



HAL
open science

Certificates in P and Subquadratic-Time Computation of Radius, Diameter, and all Eccentricities in Graphs

Feodor F. Dragan, Guillaume Ducoffe, Michel Habib, Laurent Viennot

► **To cite this version:**

Feodor F. Dragan, Guillaume Ducoffe, Michel Habib, Laurent Viennot. Certificates in P and Subquadratic-Time Computation of Radius, Diameter, and all Eccentricities in Graphs. [Research Report] Inria Paris; Université paris diderot; Kent State University; University of Bucharest, Faculty of Mathematics and Computer Science; National Institute for Research and Development in Informatics, Romania; DI ENS. 2024. hal-01729748v3

HAL Id: hal-01729748

<https://inria.hal.science/hal-01729748v3>

Submitted on 17 Jan 2025 (v3), last revised 23 Jan 2025 (v4)

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

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



Distributed under a Creative Commons Attribution 4.0 International License

Certificates in P and Subquadratic-Time Computation of Radius, Diameter, and all Eccentricities in Graphs*

Feodor Dragan[†] Guillaume Ducoffe[‡] Michel Habib[§] Laurent Viennot[¶]

January 17, 2025

Abstract

In the context of fine-grained complexity, we investigate the notion of certificate enabling faster polynomial-time algorithms. We specifically target radius (minimum eccentricity), diameter (maximum eccentricity), and all-eccentricity computations for which quadratic-time lower bounds are known under plausible conjectures. In each case, we introduce a notion of certificate as a specific set of nodes from which appropriate bounds on all eccentricities can be derived in subquadratic time when this set has sublinear size. The existence of small certificates for radius, diameter and all eccentricities is a barrier against SETH-based lower bounds for these problems. We indeed prove that for graph classes with certificates of bounded size, there exist randomized subquadratic-time algorithms for computing the radius, the diameter, and all eccentricities respectively.

Moreover, these notions of certificates are tightly related to algorithms probing the graph through one-to-all distance queries and allow to explain the efficiency of practical radius and diameter algorithms from the literature. In particular, our formalization enables a novel primal-dual analysis of a classical approach for diameter computation. Based on our novel insights for these problems, we introduce several new algorithmic techniques related to eccentricity computation and propose algorithms for radius, diameter and all eccentricities with theoretical guarantees with respect to certain graph parameters. This is complemented by experimental results on various types of real-world graphs showing that these parameters appear to be low in practice. Finally, we obtain refined results in the case where the input graph is a power-law random graph, has low doubling dimension, has low hyperbolicity, is chordal, satisfies some Helly-type property, or has bounded asteroidal number.

Keywords: certificate, fine-grained complexity, diameter, radius, all eccentricities, algorithm.

*This work was supported by the French ANR projects ANR-22-CE48-0001 (TEMPOGRAL), ANR-24-CE48-4377 (GODASse) and ANR-23-PEIA-005 (REDEEM).

[†]Kent State University, Department of Computer Science, Kent, Ohio, USA, dragan@cs.kent.edu.

[‡]University of Bucharest, Faculty of Mathematics and Computer Science, and National Institute for Research and Development in Informatics, Romania, guillaume.ducoffe@ici.ro.

[§]IRIF, Université Paris Cité & CNRS, Paris, France, habib@irif.fr.

[¶]Inria, DI ENS, Paris, France, laurent.viennot@inria.fr.

Contents

1	Introduction	4
1.1	Our contributions	5
1.2	Related work	9
1.3	Structure of the paper	10
I	Certificates for radius, diameter, all eccentricities	10
2	Preliminaries and definitions	10
3	Lower and upper certificates for eccentricities	11
4	Characterization of the unique minimum tight upper certificate	13
5	Lower bound for radius computation	14
6	Extended diameter certificates	14
7	Lower bound for diameter computation	15
II	Graphs with small certificates	15
8	Computing the radius in graphs with small radius certificate	15
9	Computing the diameter in graphs with small diameter certificate	17
10	Approximating radius certificate	19
11	Approximating all-eccentricity certificate	20
III	Practical algorithms	23
12	Radius computation and certification	23
13	Minimum eccentricity selection	24
14	Diameter computation and certification	26
15	All eccentricities	28
16	Experiments	29
IV	Graph classes with specific certificates	31
17	Power law random graphs	31

18	Graphs with low doubling dimension	33
18.1	Exact diameter computation with few antipodes	33
18.2	Approximating radius and diameter	35
19	Negatively curved graphs	36
20	Chordal graphs	39
21	Graph analogues of hyperconvex metric spaces	42
22	Graphs of bounded asteroidal number	45
23	Conclusion	46

1 Introduction

We investigate the notion of certificate in P, or more precisely how the existence of some small specific certificate for a given problem can enable a polynomial-time algorithm with smaller exponent compared to the situation where no specific certificate is known. This question appears particularly interesting about problems in P where the best exponent of a polynomial-time algorithm is not completely settled. In particular, we target the problems of radius and diameter computation in graphs, as there is a large gap between known quadratic-time lower-bounds [46, 1] based on the Strong Exponential Time Hypothesis (SETH) or other plausible conjectures, and the efficiency of exact practical algorithms that appear to succeed in computing the radius [9] and diameter [48, 22, 4] of various types of real-world graphs with few Breadth First Search (BFS) traversals [6]. The efficiency and the correctness of these algorithms suggest that such certificates exist in real-world graphs.

The main approach of these algorithms dates back to [48] and consists in maintaining for each vertex v of the input graph G a lower bound $\underline{e}(v)$ and/or an upper bound $\bar{e}(v)$ of its eccentricity $e(v) = \max_{w \in V(G)} d_G(v, w)$ which is the maximum distance of any node from v . Recall that the radius is the minimum eccentricity, while the diameter is the maximum eccentricity. Each time a BFS traversal is performed from a node x , the eccentricity $e(x)$ is obtained, and for each vertex v , its lower bound can be updated to $\max\{\underline{e}(v), d_G(v, x)\}$ while its upper bound can be replaced by $\min\{\bar{e}(v), d_G(v, x) + e(x)\}$ by triangle inequality. These practical algorithms [4, 9, 22, 48, 49] perform BFS traversals from specifically chosen vertices x until some stopping condition is met. If some vertex x with eccentricity R was found and if all lower bounds are greater or equal to R , we can then conclude that the input graph has radius R . Similarly, if some vertex with eccentricity D was encountered while at some point all eccentricity upper-bounds appear to be D or less, we can stop and conclude that the diameter is D . A typical choice of appropriate sources x for BFS traversals is to alternate between a vertex with minimum lower bound and a vertex with maximum upper bound [49, 9]. In a seminal work on understanding the efficiency of practical algorithms on real-world graphs [10], an analysis of a randomized variant of this algorithm within power-law random graphs bounds the number of sources used to $o(n)$.

We see the set X of sources x from which such an algorithm performs BFS traversals before returning values R and D as a certificate that the input graph has radius R and diameter D . Indeed, if X is given, then we can perform a BFS traversal from each vertex $x \in X$, update lower bounds accordingly and check that the radius is at least R in $O(|X|m)$ time where $m = |E(G)|$ denotes the number of edges in the input graph G . If X also contains a vertex of eccentricity R , we can then certify that the radius of the input graph is indeed R . As an example, for odd k , we will see that a $k \times k$ square grid has a five-node certificate (its center and its four corners). Similarly, we can check that X contains a vertex of eccentricity D and that all upper bounds are lower or equal to D for certifying that the diameter is indeed D in $O(|X|m)$ time. In the worst case, the algorithm may use $X = V(G)$ and require quadratic time, but it appears that various type of real-world graph do have such a set X with few dozens of vertices. Coming back to a $k \times k$ square grid with odd k , it has a two-node certificate (its center and a corner). It should be noted that this notion of certificate is independent of any algorithm: it is a graph property to have small or large certificates.

Note that the existence of a certificate of size $o(n)$ enables a non-deterministic subquadratic-time algorithm for computing the diameter (by first guessing the certificate and then computing the appropriate bounds on all eccentricities). Following the results of [12], this implies that for any graph class with truly sublinear certificates, the existence of meaningful lower bounds for diameter computation based on SETH is unlikely. Furthermore, this opens the possibility of breaking the

quadratic barrier for (deterministic or probabilistic) diameter computation (or similarly radius computation) if one can efficiently find a small certificate when there exists one. We thus ask the following.

Main question: For a given class of graphs, does the existence of small specific certificates coincides with the existence of a truly subquadratic-time algorithm for computing either the radius, or the diameter, or even all eccentricities?

We believe that the same question could be investigated in other classical problems studied in fine-grained complexity such as for example the All-Pairs Shortest Path Problem (APSP) in the light of practical algorithms for fast shortest-path computation such as Contraction Hierarchies [36] or Hub Labeling [2].

The paper is presented in the context of unweighted undirected graphs but all the notions and algorithms extend to the weighted and/or directed cases as shown in a short experimental part. In the directed setting, we restrict ourselves to the strongly directed case where diameter is again well defined as the maximum (outward or inward) eccentricity while two notions of radius arise as minimum outward (resp. inward) eccentricity. A more general setting for diameter in weakly-connected directed graphs is explored in [4, 9]. We believe that our certificate notions could naturally be extended to this setting.

1.1 Our contributions

Our general idea of certificate consists in a set of nodes such that the distances from these nodes to all nodes (rather than all-to-all pairs) allow to deduce the value of either the radius, or the diameter, or even all eccentricities, with certainty. It is thus tightly related to *one-to-all distance based algorithms*, that is algorithms probing the graph through one-to-all distances queries where a query for a vertex x returns the vector of all distances from x . Several concepts of certificates for radius, diameter and all eccentricities arise from this general idea. We now detail them and our related results. When considering a graph $G = (V, E)$, we assume that it is connected, and we generally let $n = |V|$ and $m = |E|$ denote its number of vertices and edges respectively. We use the terms vertex and node interchangeably.

Radius. Given a graph $G = (V, E)$ with radius $\text{rad}(G) = r$, we define a *radius certificate* as a set L of nodes such that any node of G is at distance at least r from some node of L . Equivalently, it can be defined as a covering of the node set V with complementary of open balls of radius $\text{rad}(G)$ (excluding nodes at distance $\text{rad}(G)$). As an example, for odd k , a $k \times k$ square grid has radius k and its four corners form a radius certificate. (See also the set R in the example of Figure 1.) In the case of radius, we obtain the following equivalence as an answer to our main question (as a consequence of Theorems 5 and 6).

Theorem 1 *Given a class of graphs \mathcal{G} and a sublinear function $\ell(n)$, we have:*

- *The existence of a one-to-all distance based randomized (Monte Carlo) algorithm for computing radius within \mathcal{G} and running with $\ell(n)$ queries implies that every graph of \mathcal{G} with n vertices has a radius certificate of size $O(\ell(n))$.*
- *If every graph with n vertices in \mathcal{G} has a radius certificate of size $\ell(n)$ at most, then there exists a one-to-all distance based randomized (Monte Carlo) algorithm computing within \mathcal{G} the radius and a radius certificate of size $O(\ell(n) \log n)$ through $O(\ell(n) \log^3 n)$ queries (in $O(\ell(n)m \log^3 n)$ time) with high probability.*

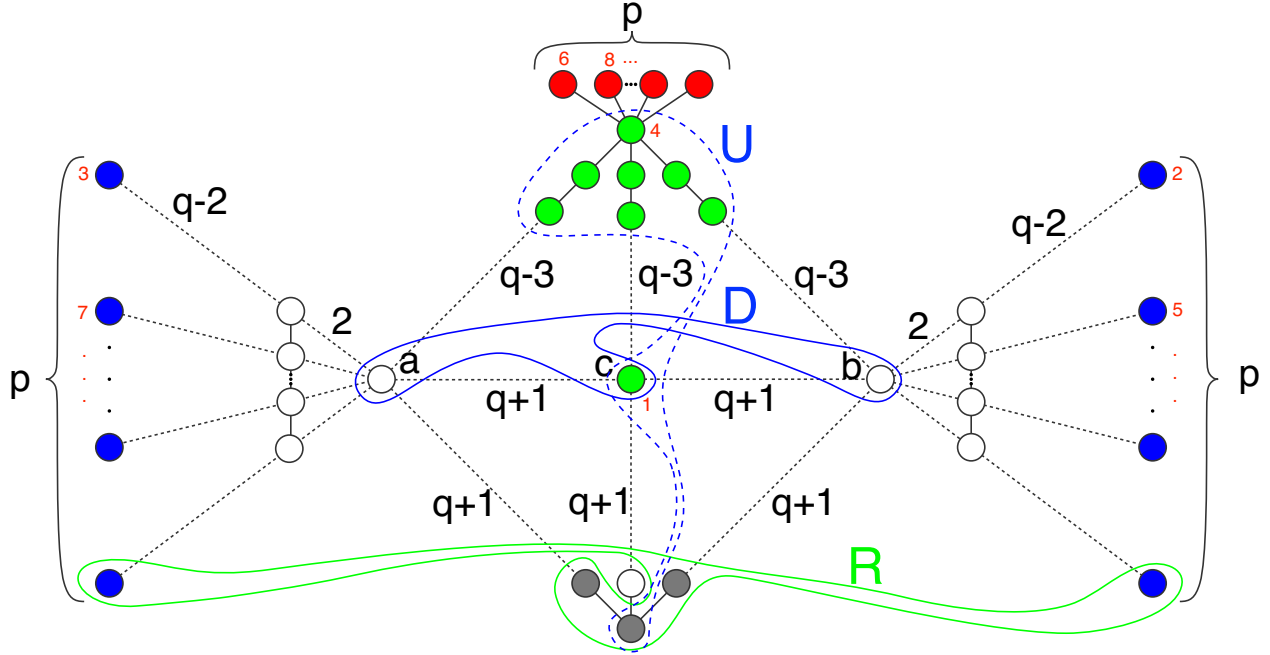


Figure 1: An example of bow-tie shaped graph $BT_{p,q}$ (for $p \geq 2$ and $q \geq 6$) with radius certificate R , diameter certificate D and all eccentricity certificate (R,U) . Its diameter is $4q - 2$ (eccentricity of blue nodes). Its radius is $2q + 1$ (eccentricity of green nodes). Plain lines correspond to edges while a dashed line with label ℓ corresponds to a path of length ℓ . All unlabeled dashed lines from a and b have length 2, while all unlabeled dashed lines from blue nodes have length $q - 2$ (similarly to given labels on upper lines).

The problem of finding a minimum radius certificate is indeed shown to be equivalent to minimum set cover. It is thus NP-hard while $O(\log n)$ -approximation (only) is doable in polynomial time. Compared to set cover, it has an additional difficulty: the sets are not directly available and computing all of them would require quadratic time at least. This setting for set cover has been considered in the literature [47], and we show how these prior results can be exploited in the design of fast randomized approximation algorithms for minimum-size radius certificate.

Diameter. Given a graph $G = (V, E)$ with diameter $\text{diam}(G) = D$, we define a *diameter certificate* as a set U of nodes such that any node v is at distance at most $D - e(x)$ from a node $x \in U$. In other words, it corresponds to a covering of V with balls $B[x, D - e(x)]$ of radius $D - e(x)$ where $e(x)$ is the eccentricity of the center x of the ball. As an example, for odd k , a $k \times k$ square grid has diameter $2k$ and its center forms a diameter certificate. (See also the set D in the example of Figure 1.) Again, the problem of finding a minimum diameter certificate is equivalent to minimum set cover. However, $O(\log n)$ -approximation in subquadratic time appears much more difficult as the radii of the balls considered depend from the (unknown) eccentricities of their centers. Nevertheless, it is still possible to show that the existence of truly sublinear diameter certificates enables a truly subquadratic-time algorithm for diameter computation (see Theorem 8).

Theorem 2 *For every class of graphs \mathcal{G} such that every graph with n vertices has a diameter certificate of size $\ell(n)$ at most, there exists a randomized (Monte Carlo) algorithm for computing the diameter within \mathcal{G} in $O(m\sqrt{\ell(n)n} \log^{3/2} n)$ time with high probability.*

All eccentricities Given a graph $G = (V, E)$, we define a *tight lower certificate* as a set L of nodes such that any node v is at distance $e(v)$ from some node x of L . We define a *tight upper certificate* as a set U of nodes such that any node v is at distance at most $e(v) - e(x)$ from some node $x \in U$. We call all-eccentricity certificate a couple (L, U) of such tight-lower and tight-upper certificates. In a $k \times k$ square grid with odd k , the four corners and the center form such a couple. (See also (R, U) in the example of Figure 1.) The problem of finding a minimum tight lower certificate again appears to be equivalent to set cover. However, the notion of tight upper certificate is related to a partial order structure that makes the minimum tight upper certificate unique and enables its computation in polynomial time although subquadratic-time computation still remains open. We can nevertheless obtain subquadratic-time computation of all eccentricities when sufficiently small certificates exist (see Theorem 10):

Theorem 3 *Given an arbitrary n -node graph G , all eccentricities, a tight lower certificate L of size $O(\ell^* \log n)$ and the minimum tight upper certificate U^{\preceq} can be computed in $O(m\ell^* |U^{\preceq}| \log^2 n)$ time with high probability where ℓ^* denotes the size of a minimum tight lower certificate.*

Practical algorithms. We propose a primal-dual analysis of a basic variant of the diameter algorithm of [48] which was not known before as far as we know. Indeed, a basic primal-dual argument implies that the maximum size π_1 of a packing for (closed) balls $B[u, \alpha(\text{diam}(G) - e(u))]$ for $\alpha = 1$ is a lower bound on the minimum size of a diameter certificate, as the latter corresponds to a covering with such balls. We prove that the nodes selected by this basic variant for performing BFS traversals form a packing P for balls with radii reduced by a factor $\alpha = 1/3$ (each such ball contains at most one node of P) implying that it indeed computes a $\pi_{1/3}/\pi_1$ approximation of the minimum diameter certificate.

We also design new algorithms for radius and all eccentricities. They both rely on a specific tight lower certificate, namely the set A of all antipodes, which are defined as follows. It is a subset of furthest nodes, i.e. the nodes that are at furthest distance from some node. More precisely, given a ranking of the nodes (e.g., their ID order), we define the antipode of a node u as the node at furthest distance from u having highest rank (the ranking is used for breaking ties among nodes at the same distance). Our all-eccentricity algorithm relies on a characterization of the (unique) minimum tight upper certificate of a graph which allows the algorithm to compute it. We obtain the following guarantees (as a consequence of Theorems 11, 12 and 13).

Theorem 4 *Given a graph G having ℓ antipodes overall (according to a given ranking), it is possible to compute:*

- *its radius, a center and a radius certificate of size ℓ at most in $O(\ell m)$ time,*
- *its diameter, a diametral node and a diameter certificate of size $\pi_{1/3}$ at most in $O(\pi_{1/3} m)$ time where $\pi_{1/3}$ is the maximum packing size for open balls $B(u, \frac{1}{3}(\text{diam}(G) - e(u)))$,*
- *all eccentricities, a lower certificate of size ℓ at most and the minimum upper certificate U^{\preceq} in $O((\ell + |U^{\preceq}|)m)$ time.*

We provide small experiments on various types of real-world graphs that confirm that these graphs have extremely small radius and diameter certificates (less than 30 nodes for all graphs when they have from tens of thousands to more than ten million nodes). Surprisingly they also have a very small tight lower certificate: the set of all antipodes. Although we do not know how to compute this set in subquadratic time, the fact that its size is very small in practical graphs

guarantees fast termination of our radius algorithm. Although the idea of using an antipode as a good candidate for a diametral node dates back to the TwoSweep heuristic [43], the idea of repeatedly using antipodes in radius and all-eccentricity computations is new. The observation of their sparsity in real-world graphs seems new also. Note that the set of furthest nodes (which contains antipodes) is also an obvious tight lower certificate, but it appears to be significantly larger in several networks. We also observe that these graphs have relatively small coverings with balls of reduced radii compared to the balls required for a diameter certificate, allowing to provide support for the efficiency of diameter algorithms based on the approach of [48]. The size of the minimum tight upper certificate appears to be quite variable in our experiments, from few percents to a large fraction of the nodes. Note that for graphs where it is larger than the number of antipodes (all our real-world graphs), our all-eccentricity algorithm is somehow optimal as its complexity is then the same as the algorithm checking the certificate.

Classical graph classes with specific certificates. We additionally refine our results in specific graph classes:

1. The analysis of [10] allows to show that, for any $\varepsilon > 0$, sufficiently large **power-law random graphs** have n^ε furthest nodes at most asymptotically almost surely, and their set is thus a tight lower certificate of size n^ε at most.
2. A refinement of our primal-dual algorithm for diameter leads to subquadratic $(1 + \varepsilon)$ -approximation of both radius and diameter in **graphs with constant doubling dimension**.
3. Our primal-dual radius algorithm runs in linear time in δ -**hyperbolic graphs** with exact computation when the degree is bounded and constant additive approximation in general.
4. The centers of any **chordal graph** form a diameter certificate, and all its eccentricities can be computed in linear time if it has bounded degree.
5. **Helly graphs** and **bipartite Helly graphs** have radius certificates of constant size and their radius can be computed in near-linear time.
6. Finally, the radius of **graphs with asteroidal number at most k** can be computed in $O(km^{3/2})$ time.

