# Obstructions to Faster Diameter Computation: Asteroidal Sets

**Guillaume Ducoffe**
National Institute of Research and Development in Informatics, Bucharest, Romania
University of Bucharest, Romania

## Abstract

An *extremity* is a vertex such that the removal of its closed neighbourhood does not increase the number of connected components. Let $Ext_\alpha$ be the class of all connected graphs whose *quotient graph* obtained from modular decomposition contains no more than $\alpha$ pairwise nonadjacent extremities. Our main contributions are as follows. First, we prove that the diameter of every $m$-edge graph in $Ext_\alpha$ can be computed in deterministic $\mathcal{O}(\alpha^3 m^{3/2})$ time. We then improve the runtime to $\mathcal{O}(\alpha^3 m)$ for bipartite graphs, to $\mathcal{O}(\alpha^5 m)$ for triangle-free graphs, $\mathcal{O}(\alpha^3 \Delta m)$ for graphs with maximum degree $\Delta$, and more generally to linear for all graphs with bounded clique-number. Furthermore, we can compute an additive $+1$-approximation of all vertex eccentricities in deterministic $\mathcal{O}(\alpha^2 m)$ time. This is in sharp contrast with general $m$-edge graphs for which, under the Strong Exponential Time Hypothesis (SETH), one cannot compute the diameter in $\mathcal{O}(m^{2-\epsilon})$ time for any $\epsilon > 0$.

As important special cases of our main result, we derive an $\mathcal{O}(m^{3/2})$-time algorithm for exact diameter computation within *dominating pair* graphs of diameter at least six, and an $\mathcal{O}(k^3 m^{3/2})$-time algorithm for this problem on graphs of *asteroidal number* at most $k$. Both results extend prior works on exact and approximate diameter computation within AT-free graphs. To the best of our knowledge, this is also the first deterministic subquadratic-time algorithm for computing the diameter within the subclasses of: chordal graphs of bounded leafage (generalizing the interval graphs), $k$-moplex graphs and $k$-polygon graphs (generalizing the permutation graphs) for any fixed $k$. We end up presenting an improved algorithm for chordal graphs of bounded asteroidal number, and a partial extension of our results to the larger class of all graphs with a *dominating target* of bounded cardinality. Our time upper bounds in the paper are shown to be essentially optimal under plausible complexity assumptions.

Our approach is purely combinatorial, that differs from most prior recent works in this area which have relied on geometric primitives such as Voronoi diagrams or range queries. On our way, we uncover interesting connections between the diameter problem, Lexicographic Breadth-First Search, graph extremities and the asteroidal number.

**2012 ACM Subject Classification** Theory of computation → Design and analysis of algorithms; Theory of computation → Graph algorithms analysis

**Keywords and phrases** Diameter computation, Asteroidal number, LexBFS

**Digital Object Identifier** 10.4230/LIPIcs.IPEC.2022.10

**Related Version** *Full Version*: https://arxiv.org/abs/2209.12438

**Funding** This work was supported by project PN-19-37-04-01 "New solutions for complex problems in current ICT research fields based on modelling and optimization", funded by the Romanian Core Program of the Ministry of Research and Innovation (MCI) 2019-2022. This work was also supported by a grant of the Ministry of Research, Innovation and Digitalization, CCCDI - UEFISCDI, project number PN-III-P2-2.1-PED-2021-2142, within PNCDI III.

## 1 Introduction

For any undefined graph terminology, see [9]. All graphs considered in this paper are undirected, simple (i.e., without loops nor multiple edges) and connected, unless stated otherwise. Given a graph $G = (V, E)$, let $n = |V|$ and $m = |E|$. For every vertices

$u$ and $v$, let $d_G(u,v)$ be their distance (minimum number of edges on a $uv$-path) in $G$. Let $e_G(u)$ denote the eccentricity of vertex $u$ (maximum distance to any other vertex). We sometimes omit the subscript if the graph $G$ is clear from the context. Finally, let $diam(G) = \max_{u,v \in V} d(u,v) = \max_{u \in V} e(u)$ be the diameter of $G$.

Computing the diameter is important in a variety of network applications such as in order to estimate the maximum latency in communication systems [32] or to identify the peripheral nodes in some complex networks with a core/periphery structure [60]. On $n$-vertex $m$-edge graphs, this can be done in $\mathcal{O}(nm)$ time by running a BFS from every vertex. This runtime is quadratic in the number $m$ of edges, even for sparse graphs (with $m \leq c \cdot n$ edges for some $c$), and therefore it is too prohibitive for large graphs with millions of vertices and sometimes billions of edges. Using Seidel's algorithm [71], the diameter of any $n$-vertex graph can be also computed in $\mathcal{O}(n^{\omega+o(1)})$ time, with $\omega$ the square matrix multiplication exponent. If $\omega = 2$, then this is almost linear-time for dense graphs, with $m \geq c \cdot n^2$ edges for some constant $c$ (currently, it is only known that $\omega < 2.37286$ [2]). However, for sparse graphs, this is still in $\Omega(m^2)$. In [69], Roditty and Vassilevska Williams proved that assuming the Strong Exponential-Time Hypothesis of Impagliazzo, Paturi and Zane [57], the diameter of $n$-vertex graphs with $n^{1+o(1)}$ edges *cannot* be computed in $\mathcal{O}(n^{2-\epsilon})$ time, for any $\epsilon > 0$. Therefore, breaking the quadratic barrier for diameter computation is likely to require additional graph structure, even for sparse graphs. In this paper, we make progress in this direction.

Let us call an algorithm truly subquadratic if it runs in $\mathcal{O}(m^{2-\epsilon})$ time on $m$-edge graphs, for some fixed positive $\epsilon$. Over the last decades, the existence of a truly subquadratic (often linear-time) algorithm for the diameter problem was proved for many important graph classes [1, 12, 14, 16, 18, 23, 29, 33, 34, 36, 37, 42, 38, 39, 40, 41, 44, 45, 49, 66]. This has culminated in some interesting connections between faster diameter computation algorithms and Computational Geometry, e.g., see the use of Voronoi diagrams for computing the diameter of planar graphs [16, 49], and of data structures for range queries in order to compute all eccentricities within bounded treewidth graphs [1, 14, 17], bounded clique-width graphs [41], or even proper minor-closed graph classes [44]. However, this type of geometric approach usually works only if certain Helly-type properties hold for the graph classes considered [3, 4, 11, 22, 39, 43, 44]. Beyond that, the finer-grained complexity of the diameter problem is much less understood, with only a few graph classes for which truly subquadratic algorithms are known [10]. The premise of this paper is that the *asteroidal number* could help in finding several new positive cases for diameter computation.

## Related work

Recall that an independent set in a graph $G$ is a set of pairwise non-adjacent vertices. An asteroidal set in a graph $G$ is an independent set $A$ with the additional property that, for every vertex $a \in A$, there exists a path between any two remaining vertices of $A \setminus \{a\}$ that does not contain $a$ nor any of its neighbours in $G$. Let the asteroidal number of $G$ be the largest cardinality of its asteroidal sets. The graphs of asteroidal number at most two are sometimes called *AT-free graphs*, and they generalize interval graphs, permutation graphs and co-comparability graphs amongst other subclasses [27]. It is worth mentioning here that all the aforementioned subclasses have unbounded treewidth and clique-width. The properties of AT-free graphs have been thoroughly studied in the literature [5, 8, 13, 15, 24, 26, 27, 46, 50, 51, 53, 56, 61, 62, 72], and some of these properties were generalized to the graphs of bounded asteroidal number [28, 58, 59]. In particular, as far as we are concerned here, there is a simple linear-time algorithm for computing a vertex in any AT-free graph whose eccentricity is within one of the diameter [23]. However, it has been only recently that a truly subquadratic algorithm for *exact* diameter computation within this class was presented [42]. This algorithm runs in deterministic $\mathcal{O}(m^{3/2})$ time on $m$-edge AT-free graphs, and it is combinatorial – that means, roughly, it does not rely on fast matrix multiplication algorithms or other algebraic techniques. In fact both algorithms from [23] and [42] are based on specific properties of *LexBFS orderings* for the AT-free graphs[1]. Roughly, the algorithm from [42] starts computing a dominating shortest path. In doing so, the search for a diametral vertex can be restricted to the closed neighbourhood of any one end of this path. However, in general this neighbourhood might be very large. The key procedure of the algorithm consists in further pruning out the neighbourhood so that it reduces to a clique. Then, we are done executing a BFS from every vertex in this clique. Both the computation of a dominating shortest path and the pruning procedure of the algorithm are taking advantage of the existence of a linear structure for AT-free graphs, that can be efficiently uncovered by using a double-sweep LexBFS [24]. Unfortunately, this linear structure no more exists for graphs of asteroidal number $\geq 3$.

## Contributions

The structure of graphs of bounded asteroidal number, and its relation to LexBFS, is much less understood than for AT-free graphs. Therefore, extending the known results for the diameter problem on AT-free graphs to the more general case of graphs of bounded asteroidal number is quite challenging. Doing just that is our main contribution in the paper. In fact, we prove even more strongly that only some types of large asteroidal sets need to be excluded in order to obtain a faster diameter computation algorithm.

More specifically, an *extremity* is a vertex such that the removal of its closed neighbourhood leaves the graph connected, see [59, 63]. Note that every subset of pairwise nonadjacent extremities forms an asteroidal set. A module in a graph $G = (V, E)$ is a subset of vertices $X$ such that every vertex of $V \setminus X$ is either adjacent to every of $X$ or nonadjacent to every of $X$. It is a strong module if it does not overlap any other module of $G$. Finally, the *quotient graph* of $G$ is the induced subgraph obtained by keeping one vertex in every inclusionwise maximal strict subset of $V$ which is a strong module of $G$ (see also Sec. 2 for a more detailed discussion about the modular decomposition of a graph). It is known that except in a few degenerate cases, the diameter of $G$ always equals the diameter of its quotient graph [29]. **We are interested in the maximum number of pairwise nonadjacent extremities in the quotient graph**, that according to the above is always a lower bound for the asteroidal number. See [59, Fig. 1] for an example where it is smaller than the asteroidal number.

Throughout the paper, let $Ext_\alpha$ denote the class of all graphs whose quotient graph contains no more than $\alpha$ pairwise nonadjacent extremities.

▶ **Theorem 1.** *For every graph $G = (V, E) \in Ext_\alpha$, we can compute estimates $\bar{e}(u)$, $u \in V$, in deterministic $\mathcal{O}(\alpha^2 m)$ time so that $e(u) \geq \bar{e}(u) \geq e(u)-1$ for every vertex $u$. In particular, we can compute a vertex whose eccentricity is within one of the diameter. Moreover, the exact diameter of $G$ can be computed in deterministic $\mathcal{O}(\alpha^3 m^{3/2})$ time.*

Let us now sketch the main lines of our approach toward proving Theorem 1. First, we replace the input graph by its quotient graph, that can be done in linear time [74]. Then, we compute $\mathcal{O}(\alpha)$ shortest paths with one common end-vertex $c$, the union of which is a

[1] It was also shown in [23] that there exist AT-free graphs such that a multi-sweep LexBFS fails in computing their diameter. Therefore, we need to further process the LexBFS orderings of AT-free graphs, resp. of graphs of bounded asteroidal number, to output their diameter.

dominating set (to be compared with the dominating shortest path computed in [42] for the AT-free graphs). For that, we prove interesting new relations between graph extremities and LexBFS, but only for graphs that are prime for modular decomposition (this is why we need to consider the quotient graph). Roughly, our algorithm computes $\mathcal{O}(\alpha)$ pairwise nonadjacent extremities, i.e., the other end-vertices of the shortest-paths than $c$, by repeatedly executing a modified LexBFS. We stress that our procedure is more complicated than a multi-sweep LexBFS due to the need to avoid getting stuck between two mutually distant extremities. In doing so, the search for a diametral vertex can be now restricted to the closed neighbourhoods of only $\mathcal{O}(\alpha^2)$ vertices (namely, to the $\mathcal{O}(\alpha)$ furthest vertices from $c$ on every shortest path). However, unlike what has been done in [42] for AT-free graphs, we failed in further pruning out each neighbourhood to a clique. Instead, we present a new procedure which given a vertex $u$ outputs a vertex in its closed neighbourhood of maximum eccentricity. This is done by iterating on some extremities at maximum distance from vertex $u$. Therefore, a key to our analyses in this paper is the number of extremities in a graph. We provide several bounds on this number. In doing so, our runtime for exact diameter computation can be improved to $\mathcal{O}(\alpha^3 m)$ time for the bipartite graphs, and more generally to linear time for every graph in $Ext_\alpha$ of constant clique number. We present some more alternative time bounds for our Theorem 1 in Sec. 5.2.

