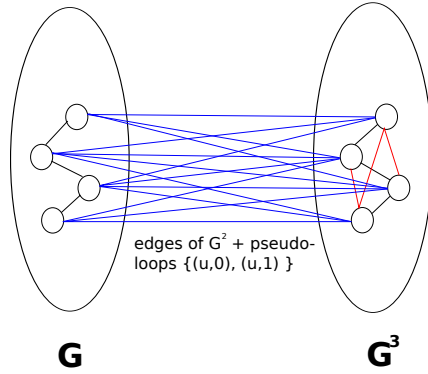


Figure 4: The construction of the graph $G^{[2]}$.

3.2.3 Transforming some obstructions into quadrangles

One of the most fundamental step for our reduction was to prove with Lemma 8 that every graph power G^i , $i \geq 1$, has to be C_4 -free so that a connected graph G is $\frac{1}{2}$ -hyperbolic. An interesting question on its own is whether it is also a sufficient condition.

We give a negative answer to this conjecture, using the graph H in Figure 6. The graph H is the union of a C_5 and a C_3 that both share a single edge; using the 4-tuple in bold in Figure 6, one can verify that $\delta(H) \geq 1$. Clearly, H does not contain a quadrangle as an induced subgraph. Moreover, $\text{diam}(H) = 3$ and so, for every $i \geq 3$, the graph power H^i is a complete graph. Finally, as there is only one couple x, y of H such that $d_H(x, y) = 3$, it follows that the square graph H^2 of H only lacks a single edge to be complete; as a result, H^2 is C_4 -free as well.

The primary aim of this section is now to show that in order to complete our reduction, we only need to decide whether only *one* additional graph is C_4 -free:

Definition 10. Let $G = (V, E)$ be a connected graph. The graph $G^{[2]} = (V^{[2]}, E^{[2]})$ is defined as follows:

- $V^{[2]} \simeq V \times \{0, 1\}$;
- $G^{[2]}[V \times \{0\}] \simeq G$;
- $G^{[2]}[V \times \{1\}] \simeq G^3$;
- $\forall u, v \in V$, the vertices $(u, 0)$ and $(v, 1)$ are adjacent in $G^{[2]}$ if, and only if, $d_G(u, v) \leq 2$.

In particular, $\forall u \in V$, there is an edge $\{(u, 0), (u, 1)\} \in E^{[2]}$.

An illustration of the construction of $G^{[2]}$ is presented in Figure 4. It might help to observe that for every edge of G^2 , there are exactly two corresponding edges in $G^{[2]}$ that connect the sets $V \times \{0\}$ and $V \times \{1\}$. Every other connecting edge of $G^{[2]}$ is a pseudo-loop $\{(u, 0), (u, 1)\}$, for some vertex $u \in V$.

We interpret the role of $G^{[2]}$ as the role of an "intermediate power" between the square graph G^2 and the cube graph G^3 . First, let us prove that it is necessary for $G^{[2]}$ to be C_4 -free so that G is $\frac{1}{2}$ -hyperbolic.

Lemma 11. *If G is a $\frac{1}{2}$ -hyperbolic graph, then the graph $G^{[2]}$ is C_4 -free.*

Proof. By contradiction, let a, b, c, d be the vertices of an induced quadrangle in $G^{[2]}$. Without loss of generality, we assume that $d_{G^{[2]}}(a, b) = d_{G^{[2]}}(c, d) = 2$. Several cases have to be considered.

If $a, b, c, d \in V \times \{0\}$, then there is an induced C_4 in G by Definition 10. Similarly, if $a, b, c, d \in V \times \{1\}$, then there is an induced C_4 in G^3 by Definition 10. In both cases, this implies that G is *not* $\frac{1}{2}$ -hyperbolic, by Lemma 8.

For all the remaining cases, we claim that there is no vertex $u \in V$ such that $\{(u, 0), (u, 1)\} \subseteq \{a, b, c, d\}$. It easily follows from the fact that $N_{G^{[2]}}((u, 0)) \cup \{(u, 0)\} \subseteq N_{G^{[2]}}((u, 1)) \cup \{(u, 1)\}$. Thus we can write $(a, b, c, d) = ((u, k), (v, k'), (x, j), (y, j'))$, with $\{k, k', j, j'\} = \{0, 1\}$ and vertices u, v, x, y are pairwise distinct.