Our results on power-law random graphs, negatively curved graphs, and chordal graphs revisit some previous results to obtain bounds on certificates. We are not aware of any specific diameter algorithm for graphs with low doubling dimension prior to this work. Our framework allows to improve the state of the art of radius computation in both Helly graphs and graphs with bounded asteroidal number.

Overall, we believe that our notion of certificate allows to better understand when the quadratic barrier can be overcome for radius, diameter, and all eccentricity computations. Moreover, it could also be fruitful for investigating other known barriers in P. It also provides new insights on the efficiency of practical algorithms for radius and diameter, and enables more robust practical algorithms with complexity guarantees. We argue that it significantly enhances the state of the art for all-eccentricity computation. The new techniques proposed also enable new types of radius and diameter algorithms with parameterized complexity with respect to parameters related to the size of such certificates. We give one such example for graphs of bounded asteroidal number, for which we present the first subquadratic-time algorithm for computing the radius. Incidentally, the latter answers an open question from [34].

1.2 Related work

Our notion of certificate is related to the broader notion of certifying algorithm [44, 5] which is an algorithm that produces, in addition to its output, a certificate or witness, that is an easy-to-verify proof that the particular output is valid. Such an algorithm is said to be efficient if the complexity of computing the output and then verifying its validity using the certificate is similar to that of the best (non-certifying) algorithm. We are mainly interested in the case where verification can be done faster than the best algorithm. The practical algorithms we propose for diameter, radius and all eccentricities are indeed certifying algorithms. We are not aware of any non-trivial certifying algorithms for these prior to this work.

As mentioned earlier, our notion of certificate is also closely related to nondeterministic algorithms [12, 42], since guessing a certificate and verifying it turns out to be a nondeterministic algorithm for solving the problem with the same complexity as the verification algorithm. However, we express this latter complexity as a function of both the size of the input and the size of the certificate. More specifically, we are interested in relating the size of a smallest certificate for a given input to the difficulty of solving the problem for that particular input.

In the context of approximating diameter [11], a h -dominating set X is used for upper-bounding the diameter: given X , the minimum h such that X is h -dominating and the eccentricity of every node in X can be computed with $|X| + 1$ BFS traversals. The diameter can then be bounded by $h + \max_{x \in X} e(x)$. Our diameter certificate can be seen as a refinement of this approach where we bound independently the eccentricity of each node u with respect to the best bound $d(u, x) + e(x)$ for $x \in X$. In general, this provides a better diameter bound.

Our work follows a long line of research around practical radius and diameter computations that dates back to the computation of a center in a tree [39]. It consists in a two-sweeps approach where the last visited node in a first BFS traversal is used as the source of a second BFS traversal. It can also be seen as a heuristic [43] providing a diameter estimate that appears to be often tight in practice [3]. It thus introduces the idea of using what we call antipodes as tentative diametral nodes. The two-sweep heuristic was also shown to provide good approximation (up to a small constant) for chordal graphs, δ -hyperbolic graphs and various other graph classes [15, 16, 20]. Note that, if one insists in approximating the diameter of an arbitrary graph in near-linear time, it turns out [23] that the simple linear-time algorithm, that outputs a 2-approximation to the diameter by performing one BFS traversal from an arbitrary vertex, is optimal. A four-sweeps heuristic is proposed in [22] and complemented with an exact diameter algorithm called iFub. The four-sweep heuristic performs twice the two-sweep method, using a mid-point of the longest path found in the first round as the starting point of the second one. The idea is that mid-points of longest paths make good candidates for central nodes or at least nodes with small eccentricity. The iFub method additionally inspects furthest nodes from the best candidate center found with the four-sweep heuristic until exact value of the diameter can be inferred.

The concept of certificate is somehow implicit in the method introduced in [48, 49] that consists in maintaining lower and upper bounds on the eccentricity of each node. After each BFS traversal these bounds are improved based on distances from the source of the traversal. The sources used for the BFS traversals performed by the algorithm form what we call a certificate. Contrarily to this approach, we distinguish nodes used for improving lower bounds (the lower certificate) from those used for improving upper bounds (the upper certificate). This approach with bounds has been extended for diameter computation in weakly connected directed graphs in parallel works [9, 4]. The Exact SumSweep method [9] additionally computes the radius in addition to the diameter. It integrates many techniques proposed in previous practical algorithms plus a heuristic based on sum of distances for discovering nodes with large eccentricity in an initial phase.

An impressive analysis of these algorithms (several heuristics, IFub and Exact SumSweep) within power-law random graphs is performed in [10]. The analysis proposed for Exact SumSweep indeed implies the existence of certificates of size $n^{O(\varepsilon)}$ for both radius and diameter for any $\varepsilon > 0$ and sufficiently large power-law random graphs (see Section 17). We revisit their analysis to show a similar bound for a tight lower certificate.

Our primal-dual approach for practical algorithms has some similarities with the study of packings and coverings of hyperbolic graphs with balls proposed in [17], although slightly different problems are considered. It would be interesting to derive similar results in hyperbolic graphs for the collections of balls (or complementary of balls) we consider here.

1.3 Structure of the paper

Part I introduces basic graph and set-cover terminology (Section 2), the notions of certificate for radius, diameter, and all eccentricities (Section 3), and then provides a characterization of minimum tight upper certificates (Section 4) as well as the link between radius certificates and one-to-all distance-based algorithms (Section 5). Part II provides partial answers to our main question about subquadratic-time computations of radius (Section 8), diameter (Section 9), and all eccentricities (Section 11). Results in terms of a stricter notion of radius certificate are given in Section 10. Part III presents our practical algorithms for radius (Section 12), diameter (Section 14) and all eccentricities (Section 15). A new technique at the core of our radius and all-eccentricity algorithms is described in Section 13. Some experimental results detail for some real-world graphs various parameters involved in our analyses (Section 16). Part IV is dedicated to our results on power-law random graphs (Section 17), graphs with constant doubling dimension (Section 18), hyperbolic graphs (Section 19), chordal graphs (Section 20), Helly graphs (Section 21) and graphs with bounded asteroidal number (Section 22).

Part I

Certificates for radius, diameter, all eccentricities

2 Preliminaries and definitions

In the sequel, we suppose that we are given a finite undirected unweighted connected graph G . We denote by V its set of nodes and by E its set of edges. We let $n = |V|$ and $m = |E|$ denote its number of nodes and edges respectively. Let $d(u, v)$ denote the distance between two nodes u and v in G , that is the length of a shortest path from u to v . The *eccentricity* $e(u)$ of a node u is the maximum length of a shortest path from u , that is $e(u) = \max_{v \in V} d(u, v)$. The *furthest nodes* of u are the nodes v at furthest distance from u , i.e. satisfying $d(u, v) = e(u)$. We let $F(u) = \{v \in V : d(u, v) = e(u)\}$ denote their set. Given a ranking r of the nodes, the *antipode* $\text{Antipode}_r(u)$ of a node u for r is its furthest node with highest rank. Formally, $\text{Antipode}_r(u) = \text{argmax}_{v \in V} (d(u, v), r(v))$ where pairs are ordered lexicographically. A node is called a furthest node (resp. an antipode) if it is a furthest node (resp. an antipode) of some other node. Given a set $W \subseteq V$, we let $\text{Antipode}_r(W) = \{\text{Antipode}_r(u) : u \in W\}$ denote the set of antipodes from nodes in W . The *diameter* $\text{diam}(G) = \max_{u \in V} e(u)$ of G is the maximum eccentricity in G and the *radius* $\text{rad}(G) = \min_{u \in V} e(u)$ is the minimum eccentricity in G . A *diametral node* b is a node

with maximum eccentricity ($e(b) = \text{diam}(G)$). A *central node* c (or simply *center*) is a node with minimum eccentricity ($e(c) = \text{rad}(G)$). Denote by $C(G) = \{v \in V : e(v) = \text{rad}(G)\}$ the set of all central nodes of G . We let $B[u, r] = \{v \in V \mid d(u, v) \leq r\}$ (resp. $B(u, r) = \{v \in V \mid d(u, v) < r\}$) denote the (closed) ball (resp. open ball) with radius r centered at a node u . Similarly, we define its *coball* of radius r as $\overline{B}(u, r) = \{v \in V \mid d(u, v) \geq r\}$, that is the complementary of $B(u, r)$.

We restrict ourselves to algorithms based on one-to-all distance queries: we suppose that an algorithm `DistFrom` for one-to-all distances is given (typically BFS or Dijkstra in the weighted case). It takes a graph G and a node u as input and returns distances from u . More precisely, `DistFrom`(G, u) returns a vector D such that $D(v) = d(u, v)$ for all $v \in V$. In particular, $e(u)$ can be defined as the maximum value in the vector and the antipode of u as the index with highest rank where this value appears in D . We may measure the complexity of an algorithm by the number of one-to-all distance queries it performs when its cost mainly comes from these operations. A one-to-all distance-based algorithm accesses the graph only through one-to-all distance queries and solely relies on distances known from queries, triangle inequality, and non-negativeness of distances for bounding unknown distances.

Given a collection \mathcal{S} of subsets of V such that $\cup_{S \in \mathcal{S}} S = V$, a *covering* with \mathcal{S} is a sub-collection $\mathcal{C} \subseteq \mathcal{S}$ of sets such that their union covers all of V : $V \subseteq \cup_{S \in \mathcal{C}} S$. (A set $S \in \mathcal{S}$ is said to cover elements in S). Recall that the set-cover problem consists in finding a covering of minimum size. We define a *packing* for \mathcal{S} as a subset $P \subseteq V$ such that any set of \mathcal{S} contains at most one element in P . The denomination comes from the fact that elements of P correspond to pairwise disjoint subsets of the dual collection $\mathcal{S}^* = \{\{S \in \mathcal{S} \mid u \in S\} : u \in V\}$. A *hitting set* for \mathcal{S} is a set P that intersects all sets of \mathcal{S} . (Equivalently, a hitting set can be defined as a covering for \mathcal{S}^* but it may be more convenient to consider a collection rather than its dual.) We let $\pi(\mathcal{S})$ denote the maximum size of a packing for \mathcal{S} , and $\kappa(\mathcal{S})$ denote the minimum size of a covering for \mathcal{S} . As a covering must cover each element of a packing with distinct sets, we obviously have $\pi(\mathcal{S}) \leq \kappa(\mathcal{S})$ (weak duality). We say that a collection \mathcal{R} is *restricted* compared to \mathcal{S} if there exists a one-to-one mapping f from \mathcal{R} to \mathcal{S} such that $S \subseteq f(S)$ for all sets $S \in \mathcal{R}$. Note that this mapping then turns any covering with \mathcal{R} into a covering with \mathcal{S} and we thus have $\kappa(\mathcal{S}) \leq \kappa(\mathcal{R})$. Similarly, a packing for \mathcal{S} is also a packing for \mathcal{R} and we have $\pi(\mathcal{S}) \leq \pi(\mathcal{R})$. In other words, restricting the sets of a collection to smaller subsets increases maximum packing size and minimum covering size.

3 Lower and upper certificates for eccentricities

Our notion of certificate is based on the fact that knowing all distances from a given node x provides some bounds on the eccentricities of other nodes:

$$\forall u \in V, d(u, x) \leq e(u) \leq d(u, x) + e(x). \quad (1)$$

The first inequality derives directly from the eccentricity definition while the second one is a consequence of the triangle inequality. A possibly tighter lower-bound of $\max\{d(u, x), e(u) - d(u, x)\}$ could be used as in [48] but this optimization would complicate our definitions without reducing significantly certificate size (see Theorem 5 in Section 5).

We say that a set L (resp. U) of nodes is a *lower certificate* (resp. an *upper certificate*) of G when it is used to obtain lower bounds (resp. upper bounds) of eccentricities in G . Given the distances from a node u to all nodes in $L \cup U$ and the eccentricities of nodes in U , we have the following lower and upper bounds for the eccentricity of any node u (as a direct consequence of

Inequation 1):

$$e_L(u) \leq e(u) \leq e^U(u), \quad \text{where } \begin{cases} e^U(u) = \min_{x \in U} d(u, x) + e(x) \\ e_L(u) = \max_{x \in L} d(u, x) \end{cases}$$

Note that the bounds would work also by taking the min and max over all nodes $x \in U \cup L$. However, we prefer to distinguish nodes that provide lower bounds from those that provide upper bounds. If needed, a node could be both in L and U .

A lower (resp. upper) certificate L (resp. U) is said to be *tight* when $e_L(u) = e(u)$ (resp. $e^U(u) = e(u)$) for all $u \in V$. An *all-eccentricity certificate* is defined as a couple (L, U) of a tight lower certificate L and a tight upper certificate U .

Given a bound D and a node x , we have $d(u, x) + e(x) \leq D$ if and only if $u \in B[x, D - e(x)]$. Given an upper certificate U we thus have $e^U(u) \leq D$ if and only if $u \in \cup_{x \in U} B[x, D - e(x)]$. A *diameter certificate* is a set U such that $e^U(u) \leq \text{diam}(G)$ for all $u \in V$. Equivalently it can be defined as a covering with $\{B[x, \text{diam}(G) - e(x)] : x \in U\}$ using balls whose radius equals $\text{diam}(G)$ minus eccentricity of the center (and identifying a ball with its center).

Similarly, given a lower certificate L and a bound R , we obviously have $e_L(u) \geq R$ for all nodes u whose coball $\overline{B}(u, R)$ intersects L (i.e., there exists a node in L at distance R at least from u). We thus define a *radius certificate* as a set L such that $e_L(u) \geq \text{rad}(G)$ for all $u \in V$, or equivalently as a hitting set L for the collection $\{\overline{B}(u, \text{rad}(G)) : u \in V\}$ of coballs of radius $\text{rad}(G)$. As $x \in \overline{B}(u, \text{rad}(G))$ if and only if $u \in \overline{B}(x, \text{rad}(G))$, the collection of coballs of radius $\text{rad}(G)$ is its own dual, and a radius certificate L can equivalently be defined as a covering for this collection.

Note that a tight lower certificate can equivalently be defined as a hitting set for the collection $\{\overline{B}(u, e(u)) : u \in V\}$. Similarly, a tight upper certificate can equivalently be defined as a covering for the collection $\{u \in V \mid d(u, x) \leq e(u) - e(x) : x \in V\}$.

Examples. A path with $2k + 1$ nodes has a radius certificate with two nodes (the two extremities) and a diameter certificate with one node (its mid-point). More generally, any graph G such that $\text{diam}(G) = 2 \text{rad}(G)$ has a one-node diameter certificate (a center). It can be shown that any tree has a radius certificate of two nodes (two well chosen leaves) while its centers (at most two nodes) form a diameter certificate. A square grid has a radius certificate with four nodes (the corners) while its centers (at most four nodes) form a diameter certificate. As an extreme example of graph with large certificates, consider a cycle C . Its only radius and diameter certificates are both the whole set of its nodes. More generally, the whole set of nodes is the only diameter certificate of any graph where all nodes have same eccentricity (when radius equals diameter).

Hardness of approximation. Similarly to [17], we note that set cover can easily be encoded with ball cover: given a collection \mathcal{S} of subsets of V of an instance of the set-cover problem, consider the split graph where the sets S of \mathcal{S} form a clique and the elements $x \in V$ form a stable set so that the nodes x and S are adjacent if and only if $x \in S$. Without loss of generality, we may assume that no subset in \mathcal{S} equals V (otherwise, the problem is trivial), no set is empty (otherwise, we can remove it) and that there exist two elements such that no set contains both of them (if needed, we add a singleton $\{z\}$ to \mathcal{S} where z is a new dummy element added to V). In this graph, sets and elements have eccentricity 2 and 3 respectively. Any minimum diameter certificate is a covering with balls of radius 1 or 0 (if centered on a set or an element). One can easily transform it into a covering with balls of radius 1 centered at nodes corresponding to sets (only). It then corresponds to an optimal solution of the original set-cover problem. Now consider the complementary graph which is also a split graph where elements form a clique while sets form

a stable set and where x and S are adjacent if and only if $x \notin S$. Similarly, a minimum radius certificate for this complementary graph corresponds to a covering with coballs of radius 2 centered at sets and is also an optimal solution to the original set-cover problem. The hardness of set-cover approximation [25] thus implies that computing a minimum diameter (resp. radius) certificate is NP-hard and no polynomial-time algorithm can approximate it with a factor $(1 - o(1)) \log n$ unless $P = NP$. Assuming that some set in \mathcal{S} does not intersect any other set (it suffices again to add a singleton $\{z\}$ to \mathcal{S} where z is a new dummy element added to V), a tight lower certificate can easily be obtained from a radius certificate and we similarly obtain that computing a tight lower certificate is NP-hard and cannot be approximated within a factor $(1 - o(1)) \log n$ unless $P = NP$. Surprisingly, the situation is different for the problem of finding a minimum tight upper certificate, which can be solved in polynomial time as shown now.

4 Characterization of the unique minimum tight upper certificate

Recall that a set U is a tight upper certificate if for any vertex u , there is a node $x \in U$ such that $d(u, x) + e(x) \leq e(u)$, or equivalently $d(u, x) + e(x) = e(u)$ as $d(u, x) + e(x) \geq e(u)$ is always satisfied by triangle inequality. We call such a node x a *tight upper vertex-certificate* for u as we have $e^{\{x\}}(u) = e(u)$. This notion yields to the following characterization.

Proposition 1 *Given a graph G , being a tight upper vertex-certificate defines a binary relation \preceq which is a partial order ($u \preceq x$ stands for $e(u) = d(u, x) + e(x)$). Moreover, the set U^\preceq of all maximum elements of this partial order is the unique tight upper certificate of G with minimum size.*

Proof. We first prove that the relation \preceq is a partial order. It is obviously reflexive as the distance from a node to itself is zero, implying $e(u) = d(u, u) + e(u)$. It is antisymmetric: if x and y are both tight upper vertex-certificates one for the other, we then have $e(x) = d(x, y) + e(y)$ and $e(y) = d(y, x) + e(x)$, and thus $d(x, y) = 0$. We finally show transitivity. Suppose that y is a tight upper vertex-certificate for x and that z is a tight upper vertex-certificate for y , *i.e.* $e(x) = d(x, y) + e(y)$ and $e(y) = d(y, z) + e(z)$. We thus have $e(x) = d(x, y) + d(y, z) + e(z)$. As triangle inequality implies $e(x) \leq d(x, z) + e(z)$, we obtain $d(x, y) + d(y, z) \leq d(x, z)$, and thus $d(x, y) + d(y, z) = d(x, z)$ by triangle inequality again. We finally get $e(x) = d(x, z) + e(z)$ and z is a tight upper vertex-certificate for x .

We now show that the set U^\preceq of maximal elements for \preceq is the unique optimal tight upper certificate of G . For any non-maximal element u , we can build a chain $u \preceq x_1 \preceq x_2 \preceq \dots$ where u has a tight upper vertex-certificate x_1 , if x_1 is not in U^\preceq , it has a tight upper vertex-certificate x_2 , and so on. As the partial order is finite, the chain must be finite and x_k must be in U^\preceq for some k . The transitivity of \preceq implies that x_k is a tight upper vertex-certificate for u implying $e^{U^\preceq}(u) \leq d(u, x_k) + e(x_k) = e(u)$. This shows that U^\preceq is a tight upper certificate. As each element x of U^\preceq is the only tight upper vertex-certificate for itself (as a maximal element), U^\preceq is included in any tight upper certificate of G . In particular, any minimum tight upper certificate must indeed equal U^\preceq . ■

Note that U^\preceq includes in particular all centers of the graph: a center c cannot have a tight upper vertex-certificate $x \neq c$ (otherwise we have $e(x) = e(c) - d(c, x) < e(c)$ in contradiction with the minimality of $e(c)$).

The tight upper vertex-certificate relation and then U^\preceq can easily be computed in $O(nm)$ time by computing all distances. A more practical algorithm is proposed in Section 15.

5 Lower bound for radius computation

We now show that the notion of radius certificate is related to the minimum number of queries a one-to-all distance-based algorithm must perform.

