Isometric path complexity of graphs[*]

Dibyayan Chakraborty[†]   Jérémie Chalopin[‡]   Florent Foucaud[§]   Yann Vaxès[¶]

August 22, 2025

Abstract

A set S of isometric paths of a graph G is "v-rooted", where v is a vertex of G, if v is one of the endpoints of all the isometric paths in S. The *isometric path complexity* of a graph G, denoted by $ipco\,(G)$, is the minimum integer k such that there exists a vertex $v \in V(G)$ satisfying the following property: the vertices of any single isometric path P of G can be covered by k many v-rooted isometric paths.

First, we provide an $O(n^2 m)$-time algorithm to compute the isometric path complexity of a graph with n vertices and m edges. Then we show that the isometric path complexity remains bounded for graphs in three seemingly unrelated graph classes, namely, *hyperbolic graphs*, *(theta, prism, pyramid)-free graphs*, and *outerstring graphs*. There is a direct algorithmic consequence of having small isometric path complexity. Specifically, we show that if the isometric path complexity of a graph G is bounded by a constant, then there exists a polynomial-time constant-factor approximation algorithm for ISOMETRIC PATH COVER, whose objective is to cover all vertices of a graph with a minimum number of isometric paths. This applies to all the above graph classes.

Keywords: Shortest paths, Isometric path complexity, Hyperbolic graphs, Truemper Configurations, Outerstring graphs, Isometric Path Cover

1   Introduction

Finding properties of graph classes that can be exploited to develop efficient algorithms is a popular direction of research in graph theory. It has motivated researchers to define varieties of graph parameters, study their combinatorial properties, and develop efficient algorithms based on them. Identifying graph classes where these parameters are bounded is also fundamental to establish their relevance. Some of them are bounded for seemingly unrelated classes, e.g. *mim-width* [4, 25] or *twin-width* [6], showing unexpected structural similarities between these classes. In this paper, we study the *isometric path complexity* that is initially motivated by an algorithmic application. We show that it can be computed in polynomial time and that it is bounded on three graph classes studied in different research areas in graph theory.

The isometric path cover problem. Recently, Chakraborty et al. [10] introduced a parameter called *isometric path antichain cover number* of a graph G, denoted as $ipacc\,(G)$ (see Definition 6) in the context of an algorithmic problem known as the ISOMETRIC PATH COVER.

A path is *isometric* if it is a shortest path between its endpoints[1]. An *isometric path cover* of a graph G is a set of isometric paths such that each vertex of G belongs to at least one of the paths. The *isometric path number* of G is the smallest size of an isometric path cover of G. Given a graph G and an
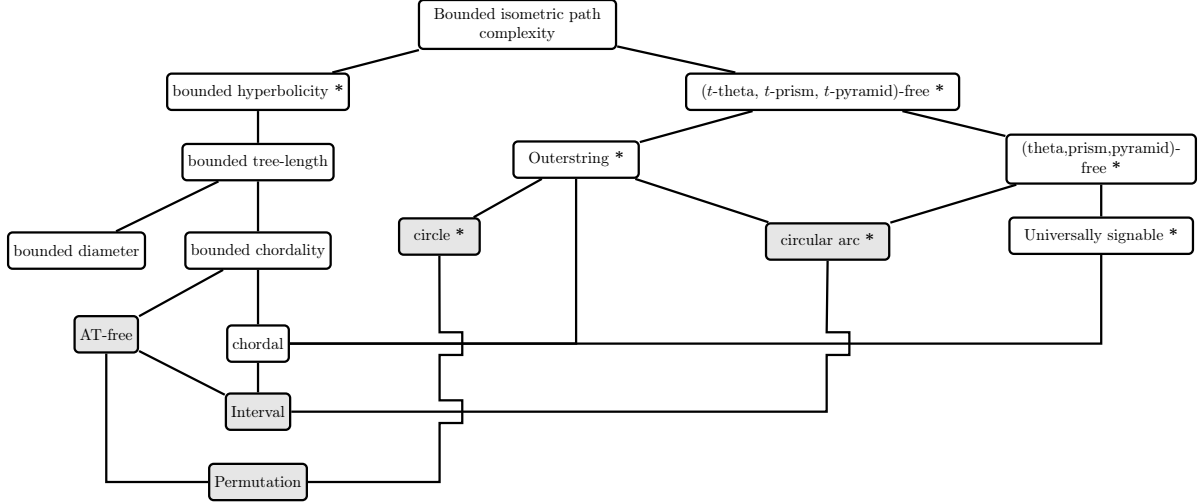
Figure 1: Inclusion diagram for graph classes. If a class A has an upward path to class B, then A is included in B. Constant bounds for the isometric path complexity on graph classes marked with * are contributions of this paper.

integer k, the objective of ISOMETRIC PATH COVER is to decide if there exists an isometric path cover of cardinality at most k.

ISOMETRIC PATH COVER has been introduced and studied in the context of pursuit-evasion games [2, 3]. However, until recently the algorithmic aspects of ISOMETRIC PATH COVER remained unexplored. After proving that ISOMETRIC PATH COVER remains NP-hard on *chordal graphs* (graphs without any induced cycle of length at least 4), Chakraborty et al. [10] provided constant-factor approximation algorithms for many graph classes, including *interval graphs*, chordal graphs, and more generally, graphs with bounded *treelength*. The authors proved the bound on the approximation ratio by showing that the isometric path antichain cover number of the above graph classes is bounded by a constant. Specifically, they proved that (*i*) when $ipacc(G)$ is bounded by a constant, ISOMETRIC PATH COVER admits a constant-factor approximation algorithm on G; and (*ii*) the isometric path antichain cover number of graphs with bounded *treelength* is bounded.

Our objectives. The objective of this paper is three-fold: **(A)** provide a more intuitive definition of isometric path antichain cover number; **(B)** provide a polynomial-time algorithm to compute $ipacc(G)$; and **(C)** prove that it remains bounded for seemingly unrelated graph classes. Along the way, we also extend the horizon of approximability of ISOMETRIC PATH COVER. To achieve **(A)** we introduce the following new metric graph parameter, that we will show to be always equal to the isometric path antichain cover number, and whose definition is simpler.

Definition 1. *Given a graph G and a vertex v of G, a set S of isometric paths of G is v-rooted if v is one of the endpoints of all the isometric paths in S. The* isometric path complexity *of a graph G, denoted by $ipco(G)$, is the minimum integer k such that there exists a vertex $v \in V(G)$ satisfying the following property: the vertices of any isometric path P of G can be covered by k many v-rooted isometric paths.*

A consequence of Dilworth's theorem [21] is that for any graph G, $ipacc(G) = ipco(G)$ (see Lemma 7). We will give a polynomial-time algorithm to compute $ipco(G)$, and therefore $ipacc(G)$ for an arbitrary undirected graph G. This achieves **(B)**.

Finally, to achieve **(C)**, we consider the following three seemingly unrelated graph classes, namely, *δ-hyperbolic graphs, (theta, prism, pyramid)-free graphs* and *outerstring* graphs, and show that their isometric path complexity is bounded by a constant.

δ-hyperbolic graphs: A graph G is said to be *δ-hyperbolic* [24] if for any four vertices u, v, x, y, the two larger of the three distance sums $\mathsf{d}(u,v) + \mathsf{d}(x,y)$, $\mathsf{d}(u,x) + \mathsf{d}(v,y)$ and $\mathsf{d}(u,y) + \mathsf{d}(v,x)$ differ by at most 2δ. A graph class $\mathcal{G}$ is *hyperbolic* if there exists a constant δ such that every graph $G \in \mathcal{G}$ is δ-hyperbolic. This parameter comes from geometric group theory and was first introduced by Gromov [24] in order to study groups via their *Cayley graphs*. The hyperbolicity of a tree is 0, and in general, the hyperbolicity

measures how much the distance function of a graph deviates from a tree metric. Many structurally defined graph classes like chordal graphs, *cocomparability* graphs [16], *asteroidal-triple free* graphs [17], graphs with bounded *chordality* or *treelength* are hyperbolic [11, 27]. Moreover, hyperbolicity has been found to capture important properties of several large practical graphs such as the Internet graph [30] or database relation graphs [34]. Due to its importance in discrete mathematics, algorithms, *metric graph theory*, researchers have studied various algorithmic aspects of hyperbolic graphs [11, 18, 13, 19]. Note that graphs with diameter 2 are hyperbolic, which may contain any graph as an induced subgraph.

(theta, prism, pyramid)-free graphs: A *theta* is a graph made of three vertex-disjoint induced paths $P_1 = a \ldots b$, $P_2 = a \ldots b$, $P_3 = a \ldots b$ of lengths at least 2, and such that no edges exist between the paths except the three edges incident to a and the three edges incident to b. A *pyramid* is a graph made of three induced paths $P_1 = a \ldots b_1$, $P_2 = a \ldots b_2$, $P_3 = a \ldots b_3$, two of which have lengths at least 2, vertex-disjoint except at a, and such that $b_1 b_2 b_3$ is a triangle and no edges exist between the paths except those of the triangle and the three edges incident to a. A *prism* is a graph made of three vertex-disjoint induced paths $P_1 = a_1 \ldots b_1$, $P_2 = a_2 \ldots b_2$, $P_3 = a_3 \ldots b_3$ of lengths at least 1, such that $a_1 a_2 a_3$ and $b_1 b_2 b_3$ are triangles and no edges exist between the paths except those of the two triangles. A graph G is *(theta, pyramid, prism)-free* if G does not contain any induced subgraph isomorphic to a theta, pyramid or prism. A graph is a 3-*path configuration* if it is a theta, pyramid or prism. The study of 3-path configurations dates back to the works of Watkins and Meisner [35] in 1967 and plays "special roles" in the proof of the celebrated *Strong Perfect Graph Theorem* [14]. Important graph classes like chordal graphs, *circular arc* graphs, *universally-signable* graphs [15] exclude all 3-path configurations. Popular graph classes like *perfect* graphs, *even hole*-free graphs exclude some (but not all) of the 3-path configurations. Note that, (theta, prism, pyramid)-free graphs are not hyperbolic. To see this, consider a cycle C of order n. Clearly, C excludes all 3-path configurations and has hyperbolicity $\Omega(n)$.