Case 1: $k = 0, k' = j = j' = 1$ Here it comes that:

$$\begin{aligned} \max\{d_G(u, x) + d_G(v, y), d_G(u, y) + d_G(v, x)\} &\leq 2 + 3 = 5, \\ \text{whereas } d_G(u, v) + d_G(x, y) &\geq 3 + 4 = 7. \end{aligned}$$

In other words, $\delta(G) \geq \delta(u, v, x, y) \geq 1$.

Case 2: $k = 1, k' = j = j' = 0$ In such a case:

$$\begin{aligned} \max\{d_G(u, x) + d_G(v, y), d_G(u, y) + d_G(v, x)\} &\leq 2 + 1 = 3, \\ \text{whereas } d_G(u, v) + d_G(x, y) &\geq 3 + 2 = 5. \end{aligned}$$

So, $\delta(G) \geq \delta(u, v, x, y) \geq 1$.

Case 3: $k = k' = 0, j = j' = 1$ It follows that:

$$\begin{aligned} \max\{d_G(u, x) + d_G(v, y), d_G(u, y) + d_G(v, x)\} &\leq 2 + 2 = 4, \\ \text{whereas } d_G(u, v) + d_G(x, y) &\geq 2 + 4 = 6. \end{aligned}$$

In other words, $\delta(G) \geq \delta(u, v, x, y) \geq 1$.

Case 4: $k = j = 0, k' = j' = 1$ Then we have:

$$\begin{aligned} \max\{d_G(u, x) + d_G(v, y), d_G(u, y) + d_G(v, x)\} &\leq \max\{1 + 3, 2 + 2\} = 4, \\ \text{whereas } d_G(u, v) + d_G(x, y) &\geq 3 + 3 = 6. \end{aligned}$$

So, we again conclude that $\delta(G) \geq \delta(u, v, x, y) \geq 1$.

□

Recall that in order to decide whether a graph is $\frac{1}{2}$ -hyperbolic, our aim is to check whether all of the three Conditions 1, 2 and 3 of [1] are satisfied, using stronger necessary conditions. So far, we only dealt with Condition 1, developing tools in Sections 3.2.1 and 3.2.2 in order to determine if every cycle of length at least 6 in the graph is bridged. The following two lemmas will show a way to combine the graph $G^{[2]}$ of Definition 10 with the graph powers defined with Definition 7, in order to ensure that both Conditions 2 and 3 are satisfied as well.

Let us start with Condition 2:

Lemma 12. *Let $G = (V, E)$ be a δ -hyperbolic graph, for some $\delta \geq \frac{1}{2}$. Suppose G does not satisfy Condition 2. Then $G^{[2]}$ is not C_4 -free, or there exists some positive integer $i \leq 2\delta$ such that G^i is not C_4 -free.*

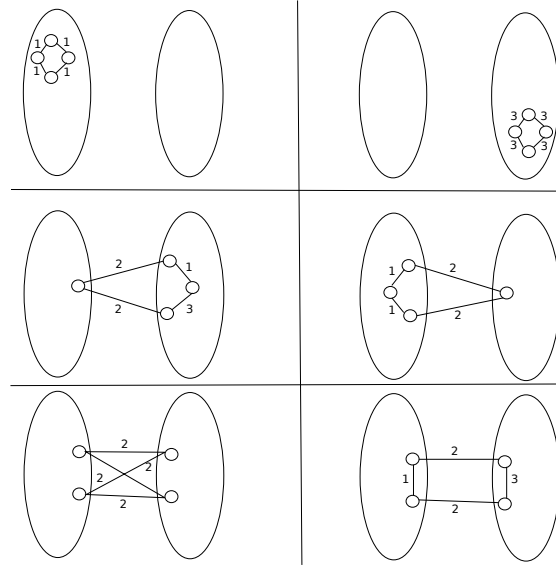


Figure 5: Possible quadrangles in $G^{[2]}$. Weights on the edges represent distances in G .

Proof. Let $u, v \in V$ be such that $N(u) \cap [u, v]$ is not a clique, and $d(u, v)$ is minimum w.r.t. this property. Let $x_1, y_1 \in N(u) \cap [u, v]$ such that x_1 and y_1 are not adjacent in G .