### Matching (Conditional) Lower bounds

The algorithm of Theorem 1 is combinatorial. In [42], the classic problem of detecting a simplicial vertex within an $n$-vertex graph is reduced in $\mathcal{O}(n^2)$ time to the diameter problem on $\mathcal{O}(n)$-vertex AT-free graphs. The best known combinatorial algorithm for detecting a simplicial vertex in an $n$-vertex graph runs in $\mathcal{O}(n^3)$ time. In the same way, in [23], Corneil et al. proved an equivalence between the problem of deciding whether an AT-free graph has diameter at most two and a disjoint sets problem which has been recently studied under the name of high-dimensional OV [30]. The high-dimensional OV problem can be reduced to Boolean Matrix Multiplication, which is conjectured not to be solvable in $\mathcal{O}(n^{3-\epsilon})$ time, for any $\epsilon > 0$, using a combinatorial algorithm [75]. It is open whether high-dimensional OV can be solved faster than Boolean Matrix Multiplication. Therefore, due to both reductions from [42] and [22], the existence of an $\mathcal{O}(f(\alpha)m^{3/2-\epsilon})$-time combinatorial algorithm for diameter computation within $Ext_\alpha$, for some function $f$ and for some $\epsilon > 0$, would be a significant algorithmic breakthrough.

### Applications to some graph classes

Let us review some interesting subclasses of graphs of $Ext_\alpha$, for some constant $\alpha$, for which to the best of our knowledge the best-known deterministic algorithm for diameter computation until this paper has been the brute-force $\mathcal{O}(nm)$-time algorithm.

A circle graph is the intersection graph of chords in a cycle. For every $k \geq 2$, a $k$-polygon graph is the intersection graph of chords in a convex $k$-polygon where the ends of each chord lie on two different sides. Note that the $k$-polygon graphs form an increasing hierarchy of all the circle graphs, and that the 2-polygon graphs are exactly the permutation graphs. Recently [40], an almost linear-time algorithm was proposed which computes a +2-approximation of the diameter of any $k$-polygon graph, for any fixed $k$. By [73], every $k$-polygon graph has asteroidal number at most $k$. Therefore, for the ***k*-polygon graphs**, we obtain an improved +1-approximation in linear time, and the first truly subquadratic algorithm for exact diameter computation.

A chordal graph is a graph with no induced cycle of length more than three. Chordal graphs are exactly the intersection graphs of a collection of subtrees of a host tree [48]. We call such a representation a tree model. The leafage of a chordal graph is the smallest number of leaves amongst its tree models. In particular, the chordal graphs of leafage at most two are exactly the interval graphs, which are exactly the AT-free chordal graphs. More generally, every chordal graph of leafage at most $k$ also has asteroidal number at most $k$ [65]. In [42], a randomized $\mathcal{O}(km\log^2 n)$-time algorithm was presented in order to compute the diameter of chordal graphs of asteroidal number at most $k$. Our Theorem 1 provides a deterministic alternative, but at the price of a higher runtime. Even more strongly, by combining the ideas of Theorem 1 with some special properties of chordal graphs, we were able to improve the runtime to $\mathcal{O}(km)$ – see our Theorem 34. Previously, such as result was only known for interval graphs [66].

A *moplex* in a graph is a module inducing a clique and whose neighbourhood is a minimal separator (the notions of module and minimal separator are recalled in Sec. 2). Moplexes are strongly related to LexBFS; indeed, every vertex last visited during a LexBFS is in a moplex [6]. This has motivated some recent studies on $k$-moplex graphs, a.k.a., the graphs with at most $k$ moplexes. In particular, every $k$-moplex graph has asteroidal number at most $k$ [8, 31]. Hence, our results in this paper can be applied to the ***k*-moplex graphs**.

Finally, a *dominating pair* consists of two vertices $x$ and $y$ such that every $xy$-path is a dominating set. Note that every AT-free graph contains a dominating pair [27]. A dominating pair graph (for short, DP graph) is one such that every connected induced subgraph contains a dominating pair. The family $(K_n^+)_{n\geq 4}$ of DP graphs in [68, Sec. 4.1] shows that for every $\alpha \geq 2$, there exists a DP graph which is not in $Ext_\alpha$. However, we here prove that every DP graph with diameter at least six is in $Ext_2$ – see our Lemma 16. In doing so, we obtain a deterministic $\mathcal{O}(m^{3/2})$-time algorithm which, given an $m$-edge DP graph, either computes its diameter or asserts that its diameter is $\leq 5$. We left open whether the diameter of DP graphs can be computed in truly subquadratic time.

### Organization of the paper

We give the necessary graph terminology for this paper in Sec. 2. Then, in Sec. 3 we present some properties of graph extremities which, to our knowledge, have not been noticed before our work. In particular if a graph is prime for modular decomposition, then there always exists a diametral path whose both ends are extremities of the graph. We think these results could be helpful in future studies on the diameter problem (for other graph classes), and in order to better understand the relevant graph structure to be considered for fast diameter computation. We complete Sec. 3 with additional properties of extremities for graphs of bounded asteroidal number and for DP graphs. In Sec. 4, we relate extremities to the properties of Lexicographic Breadth-First Search. Doing so, we design a general framework in order to compute extremities under various constraints. We prove Theorem 1 in Sec. 5, then we discuss some of its extensions in Sec. 6. We conclude this paper and propose some open questions in Sec. 7.

Due to lack of space, most proofs are either omitted or postponed to the appendix.

## 2 Preliminaries

We introduce in this section the necessary graph terminology for our proofs. Let $G = (V, E)$ be a graph. For any vertex $v \in V$, let $N_G(v) = \{u \in V \mid uv \in E\}$ be its (open) neighbourhood and let $N_G[v] = N_G(v) \cup \{v\}$ be its closed neighbourhood. Similarly, for any vertex-subset

$S \subseteq V$, let $N_G[S] = \bigcup_{v \in S} N_G[v]$ and let $N_G(S) = N_G[S] \setminus S$. For any vertices $u$ and $v$, we call a subset $S \subseteq V$ a $uv$-separator if $u$ and $v$ are in separate connected components of $G \setminus S$. A minimal $uv$-separator is an inclusion-wise minimal $uv$-separator. We call a subset $S$ a (minimal) separator if it is a (minimal) $uv$-separator for some vertices $u$ and $v$. Alternatively, a full component for $S$ is a connected component $C$ of $G \setminus S$ such that $N_G(C) = S$. It is known [52] that $S$ is a minimal separator if there exist at least two full components for $S$.

### Distances

Recall that the distance $d_G(u, v)$ between two vertices $u$ and $v$ equals the minimum number of edges on a $uv$-path. Let the interval $I_G(u, v) = \{w \in V \mid d_G(u, v) = d_G(u, w) + d_G(w, v)\}$ contain all vertices on a shortest $uv$-path. Furthermore, for every $\ell \geq 0$, let $N_G^{\ell}[u] = \{v \in V \mid d_G(u, v) \leq \ell\}$ be the ball of center $u$ and radius $\ell$ in $G$. We recall that the eccentricity of a vertex $v \in V$ is defined as $e_G(v) = \max_{u \in V} d_G(u, v)$. We sometimes omit the subscript if the graph $G$ is clear from the context. Let $F(v) = \{u \in V \mid d(u, v) = e(v)\}$ be the set of vertices most distant to vertex $v$. The diameter and the radius of $G$ are defined as $diam(G) = \max_{v \in V} e(v)$ and $rad(G) = \min_{v \in V} e(v)$, respectively. We call $(x, y)$ a diametral pair if $d(x, y) = diam(G)$.

### Modular decomposition

Two vertices $u, v \in V$ are twins if we have $N(u) \setminus \{v\} = N(v) \setminus \{u\}$. A twin class is a maximal vertex-subset of pairwise twins. More generally, a module is a vertex-subset $M \subseteq V$ such that $N(x) \setminus M = N(y) \setminus M$ for any $x, y \in M$. Note that any twin class is also a module. We call $G$ *prime* if its only modules are: $\emptyset, V$ and $\{v\}$ for every $v \in V$ (trivial modules). A module $M$ is *strong* if it does not overlap any other module, i.e., for any module $M'$ of $G$, either one of $M$ or $M'$ is contained in the other or $M$ and $M'$ do not intersect. We denote by $\mathcal{M}(G)$ the family of all inclusion wise maximal strong modules of $G$ that do not contain all the vertices of $G$. Finally, the *quotient graph* of $G$ is the graph $G'$ with vertex-set $\mathcal{M}(G)$ and an edge between every two $M, M' \in \mathcal{M}(G)$ such that every vertex of $M$ is adjacent to every vertex of $M'$. The following well-known result is due to Gallai:

▶ **Theorem 2** ([47]). *For an arbitrary graph $G$ exactly one of the following conditions is satisfied.*
1. *$G$ is disconnected;*
2. *its complement $\overline{G}$ is disconnected;*
3. *or its quotient graph $G'$ is prime for modular decomposition.*

For general graphs, there is a tree representation of all the modules in a graph, sometimes called the modular decomposition, that can be computed in linear time [74]. Note that since we only consider connected graphs, only the two last items of Theorem 2 are relevant to our study. Moreover, it is easy to prove that if the complement of a graph $G$ is disconnected, then we have $diam(G) \leq 2$. Therefore, the following result is an easy byproduct of Gallai's theorem for modular decomposition [47]:

▶ **Lemma 3** (cf. Theorem 14 in [29]). *Computing the diameter (resp., all eccentricities) of any graph $G$ can be reduced in linear time to computing the diameter (resp., all eccentricities) of its quotient graph $G'$.*

An important observation for what follows is that, if a graph $G$ belongs to $Ext_\alpha$ for some $\alpha$, then so does its quotient graph $G'$. Hence, we may only consider *prime* graphs in $Ext_\alpha$.

### Hyperbolicity

The hyperbolicity of a graph $G$ [54] is the smallest half-integer $\delta \geq 0$ such that, for any four vertices $u, v, w, x$, the two largest of the three distance sums $d(u,v)+d(w,x)$, $d(u,w)+d(v,x)$, $d(u,x)+d(v,w)$ differ by at most $2\delta$. In this case we say that $G$ is $\delta$-hyperbolic. To quote [34]: "As the tree-width of a graph measures its combinatorial tree-likeness, so does the hyperbolicity of a graph measure its metric tree-likeness. In other words, the smaller the hyperbolicity $\delta$ of $G$ is, the closer $G$ is to a tree metrically." We will use in what follows the following "tree-likeness" properties of hyperbolic graphs:

▶ **Lemma 4** ([20]). *If $G$ is $\delta$-hyperbolic, then $diam(G) \geq 2rad(G) - 4\delta - 1$.*

▶ **Lemma 5** (Proposition 3(c) in [21]). *Let $G$ be a $\delta$-hyperbolic graph and let $u, v$ be a pair of vertices of $G$ such that $v \in F(u)$. We have $e(v) \geq diam(G) - 8\delta \geq 2rad(G) - 12\delta - 1$.*

▶ **Lemma 6** ([20]). *Let $u$ be an arbitrary vertex of a $\delta$-hyperbolic graph $G$. If $v \in F(u)$ and $w \in F(v)$, then let $c \in I(v,w)$ be satisfying $d(c,v) = \lfloor d(v,w)/2 \rfloor$. We have $e(c) \leq rad(G)+5\delta$.*

## 3 Properties of graph extremities

We present several simple properties of graph extremities in what follows. In Sec. 3.1, we give bounds on the number of extremities in a graph. Then, we show in Sec. 3.2 that, for computing the vertex eccentricities of a prime graph (and so, its diameter), it is sufficient to only consider its so-called extremities. In Sec. 3.3, we relate the location of the extremities in a graph to the one of an arbitrary dominating target. We state in Sec. 3.4 a relationship between extremities and the hyperbolicity of a graph (this result easily follows from [59]). In Sec. 3.5, additional properties of extremities in some graph classes are discussed.