Outerstring graphs: A set S of simple curves on the plane is *grounded* if there exists a horizontal line containing one endpoint of each of the curves in S. A graph G is an *outerstring* graph if there is a collection C of grounded simple curves and a bijection between $V(G)$ and C such that two curves in S intersect if and only if the corresponding vertices are adjacent in G. The term "outerstring graph" was first used in the early 90's [28] in the context of studying intersection graphs of simple curves on the plane. Many well-known graph classes like chordal graphs, *circular arc* graphs [23], *circle* graphs (intersection graphs of chords of a circle [20]), and cocomparability graphs [16] are also outerstring graphs and thus, motivated researchers from the *geometric graph theory* and *computational geometry* communities to study algorithmic and structural aspects of outerstring graphs and its subclasses [5, 7, 8, 26, 29]. Note that in general, outerstring graphs may contain a prism, pyramid or theta as an induced subgraph. Moreover, cycles of arbitrary order are outerstring graphs, implying that outerstring graphs are not hyperbolic.

It is clear from the above discussion that the classes of hyperbolic graphs, (theta, prism, pyramid)-free graphs, and outerstring graphs are pairwise incomparable (with respect to the containment relationship). We show that the isometric path complexities of all the above graph classes are small.

Our contributions

The main technical contribution of this paper are as follows. First we prove that the isometric path complexity can be computed in polynomial time.

Theorem 2. *Given a graph G with n vertices and m edges, it is possible to compute $ipco(G)$ in $O(n^2 m)$ time.*

Then we show that the isometric path complexity remains bounded on hyperbolic graphs, (theta, pyramid, prism)-free graphs, and outerstring graphs. Specifically, we prove the following theorem.

Theorem 3. *Let G be a graph.*

(a) *If the hyperbolicity of G is at most δ, then $ipco(G) \leq 4\delta + 3$.*

(b) *If G is a (theta, pyramid, prism)-free graph, then $ipco(G) \leq 71$.*

(c) *If G is an outerstring graph, then $ipco(G) \leq 95$.*

3

To the best of our knowledge, the isometric path complexity being bounded (by constant(s)) is the only known non-trivial property shared by any two or all three of these graph classes. Theorem 3 shows that isometric path complexity (equivalently isometric path antichain cover number), as recently introduced graph parameters, are general enough to unite these three graph classes by their metric properties. We hope that this definition will be useful for the field of metric graph theory, for example by enabling us to study (theta,prism,pyramid)-free graphs and outerstring graphs from the perspective of metric graph theory.

We provide a unified proof for Theorem 3(b) and 3(c) by proving that the isometric path complexity of *(t-theta, t-pyramid, t-prism)*-free graphs [32] (see Section 4 for a definition) is bounded by a linear function of t. Due to the above theorems, we also have as corollaries that there is a polynomial-time approximation algorithm for ISOMETRIC PATH COVER with approximation ratio (a) $4\delta+3$ on δ-hyperbolic graphs, (b) 73 on (theta, prism, pyramid)-free graphs, (c) 95 on outerstring graphs, and (d) $8t + 63$ on (t-theta, t-pyramid, t-prism)-free graphs.

To contrast with Theorem 3, we construct highly structured graphs with small *tree-width* and large isometric path complexity. A *wheel* consists of an induced cycle C of order at least 4 and a vertex $w \notin V(C)$ adjacent to at least three vertices of C. The 3-path configurations introduced earlier and the wheel together are called *Truemper configurations* [33] and they are important objects of study in structural and algorithmic graph theory [1, 22].

Theorem 4. *For every $k \geq 1$,*
  (a) *there exists a (pyramid, prism, wheel)-free graph G with tree-width 2, hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$ and $ipco(G) \geq k$;*

  (b) *there exists a (theta, prism, wheel)-free planar graph G with tree-width at most 3, hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$ and $ipco(G) \geq k$;*

  (c) *there exists a (theta, pyramid, wheel)-free planar graph G with hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$ and $ipco(G) \geq k$;*

  (d) *there exists a (prism, pyramid, wheel)-free planar bipartite graph G such that $|V(G)|$ is $O(k^2)$, G has an isometric path cover of size $3k+1$ and any v-rooted isometric path cover of G has cardinality at least k^2 for any $v \in V(G)$.*

Theorem 4(d) proves that the approximation algorithm for ISOMETRIC PATH COVER proposed by Chakraborty et al. [10] cannot provide a $o(\sqrt{n})$ approximation ratio (even if the inputs are restricted to planar bipartite graphs of order n). Note that previous known lower bound (stated in [10]) was $o(\sqrt{\log n})$.

Organisation. In Section 2, we recall some definitions and some results. In Section 3, we present an algorithm to compute the isometric path complexity of a graph and prove Theorem 2. In Section 4, we prove Theorem 3. In Section 5, we prove Theorem 4. We conclude in Section 6.

2 Definitions and preliminary observations

In this section, we recall some definitions and some related observations. A sequence of distinct vertices forms a *path P* if any two consecutive vertices are adjacent. Whenever we fix a path P of G, we shall refer to the subgraph formed by the edges between the consecutive vertices of P. The *length* of a path P, denoted by $|P|$, is the number of its vertices minus one. A path is *induced* if there are no graph edges joining non-consecutive vertices. A path is *isometric* if it is a shortest path between its endpoints. For two vertices u, v of a graph G, $d(u, v)$ denotes the length of an isometric path between u and v.

For a path P and a vertex $u \in V(P)$, we will often use the phrase "(u, w)-subpath" of P to denote a subpath that satisfies certain properties (which will be clear from the context), and has u, w as endpoints, where w is the endpoint distinct from u. We note that w and the subpath are getting defined simultaneously, but for the sake of readability, we use the above notation.

In a directed graph, a *directed path* is a path in which all arcs are oriented in the same direction. For a path P of a graph G between two vertices u and v, the vertices $V(P) \setminus \{u, v\}$ are *internal vertices* of P. A path between two vertices u and v is called a *(u, v)-path*. Similarly, we have the notions of *isometric (u, v)-path* and *induced (u, v)-path*. The interval $I(u, v)$ between two vertices u and v consists of all vertices that belong to an isometric (u, v)-path. For a vertex r of G and a set S of vertices of

G, the *distance of S from r*, denoted as $\mathsf{d}(r, S)$, is the minimum of the distance between any vertex of S and r. For a subgraph H of G, the *distance of H w.r.t. r* is $\mathsf{d}(r, V(H))$. Formally, we have $\mathsf{d}(r, S) = \min\{\mathsf{d}(r, v) : v \in S\}$ and $\mathsf{d}(r, H) = \mathsf{d}(r, V(H))$.

For a graph G and a vertex $r \in V(G)$, consider the following operations on G. First, remove all edges xy from G such that $\mathsf{d}(r, x) = \mathsf{d}(r, y)$. Let G'_r be the resulting graph. Then, for each edge $e = xy \in E(G'_r)$ with $\mathsf{d}(r, x) = \mathsf{d}(r, y) - 1$, orient e from y to x. Let $\overrightarrow{G_r}$ be the directed acyclic graph formed after applying the above operation on G'. Note that this digraph can easily be computed in linear time using a Breadth-First Search (BFS) traversal with starting vertex r.

The known approximation algorithm for ISOMETRIC PATH COVER from [10] can now be stated as follows: (i) For each vertex $r \in V(G)$, compute $\overrightarrow{G_r}$ and find a minimum path cover $\mathcal{C}_r$ of $\overrightarrow{G_r}$, and then (ii) report a $\mathcal{C}_r$ with minimum cardinality. The following definition is inspired by the terminology of posets (as the graph $\overrightarrow{G_r}$ can be seen as the Hasse diagram of a poset) and will be useful to analyze the above algorithm.

Definition 5. *For a graph G and a vertex $r \in V(G)$, two vertices $x, y \in V(G)$ are* antichain vertices *w.r.t r, if there are no directed paths from x to y or from y to x in $\overrightarrow{G_r}$.*

For a graph G and a vertex $r \in V(G)$, a set X of vertices of G is an *antichain set* if any two vertices in X are antichain vertices w.r.t r.

Definition 6 ([10])**.** *Let r be a vertex of a graph G. For a subgraph H, $A_r(H)$ shall denote the maximum antichain set of H in $\overrightarrow{G_r}$. The* isometric path antichain cover number *of $\overrightarrow{G_r}$, denoted by $ipacc\left(\overrightarrow{G_r}\right)$, is defined as follows:*

$$ipacc\left(\overrightarrow{G_r}\right) = \max\left\{|A_r(P)| : \ P \text{ is an isometric path in } G\right\}.$$

The isometric path antichain cover number *of graph G, denoted as $ipacc(G)$, is defined as the minimum over all possible antichain covers of its associated directed acyclic graphs:*

$$ipacc(G) = \min\left\{ipacc\left(\overrightarrow{G_r}\right) : r \in V(G)\right\}.$$

For technical purposes, we also introduce the following definition. For a graph G and a vertex r of G, let $ipco\left(\overrightarrow{G_r}\right)$ denote the minimum integer k such that any isometric path P of G can be covered by k r-rooted isometric paths (The notation reflects that it is a dual notion of $ipacc\left(\overrightarrow{G_r}\right)$). Using Dilworth's Theorem [21] we prove the following important lemma.

Lemma 7. *For any graph G and vertex r, $ipco\left(\overrightarrow{G_r}\right) = ipacc\left(\overrightarrow{G_r}\right)$. Therefore, $ipco(G) = ipacc(G)$.*

Proof. Let r be a vertex of G such that any isometric path of G can be covered by $ipco\left(\overrightarrow{G_r}\right)$ r-rooted isometric paths. Let P be an arbitrary isometric path of G. Since two vertices of an antichain of $\overrightarrow{G_r}$ cannot be covered by a single r-rooted path and P is covered by $ipco\left(\overrightarrow{G_r}\right)$ many r-rooted paths, we deduce $|A_r(P)| \leq ipco\left(\overrightarrow{G_r}\right)$. This is true for any isometric path P of G. Hence, $ipacc\left(\overrightarrow{G_r}\right) \leq ipco\left(\overrightarrow{G_r}\right)$. Conversely, consider a vertex $r \in V(G)$. By definition of $ipco\left(\overrightarrow{G_r}\right)$, there is an isometric path P that cannot be covered by $\left(ipco\left(\overrightarrow{G_r}\right) - 1\right)$ r-rooted isometric paths. By Dilworth's theorem [21], P contains an antichain of $\overrightarrow{G_r}$ of size $ipco\left(\overrightarrow{G_r}\right)$. Hence $|A_r(P)| \geq ipco\left(\overrightarrow{G_r}\right)$ and $ipacc\left(\overrightarrow{G_r}\right) \geq ipco\left(\overrightarrow{G_r}\right)$. The second part of the lemma follows immediately. $\square$

We also recall the following theorem and proposition from [10].