- Note that if $d(u, v) = 2$, then we are done as u, v, x_1, y_1 are the vertices of an induced C_4 in G .
- Similarly, if $d(u, v) = 3$, then we have:
 - $d(u, x_1) = d(u, y_1) = 1$;
 - $d(x_1, y_1) = d(v, x_1) = d(v, y_1) = 2$;

as a consequence, $(u, 0), (x_1, 0), (y_1, 0), (v, 1)$ are the vertices of an induced quadrangle in $G^{[2]}$.

In the sequel, we will assume $d(u, v) \geq 4$. Let us define the two uv -shortest paths:

$$P_1 = u, x_1, x_2, x_3, \dots, x_{d(u,v)-1}, v$$

$$\text{and } P_2 = u, y_1, y_2, y_3, \dots, y_{d(u,v)-1}, v$$

Note that for every $i \leq j$ we have $d(x_i, y_j) \geq d(y_i, y_j) = j - i$, and in the same way $d(x_j, y_i) \geq d(x_j, x_i) = j - i$. We now claim that for every $i \leq j$, the inequalities above are strict, or equivalently $d(x_i, y_j) > j - i$ and $d(x_j, y_i) > j - i$. By contradiction, suppose that there is some $i \leq j$ satisfying $d(x_i, y_j) = j - i$. Then in such a case we have $x_1, y_1 \in N(u) \cap [u, y_j]$, contradicting the minimality of $d(u, v)$. The case when $d(x_j, y_i) = j - i$ is dealt with similarly. To sum up, by the minimality of $d(u, v)$ we have that for any $i \leq j$:

$$d(x_i, y_j) > d(y_i, y_j) = j - i \text{ and } d(x_j, y_i) > d(x_j, x_i) = j - i.$$

In particular, for all i, j this yields that x_i and y_j are pairwise distinct.

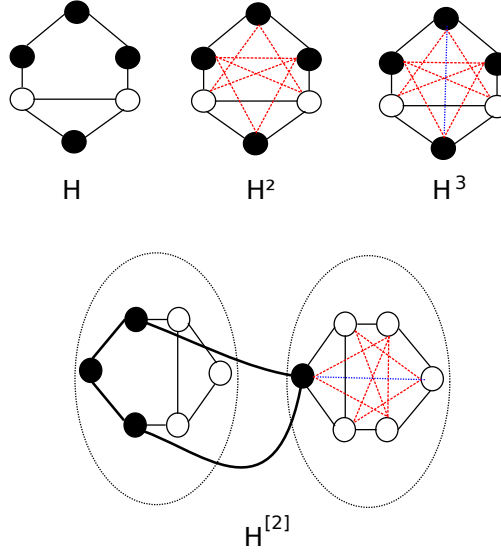


Figure 6: A 1-hyperbolic graph whose powers are C_4 -free. A quadrangle in $G^{[2]}$ is drawn in bold.

Also, note that for all i , $\delta(u, v, x_i, y_i) = \frac{d(x_i, y_i)}{2}$ and so, we have that for every i :

$$d(x_i, y_i) \leq 2\delta.$$

Let $l \geq 3$ be the least index greater than 2 such that $d(x_l, y_l) \leq l - 1$. One has to observe that since for all i , $d(x_i, y_i) \leq 2\delta$, it holds that $l \leq \min\{2\delta + 1, d(u, v) - 1\}$. In such a case:

- $d(x_1, x_l) = d(y_1, y_l) = l - 1$;
- $d(x_1, y_1), d(x_l, y_l) \leq l - 1$;
- $d(x_1, y_l), d(x_l, y_1) > l - 1$.

Consequently, x_1, y_1, x_l, y_l are the vertices of an induced quadrangle in G^{l-1} . \square

Let us finally prove with Lemma 13 that we can verify whether Condition 3 is satisfied in the same way as we dealt with Condition 2 in Lemma 12.

Lemma 13. *Let $G = (V, E)$ be a connected graph that does not satisfy Condition 3. Then there is an induced C_4 in the graph $G^{[2]}$, or there is an induced C_4 in the square graph G^2 .*

Proof. We proceed by contradiction. Let us assume that one of the graphs of Figure 1 is an isometric subgraph of G . For each of the forbidden graphs of Condition 3, we will only consider the 4-tuple of vertices that are drawn in bold in Figure 1, denoted u, y, v, x .