### 3.1 Bounds on the Number of Graph extremities

Every non-complete prime graph has at least two extremities [59]. The remainder of this section is devoted to proving an upper bound on the number of extremities in a prime graph. Unfortunately, there may be up to $\Theta(n)$ extremities in an $n$-vertex graph, even if it is AT-free. See the construction of [23, Fig. 2] for an example. It is worth mentioning this example also has clique-number equal to $\Theta(n)$. Our general upper bounds in what follows show that only dense prime graphs may have $\Omega(n)$ extremities.

▶ **Lemma 7.** *If $G \in Ext_\alpha$ is prime, then the number of its extremities is at most:*

- *$\alpha \cdot \chi(G)$, where $\chi(G)$ denotes the chromatic number of $G$;*
- *$R(\alpha+1, \omega(G)+1) - 1$, where $\omega(G)$ is the clique number of $G$, and $R(\cdot,\cdot)$ is a Ramsey number.*

*In particular, it is in $\mathcal{O}(\alpha\sqrt{m})$.*

**Proof.** Let us denote by $q$ the number of extremities of $G$, and let $H$ be induced by all the extremities. Note that $H$ is not necessarily connected. Since we assume that $G \in Ext_\alpha$, the independence number of $H$ is at most $\alpha$. In this situation, $q < R(\alpha+1, \omega(G)+1)$ (otherwise, either $H$ would contain an independent set of size $\alpha+1$, or $H$ and so, $G$, would contain a clique of size $\omega(G)+1$). Since the chromatic number of $H$ is at most $\chi(G)$, we also have that $H$ can be partitioned in at most $\chi(G)$ independent sets, and so, $q \leq \alpha \cdot \chi(G)$. In particular, $q = \mathcal{O}(\alpha\sqrt{m})$ because $\chi(G) = \mathcal{O}(\sqrt{m})$ for any graph $G$. ◀

Let $G \in Ext_\alpha$ be prime, with $q$ extremities. Note that, using $R(s,t) = \mathcal{O}(t^{s-1})$ for any fixed $s$, we get that $q = \mathcal{O}(\alpha^{\omega(G)+1})$. That is in $\mathcal{O}(\alpha^3)$ for triangle-free graphs. For graphs of constant chromatic number, the bound of Lemma 7 is linear in $\alpha$. In particular, $q = \mathcal{O}(\alpha\Delta)$ for the graphs of maximum degree $\Delta$, and $q = \mathcal{O}(\alpha)$ for bipartite graphs.

## 3.2 Relationships with the diameter

To the best of our knowledge, the following relation between extremities and vertex eccentricities has not been noticed before:

▶ **Lemma 8.** *If $x$ is a vertex of a prime graph $G = (V,E)$ with $|V| \geq 3$, then there exists an extremity $y$ of $G$ such that $d(x,y) = e(x)$.*

*In particular, for every $y' \in F(x)$, there is an extremity $y \in F(x)$ so that $d(y,y') \leq 2$.*

The proof of Lemma 8 is postponed to Appendix A. We observe that a slightly weaker version of Lemma 8 could be also deduced from Lemma 19 (proved in the next section).

▶ **Corollary 9.** *If $G = (V,E)$ is prime and $diam(G) \geq 2$, then there exist extremities $x,y$ such that $d(x,y) = diam(G)$.*

Overall if we were given the $q$ extremities of a prime graph $G$, then by Lemma 8, we could compute all eccentricities (and so, the diameter) in $\mathcal{O}(qm)$ time. By Lemma 7, this runtime is in $\mathcal{O}(\alpha m^{3/2})$ for the graphs within $Ext_\alpha$, which is subquadratic for any fixed $\alpha$. This bound can be improved to linear time for any graph of $Ext_\alpha$ with constant clique number. However, the best-known algorithms for computing the extremities run in $\mathcal{O}(nm)$ time and in $\mathcal{O}(n^{2.79})$ time [63], respectively. Furthermore, computing the extremities is at least as hard as triangle detection, even for AT-free graphs [63]. We leave as an open problem whether there exists a truly subquadratic algorithm for computing all extremities in a graph.

## 3.3 Relationships with Dominating targets

A dominating target in a graph $G$ is a vertex-subset $D$ with the property that any connected subgraph of $G$ containing all of $D$ must be a dominating set. Dominating targets of cardinality two have been studied under the different name of dominating pairs. In particular, every AT-free graph contains a dominating pair [27].

▶ **Lemma 10** (special case of Theorems 6 and 7 in [59]). *If $G$ is prime, then every inclusion-wise maximal subset of pairwise nonadjacent extremities in $G$ is a dominating target.*

We have the following relation between extremities and dominating targets:

▶ **Lemma 11.** *If $D$ is a dominating target of a graph $G$ (not necessarily prime), then every extremity of $G$ is contained in $N[D]$. In particular, there are at most $(\Delta+1) \cdot |D|$ extremities, where $\Delta$ denotes the maximum degree of $G$.*

**Proof.** Suppose by contradiction the existence of some extremity $v \notin N[D]$. Then, $H = G \setminus N[v]$ is a connected subgraph of $G$ that contains all of $D$ but such that $v$ has no neighbour in $V(H)$. The latter contradicts that $D$ is a dominating target of $G$. ◀

It follows from both Lemma 8 and Lemma 11 that, for any dominating target $D$ in a prime graph, there is a diametral vertex in $N[D]$. We slightly strengthen this result, as follows. The following simple lemmas also generalize prior results on AT-free graphs [23] and graphs with a dominating pair [42].

▶ **Lemma 12.** *If $D$ is a dominating target, then $F(x) \cap N[D] \neq \emptyset$ for any vertex $x$. In particular if $F(x) \cap D = \emptyset$, then $F(x) \subseteq N(D)$.*

**Proof.** We may assume that $F(x) \cap D = \emptyset$ (else, we are done). In this situation, let $y \in F(x)$ be arbitrary. Then, let $H$ be the union of shortest $xu$-paths, for every $u \in D$. Since $H$ is a connected subgraph, we have $y \in N[H]$. In particular, there is a shortest $ux$-path $P$, for some fixed $u \in D$, such that $y \in N[P]$. Observe that $y \notin V(P)$ (otherwise, $d(x,y) \leq d(x,u)$, and therefore $u \in F(x)$). So, let $y^* \in V(P) \cap N(y)$. If $y^* \neq u$, then $d(x,y) \leq d(x,y^*) + 1 \leq (d(x,u) - 1) + 1 = d(x,u)$, and therefore, $u \in F(x)$. A contradiction. As a result, $y \in N(u) \setminus D \subseteq N(D)$. ◀

▶ **Corollary 13.** *If $D$ is a dominating target of a graph $G$, and no vertex of $D$ is in a diametral pair, then $x, y \in N(D)$ for every diametral pair $(x, y)$.*

This above Corollary 13 suggests the following strategy in order to compute the diameter of a prime graph $G$. First, we compute a small dominating target $D$. Then, we search for a diametral vertex within the neighbourhood of each of its $|D|$ vertices. If $G \in Ext_\alpha$, then according to Lemma 10 there always exists such a $D$ with $\mathcal{O}(\alpha)$ vertices. However, we are not aware of any truly subquadratic algorithm for computing this dominating target. By Lemma 10, it is sufficient to compute a maximal independent set of extremities, but then we circle back to the aforementioned problem of computing all extremities in a graph. In Sec. 5 in the paper, we prove that we needn't compute a dominating target in full in order to determine what the diameter is. Specifically, we may only compute a strict subset $D' \subset D$ of a dominating target (for that, we use the techniques presented in Sec. 4). However, the price to pay is that while doing so, we also need to consider a bounded number of vertices outside of $N[D']$ and their respective neighbourhoods. Hence, the number of neighbourhoods to be considered grows to $\mathcal{O}(\alpha^2)$. This will be our starting approach for proving Theorem 1.

## 3.4 Relationships with Hyperbolicity

Recall the definition of $\delta$-hyperbolic graphs in Sec. 2. In a $\delta$-hyperbolic graph $G$, an "almost central" vertex of eccentricity $\leq rad(G) + c\delta$, for some $c > 0$, can be computed in linear time, using a double-sweep BFS (see Lemma 6). Then, according to Lemma 4, any diametral vertex must at a distance $\geq rad(G) - c'\delta$, for some $c' > 0$, to this almost central vertex. Roughly, we wish to combine these properties with the computation of some subset $D'$ of a small dominating target (see Sec. 3.3) in order to properly locate some neighbourhood that contains a diametral vertex. For that, we need to prove here that graphs in $Ext_\alpha$ are $\delta$-hyperbolic for some $\delta$ depending on $\alpha$. Namely:

▶ **Lemma 14.** *Every graph $G \in Ext_\alpha$ is $(3\alpha - 1)$-hyperbolic.*

## 3.5 Extremities in some Graph classes

We complete this section with the following inclusions between graph classes.

▶ **Lemma 15.** *Every graph $G$ of asteroidal number $k$ belongs to $Ext_k$.*

While this above Lemma 15 trivially follows from the respective definitions of $Ext_k$ and the asteroidal number, the following result is less immediate:

▶ **Lemma 16.** *Every DP graph $G = (V, E)$ of diameter at least six belongs to $Ext_2$.*

The proof of Lemma 16 is postponed to Appendix B. Lemma 16 does not hold for diameter-five DP graphs, as it can be shown from the example in [68, Fig. 6], that has three pairwise nonadjacent extremities. Moreover, for every $n \geq 4$, there exists a diameter-two DP graph $K_n^+$ with $n$ pairwise nonadjacent extremities [68]. We left open whether, for any $d \in \{3, 4, 5\}$, there exists some constant $\alpha(d) \geq 3$ such that all diameter-$d$ DP graphs belong to $Ext_{\alpha(d)}$.

# 4 A framework for computing extremities

We identify sufficient conditions for computing an independent set of extremities (not necessarily a maximal one). To the best of our knowledge, before this work there was no faster known algorithm for computing *one* extremity than for computing all such vertices. We present a simple linear-time algorithm for this problem on prime graphs – see Sec. 4.2. Then, we refine our strategy in Sec. 4.3 so as to compute one extremity avoided by some fixed connected subset. This procedure is key to our proof of Theorem 1, for which we need to iteratively compute extremities, and connect those to some pre-defined vertex $c$ using shortest paths, until we obtain a connected dominating set. In fact, our approach in Sec. 4.3 works under more general conditions which we properly state in Def. 21. Our main algorithmic tool here is LexBFS, of which we first recall basic properties in Sec. 4.1.

## 4.1 LexBFS

The Lexicographic Breadth-First Search (LexBFS) is a standard algorithmic procedure, that runs in linear time [70]. We give a pseudo-code in Algorithm 1. Note that we can always enforce a start vertex $u$ by assigning to it an initial non empty label. Then, for a given graph $G = (V, E)$ and a start vertex $u$, $LexBFS(u)$ denotes the corresponding execution of LexBFS. Its output is a numbering $\sigma$ over the vertex-set (namely, the reverse of the ordering in which vertices are visited during the search). In particular, if $\sigma(i) = x$, then $\sigma^{-1}(x) = i$.

**Algorithm 1** LexBFS [70].

**Require:** A graph $G = (V, E)$.
assign the label $\emptyset$ to each vertex;
**for** $i = n$ **to** 1 **do**
  pick an unnumbered vertex $x$ with the largest label in the lexicographic order;
  **for all** unnumbered neighbours $y$ of $x$ **do**
    add $i$ to $label(y)$;
  $\sigma(i) \leftarrow x$ /* *number $x$ by $i$* */;

We use some notations from [24]. Fix some LexBFS ordering $\sigma$. Then, for any vertices $u$ and $v$, $u \prec v$ if and only if $\sigma^{-1}(u) < \sigma^{-1}(v)$. Similarly, $u \preceq v$ if either $u = v$ or $u \prec v$. Let us define $N_{\prec}(v) = \{u \in N(v) \mid u \prec v\}$ and $N_{\succ}(v) = \{u \in N(v) \mid v \prec u\}$. Let also $\lhd$ denote the lexicographic total order over the sets of LexBFS labels. For every vertices $u$ and $v$, $u \preceq v$, let $\lambda(u, v)$ be the label of vertex $u$ when vertex $v$ was about to be numbered. We stress that $\lambda(u, v) \unlhd \lambda(v, v)$ (i.e., the vertex selected to be numbered at any step has maximum label for the lexicographic order). Furthermore, a useful observation is that $\lambda(u, v)$ is just the list of all neighbours of $u$ which got numbered before $v$, ordered by decreasing LexBFS number. In particular, for any $u \preceq v$ we have $\lambda(u, v) = \lambda(v, v)$ if and only if $N_{\succ}(v) \subseteq N(u)$. We often use this latter property in our proofs.