Theorem 8 ([10])**.** *For a graph G, if $ipacc(G) \leq c$, then* ISOMETRIC PATH COVER *admits a polynomial-time c-approximation algorithm on G.*

Proposition 9 ([10]). *Let G be a graph and r, an arbitrary vertex of G. Consider the directed acyclic graph $\overrightarrow{G_r}$, and let P be an isometric path between two vertices x and y in G. Then $|P| \geq |\mathsf{d}(r,x) - \mathsf{d}(r,y)| + |A_r(P)| - 1$.*

Proof. Orient the edges of P from y to x in G. First, observe that $\overrightarrow{P}$ must contain a set E_1 of oriented edges such that $|E_1| = |\mathsf{d}(r,y) - \mathsf{d}(r,x)|$ and for any $\overrightarrow{ab} \in E_1$, $\mathsf{d}(r,a) = \mathsf{d}(r,b) + 1$. Let the vertices of the largest antichain set of P in $\overrightarrow{G_r}$, i.e., $A_r(P)$, be ordered as $a_1, a_2, \ldots, a_t$ according to their occurrence while traversing P from y to x. For $i \in [2,t]$, let P_i be the subpath of P between a_{i-1} and a_i. Observe that for any $i \in [2,t]$, since a_i and a_{i-1} are antichain vertices, there must exist an oriented edge $\overrightarrow{b_i c_i} \in E(\overrightarrow{P_i})$ such that either $\mathsf{d}(r,b_i) = \mathsf{d}(r,c_i)$ or $\mathsf{d}(r,b_i) = \mathsf{d}(r,c_i) - 1$. Let $E_2 = \{b_i c_i\}_{i \in [2,t]}$. Observe that $E_1 \cap E_2 = \emptyset$ and therefore $|P| \geq |E_1| + |E_2| = |\mathsf{d}(r,y) - \mathsf{d}(r,x)| + |A_r(P)| - 1$. $\qquad\square$

3   Proof of Theorem 2

In this section we provide a polynomial-time algorithm to compute the isometric path complexity of a graph. Let G be a graph. In the following lemma, we provide a necessary and sufficient condition for two vertices of an isometric path to be covered by the same isometric r-rooted path in $\overrightarrow{G_r}$ for some vertex $r \in V(G)$.

Lemma 10. *Let r be vertex of G. If $P = (u = v_0, \ldots, v_k = v)$ is an isometric (u,v)-path with $\mathsf{d}(r,u) \leq \mathsf{d}(r,v)$ then there exists an isometric r-rooted path containing u,v in $\overrightarrow{G_r}(P)$ if and only if $\mathsf{d}(v_{i+1},r) = \mathsf{d}(v_i,r) + 1$ for all $i \in \{0, \ldots, k-1\}$.*

Proof. If $\mathsf{d}(v_{i+1},r) = \mathsf{d}(v_i,r) + 1$ for every $i \in \{0, \ldots, k-1\}$ then the path obtained by concatenating an isometric (r,u)-path and the path P is an isometric r-rooted (r,v)-path containing u,v in $\overrightarrow{G_r}(P)$. Now suppose that there exists an isometric r-rooted path containing u,v in $\overrightarrow{G_r}(P)$, i.e., $\mathsf{d}(r,v) - \mathsf{d}(r,u) = \mathsf{d}(u,v)$. Then, along any path from u to v, we need to traverse at least $\mathsf{d}(u,v)$ edges increasing the distance to r. Since P is an isometric (u,v)-path, it contains exactly $\mathsf{d}(u,v)$ edges. Hence, $\mathsf{d}(r,v_{i+1}) = \mathsf{d}(r,v_i) + 1$ for every $i \in \{0, \ldots, k-1\}$. $\qquad\square$

3.1   Notations and preliminary observations

We now introduce some notations that will be used to describe the algorithm and prove its correctness. Consider three vertices r, x, v of G such that $x \neq v$. Let $\mathcal{P}^r_{\searrow}(x,v)$ denote the set of all isometric (x,v)-paths P containing a vertex u that is adjacent to v and satisfies $\mathsf{d}(r,u) = \mathsf{d}(r,v) - 1$. Analogously, let $\mathcal{P}^r_{\rightarrow}(x,v)$ denote the set of all isometric (x,v)-paths P containing a vertex u that is adjacent to v and satisfies $\mathsf{d}(r,u) = \mathsf{d}(r,v)$ and let $\mathcal{P}^r_{\nearrow}(x,v)$ denote the set of all isometric (x,v)-paths P containing a vertex u that is adjacent to v and satisfies $\mathsf{d}(r,u) = \mathsf{d}(r,v) + 1$. Observe that the set of isometric (x,v)-paths is precisely $\mathcal{P}^r_{\searrow}(x,v) \cup \mathcal{P}^r_{\rightarrow}(x,v) \cup \mathcal{P}^r_{\nearrow}(x,v)$ and that some of these sets may be empty.

Given a path P, we denote by $|S_r(P)|$ the minimum size of a set of isometric r-rooted paths covering the vertices of P. We denote by $\gamma^r_{\searrow}(x,v)$ and $\beta^r_{\searrow}(x,v)$ respectively the maximum of $|S_r(P)|$ and $|S_r(P - \{v\})|$ over all paths $P \in \mathcal{P}^r_{\searrow}(x,v)$. More formally,

$$\gamma^r_{\searrow}(x,v) = \max\{|S_r(P)| : P \in \mathcal{P}^r_{\searrow}(x,v)\},$$
$$\beta^r_{\searrow}(x,v) = \max\{|S_r(P - \{v\})| : P \in \mathcal{P}^r_{\searrow}(x,v)\}.$$

Note that if $\mathcal{P}^r_{\searrow}(x,v)$ is empty, we have $\gamma^r_{\searrow}(x,v) = \beta^r_{\searrow}(x,v) = 0$. We define similarly $\gamma^r_{\nearrow}(x,v)$, $\beta^r_{\nearrow}(x,v)$, and $\gamma^r_{\rightarrow}(x,v)$:

$$\gamma^r_{\nearrow}(x,v) = \max\{|S_r(P)| : P \in \mathcal{P}^r_{\nearrow}(x,v)\},$$
$$\beta^r_{\nearrow}(x,v) = \max\{|S_r(P - \{v\})| : P \in \mathcal{P}^r_{\nearrow}(x,v)\},$$
$$\gamma^r_{\rightarrow}(x,v) = \max\{|S_r(P)| : P \in \mathcal{P}^r_{\rightarrow}(x,v)\}.$$

Finally, let $\gamma^r(x,v) = \max\left\{\gamma^r_{\searrow}(x,v), \gamma^r_{\rightarrow}(x,v), \gamma^r_{\nearrow}(x,v)\right\}$ be the maximum of $|S_r(P)|$ over all isometric (x,v)-paths P. In our algorithm, we will need also to consider the case where $v = x$ as an initial case. For practical reasons, we let $\gamma^r(x,x) = \gamma^r_{\searrow}(x,x) = \gamma^r_{\rightarrow}(x,x) = \gamma^r_{\nearrow}(x,x) = 1$ and $\beta^r_{\searrow}(x,x) = \beta^r_{\nearrow}(x,x) = 0$. Based on the above notations and Lemma 7, we have the following observation.

Observation 11. *For any graph G and any vertex r of G, we have $ipco\left(\overrightarrow{G_r}\right) = ipacc\left(\overrightarrow{G_r}\right) = \max_{x,v} \gamma^r(x,v)$ and $ipco\left(G\right) = ipacc\left(G\right) = \min_r \max_{x,v} \gamma^r(x,v)$.*

Observation 11 implies that to compute the isometric path complexity of a graph it is enough to compute the parameter $\gamma^r(x,v)$ for all $r,x,v \in V(G)$ in polynomial time. In the next section, we focus on achieving this goal without computing explicitly any of the sets $\mathcal{P}^r_\searrow(x,v)$, $\mathcal{P}^r_\rightarrow(x,v)$ or $\mathcal{P}^r_\nearrow(x,v)$. (Note that the size of these sets could be exponential in the number of vertices of the graph).

3.2 An algorithm to compute $\gamma^r(x,v)$

Throughout this section, let r and x be two fixed vertices of G. We shall call r as the "root" and x as the "source" vertex. The objective of this section is to compute the parameter $\gamma^r(x,v)$ for all vertices $v \in V(G)$.

In the sequel, since we always refer to a fixed root r and source x, we omit r and x and use the shorthand $\gamma(v)$ for $\gamma^r(x,v)$. We do the same with the notations $\gamma_\nearrow(v)$, $\gamma_\rightarrow(v)$, $\gamma_\searrow(v)$, $\beta_\nearrow(v)$, and $\beta_\searrow(v)$ that also refer to fixed vertices r and x In the following lemmas, we shall provide explicit (recursive) formulas to compute $\gamma_\nearrow(v)$, $\gamma_\rightarrow(v)$, $\gamma_\searrow(v)$, $\beta_\nearrow(v)$, and $\beta_\searrow(v)$. Using these formulas, we will show how to compute $\gamma(v)$ for all $v \in V(G)$ in a total of $O(|E(G)|)$-time.

Observation 12. *If r is the root vertex, x the source vertex, and v is distinct from x, then*

$$\beta_\searrow(v) = \max\{\gamma(u) : u \in I(x,v) \cap N(v);\ \mathsf{d}(r,u) = \mathsf{d}(r,v) - 1\},$$
$$\beta_\nearrow(v) = \max\{\gamma(u) : u \in I(x,v) \cap N(v);\ \mathsf{d}(r,u) = \mathsf{d}(r,v) + 1\}.$$

Lemma 13. *If r is the root vertex, x the source vertex, and v is distinct from x, then $\gamma_\rightarrow(v) = 0$ if $\mathcal{P}^r_\rightarrow(x,v) = \emptyset$ and $\gamma_\rightarrow(v) = \max\{1 + \gamma(u) : u \in I(x,v) \cap N(v);\ \mathsf{d}(r,u) = \mathsf{d}(r,v)\}$ otherwise.*

Proof. Observe that $\mathcal{P}^r_\rightarrow(x,v)$ is empty if and only if there is no vertex $u \in I(x,v) \cap N(v)$ such that $\mathsf{d}(r,u) = \mathsf{d}(r,v)$. If $\mathcal{P}^r_\rightarrow(x,v)$ is empty, then $\gamma_\rightarrow(v) = 0$ and we are done.