Cases 1a and 1b One can easily check that in both cases, the four vertices in bold are the vertices of an induced quadrangle in the square graph G^2 .

Case 1c Observe that $d(u, x) = 1$, $d(u, y) = d(v, x) = 2$, $d(u, v) = d(x, y) = d(v, y) = 3$. So, $(u, 0), (y, 1), (v, 1), (x, 0)$ are the vertices of an induced quadrangle in $G^{[2]}$. A contradiction.

Case 1d We have that all distances but $d(u, v)$ equal 2, and that $d(u, v) = 4$. Therefore, $(u, 1), (y, 0), (v, 1), (x, 0)$ are the vertices of an induced C_4 in $G^{[2]}$. Again, this is not possible.

Case 1e Observe that $d(u, x) = d(u, y) = 2$, $d(x, y) = 4$, and all the remaining distances are equal to 3. As a consequence, $(u, 0), (y, 1), (v, 1), (x, 1)$ are the vertices of an induced quadrangle in $G^{[2]}$, which contradicts the fact that $G^{[2]}$ is C_4 -free.

Case 1f The vertices $(u, 1), (y, 1), (v, 1), (x, 1)$ induce a C_4 in G^3 , hence in $G^{[2]}$, that is once more a contradiction. □

To sum up, we obtain as a byproduct of our reduction the following new characterization of $\frac{1}{2}$ -hyperbolic graphs:

Characterization 1. *A connected graph G is $\frac{1}{2}$ -hyperbolic if, and only if, every graph power G^i , $i \geq 1$, is C_4 -free, and the graph $G^{[2]}$ is C_4 -free.*

3.2.4 The reduction

Theorem 14. *There is a subcubic reduction from the $\frac{1}{2}$ -hyperbolic graph recognition problem to the problem of detecting an induced quadrangle in a graph.*

Proof. Let $G = (V, E)$ be a connected graph. Since there exists a linear-time algorithm to recognize 0-hyperbolic graphs⁶, we will assume for the proof $\delta(G) > 0$, or equivalently $\delta(G) \geq \frac{1}{2}$. Let us fix any $c = O(\log^{O(1)} n)$, $c \geq 1$, such that we can compute a c -factor approximation of the hyperbolicity in subcubic-time. Two possible choices for c are given in Lemma 4, and two others ones are discussed in Section 3.2.1.⁷ In the sequel, let $\delta_c(G)$ be a c -factor approximation of the hyperbolicity.

Recall that we have $\delta(G) \leq \delta_c(G) \leq c \cdot \delta(G)$ by the hypothesis. So, if $\delta_c(G) > \frac{c}{2}$, then we are done as the graph G is not $\frac{1}{2}$ -hyperbolic. Let us now assume that $\delta_c(G) \leq \frac{c}{2}$. By Corollary 6, every isometric cycle of G has length upper-bounded by $4\delta_c(G) + 3 \leq 2c + 3$, which is polylogarithmically upper-bounded.

We then compute all the graph powers G^i , for $1 \leq i \leq 2\delta_c(G) + 1$. This can be done in subcubic-time, by first computing the distance-matrix of G in $O(n^{\omega(1)} \log n)$ time (see [34]). Note that every G^i has to be C_4 -free by Lemma 8. If so, then by Lemma 9, there is no isometric cycle C_l of length $6 \leq l \leq 4\delta_c(G) + 3$. Consequently, the graph G satisfies Condition 1.

Finally, let us build $G^{[2]}$, which can also be done in subcubic-time using the distance-matrix of G . By Lemma 11, the graph $G^{[2]}$ has to be C_4 -free so that G is $\frac{1}{2}$ -hyperbolic. If it is indeed the case, then we have:

- $G^{[2]}$ and all of G^i , $1 \leq i \leq 2\delta_c(G)$ are C_4 -free, hence G satisfies Condition 2 by Lemma 12;
- $G^{[2]}$ and the square graph G^2 are both C_4 -free and so, G satisfies Condition 3 by Lemma 13.