Theorem 5 *Given a graph G , if an execution of a one-to-all distance-based randomized Monte Carlo algorithm successfully computes its radius by querying a set L of nodes, then $L \cup \text{Antipode}_r(L)$ is a radius certificate for any ranking r . Such a radius algorithm thus requires at least $\frac{1}{2}|L_{OPT}|$ one-to-all distance queries in a successful execution where $|L_{OPT}| = \kappa(\{\overline{B}(u, \text{rad}(G)) : u \in V\})$ is the minimum size of a radius certificate.*

Proof. Consider any Monte Carlo algorithm for computing the radius with positive probability. There must exist an execution with input G that succeeds. Let L denote the set of nodes queried for one-to-all distances in such an execution. A proof of correctness of the algorithm allows to conclude that all nodes have eccentricity $\text{rad}(G)$ at least based on triangle inequality and the distances known to the algorithm. That is for each node u , there is a node v such that we can prove $d(u, v) \geq \text{rad}(G)$ based on triangle inequality and distances from nodes in L . Consider a node v such that the proof uses a minimum number of triangle inequalities. If neither u nor v is in L , the proof must use a triangle inequality $d(u, v) \geq |d(u, x_1) - d(v, x_1)|$ for some node x_1 and a proof of $|d(u, x_1) - d(v, x_1)| \geq \text{rad}(G)$. In the case $|d(u, x_1) - d(v, x_1)| = d(u, x_1) - d(v, x_1)$, we would have a shorter proof $d(u, x_1) \geq \text{rad}(G)$ in contradiction with the choice of v . We thus continue with the case $|d(u, x_1) - d(v, x_1)| = d(v, x_1) - d(u, x_1)$. We then have a proof of $d(v, x_1) \geq \text{rad}(G) + d(u, x_1)$. Either $x_1 \in L$ or the proof uses a node x_2 such that $|d(v, x_2) - d(x_2, x_1)| \geq \text{rad}(G) + d(u, x_1)$. The choice of v again implies $|d(v, x_2) - d(x_2, x_1)| = d(v, x_2) - d(x_2, x_1)$ (otherwise x_2 would provide a shorter proof). By repeating this argument, we deduce that a shortest proof of $d(u, v) \geq \text{rad}(G)$ uses a sequence x_1, \dots, x_p of nodes such that $d(v, x_i) \geq \text{rad}(G) + d(u, x_1) + d(x_1, x_2) + \dots + d(x_{i-1}, x_i)$ for $i = 1, \dots, p$ and $x_p \in L$. Consider the antipode $a = \text{Antipode}_r(x_p)$. We then have $d(a, x_p) \geq d(v, x_p) \geq \text{rad}(G) + d(u, x_1) + \dots + d(x_{p-1}, x_p)$. By triangle inequality, we have $d(u, a) \geq d(a, x_p) - d(u, x_p)$ and $d(u, x_p) \leq d(u, x_1) + \dots + d(x_{p-1}, x_p)$. We thus have $d(u, a) \geq \text{rad}(G)$. In all cases, $L \cup \text{Antipode}_r(L)$ must contain a node at distance $\text{rad}(G)$ or more from u , it is thus a radius certificate. \blacksquare

Note that a similar result does not hold for diameter certificate. A one-to-all distance based algorithm for diameter could query a set U of nodes such that for any pair u, v there is $x \in U$ satisfying $d(u, x) + d(x, v) \leq \text{diam}(G)$ which implies $d(u, v) \leq \text{diam}(G)$ by triangle inequality. Note that checking that a set U has this property requires quadratic time in general (under SETH) even if U has size $O(\log n)$ (see the reduction from SAT to diameter computation in [46]). Our diameter certificate definition requires that for each node u a single node x allows to bound all distances $d(u, v)$ for $v \in V$ using $d(u, v) \leq d(u, x) + e(x)$. The reason for this stronger requirement is to enable subquadratic-time verification that a certificate is indeed a certificate when it has $o(n)$ size.

6 Extended diameter certificates

An *extended diameter certificate* is defined as a set X of nodes satisfying:

$$\forall u \in V, \forall v \in V, \exists x \in X, d(u, x) + d(x, v) \leq \text{diam}(G).$$

In contrast, X is a diameter certificate if and only if $\forall u \in V, \exists x, \forall v \in V \in X, d(u, x) + d(x, v) \leq \text{diam}(G)$. Note that a diameter certificate is indeed an extended diameter certificate but the converse might not be true.

Proposition 2 *Given a graph $G = (V, E)$ and an extended diameter certificate X of G , we can compute its diameter in $O(|X|m + n \log^{|X|} n)$ time.*

Proof. Perform a BFS traversal from each vertex $x \in X$ to obtain all distances $d(x, v)$ for $(x, v) \in X \times V$ in $O(|X|m)$ time. Let x_1, \dots, x_k denote the nodes of X in an arbitrary order. We can then associate a point $p_v = (d(x_1, v), \dots, d(x_k, v))$ in \mathbb{N}^k to each vertex $v \in V$. Compute a data-structure enabling efficient range queries on these points. Given a value D , we can test if D is smaller or equal to the diameter of G as follows. For each node u , we then test if the range between points $(D - d(x_1, u), \dots, D - d(x_k, u))$ and (∞, \dots, ∞) is empty. If it is not, there is a node v such that $d(x, v) \geq D - d(x, u)$, or equivalently $d(u, x) + d(x, v) \geq D$, for all $x \in X$. However, since X is an extended diameter certificate, there exists $x \in X$ such that $d(u, x) + d(x, v) \leq \text{diam}(G)$, which thus implies $D \leq \text{diam}(G)$. Conversely, if $D \leq \text{diam}(G)$, then we can show that at least one such range is empty. Indeed, consider a diametral pair a, b (i.e. satisfying $d(a, b) = \text{diam}(G)$). By triangle inequality, we have $d(a, x) + d(x, b) \geq d(a, b) = \text{diam}(G) \geq D$ for all $x \in X$. This implies $d(x, b) \geq D - d(x, a)$ for all $x \in X$, and p_b is thus in the range between $(D - d(x_1, a), \dots, D - d(x_k, a))$ and (∞, \dots, ∞) . We can thus find $\text{diam}(G)$ by dichotomy using $O(n \log n)$ range queries. ■

7 Lower bound for diameter computation

Part II

Graphs with small certificates

8 Computing the radius in graphs with small radius certificate

The algorithmic results presented in Sections 12 and 14 are based on the existence of certificates for radius and diameter of size the number of antipodes at most. However, the size of these certificates may be far from optimum. In this section, we further discuss the consequences of having small radius or diameter certificates. As a starter, let us observe that for any graph $G = (V, E)$, for every positive integer r , there exists a *non-deterministic* linear-time algorithm that decides whether $\text{rad}(G) \leq r$: specifically, we choose non-deterministically a vertex c of the graph, then we perform a BFS traversal with starting vertex c in order to compute $e(c)$, and finally, we accept if and only if $e(c) \leq r$. If $\text{rad}(G) \leq r$, then there always exists a choice of vertex c such that the algorithm accepts (*e.g.*, for a central vertex c); conversely, if $\text{rad}(G) > r$, then the algorithm must reject for any choice of vertex c . Conversely, if G has a radius certificate with $\ell(n)$ vertices at most, then we can non-deterministically decide in $O(\ell(n)m)$ time whether $\text{rad}(G) > r$: specifically, we select non-deterministically a subset L of $\ell(n)$ vertices at most, then we perform a BFS traversal for every vertex of L , and finally we accept if and only if $e_L(v) > r$ for every $v \in V$. If $\text{rad}(G) > r$, then as before there always exists a choice of subset L such that the algorithm accepts (*e.g.*, if L is a radius certificate of minimum size); conversely, if $\text{rad}(G) \leq r$, then for any subset L , for any central vertex v , we get $e_L(v) \leq r$, and so the algorithm always rejects. Note that similar results can be derived for diameter computation on graphs with a diameter certificate of size $\ell(n)$ at most. Now, according to [12], if a problem can be solved both non-deterministically and nondeterministically in $O(T(N))$ time, where N denotes the input size, then a conditional lower

bound in $\Omega(T(N)^{1+\gamma})$, for any $\gamma > 0$, cannot be proved based on the strong exponential-time hypothesis, unless some nondeterministic version of the latter can be falsified. As a result, for any class of graphs \mathcal{G} , *the existence of sublinear certificates for radius (for diameter, resp.) is a barrier against quadratic-time lower bounds for radius computation (for diameter computation, resp.) under the strong exponential time hypothesis.* This result holds even if there is no known efficient algorithm for computing such certificates. However, we note that it does not immediately imply the existence of a truly subquadratic-time algorithm for computing the radius (the diameter, resp.) when restricted to graphs in \mathcal{G} .

Our main result in this section is as follows:

Theorem 6 *For every class of graphs \mathcal{G} such that every graph with n vertices has a radius certificate of size $\ell(n)$ at most, there exists a randomized (Monte Carlo) algorithm for computing the radius (and a radius certificate) within \mathcal{G} in $O(\ell(n)m \log^3 n)$ time with high probability.*

Our approach for proving Theorem 6 is based on the existence of an efficient approximation algorithm for computing a hitting set in any family of coballs in a graph, where coballs $\overline{B}(v, r)$ are implicitly represented by the vertex-value pairs (v, r) . This approach is indeed developed in [47] where a set-cover of a universe U with N elements is computed even though the collection \mathcal{C} of M sets is not explicitly given. Instead their algorithm processes through queries on elements and sets: a *containment query* on $u \in U$ consists in listing the identifiers of all sets of \mathcal{C} that contain it; a *subset query* on $S \in \mathcal{C}$ consists in listing the elements in S .

Theorem 7 ([47]) *A set-cover of size at most $O(OPT \cdot \log(N + M))$ can be computed with high probability using at most $O(OPT \cdot \log^2(N + M))$ containment and subset queries through a Monte Carlo algorithm.*

We believe that their technique can be turned into a Las Vegas algorithm with few efforts, but leave it for future work. We nevertheless make a step in that direction in Section 10. Our proof, in what follows, is based on the observation that queries for coballs can be simulated using BFS traversals, namely:

Proof.[of Theorem 6] It suffices to present an $O(\ell(n)m \log^2 n)$ -time algorithm that for any graph $G = (V, E)$ that is in \mathcal{G} , for any positive integer r , decides whether $\text{rad}(G) \geq r$. Indeed, we can compute $\text{rad}(G)$ with $O(\log n)$ calls to this algorithm, using binary search.

We observe that $\text{rad}(G) \geq r$ if and only if there exists a hitting set of size $\ell(n)$ at most for the family of coballs $\{\overline{B}(v, r) : v \in V\}$. In one direction, if $\text{rad}(G) \geq r$ then any radius certificate can be selected as the desired hitting set. In particular, there is one with $\ell(n)$ vertices at most. In the other direction, if $\text{rad}(G) < r$ then $\overline{B}(c, r) = \emptyset$ for any central vertex c , which implies the nonexistence of a hitting set. As a result, we are left presenting an algorithm that either computes a hitting set of size $O(\ell(n) \log n)$ for $\{\overline{B}(v, r) : v \in V\}$, or correctly asserts that no such hitting set with $\ell(n)$ vertices at most can exist. We next explain how such an algorithm can be derived from Theorem 7.

More specifically, we set $U = V$, $\mathcal{C} = \{\overline{B}(v, r) : v \in V\}$. Note that $N = M = n$. We apply Theorem 7, but we add the restriction that at most $O(\ell(n) \log^2 n)$ queries can be performed (else, we abort). By doing so, we either compute a set-cover $\mathcal{S} \subseteq \mathcal{C}$ of size $O(\ell(n) \log n)$ at most, or we correctly assert that no such set-cover of size $\ell(n)$ at most can exist. A containment query for $u \in V$ can be simulated using a BFS traversal with starting vertex u . Similarly, a subset query for $\overline{B}(v, r)$ can be also simulated using a BFS traversal with starting vertex v . Hence, the above procedure requires $O(\ell(n)m \log^2 n)$ time. Furthermore, by duality, a hitting set L can be transformed in the set-cover $\{\overline{B}(x, r) : x \in L\}$. Therefore, if there is no set-cover of size $\ell(n)$ at

most, then there is no hitting set of size $\ell(n)$ at most either. Let us now assume the existence of a set-cover \mathcal{S} of size $O(\ell(n) \log n)$ at most. Then, again by duality, $L = \{x : \overline{B}(x, r) \in \mathcal{S}\}$ is a hitting set. \blacksquare

9 Computing the diameter in graphs with small diameter certificate

The existence of a similar result for diameter certificates remains open. However, the following weaker result can be proved:

Theorem 8 *For every class of graphs \mathcal{G} such that every graph with n vertices has a diameter certificate of size $\ell(n)$ at most, there exists a randomized (Monte Carlo) algorithm for computing the diameter within \mathcal{G} in $O(m\sqrt{\ell(n)n} \log^{3/2} n)$ time with high probability.*

In order to prove this above result, we need the following variant of [35, Lemma 2]:

Lemma 1 *For every graph $G = (V, E)$, for any $\varepsilon \in (0, 1)$, there is an algorithm that computes, for every vertex v , some value $r(v) \leq e(v)$ such that $|B[v, r(v)]| \geq (1 - \varepsilon)n$ with high probability. The algorithm runs in $O(\varepsilon^{-1}m \log n)$ time with high probability.*

Proof. If $\varepsilon \leq \frac{2 \log n}{n}$ we can perform a BFS traversal from each vertex in time $O(nm) = O(\varepsilon^{-1}m \log n)$ and get $e(v)$ for all $v \in V$, allowing to set $r(v) = e(v)$ with the desired complexity. Now assume $p = \frac{2 \log n}{\varepsilon n} < 1$. Every vertex v is added independently at random with probability p in a vertex subset L . By Chernoff bounds, the size of L is $O(\varepsilon^{-1} \log n)$ with high probability. Then, for every vertex v of G , we set $r(v) = e_L(v)$. For that, it suffices to perform a BFS traversal for every vertex of L , which can be done in $O(m|L|) = O(\varepsilon^{-1}m \log n)$ time with high probability.

Let $v \in V$ be arbitrary. By construction we have $r(v) \leq e(v)$. Conversely, for every $v \in V$, let $\rho(v)$ be the largest integer k such that $|B[v, k]| < (1 - \varepsilon)n$ (this value is independent from the random subset L). The probability of the event $\{L \setminus B[v, \rho[v]] = \emptyset\}$ is at most $(1 - p)^{\varepsilon n} = (1 - p)^{\frac{1}{p} \cdot 2 \log n} \leq n^{-2}$. Therefore, by a union bound over all vertices, the property that $|B[v, r(v)]| \geq (1 - \varepsilon)n$ for all vertices v simultaneously must hold with probability at least $1 - n^{-1}$. \blacksquare

We are now ready to prove Theorem 8. Roughly, the proof follows from a combination of Theorem 7 and Lemma 1.

Proof.[of Theorem 8] Let $\varepsilon \in (0, 1)$ to be fixed later in the proof. First we apply Lemma 1 for G, ε . This can be done in $O(\varepsilon^{-1}m \log n)$ time with high probability. Furthermore, with high probability it results in the computation, for every vertex v of G , of some value $r(v) \leq e(v)$ such that $|B[v, r(v)]| \geq (1 - \varepsilon)n$. Then in what follows, we present an $O(\varepsilon n \ell(n) m \log n)$ -time algorithm that for any graph $G = (V, E)$ that is in \mathcal{G} , for any positive integer D , decides whether $\text{diam}(G) \leq D$. We can compute $\text{diam}(G)$ with $O(\log n)$ calls to this algorithm, using binary search.

Our algorithm works as follows. We set $\mathcal{C} = \{B[v, D - r(v)] : v \in V\}$. Then, we apply Theorem 7 to V, \mathcal{C} , but we add the restriction that at most $O(\ell(n) \log^2 n)$ queries can be performed (else, we abort). By doing so, we either compute a set-cover $\mathcal{S} \subseteq \mathcal{C}$ of size $O(\ell(n) \log n)$ at most, or we correctly assert that no such set-cover of size $\ell(n)$ at most can exist. As it was argued before in the proof of Theorem 6, both containment queries and subset queries can be simulated using BFS traversals. Hence, the above procedure requires $O(\ell(n)m \log^2 n)$ time.

Assume first there is no set-cover of size $\ell(n)$ at most. We claim that $\text{diam}(G) > D$. Indeed, suppose by contradiction $\text{diam}(G) \leq D$, and let us consider a diameter certificate $U = \{u_1, u_2, \dots, u_q\}$ of size $q \leq \ell(n)$. We have $V = \bigcup_{i=1}^q B[u_i, \text{diam}(G) - e(u_i)]$ and for every $1 \leq i \leq q$ we get $D - r(u_i) \geq \text{diam}(G) - e(u_i)$. But then, $\{B[u_i, D - r(u_i)] : 1 \leq i \leq q\}$ should be a set-cover of size $\ell(n)$ at most, a contradiction.

From now on, we assume there is a set-cover $\mathcal{S} = \{B[v_1, D - r(v_1)], B[v_2, D - r(v_2)], \dots, B[v_q, D - r(v_q)]\}$ of size $q = O(\ell(n) \log n)$. For every $1 \leq i \leq q$, we compute $L_i = V \setminus B[v_i, r(v_i)]$. This can be done in $O(mq) = O(m\ell(n) \log n)$ time using BFS traversals. Let $L = \bigcup_{i=1}^q L_i$. We compute $e_L(v)$, for every vertex v of G , that can be done in $O(m|L|)$ time. Recall that with high probability, for every $1 \leq i \leq q$ we have $|L_i| < \varepsilon n$, and so we have $|L| = O(\varepsilon \ell(n) n \log n)$ with high probability. Furthermore, we claim that $\text{diam}(G) \leq D$ if and only if we have $\max\{e_L(v) : v \in V\} \leq D$. In one direction, if $D \geq \text{diam}(G)$, then $D \geq \text{diam}(G) = \max\{e(v) : v \in V\} \geq \max\{e_L(v) : v \in V\}$. In the other direction, let us assume that $\max\{e_L(v) : v \in V\} \leq D$, and suppose, for the sake of contradiction, $\text{diam}(G) > D$. Let $x, y \in V$ be such that $d(x, y) > D$. Since \mathcal{S} is a set-cover, we may assume without loss of generality that $x \in B[v_1, D - r(v_1)]$. We must have $y \notin L_1$ because $e_{L_1}(x) \leq e_L(x) \leq D < d(x, y)$. However, this implies that $y \in B[v_1, r(v_1)]$, and so, by the triangle inequality, $d(x, y) \leq d(x, v_1) + d(v_1, y) \leq D - r(v_1) + r(v_1) = D < d(x, y)$, giving a contradiction.

The total running time of the algorithm is in $O(\varepsilon^{-1} m \log n + \log n \times \varepsilon n \ell(n) m \log n)$ with high probability. This is minimized for $\varepsilon = \Theta\left(\frac{1}{\sqrt{n\ell(n)\log n}}\right)$. \blacksquare

Before concluding this section, the last two results are simple conditions ensuring the existence of constant-size certificates for radius or diameter.

Proposition 3 *For an arbitrary graph G with $\text{diam}(G) \geq 2 \text{rad}(G) - 1$, any diametral pair of vertices x, y forms a minimum radius certificate of G . Furthermore, $\text{rad}(G) = \lfloor \frac{d(x, y) + 1}{2} \rfloor$.*

Proof. Let x, y be an arbitrary diametral pair of G , i.e., $d(x, y) = \text{diam}(G)$. If there is a vertex $z \in V$ with $\max\{d(z, x), d(z, y)\} \leq \text{rad}(G) - 1$ then $\text{diam}(G) = d(x, y) \leq d(z, x) + d(z, y) \leq 2 \text{rad}(G) - 2$, and a contradiction arises. The equality $\text{rad}(G) = \lfloor \frac{d(x, y) + 1}{2} \rfloor$ comes from the hypothesis $\frac{\text{diam}(G) + 1}{2} \geq \text{rad}(G)$ and from $\text{diam}(G) \leq 2 \text{rad}(G)$ which implies $\frac{\text{diam}(G) + 1}{2} \leq \text{rad}(G) + \frac{1}{2}$. \blacksquare

By Theorem 6, the above condition implies the existence of an $O(m \log^3 n)$ -time algorithm for computing the radius with high probability in that case. In particular, let us briefly introduce a class of graphs such that the condition of Proposition 3 always holds. Namely, an *eccentricity-preserving spanning tree* of a graph $G = (V, E)$ is a spanning tree T such that $e_T(v) = e_G(v)$ for every vertex v . Note that being given an efficient algorithm for computing an eccentricity-preserving spanning tree of G when it exists, we could compute all eccentricities of G in additional $O(n)$ time. But the existence of such an algorithm is open. However, it was observed in [45] that graphs G that admit an eccentricity-preserving spanning tree must satisfy $\text{diam}(G) \geq 2 \text{rad}(G) - 1$. Therefore, to our best knowledge, Theorem 6, in combination with Proposition 3, implies the first almost-linear time algorithm for radius computation in this class of graphs. Other such examples are discussed in Section 21.