Suppose now that $\mathcal{P}^r_\rightarrow(x,v) \neq \emptyset$. Let $P = (x = v_0, \ldots, v_{i-1}, v_i = v)$ be a path such that $|S_r(P)| = \gamma_\rightarrow(v)$. Observe that $\mathsf{d}(r,v_{i-1}) = \mathsf{d}(r,v_i)$. Let $Q = (v_0, \ldots, v_{i-1})$ and consider a set S of isometric r-rooted paths covering the vertices of Q of size $|S_r(Q)|$ and a (r,v_i)-shortest path P_i. Observe that $S \cup \{P_i\}$ is a set of isometric r-rooted paths covering the vertices of P. Consequently $\gamma_\rightarrow(v_i) = |S_r(P)| \leq |S_r(Q)| + 1 \leq \gamma(v_{i-1}) + 1$.

Consider now an isometric (x,v_{i-1})-path Q' such that $\gamma(v_{i-1}) = |S_r(Q')|$. Let P' be the isometric (x,v_i)-path obtained by appending v_i to Q'. Consider a set S' of isometric r-rooted paths covering the vertices of P' of size $|S_r(P')|$ and let P'_i be a path of S' covering v_i. By Lemma 10, no vertex of Q' is covered by P'_i. Consequently, $S' \setminus \{P'_i\}$ is a set of isometric r-rooted paths covering all vertices of Q' and thus $\gamma(v_{i-1}) \leq |S_r(P')| - 1 \leq \gamma_\rightarrow(v_i) - 1$. Thus, we have $\gamma_\rightarrow(v_i) = \gamma(v_{i-1}) + 1$. $\qquad\square$

Lemma 14. *If r is the root vertex, x the source vertex, and v is a vertex distinct from x, then $\gamma_\searrow(v) = 0$ if $\mathcal{P}^r_\searrow(x,v) = \emptyset$ and $\gamma_\searrow(v) = \max\{\max\{\gamma_\searrow(u), \gamma_\rightarrow(u), \beta_\nearrow(u)+1\} : u \in I(x,v) \cap N(v);\ \mathsf{d}(r,u) = \mathsf{d}(r,v)-1\}$ otherwise.*

Proof. Observe that $\mathcal{P}^r_\searrow(x,v)$ is empty if and only if there is no vertex $u \in I(x,v) \cap N(v)$ such that $\mathsf{d}(r,u) = \mathsf{d}(r,v) - 1$. If $\mathcal{P}^r_\searrow(x,v)$ is empty, then $\gamma_\searrow(v) = 0$ and we are done. Assume now that $\mathcal{P}^r_\searrow(x,v) \neq \emptyset$. If v is adjacent to x, then $P = (x,v)$ is the unique isometric (x,v)-path, and since $\mathcal{P}^r_\searrow(x,v) \neq \emptyset$, we have $\mathsf{d}(r,x) = \mathsf{d}(r,v) - 1$. Then P can be covered by any isometric (r,v)-path containing x, and thus $\gamma_\searrow(v) = |S_r(P)| = 1 = \gamma_\searrow(x) = \gamma_\rightarrow(x) = 1 + \beta_\nearrow(x)$.

Assume now that v is not adjacent to x. Let $P = (x = v_0, \ldots, v_{i-1}, v_i = v)$ be a path such that $|S_r(P)| = \gamma_\searrow(v)$, let $Q = (v_0, \ldots, v_{i-1})$, and let $R = (v_0, \ldots, v_{i-2})$. Note that $\mathsf{d}(r,v_{i-1}) = \mathsf{d}(r,v_i) - 1$.

First suppose that $\mathsf{d}(r,v_{i-2}) = \mathsf{d}(r,v_{i-1}) - 1$. We claim that $|S_r(P)| \leq |S_r(Q)|$. Indeed, consider a set S of isometric r-rooted paths covering the vertices of Q of size $|S_r(Q)|$. Let $P_{i-1} \in S$ be a path covering v_{i-1}. By Lemma 10 and since $\mathsf{d}(r,v_{i-2}) = \mathsf{d}(r,v_{i-1}) - 1$, we can assume that P_{i-1} is an isometric (r,v_{i-1})-path. Consider the path P_i obtained by appending v_i at the end of P_{i-1} and observe that P_i is an isometric (r,v_i)-path covering the same vertices as P_{i-1} as well as v_i. Consequently, replacing P_{i-1} by P_i in S, we obtain a set of isometric r-rooted paths of size $|S| = |S_r(Q)|$ covering all vertices of P, establishing
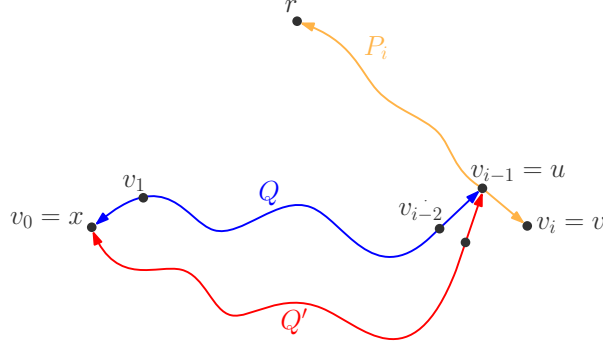
Figure 2: An illustration for the third case in the proof of Lemma 14.

that $|S_r(P)| \leq |S_r(Q)|$. Since $|S_r(Q)| \leq \gamma_\searrow(v_{i-1}) \leq \gamma(v_{i-1}) \leq \gamma_\searrow(v) = |S_r(P)| \leq |S_r(Q)|$, we have $\gamma_\searrow(v) = \gamma_\searrow(v_{i-1})$.

Suppose now that $\mathsf{d}(r, v_{i-2}) = \mathsf{d}(r, v_{i-1})$. As in the previous case, we show that $|S_r(P)| \leq |S_r(Q)|$. Indeed, consider a set S of isometric r-rooted paths covering the vertices of Q of size $|S_r(Q)|$. Let $P_{i-1} \in S$ be a path covering v_{i-1}. By Lemma 10 and since $\mathsf{d}(r, v_{i-2}) = \mathsf{d}(r, v_{i-1})$, v_{i-1} is the unique vertex of Q covered by P_{i-1}. Consequently, if we replace P_{i-1} in S by an isometric (r, v_i)-path going through v_{i-1}, we obtain a set of isometric r-rooted paths of size $|S| = |S_r(Q)|$ covering all vertices of P, establishing that $|S_r(P)| \leq |S_r(Q)|$. Since $|S_r(Q)| \leq \gamma_\rightarrow(v_{i-1}) \leq \gamma(v_{i-1}) \leq \gamma_\searrow(v) = |S_r(P)| \leq |S_r(Q)|$, we have $\gamma_\searrow(v) = \gamma_\rightarrow(v_{i-1})$.

Finally, suppose that $\mathsf{d}(r, v_{i-2}) = \mathsf{d}(r, v_{i-1}) + 1$ (see Figure 2 for an illustration of this case). Consider a set S of isometric r-rooted paths covering the vertices of R of size $|S_r(R)|$ and a (r, v_i)-shortest path P_i containing v_{i-1}. Observe that $S \cup \{P_i\}$ is a set of isometric r-rooted paths covering the vertices of P. Consequently, $\gamma_\searrow(v_i) = |S_r(P)| \leq |S_r(R)| + 1 \leq \gamma(v_{i-2}) + 1 \leq \beta_\nearrow(v_{i-1}) + 1$. Consider now an isometric (x, v_{i-1})-path Q' such that $\beta_\nearrow(v_{i-1}) = |S_r(R')|$ where $R' = Q' - \{v_{i-1}\}$. Let P' be the isometric (x, v_i)-path obtained by appending v_i to Q'. Consider a set S' of isometric r-rooted paths covering the vertices of P' of size $|S_r(P')|$ and let P_i' be the path of S' covering v_i. By Lemma 10, the only vertex of Q' that can be covered by P_i' is v_{i-1}. Consequently, $S' \setminus \{P_i'\}$ is a set of isometric r-rooted paths covering all vertices of R' and thus $\beta_\nearrow(v_{i-1}) = |S_r(R')| \leq |S_r(P')| - 1 \leq \gamma_\searrow(v_i) - 1$. Thus, we have $\gamma_\searrow(v_i) = \beta_\nearrow(v_{i-1}) + 1$.

Since the formula for computing $\gamma_\searrow(v)$ (given in the statement of the lemma) takes into account these three exclusive alternatives, it computes $\gamma_\searrow(v)$ correctly. $\qquad\square$

Lemma 15. *If r is the root vertex, x the source vertex, and v is a vertex distinct from x, then $\gamma_\nearrow(v) = 0$ if $\mathcal{P}_\nearrow^r(x, v) = \emptyset$ and $\gamma_\nearrow(v) = \max\{\max\{\gamma_\nearrow(u), \gamma_\rightarrow(u), \beta_\searrow(u) + 1\} : u \in I(x, v) \cap N(v); \mathsf{d}(r, u) = \mathsf{d}(r, v) + 1\}$ otherwise.*

Proof. The proof is similar to the the proof of Lemma 14. $\qquad\square$

Now we provide a BFS based algorithm to compute the above parameters. Let r and x be fixed root and source vertices of G, respectively. For a vertex $u \in V(G)$, let $\mathcal{D}(u) = \{\gamma(u), \gamma_\nearrow(u), \gamma_\rightarrow(u), \gamma_\searrow(u), \beta_\nearrow(u), \beta_\searrow(u)\}$. Clearly, the set $\mathcal{D}(x)$ can be computed in constant time. Now let X_i be the set of vertices at distance i from x. Clearly, the sets X_i can be computed in $O(|E(G)|)$-time (using a BFS) and $X_0 = \{x\}$. Let $i \geq 1$ be an integer and assume that for all vertices $u \in \bigcup_{j=0}^{i-1} X_j$, the set $\mathcal{D}(u)$ is already computed. Let $v \in X_i$ be a vertex. Then due to the formulas given in Observation 12 and Lemmas 13–15, the set $\mathcal{D}(v)$ can be computed by observing only the sets $\mathcal{D}(u)$, $u \in N(v) \cap X_{i-1}$. Hence, for all vertices $v \in V(G)$, the sets $\mathcal{D}(v)$ can be computed in a total of $O(|E(G)|)$ time. Hence, we have the following lemma.