Thus we can conclude by [1] that G is a $\frac{1}{2}$ -hyperbolic graph. □

Corollary 15. *There is a subcubic equivalence between the $\frac{1}{2}$ -hyperbolic graph recognition problem and the C_4 -free graph recognition problem.*

⁶Recall that a graph is 0-hyperbolic if, and only if, it is a *block-graph*, i.e. a graph whose biconnected components are complete subgraphs [3, 23].

⁷A careful reader will remark that the choice of c determines the maximum number of calls in the reduction to the algorithm for detecting an induced quadrangle.

4 Finding a quadrangle

We will conclude this paper with an improved algorithm for the C_4 -free graph recognition problem, that hence improves the best-known time complexity of both Problem 1 and Problem 2 by the subcubic equivalence of Corollary 15. While the algorithm proposed in [35] relies on transitive orientation, our algorithm merely reduces the whole Problem 2 to a fast rectangular matrix multiplication.

A quick reminding of the problem of matrix multiplication is given in Section 4.1, before we present our algorithm to detect an induced quadrangle in Section 4.2.

4.1 Fast rectangular matrix multiplication

The study of fast matrix multiplication mainly focuses on the $O(n^{\omega(1)})$ time complexity of multiplying two $n \times n$ matrices, also known as the square matrix product. Currently, $\omega(1)$ is known to be less than 2.3727 [36]. The *rectangular* matrix multiplication problem has received less attention, maybe because the product of an $n \times m$ matrix with an $m \times p$ matrix is known to be reducible to square matrix multiplications, yielding a time complexity in $O(q^{\omega(1)-2} \cdot \max\{mn, mp, np\})$, for $q = \min\{m, n, p\}$ [24].

On the other hand, there is evidence that faster methods for the rectangular matrix multiplication which do not rely on the square matrix product may exist. This is known to be the case even for truly practical improvements of the matrix multiplication such as the Strassen algorithm and its variations [25, 27]. As stated below, the conjecture is (numerically) true w.r.t. the best known algorithms for square and rectangular matrix multiplications.

Lemma 16 ([12, 24, 36]). *Given any rational $r \geq 1$, let M_1 be a $n \times n^r$ matrix, and M_2 be a $n^r \times n$ matrix.*

There exists a non-decreasing function $\omega : [1; +\infty[\rightarrow [2.3727; +\infty[$ such that multiplying M_1 with M_2 can be done in $O(n^{\omega(r)})$ time.

Furthermore, $\omega(2) \leq 3.333953$, and for every $r \geq 1$ we have $\omega(r) \leq r + \omega(1) - 1$.

Note that reducing the rectangular matrix product to square matrix multiplications would have only yielded $\omega(2) \leq 3.3727$.

A more efficient method is known for sparse matrices (*e.g.* see [39]).

4.2 An algorithm to count quadrangles in a graph

In this section, we essentially apply the local characterization of Lemma 2. There are two main steps of computation in our algorithm, that are described below.

Fact 17 ([34]). *Given a graph $G = (V, E)$, let $A = (A_{u,v})_{u,v \in V}$ be the adjacency matrix of G .*

For every pair $u, v \in V$, we have $A_{u,v}^2 = |N(u) \cap N(v)|$.

Hence, $d(u, v) = 2$ if, and only if, $u \neq v$, $A_{u,v} = 0$ and $A_{u,v}^2 \neq 0$.

Proof. By the definition of matrix multiplication, we have for every pair $u, v \in V$ that:

$$\begin{aligned} A_{u,v}^2 &= \sum_{x \in V} A_{u,x} A_{x,v} = \sum_{x \in V} \mathbb{I}_{\{x,u\} \in E} \mathbb{I}_{\{x,v\} \in E} \\ &= \sum_{x \in V} \mathbb{I}_{\{x \in N(u)\}} \mathbb{I}_{\{x \in N(v)\}} = \sum_{x \in V} \mathbb{I}_{\{x \in N(u) \cap N(v)\}} \\ &= |N(u) \cap N(v)|. \end{aligned}$$

Moreover, $d(u, v) = 2$ if, and only if, $u \neq v$, u and v are not adjacent in G (e.g. $A_{u,v} = 0$), and $N(u) \cap N(v) \neq \emptyset$. Clearly, we have that $N(u) \cap N(v) \neq \emptyset$ if, and only if, $A_{u,v}^2 \neq 0$. \square

Lemma 18. *Given a graph $G = (V, E)$, let $T = (T_{u,e})_{u \in V, e \in E}$ be such that $T_{u,e} = \mathbb{I}_{\{e \subseteq N(u)\}}$ for every $u \in V, e \in E$.*

For every pair $u, v \in V$, we have $TT_{u,v}^\top = |\{e \in E : e \subseteq N(u) \cap N(v)\}|$.