▶ **Lemma 17** (monotonicity property [24]). *Let $a, b, c$ and $d$ be vertices of a graph $G$ such that: $a \preceq c$, $b \preceq c$ and $c \prec d$. If $\lambda(a, d) \lhd \lambda(b, d)$, then $\lambda(a, c) \lhd \lambda(b, c)$.*

▶ **Corollary 18.** *Let $x, y, z$ be vertices of a graph $G$ such that: $x \preceq y \preceq z$, and $\lambda(x,z) = \lambda(z,z)$. Then, $\lambda(y,z) = \lambda(z,z)$.*

## 4.2 Finding one extremity

It turns out that finding *one* extremity is simple, namely:

▶ **Lemma 19.** *If $G = (V, E)$ is a prime graph with $|V| \geq 3$, and $\sigma$ is any LexBFS order, then $v = \sigma(1)$ is an extremity of $G$.*

**Proof.** Suppose by contradiction $G \setminus N[v]$ to be disconnected. Let $u = \sigma(n)$ be the start vertex of the LexBFS ordering, and let $C$ be any component of $G \setminus N[v]$ which does not contain vertex $u$. We denote by $z = \sigma(i)$, $n > i > 1$ the vertex of $C$ with maximum LexBFS number. By maximality of $z$, we obtain that $N_{\succ}(z) \subseteq N(v)$. In particular, $\lambda(v,z) = \lambda(z,z)$. Then, let $M = \{w \in V \mid v \preceq w \preceq z\}$. Since we have $\lambda(v,z) = \lambda(z,z)$, by Corollary 18 we also get $\lambda(w,z) = \lambda(z,z)$ for every $w \in M$. But this implies $N(w) \setminus M = N_{\succ}(z)$ for each $w \in M$, therefore $M$ is a nontrivial module of $G$. A contradiction. ◀

An alternative proof of Lemma 19 could be deduced from the work of Berry and Bordat on the relations between moplexes and LexBFS [7]. However, to the best of our knowledge, Lemma 19 has not been proved before.

## 4.3 Generalization

Before generalizing Lemma 19, we need to introduce a few more notions and terminology.

▶ **Definition 20.** *Let $G = (V, E)$ be a graph and let $u, v, w \in V$ be pairwise independent. We write $u \perp_w v$ if and only if $u, v$ are in separate connected components of $G \setminus N[w]$.*

A vertex $w$ intercepts a path $P$ if $N[w] \cap V(P) \neq \emptyset$, and it misses $P$ otherwise. We can check that if $u \perp_w v$, then $w$ intercepts all $uv$-paths, and conversely if $u \not\perp_w v$ and $uw, vw \notin E$, then $w$ misses a $uv$-path.

▶ **Definition 21.** *Let $u$ and $S$ be, respectively, a vertex and a vertex-subset of some graph $G = (V, E)$. We call $S$ a $u$-transitive set if, for every $x \in S$ and $y \in V$ nonadjacent, $x \perp_y u \Longrightarrow y \in S$.*

Next, we give examples of $u$-transitive sets.

▶ **Lemma 22.** *If $H$ is a connected subgraph of a graph $G$, then its closed neighbourhood $N[H]$ is $u$-transitive for every $u \in V(H)$. In particular, every ball centered at $u$ and of arbitrary radius is $u$-transitive.*

**Proof.** Let $x \in N[H]$ and $y$ satisfy $x \perp_y u$. Since $x \in N[H]$ and $u \in V(H)$, there exists a $xu$-path $P$ of which all vertices except maybe $x$ are in $H$. Furthermore, since we assume that $x \perp_y u$, and so $x$ and $y$ are nonadjacent, we obtain $y \in N[P \setminus \{x\}] \subseteq N[H]$. ◀

For an example of non-connected $u$-transitive set, we may simply consider three pairwise nonadjacent vertices $u, v, w$ in a cycle. Then, $S = \{v, w\}$ is $u$-transitive.

We are now ready to state the following key lemma:

▶ **Lemma 23.** *Let $G = (V, E)$ be a prime graph with $|V| \geq 3$, let $u \in V$ be arbitrary and let $S \subseteq V$ be $u$-transitive. If $V \neq S \cup N[u]$, then we can compute in linear time an extremity $v \notin S \cup N[u]$ such that $d(u, v)$ is maximized.*

The proof of Lemma 23 is postponed to Appendix C.

## 5 Proof of Theorem 1

In Sec. 5.1, we present a linear-time algorithm for computing a vertex whose eccentricity is within one of the true diameter. This part of the proof is simpler, and it gives some intuition for our exact diameter computation algorithm, which we next present in Sec. 5.2.

### 5.1 Approximation algorithm

▶ **Theorem 24.** *For every graph $G = (V, E) \in Ext_\alpha$, we can compute in deterministic $\mathcal{O}(\alpha^2 m)$ time estimates $e(v) \geq \bar{e}(v) \geq e(v) - 1$ for every vertex $v$.*

**Proof.** We may assume $G$ to be prime by Lemma 3. Furthermore, let us assume that $|V| \geq 3$. We subdivide the algorithm in three main phases.

- First, we compute some shortest path by using a double-sweep LexBFS. More specifically, let $x_1$ be the last vertex numbered in a LexBFS. Let $x_2$ be the last vertex numbered in a LexBFS($x_1$). We compute a vertex $c \in I(x_1, x_2)$ so that $d(c, x_1) = \lfloor d(x_1, x_2)/2 \rfloor$. Then, let $P_1$ (resp., $P_2$) be an arbitrary shortest $cx_1$-path (resp., $cx_2$-path). – Note that for AT-free graphs (but not necessarily in our case), the shortest $x_1x_2$-path $P_1 \cup P_2$ is dominating [24]. –
- Second, we set $H := P_1 \cup P_2$. While $H$ is not a dominating set of $G$, we compute an extremity $x_i \notin N[H]$ and we add an arbitrary shortest $cx_i$-path to $H$. – We stress that such an extremity $x_i$ always exists due to $H$ being connected and therefore $N[H]$ being $c$-transitive (see Lemma 22) and by Lemma 23.– Let $x_3, x_4, \ldots, x_t$ denote all the extremities computed. By construction, $H$ is the union of $t$ shortest paths $P_1, P_2, \ldots, P_t$ with one common end-vertex $c$.
- Finally, for every $1 \leq i \leq t$, let the subset $U_i$ be composed of the $\min\{d(x_i, c)+1, 66\alpha-19\}$ closest vertices to $x_i$ in $P_i$ (including $x_i$ itself). Let $U = \bigcup_{i=1}^{t} U_i$. For every vertex $v \in V$, we set $\bar{e}(v) := \max\{d(u, v) \mid u \in U\}$.

**Correctness.** Let $v \in V$ be arbitrary. Since $H$ is a dominating set of $G$, some vertex $u \in H$, in the closed neighbourhood of any vertex of $F(v)$, must satisfy $d(u, v) \geq e(v) - 1$. In order to prove correctness of our algorithm, it suffices to prove the existence of one such vertex in $U$. For that, let $\delta$ be chosen such that $G$ is $\delta$-hyperbolic. By Lemma 6, $e(c) \leq rad(G) + 5\delta$. Furthermore if $u \in H$ satisfies $N[u] \cap F(v) \neq \emptyset$, then by Lemma 5, $e(u) \geq 2rad(G) - 12\delta - 2$. In particular, $d(u, c) \geq rad(G) - 17\delta - 2$ (else, $e(u) \leq d(u, c) + e(c) \leq 2rad(G) - 12\delta - 3$). For $1 \leq i \leq t$ such that $u \in V(P_i)$, since $P_i$ is a shortest $x_ic$-path of length at most $e(c) \leq rad(G) + 5\delta$, we get that $u$ must be one of the $(rad(G)+5\delta) - (rad(G)-17\delta-2)+1 = 22\delta+3$ closest vertices to $x_i$. By Lemma 14, $\delta \leq 3\alpha - 1$, therefore $22\delta + 3 \leq 66\alpha - 19$.

**Complexity.** Recall that the first phase of the algorithm consists in a double-sweep LexBFS. Hence, it can be done in linear time. Then, at every step of the second phase we must decide whether $H$ is a dominating set of $G$, that can be done in linear time. If $H$ is not a dominating set, then we compute an extremity $x_i \notin N[H]$, that can also be done in linear time by Lemma 23. We further compute an arbitrary shortest $x_ic$-path $P_i$, that can be done in linear time using BFS. Overall, the second phase takes $\mathcal{O}(tm)$ time, with $t$ the number of extremities computed. Finally, in the third phase, we need to execute $\mathcal{O}(\alpha)$ BFS for every shortest path $P_1, P_2, \ldots P_t$, that takes $\mathcal{O}(\alpha tm)$ time.

By Lemma 19, both $x_1$ and $x_2$ are also extremities. We observe that $x_1x_2 \notin E$ (else, since $x_2 \in F(x_1)$, $x_1$ would be universal, thus contradicting either that $G$ is prime or $|V| \geq 3$). By construction, $x_3, x_4, \ldots, x_t$ are pairwise nonadjacent, and they are also nonadjacent to both $x_1$ and $x_2$. Altogether combined, we obtain that $x_1, x_2, \ldots, x_t$ are pairwise nonadjacent extremities. As a result, $t \leq \alpha$. It implies that the total runtime is in $\mathcal{O}(\alpha^2 m)$. ◀

## 5.2 Exact computation

The following general result, of independent interest, is the cornerstone of Theorem 26:

▶ **Lemma 25.** *Let $u$ be a vertex in a prime graph $G = (V, E)$. If $G$ has $q$ extremities, then we can compute in $\mathcal{O}(qm)$ time the value $\ell(u) = \max\{e(x) \mid x \in N[u]\}$, and a $x \in N[u]$ of eccentricity $\ell(u)$. This is $\mathcal{O}(\alpha m^{3/2})$ time if $G \in Ext_\alpha$.*

The proof of Lemma 25 involves several cumbersome intermediate lemmas. We postpone the proof of Lemma 25 to the end of this section, proving first our main result:

▶ **Theorem 26.** *For every graph $G = (V, E) \in Ext_\alpha$, we can compute its diameter in deterministic $\mathcal{O}(\alpha^3 m^{3/2})$ time.*

**Proof (Assuming Lemma 25).** By Lemma 3, we may assume $G$ to be prime. Let us further assume that $|V| \geq 3$, and so that $G$ cannot have a universal vertex.

**Algorithm.** We subdivide the procedure in three main phases, the two first of which being common to both Theorems 24 and 26.

- Let $x_1$ be the last vertex numbered in a LexBFS. We execute a LexBFS with start vertex $x_1$. Let $x_2$ be the last vertex numbered in a LexBFS($x_1$). We compute a $c \in I(x_1, x_2)$ so that $d(c, x_1) = \lfloor d(x_1, x_2)/2 \rfloor$. Let $P_1, P_2$ be shortest $x_1c$-path and $x_2c$-path, respectively.
- We set $H := P_1 \cup P_2$. While $H$ is not a dominating set of $G$, we compute a new extremity $x_i \notin N[H]$, an arbitrary shortest $x_ic$-path $P_i$, then we set $H := H \cup P_i$. In what follows, we denote by $x_1, x_2, \ldots, x_t$ the extremities computed in the two first phases of the algorithm. Let $P_1, P_2, \ldots, P_t$ be the corresponding shortest paths, whose union equals $H$.
- Finally, for every $1 \leq i \leq t$, let $U_i \subseteq V(P_i)$ contain the $\min\{d(x_i, c) + 1, 42\alpha - 11\}$ closest vertices to $x_i$. Let $L_i := \max\{\ell(u_i) \mid u_i \in U_i\}$. We output $L := \max_{1 \leq i \leq t} L_i$ as the diameter value.