Proposition 4 *For an arbitrary graph G with $\text{diam}(G) = 2 \text{rad}(G)$, any central vertex c forms a smallest diameter certificate of G .*

Proof. In every graph G with $\text{diam}(G) = 2\text{rad}(G)$, for an arbitrary central vertex $c \in C(G)$ and every vertex $v \in V$, we have $d(v, c) + e(c) \leq e(c) + e(c) = 2\text{rad}(G) = \text{diam}(G)$. \blacksquare

Note that on graphs G such that $\text{diam}(G) = 2\text{rad}(G)$, their radius, and so their diameter as well, can be computed in $O(m \log^3 n)$ time with high probability.

10 Approximating radius certificate

We define a *strict radius certificate* of $G = (V, E)$ as a set $L \subseteq V$ such that $e_L(v) \geq \text{rad}(G)$ for all $v \in V$, and $e_L(v) > \text{rad}(G)$ for all $v \in V \setminus C(G)$. In other words, a strict radius certificate is a radius certificate such that the centers are the only nodes whose lower bound matches the radius. The following result shows that it is possible to compute a radius certificate having size within a logarithmic factor from a minimum strict radius certificate.

Theorem 9 *Given an arbitrary graph G , its radius, a center, and a radius certificate of size $O(\ell^* \log n)$ can be computed using $O(\ell^* \log^2 n)$ one-to-all distance queries in $O(\ell^* m \log^2 n)$ time with high probability where ℓ^* is the minimum size of a strict radius certificate.*

The main idea is to find a constant approximation of the size of a minimum strict radius certificate by binary search without relying on a guess of the radius R of the graph. For that, we need to dive into the technique of [47] and use a key property of the respective set-cover instances we consider for all possible values of R to obtain a Las Vegas algorithm.

Proof.[of Theorem 9] Consider a graph $G = (V, E)$ and a strict radius certificate L^* of G with minimum size. Starting from an estimate ℓ of $|L^*|$, we try to construct a radius certificate L as described below. If the construction fails, then we double our estimate ℓ and start again until the construction succeeds. Initially, we set $\ell = 2$.

Given the current estimate ℓ , we grow a lower certificate L which is initially empty. We use random sampling to find an appropriate vertex x to add to L . First, choose a constant c according to multiplicative Chernoff bounds so that the sum S of $c\ell \log n$ independent Bernoulli random variables taking value 1 with probability $\frac{1}{\ell}$ satisfies $\frac{c \log n}{2} < S < 2c \log n$ with probability greater than $1 - 1/n^2$ for any integer $\ell' > 1$. We randomly, uniformly and independently sample a set Y of $2c\ell \log n$ nodes in V . We can then perform a BFS traversal from each sampled node $y \in Y$ and obtain distances $d(x, y)$ for all $(x, y) \in V \times Y$. In particular, we obtain the eccentricity $e(y)$ of all vertices $y \in Y$.

As L^* is a radius certificate, there must exist a node $x \in L^*$ hitting at least $2c \frac{\ell}{|L^*|} \log n$ coballs $\overline{B}(y, \text{rad}(G))$ centered at vertices $y \in Y$ (we say that x hits $\overline{B}(y, r)$ when $d(x, y) \geq r$). Moreover, for $\ell \geq |L^*|$, the choice of c implies that x hits $c \log n$ such coballs with probability greater than $1 - 1/n^2$. For a given set $U \subseteq V$ of vertices, a node $v \in V$ and a distance r , we define the number $h_U(v, r)$ of coballs of radius r with center in U that are hit by v . A key observation is that $h_U(v, r)$ is non-increasing in r . Using the selection algorithm of [7] for finding the element with rank $c \log n$ in the vector of distances $d(v, y)$ for fixed v and $y \in Y$, we can thus find the largest value r_v such that $h_Y(v, r_v) \geq c \log n$. We then select a node x such that $r_x = \max_{v \in V} r_v$, and add x to L . Note that the choice of c also implies that x hits at least a fraction $\frac{1}{2\ell}$ of coballs of radius r_x with probability greater than $1 - 1/n^2$. We now have an estimate $R = r_x$ of the radius of G . If $\ell \geq |L^*|$, recall that it satisfies $R \geq \text{rad}(G)$ with probability greater than $1 - 1/n^2$.

We iterate this operation with the set $W = \{v \in V : e_L(v) < R\}$ of vertices with lower bound less than our radius estimate R by sampling nodes in W and proceeding similarly as above. Each

time we add a node x to L , we update L and $R = \min_{x \in L} r_x$. As a fraction at least $\frac{1}{2\ell}$ of coballs of radius R and center in W are hit by x with probability greater than $1 - 1/n^2$, the size of W shrinks by a factor at most $1 - \frac{1}{2\ell}$ in each iteration. Note also that the nodes v removed from W by the update of $e_L(v)$ or the update of R satisfy $e(v) \geq e_L(v) \geq r_x$. After $2\ell \log n$ iterations, the size of W is less than $\ell \log n$ with high probability. If it is not the case, we start again from $L = \emptyset$. (As it is a rare event, the complexity is not impacted.) Otherwise, we then perform a BFS from each remaining node and add a furthest node f of v to L for each $v \in W$. For such nodes f , we define $r_f = e(v)$ which ensures $r_f \geq \text{rad}(G)$ and $e(v) = e_L(v) \geq r_f$. At this point, we check that we have $R = \min_{v \in V} e_L(v)$. If our estimate ℓ is at least $|L^*|$, we have $r_x \geq \text{rad}(G)$ for all $x \in L$ with probability greater than $1 - \frac{2\ell \log n}{n^2}$ by union bound, and we can then deduce $R \geq \text{rad}(G)$. As each node v gets removed from W at some point, it satisfies $e(v) \geq e_L(v) \geq r_x$ for some $x \in L$, which implies $R \leq \text{rad}(G)$. If ever we observe $R \neq \min_{v \in V} e_L(v)$, we conclude that our estimate ℓ was too small, we double it, and start again from $L = \emptyset$.

To confirm that the construction has succeeded, we try to find a center c with a similar procedure using the fact that L^* is a strict radius certificate. Consider the set $D = \{v \in V : e_L(v) = R\}$ of vertices with minimum lower-bound. We randomly, uniformly and independently sample a set Y of $2\ell \log n$ nodes in D , perform a BFS traversal from each $y \in Y$ to obtain distances $d(x, y)$ for all $(x, y) \in V \times Y$. If some vertex $y \in Y$ has eccentricity $e(y) = R$, we have succeeded: y is a center, $R = \text{rad}(G)$ and L is a radius certificate. Otherwise, we have $e(y) > R$ for all $y \in Y$, and Y is a uniform random sample of $D \setminus C(G)$. We can thus proceed similarly as above and find a node x such that $c \log n$ vertices $y \in Y$ at least satisfy $d(x, y) > R$. If we do not find such a node, we again conclude that ℓ was too small and start from scratch with new estimate 2ℓ . Otherwise, the choice of x implies that a fraction at least $\frac{1}{2\ell}$ of vertices $v \in D \setminus C(G)$ satisfy $d(x, v) > R$ with probability greater than $1 - 1/n^2$. These vertices are then removed from D and we iterate until we either find a center, conclude that ℓ is too small or D has size less than $\ell \log n$ after at most $2\ell \log n$ iterations with high probability. (If none of the three cases occurs, we start again from $L = \emptyset$.) In the latter case we can perform a BFS traversal for each remaining node and either find a center or conclude that all nodes have eccentricity larger than R , and conclude again that ℓ was too small.

Overall, we stop with high probability as soon as $\ell \geq |L^*|$. We may (with very low probability) double ℓ several times, but the algorithm stops in any case if ever $\ell \geq n$ as we then sample all nodes. The algorithm thus always computes a radius certificate L and a center c . With high probability, we have $|L| = O(|L^*| \log n)$ and $O(|L^*| \log^2 n)$ BFS traversals are performed in total. The rest of the computation is dominated by selection algorithm of [7] which costs $O(n\ell \log n) = O(m\ell \log n)$ per sampling. The overall computation time is thus $O(m|L^*| \log^2 n)$ with high probability. \blacksquare

11 Approximating all-eccentricity certificate

Theorem 10 *Given an arbitrary n -node graph G , all eccentricities, a tight lower certificate L of size $O(\ell^* \log n)$ and the minimum tight upper certificate U^\preceq can be computed in $O(m\ell^* |U^\preceq| \log^2 n)$ time with high probability where ℓ^* denotes the size of a minimum tight lower certificate.*

We provide a Las Vegas algorithm by combining the greedy set-cover approach of Theorem 6 with properties of the minimum tight upper certificate developed in Section 4. More precisely, we rely on the following lemmas.

Lemma 2 *Given a graph G , a vertex u and a furthest node f of u (i.e. $e(u) = d(u, f)$), and a shortest path $u = v_1, v_2, \dots, v_k = f$, all tight upper vertex-certificates for u on the path form a prefix v_1, \dots, v_i for some $i \in [k]$.*

Proof. Suppose that for some $j \in [k]$, v_j is a tight upper vertex-certificate for u , that is $e(u) = d(u, f) = d(u, v_j) + e(v_j)$. We thus have $e(v_j) = d(u, f) - d(u, v_j) = d(v_j, f)$ as $v_1, v_2, \dots, v_k = f$ is a shortest path. Now consider v_i with $i \leq j$. By triangle inequality, we have $e(v_i) \leq d(v_i, v_j) + e(v_j) = d(v_i, v_j) + d(v_j, f) = d(v_i, f)$. As $v_i, \dots, v_k = f$ is a shortest path, we also have $e(v_i) \geq d(v_i, f)$ and conclude $e(v_i) = d(v_i, f)$. Finally, as v_i is on a shortest path from u to f , we obtain $d(u, f) = d(u, v_i) + d(v_i, f) = d(u, v_i) + e(v_i)$. This allows to conclude. ■

Lemma 3 *Given a graph G , a vertex u and a furthest node f of u (i.e. $e(u) = d(u, f)$), we have $u \in U^{\preceq}$ if and only if $e(z) > d(z, f)$ for all vertices $z \in I(u, f) \setminus \{u\}$ where $I(u, f) = \{v : d(u, v) + d(v, f) = d(u, f)\}$ is the interval between u and f .*

Proof. If $u \notin U^{\preceq}$, then there exists a tight upper vertex-certificate $z \in U^{\preceq}$ for u by Proposition 1. It thus satisfies $e(u) = d(u, z) + e(z) = d(u, f)$ which implies $d(u, f) \geq d(u, z) + d(z, f)$ and thus $d(u, f) = d(u, z) + d(z, f)$ and $d(z, f) = e(z)$ by triangle inequality. We therefore get $z \in I(u, f)$. Moreover, it satisfies $e(z) \leq d(z, f)$ and $z \neq u$ since $z \in U^{\preceq}$ and $u \notin U^{\preceq}$.

Conversely, assume $u \in U^{\preceq}$ and consider a vertex $z \in I(u, f) \setminus \{u\}$. We thus have $d(u, f) = d(u, z) + d(z, f)$. By definition of U^{\preceq} , u is maximal for \preceq , and we cannot have $u \preceq z$, that is $e(z) \neq e(u) - d(u, z)$. As $e(u) - d(u, z) = d(z, f) \leq e(z)$, we deduce $e(z) > d(z, f)$. ■

Lemma 4 *Given a graph $G = (V, E)$ with minimum tight lower certificate L^* , and a vertex $u \in V$, a tight upper vertex-certificate z for u can be computed in $O(m|L^*| \log^2 n)$ time with probability greater than $1 - O(1/n^2)$.*

Proof. Consider a furthest node f of u (i.e. $e(u) = d(u, f)$) and the interval $I(u, f) = \{v : d(u, v) + d(v, f) = d(u, f)\}$. This interval can be identified by performing two BFS traversals, one from u , and one from f . We search for a tight upper vertex-certificate $z \in I(u, f)$ for u such that $r = d(z, f)$ is minimal. Note that a vertex $z \in I(u, f)$ satisfies $d(u, f) = d(u, z) + d(z, f)$ and it is therefore a tight upper vertex-certificate for u if and only if it has eccentricity $e(z) = d(z, f)$. We thus search for the minimum distance r such that some vertex $z \in I(u, f)$ at distance r from f satisfies $e(z) = d(z, f) = r$. We use binary search to find r by building a tight lower certificate L' as follows. Starting with an estimate $\ell = 2$ of $|L^*|$, $L' = \emptyset$ and a distance $r_1 \in [0, d(u, f)]$ we consider the set $W \subseteq I(u, f)$ of vertices v that are at distance r_1 from f and such that $e_{L'}(v) \leq r_1$ (we have initially $W = \{v \in I(u, f) : d(v, f) = r_1\}$). We randomly, uniformly and independently sample a set Y of $2c\ell \log n$ nodes in W where c is chosen according to multiplicative Chernoff bounds so that the sum S of $c\ell \log n$ independent Bernoulli random variables taking value 1 with probability $\frac{1}{\ell}$ satisfies $\frac{c \log n}{2} < S < 2c \log n$ with probability greater than $1 - 1/n^4$ for any integer $\ell' > 1$. If $|W| < 2c\ell \log n$, we use $Y = W$. We perform a BFS traversal from each sampled node $y \in Y$ and obtain distances $d(x, y)$ for all $(x, y) \in V \times Y$. In particular, we obtain the eccentricity $e(y)$ of all vertices $y \in Y$.

If some vertex $y \in Y$ has eccentricity $e(y) = d(y, f) = r_1$, then y is a tight upper vertex-certificate for u and we know $r \leq r_1$. Otherwise, we have $e(y) > r_1$ for all $y \in Y$. We then count

for each node $x \in V$ the number of vertices $y \in Y$ for which it is a furthest node, that is when $e(y) = d(x, y)$. The definition of L^* implies that some node $x \in L^*$ is a furthest node of at least $2c \frac{\ell}{|L^*|} \log n$ vertices of Y . If $\ell \geq |L^*|$ it is the case for at least $c \log n$ vertices with probability greater than $1 - 1/n^4$ by the choice of c . If we cannot find such a vertex x , we conclude that our estimate ℓ is smaller than $|L^*|$ and restart from scratch with estimate 2ℓ . If we do find a vertex x that is a furthest node of at least $c \log n$ vertices of Y , we add it to L' . Note that the choice of c also implies that x is a furthest node of a fraction at least $\frac{1}{2\ell}$ of W with probability greater than $1 - 1/n^4$. Note that those vertices v may be removed from W after adding x to L' and updating lower-bounds $e_{L'}(v)$ for $v \in I(u, f)$ if $e_{L'}(v) > r_1$. In particular, all vertices from Y are removed. We iterate this sampling process until we either find y with eccentricity $e(y) = d(y, f) = r_1$, or restart with estimate 2ℓ , or W gets size less than $2c\ell \log n$. This latter case occurs after $2\ell \log n$ iterations at most and we then compute the eccentricities of all remaining vertices in W . Finally, if no vertex y with eccentricity $e(y) = d(y, f) = r_1$ is found, we have $e(y) > d(y, f)$ for all vertices of $I(u, f)$ at distance at most r_1 from f by Lemma 2, and we know $r > r_1$.

We then repeat this process for appropriate distances r_2, \dots, r_k . For $\ell \geq |L^*|$, we find the value of r , and a vertex $y \in I(u, f)$ satisfying $e(y) = r = d(y, f)$, after $k = O(\log \text{diam}(G))$ probes for finding r using $O(c\ell \log^2 n)$ BFS traversals with probability greater than $1 - O(\frac{\log n}{n^3})$. Moreover, the node y found is in U^{\preceq} by Lemma 3. The overall number of traversals for all trials with estimates $\ell = 2, 4, \dots$ is $O(|L^*| \log^2 n)$ with probability greater than $1 - O(1/n^2)$. ■

Proof.[of Theorem 10] Let L^* be a minimum tight lower certificate, and let $\ell^* = |L^*|$ be its size. We use binary search to find a value $\ell \geq \ell^*$. Starting with $\ell = 2$, we grow a lower certificate L and an upper certificate U as follows (both are initially empty). Let W denote the set of nodes v with non-matching lower and upper bounds, i.e. satisfying $e_L(v) < e^U(v)$ ($W = V$ initially as we consider that $e^U(v) = \infty$ when $U = \emptyset$). Randomly, uniformly and independently sample a set Y of $2c\ell \log n$ nodes in W where c is chosen similarly as in the proof of Lemma 4. We perform a BFS traversal from each sampled node $y \in Y$ and obtain distances $d(x, y)$ for all $(x, y) \in V \times Y$. In particular, we obtain the eccentricity $e(y)$ of the $2c\ell \log n$ vertices $y \in Y$.

As long as we find $2c\ell \log n$ vertices y such that $e_L(y) < e(y)$, we proceed as in Lemma 4 by adding to L a vertex x that is a tight lower vertex-certificate for at least $c \log n$ vertices in Y . With probability greater than $1 - 1/n^2$, it is also a tight lower vertex-certificate for a fraction at least $\frac{1}{2\ell}$ of vertices in W with untight lower-bound. Again, if no such vertex is found, we double our estimate ℓ of ℓ^* and restart with empty certificates.

As soon as we sample a node y such that $e(y) = e_L(y)$, we apply Lemma 4 with $u = y$ to find a tight upper vertex-certificate z for y using at most $O(\ell^* \log^2 n)$ BFS traversals with probability greater than $1 - O(1/n^2)$, and add z to U . We then update W by removing vertices satisfying $e_L(v) = e^U(v)$. Note that we remove y in particular. We then continue sampling until we find $2c\ell \log n$ vertices with untight lower bound. If ever the size of W is less than $2c\ell \log n$, we finish by performing a BFS traversal from each remaining vertex v and adding a furthest vertex of v to L for each $v \in W$.

Similarly as in the proof of Lemma 4, we overall perform $O(\ell^* \log^2 n)$ BFS traversals with high probability for the computation of L . However, each time we sample a vertex u with tight lower-bound we perform $O(\ell^* \log^2 n)$ BFS traversals with probability greater than $1 - O(1/n^2)$ to find a tight upper vertex-certificate $z \in U^{\preceq}$ for u . Note that z is added only once to U . The reason is that if there exists $u' \in W$ such that z is also a tight upper vertex-certificate for u' , we have $e^U(u') = e(u')$ after adding z to U . Although u' may still be in W when $e_L(u') < e(u')$, it will be removed as soon as L is modified so that $e_L(u') = e(u')$. The computation of U thus costs

$O(\ell^* |U^\preceq| \log^2 n)$ BFS traversals overall with high probability. ■

Part III

Practical algorithms

12 Radius computation and certification

We now propose a radius algorithm with complexity parameterized by the number of antipodes in the input graph. Similarly to previous algorithms [48, 9], it maintains lower bounds on eccentricities of all nodes and performs one-to-all distance queries from nodes with minimal lower bound as a first ingredient. Similarly to the two-sweeps and four-sweeps heuristics [39, 43, 22], it performs one-to-all distance queries from antipodes of previous query source as a second ingredient. However, contrarily to these heuristics, it iterates until an exact solution is obtained (together with a radius certificate).

The idea of the algorithm is to maintain both a set K of nodes with distinct antipodes and a lower certificate L (initially empty). We iteratively select a node u with minimal lower-bound $e_L(u)$ and perform a one-to-all distance query from u . As long as this bound is not tight (i.e., $e_L(u) < e(u)$), we add $\text{Antipode}_r(u)$ to L and u to K while eccentricity lower-bounds are improved accordingly. (The fact that the bound is not tight implies that no antipode of u could previously be in L .) As soon as the bound is tight (i.e., $e_L(u) = e(u)$), we then claim that u is a center (i.e., its eccentricity is minimal) and return $e(u)$ as the radius and L as radius certificate. Algorithm 1 formally describes the whole method.

```

Input: A graph  $G$  and a ranking  $r$  of its node set  $V$ .
Output: The radius  $\text{rad}(G)$  of  $G$ , a center  $c$  and a radius certificate  $L$ .
 $L := \emptyset$  /* Lower certificate (tentative covering with  $\{\overline{B}(u, \text{rad}(G)) : u \in V\}$ ).
*/
Maintain  $e_L(v) = \max_{x \in L} d(v, x)$  (initially 0) for all  $v \in V$ .
 $K := \emptyset$  /* Packing for  $\{\text{Antipode}_r^{-1}(u) : u \in V\}$ . */
Do
  Select  $u \in V$  such that  $e_L(u)$  is minimal.
   $D_u := \text{DistFrom}(G, u)$  /* Distances from  $u$ . */
   $e(u) := \max_{v \in V} D_u(v)$  /* Eccentricity of  $u$ . */
  If  $e(u) = e_L(u)$  then
    | return  $e(u)$ ,  $u$ , and  $L$ 
  else
    |  $a := \text{argmax}_{v \in V} (D_u(v), r(v))$  /* Antipode of  $u$  for  $r$ . */
    |  $D_a := \text{DistFrom}(G, a)$  /* Distances from  $a$ . */
    |  $K := K \cup \{u\}$ 
    |  $L := L \cup \{a\}$ 
    | For  $v \in V$  do  $e_L(v) := \max(e_L(v), D_a(v))$ 
while  $\min_{u \in V} e_L(u) < \min_{u \in K} e(u)$ .
Return  $e(c)$ ,  $c$  and  $L$  where  $c = \text{argmin}_{u \in K} e(u)$ .