Lemma 16. *For a root vertex r and source vertex x, for each vertex $v \in V(G)$, the value $\gamma^r(x, v)$ can be computed in $O(|E(G)|)$ time.*

We can now finish the proof of Theorem 2. Let G be a graph with n vertices and m edges. For a root vertex r, by applying Lemma 16, for every source $x \in V(G)$, it is possible to compute $ipco\left(\overrightarrow{G_r}\right) =$
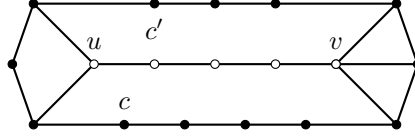
Figure 3: An example of a 4-fat turtle. Let C be the cycle induced by the black vertices, P be the path induced by the white vertices. Then the tuple $(4, C, P, c, c')$ defines a 4-fat turtle.

$\max_{x,v} \gamma^r(x, v)$ in $O(nm)$ time. By repeating this for every root $r \in V(G)$, it is possible to compute $ipco(G) = \min_r ipco\left(\overrightarrow{G_r}\right)$ in $O(n^2 m)$ time.

4 Proof of Theorem 3

First we prove Theorem 3(a). We recall the definition of Gromov products [24] and its relation with hyperbolicity. For three vertices r, x, y of a graph G, the Gromov product of x, y with respect to r is defined as $(x|y)_r = \frac{1}{2}(\mathsf{d}(x, r) + \mathsf{d}(y, r) - \mathsf{d}(x, y))$. Then, a graph G is δ-hyperbolic [12, 24] if and only if for any four vertices x, y, z, r, we have $(x|y)_r \geq \min\{(x|z)_r, (y|z)_r\} - \delta$.

Let G be a graph with hyperbolicity at most δ. Due to Lemma 7, in order to prove Theorem 3(a), it is enough to show that $ipacc(G) \leq 4\delta + 3$. Aiming for a contradiction, let r be a vertex of G and P be an isometric path such that $|A_r(P)| \geq 4\delta + 4$. Let $a_1, a_2, \ldots, a_{2\delta+2}, \ldots, a_{4\delta+4}$ be the vertices of $A_r(P)$ ordered as they are encountered while traversing P from one endpoint to the other. Let $x = a_1, z = a_{2\delta+2}, y = a_{4\delta+4}$. Let Q denote the (y, z)-subpath of P. Observe that, $|A_r(Q)| \geq 2\delta + 2$. Then we have $(x|y)_r \geq \min\{(x|z)_r, (y|z)_r\} - \delta$. Without loss of generality, assume that $(x|z)_r \leq (y|z)_r$. Hence,

$$(x|y)_r \geq (x|z)_r - \delta$$
$$\mathsf{d}(x, r) + \mathsf{d}(y, r) - \mathsf{d}(x, y) \geq \mathsf{d}(x, r) + \mathsf{d}(z, r) - \mathsf{d}(x, z) - 2\delta$$
$$\mathsf{d}(y, r) - \mathsf{d}(x, y) \geq \mathsf{d}(z, r) - \mathsf{d}(x, z) - 2\delta$$
$$\mathsf{d}(y, r) - \mathsf{d}(z, r) + 2\delta \geq \mathsf{d}(x, y) - \mathsf{d}(x, z)$$
$$\mathsf{d}(y, r) - \mathsf{d}(z, r) + 2\delta \geq \mathsf{d}(y, z)$$
$$\mathsf{d}(y, z) \leq |\mathsf{d}(y, r) - \mathsf{d}(z, r)| + 2\delta.$$

But this directly contradicts Proposition 9, which implies that $\mathsf{d}(y, z) \geq |\mathsf{d}(y, r) - \mathsf{d}(z, r)| + |A_r(Q)| - 1 \geq |\mathsf{d}(y, r) - \mathsf{d}(z, r)| + 2\delta + 1$. This completes the proof of Theorem 3(a).

Now, we shall prove Theorems 3(b) and 3(c). First, we shall define the notions of t-*theta*, t-*prism*, and t-*pyramid* [32]. For an integer $t \geq 1$, a t-*prism* is a graph made of three vertex-disjoint induced paths $P_1 = a_1 \ldots b_1$, $P_2 = a_2 \ldots b_2$, $P_3 = a_3 \ldots b_3$ of lengths at least t, such that $a_1 a_2 a_3$ and $b_1 b_2 b_3$ are triangles and no edges exist between the paths except those of the two triangles. For an integer $t \geq 1$, a t-*pyramid* is a graph made of three induced paths $P_1 = a \ldots b_1$, $P_2 = a \ldots b_2$, $P_3 = a \ldots b_3$ of lengths at least t, two of which have lengths at least $t + 1$, they are pairwise vertex-disjoint except at a, such that $b_1 b_2 b_3$ is a triangle and no edges exist between the paths except those of the triangle and the three edges incident to a. For an integer $t \geq 1$, a t-*theta* is a graph made of three internally vertex-disjoint induced paths $P_1 = a \ldots b$, $P_2 = a \ldots b$, $P_3 = a \ldots b$ of lengths at least $t + 1$, and such that no edges exist between the paths except the three edges incident to a and the three edges incident to b. A graph G is *(t-theta, t-pyramid, t-prism)*-free if G does not contain any induced subgraph isomorphic to a t-theta, t-pyramid or t-prism. When $t = 1$, (t-theta, t-pyramid, t-prism)-free graphs are exactly (theta, prism, pyramid)-free graphs.

Now, we shall show that the isometric path antichain cover number of (t-theta, t-pyramid, t-prism)-free graphs are bounded above by a linear function on t. We shall show that, when the isometric path antichain cover number of a graph is large, the existence of a structure called "t-fat turtle" (defined below) as an induced subgraph is forced, which, cannot be present in a $((t-1)$-theta, $(t-1)$-pyramid, $(t-1)$-prism)-free graph.

Definition 17. *For an integer $t \geq 1$, a "t-fat turtle" consists of a cycle C and an induced (u,v)-path P of length at least t such that all of the following hold:*

(a) *$V(P) \cap V(C) = \emptyset$,*

(b) *For any vertex $w \in (V(P) \setminus \{u,v\})$, $N(w) \cap V(C) = \emptyset$ and both u and v have at least one neighbour in C,*

(c) *For any vertex $w \in N(u) \cap V(C)$ and $w' \in N(v) \cap V(C)$, the distance between w and w' in C is at least t,*

(d) *There exist two vertices $\{c, c'\} \subset V(C)$ and two distinct components C_u, C_v of $C - \{c, c'\}$ such that $N(u) \cap V(C) \subseteq V(C_u)$ and $N(v) \cap V(C) \subseteq V(C_v)$.*

The tuple (t, C, P, c, c') denotes the t-fat turtle. See Figure 3 for an example.

In the following observation, we show that any (t-theta, t-pyramid,t-prism)-free graph cannot contain a $(t+1)$-fat turtle as an induced subgraph.

Lemma 18. *For some integer $t \geq 1$, let G be a graph containing a $(t+1)$-fat turtle as an induced subgraph. Then G is not (t-theta, t-pyramid, t-prism)-free.*

Proof. Let $(t+1, C, P, c, c')$ be a $(t+1)$-fat turtle in G. Let the vertices of C be named $c = a_0, a_1, \ldots, a_k = c', a_{k+1}, \ldots, a_{|V(C)|}$ as they are encountered while traversing C starting from c in a counter-clockwise manner. Denote by u, v the endpoints of P. By definition, there exist two distinct components C_u, C_v of $C - \{c, c'\}$ such that $N(u) \cap V(C) \subseteq V(C_u)$ and $N(v) \cap V(C) \subseteq V(C_v)$. Without loss of generality, assume $V(C_u) = \{a_1, a_2, \ldots, a_{k-1}\}$ and $V(C_v) = \{a_{k+1}, a_{k+2}, \ldots, a_{|V(C)|}\}$. Let i^- and i^+ be the minimum and maximum indices such that a_{i^-} and a_{i^+} are adjacent to u. Let j^- and j^+ be the minimum and maximum indices such that a_{j^-} and a_{j^+} are adjacent to v. By definition, $i^- \leq i^+ < j^- \leq j^+$. Let P_1 be the (a_{i^-}, a_{j^+})-subpath of C containing c. Let P_2 be the (a_{i^+}, a_{j^-})-subpath of C that contains c'. Observe that P_1 and P_2 have length at least t (by definition). Now we show that P, P_1, P_2 together form one of theta, pyramid or prism. If $a_{i^-} = a_{i^+}$ and $a_{j^-} = a_{j^+}$, then P, P_1, P_2 form a t-theta. If $i^- \leq i^+ - 2$ and $j^- \leq j^+ - 2$, then also P, P_1, P_2 form a t-theta. If $j^- = j^+ - 1$ and $i^- = i^+ - 1$, then P, P_1, P_2 form a t-prism. In any other case, P, P_1, P_2 form a t-pyramid. $\square$

In the remainder of this section, we shall prove that there exists a linear function $f(t)$ such that if the isometric path antichain cover number of a graph is more than $f(t)$, then G is forced to contain a $(t+1)$-fat turtle as an induced subgraph, and therefore is not (t-theta, t-pyramid,t-prism)-free. We shall use the following observation.