**Complexity.** The first phase of the algorithm can be done in $\mathcal{O}(m)$ time, and its second phase in $\mathcal{O}(tm)$ time, with $t$ the number of extremities computed. During the third and final phase, we need to apply Lemma 25 $\mathcal{O}(\alpha)$ times for every shortest path $P_1, P_2, \ldots, P_t$. Hence, the above algorithm runs in $\mathcal{O}(t\alpha qm)$ time, with $q$ the total number of extremities of $G$. By Lemma 7, we have that $q = \mathcal{O}(\alpha\sqrt{m})$. See Theorem 24 for a proof that $t \leq \alpha$. As a result, the above algorithm runs in $\mathcal{O}(\alpha^3 m^{3/2})$ time.

**Correctness.** Let us consider an arbitrary diametral pair $(u, v)$. Since $H$ is a dominating set of $G$, we have $N[u] \cap H \neq \emptyset$. Let $u^* \in N[u] \cap H$, and observe that $e(u^*) \geq diam(G) - 1$. Then we claim that if $u^* \in V(P_i)$ for some $1 \leq i \leq t$, we must have $u^* \in U_i$. The proof is similar to what we did for Theorem 24. Specifically, let us choose $\delta$ such that $G$ is $\delta$-hyperbolic. By Lemma 6, $e(c) \leq rad(G) + 5\delta$. Furthermore, if $u^* \in H$ satisfies $e(u^*) \geq diam(G) - 1$, then by Lemma 4, $e(u^*) \geq 2rad(G) - 4\delta - 2$. In particular, $d(u^*, c) \geq rad(G) - 9\delta - 2$ (else, $e(u^*) \leq d(u^*, c) + e(c) \leq 2rad(G) - 4\delta - 3 < diam(G) - 1$). For $1 \leq i \leq t$ such that $u^* \in V(P_i)$, since $P_i$ is a shortest $x_i c$-path of length at most $e(c) \leq rad(G) + 5\delta$, we get that $u^*$ must be one of the $(rad(G) + 5\delta) - (rad(G) - 9\delta - 2) + 1 = 14\delta + 3$ closest vertices to $x_i$. By Lemma 14, $\delta \leq 3\alpha - 1$, therefore $14\delta + 3 \leq 42\alpha - 11$. In this situation, $L \geq L_i \geq \ell(u^*) = e(u) = diam(G)$. Combined with the trivial inequality $L \leq diam(G)$, it implies that $L = diam(G)$. ◀

The actual runtime of Theorem 26 is $\mathcal{O}(\alpha^2 qm)$, where $q$ denotes the number of extremities. By Lemma 7, this is in $\mathcal{O}(\alpha^3 m)$ for bipartite graphs, $\mathcal{O}(\alpha^3 \Delta m)$ for graphs with maximum degree $\Delta$ and in $\mathcal{O}(\alpha^5 m)$ for triangle-free graphs. More generally, this is linear time for all graphs of $Ext_\alpha$ with bounded clique number.

The remainder of this section is devoted to the proof of Lemma 25. The key idea here is that for every vertex $u$, there is an extremity $v \in F(u)$ such that $\max\{d(v,x) \mid x \in N[u]\} = \ell(u)$. Therefore, in order to achieve the desired runtime for Lemma 25, it would be sufficient to iterate over all extremities of $G$ that are contained in $F(u)$. However, due to our inability to compute all extremities in subquadratic time, we are bound to use Lemma 23 for only computing some of these extremities. Therefore, throughout the algorithm, we further need to grow some $u$-transitive set whose vertices must be carefully selected so that they can be discarded from the search space. The next Lemmas 27 and 28 are about the construction of this $u$-transitive set.

▶ **Lemma 27.** *Let $u$ and $v$ be vertices of a graph $G$ such that $v \in F(u)$, and let $X = \{x \in N[u] \mid d(x,v) = e(u)\}$. In $\mathcal{O}(m)$ time we can construct a set $Y$ where:*
- *$v \in Y$; $Y$ is $u$-transitive;*
- *$d(y,x) \leq d(v,x)$ for every $y \in Y$ and $x \in X$.*

We now complete Lemma 27, as follows:

▶ **Lemma 28.** *Let $u$ and $v$ be vertices of a graph $G$ such that $v \in F(u)$, and $d(x,v) \leq e(u)$ for every $x \in N[u]$. In $\mathcal{O}(m)$ time we can construct a set $S'$ where:*
- *$v \in S'$; $S'$ is $u$-transitive;*
- *$d(s,x) \leq e(u)$ for every $s \in S'$ and $x \in N[u]$.*

Using Lemma 28 we end up the section proving Lemma 25:

**Proof of Lemma 25.** We may assume that $|V| \geq 3$ and therefore (since $G$ is prime) that $u$ is not a universal vertex. We search for a $v \in V$ at a distance $\ell(u)$ from some vertex of $N[u]$. Note that we have $e(u) \leq \ell(u) \leq e(u) + 1$. In particular we only need to consider the vertices $v \in F(u)$. We next describe an algorithm that iteratively computes some pairs $(v_0, S_0), (v_1, S_1), \ldots$ such that, for any $i \geq 0$: `(i)` $v_i \in F(u)$ is an extremity; and `(ii)` $S_i$ is $u$-transitive. We continue until either $d(v_i, x) = e(u) + 1$ for some $x \in N(u)$ or $F(u) \subseteq S_i$.

If $i = 0$ then, let $v_0$ be the last vertex numbered in a LexBFS($u$), that is an extremity by Lemma 19. Otherwise ($i > 0$), let $v_i$ be the output of Lemma 23 applied to $u$ and $S = S_{i-1}$. Furthermore, being given the extremity $v_i$, let $S'_i$ be computed as in Lemma 27 applied to $u$ and $v_i$. Let $S_i = S'_i \cup S_{i-1}$ (with the convention that $S_{-1} = \emptyset$). We stress that $S_i$ is $u$-transitive since it is the union of two $u$-transitive sets.

By Lemma 28, all vertices in $S_i$ can be safely discarded since they cannot be at distance $e(u) + 1$ from any vertex of $N[u]$. Furthermore, since for every $i$ we have $v_i \in S_i \setminus S_{i-1}$, the sequence $(F(u) \setminus S_{i-1})_{i \geq 0}$ is strictly decreasing with respect to set inclusion. Each step $i$ takes linear time, and the total number of steps is bounded by the number of extremities in $F(u)$. For a graph within $Ext_\alpha$, this is in $\mathcal{O}(\alpha\sqrt{m})$ according to Lemma 7. ◀

# 6 Extensions

Our algorithmic framework in Sec. 5.2 can be refined in several ways. We present such refinements for the larger class of all graphs having a dominating target of bounded cardinality. Some general results are first discussed in Sec. 6.1 before we address the special case of chordal graphs in Sec. 6.2.

## 6.1 More results on dominating targets

We start with the following observation:

▶ **Lemma 29.** *If a graph $G$ contains a dominating target of cardinality at most $k$, then so does its quotient graph $G'$.*

By Lemma 29, we may only consider in what follows *prime* graphs with a dominating target of bounded cardinality.

By $K_{1,t}$, we mean the star with $t$ leaves. A graph is $K_{1,t}$-free if it has no induced subgraph isomorphic to $K_{1,t}$.

▶ **Lemma 30.** *Every $K_{1,t}$-free graph $G$ with a dominating target of cardinality at most $k$ belongs to $Ext_{k(t-1)}$.*

▶ **Corollary 31.** *We can compute the diameter of $K_{1,t}$-free graphs with a dominating target of cardinality at most $k$ in deterministic $\mathcal{O}((kt)^3 m^{3/2})$ time.*

Let us now consider graphs with a dominating target of cardinality at most $k$ and bounded maximum degree $\Delta$. These graphs are $K_{1,\Delta+1}$-free and therefore, according to Corollary 31, we can compute their diameter in deterministic $\mathcal{O}((k\Delta)^3 m^{3/2})$ time. We improve this runtime to quasi linear, while also decreasing the dependency on $\Delta$, namely:

▶ **Theorem 32.** *For every $G = (V, E)$ with a dominating target of cardinality at most $k$ and maximum degree $\Delta$, we can compute its diameter in deterministic $\mathcal{O}(k^3 \Delta m \log n)$ time.*

Since under SETH we cannot compute the diameter of graphs with a dominating edge in subquadratic time [42], the dependency on $\Delta$ in Theorem 32 is conditionally optimal.

The proof of Theorem 32 is postponed to Appendix D. Being given a vertex $c$ of small eccentricity, our main difficulty here is to compute efficiently a small number of shortest-paths starting from $c$ whose union is a dominating set of $G$. This could be done by computing a dominating target of small cardinality. However, our framework in the prior Sec. 4 only allows us to compute extremities, and not directly a dominating target. Our strategy consists in including in some candidate subset all the neighbours of the extremities that are computed by our algorithm. By using Lemma 11, we can bound the size of this candidate subset by an $\mathcal{O}(k\Delta)$. Then, by using a greedy set cover algorithm, we manage to compute from this candidate subset a set of $\mathcal{O}(k \log n)$ shortest-paths that cover all but $\mathcal{O}(k\Delta \log n)$ vertices. We apply the prior techniques of Sec. 5 to all these shortest-paths (calling upon Lemma 25), while for the $\mathcal{O}(k\Delta \log n)$ vertices that they miss we compute their eccentricities directly.

## 6.2 Chordal graphs

By [42, Theorems 6 & 9], we can decide in linear time the diameter of chordal graphs with a dominating pair (resp., with a dominating triple) if the former value is at least 4 (resp., at least 10). It was also asked in [42] whether for every $k \geq 4$, there exists a threshold $d_k$ such that the diameter of chordal graphs with a dominating target of cardinality at most $k$ can be decided in truly subquadratic time if the former value is at least $d_k$. We first answer to this question in the affirmative:

▶ **Theorem 33.** *If $G = (V, E)$ is chordal, with a dominating target of cardinality at most $k$, and such that $diam(G) \geq 4$, then we can compute its diameter in deterministic $\mathcal{O}(km)$ time.*

Roughly, we lower the runtime of Theorem 33 to linear by avoiding calling upon Lemma 25. Specifically, we prove that whenever $diam(G) \geq 4$ there is always one of the $\leq k$ extremities which we compute whose eccentricity equals the diameter. The analysis of Theorem 33 is involved, as it is based on several nontrivial properties of chordal graphs [19, 23, 25, 64, 70].

We complete the above Theorem 33 with an improved algorithm for computing the diameter of chordal graphs with bounded asteroidal number. For that, we combine the algorithmic scheme of Theorem 33 with a previous approach from [42]. More specifically, for the special case $rad(G) = 2$, we considerably revisit a prior technique from [42, Proposition 2], for split graphs, so that is also works on diameter-three chordal graphs.

▶ **Theorem 34.** *For every chordal graph $G = (V, E)$ of asteroidal number at most $k$, we can compute its diameter in deterministic $\mathcal{O}(km)$ time.*

## 7 Conclusion

We generalized most known algorithmic results for diameter computation within AT-free graphs to the graphs within the much larger class $Ext_\alpha$, for any $\alpha \geq 2$. The AT-free graphs, but also every DP graph with diameter at least six, belong to $Ext_2$. Furthermore, for every $\alpha \geq 3$, every graph of asteroidal number $\alpha$ (and so, every $\alpha$-polygon graph, every $\alpha$-moplex graph and every chordal graph of leafage equal to $\alpha$) belong to $Ext_\alpha$.

We left open whether our results could be extended to the problem of computing exactly all eccentricities. The algorithm in [42] for the AT-free graphs can also be applied to this more general setting. Specifically, in every AT-free graph, there exist three vertices $u, v, w$ such that every vertex $x$ is at distance $e(x)$ from a vertex in $N[u] \cup N[v] \cup N[w]$. An algorithm is proposed in [42] in order to prune out each of the three neighbourhoods to a clique while keeping the latter property. Now, being given a graph from $Ext_\alpha$, using our techniques we could also compute a union of $\mathcal{O}(\alpha^2)$ neighbourhoods such that every vertex $x$ is at maximum distance from some vertex in it. However, insofar our approach only allows us to extract from each neighbourhood a vertex of maximum eccentricity, that is less powerful than the pruning method in [42].