Proof. Similarly to the proof of Fact 17, we have by the definition of matrix multiplication that for every pair $u, v \in V$:

$$\begin{aligned} TT_{u,v}^\top &= \sum_{e \in E} T_{u,e} T_{e,v}^\top = \sum_{e \in E} T_{u,e} T_{v,e} = \sum_{e \in E} \mathbb{I}_{\{e \subseteq N(u)\}} \mathbb{I}_{\{e \subseteq N(v)\}} \\ &= \sum_{e \in E} \mathbb{I}_{\{e \subseteq N(u) \cap N(v)\}} = |\{e \in E : e \subseteq N(u) \cap N(v)\}|. \end{aligned}$$

\square

Combining Fact 17 and Lemma 18, we can now rely on Lemma 2 in order to detect, to count and to output induced quadrangles in the graph as follows.

Proposition 19. *Counting the number of induced quadrangles in a graph G , and returning an induced C_4 of G if any, can be done in $O(n^{\omega(\log_n m)}) = O(n^{3.333953})$ time.*

Proof. First, it is straightforward that we can compute the adjacency matrix A of G in quadratic-time. Using A , we can compute the matrix $T = (\mathbb{I}_{\{e \subseteq N(u)\}})_{u \in V, e \in E}$, hence the transpose matrix T^\top as well, as they are defined in Lemma 18, in $O(nm)$ time.

Let us now compute A^2 and TT^\top . This can be done, respectively, in $O(n^{\omega(1)}) = O(n^{2.3727})$ time and in $O(n^{\omega(\log_n m)}) = O(n^{\omega(2)}) = O(n^{3.333953})$ time by Lemma 16.

By Lemma 2, G is C_4 -free if, and only if, for every pair $u, v \in V$ of non-adjacent vertices, we have that $N(u) \cap N(v)$ is a clique, e.g. that:

$$|\{e \in E : e \subseteq N(u) \cap N(v)\}| = \frac{|N(u) \cap N(v)|(|N(u) \cap N(v)| - 1)}{2}.$$

This is equivalent to have that $TT_{u,v}^\top = \frac{A_{u,v}^2(A_{u,v}^2 - 1)}{2}$ by Fact 17 and Lemma 18. Hence, it can be checked with an enumeration of all the possible pairs $u, v \in V$, in quadratic-time.

We actually note that if there is some pair $u, v \in V$ of non-adjacent vertices such that $TT_{u,v}^\top < \frac{A_{u,v}^2(A_{u,v}^2 - 1)}{2}$, then there are exactly $\frac{A_{u,v}^2(A_{u,v}^2 - 1)}{2} - TT_{u,v}^\top$ induced quadrangles of G which contain this pair of vertices (that is one quadrangle for each of the missing edges in $G[N(u) \cap N(v)]$).

Consequently, there are exactly $\frac{1}{4} \sum_{u,v \in V} (1 - A_{u,v}) (\frac{A_{u,v}^2(A_{u,v}^2 - 1)}{2} - TT_{u,v}^\top)$ induced quadrangles in the graph, which can also be computed in quadratic-time, by an enumeration of all the possible pairs $u, v \in V$.

To conclude, note that if $TT_{u,v}^\top \neq \frac{A_{u,v}^2(A_{u,v}^2 - 1)}{2}$ for some pair $u, v \in V$ of non-adjacent vertices, then it is straightforward to compute an induced quadrangle of G containing u, v in quadratic-time. \square

We remind the reader that there is only *numerical* evidence that a fast rectangular matrix product can be computed faster than up to the reduction to fast square matrix multiplications. However, we want to highlight that, from a *theoretical* point of view, our algorithm for detecting an induced quadrangle is never slower than the algorithm of [35]. The dominant term for the complexity of our algorithm is indeed the fast rectangular matrix multiplication TT^\top (e.g.