Observation 19. *Let G be a graph, r be an arbitrary vertex, P be an isometric (u,v)-path in G and Q be a subpath of an isometric (v,r)-path in G such that one endpoint of Q is v. Let P' be the maximum (u,w)-subpath of P such that no internal vertex of P' is a neighbour of some vertex of Q. We have that $|A_r(P')| \geq |A_r(P)| - 3$.*

Proof. Suppose $|A_r(P')| \leq |A_r(P)| - 4$ and consider the (w,v)-subpath, say P'', of P. Observe that $|A_r(P'')| \geq 4$. Now let w' be a vertex of Q which is a neighbour of w. Observe that $|\mathsf{d}(r,w) - \mathsf{d}(r,w')| \leq 1$ and therefore $\mathsf{d}(w,v) = |E(P'')| \leq |\mathsf{d}(r,w) - \mathsf{d}(r,v)| + 2$. But this contradicts Proposition 9, which implies that the length of P'' is at least $|\mathsf{d}(r,w) - \mathsf{d}(r,v)| + 3$. $\square$

Lemma 20. *For an integer $t \geq 1$, let G be a graph with $\mathrm{ipacc}(G) \geq 8t + 64$. Then G has a $(t+1)$-fat turtle as an induced subgraph.*

Proof. Let r be a vertex of G such that $ipacc\left(\overrightarrow{G_r}\right)$ is at least $8t + 64$. Then there exists an isometric path P such that $|A_r(P)| \geq 8t + 64$. Let the two endpoints of P be a and b. (See Figure 4.) Let u be a vertex of P such that $\mathsf{d}(r,u) = \mathsf{d}(r,P)$. Let $P(a,u)$ be the (a,u)-subpath of P and $P(b,u)$ be the (b,u)-subpath of P. Both $P(a,u)$ and $P(b,u)$ are isometric paths and observe that either $|A_r(P(a,u))| \geq 4t + 32$ or $|A_r(P(b,u))| \geq 4t + 32$. Without loss of generality, assume that $|A_r(P(b,u))| \geq 4t + 32$. Let $Q(r,b)$ be an isometric (b,r)-path in G. First observe that u is not adjacent to any vertex of $Q(r,b)$. Otherwise, $\mathsf{d}(u,b) \leq 2 + \mathsf{d}(r,b) - \mathsf{d}(r,u)$, which contradicts Proposition 9. Let $P(u,w)$ be the maximum (u,w)-subpath, of $P(b,u)$ such that no internal vertex of $P(u,w)$ is a neighbour of $Q(r,b)$. Note that $P(u,w)$ is an isometric path and w has a neighbour in $Q(r,b)$. Applying Observation 19, we have the following:
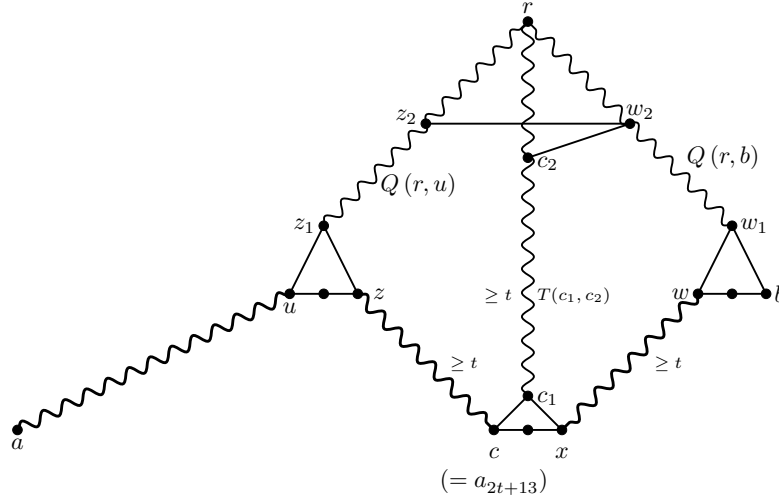
10

Figure 4: Illustration of the notations used in the proof of Lemma 20. The thick wiggly line along with the horizontal edges indicate the (a, b)-isometric path P.

Claim 20.1. $|A_r(P(u, w))| \geq 4t + 29$.

Let $Q(r, u)$ be any isometric (u, r)-path of G. Observe that w is not adjacent to any vertex of $Q(r, u)$. Otherwise, $\mathsf{d}(u, w) \leq 2 + \mathsf{d}(r, u) - \mathsf{d}(r, w)$, which contradicts Proposition 9. Let $P(z, w)$ be the maximum (z, w)-subpath of $P(u, w)$ such that no internal vertex of $P(z, w)$ has a neighbour in $Q(r, u)$. Observe that $P(z, w)$ is an isometric path, and z has a neighbour in $Q(r, u)$. Again applying Observation 19, we have the following:

Claim 20.2. $|A_r(P(z, w))| \geq 4t + 26$.

Let $a_1, a_2, \ldots, a_k$ be the vertices of $A_r(P(z, w))$ ordered according to their appearance while traversing $P(z, w)$ from z to w. Due to Claim 20.2, we have that $k \geq 4t + 26$. Let $c = a_{2t+13}$ and $Q(r, c)$ denote an isometric (c, r)-path. Let $T(r, c_1)$ be the maximum subpath of $Q(r, c)$ such that no internal vertex of $T(r, c_1)$ is adjacent to any vertex of $P(z, w)$. Observe that neither z nor w can be adjacent to c_1 (due to Proposition 9). Morevoer, if c_1 is a vertex of $P(z, w)$ then we must have $c_1 = c$.

Claim 20.3. *Let x be a neighbour of c_1 in $P(z, w)$, X be the (x, b)-subpath of $P(u, b)$ and Y be the (x, u)-subpath of $P(u, b)$. Then $|A_r(X)| \geq 2t + 11$ and $|A_r(Y)| \geq 2t + 11$.*

Proof. Let $P(c, w)$ denote the (c, w)-subpath of $P(z, w)$. Observe that $|A_r(P(c, w))| \geq 2t + 14$. First, consider the case when x lies in the (z, c)-subpath of $P(z, w)$. In this case, $P(c, w)$ is a subpath of X and therefore $|A_r(X)| \geq 2t + 14$. Now consider the case when x lies in $P(c, w)$. In this case, applying Observation 19, we have that $|A_r(X)| \geq |A_r(P(c, w))| - 3 \geq 2t + 11$. Using a similar argument, we have that $|A_r(Y)| \geq 2t + 11$. ∎

Let $T(c_1, c_2)$ be the maximum (c_1, c_2)-subpath of $T(c_1, r)$ such that no internal vertex of $T(c_1, c_2)$ is adjacent to a vertex of $Q(r, b)$ or $Q(r, u)$. Note that, if c_2 lies on $Q(r, b)$ or $Q(r, u)$, we must have $c_2 = r$. We have the following claim.

Claim 20.4. *The length of $T(c_1, c_2)$ is at least $t + 3$.*

Proof. Assume that the length of $T(c_1, c_2)$ is at most $t + 2$ and x be a neighbour of c_1 in $P(z, w)$. Observe that all vertices of $P(z, w)$ are at distance at least $\mathsf{d}(r, u)$ i.e. $\mathsf{d}(r, P(z, w)) \geq \mathsf{d}(r, u)$, since $\mathsf{d}(r, u) = \mathsf{d}(r, P)$. Hence,

$(+)$ $\mathsf{d}(r, x) \geq \mathsf{d}(r, u)$ and $\mathsf{d}(r, c_1) \geq \mathsf{d}(r, u) - 1$.

Now, suppose c_2 has a neighbour c_3 in $Q(r, u)$. Hence $\mathsf{d}(c_3, x) \leq \mathsf{d}(c_3, c_2) + \mathsf{d}(c_2, c_1) + \mathsf{d}(c_1, x) \leq t + 4$. Now, using $(+)$ and the fact that c_3 lies on an isometric (r, u)-path ($Q(r, u)$), we have that $\mathsf{d}(c_3, u) \leq t + 4$.

Therefore, $d(u,x) \leq d(c_3,u) + d(c_3,x) \leq 2t+8$. But this contradicts Proposition 9 and Claim 20.3, as they together imply that $d(u,x)$ is at least $d(r,x) - d(r,u) + 2t + 10 \geq 2t + 10$.

Hence, c_2 must have a neighbour c_3 in $Q(r,b)$. First, assume that $d(r,x) \geq d(r,b)$. Then, as $d(c_3,x) \leq d(c_3,c_2) + d(c_2,c_1) + d(c_1,x) \leq t+4$ and c_3 lies on an isometric (r,b)-path $(Q(r,b))$, we have that $d(x,b) \leq 2t+8$. But again this contradicts Proposition 9 and Claim 20.3, as they together imply that the length of $d(x,b)$ is at least $d(r,x) - d(r,u) + 2t + 10$. Now, assume that $d(r,x) < d(r,b)$. Let b' be a vertex of $Q(r,b)$ such that $d(r,b') = d(r,x)$. Using a similar argumentation as before, we have that $d(x,b') \leq 2t+8$. Hence, $d(x,b) \leq d(x,b') + d(b',b) \leq d(r,b) - d(r,x) + 2t + 8$. But this contradicts Proposition 9 which, due to Claim 20.3, implies that $d(x,b) \geq d(r,b) - d(r,x) + 2t + 10$. ∎

The path $T(c_1,c_2)$ forms the first ingredient to extract a $(t+1)$-fat turtle. Let z_1 be the neighbour of z in $Q(r,u)$ and w_1 be the neighbour of w in $Q(r,b)$. We have the following claim.

Claim 20.5. *The vertices w_1 and z_1 are non adjacent.*

Proof. Recall that z_1 lies in $Q(r,u)$ and $d(r,z) \geq d(r,u)$. Hence z_1 must be a neighbour of u. If w_1 and z_1 are adjacent, then observe that $d(u,b) \leq d(r,b) - d(r,w_1) + 2 \leq$. This implies $d(u,b) \leq d(r,b) - d(r,u) + 3$. But this shall again contradict Proposition 9. ∎

Now we shall construct a (w_1,z_1)-path as follows: Consider the maximum (w_1,w_2)-subpath, say $T(w_1,w_2)$, of $Q(r,b)$ such that no internal vertex of $T(w_1,w_2)$ has a neighbour in $Q(r,u)$. Similarly, consider the maximum (z_1,z_2)-subpath, say $T(z_1,z_2)$, of $Q(r,u)$ such that no internal vertex of $T(z_1,z_2)$ is a neighbour of w_2. (Note that it is possible that $z_2 = w_2 = r$.) Let T be the path obtained by taking the union of $T(w_1,w_2)$ and $T(z_1,z_2)$. Observe that z_2 must be a neighbour of w_2 and T is an induced (w_1,z_1)-path. The definitions of T and $P(z,w)$ imply that their union induces a cycle Z. Here we have the second and final ingredient to extract the $(t+1)$-fat turtle.

Suppose that c_2 has a neighbour in T. Let T' be the maximum subpath of $T(c_1,c_2)$ which is vertex-disjoint from Z. (Note that if $c_1 = c$ or $c_2 \in \{w_2,z_2\}$ (e.g. when $c_2 = w_2 = z_2 = r$), $T(c_1,c_2)$ may share vertices with Z.) Due to Claim 20.4, the length of T' is at least $t+1$. Let e_1 and e_2 be the endpoints of T'. Observe the following.

- Each of e_1 and e_2 has at least one neighbour in Z.

- $Z - \{z,w\}$ contains two distinct components C_1, C_2 such that for $i \in \{1,2\}$, $N(e_i) \cap V(Z) \subseteq V(C_i)$.