```

Algorithm 1: Computing the radius, a center and a radius certificate.

Note the primal-dual flavor of this algorithm as the set K (which has same size as L) is a packing for $\{\text{Antipode}_r^{-1}(u) : u \in V\}$ which is a restricted collection of $\{\overline{B}(u, \text{rad}(G)) : u \in V\}$ for which the computed certificate L is a covering.

Theorem 11 *Given a graph G and a ranking r on its node set V , Algorithm 1 computes its radius $\text{rad}(G)$, a center c and a radius certificate $L \subseteq \text{Antipode}_r(V)$ with $2|L| + 1 = O(|\text{Antipode}_r(V)|)$ one-to-all distance queries in $O(m|\text{Antipode}_r(V)|)$ time.*

Proof. We first prove the termination of Algorithm 1. Consider an iteration where we add the antipode a of the selected node u to L . We cannot have $a \in L$ as we would then have $e_L(u) = e(u)$ which is the termination case. In other words, nodes added to K have distinct antipodes and K is a packing for $\{\text{Antipode}_r^{-1}(u) : u \in V\}$. As long as the do-while loop runs, each iteration adds a new node to L . If ever we reach the point where $L = \text{Antipode}_r(V)$, then the lower-bound of each node $u \in V$ is tight: $e_L(u) = e(u)$. The next iteration must then terminate. The complexity is straightforward: at most $2|L|+1$ one-to-all distance queries are performed and $|L| \leq |\text{Antipode}_r(V)|$ as $L \subseteq \text{Antipode}_r(V)$.

We now prove the correctness of Algorithm 1. Consider an iteration of the do-while loop. By the choice of u , we then have $e_L(v) \geq e_L(u)$ for all $v \in V$. If the termination case $e_L(u) = e(u)$ occurs, we have $e(v) \geq e_L(v) \geq e_L(u) = e(u)$ for all $v \in V$. This ensures that u has minimum eccentricity (it is a center). We thus have $\text{rad}(G) = e(u)$ and L is a radius certificate as $e_L(v) \geq \text{rad}(G)$ for all $v \in V$. Finally, if ever the condition for continuing the do-while loop is false, we have $\min_{u \in V} e_L(u) \geq \min_{u \in K} e(u)$. For $c = \text{argmin}_{u \in K} e(u)$, we thus have $\text{rad}(G) = \min_{u \in V} e(u) \geq \min_{u \in V} e_L(u) \geq e(c) \geq \text{rad}(G)$. That is, L is a radius certificate and c is a center. \blacksquare

In practice, we observe very fast convergence compared to $|\text{Antipode}_r(V)|$ (see Section 16). We can give the following argument for that. The node u selected at each iteration satisfies $e_L(u) = \min_{v \in V} e_L(v) \leq \min_{v \in V} e(v) \leq \text{rad}(G)$. We thus have $\max_{x \in L} d(u, x) \leq \text{rad}(G)$, that is $u \in \cap_{x \in L} B[x, \text{rad}(G)]$. It appears that the eccentricity of antipodes is generally large compared to radius in practical graphs, and the set $\cap_{x \in L} B[x, \text{rad}(G)]$ tends to quickly shrink toward the set of centers as we add antipodes to L .

13 Minimum eccentricity selection

The core of the above radius algorithm is a general technique depending on a user-defined function f that we call *minimum eccentricity selection* (minES) for f . It is a procedure that returns a node with minimum eccentricity with respect to f . Its amortized complexity is low in graphs with few antipodes. More precisely, for a given graph G and a function f that maps a node v and an estimation ℓ of $e(v)$ to a value, it provides a function argminES returning a node u such that $f(u, e(u))$ is minimum as long as f is non-decreasing, i.e., $f(v, \ell) \leq f(v, \ell')$ for $\ell \leq \ell'$ for all v . A similar function minES returns the value of $f(u, e(u))$ for such node u . The challenge here is to avoid the computation of all eccentricities.

We implement such a selection by maintaining lower bounds of all eccentricities as in Algorithm 1 and by using these lower bounds as estimates for true eccentricities. When the selection procedure is called, a node u which is minimum according to lower bounds is considered. Such a node is found by evaluating $f(v, e_L(v))$ for all $v \in V$ where $e_L(v)$ denotes the lower bound stored for a node v . A one-to-all distance query from u is then performed. If its eccentricity happens to be equal to

its lower-bound $e_L(u)$ we claim that $f(u, e(u))$ is minimum and return that node. Otherwise, the antipode of u is used to improve lower bounds before trying again. Algorithm 2 formally describes this.

```

L := ∅ /* Lower certificate. */
Maintain  $e_L(v) = \max_{x \in L} d(v, x)$  (initially 0) for all  $v \in V$ .
Function argminES( $G, r, L, e_L, f$ )
  Repeat
     $u := \operatorname{argmin}_{v \in V} f(v, e_L(v))$ 
     $D_u := \operatorname{DistFrom}(G, u)$  /* Distances from  $u$ . */
     $e(u) := \max_{v \in V} D_u(v)$  /* Eccentricity of  $u$ . */
    If  $e_L(u) = e(u)$  then
      | return  $u$ 
    else
      |  $a := \operatorname{argmax}_{v \in V} (D_u(v), r(v))$  /* Antipode of  $u$  for  $r$ . */
      |  $D_a := \operatorname{DistFrom}(G, a)$  /* Distances from  $a$ . */
      |  $L := L \cup \{a\}$ 
      | For  $v \in V$  do  $e_L(v) := \max(e_L(v), D_a(v))$ 
  Function minES( $G, r, L, e_L, f$ )
  |  $u := \operatorname{argminES}(G, r, L, e_L, f)$ 
  | Return  $f(u, e_L(u))$ 

```

Algorithm 2: Minimum eccentricity selection with respect to function f .

Proposition 5 *Given a graph G and a ranking r of its node set V , we consider a function f such that $f(v, \ell)$ can be evaluated for any $v \in V$ and $\ell \leq e(v)$. If $f(v, \cdot)$ is non-decreasing for all $v \in V$, i.e., $f(v, \ell) \leq f(v, \ell')$ for $\ell \leq \ell'$, function `argminES` of Algorithm 2 returns a node u such that $f(u, e(u))$ is minimal and updates the lower certificate L such that $e_L(u) = e(u)$ and $f(v, e_L(v)) \geq f(u, e(u))$ for all $v \in V$. Moreover it can perform k computations of `argmin` $f(u, e(u))$ using $k + 2|L'|$ one-to-all distance queries and $(k + 2|L'|)n$ calls to f where $L' \subseteq \operatorname{Antipode}_r(V)$ denotes the set of nodes added to L . It runs in $O(m|\operatorname{Antipode}_r(V)|)$ time when $f(v, \ell)$ can be evaluated in $O(\deg(v))$ time.*

Proof. The correctness of the selection comes from the fact that $f(v, \cdot)$ is non-decreasing: if $e_L(u) = e(u)$, we then have $f(u, e(u)) = f(u, e_L(u)) \leq \min_{v \in V} f(v, e_L(v)) \leq \min_{v \in V} f(v, e(v))$. The case $e_L(u) < e(u)$ can only occur if the antipode of u was not in L and happens at most $|\operatorname{Antipode}_r(V)|$ times in total. In particular, each call to `argminES` terminates. If an algorithm makes x calls to the `argminES`, the number of successful iterations where $e_L(u) = e(u)$ is precisely x while the number of unsuccessful iterations is at most the number of nodes added to L . For each such iteration we perform 2 one-to-all distance queries instead of 1. The total number of queries is thus $k + 2|L'|$. In all cases, we perform n calls to f per iteration: one for each node v . Assuming that $f(v, \ell)$ can be computed in $O(\deg(v))$ time, this costs $O(m)$ per iteration. The overall time required is thus in $O(m|\operatorname{Antipode}_r(V)|)$. ■

As an example of usage, Algorithm 1 for radius is equivalent to the following algorithm using our minimum eccentricity selection for the basic function $v, \ell \mapsto \ell$.

$L := \emptyset; e_L(v) := 0$ for all $v \in V$.
Function $\text{ecc}(v, \ell)$: **return** ℓ
 $c := \text{argminES}(G, r, L, e_L, \text{ecc})$
return $e_L(c), c, L$

Algorithm 3: Radius computation through minimum eccentricity selection with respect to $f(v, \ell) = \ell$ (equivalent to Algorithm 1).

As another example, the function f can be used to select a node with minimum eccentricity in a set W of nodes when $f(v, \ell)$ returns ℓ if $v \in W$ and ∞ otherwise. One can easily check that $f(v, \cdot)$ is non-decreasing for all $v \in V$. Using this function in a variant of Algorithm 3 then allows to find a node with minimum eccentricity in a given subset W of nodes. We use our minimum eccentricity selection as an optimization for diameter computation and as a core tool for computing all eccentricities in the next sections.

14 Diameter computation and certification

We now analyze a simple diameter algorithm. The main ingredient of the algorithm consists in maintaining upper bounds of all eccentricities and performing one-to-all distance queries from nodes with maximum upper bound. It thus follows the main line of previous practical algorithms [48, 9]. However, we present the algorithm with a more general primal-dual approach which was not noticed before. Moreover, we introduce a new technique called *delegate certificate*: after selecting a node u with maximal upper bound, it consists in performing a one-to-all distance query from any tight upper vertex-certificate for u , that is a node x such that $d(u, x) + e(x) = e(u)$ (see Section 4). A possible choice for x is u itself in which case the algorithm becomes equivalent to the variant of [48] where the selection procedure for BFS sources always selects a node with maximum upper-bound. However, we observe that choosing a node x with minimal eccentricity offers much better performances in practice (see Section 16). Our complexity analysis is independent of the choice of x , we thus present the algorithm in the most general manner.

The algorithm grows both a packing K and an upper certificate U until the upper bound $e^U(u)$ on the eccentricity of any node u is at most the maximum eccentricity of nodes in K . As long as this condition is not satisfied, a node u with maximal upper bound is selected and added to K . We then choose a tight upper vertex-certificate x for u and add it to U . Note that we now have $e^U(u) = e(u) \leq \max_{v \in K} e(v)$ and u cannot be selected again. This ensures that the termination condition is reached at some point when U is a certificate that all nodes have eccentricity at most that of a maximum-eccentricity node in K , which must thus be equal to diameter. See Algorithm 4 for a formal description.

We claim that the set K is a packing for the collection $\mathcal{D}_{1/3} = \{B(u, \frac{1}{3}(\text{diam}(G) - e(u))) : u \in V\}$ of open balls. As it has same size as the certificate U returned by the algorithm in the end, this allows to state the following theorem.

Theorem 12 *Given a graph G , Algorithm 4 computes the diameter of G , a diametral node b and a diameter certificate U of size $\pi_{1/3}$ at most, with $O(\pi_{1/3})$ one-to-all distance queries in $O(m\pi_{1/3})$ time, where π_α is the maximum packing size for the collection of open balls $\mathcal{D}_\alpha = \{B(u, \alpha(\text{diam}(G) - e(u))) : u \in V\}$ for $\alpha > 0$. It approximates minimum diameter certificate within a factor $\frac{\pi_{1/3}}{\pi_{[1]}}$ where $\pi_{[1]}$ is the maximum packing size for the collection $\mathcal{D}_{[1]} = \{B[u, \text{diam}(G) - e(u)] : u \in V\}$.*

<p>Input: A graph G and a ranking r of its node set V.</p> <p>Output: The diameter $\text{diam}(G)$ of G, a diametral node b and a diameter certificate U.</p> <p>$U := \emptyset$ /* Upper certificate (tentative covering with $\{B[u, \text{diam}(G) - e(u)] : u \in V\}$). */</p> <p>Maintain $e^U(u) = \min_{x \in U} d(u, x) + e(x)$ (initially ∞) for all $v \in V$.</p> <p>$K := \emptyset$ /* Packing for $\{B(u, \frac{1}{3}(\text{diam}(G) - e(u))) : u \in V\}$. */</p> <p>Do</p> <table style="border-left: 1px solid black; border-right: 1px solid black; border-bottom: 1px solid black; padding-left: 10px;"> <tr> <td style="padding: 2px 5px 2px 10px;">Select u such that $e^U(u)$ is maximal.</td> <td style="padding: 2px 5px 2px 10px;">*/</td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">$D_u := \text{DistFrom}(G, u)$ /* Distances from u.</td> <td style="padding: 2px 5px 2px 10px;">*/</td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">$e(u) := \max_{v \in V} D_u(v)$ /* Eccentricity of u.</td> <td style="padding: 2px 5px 2px 10px;">*/</td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">$K := K \cup \{u\}$</td> <td></td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">1 Select x such that $d(u, x) + e(x) = e(u)$. /* Delegate certificate for u.</td> <td style="padding: 2px 5px 2px 10px;">*/</td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">2 $D_x := \text{DistFrom}(G, x)$ /* Distances from x.</td> <td style="padding: 2px 5px 2px 10px;">*/</td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">$e(x) := \max_{v \in V} D_x(v)$ /* Eccentricity of x.</td> <td style="padding: 2px 5px 2px 10px;">*/</td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">$U := U \cup \{x\}$</td> <td></td> </tr> <tr> <td style="padding: 2px 5px 2px 10px;">For $v \in V$ do $e^U(v) := \min(e^U(v), D_x(v) + e(x))$</td> <td></td> </tr> </table> <p>while $\max_{u \in K} e(u) < \max_{u \in V} e^U(u)$</p> <p>Return $e(b)$, b and U where $b \in K$ satisfies $e(b) = \max_{u \in K} e(u)$.</p>	Select u such that $e^U(u)$ is maximal.	*/	$D_u := \text{DistFrom}(G, u)$ /* Distances from u .	*/	$e(u) := \max_{v \in V} D_u(v)$ /* Eccentricity of u .	*/	$K := K \cup \{u\}$		1 Select x such that $d(u, x) + e(x) = e(u)$. /* Delegate certificate for u .	*/	2 $D_x := \text{DistFrom}(G, x)$ /* Distances from x .	*/	$e(x) := \max_{v \in V} D_x(v)$ /* Eccentricity of x .	*/	$U := U \cup \{x\}$		For $v \in V$ do $e^U(v) := \min(e^U(v), D_x(v) + e(x))$	
Select u such that $e^U(u)$ is maximal.	*/																	
$D_u := \text{DistFrom}(G, u)$ /* Distances from u .	*/																	
$e(u) := \max_{v \in V} D_u(v)$ /* Eccentricity of u .	*/																	
$K := K \cup \{u\}$																		
1 Select x such that $d(u, x) + e(x) = e(u)$. /* Delegate certificate for u .	*/																	
2 $D_x := \text{DistFrom}(G, x)$ /* Distances from x .	*/																	
$e(x) := \max_{v \in V} D_x(v)$ /* Eccentricity of x .	*/																	
$U := U \cup \{x\}$																		
For $v \in V$ do $e^U(v) := \min(e^U(v), D_x(v) + e(x))$																		

Algorithm 4: Computing diameter and a diameter certificate. The basic version of the algorithm consists in selecting $x := u$ in Line 1. (The redundant one-to-all distance query in Line 2 can then be omitted.)

Proof. We already argued the termination and the correctness of the algorithm above. We thus show the packing property of K . Suppose for the sake of contradiction that K is not a packing for $\mathcal{D}_{1/3}$. Consider the first iteration where a node v is added to K while some open ball $B(y, \frac{1}{3}(\text{diam}(G) - e(y)))$ in $\mathcal{D}_{1/3}$ contains both v and some node $u \in K$ added previously. Let x be the tight upper vertex-certificate for u that was added to U . By triangle inequality, we have $d(x, v) \leq d(x, u) + d(u, y) + d(y, v)$. The choice of x implies $d(x, u) = e(u) - e(x) \leq d(u, y) + e(y) - e(x)$. Combining the two inequalities, we obtain $d(x, v) \leq 2d(u, y) + d(y, v) + e(y) - e(x)$. As u and v are in $B(y, \frac{1}{3}(\text{diam}(G) - e(y)))$, we have $2d(u, y) + d(y, v) < \text{diam}(G) - e(y)$ and finally get $d(x, v) < \text{diam}(G) - e(x)$ which implies $e^U(v) < \text{diam}(G)$. However, it is required that v has maximal upper bound $e^U(v) = \max_{w \in V} e^U(w)$ when it is selected for being added to K , in contradiction with $\max_{w \in V} e^U(w) \geq \max_{w \in V} e(w) = \text{diam}(G)$. We conclude that K must be a packing for $\mathcal{D}_{1/3}$. Both sizes of K and U are thus bounded by $\pi_{1/3}$. As any diameter certificate is a covering for $\mathcal{D}_{[1]}$ and has size $\pi_{[1]}$ at least, this guarantees that the size of U is within a factor $\frac{\pi_{1/3}}{\pi_{[1]}}$ at most from optimum. ■

This analysis can be complemented when we start Algorithm 4 with $K := \{c\}$ and $U := \{c\}$ initially where c is a center of the graph computed with Algorithm 1. We reference this combination as Algorithm 1+4 in the sequel. A similar proof then allows to show that K is a packing for $\mathcal{D}_{1/3}^c(G) = \{B(u, \beta_u(\text{diam}(G) - e(u))) : u \in V\}$ where $\beta_u = 1/3$ for $u \neq c$ and $\beta_c = 1$. We obtain the following corollary from Theorems 11 and 12.

Corollary 1 *Given a graph G and a ranking r of its node set V , Algorithm 1+4 computes the diameter of G , a diametral node b and a diameter certificate U of size $\pi_{1/3}^c$ at most with $|U| +$*

$2|\text{Antipode}_r(V)|+1$ one-to-all distance queries at most in $O(m(\pi_{1/3}^c + |\text{Antipode}_r(V)|))$ time where c is a center of G returned by Algorithm 1 and $\pi_{1/3}^c = \pi(\mathcal{D}_{1/3}^c)$ is the maximum packing size for the collection $\mathcal{D}_{1/3}^c = \{B(u, \beta_u(\text{diam}(G) - e(u))) : u \in V\}$ of open balls with radii factors $\beta_u = \frac{1}{3}$ for $u \neq c$ and $\beta_c = 1$ (for $u = c$).

Proof. In addition to the proof of Theorem 12, we just have to consider the case when a node v would be added to K while having $v \in B(c, \text{diam}(G) - e(c))$. As $c \in U$, we then have $e^U(v) \leq d(c, v) + e(c) < \text{diam}(G)$. This would raise a contradiction as the choice of v relies on $e^U(v) = \max_{w \in V} e^U(w) \geq \max_{w \in V} e(w) \geq \text{diam}(G)$. ■

This explains efficiency of practical algorithms as we observe that coverings of small size often exist for $\mathcal{D}_{1/3}^c(G)$ in practical graphs (see Section 16). As mentioned before, a further optimization consists in selecting a tight upper vertex-certificate x for u with minimal eccentricity. Using a function f such that $f(v, \ell)$ returns ℓ when $D_u(v) + \ell \leq e(u)$ and returns ∞ otherwise, it can be obtained through our minimum eccentricity selection procedure by replacing Line 1 with $x := \text{argminES}(G, r, L, e_L, f)$. The algorithm is referenced as Algorithm 1+4' in the sequel. This optimization through the delegate certificate technique provides performances similar to previous practical algorithms (see Section 16) while providing the complexity guarantee of Corollary 1.

15 All eccentricities

We now present a novel algorithm for all eccentricities. It relies on minimum eccentricity selection and the characterization of the minimum tight upper certificates presented in Section 4. We propose to compute all eccentricities of a graph as follows (see Algorithm 5 for a formal description). We maintain both a lower certificate L and an upper certificate U . As long as some node has untight upper bound, we select a node u with untight upper bound and minimal eccentricity using our minimum eccentricity selection procedure which then additionally ensures $e_L(u) = e(u)$. (We use for that purpose a function returning ∞ when the eccentricity value equals the upper bound.) We claim that u is in U^\preceq (see Lemma 5 below). We thus add u to the upper certificate U and update upper bounds accordingly. When our minimum eccentricity selection procedure detects that all nodes have tight upper bounds, lower bounds must be tight also. The algorithm then terminates with the following guarantees.