Lemma 18), which can be computed in $O(mn^{\omega(1)-1})$ time in the worst case, using the reduction to fast square matrix products that we described earlier in Section 4.1. In comparison, the time complexity of the algorithm of [35] is $O(n^{\omega(1)+1})$.

Also, we emphasize that for graphs with few C_3 's, a speed-up for the computation of TT^\top can be achieved using the results from [39] for sparse matrix multiplication. Indeed, the number of non-zero elements in the matrix T is exactly $3t(G)$, where $t(G)$ denotes the number of C_3 in the graph.

5 Conclusion

In this work, we proved an interesting equivalence between the complexity of the purely *metric* problem of recognizing $\frac{1}{2}$ -hyperbolic graphs, and the purely *structural* problem of detecting an induced quadrangle in a graph. This shows a surprising gap in the complexity for recognizing graphs with small hyperbolicity, as in comparison there is a linear-time algorithm to decide whether a graph is 0-hyperbolic.

Our reduction being subcubic, it remains open whether $\frac{1}{2}$ -hyperbolic graphs can be recognized in linear-time, for some classes of graphs for which detecting an induced quadrangle is easy, like for instance planar graphs [19]. Also, it would be nice to extend our results to improve the time complexity of the problem of deciding if a graph is 1-hyperbolic. Note that this latter problem might be easier than the recognition of $\frac{1}{2}$ -hyperbolic graphs, as the true difficulty may only lie in the distinction between graphs with hyperbolicity *exactly* 1 and *exactly* $\frac{1}{2}$. Any recognition algorithm in $O(f(n))$ time for 1-hyperbolic graphs would furthermore yield a 4-factor approximation algorithm for the hyperbolicity that runs in $\tilde{O}(f(n) + n^{\omega(1)})$ time.

Acknowledgments

We would like to thank Frédéric Havet and Fatima Zahra Moataz for helpful comments on this work.

References

- [1] H.-J. Bandelt and V. Chepoi. 1-hyperbolic graphs. *SIAM Journal on Discrete Mathematics*, 16(2):323–334, 2003.
- [2] H.-J. Bandelt and V. Chepoi. Metric graph theory and geometry: a survey. *Contemporary Mathematics*, 453:49–86, 2008.
- [3] H.-J. Bandelt and H.M. Mulder. Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B*, 41(2):182–208, 1986.
- [4] J. Baras. Hyperbolic embedding to the rescue in communication and social networks. In *Bell Labs-NIST Workshop on Large-Scale Networks*, 2013.
- [5] S. Bermudo, J.M. Rodríguez, J.M. Sigarreta, and J.-M. Vilaire. Gromov hyperbolic graphs. *Discrete Mathematics*, 313(15):1575–1585, 2013.
- [6] M. Boguna, F. Papadopoulos, and D. Krioukov. Sustaining the Internet with Hyperbolic Mapping. *Nature Communications*, 1(62), Oct 2010.