- For a vertex $e_1' \in N(e_1) \cap V(Z)$ and $e_2' \in N(e_2) \cap V(Z)$, the distance between e_1' and e_2' is at least $t+1$. This statement follows from Claim 20.3.

Hence, we have that the tuple $(t+1, Z, T', z, w)$ defines a $(t+1)$-fat turtle. Now consider the case when c_2 does not have a neighbour in T. By definition, c_2 has at least one neighbour in $Q(r,u)$ or $Q(r,b)$. Without loss of generality, assume that c_2 has a neighbour c_3 in $Q(r,u)$ such that the (z_2,c_3)-subpath, say, T'' of $Q(r,u)$ has no neighbour of c_2 other than c_3. Observe that the path $T^* = (T' \cup (T'' - \{z_2\}))$ is vertex-disjoint from Z and has length at least $t+1$. Let e_1,e_2 be the two endpoints of T^*. Observe the following.

- Each of e_1 and e_2 has at least one neighbour in Z.

- $Z - \{z,w\}$ contains two distinct components C_1, C_2 such that for $i \in \{1,2\}$, $N(e_i) \cap V(Z) \subseteq V(C_i)$.

- For a vertex $e_1' \in N(e_1) \cap V(Z)$ and $e_2' \in N(e_2) \cap V(Z)$, the distance between e_1' and e_2' is at least $t+1$. This statement follows from Claim 20.3.

Hence, $(t+1, Z, T^*, z, w)$ is a $(t+1)$-fat turtle ☐

Proof of Theorem 3(b): Lemma 7, 18 and 20 together imply Theorem 3(b).

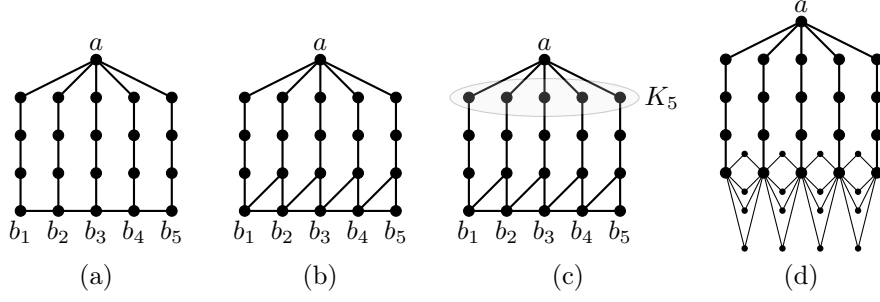**Lemma 21.** *Any outerstring graph is (4-theta, 4-prism, 4-pyramid)-free.*

Figure 5: (a) X_4 (b) Y_4 (c) Z_4 (d) W_4.

Proof. To prove the lemma, we shall need to recall a few definitions and results from the literature. A graph G is a *string* graph if there is a collection S of simple curves on the plane and a bijection between $V(G)$ and S such that two curves in S intersect if and only if the corresponding vertices are adjacent in G. Let G be a graph with an edge e. The graph G/e is obtained by *contracting* the edge e into a single vertex. Observe that string graphs are closed under edge contraction [28]. We shall use the following result.

Proposition 22 ([28]). *Let G be an outerstring graph with an edge e. Then G/e is an outerstring graph.*

A *full subdivision* of a graph is a graph obtained by replacing each edge of G with a new path of length at least 2. We shall use the following result implied from Theorem 1 of [28].

Proposition 23 ([28]). *Let G be a string graph. Then G does not contain a full subdivision of $K_{3,3}$ as an induced subgraph.*

Indeed, if the full subdivision of $K_{3,3}$ was a string graph, then the corresponding intersection representation with simple curves could be used to construct a planar drawing of $K_{3,3}$, which is impossible. For a graph G, the graph G^+ is constructed by introducing a new *apex* vertex a and connecting a with all vertices of G by new copies of paths of length at least 2. We shall use the following result of Biedl *et al.* [5].

Proposition 24 (Lemma 1, [5]). *A graph G is an outerstring graph if and only if G^+ is a string graph.*

Now we are ready to prove the lemma. Let G be an outerstring graph. Assume for the sake of contradiction that G contains an induced subgraph H which is a 4-theta, 4-pyramid, or a 4-prism. Since every induced subgraph of an outerstring graph is also an outerstring graph, we have that H is an outerstring graph. Let E be the set of edges of H whose both endpoints are part of the same triangle. Now consider the graph $H_1 = H/E$ which is obtained by contracting all edges in E. By Proposition 22, H_1 is an outerstring graph and it is easy to check that H_1 is a 3-theta. Let u and v be the vertices of H_1 with degree 3 and w_1, w_2, w_3 be the set of mutually non-adjacent vertices such that for each $i \in \{1, 2, 3\}$ $\mathsf{d}(u, w_i) = 2$ and $\mathsf{d}(v, w_i) \geq 2$. Since H_1 is a 3-theta, w_1, w_2, w_3 exist. Now consider the graph H_1^+ and let a be the new apex vertex. Due to Proposition 24, we have that H_1^+ is a string graph. But notice that, for each pair of vertices in $\{x, y\} \subset \{w_1, w_2, w_3, u, v, a\}$, there exists a unique path of length at least 2 connecting x, y. This implies that H_1^+ (which is a string graph) contains a full subdivision of $K_{3,3}$, which contradicts Proposition 23. $\qquad\square$

Proof of Theorem 3(c): Lemma 7, 18, 20, and 21 together imply Theorem 3(c).

5 Proof of Theorem 4

We shall provide a construction for every $k \geq 4$, this implies the statement of Theorem 4 for any $k \geq 1$. First we shall prove Theorem 4(a). For a fixed integer $k \geq 4$, first we describe the construction of a graph X_k as follows. Consider $k + 1$ paths $P_1, P_2, \ldots, P_{k+1}$ each of length k and having a common endpoint a. For $i \in [k + 1]$, let the other endpoint of P_i be denoted as b_i. Moreover, for $i \in [k + 1]$, let the neighbours of a and b_i in P_i be denoted as a'_i and b'_i, respectively. For $i \in [k]$, introduce an edge between b_i and b_{i+1}. The resulting graph is denoted X_k and the special vertex a is the *apex* of X_k. See Figure 5(a).

For a fixed integer $k \geq 4$, consider the graph X_k and for each $i \in [k]$, introduce an edge between b_i and b'_{i+1}. Let Y_k denote the resulting graph and the special vertex a is the *apex* of Y_k. See Figure 5(b). For a fixed integer $k \geq 4$, consider the graph Y_k and for each $\{i, j\} \subseteq [k]$, introduce an edge between a'_i and a'_j. Let Z_k denote the resulting graph and the special vertex a is the *apex* of Z_k. See Figure 5(c). For a fixed integer $k \geq 4$, consider the graph X_k. For each $i \in [k]$, delete the edge $b_i b_{i+1}$ and introduce k new vertices, each of which is adjacent to only b_i and b_{i+1}. Call the resulting graph W_k. See Figure 5(d).

We shall use the following result relating hyperbolicity and *isometric cycles*. An induced cycle C of a graph G is *isometric* if for any two vertices u, v of C, the distance between u, v in C is the same as that in G.

Proposition 25 (Theorem 2, [36]). *Let G be a graph containing an isometric cycle of order k with $k \equiv c \pmod 4$. Then the hyperbolicity of G is at least $\lceil \frac{k}{4} \rceil - \frac{1}{2}$ if $c = 1$ and $\lceil \frac{k}{4} \rceil$, otherwise.*

We now prove the following lemmas.

Lemma 26. *For $k \geq 4$, let G be the graph constructed by taking two distinct copies of X_k and identifying the two apex vertices. Then G is a (pyramid, prism)-free graph with tree-width 2, hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$ and $ipacc(G) \geq k$.*

Proof. Since G is triangle-free, clearly G is (pyramid, prism)-free. Moreover, for any induced cycle C of G, and any vertex $w \notin C$, observe that w has only one neighbour in C. Therefore, G is also wheel-free. Observe that G has an isometric cycle of length at least $2k$. Therefore, due to Proposition 25, G has hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$. Since removing the vertex a from G makes it acyclic, the tree-width of G is two. Let H and H' denote the two copies of X_k used to construct G. Let r be any vertex of G and, without loss of generality, assume that r is a vertex of H'. Consider the graph $\overrightarrow{G_r}$. Now recall the construction of H (which is isomorphic to X_k) and consider the path $Q = b_1\, b_2 \ldots b_k$. Observe that Q is an isometric path and for any two vertices $u, v \in V(Q)$ we have $\mathsf{d}(r, u) = \mathsf{d}(r, v)$. Therefore, $A_r(Q) \geq k$. Hence, $ipacc(G) \geq k$. $\square$

Lemma 27. *For $k \geq 4$, let G be the graph constructed by taking two distinct copies of Y_k and identifying the two apex vertices. Then G is a (theta, prism)-free graph with tree-width 3, hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$, and $ipacc(G) \geq k$.*

Proof. Since removing the special vertex a from G results in a graph with tree-width 2, it follows that G has tree-width at most 3. Observe that G has an isometric cycle of length at least $2k$. Therefore, due to Proposition 25, G has hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$. Let H and H' denote the two copies of Y_k used to construct G. First we shall show that H does not contain a theta or a prism. Consider the graph H_1 obtained by removing the apex of H. Observe that H_1 does not contain a vertex v such that the vertices in $N[v]$ induce a $K_{1,3}$. Hence H does not contain a theta. It also can be verified that H_1 does not contain a prism. Since the neighbourhood of a is triangle-free, it follows that H does not contain a prism. Similarly, H' does not contain a theta or a prism. Now, from our construction, it follows that G does not contain a theta or a prism. Moreover, for any induced cycle C of G, and any vertex $w \notin C$, observe that w has at most two neighbours in C. Therefore, G is wheel-free. Using arguments similar to the ones used in the proof of Lemma 26, we have that $ipacc(G) \geq k$. $\square$

Lemma 28. *For $k \geq 4$, let G be the graph constructed by taking two distinct copies of Z_k and identifying the two apex vertices. Then G is a (theta, pyramid)-free graph with hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$ and $ipacc(G) \geq k$.*

Proof. Observe that G has an isometric cycle of length at least $2k$. Therefore, due to Proposition 25, G has hyperbolicity at least $\lceil \frac{k}{2} \rceil - 1$. Let H and H' denote the two copies of Y_k used to construct G. Observe that H does not contain a vertex v such that the vertices in $N[v]$ induce a $K_{1,3}$. Therefore, H does not contain a theta or a pyramid. Similarly, H' does not contain a theta or a pyramid. Due to our construction, it follows that G does not contain a theta or a pyramid. Moreover, for any induced cycle C of G, and any vertex $w \notin C$, observe that w has at most two neighbours in C. Therefore, G is wheel-free. Using arguments similar to the ones used in the proof of Lemma 26, we have that $ipacc(G) \geq k$. $\square$

An isometric path cover C of a graph G is *rooted* if there exists a vertex v such that all paths in C are v-rooted isometric paths.