Theorem 13 *Given a graph G and a ranking r of its node set V , Algorithm 5 computes all eccentricities, a tight lower certificate $L \subseteq \text{Antipode}_r(V)$ and the optimal tight upper certificate U^\preceq with $|U^\preceq| + 2|L|$ one-to-all distance queries in $O(m(|U^\preceq| + 2|L|))$ time.*

In practical graphs, we observe that the size of U^\preceq is much larger than that of the computed lower certificate L and the algorithm roughly costs $|U^\preceq|$ BFS traversals (see Section 16). This seems almost optimal for an algorithm performing a BFS traversal from each node of U^\preceq . In particular, we expect a speed-up factor of 2 at least compared to [49] which suggest to alternate its selection of BFS sources between nodes of minimum lower-bound and nodes of maximum upper-bound (in addition there is no guarantee that nodes with minimum lower-bound are in U^\preceq).

The correctness of Algorithm 5 mainly rely on the following lemma.

Lemma 5 *Consider an upper certificate U and the set S_U of nodes that do not have a tight upper vertex-certificate in U . Any node $v \in S_U$ with minimum eccentricity (having $e(v) = \min_{u \in S_U} e(u)$) is its unique tight upper vertex-certificate (i.e., $v \in U^\preceq$).*

```

Input: A graph  $G$  and a ranking  $r$  of  $V$ .
Output: All eccentricities, a tight lower certificate  $L$  of  $G$  and a tight upper certificate  $U$ 
of  $G$ .
 $L := \emptyset$  /* Lower certificate (tentative hitting set for  $\{\overline{B}(u, e(u)) : u \in V\}$ )
*/
Maintain  $e_L(v) = \max_{x \in L} d(v, x)$  (initially 0) for all  $v \in V$ .
 $U := \emptyset$  /* Upper certificate (maximal nodes for  $\preceq$ ) */
Maintain  $e^U(u) = \min_{x \in U} d(u, x) + e(x)$  (initially  $\infty$ ) for all  $v \in V$ .
Function ecc-untight( $v, \ell$ )
| If  $\ell < e^U(v)$  then return  $\ell$ 
| else return  $\infty$ 
While minES( $L, e_L, \text{ecc-untight}$ )  $< \infty$  do
|  $u := \text{argminES}(G, r, L, e_L, \text{ecc-untight})$ 
|  $D_u := \text{DistFrom}(G, u)$  /* Distances from  $u$ . */
|  $e(u) := \max_{v \in V} D_u(v)$  /* Eccentricity of  $u$ . */
|  $U := U \cup \{u\}$ 
| For  $v \in V$  do  $e^U(v) := \min(e^U(v), D_u(v) + e(u))$ 
Return  $e^U, L, U$ 

```

Algorithm 5: Computing all eccentricities and tight lower/upper certificates.

Proof. For the sake of contradiction, suppose that $v \in S_U$ has a tight upper vertex-certificate $x \neq v$. As $e(v) = d(v, x) + e(x)$, we have $e(x) < e(v)$ and x cannot be in S_U by the minimality of $e(v)$. It thus has a tight upper vertex-certificate $y \in U$. But the transitivity of being a tight upper vertex-certificate (Proposition 1) implies that y is also a tight upper certificate for v which is in contradiction with $v \in S_U$. ■

Proof.[of Theorem 13] We first prove $U \subseteq U^\preceq$. As in Lemma 5, let S_U denote the set of nodes that do not have a tight upper vertex-certificate in U ($S_U = \{v \in V \mid e(v) < e^U(v)\}$). Now consider the node u selected at some iteration of the while loop. We prove $u \in U^\preceq \setminus U$. The correctness of our minimum eccentricity selection (Proposition 5) implies that u has minimum eccentricity in S_U and is thus in U^\preceq by Lemma 5. (Note that $\text{ecc-untight}(v, \cdot)$ is non-decreasing for all $v \in V$ as $\text{ecc-untight}(v, \ell) = \ell$ for $\ell < e(v)$ and $\text{ecc-untight}(v, \ell) = \infty$ for $\ell \geq e(v)$.) Additionally, $u \in S_U$ implies that u has untight upper bound and is not in U until we add it at that iteration.

The termination of the algorithm is guaranteed by the fact that U grows at each iteration. The algorithm ends when minimum eccentricity selection returns a node u such that $\text{ecc-untight}(u, e(u)) = \infty$. Proposition 5 then ensures $\text{ecc-untight}(v, e_L(v)) = \infty$ for all v . That is $e_L(v) = e^U(v)$ for all v and both bounds must equal $e(v)$. This implies that L (resp. U) is then a tight lower (resp. upper) certificate of G . Moreover, $U \subseteq U^\preceq$ then implies $U = U^\preceq$ by Proposition 1. ■

16 Experiments

We test social networks (Epinions, Hollywood, Slashdot, Twitter, dblp), computer networks (Gnutella, Skitter), web graphs (BerkStan, IndoChina, NotreDame), road networks (CAL-

type	name	n	m/n	d	w	$\frac{\text{diam}}{\text{rad}}$	D	$\pi_{0.8}^c$	$\pi_{1/3}^c$	n_c/n	R	A_{ID}	F	L_{all}	U_{all}
comm	Gnutella	14149	3.60	•	◦	1.58	19	47	1912	0.27	4	10	23	7	10039
comm	skitter	1694616	13.09	◦	◦	1.94	3	5	7	0.99	3	6	6	3	157780
game	FrozenSea	753343	7.70	◦	•	1.80	15	35	191	0.91	7	384	388	381	94370
geom	buddha	543652	6.00	◦	•	1.87	27	63	385	0.95	14	897	897	897	233263
road	CAL-d	1890815	2.45	◦	•	1.89	3	17	90	0.99	3	11	11	11	190210
road	CAL-t	1890815	2.45	◦	•	1.83	7	17	105	0.98	5	13	13	13	2810
road	CAL-u	1890815	2.45	◦	◦	1.99	2	4	6	0.99	3	7	11	7	170127
road	FLA-t	1070376	2.51	◦	•	1.99	2	4	4	0.99	2	2	2	2	934
road	europa-t	18010173	2.34	•	•	1.99	2	4	5	1	2	-	-	2	535138
soc	Epinions	32223	13.76	•	◦	2	2	4	7	0.99	2	7	20	3	1765
soc	Hollywood	1069126	106.33	◦	◦	1.71	29	603	4183	0.99	3	34	335	10	120626
soc	Slashdot	71307	12.80	•	◦	1.86	4	40	65	0.99	4	12	135	9	64336
soc	Twitter	68413	24.63	•	◦	2.50	2	6	8	0.99	4	31	4753	9	11166
soc	dblp	226413	6.33	◦	◦	2	1	11	20	1	5	6	43	6	13113
synth	bowtie500	505002	2.00	◦	◦	1.99	3	2001	4001	0.99	5	5	1507	5	9
synth	grid1500-wd	296680	1.66	•	•	2.38	3	12	69	0.04	4	5	6	5	295
synth	grid500-10	250976	3.59	◦	◦	2	1	5	5	1	3	4	4	4	165
synth	pwlaw2.5	1000000	3.85	◦	◦	1.90	4	24	24	0.99	2	19	47	6	90348
synth	udg10	999888	9.99	◦	◦	1.99	5	17	86	0.99	4	5	10	4	95731
vlsi	alue7065	34046	3.22	◦	◦	2	1	5	5	1	3	4	4	4	296
web	BerkStan	334857	13.51	•	◦	2.73	2	7	28	2e-03	3	16	17	16	20
web	Indochina	3806327	25.96	•	◦	6.91	2	4	6	0.99	3	-	-	5	376761
web	NotreDame	53968	5.65	•	◦	2.11	2	5	22	0.01	2	2	2	2	45

Table 1: Diameter and radius certificate sizes (D, R) and sizes of all eccentricity certificates (L_{all}, U_{all}) for various graphs, and related parameters.

t, CAL-d, CAL-u, FLA-t, europa-t), a 3D triangular mesh (buddha), and grid like graphs from VLSI applications (alue7065) and from computer games (FrozenSea). The data is available from snap.stanford.edu, webgraph.di.unimi.it, www.dis.uniroma1.it/challenge9, graphics.stanford.edu, steinlib.zib.de and movingai.com. We also test synthetic inputs: bowtie500 is the graph $BT_{500,500}$ represented in Figure 1, grid500-10 is a 501×501 square grid with random deletion of 10% of the edges, grid1500-wd is a weighted directed graph obtained from a 1501×1501 square grid where each edge is oriented randomly (with probability 1/2 for each direction) and assigned a random weight uniformly in $\{0, 1, \dots, 9\}$, pwlaw2.5 is a random graph generated according to the configuration model with a degree sequence following a power law with exponent 2.5, udg10 is a random unit disk graph where field size is parameterized to obtain average degree 10 roughly. Each graph is restricted to its largest (strongly) connected component.

Table 1 summarizes our main practical observations. For each instance G , we show its type, the number n of nodes in the largest (strongly) connected component, the average out-degree m/n , whether it is directed (d) and weighted (w), and the diameter to radius ratio $\frac{\text{diam}(G)}{\text{rad}(G)}$. We then show the size D of the diameter certificate computed by Algorithm 1+4', bounds on maximum packing sizes $\pi_{0.8}^c$ and $\pi_{1/3}^c$ (defined in Section 14), the proportion n_c/n of nodes in the (in-)ball of radius $\text{diam}(G) - \text{rad}(G)$ centered at a center c , the size R of the radius certificate computed by Algorithm 1, the number A_{ID} of antipodes for ID ranking and the number F of furthest nodes. The two latter numbers were obtained by performing a traversal per node of the graph (in quadratic time). A dash indicates a value that could not be obtained in less than few days of computation.

The first observation is that diameter and radius certificates are extremely small for all instances

(less than 30 nodes for all of them). Several observations allow to explain this phenomenon. First, all graphs have high diameter to radius ratio (over 1.5 for all of them). Note that this ratio is at most 2 for an undirected graph (it is unbounded in general directed graphs). Undirected graphs with ratio 2 have a one node diameter certificate: a center. This concerns two practical graphs while several ones have ratio very close to 2. Coherently, the concentration of nodes around the center is also high with respect to the diameter minus radius difference. This is measured by the ratio n_c/n where c is a center computed by our radius algorithm ($e(c) = \text{rad}(G)$) and n_c denotes the number of nodes in $B[c, \text{diam}(G) - \text{rad}(G)]$ (in directed graphs we count the number of nodes u such that $d(u, c) \leq \text{diam}(G) - \text{rad}(G)$). It counts the proportion of nodes u such that $e^{\{c\}}(u) \leq \text{diam}(G)$ which appears to be very close to 1 for most of the graphs. Notable exceptions are Gnutella, BerkStan and NotreDame. This may be explained by the low (compared to others) diameter to radius ratio (1.58) of the first one and probably to the highly asymmetric nature of the two others. Seeing diameter certification as a covering problem with balls $B[x, \text{diam}(G) - e(x)]$, there are thus few nodes that are not covered by a center c . Additionally, other nodes can be covered using few balls with reduced radii: the columns $\pi_{0.8}^c$ and $\pi_{1/3}^c$ indicate the size of coverings we could find using balls with radii reduced by a factor .8 and 1/3 respectively. These numbers upper bound maximum packing sizes $\pi_{0.8}^c$ and $\pi_{1/3}^c$ of the associated collections of balls with reduced radii. Our theoretical upper-bound of $\pi_{1/3}^c$ thus explains fast diameter computation for most of the graphs. A notable exception is $BT_{500,500}$ alias bowtie500 which was tailored for making former diameter algorithms slow (including Algorithm 1+4) and thus have large $\pi_{1/3}^c$ value. (Note that Algorithm 1+4' performs roughly $2D+2R$ distances queries and at most $2D+2A_{ID}$ distance queries, that is 16 for bowtie500.) Other exceptions are Hollywood and Gnutella for which the diameter to radius ratios are not so high either (compared to other graphs). However the parameter is still much smaller than the number of nodes.

Concerning radius computation, we observe that most graphs have very few antipodes as indicated by the A_{ID} column although the number F of furthest nodes can be significantly larger as observed among several social networks. A notable exception is the buddha graph which is a triangulated 3D surface and thus has more or less a sphere like topology (the arms form handles) that may explain why all furthest nodes are antipodes. However the number of antipodes remains much smaller than the number of nodes. FrozenSea also has a relatively large number of furthest nodes that are almost all antipodes. This might come from the design of the graph as a map where players of a video game evolve and should find dead ends.

Part IV

Graph classes with specific certificates

17 Power law random graphs

A β -power-law random graph can be defined according to the configuration model [8] starting from a degree distribution following a power law with exponent β (self-loops and multiple edges are allowed). We now translate some results of [10] in terms of certificates.

Proposition 6 ([10]) *Given $\varepsilon > 0$ and $\beta > 1$, any β -power-law random graph G has asymptotically almost surely:*

- a tight lower certificate (and thus a radius certificate) of size $n^{O(\varepsilon)}$ when $\beta > 2$,

- a tight lower certificate (and thus a radius certificate) of size $O\left(n^{1-\frac{2-\beta}{\beta-1}\left(\lfloor\frac{\beta-1}{2-\beta}-\frac{3}{2}\rfloor-\frac{1}{2}\right)}\right)$ when $1 < \beta < 2$;
- a diameter certificate of size $n^{O(\varepsilon)}$ when $1 < \beta < 3$,
- a diameter certificate of size $n^{\frac{1}{1+\Omega(\frac{\beta-1}{\beta-3})}+O(\varepsilon)}$ when $\beta > 3$.

We now detail the analyses of [10] from which these bounds are derived. The following variant of SumSweep heuristic [9] is proposed: given a graph $G = (V, E)$, select independently and uniformly at random $s_1, \dots, s_k \in V$, then iteratively choose t_1, \dots, t_k such that each t_j is selected in $V \setminus \{t_1, \dots, t_{j-1}\}$ so as to maximize $\sum_{i \leq j} d(s_i, t_j)$. The algorithm performs a BFS from each of these nodes and maintains corresponding lower bounds for all vertices. The analysis of this variant (see Section 9 of [10]) shows that for any $\varepsilon > 0$ and for sufficiently large graph G , all lower bounds are tight for $k = n^{3\varepsilon} + n^{O(\varepsilon)}$ when $\beta > 2$, and for $k = n^{3\varepsilon} + n^{f(\beta)}$ for $1 < \beta < 2$ where $f(\beta) = 1 - \frac{2-\beta}{\beta-1} \left(\lfloor \frac{\beta-1}{2-\beta} - \frac{3}{2} \rfloor - \frac{1}{2} \right)$. We note that we have $f(\beta) < 1$ for $\beta > 1.72$. In our terminology, $L = \{s_1, \dots, s_k, t_1, \dots, t_{j-1}\}$ is a tight lower certificate and the analysis shows that for any $\varepsilon > 0$ a β -power-law random graph has asymptotically almost surely (a.a.s. for short) a tight lower certificate of size $n^{O(\varepsilon)}$ when $\beta > 2$. When $1.72 < \beta < 2$, it has a.a.s. a tight lower certificate of sublinear size $O(n^{f(\beta)})$.

The analysis of the Exact SumSweep algorithm [9] then shows that after finding a center c , the number of nodes with upper bound greater than the diameter is proven to be $n^{O(\varepsilon)}$ a.a.s. for $1 < \beta < 3$ by bounding the number of nodes that are not in $B[c, \text{diam}(G) - e(c)]$ (see Section 11.1 of [10]). In our terminology, $U = \{c\} \cup (V \setminus B[c, \text{diam}(G) - e(c)])$ is a diameter certificate of size $n^{O(\varepsilon)}$. For $\beta > 3$, the analysis is more involved. It proves the existence of a diameter certificate of size $n^{g(\beta)+O(\varepsilon)}$ with $g(\beta) = \frac{1}{1+\Omega(\frac{\beta-1}{\beta-3})}$ (see Section 11.2 of [10]). Note that $n^{g(\beta)}$ is always sublinear.

Concerning the lower certificate bound, the tools provided in [10] allow to more generally bound the overall number of furthest nodes. We show that the set $F = \{t \in V : \exists s \in V, d(s, t) = e(s)\}$ of furthest nodes has size $n^{O(\varepsilon)}$. We rely on the following results of [10] where $\tau_s(p) = \min\{\ell \in \mathbb{N} : |B[s, \ell] \setminus B[s, \ell - 1]| > p\}$ denotes the minimum distance ℓ such that s has more than p nodes at distance exactly ℓ and $T(1 \rightarrow p)$ is the average of $\tau_s(p)$ over vertices s of degree one: more precisely, it satisfies $T(1 \rightarrow p) = 2$ for $1 < \beta < 2$, $T(1 \rightarrow p) = (1 + o(1)) \log_{1/(\beta-2)} \log p$ for $1 < \beta < 2$, and $T(1 \rightarrow p) = O(\log p)$ for $3 < \beta$.

Lemma 6 ([10]) *Given $\varepsilon > 0$ and $2 < \beta < 3$, there exists a positive constant $c < 1$ such that any β -power-law random graph G satisfies a.a.s.:*

- (i) *for any node s and t at furthest distance from s , $\tau_t(n^{1/2}) \geq \lceil (1 - O(\varepsilon))(T(1 \rightarrow n^{1/2}) + \frac{\log n}{-\log c} - \frac{3}{2}) \rceil$ (Lemma 7.1 in [10] with $x = 1/2$);*
- (ii) *the number of vertices satisfying $\tau_s(n^{1/2}) \geq (1 + \varepsilon)(T(1 \rightarrow n^{1/2}) + \alpha)$ is $O(nc^{\alpha-1/2})$ (Property 2.1 in [10] with $x = 1/2$).*

We can now state the following.

Proposition 7 *Given $\varepsilon > 0$ and $\beta > 1$, any β -power-law random graph G has a.a.s. $n^{O(\varepsilon)}$ furthest nodes.*

Note that the set of furthest nodes is obviously a tight lower certificate.

Proof. Consider a furthest node $t \in F$. According to Lemma 6(i), we have $\tau_t(n^{1/2}) \geq (1 - a\varepsilon)(T(1 \rightarrow n^{1/2}) + \frac{\log n}{-\log c} - \frac{3}{2})$ for some constant $a > 0$. Set α to obtain $(1 - a\varepsilon)(T(1 \rightarrow n^{1/2}) + \frac{\log n}{-\log c} - \frac{3}{2}) = (1 + \varepsilon)(T(1 \rightarrow n^{1/2}) + \alpha)$, that is $\alpha = \frac{\log n}{-\log c} - \frac{3}{2} - O(\varepsilon)(T(1 \rightarrow n^{1/2}) + \frac{\log n}{-\log c} - \frac{3}{2})$ (we assume $\varepsilon < 1/2$ without loss of generality). In all three regimes for β , we have $\alpha - \frac{1}{2} = \frac{\log n}{-\log c} - 2 - O(\varepsilon \log n)$. Now Lemma 6(ii) bounds the number of such t , yielding: $|F| = O(nc^{\alpha-1/2}) = c^{-2-O(\varepsilon \log n)} = n^{O(\varepsilon)}$. \blacksquare

Note that the results of [10] also apply to Rank-1 Inhomogeneous Random Graph models (see Appendix A in [10]).

18 Graphs with low doubling dimension

A graph is γ -doubling if every ball of positive radius is included in the union of at most γ balls with half radius.

18.1 Exact diameter computation with few antipodes

In the case of γ -doubling graphs, the following theorem shows that we can obtain a diameter certificate with almost linear size compared to the maximum packing size $\pi(\mathcal{D}_\alpha)$ for the collection \mathcal{D}_α (see Section 14). This complements Theorem 12 in the range $\frac{1}{3} \leq \alpha < 1$.

Theorem 14 *Given a γ -doubling graph G and $\alpha < 1$, the diameter $\text{diam}(G)$, a diametral node p and a diameter certificate U satisfying $|U| \leq \pi_\alpha \gamma^{O(1)+\log \frac{1}{1-\alpha}} \log \text{diam}(G)$ can be computed with $2|\text{Antipode}(V)| + |U|$ one-to-all distance queries, where π_α is the maximum size of a packing for the collection $\mathcal{D}_\alpha = \{B(u, \alpha(\text{diam}(G) - e(u))) : u \in V\}$.*

Note that this implies that minimum diameter certificate can be approximated within a factor $\gamma^{O(1)+\log(\text{diam}(G)-\text{rad}(G))} \frac{\pi_1}{\pi_{[1]}}$ when G is γ -doubling as $\mathcal{D}_\alpha = \mathcal{D}_1$ for $\alpha > 1 - \frac{1}{r+1}$ where $r = \text{diam}(G) - \text{rad}(G)$ is the maximum radius of a ball in \mathcal{D}_1 (recall that $\pi_{[1]}$ lower bounds the size of a minimum diameter certificate).