-
- [7] J. A. Bondy and U. Murty. *Graph theory with applications*, volume 290. Macmillan London, 1976.
- [8] W. Carballosa, J.M. Rodríguez, J.M. Sigarreta, and M. Villeta. Gromov hyperbolicity of line graphs. *The Electronic Journal of Combinatorics*, 18(P210):1–18, 2011.
- [9] J. Chalopin, V. Chepoi, P. Pappasoglou, and T. Pecatte. Cop and robber game and hyperbolicity. *CoRR*, abs/1308.3987, 2013.
- [10] V. Chepoi and F. Dragan. A note on distance approximating trees in graphs. *European Journal of Combinatorics*, 21(6):761–766, 2000.
- [11] N. Cohen, D. Coudert, and A. Lancin. Exact and approximate algorithms for computing the hyperbolicity of large-scale graphs. Rapport de recherche RR-8074, INRIA, September 2012.
- [12] D. Coppersmith. Rectangular matrix multiplication revisited. *Journal of Complexity*, 13(1):42–49, 1997.
- [13] P. de La Harpe and E. Ghys. *Sur les groupes hyperboliques d’après Mikhael Gromov*, volume 83. Progress in Mathematics, 1990.
- [14] R. Diestel. *Graph theory, graduate texts in mathematics*, vol. 173, 1997.
- [15] D. Dor, S. Halperin, and U. Zwick. All-pairs almost shortest paths. *SIAM Journal on Computing*, 29(5):1740–1759, 2000.
- [16] F.F. Dragan. Tree-like structures in graphs: A metric point of view. In *Graph-Theoretic Concepts in Computer Science*, pages 1–4. Springer, 2013.
- [17] F.F. Dragan and E. Köhler. An approximation algorithm for the tree t-spanner problem on unweighted graphs via generalized chordal graphs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 171–183. Springer, 2011.
- [18] A. Dress, K. Huber, J. Koolen, V. Moulton, and A. Spillner. *Basic Phylogenetic Combinatorics*. Cambridge University Press, 2012.
- [19] D. Eppstein. Subgraph isomorphism in planar graphs and related problems. In *Proceedings of the sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 632–640. Society for Industrial and Applied Mathematics, 1995.
- [20] H. Fournier, A. Ismail, and A. Vigneron. Computing the Gromov hyperbolicity of a discrete metric space. Technical Report arXiv:1210.3323, ArXiv, October 2012.
- [21] C. Gavoille, D. Peleg, S. Pérennes, and R. Raz. Distance labeling in graphs. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 210–219. Society for Industrial and Applied Mathematics, 2001.
- [22] M. Gromov. Hyperbolic groups. *Essays in Group Theory*, 8:75–263, 1987.
- [23] E. Howorka. On metric properties of certain clique graphs. *Journal of Combinatorial Theory, Series B*, 27(1):67–74, 1979.
- [24] X. Huang and V.Y. Pan. Fast rectangular matrix multiplication and applications. *Journal of complexity*, 14(2):257–299, 1998.

-
- [25] S. Huss-Lederman, E.M. Jacobson, J.R. Johnson, A. Tsao, and T. Turnbull. Implementation of strassen’s algorithm for matrix multiplication. In *Supercomputing, 1996. Proceedings of the 1996 ACM/IEEE Conference on*, pages 32–32. IEEE, 1996.
- [26] E. Jonckheere and P. Lohsoonthorn. A hyperbolic geometric approach to multipath routing. *Conference on Control and Automation*, 2002.
- [27] P.A. Knight. Fast rectangular matrix multiplication and QR decomposition. *Linear algebra and its applications*, 221:69–81, 1995.
- [28] A. Kosowski, B. Li, N. Nisse, and K. Suchan. k -chordal graphs: From cops and robber to compact routing via treewidth. In *Automata, Languages, and Programming*, pages 610–622. Springer, 2012.
- [29] R. Laskar and D. Shier. On powers and centers of chordal graphs. *Discrete Applied Mathematics*, 6(2):139–147, 1983.
- [30] D. Lokshtanov. Finding the longest isometric cycle in a graph. *Discrete Applied Mathematics*, 157(12):2670–2674, 2009.
- [31] J. Michel, J.M. Rodríguez, J.M. Sigarreta, and M. Villeta. Hyperbolicity and parameters of graphs. *Ars Combinatoria*, 100:43–63, 2011.
- [32] L. Roditty and V. Vassilevska Williams. Minimum weight cycles and triangles: Equivalences and algorithms. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 180–189. IEEE, 2011.
- [33] J.M. Rodríguez and J.M. Sigarreta. Bounds on Gromov hyperbolicity constant in graphs. In *Proceedings Indian Acad. Sci. (Mathematical Sciences)*, volume 122, pages 53–65, 2012.
- [34] R. Seidel. On the all-pairs-shortest-path problem in unweighted undirected graphs. *Journal of Computer and System Sciences*, 51(3):400–403, 1995.
- [35] J.P. Spinrad. Finding large holes. *Information Processing Letters*, 39(4):227–229, 1991.
- [36] V. Vassilevska Williams. Multiplying matrices faster than coppersmith-winograd. In *Proceedings of the 44th symposium on Theory of Computing*, pages 887–898. ACM, 2012.
- [37] V. Vassilevska Williams and R. Williams. Subcubic equivalences between path, matrix and triangle problems. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 645–654. IEEE, 2010.
- [38] Y. Wu and C. Zhang. Hyperbolicity and chordality of a graph. *Electr. J. Comb.*, 18(1), 2011.
- [39] R. Yuster and U. Zwick. Fast sparse matrix multiplication. *ACM Transactions on Algorithms (TALG)*, 1(1):2–13, 2005.



**RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE**

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399