It could be also interesting to give bounds on the number of extremities (resp., of pairwise nonadjacent extremities) in other graph classes. Finally, we left as an open problem what the complexity of computing the diameter is within the graphs which can be made AT-free by removing at most $k$ vertices. For stars and their subdivisions it suffices to remove one vertex, and therefore this class of graphs has unbounded asteroidal number even for $k = 1$.

## References

1 A. Abboud, V. Vassilevska Williams, and J. Wang. Approximation and fixed parameter subquadratic algorithms for radius and diameter in sparse graphs. In *Proceedings of the Twenty-seventh Annual ACM-SIAM symposium on Discrete Algorithms (SODA)*, pages 377–391. SIAM, 2016.

2 J. Alman and V. Vassilevska Williams. A refined laser method and faster matrix multiplication. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 522–539. SIAM, 2021.

3 H.-J. Bandelt, A. Dählmann, and H. Schütte. Absolute retracts of bipartite graphs. *Discrete Applied Mathematics*, 16(3):191–215, 1987.

4 H.-J. Bandelt and E. Pesch. Efficient characterizations of $n$-chromatic absolute retracts. *Journal of Combinatorial Theory, Series B*, 53(1):5–31, 1991.

5 J. Beisegel. Characterising AT-free graphs with BFS. In *International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, pages 15–26. Springer, 2018.

6 A. Berry and J.-P. Bordat. Separability generalizes Dirac's theorem. *Discrete Applied Mathematics*, 84(1-3):43–53, 1998.

7 A. Berry and J.-P. Bordat. Local LexBFS properties in an arbitrary graph. In *Proceedings of Journéees Informatiques Messines (JIM 2000)*, 2000.

8 A. Berry and J.-P. Bordat. Asteroidal triples of moplexes. *Discrete Applied Mathematics*, 111(3):219–229, 2001.

9 J. A. Bondy and U. S. R. Murty. *Graph theory*. Springer London, 2008.

10 M. Borassi, P. Crescenzi, and L. Trevisan. An axiomatic and an average-case analysis of algorithms and heuristics for metric properties of graphs. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 920–939. SIAM, 2017.

11 N. Bousquet and S. Thomassé. VC-dimension and Erdős–Pósa property. *Discrete Mathematics*, 338(12):2302–2317, 2015.

12 A. Brandstädt, V. Chepoi, and F.F. Dragan. The algorithmic use of hypertree structure and maximum neighbourhood orderings. *Discrete Applied Mathematics*, 82(1-3):43–77, 1998.

13 A. Brandstädt, P. Fičur, A. Leitert, and M. Milanič. Polynomial-time algorithms for weighted efficient domination problems in AT-free graphs and dually chordal graphs. *Information Processing Letters*, 115(2):256–262, 2015.

14 K. Bringmann, T. Husfeldt, and M. Magnusson. Multivariate Analysis of Orthogonal Range Searching and Graph Distances. *Algorithmica*, pages 1–24, 2020.

15 H. Broersma, T. Kloks, D. Kratsch, and H. Müller. Independent sets in asteroidal triple-free graphs. *SIAM Journal on Discrete Mathematics*, 12(2):276–287, 1999.

16 S. Cabello. Subquadratic algorithms for the diameter and the sum of pairwise distances in planar graphs. *ACM Transactions on Algorithms (TALG)*, 15(2):1–38, 2018.

17 S. Cabello and C. Knauer. Algorithms for graphs of bounded treewidth via orthogonal range searching. *Computational Geometry*, 42(9):815–824, 2009.

18 V. Chepoi, F. Dragan, and Y. Vaxès. Center and diameter problems in plane triangulations and quadrangulations. In *Symposium on Discrete Algorithms (SODA'02)*, pages 346–355, 2002.

19 V. Chepoi and F.F. Dragan. A linear-time algorithm for finding a central vertex of a chordal graph. In *European Symposium on Algorithms*, pages 159–170. Springer, 1994.

20 V. Chepoi, F.F. Dragan, B. Estellon, M. Habib, and Y. Vaxès. Diameters, centers, and approximating trees of $\delta$-hyperbolic geodesic spaces and graphs. In Monique Teillaud, editor, *Proceedings of the 24th ACM Symposium on Computational Geometry (SoCG)*, pages 59–68. ACM, 2008.

21 V. Chepoi, F.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.

22 V. Chepoi, B. Estellon, and Y. Vaxès. Covering planar graphs with a fixed number of balls. *Discrete & Computational Geometry*, 37(2):237–244, 2007.

23 D. Corneil, F. Dragan, M. Habib, and C. Paul. Diameter determination on restricted graph families. *Discrete Applied Mathematics*, 113(2-3):143–166, 2001.

24 D. Corneil, S. Olariu, and L. Stewart. Linear time algorithms for dominating pairs in asteroidal triple-free graphs. *SIAM Journal on Computing*, 28(4):1284–1297, 1999.

**25** D.G. Corneil, F.F. Dragan, and E. Köhler. On the power of BFS to determine a graph's diameter. *Networks: An International Journal*, 42(4):209–222, 2003.

**26** D.G. Corneil, S. Olariu, and L. Stewart. A linear time algorithm to compute a dominating path in an AT-free graph. *Information Processing Letters*, 54(5):253–257, 1995.

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

**28** D.G. Corneil and J. Stacho. Vertex ordering characterizations of graphs of bounded asteroidal number. *Journal of Graph Theory*, 78(1):61–79, 2015.

**29** D. Coudert, G. Ducoffe, and A. Popa. Fully polynomial FPT algorithms for some classes of bounded clique-width graphs. *ACM Transactions on Algorithms (TALG)*, 15(3):1–57, 2019.

**30** M. Dalirrooyfard and J. Kaufmann. Approximation Algorithms for Min-Distance Problems in DAGs. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *48th International Colloquium on Automata, Languages, and Programming (ICALP 2021)*, volume 198 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 60:1–60:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

**31** C. Dallard, R. Ganian, M. Hatzel, M. Krnc, and M. Milanič. Graphs with two moplexes. In *Proceedings of LAGOS'21*, volume 195 of *Procedia Computer Science*, pages 248–256. Elsevier, 2021.

**32** J. De Rumeur. *Communications dans les réseaux de processeurs*. Masson Paris, 1994.

**33** F.F. Dragan. HT-graphs: centers, connected r-domination and Steiner trees. *The Computer Science Journal of Moldova*, 1(2):64–83, 1993.

**34** F.F. Dragan, G. Ducoffe, and H. Guarnera. Fast Deterministic Algorithms for Computing All Eccentricities in (Hyperbolic) Helly Graphs. In Anna Lubiw and Mohammad R. Salavatipour, editors, *Algorithms and Data Structures (WADS)*, volume 12808 of *Lecture Notes in Computer Science*, pages 300–314. Springer, 2021.

**35** F.F. Dragan and E. Köhler. An approximation algorithm for the tree t-spanner problem on unweighted graphs via generalized chordal graphs. *Algorithmica*, 69(4):884–905, 2014.

**36** F.F. Dragan and F. Nicolai. LexBFS-orderings of distance-hereditary graphs with application to the diametral pair problem. *Discrete Applied Mathematics*, 98(3):191–207, 2000.

**37** F.F. Dragan, F. Nicolai, and A. Brandstädt. LexBFS-orderings and power of graphs. In Fabrizio d'Amore, Paolo Giulio Franciosa, and Alberto Marchetti-Spaccamela, editors, *Graph-Theoretic Concepts in Computer Science (WG)*, volume 1197 of *Lecture Notes in Computer Science*, pages 166–180. Springer, 1996.

**38** G. Ducoffe. A New Application of Orthogonal Range Searching for Computing Giant Graph Diameters. In *SOSA*, 2019.

**39** G. Ducoffe. Beyond Helly Graphs: The Diameter Problem on Absolute Retracts. In Łukasz Kowalik, Michał Pilipczuk, and Paweł Rzażewski, editors, *Graph-Theoretic Concepts in Computer Science (WG)*, pages 321–335. Springer International Publishing, 2021.

**40** G. Ducoffe. Isometric Embeddings in Trees and Their Use in Distance Problems. In Filippo Bonchi and Simon J. Puglisi, editors, *46th International Symposium on Mathematical Foundations of Computer Science (MFCS 2021)*, volume 202 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 43:1–43:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

**41** G. Ducoffe. Optimal centrality computations within bounded clique-width graphs. In *16th International Symposium on Parameterized and Exact Computation (IPEC)*, volume 214 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 16:1–16:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

**42** G. Ducoffe. The diameter of AT-free graphs. *Journal of Graph Theory*, 99(4):594–614, 2022.

**43** G. Ducoffe and F.F. Dragan. A story of diameter, radius, and (almost) Helly property. *Networks*, 77(3):435–453, 2021.

**44** G. Ducoffe, M. Habib, and L. Viennot. Diameter computation on $H$-minor free graphs and graphs of bounded (distance) VC-dimension. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1905–1922. SIAM, 2020.

**45** A. Farley and A. Proskurowski. Computation of the center and diameter of outerplanar graphs. *Discrete Applied Mathematics*, 2(3):185–191, 1980.

**46** F.V. Fomin, M. Matamala, E. Prisner, and I. Rapaport. AT-free graphs: linear bounds for the oriented diameter. *Discrete Applied Mathematics*, 141(1-3):135–148, 2004.

**47** T. Gallai. Transitiv orientierbare graphen. *Acta Math. Hungarica*, 18(1):25–66, 1967.

**48** F. Gavril. The intersection graphs of subtrees in trees are exactly the chordal graphs. *Journal of Combinatorial Theory, Series B*, 16(1):47–56, 1974.

**49** P. Gawrychowski, H. Kaplan, S. Mozes, M. Sharir, and O. Weimann. Voronoi diagrams on planar graphs, and computing the diameter in deterministic $\tilde{O}(n^{5/3})$ time. In *Symposium on Discrete Algorithms (SODA)*, pages 495–514. SIAM, 2018.

**50** P. Golovach, P. Heggernes, D. Kratsch, D. Lokshtanov, D. Meister, and S. Saurabh. Bandwidth on AT-free graphs. In *International Symposium on Algorithms and Computation (ISAAC)*, pages 573–582. Springer, 2009.

**51** P.A. Golovach, D. Paulusma, and E.J. van Leeuwen. Induced disjoint paths in AT-free graphs. In *Scandinavian Workshop on Algorithm Theory (SWAT)*, pages 153–164. Springer, 2012.

**52** M. C. Golumbic. *Algorithmic graph theory and perfect graphs*, volume 57. Elsevier, 2004.

**53** J. Gorzny and J. Huang. End-vertices of LBFS of (AT-free) bigraphs. *Discrete Applied Mathematics*, 225:87–94, 2017.

**54** M. Gromov. *Hyperbolic Groups*, pages 75–263. Springer, New York, NY, 1987.

**55** M. Habib, R. McConnell, C. Paul, and L. Viennot. Lex-BFS and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theoretical Computer Science*, 234(1-2):59–84, 2000.

**56** H. Hempel and D. Kratsch. On claw-free asteroidal triple-free graphs. *Discrete Applied Mathematics*, 121(1-3):155–180, 2002.

**57** R. Impagliazzo and R. Paturi. On the complexity of $k$-SAT. *Journal of Computer and System Sciences*, 62(2):367–375, 2001.

**58** T. Kloks, D. Kratsch, and H. Müller. Asteroidal sets in graphs. In *International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, pages 229–241. Springer, 1997.

**59** T. Kloks, D. Kratsch, and H. Müller. On the structure of graphs with bounded asteroidal number. *Graphs and Combinatorics*, 17(2):295–306, 2001.

**60** Dirk Koschützki, Katharina Anna Lehmann, Leon Peeters, Stefan Richter, Dagmar Tenfelde-Podehl, and Oliver Zlotowski. Centrality indices. In *Network Analysis*, pages 16–61. Springer, 2005.

**61** D. Kratsch and H. Müller. Colouring AT-free graphs. In *European Symposium on Algorithms (ESA)*, pages 707–718. Springer, 2012.