The above theorem is a consequence of Algorithm 6 which follows a primal-dual approach by constructing both a packing K for \mathcal{D}_α together with a covering U with $\mathcal{D}_{[1]}$ such that $|U| \leq |K| \gamma^{O(1)+\log \frac{1}{1-\alpha}} \log \text{diam}(G)$. Given a node u , let $S_u = \{v \in V \mid \exists B \in \mathcal{D}_\alpha \text{ s.t. } u, v \in B\}$ denote the set of nodes v that cannot be in a packing for \mathcal{D}_α containing u . The idea is to iteratively add a node u to K with highest eccentricity according to e^U and then to add sufficiently many nodes to U so that any node v in S_u gets an eccentricity upper bound $e^U(v)$ equal to $\text{diam}(G)$ or less. This will guarantee that no such node is added later to K and that K is a packing for \mathcal{D}_α .

Our selection rule for adding nodes to U is based on comparing their eccentricity to their distance to u . The rough idea is to add a node v as certificate in U when $e^U(v) > \text{diam}(G)$ and $\text{diam}(G) - e(v) = \Omega(d(u, v))$. Note that any node $w \in B[v, \text{diam}(G) - e(v)]$ then satisfies $e^U(w) \leq \text{diam}(G)$ and the doubling property will allow us to bound the number of nodes added to U . As the eccentricity of v is not known precisely until we perform a one-to-all distance query from v , we use our minimum eccentricity selection technique using a lower certificate L . As $\text{diam}(G)$

```

Input: A graph  $G$  and a parameter  $\alpha$  with  $0 < \alpha < 1$ .
Output: The diameter  $\text{diam}(G)$  of  $G$  and a diameter certificate  $p, U$ .
 $K := \emptyset$  /* Packing for  $\{B(u, \alpha(\text{diam}(G) - e(u))) : u \in V\}$ . */
 $U := \emptyset$  /* Upper certificate. */
Maintain  $e^U(u) = \min_{x \in U} d(u, x) + e(x)$  (initially  $\infty$ ) for all  $v \in V$ .
 $L := \emptyset$  /* Lower certificate. */
Maintain  $e_L(v) = \max_{x \in L} d(v, x)$  (initially 0) for all  $v \in V$ .
While  $\max_{p \in K} e(p) < \max_{u \in V} e^U(u)$  do
    Select  $u$  such that  $e^U(u)$  is maximal.
     $D_u := \text{DistFrom}(u)$ 
     $e(u) := \max_{w \in V} D_u(w)$  /* Eccentricity of  $u$ . */
     $K := K \cup \{u\}$ 
     $U := U \cup \{u\}$ 
    For  $w \in V$  do  $e^U(w) := \min(e^U(w), D_u(w) + e(u))$ 
    Function  $\text{ecc-slack-far}(v, \ell)$ 
        If  $e^U(v) - \ell > \frac{1-\alpha}{2\alpha} D_u(v)$  then return  $-D_u(v)$ 
        else return  $\infty$ 
    While  $\text{minES}(G, r, L, e_L, \text{ecc-slack-far}) < \infty$  do
         $v := \text{argminES}(G, r, L, e_L, \text{ecc-slack-far})$ 
         $D_v := \text{DistFrom}(v)$ 
         $e(v) := \max_{w \in V} D_v(w)$  /* Eccentricity of  $v$ . */
         $U := U \cup \{v\}$ 
        For  $w \in V$  do  $e^U(w) := \min(e^U(w), D_v(w) + e(v))$ 
     $p := \text{argmax}_{p \in K} e(p)$ 
Return  $e(p)$  and  $p, U$ .

```

Algorithm 6: Computing diameter and a diameter certificate assuming doubling property.

is not known either, we select v such that $e^U(v) - e(v) = \Omega(d(u, v))$. This ensures that a ball of radius $\Omega(d(u, v))$ will then be covered. We thus prefer v such that $d(u, v)$ is additionally maximal.

Proof.[of Theorem 14] The main arguments of the proof are the following.

The function $\ell \mapsto \text{ecc-slack-far}(v, \ell)$ is non-decreasing as it returns $-d(u, v)$ for $\ell \leq e^U(v) - \frac{1-\alpha}{2\alpha}d(u, v)$ and ∞ otherwise. Note that the minimum eccentricity selection for ecc-slack-far thus returns a node v such that $e^U(v) - e(v) \geq \frac{1-\alpha}{2\alpha}d(u, v)$ and $d(u, v)$ is maximal.

We first prove that the inner loop performs at most $\gamma^{O(1)+\log \frac{1}{1-\alpha}} \log \text{diam}(G)$ iterations. This bounds the number of nodes added to U when one node is added to K and will thus ensure $|U| \leq |K| \gamma^{O(1)+\log \frac{1}{1-\alpha}} \log \text{diam}(G)$. Consider an iteration of the main loop where u is added to K . Just after adding v to U in the inner loop, consider w s.t. $d(v, w) \leq \frac{\rho}{2+\rho}d(u, v)$ where $\rho = \frac{1-\alpha}{2\alpha}$. We then have $e^U(w) \leq d(v, w) + e(v)$ as $v \in U$ and $e(w) \geq e(v) - d(v, w)$ by triangle inequality. This gives $e^U(w) - e(w) \leq \frac{2\rho}{2+\rho}d(u, v)$. As $d(u, w) \geq d(u, v) - d(v, w) \geq \frac{2}{2+\rho}d(u, v)$, we get $e^U(w) - e(w) \leq \rho d(u, w)$ and nodes in $B[v, \frac{\rho}{2+\rho}d(u, v)]$ do not satisfy the condition of the inner loop. The doubling property implies that the number of iterations where we select v such that $d(u, v) > \frac{e(u)}{2}$ is at most $\gamma^{O(1)+\log \frac{1}{1-\alpha}}$. Then we may select v such that $d(u, v) > \frac{e(u)}{4}$ during the same number of iterations at most, and so on until we eventually select u itself (the only node v such $d(u, v) = 0$). The overall number of iterations of the inner loop is thus bounded by

$\gamma^{O(1)+\log \frac{1}{1-\alpha}} \log e(u)$.

For the sake of contradiction, suppose that K is not a packing and consider $u, u' \in K$ and $x \in V$ such that both u and u' are in $B(x, \alpha(\text{diam}(G) - e(x)))$. Assume without loss of generality that u was added to K before u' . After the inner loop for u , we have $e^U(x) - e(x) \leq \frac{1-\alpha}{2\alpha}d(u, x)$. Therefore there exists $y \in U$ such that $d(x, y) + e(y) \leq e(x) + \frac{1-\alpha}{2\alpha}d(u, x)$. We thus have $e^U(u') \leq d(u', y) + e(y) \leq d(u', x) + d(x, y) + e(y) \leq d(u', x) + e(x) + \frac{1-\alpha}{2\alpha}d(u, x)$. As $u, u' \in B(x, \alpha(\text{diam}(G) - e(x)))$, we get $e^U(u') < \frac{1+\alpha}{2}\text{diam}(G) + \frac{1-\alpha}{2}e(x)$. As $e(x) \leq \text{diam}(G)$, we obtain $e^U(u') < \text{diam}(G)$. This is a contradiction since the choice of u' implies $e^U(u') = \max_{v \in V} e^U(v) \geq \text{diam}(G)$. \blacksquare

18.2 Approximating radius and diameter

Interestingly, the following lemma links the gap between the bound provided by a lower/upper certificate for a node u and the distance from the certificate to a tight lower/upper certificate for u . Recall that a tight upper certificate for u is a node x such that $e(u) = d(u, x) + e(x)$. We similarly define a tight lower certificate for u as a node x such that $e(u) = d(u, x)$ (equivalently, x is a furthest node from u).

Lemma 7 *Given a lower certificate L (resp. an upper certificate U) and a node u , we have $e(u) - e_L(u) \leq d(x, L)$ (resp. $e^U(u) - e(u) \leq 2d(x, U)$) for any tight lower (resp. upper) certificate x for u .*

Proof. Consider a tight lower certificate x for a node u ($e(u) = d(u, x)$). Let $y \in L$ be a closest node to x in L ($d(x, y) = d(x, L)$). By triangle inequality, we have $e(u) = d(u, x) \leq d(u, y) + d(y, x) \leq e_L(u) + d(x, L)$.

Similarly, consider a tight upper certificate x for a node u ($e(u) = d(u, x) + e(x)$). Let $y \in U$ be a closest node to x in U ($d(x, y) = d(x, U)$). By triangle inequality, we have $e(u) = d(u, x) + e(x) \geq d(u, y) - d(x, y) + e(y) - d(x, y) \geq e^U(u) - 2d(x, U)$. \blacksquare

Now consider the choice of a node u with minimal eccentricity lower bound in Algorithm 1. This choice implies $e_L(u) \leq \text{rad}(G)$ and Lemma 7 then implies $e(u) \leq \text{rad}(G) + d(a, L)$ where a is the antipode of u which is added to L (if u is not a center). As long as the selected node has eccentricity greater than $(1 + \varepsilon)\text{rad}(G)$, the nodes in L are $\varepsilon\text{rad}(G)$ far apart and form a packing for the collection of balls of radius $\varepsilon\text{rad}(G)/2$. Similarly, the choice of a node u with maximal eccentricity upper bound in Algorithm 4 implies $e^U(u) \geq \text{diam}(G)$ and Lemma 7 then implies $e(u) \geq \text{diam}(G) - 2d(x, U)$ where x is the tight upper certificate chosen for u that is added to U . As long as the selected node has eccentricity less than $(1 - \varepsilon)\text{diam}(G)$, the nodes in U form a packing for the collection of balls of radius $\frac{\varepsilon}{2}\text{diam}(G)/2$. As the doubling property implies that such packings have size bounded by $\gamma^{\lceil \log \frac{2}{\varepsilon} \rceil + 1}$, we obtain the following approximation results for radius and diameter.

Proposition 8 *Given a γ -doubling graph G and $\varepsilon > 0$, Algorithm 1 (resp. Algorithm 4) provides a node u with eccentricity $(1 + \varepsilon)\text{rad}(G)$ at most (resp. $(1 - \varepsilon)\text{diam}(G)$ at least) with $O(\gamma^{\lceil \log \frac{2}{\varepsilon} \rceil + 1})$ traversals.*

Proof. As discussed above, the set L is a packing for balls of radius $\varepsilon\text{rad}(G)$ until a node whose eccentricity approximates the radius is found. We use the fact that packing size is bounded by

covering size. By the γ -doubling property, the whole graph can be covered by γ^i balls of radius $\frac{2\text{rad}(G)}{2^i}$ as any ball of radius $2\text{rad}(G)$ contains all nodes. For $i \geq \log \frac{2}{\varepsilon} + 1$ these balls have radius $\varepsilon \text{rad}(G)/2$ at most. Using that packing size is bounded by covering size, we can bound the number of iterations of Algorithm 1 where chosen nodes u have eccentricity greater than $(1 + \varepsilon)\text{rad}(G)$. If we stop the algorithm after $\gamma^{\lceil \log \frac{2}{\varepsilon} \rceil + 1}$ iterations, the node $u \in K$ with smallest eccentricity is guaranteed to have eccentricity $(1 + \varepsilon)\text{rad}(G)$ at most. The argument for diameter approximation is similar. \blacksquare

In particular, for any $\eta \in (0, 1)$, if G is γ -doubling, has at most n^η antipodes and $\text{diam}(G) < n^{\frac{\eta}{\log \gamma}}/8$, then Proposition 8 (applied for $\varepsilon = 8n^{-\frac{\eta}{\log \gamma}}$) implies that we can compute both the diameter and the radius in $O(n^\eta m)$ time, which is truly subquadratic.

19 Negatively curved graphs

In this section we analyze the behavior of our algorithms in the class of negatively curved graphs, alias, δ -hyperbolic graphs.

Let (X, d) be a metric space and $w \in X$. The *Gromov product* of $y, z \in X$ with respect to w is defined to be

$$(y|z)_w = \frac{1}{2}(d(y, w) + d(z, w) - d(y, z)).$$

Let $\delta \geq 0$. A metric space (X, d) is said to be δ -hyperbolic [37] if

$$(x|y)_w \geq \min\{(x|z)_w, (y|z)_w\} - \delta$$

for all $w, x, y, z \in X$. Equivalently, (X, d) is δ -hyperbolic if for any four points u, v, x, y of X , the two larger of the three distance sums $d(u, v) + d(x, y)$, $d(u, x) + d(v, y)$, $d(u, y) + d(v, x)$ differ by at most $2\delta \geq 0$. A graph $G = (V, E)$ is δ -hyperbolic if the associated shortest-path metric space (V, d) is δ -hyperbolic.

The hyperbolicity can be viewed as a local measure of how close a graph is metrically to a tree: the smaller the hyperbolicity is, the closer its metric is to a tree metric (trees are 0-hyperbolic). Recent empirical studies showed that many real-world graphs (including Internet application networks, web networks, collaboration networks, social networks, biological networks, and others) have small hyperbolicity [3, 41, 24]. It is known [15, 16] that if G is a δ -hyperbolic graph and $\{y, z\}$ is a pair returned after two BFS scans, then $d(y, z) \geq \text{diam}(G) - 2\delta$, $\text{diam}(G) \geq 2\text{rad}(G) - 4\delta - 1$, $\text{diam}(C(G)) \leq 4\delta + 1$, and $C(G)$ is contained in a small ball centered at a middle vertex of any shortest (y, z) -path. Recall that $C(G)$ denotes the set of centers of G (see Section 2). Consequently, there exist linear-time algorithms for the diameter and radius problems with additive errors linearly depending on the input graph's hyperbolicity.

In what follows, we will analyze the behavior of our algorithms in the class of δ -hyperbolic graphs. From the definition of a δ -hyperbolic graph, we immediately get the following simple but very useful auxiliary lemma.

Lemma 8 *Let $G = (V, E)$ be a δ -hyperbolic graph. For every vertices $c, v, x, y \in V$, $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ or $d(y, v) - d(x, y) \geq d(c, v) - d(x, c) - 2\delta$ holds.*

Proof. Assume, without loss of generality, that $d(x, c) + d(v, y) \leq d(y, c) + d(x, v)$. If also $d(c, v) + d(x, y) \geq d(y, c) + d(x, v)$ then, by δ -hyperbolicity of G , $d(c, v) + d(x, y) - d(y, c) - d(x, v) \leq 2\delta$, i.e., $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$. If $d(c, v) + d(x, y) \leq d(y, c) + d(x, v)$, then

$$d(x, v) - d(x, y) \geq d(c, v) - d(y, c) \geq d(c, v) - d(y, c) - 2\delta. \quad \blacksquare$$

Using this simple lemma, we get as easy corollaries two useful results from [15]. Denote by $F(s) := \{v \in V : d(s, v) = e(s)\}$ the set of vertices furthest from s .

Corollary 2 *For every δ -hyperbolic graph G , $\text{diam}(G) \geq 2 \text{rad}(G) - 4\delta - 1$.*

Proof. Let x, y be vertices of G such that $d(x, y) = \text{diam}(G)$. Let c be a middle vertex of any shortest path connecting x with y . Applying Lemma 8 to c, v, x, y , where $v \in F(c)$, assume without loss of generality that $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ holds. Then, since $d(x, y) \geq d(x, v)$, $d(y, c) \geq d(c, v) - 2\delta$. Hence, $d(x, y) = d(x, c) + d(c, y) \geq 2d(c, y) - 1 \geq 2d(c, v) - 4\delta - 1 = 2e(c) - 4\delta - 1 \geq 2 \text{rad}(G) - 4\delta - 1$. \blacksquare

Corollary 3 *Let $G = (V, E)$ be a δ -hyperbolic graph. For every vertices $c, v \in V$ such that $v \in F(c)$, $e(v) \geq \text{diam}(G) - 2\delta \geq 2 \text{rad}(G) - 6\delta - 1$.*

Proof. Apply Lemma 8 to c, v and vertices x, y such that $d(x, y) = \text{diam}(G)$. Without loss of generality, assume that $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ holds. Then, since $d(c, v) \geq d(c, y)$, $e(v) \geq d(x, v) \geq d(x, y) + d(c, v) - d(y, c) - 2\delta \geq d(x, y) - 2\delta = \text{diam}(G) - 2\delta$. \blacksquare

We are ready to analyze Algorithm 1. Let u_i and $a_i \in F(u_i)$ be the vertices picked in iteration i of the do-while loop. Let $K_i := \{u_1, u_2, \dots, u_i\}$ and $L_i := \{a_1, a_2, \dots, a_i\}$. According to the algorithm, u_1 is picked arbitrarily (as initially $L = \emptyset$), a_1 is a vertex furthest from u_1 , $u_2 = a_1$ (as $L_1 = \{a_1\}$ is a singleton), a_2 is a vertex most distant from $u_2 = a_1$, u_3 is a middle vertex of a shortest (a_1, a_2) -path. By Corollary 3, we already have $d(a_1, a_2) \geq \text{diam}(G) - 2\delta$. We can also show that $e(u_3) \leq \text{rad}(G) + 3\delta$.

Proposition 9 *If G is a δ -hyperbolic graph, then $d(a_1, a_2) \geq \text{diam}(G) - 2\delta$ and $e(u_3) \leq \text{rad}(G) + 3\delta$.*

Proof. We only need to estimate the eccentricity of the vertex u_3 . As u_3 is a middle vertex of a shortest (a_1, a_2) -path, $\min\{d(u_3, a_1), d(u_3, a_2)\} \geq \lfloor \frac{d(a_1, a_2)}{2} \rfloor \geq \lfloor \frac{\text{diam}(G)}{2} \rfloor - \delta$. Now, without loss of generality, assume (see Lemma 8) that for vertices u_3, a_3, a_2, a_1 , $d(a_2, a_3) - d(a_2, a_1) \geq d(u_3, a_3) - d(a_1, u_3) - 2\delta$ holds. Then, $e(u_3) = d(u_3, a_3) \leq d(a_2, a_3) - d(a_2, a_1) + d(a_1, u_3) + 2\delta = d(a_2, a_3) - d(a_2, u_3) + 2\delta \leq \text{diam}(G) - \lfloor \frac{\text{diam}(G)}{2} \rfloor + \delta + 2\delta = \lceil \frac{\text{diam}(G)}{2} \rceil + 3\delta \leq \lceil \frac{2 \text{rad}(G)}{2} \rceil + 3\delta = \text{rad}(G) + 3\delta$. \blacksquare

Thus, in δ -hyperbolic graphs, a vertex with eccentricity at most $\text{rad}(G) + 3\delta$ and a pair of vertices that are at least $\text{diam}(G) - 2\delta$ apart from each other are found by Algorithm 1 in at most 3 iterations, i.e., in linear time. Note that a similar linear-time algorithm was already reported in [15]: in δ -hyperbolic graphs, a vertex with eccentricity at most $\text{rad}(G) + 5\delta$ and a pair of vertices that are at least $\text{diam}(G) - 2\delta$ apart from each other can be found in linear time.

Next, we show that a vertex with eccentricity at most $\text{rad}(G) + 2\delta$ is found by Algorithm 1 in at most $2\delta + 2$ iterations. Consider iteration $i \geq 3$ and let p', p'' be vertices of L_{i-1} with the largest distance, i.e., $d(p', p'') = \max\{d(x, y) : x, y \in L_{i-1}\} = \text{diam}(L_{i-1})$. If $e(u_i) > \text{rad}(G) + 2\delta$, then applying Lemma 8 to $u_i, a_i, p', p'' \in V$, we get $d(p', a_i) - d(p', p'') \geq d(u_i, a_i) - d(p'', u_i) - 2\delta$ or $d(p'', a_i) - d(p', p'') \geq d(u_i, a_i) - d(p', u_i) - 2\delta$. Hence, $\max\{d(p', a_i) - d(p', p''), d(p'', a_i) - d(p', p'')\} \geq$

$e(u_i) - \text{rad}(G) - 2\delta > 0$ (as $\max\{d(p'', u_i), d(p', u_i)\} \leq \min_{u \in V} e_{L_{i-1}}(u) \leq \text{rad}(G)$). That is, if $e(u_i) > \text{rad}(G) + 2\delta$ then $\text{diam}(L_i) > \text{diam}(L_{i-1})$. As $\text{diam}(L_2) = d(a_1, a_2) \geq \text{diam}(G) - 2\delta$, in at most $2\delta + 2$ iterations of the while-loop of Algorithm 1 we will get $\text{diam}(L_i) = \text{diam}(L_{i-1})$ (with $i \leq 2\delta + 2$) and hence $e(u_i) \leq \text{rad}(G) + 2\delta$ must hold. Thus, we proved the following proposition.

Proposition 10 *If G is a δ -hyperbolic graph, then there is an index $i \leq 2\delta + 2$ such that $e(u_i) \leq \text{rad}(G) + 2\delta$. Furthermore, $e(u_j) \leq \text{rad}(G) + 2\delta$ for all $j \geq 2\delta + 2$.*

The second part of Proposition 10 says that all u_i vertices generated by Algorithm 1 after $2\delta + 1$ iterations have eccentricity at most $\text{rad}(G) + 2\delta$. Hence, in δ -hyperbolic graphs where the set $C^{2\delta}(G) := \{c \in V : e(c) \leq \text{rad}(G) + 2\delta\}$ has cardinality bounded by some function $g(\delta)$, depending only on δ , our algorithm will produce a vertex with eccentricity $\text{rad}(G)$ (i.e., a central vertex) in at most $g(\delta) + 2\delta + 1$ iterations.