Lemma 29. *For $k \geq 4$, let G be the graph constructed by taking two distinct copies of W_k and identifying the two apex vertices. Then G is a (prism, pyramid, wheel)-free planar graph such that any rooted isometric path cover of G has cardinality at least k^2 but there is an isometric path cover of G of cardinality $3k + 1$.*

Proof. The construction ensures that G is a (prism, pyramid, wheel)-free planar graph. Let H and H' denote the two copies of W_k used to construct G and a denote the apex vertex. Observe that there are k^2 vertices at maximum distance from the apex vertex a in H and a a-rooted isometric path can only cover one of them. Therefore, at least k^2 many a-rooted isometric paths are needed to cover the graph H. As H' is isomorphic to H, it has the above properties. Since a is a cut-vertex in G, it is easy to verify that for any vertex $v \in V(G)$, any v-rooted isometric path cover of G requires k^2 many paths. On the other hand, it is easy to check that G has an isometric path cover of cardinality $3k + 1$. Indeed $k + 1$ geodesics are sufficient to cover the vertices of the maximal isometric paths containing a, $2k$ geodesics are sufficient to cover the remaining vertices of G. □

Lemma 7, 26, 27, 28, 29 imply Theorem 4.

6   Conclusion

In this paper, we have introduced the new graph parameter *isometric path complexity*. We have shown that the isometric path complexity of a graph with n vertices and m edges can be computed in $O(n^2m)$-time. It would be interesting to provide a faster algorithm to compute the isometric path complexity of a graph. Note that, no non-trivial lower bound on the achievable running time is known.

We have derived upper bounds on the isometric path complexity of three seemingly (structurally) different classes of graphs, namely hyperbolic graphs, (theta, pyramid, prism)-free graphs and outerstring graphs. An interesting direction of research is to generalise the properties of hyperbolic graphs or (theta, pyramid, prism)-free graphs to graphs with bounded isometric path complexity.

Note that, in our proofs we essentially show that, for any graph G that belongs to one of the above graph classes, any vertex v of G, and any isometric path P of G, the path P can be covered by a small number of v-rooted isometric paths. This implies our "choice of the root" is arbitrary. This motivates the following definition. The *strong isometric path complexity* of a graph G is the minimum integer k such that for each vertex $v \in V(G)$ we have that the vertices of any isometric path P of G can be covered by k many v-rooted isometric paths. Our proofs imply that the strong isometric path complexity of graphs from all the graph classes addressed in this paper are bounded. We also wonder whether one can find other interesting graph classes with small (strong) isometric path complexity.

Our results imply a constant-factor approximation algorithm for ISOMETRIC PATH COVER on hyperbolic graphs, (theta, pyramid, prism)-free graphs and outerstring graphs. However, the existence of a constant-factor approximation algorithm for ISOMETRIC PATH COVER on general graphs is not known (an $O(\log n)$-factor approximation algorithm is designed in [31]).

References

[1] P. Aboulker, M. Chudnovsky, P. Seymour, and N. Trotignon. Wheel-free planar graphs. *European Journal of Combinatorics*, 49:57–67, 2015.

[2] I. Abraham, C. Gavoille, A. Gupta, O. Neiman, and K. Talwar. Cops, robbers, and threatening skeletons: Padded decomposition for minor-free graphs. *SIAM Journal on Computing*, 48(3):1120–1145, 2019.

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

[4] R. Belmonte and M. Vatshelle. Graph classes with structured neighborhoods and algorithmic applications. *Theoretical Computer Science*, 511:54–65, 2013.

[5] T. Biedl, A. Biniaz, and M. Derka. On the size of outer-string representations. In *Proceedings of the 16th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2018)*, 2018.

[6] É. Bonnet, E. J. Kim, S. Thomassé, and R. Watrigant. Twin-width I: Tractable fo model checking. *Journal of the ACM*, 69(1), 2021.

[7] P. Bose, P. Carmi, J. M. Keil, A. Maheshwari, S. Mehrabi, D. Mondal, and M. Smid. Computing maximum independent set on outerstring graphs and their relatives. *Computational Geometry*, 103:101852, 2022.

[8] J. Cardinal, S. Felsner, T. Miltzow, C. Tompkins, and Birgit Vogtenhuber. Intersection graphs of rays and grounded segments. In *Proceedings of the International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2017)*, pages 153–166. Springer, 2017.

[9] D. Chakraborty, J. Chalopin, F. Foucaud, and Y. Vaxès. Isometric path complexity of graphs. In Jérôme Leroux, Sylvain Lombardy, and David Peleg, editors, *48th International Symposium on Mathematical Foundations of Computer Science (MFCS 2023)*, volume 272 of *LIPIcs*, pages 32:1–32:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023.

[10] D. Chakraborty, A. Dailly, S. Das, F. Foucaud, H. Gahlawat, and S. K. Ghosh. Complexity and algorithms for ISOMETRIC PATH COVER on chordal graphs and beyond. In *Proceedings of the 33rd International Symposium on Algorithms and Computation (ISAAC 2022)*, volume 248 of *LIPIcs*, pages 12:1–12:17, 2022.

[11] V. Chepoi, F. Dragan, B. Estellon, M. Habib, and Y. Vaxès. Diameters, centers, and approximating trees of δ-hyperbolic geodesic spaces and graphs. In *Proceedings of the 24th ACM Symposium on Computational Geometry (SCG 2008)*, pages 59–68. ACM, 2008.

[12] V. Chepoi, F. Dragan, M. Habib, Y. Vaxès, and H. Alrasheed. Fast approximation of eccentricities and distances in hyperbolic graphs. *Journal of Graph Algorithms and Applications*, 23(2):393–433, 2019.

[13] V. Chepoi, F. F. Dragan, and Y. Vaxes. Core congestion is inherent in hyperbolic networks. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2017)*, pages 2264–2279. SIAM, 2017.

[14] M. Chudnovsky, N. Robertson, P. Seymour, and R. Thomas. The strong perfect graph theorem. *Annals of mathematics*, pages 51–229, 2006.

[15] M. Conforti, G. Cornuéjols, A. Kapoor, and K. Vušković. Universally signable graphs. *Combinatorica*, 17(1):67–77, 1997.

[16] D. G. Corneil, B. Dalton, and M. Habib. LDFS-based certifying algorithm for the minimum path cover problem on cocomparability graphs. *SIAM Journal on Computing*, 42(3):792–807, 2013.

[17] D. G. Corneil, S. Olariu, and L. Stewart. Asteroidal triple-free graphs. *SIAM Journal on Discrete Mathematics*, 10(3):399–430, 1997.

[18] D. Coudert, A. Nusser, and L. Viennot. Enumeration of far-apart pairs by decreasing distance for faster hyperbolicity computation. *ACM Journal of Experimental Algorithmics*, 27:1.15:1–1.15:29, 2022.

[19] B. Das Gupta, M. Karpinski, N. Mobasheri, and F. Yahyanejad. Effect of Gromov-hyperbolicity parameter on cuts and expansions in graphs and some algorithmic implications. *Algorithmica*, 80(2):772–800, 2018.

[20] J. Davies and R. McCarty. Circle graphs are quadratically χ-bounded. *Bulletin of the London Mathematical Society*, 53(3):673–679, 2021.

[21] R. P. Dilworth. A decomposition theorem for partially ordered sets. *Annals of Mathematics*, 51(1):161–166, 1950.

[22] É. Diot, M. Radovanović, N. Trotignon, and K. Vušković. The (theta, wheel)-free graphs Part I: only-prism and only-pyramid graphs. *Journal of Combinatorial Theory, Series B*, 143:123–147, 2020.

[23] M. Francis, P. Hell, and J. Stacho. Forbidden structure characterization of circular-arc graphs and a certifying recognition algorithm. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2014)*, pages 1708–1727. SIAM, 2014.

[24] M. Gromov. Hyperbolic groups. In *Essays in group theory*, pages 75–263. Springer, 1987.

[25] L. Jaffke, O. Kwon, and J. A. Telle. Mim-width I. induced path problems. *Discrete Applied Mathematics*, 278:153–168, 2020. Eighth Workshop on Graph Classes, Optimization, and Width Parameters.

[26] J. M. Keil, J.S.B Mitchell, D. Pradhan, and M. Vatshelle. An algorithm for the maximum weight independent set problem on outerstring graphs. *Computational Geometry*, 60:19–25, 2017.

[27] A. Kosowski, B. Li, N. Nisse, and K. Suchan. K-chordal graphs: From cops and robber to compact routing via treewidth. *Algorithmica*, 72(3):758–777, 2015.

[28] J. Kratochvíl. String graphs. I. The number of critical nonstring graphs is infinite. *Journal of Combinatorial Theory, Series B*, 52(1):53–66, 1991.

[29] A. Rok and B. Walczak. Outerstring graphs are χ-bounded. *SIAM Journal on Discrete Mathematics*, 33(4):2181–2199, 2019.

[30] Y. Shavitt and T. Tankel. On the curvature of the internet and its usage for overlay construction and distance estimation. In *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM 2004)*. IEEE, 2004.

[31] M. Thiessen and T. Gaertner. Active learning of convex halfspaces on graphs. In *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS 2021)*, volume 34, pages 23413–23425. Curran Associates, Inc., 2021.

[32] N. Trotignon. Private communication, 2022.

[33] K. Vušković. The world of hereditary graph classes viewed through Truemper configurations. *Surveys in Combinatorics 2013*, 409:265, 2013.

[34] J. A. Walter and H. Ritter. On interactive visualization of high-dimensional data using the hyperbolic plane. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (SIGKDD 2002)*, pages 123–132, 2002.

[35] M. E. Watkins and D. M. Mesner. Cycles and connectivity in graphs. *Canadian Journal of Mathematics*, 19:1319–1328, 1967.

[36] Y. Wu and C. Zhang. Hyperbolicity and chordality of a graph. *The Electronic Journal of Combinatorics*, 18(1):P43, 2011.