**62** D. Kratsch, H. Müller, and I. Todinca. Feedback vertex set on AT-free graphs. *Discrete Applied Mathematics*, 156(10):1936–1947, 2008.

**63** D. Kratsch and J. Spinrad. Between $O(nm)$ and $O(n^{\alpha})$. *SIAM Journal on Computing*, 36(2):310–325, 2006.

**64** R. Laskar and D. Shier. On powers and centers of chordal graphs. *Discrete Applied Mathematics*, 6(2):139–147, 1983.

**65** I.-J. Lin, T.A. McKee, and D.B. West. The leafage of a chordal graph. *Discussiones Mathematicae Graph Theory*, 18(1):23–48, 1998.

**66** S. Olariu. A simple linear-time algorithm for computing the center of an interval graph. *International Journal of Computer Mathematics*, 34(3-4):121–128, 1990.

**67** R. Paige and R. Tarjan. Three partition refinement algorithms. *SIAM Journal on Computing*, 16(6):973–989, 1987.

**68** N. Pržulj, D.G. Corneil, and E. Köhler. Hereditary dominating pair graphs. *Discrete Applied Mathematics*, 134(1-3):239–261, 2004.

69 L. Roditty and V. Vassilevska Williams. Fast approximation algorithms for the diameter and radius of sparse graphs. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing (STOC)*, pages 515–524, 2013.
70 D. Rose, R. Tarjan, and G. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5(2):266–283, 1976.
71 R. Seidel. On the all-pairs-shortest-path problem in unweighted undirected graphs. *J. Comput. Syst. Sci.*, 51(3):400–403, 1995.
72 J. Stacho. 3-colouring AT-free graphs in polynomial time. In *International Symposium on Algorithms and Computation (ISAAC)*, pages 144–155. Springer, 2010.
73 L. Stewart and R. Valenzano. On polygon numbers of circle graphs and distance hereditary graphs. *Discrete Applied Mathematics*, 248:3–17, 2018.
74 M. Tedder, D. Corneil, M. Habib, and C. Paul. Simpler linear-time modular decomposition via recursive factorizing permutations. In *ICALP*, pages 634–645. Springer, 2008.
75 V. Vassilevska Williams and R.R. Williams. Subcubic equivalences between path, matrix, and triangle problems. *Journal of the ACM (JACM)*, 65(5):1–38, 2018.

## A Proof of Lemma 8

The following lemma shall be used in our proofs:

▶ **Lemma 35** ([59]). *Let $S$ be a minimal separator for a prime graph $G = (V, E)$. For any component $C$ of $G \setminus S$, if $C$ does not contain an extremity of $G$, then $N(c) \cap S$ is a separator of $G \setminus C$ for every $c \in C$.*

Recall for what follows that a vertex is called universal if and only if all other vertices are adjacent to it.

**Proof of Lemma 8.** Since we assume that $|V| \geq 3$, we have that $x$ cannot be a universal vertex (otherwise, $V \setminus \{x\}$ would be a nontrivial module, thus contradicting that $G$ is prime). Let $y \in F(x)$ be arbitrary and we assume that $y$ is not an extremity. We shall replace $y$ by some extremity $y^*$ so that $d(x, y) = d(x, y^*) = e(x)$. First we observe that $x \notin N[y]$ because we assume that $x$ is not a universal vertex. Let $w$ be disconnected from $x$ in $G \setminus N[y]$, and let $S \subseteq N[y]$ be a minimal $wx$-separator of $G$ (obtained by iteratively removing vertices from $N[y]$ while $w$ and $x$ stay disconnected) – possibly, $y \notin S$. –

We continue with a useful property of the connected components of $G \setminus S$. Specifically, let $C$ be any connected component of $G \setminus S$ not containing vertex $x$. We claim that $d(x, c) = e(x)$ for each $c \in C$. Indeed, every shortest $xc$-path contains a vertex $s \in S$ and therefore, $d(x, y) \leq 1 + d(s, x) \leq d(c, s) + d(s, x) = d(c, x)$. In particular, $C \subseteq N(S)$ (otherwise, $d(x, y) < 2 + d(s, x) \leq d(x, c)$ for any $c \in C \setminus N(S)$ and any $s \in S$ that is on a shortest $cx$-path). It implies that, if $c \in C$ is an extremity, then $d(c, y) \leq 2$.

In what follows, let $X$ the connected component of $x$ in $G \setminus S$. Then, amongst all vertices of $G \setminus S$ in another connected component than $x$, let $a$ be minimizing $|N(a) \cap S|$ and let $A$ be its connected component in $G \setminus S$. Let us assume that $A$ does not contain an extremity of $G$ (else, we are done). By Lemma 35, $N(a) \cap S$ is a separator of $G \setminus A$. Let $b$ be separated from vertex $x$ in $G \setminus (A \cup N(a))$. We claim that $b \notin S$. In order to see that, we first need to observe that $X$ is a full component for $S$ (otherwise, $S$ could not be a minimal $wx$-separator). Therefore if $b \in S$, then the subset $X \cup \{b\}$ would be connected, thus contradicting that $x$ and $b$ are disconnected in $G \setminus (A \cup N(a))$. This proves our claim, and from now on we denote $B$ the connected component of $b$ in $G \setminus S$. Observe that $B \neq X$ (otherwise, $N(a) \cap S$ could not be a $bx$-separator in $G \setminus A$). We further claim that $N(B) \subseteq N(a)$. Indeed, recall that $X$ is a full component for $S$. Hence, if it were not the case that $N(B) \subseteq N(a)$ then the subset $X \cup B \cup (N(B) \setminus N(a))$ would be connected, thus contradicting that $x$ and $b$ are disconnected in $G \setminus (A \cup N(a))$.

By the above claim, we get $N(b') \cap S \subseteq N(B) \subseteq N(a) \cap S$, for every $b' \in B$. Thus, by minimality of $|N(a) \cap S|$, we obtain $N(b') \cap S = N(a) \cap S = N(B)$ for every $b' \in B$. But then, $B$ is a module of $G$, and therefore $B = \{b\}$ because $G$ is prime. Finally, suppose by contradiction $b$ is not an extremity. By repeating the exact same arguments for $b$ instead of $a$, we find another connected component $C = \{c\}$ of $G \setminus S$ so that: $c$ is separated from $x$ in $G \setminus (B \cup N(b)) = G \setminus N[b]$, and $N(c) = N(b) = N(a) \cap S$ (possibly, $C = A$ and $a = c$). However, it implies that $b$ and $c$ are twins, a contradiction. Overall, we may choose for our vertex $y^*$ either an extremity of $A$ (if there exists one) or vertex $b$. ◀

## B Proof of Lemma 16

**Proof of Lemma 16.** Since the property of being a DP graph is hereditary, the quotient graph of any DP graph is also a DP graph. In particular, it suffices to prove even more strongly that an arbitrary DP graph $G$ of diameter at least six (not necessarily prime) cannot contain three pairwise nonadjacent extremities. Suppose by contradiction the existence of three such extremities $u, v, w$. Let $(x, y)$ be a dominating pair. By Lemma 11, $u, v, w \in N[x] \cup N[y]$. Without loss of generality, let $u, v \in N(x)$.

We claim that $S = N(u) \cap N(v)$ is not a separator of $G$. Suppose by contradiction that it is the case. Let $A = N[u] \setminus N(v)$, $B = N[v] \setminus N(u)$ and $X = V \setminus (A \cup B \cup S)$. Since $S \subseteq N(u)$ and $u$ is an extremity of $G$, $B \cup X$ must be contained in some connected component of $G \setminus S$. But similarly, since $S \subseteq N(v)$ and $v$ is an extremity of $G$, $A \cup X$ must be also contained in some connected component of $G \setminus S$. As a result, $X = \emptyset$, and the only two components of $G \setminus S$ are $A$ and $B$. In particular, $w \in A \cup B \subseteq N[u] \cup N[v]$, that is a contradiction. Therefore, we proved as claimed that $S$ is not a separator of $G$.

We now claim that $u, v$ are still extremities in the subgraph $G \setminus S$. By symmetry, it suffices to prove the result for vertex $u$. Observing that removing all of $N[u] \setminus S$ leaves us with $G \setminus N[u]$, we are done because $u$ is an extremity of $G$. – However, please note that vertex $w$ may not be an extremity of $G \setminus S$. –

Since $G \setminus S$ is a DP graph, there exists a dominating pair $(x', y')$ in this subgraph. Again by Lemma 11 we have $u, v \in N[x'] \cup N[y']$. Without loss of generality, let $u \in N[x']$, $v \in N[y']$ (possibly, $u = x'$, resp. $v = y'$). Now, let $P = (z_0 = x', z_1, \ldots, z_\ell = y')$ be a shortest $x'y'$-path of $G \setminus S$. By construction, $P$ is a dominating path of $G \setminus S$. In order to derive a contradiction, we shall prove, using $P$, that $e(x) \leq 4$. Indeed, doing so, since $(x, y)$ is a dominating pair, we obtain that $diam(G) \leq e(x) + 1 \leq 5$, a contradiction. For that, let $t \in V$ be arbitrary. We may further assume $t \notin N[x]$. If $t \in S$, then $u, v \in N(t) \cap N(x)$, therefore $d(x, t) \leq 2$. From now on, let us assume that $t \notin S$. Consider some index $i$ such that $t \in N[z_i]$. Let $Q_u$ be an induced $xz_i$-path such that $V(Q_u) \subseteq \{x, u, z_0, z_1, \ldots, z_i\}$. In the same way, let $Q_v$ be an induced $xz_i$-path such that $V(Q_v) \subseteq \{x, v, z_\ell, z_{\ell-1}, \ldots, z_i\}$. Let us first assume that $V(Q_u) \cup V(Q_v)$ induces a cycle $C$. Then, the length of $C$ must be $\leq 6$ because $C$ is a DP graph and no cycle of length $\geq 7$ contains a dominating pair. As a result, $d(x, z_i) \leq 3$, and so $d(x, t) \leq 4$. For the remainder of the proof, we assume that there exists a chord in the cycle $C$ induced by $V(Q_u) \cup V(Q_v)$. Since the three of $P, Q_u, Q_v$ are induced paths, the only possible chords are: $uz_j$, for some $i + 1 \leq j \leq \ell$; or $vz_j$, for some $0 \leq j \leq i - 1$. By symmetry, let $uz_j$ be a chord of $C$. Since $P$ is a shortest $x'y'$-path of $G \setminus S$, and $u \in N[x']$, we obtain that $j \in \{0, 1, 2\}$. In particular, we obtain that $i \leq 1$, and so, $d(x, t) \leq 1 + d(x, z_i) \leq 2 + d(u, z_i) \leq 4$. ◀

# C Proof of Lemma 23

**Proof of Lemma 23.** We describe the algorithm before proving its correctness and analysing its runtime.

**Algorithm.** Let $\sigma$ be any LexBFS($u$) order of $G$, and let $w \notin S$ be minimizing $\sigma^{-1}(w)$. First we compute the maximum index $i$, $n > i \geq \sigma^{-1}(w)$, so that $\lambda(w, \sigma(i)) = \lambda(\sigma(i), \sigma(i)) = \lambda_i$. We set $j := 0$, $S_j := S$ and $M_j := \{v \notin S \mid w \preceq v \preceq \sigma(i)\}$. Then, while $|M_j| > 1$, we apply the following procedure:

- We partition $M_j$ into groups $A_j^1, A_j^2, \ldots, A_j^{p_j}$ so that two vertices are in the same group if and only if they have the same neighbours in $S_j$.
- Without loss of generality let $A_j^1$ be minimizing $|N(A_j^1) \cap S_j|$. We set $M_{j+1} = A_j^1$, $S_{j+1} = M_j \setminus M_{j+1}$.
- We set $j := j + 1$.

If $|M_j| = 1$ (end of the while loop), then we output the unique vertex $v \in M_j$.

**Correctness.** We prove by induction that the following three properties hold, for any $j \geq 0$: `(i)` $M_j$ is a module of $G \setminus S_j$; `(ii)` every $v \in M_j$ is a vertex of $V \setminus S$ such that $d(u, v)$ is maximized; and `(iii)` for every $v \in M_j$, either $v$ is an extremity of $G$ or every connected component $C$ of $G \setminus N[v]$ that does not contain vertex $u$ satisfies $C \subseteq M_j$.

First, we consider the base case $j = 0$. Recall that $M_0 = \{v \notin S \mid w \preceq v \preceq \sigma(i)\}$, where $i$ is the maximum index such that $\lambda(w, \sigma(i)) = \lambda(\sigma(i), \sigma(i)) = \lambda_i$. By Corollary 18 we have $\lambda(v, \sigma(i)) = \lambda_i$ for each $v \in M_0$, and so, $N_{\succ}(\sigma(i)) \subseteq N(v)$. The latter implies that $d(u, v) = d(u, \sigma(i))$ because $\sigma$ is a (Lex)BFS order. Equivalently, $d(u, v) = d(u, w)$, and by the minimality of $\sigma^{-1}(w)$ we have that $w$ is a vertex of $V \setminus S$ maximizing $d(u, w)$ (Property `(ii)`). Moreover, $N(v) \setminus (S \cup M_0) = N_{\succ}(\sigma(i)) \setminus S$ for each $v \in M_0$, therefore $M_0$ is a module of $G \setminus S$ (Property `(i)`). Now, let $v \in M_0$ be arbitrary. If $v$ is not an extremity of $G$, then let $C$ be any connected component of $G \setminus N[v]$ not containing vertex $u$. We claim that $C \subseteq M_0$. Indeed, suppose by contradiction $C \cap S \neq \emptyset$. By maximality of $d(u, v)$, we have $v \notin N[u]$ (for else, $V \setminus S \subseteq N[u]$). But then, we would get $s \perp_v u$ for any $s \in C \cap S$, and therefore by the definition of $S$ we should have $v \in S$. A contradiction. As a result we have $C \cap S = \emptyset$. Furthermore, we have $w \preceq c$ for each $c \in C$, that follows from the minimality of $\sigma^{-1}(w)$. Let $z \in C$ be maximizing $\sigma^{-1}(z)$. In order to prove that $C \subseteq M_0$, it now suffices to prove that $\sigma^{-1}(z) \leq i$. Suppose by contradiction that it is not the case. We first observe $N_{\succ}(z) \subseteq N(v)$ by maximality of $\sigma^{-1}(z)$. Therefore (since in addition, $v \preceq \sigma(i) \prec z$), $\lambda(v, z) = \lambda(z, z)$. Since we suppose $v \preceq \sigma(i) \prec z$, we also get by Corollary 18 that $\lambda(\sigma(i), z) = \lambda(z, z)$. Then, $N_{\succ}(z) \subseteq N_{\succ}(\sigma(i)) \subseteq N(w)$. However, it implies $\lambda(w, z) = \lambda(z, z)$, thus contradicting the maximality of $i$ ($< \sigma^{-1}(z)$) for this property.

Then, let us assume that $M_j, S_j$ satisfy all of properties `(i)`, `(ii)` and `(iii)`, and that $|M_j| > 1$. By construction, all vertices in $M_{j+1}$ have the same neighbours in $S_j$. Since $M_{j+1} \subset M_j$ and $M_j$ is a module of $G \setminus S_j$, we obtain that $M_{j+1}$ is a module of $G \setminus (M_j \setminus M_{j+1}) = G \setminus S_{j+1}$ (Property `(i)`). Property `(ii)` also holds because it holds for $M_j$ and $M_{j+1} \subset M_j$. Now, let $v \in M_{j+1}$ be arbitrary, and let us assume it is not an extremity of $G$. Let $C$ be any component of $G \setminus N[v]$ not containing vertex $u$. By Property `(iii)`, $C \subseteq M_j$. Suppose by contradiction $C \not\subseteq M_{j+1}$. Let $z \in C \setminus M_{j+1}$. Since we have $C \cap S_j = \emptyset$, we obtain $N(z) \cap S_j \subseteq N(v) \cap S_j$. By minimality of $|N(v) \cap S_j| = |N(M_{j+1}) \cap S_j|$, we get $N(z) \cap S_j = N(v) \cap S_j$, which contradicts that $z \notin M_{j+1}$.

The above Property (iii) implies that, if $|M_j| = 1$, then the unique vertex $v \in M_j$ is indeed an extremity. Furthermore, by Property (ii), $v$ is a vertex of $V \setminus S$ that maximizes $d(u, v)$. Hence, in order to prove correctness of the algorithm, all that remains to prove is that this algorithm eventually halts. For that, we claim that if $|M_j| > 1$ then $|M_{j+1}| < |M_j|$. Indeed, Property (i) asserts that $M_j$ is a module of $G \setminus S_j$. Let $A_j^1, A_j^2, \ldots, A_j^{p_j}$ be the partition of $M_j$ such that two vertices are in the same group if and only if they have the same neighbours in $S_j$. Since $G$ is prime, $M_j$ cannot be a nontrivial module of $G$, and therefore $p_j \geq 2$. Hence, $|M_{j+1}| = |A_j^1| < |M_j|$, as claimed. This above claim implies that eventually we reach the case when $|M_j| = 1$, and so, the algorithm eventually halts.

**Complexity.** Computing the LexBFS ordering $\sigma$ can be done in linear time [70]. Then once we computed vertex $w$ in additional $\mathcal{O}(n)$ time, we can compute the largest index $i$ such that $\lambda(w, \sigma(i)) = \lambda(\sigma(i), \sigma(i))$ as follows. We mark all the neighbours of vertex $w$, then we scan the vertices by decreasing LexBFS number, and we stop at the first encountered vertex $x \neq u$ such that all vertices in $N_{\succ}(x)$ are marked. Since all the neighbour-sets need to be scanned at most once, the total runtime for this step is linear. Finally, we dynamically maintain some partition such that, at the beginning of any step $j \geq 0$, this partition equals $(M_j)$. If $|M_j| > 1$, then we consider each vertex $s \in S_j$ sequentially, and we replace every group $X$ in the partition by the nonempty groups amongst $X \setminus N(s), X \cap N(s)$. In doing so, we obtain the partition $(A_j^1, A_j^2, \ldots, A_j^{p_j})$. We remove all groups but $M_{j+1}$, computing $S_{j+1} = M_j \setminus M_{j+1}$ along the way. By using standard partition refinement techniques [55, 67], after an initial processing in $\mathcal{O}(|M_0|)$ time each step $j$ can be done in $\mathcal{O}\left(\sum_{s \in S_j} |N(s)| + |M_j \setminus M_{j+1}|\right)$ time. Because all the sets $S_j$ are pairwise disjoint the total runtime is linear. ◀

## D Proof of Theorem 32

**Proof of Theorem 32.** By Lemma 29, we may assume $G$ to be prime. We proceed as follows:

- We compute a vertex $c$ of eccentricity at most $rad(G) + 15k - 5$.
- Then, we set $H = \{c\}$, $X = \emptyset$. While $H$ is not a dominating set of $G$, we compute an extremity $x_i \notin N[H]$, we add in $X$ *all* vertices of $N[x_i]$ and, for every $y \in N[x_i]$, we add to $H$ some arbitrary shortest $yc$-path $P_y$.
- Let $\mathcal{S} = \{N[P_x] \mid x \in X\} \cup \{N[v] \mid v \in V\}$. We apply a greedy set cover algorithm in order to extract from $\mathcal{S}$ a sub-family $\mathcal{S}'$ so that $\bigcup \mathcal{S}' = V$. Specifically, we set $\mathcal{S}' := \emptyset$, $U := V$, where $U$ represents the uncovered vertices. While $U \neq \emptyset$, we add to $\mathcal{S}'$ any subset $S \in \mathcal{S}$ such that $|S \cap U|$ is maximized, and then we set $U := U \setminus S$.
- Let $A = \{x \in X \mid N[P_x] \in \mathcal{S}'\}$ and let $B = \{v \in V \setminus A \mid N[v] \in \mathcal{S}'\}$.
  - For every $x \in A$, let $W_x$ contain the $\min\{d(x, c) + 1, 42k - 11\}$ closest vertices to $x$ in $P_x$. For each $w \in W_x$, we compute $\ell(w) = \max\{e(w') \mid w' \in N[w]\}$.
  - For every $v \in B$, we directly compute the eccentricities of *all* vertices in $N[v]$.
- We output the maximum eccentricity computed as the diameter value.

**Correctness.** In order to prove correctness of this above algorithm, let $u \in V$ be such that $e(u) = diam(G)$. By construction, $\mathcal{S}'$ covers $V$, and therefore, either there exists a $x \in A$ such that $u \in N[P_x]$, or there exists a $v \in B$ such that $u \in N[v]$. In the latter case, we computed the eccentricities of all vertices in $N[v]$, including $e(u) = diam(G)$. Therefore, we only need to consider the former case. Specifically, let $u^* \in V(P_x) \cap N[u]$. To prove correctness of the algorithm, it suffices to prove that $u^* \in W_x$. The proof that it is indeed the case is identifical to that of Theorem 26. Indeed, since $G$ has a dominating target of cardinality at most $k$, it contains an additive tree $(3k-1)$-spanner [59] and so – the same as graphs in $Ext_k$ –, it is $(3k-1)$-hyperbolic [20, 35].

**Complexity.** By Lemma 6, we can compute a vertex $c$ of eccentricity at most $rad(G)+15k-5$ in linear time. Then, during the second phase of the algorithm, we claim that each step can be done in $\mathcal{O}(\Delta m)$ time. Indeed, while $H$ is not a dominating set of $G$, a new extremity $x_i \notin N[H]$ can be computed by applying Lemma 23. Furthermore, we can compute the shortest paths $P_y$, for $y \in N[x_i]$, by executing at most $\Delta + 1$ BFS. Overall, the runtime of this second phase is in $\mathcal{O}(t\Delta m)$, with $t$ the number of extremities computed. We claim that $t \leq k$. Indeed, let $D$ be a fixed (but unknown) dominating target of cardinality at most $k$. Let $x_1, x_2, \ldots, x_t$ denote all the extremities computed during this second phase. By Lemma 11, each extremity $x_i$ computed is in $N[D]$. In order to prove that there are at most $k$ extremities computed, it suffices to prove that $N[x_i] \cap N[x_j] \cap D = \emptyset$ for every $i < j$. Suppose by contradiction that there exists a vertex $v \in D$ such that $v \in N[x_i] \cap N[x_j]$. At step $i$, we add $v$ in $X$, and therefore from this point on $x_j \in N[H]$. It implies that we cannot select $x_j$ at step $j$, a contradiction. Hence, the total runtime of this second phase is in $\mathcal{O}(k\Delta m)$.

In order to bound the runtime of the third phase of the algorithm, we first need to bound the minimum number of subsets of $\mathcal{S}$ needed in order to cover $V$. We claim that it is no more than $2k$. Indeed, as before let $D$ be a fixed (but unknown) dominating target of cardinality at most $k$. For each $x \in D \cap X$, we select the set $P_x$. For each $v \in D \setminus X$, we select the sets $N[v]$ and $P_{x'}$, for some arbitrary $x' \in X$ such that $v \in N(P_{x'})$. The claim follows since we assume $D$ to be a dominating target. It implies that the greedily computed sub-family $\mathcal{S}'$ has its cardinality in $\mathcal{O}(k \log n)$. The cumulative size of all subsets in $\mathcal{S}$ is at most $|X|n + 2m \leq k(\Delta+1)n + 2m$. Furthermore, the greedy set cover algorithm runs in $|\mathcal{S}'| = \mathcal{O}(k \log n)$ steps. As a result, the total runtime for this third phase is in $\mathcal{O}(k \log n \cdot (k\Delta n + m)) = \mathcal{O}(k^2 \Delta m \log n)$.

Lastly, during the fourth and final phase, we need to apply Lemma 25 $\sum_{x \in A} |W_x| = \mathcal{O}(k|A|)$ times, and we need to execute $\sum_{v \in B} |N[v]| = \mathcal{O}(\Delta|B|)$ BFS. Each call to Lemma 25 takes $\mathcal{O}(qm)$ time, with $q$ the number of extremities. By Lemma 11, $q = \mathcal{O}(k\Delta)$. Furthermore, we recall that $|A| + |B| \leq |\mathcal{S}'| = \mathcal{O}(k \log n)$. Therefore, the total runtime for this phase – and also for the whole algorithm – is in $\mathcal{O}(k^3 \Delta m \log n)$. ◀