Next we show that the set $C^{2\delta}(G)$ of a δ -hyperbolic graph has bounded diameter. Before our work, it was known that $\text{diam}(C(G)) \leq 4\delta + 1$ and there exists a vertex $c \in V$ such that $d(v, c) \leq 5\delta + 1$ for every $v \in C(G)$ [15].

Proposition 11 *If G is a δ -hyperbolic graph, then for every $x, y \in C^{2\delta}(G)$, $d(x, y) \leq 8\delta + 1$. Furthermore, there is a vertex $c \in V$ such that $d(v, c) \leq 6\delta + 1$ for every $v \in C^{2\delta}(G)$.*

Proof. Let c be a middle vertex of any shortest path connecting x with y . Apply Lemma 8 to c, v, x, y , where $v \in F(c)$. Without loss of generality, assume that $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ holds. Then, $d(x, c) = d(x, y) - d(y, c) \leq d(x, v) - d(c, v) + 2\delta \leq e(x) - e(c) + 2\delta \leq \text{rad}(G) + 2\delta - \text{rad}(G) + 2\delta = 4\delta$. Hence, $d(x, y) = d(x, c) + d(y, c) \leq 2d(x, c) + 1 \leq 8\delta + 1$.

To prove the second assertion, consider a pair of vertices $x, y \in V$ with $d(x, y) = \text{diam}(G)$ and a middle vertex c of any shortest (x, y) -path. Apply Lemma 8 to c, v, x, y , where v is an arbitrary vertex from $C^{2\delta}(G)$. Without loss of generality, assume that $d(x, v) - d(x, y) \geq d(c, v) - d(y, c) - 2\delta$ holds. We know also that $d(x, c) \geq \lfloor \frac{d(x, y)}{2} \rfloor = \lfloor \frac{\text{diam}(G)}{2} \rfloor \geq \lfloor \frac{2\text{rad}(G) - 4\delta - 1}{2} \rfloor \geq \text{rad}(G) - 2\delta - 1$ (see Corollary 2). Hence, $d(c, v) \leq d(x, v) - d(x, y) + d(y, c) + 2\delta = d(x, v) - d(x, c) + 2\delta \leq e(v) - \text{rad}(G) + 2\delta + 2\delta \leq \text{rad}(G) + 2\delta + 1 - \text{rad}(G) + 4\delta = 6\delta + 1$. \blacksquare

If the vertex degrees of a δ -hyperbolic graph are bounded by a constant Δ then $C^{2\delta}(G)$ has at most $\Delta^{O(\delta)}$ vertices. Summarizing, we have the following result.

Theorem 15 *Let $G = (V, E)$ be a δ -hyperbolic graph with m edges. Algorithm 1 finds*

1. *a vertex with eccentricity at most $\text{rad}(G) + 3\delta$ in at most $O(m)$ time,*
2. *a vertex with eccentricity at most $\text{rad}(G) + 2\delta$ in at most $O(\delta m)$ time,*
3. *a central vertex and a $O(1)$ -size radius certificate in at most $O(m)$ time, if the vertex degrees and δ are bounded by constants.*

Another linear-time algorithm for finding a central vertex of a δ -hyperbolic graph with δ and vertex degrees bounded by constants was proposed in [15].

20 Chordal graphs

Recall that $F(s) = \{v \in V : d(v, s) = e(s)\}$ denotes the set of all vertices of G that are furthest from s and $C(G) = \{c \in V : e(c) = \text{rad}(G)\}$ denotes the set of all central vertices of G . The *metric interval* $I(u, v)$ between vertices u and v is defined by $I(u, v) = \{x \in V : d(u, x) + d(x, v) = d(u, v)\}$, i.e., it consists of all vertices of G that lie on shortest paths between u and v .

In this section we analyze the behavior of our algorithms in the class of chordal graphs. Recall that a graph G is *chordal* if every induced cycle of length at least 4 has a chord. Chordal graphs are interesting because a central vertex in them can be found in linear time [14] but finding the diameter in truly subquadratic time would refute the Orthogonal Vectors Conjecture [19, 46].

First we give an example of an n -vertex chordal graph G on which Algorithm 1 will need $n/2$ iterations although G has a certificate for the radius consisting of only two vertices in L . Set $n = 2k$ and consider two sets of vertices $X = \{x_1, \dots, x_k\}$ and $Y = \{y_1, \dots, y_k\}$. The vertex set of G is $X \cup Y$. Make X a clique and Y an independent set in G . Make every vertex x_i adjacent to all vertices y_j with $j \leq i$. Algorithm 1 may place vertices $x_1, y_2, x_2, y_3, \dots, x_k$ (in this order) into K and vertices $y_2, y_1, y_3, y_4, \dots, y_k$ (in this order) into L . The central vertex x_k will be determined only when all Y -vertices are in L . On the other hand, $\{y_1, y_k\}$ is a radius certificate of G .

Note that the graph G constructed has vertices of large degrees (up-to $n - 1$). As every chordal graph G has hyperbolicity at most 1, it follows from Theorem 15 that our algorithm finds a central vertex in linear time in every chordal graph with vertex degrees bounded by a constant. It should be noted that there is a linear-time algorithm that finds a central vertex of an arbitrary chordal graph [14]; it uses additional metric properties of chordal graphs.

To analyze possible radius and diameter certificates in the class of chordal graphs, we will need the following important lemma.

Lemma 9 ([13]) *Let G be a chordal graph. Let x, y, v, u be vertices of G such that $v \in I(x, y)$, $x \in I(v, u)$, and x and v are adjacent. Then $d(u, y) \geq d(u, x) + d(v, y)$. Furthermore, $d(u, y) = d(u, x) + d(v, y)$ if and only if there exist a neighbor x' of x in $I(x, u)$, a neighbor v' of v in $I(v, y)$ and a vertex w with $N(w) \supseteq \{x', x, v, v'\}$; in particular, x', v' and w lie on a common shortest path of G between u and y .*

Our analysis is based on the following propositions which are also of independent interest. Recall that $C^1(G) := \{v \in V : e(v) \leq \text{rad}(G) + 1\}$.

Proposition 12 *Let $G = (V, E)$ be a chordal graph.*

- (i) *If $\text{diam}(G) < 2 \text{rad}(G)$ then, for every vertices $s \in V$ and $t \in F(s)$, there is a vertex $w \in I(s, t) \cap C(G)$ such that $t \in F(w)$.*
- (ii) *If $\text{diam}(G) = 2 \text{rad}(G)$ then, for every vertices $s \in V$ and $t \in F(s)$, there is a vertex $w \in I(s, t) \cap C^1(G)$ such that $t \in F(w)$.*

Proof. First we show that for every vertex x of G with $e(x) = k > \text{rad}(G)$ there is a vertex y such that $e(y) = k - 1$ and $d(x, y) \leq 2$. This is true even in the case when $\text{diam}(G) = 2 \text{rad}(G)$. Consider a vertex y in G with $e(y) = k - 1$ that is closest to x . Let z be any neighbor of y in $I(y, x)$. Necessarily, $e(z) = k$. Consider a vertex $u \in F(z)$. Since $d(y, u) \leq e(y) = k - 1 = e(z) - 1 = d(z, u) - 1 \leq d(y, u)$, we have $y \in I(z, u)$. Applying Lemma 9 to $y \in I(z, u)$ and $z \in I(y, x)$, we get $d(x, u) \geq d(x, y) - 1 + d(y, u) = d(x, y) + k - 2$. As $d(x, u) \leq e(x) = k$, we conclude $d(x, y) \leq 2$.

Next we claim that if $\text{diam}(G) < 2\text{rad}(G)$ then for every vertex x of G with $e(x) = k > \text{rad}(G)$ there is in fact a vertex $z \in N(x)$ such that $e(z) = k - 1$. Furthermore, if $\text{diam}(G) = 2\text{rad}(G)$, such a neighbor z with $e(z) = k - 1$ exists for every vertex x of G with $e(x) = k > \text{rad}(G) + 1$. Assume, by way of contradiction, that no neighbor of x has eccentricity $k - 1$ and let y be an arbitrary vertex of G with $d(x, y) = 2$ and $e(y) = k - 1$. Let also z be a vertex from $N(x) \cap N(y)$ for which the set $S_x(z) = \{v \in F(x) : z \in I(x, v)\}$ is largest. Necessarily, $e(z) = k$. As before, consider a vertex $u \in F(z)$. By Lemma 9, applied to $y \in I(z, u)$ and $z \in I(y, x)$, we get $d(x, u) \geq d(y, u) + d(x, z) = k - 1 + 1 = k$. As $d(x, u) \leq e(x) = k$, we conclude $d(x, u) = k$. Hence, by the second part of Lemma 9, there must exist a vertex w adjacent to y, z, x and at distance $k - 1$ from u . As $u \in F(x)$, $w \in I(x, u)$, $z \notin I(x, u)$, by the maximality of $|S_x(z)|$, there must exist a vertex $u' \in F(x)$ with $w \notin I(x, u')$, $z \in I(x, u')$. We have $d(z, u') = k - 1$, $d(w, u') = k$, and hence $z \in I(w, u')$ and $w \in I(z, u')$. By Lemma 9, $d(u, u') \geq d(u, w) + d(z, u') = k - 1 + k - 1 = 2k - 2$. Hence, $\text{diam}(G) \geq d(u, u') > 2\text{rad}(G)$, if $k > \text{rad}(G) + 1$, and $\text{diam}(G) \geq d(u, u') \geq 2\text{rad}(G)$, if $k = \text{rad}(G) + 1$. These contradictions prove the claim.

Now we can conclude our proof. Consider arbitrary vertices $s \in V$ and $t \in F(s)$ and proceed by induction on $k = e(s)$. If $k = \text{rad}(G)$ then $w = s$ and we are done. If $k = \text{rad}(G) + 1$ and $\text{diam}(G) = 2\text{rad}(G)$ then again $w = s$ and we are done. If $k > \text{rad}(G) + 1$ or $k = \text{rad}(G) + 1$ and $\text{diam}(G) < 2\text{rad}(G)$ then a neighbor z of s with $e(z) = k - 1$ satisfies $t \in F(z)$, and we can apply the induction hypothesis. ■

A pair x, y of vertices is called a *diametral pair* of a graph G if $d(x, y) = \text{diam}(G)$.

Proposition 13 *The center $C(G)$ of a chordal graph G is a diameter certificate of G (not necessarily a smallest one).*

Proof. It follows from Proposition 4 and Proposition 12(i). ■

Proposition 14 *For every chordal graph G , the set $C^1(G)$ is a tight upper certificate of G (not necessarily a smallest one).*

Proof. The statement follows from Proposition 12 and the definition of a tight upper certificate. ■

So, it is interesting that if the center $C(G)$ is known for a chordal graph G then its diameter can be computed in $O(|C(G)|m)$ time. However, there is no way to bound the cardinality of the set $C(G)$ in an arbitrary chordal graph G . In fact $C(G)$ may contain $n - 2$ vertices in some chordal graphs. To construct such a graph G , take a complete graph K_{n-2} on $n - 2$ vertices. Add two new vertices u and v adjacent to all vertices of K_{n-2} but not to each other. It is easy to see that $C(G)$ contains exactly the vertices of K_{n-2} .

Nevertheless, it is known that for every chordal graph G and any two vertices x, y from $C(G)$, $d(x, y) \leq 3$ holds [18]. This suggest the following approach for computing the diameter of a chordal graph $G = (V, E)$.

- Use the linear-time algorithm from [14] to find a central vertex c of G .
- Set $C_3 := \{x \in V : d(x, c) \leq 3\}$. /* $C(G) \subseteq C_3$ */
- Find a vertex p such that $e^{C_3}(p)$ is maximum.
- Report $e(p)$ as the diameter value.

The complexity of this approach is $O(|C_3|m)$. As a consequence, we have that when the vertex degrees are bounded in a chordal graph by a constant Δ then its diameter can be computed in linear time (as $|C_3|$ is bounded by a constant Δ^3). We are not aware if such a result was known before. Note also that in general chordal graphs the cardinality of C_3 cannot be bounded by a constant since otherwise the diameter of an arbitrary chordal graph could be computed in linear time, refuting the Orthogonal Vectors Conjecture [19, 46].

From the proof of Proposition 12 it follows also that, for every vertex v with $e(v) = \text{rad}(G) + 1$, $d(v, C(G)) \leq 2$ holds. This suggests the following approach for computing the eccentricities of all vertices of a chordal graph $G = (V, E)$.

- Use the linear-time algorithm from [14] to find a central vertex c of G .
- Set $C_5 := \{x \in V : d(x, c) \leq 5\}$. /* $C^1(G) \subseteq C_5$ */
- For every vertex $v \in V$ report $e(v) = \min_{c \in C_5} d(v, c) + e(c)$.

The complexity of this approach is $O(|C_5|m)$. As a consequence, we have that when the vertex degrees are bounded in a chordal graph by a constant Δ then the eccentricities of all its vertices can be computed in linear time (as $|C_5|$ is bounded by a constant Δ^5). We are not aware if such a result was known before.

Summarizing, we have the following result.

Theorem 16 *Let $G = (V, E)$ be a chordal graph with m edges and whose vertex degrees are bounded by a constant. Then, eccentricities of all vertices of G can be computed in total $O(m)$ time.*

By Proposition 13, the center $C(G)$ of a chordal graph G is a diameter certificate of G . Next we will show that the set of all diametral vertices of a chordal graph G forms a radius certificate of G .

Let $D^k(G) := \{v \in V : e(v) \geq \text{diam}(G) - k\}$. It is known that for every vertex v of a chordal graph G there is a vertex $u \in F(v)$ with $e(u) \geq \text{diam}(G) - 1$ [30]. Hence, the set $D^1(G)$ contains the output set L of Algorithm 1 and, therefore, it gives already a radius certificate for a chordal graph G . In fact, we can prove a stronger result.

Proposition 15 *For every chordal graph G , the set $D(G) := D^0(G)$ is a radius certificate of G (not necessarily a smallest one).*

Proof. By Proposition 3 and the fact that in any chordal graph G , $\text{diam}(G) \geq 2\text{rad}(G) - 2$ [18], we need to consider only the case when $\text{diam}(G) = 2\text{rad}(G) - 2$.

Assume that there is a vertex u in G such that $d(u, t) \leq \text{rad}(G) - 1$ for every vertex $t \in D(G)$. Denote by S the set of all such vertices u . Denote by S' those vertices from S that have the minimum eccentricity. Finally, denote by S'' those vertices u from S' that have the smallest number of vertices in $F(u)$.

Consider a vertex $u \in S''$, a vertex $v \in F(u)$ and a neighbor w of u on a shortest path from u to v . Consider also an arbitrary vertex $x \in D(G)$ and an arbitrary vertex $y \in F(x) \subset D(G)$. Since $d(x, y) = 2\text{rad}(G) - 2$, we have $d(x, u) = d(y, u) = \text{rad}(G) - 1$ and hence $u \in I(x, y)$. We claim that $d(w, x) \leq \text{rad}(G) - 1$ as well. Indeed, suppose that $d(w, x) > \text{rad}(G) - 1$. Then, $u \in I(x, w)$ and $w \in I(u, v)$. By Lemma 9, $d(x, v) \geq d(x, u) + d(w, v) \geq \text{rad}(G) - 1 + e(u) - 1 \geq 2\text{rad}(G) - 2$, i.e., $d(x, v) = 2\text{rad}(G) - 2$ (hence v must belong to $D(G)$) and $d(u, v) = e(u) = \text{rad}(G)$. The latter contradicts with the choice of u (as $u \in S$). Thus, $d(w, x) \leq \text{rad}(G) - 1$ for every vertex $x \in D(G)$, i.e., $w \in S$.

As $u \in S'$, $e(u) \leq e(w)$. First assume that $e(u) = e(w)$, i.e., $w \in S'$. Since $v \in F(u)$ and $v \notin F(w)$ (note that $d(w, v) = d(u, v) - 1 = e(u) - 1 \leq e(w) - 1$), by the choice of u (as

$u \in S''$), there must exist a vertex $t \in V$ such that $t \in F(w)$ and $t \notin F(u)$. We necessarily have $d(t, u) = e(w) - 1 \geq \text{rad}(G) - 1$ and $d(v, w) = e(u) - 1 \geq \text{rad}(G) - 1$. One can apply now Lemma 9 to $u \in I(w, t)$ and $w \in I(u, v)$ and get $d(v, t) \geq d(v, w) + d(t, u) \geq 2\text{rad}(G) - 2 = \text{diam}(G)$. That is, both v and t must be in $D(G)$, contradicting again with the choice of u (as $u \in S$ and $d(u, v) = e(u) \geq \text{rad}(G)$).

Assume now that $e(u) < e(w)$ and consider an arbitrary vertex $t \in F(w)$. Necessarily, $u \in I(w, t)$. Hence again one can apply Lemma 9 to $u \in I(w, t)$ and $w \in I(u, v)$ and get $d(v, t) \geq d(v, w) + d(t, u) = e(u) - 1 + e(w) - 1 \geq 2\text{rad}(G) - 1 > \text{diam}(G)$, and a contradiction arises.

Contradictions obtained prove that for every vertex $u \in V$ there is a vertex $t \in D(G)$ such that $d(u, t) \geq \text{rad}(G)$, i.e., $D(G)$ is a radius certificate of G . \blacksquare

21 Graph analogues of hyperconvex metric spaces

In this section, we analyze radius and diameter certificates in Helly graphs and in bipartite Helly graphs. A graph $G = (V, E)$ is *Helly* if its family of balls $\mathcal{B}(G) := \{B[v, r] : v \in V, r \in \mathbb{N}_{\geq 0}\}$ satisfies the Helly property (i.e., every collection of pairwise intersecting balls has a nonempty common intersection). A bipartite graph $G = (X \cup Y, E)$ is *bipartite Helly* if its family of half-balls $\mathcal{B}_X(G) \cup \mathcal{B}_Y(G)$ satisfies the Helly property where $\mathcal{B}_X(G) := \{B[v, r] \cap X : v \in X \cup Y, r \in \mathbb{N}_{\geq 0}\}$ and $\mathcal{B}_Y(G) := \{B[v, r] \cap Y : v \in X \cup Y, r \in \mathbb{N}_{\geq 0}\}$. Helly graphs (and bipartite Helly graphs) are discrete analogues of *hyperconvex metric spaces* (sometimes called also *injective metric spaces*). It is well known that every (bipartite) graph G isometrically embeds into an unique smallest (bipartite) Helly graph $H(G)$ called the *injective hull* of G [31, 40]. The diameter of the injective hull $H(G)$ of a graph G is equal to the diameter of G (see, e.g., [29]). The latter may suggest that finding an efficient algorithm for the diameter of a (bipartite) Helly graph $H(G)$ is all what one needs to compute efficiently the diameter of an arbitrary graph G (given $H(G)$). Unfortunately, the injective hull $H(G)$ of a graph G may have exponentially more vertices than G has (see [38] for several restricted graph classes where injective hulls are exponentially large).

It was open until recently whether there are truly subquadratic-time algorithms for the radius and diameter problems on Helly graphs. Such algorithms were recently presented in [28, 35]. One can compute in $\mathcal{O}(m\sqrt{n})$ time for a given n -vertex m -edge Helly graph its radius, diameter and all vertex eccentricities. The algorithms make use of the Helly property and of the unimodality of the eccentricity function in Helly graphs [26, 27]: every vertex of locally minimum eccentricity is a central vertex.

We will need the following result establishing relationships between the diameter and the radius. Recall that, for an arbitrary graph G , $\text{rad}(G) \leq \text{diam}(G) \leq 2\text{rad}(G)$ holds.

Proposition 16 *For every Helly graph G , $\text{diam}(G) \geq 2\text{rad}(G) - 1$ [26, 27]. For every bipartite Helly graph G , $\text{diam}(G) \geq 2\text{rad}(G) - 2$.*

Proof. We give here a simple proof for both statements. The cases where $\text{rad}(G) = 1$ being obvious, we assume $\text{diam}(G) \geq \text{rad}(G) \geq 2$.

Let $G = (V, E)$ be a Helly graph and assume, by way of contradiction, $\text{diam}(G) \leq 2\text{rad}(G) - 2$. Consider in G a system of balls $\mathcal{F} = \{B[v, \text{rad}(G) - 1] : v \in V\}$. For every two vertices x, y of G , we have $d(x, y) \leq \text{diam}(G) \leq 2\text{rad}(G) - 2$. Hence, all balls of \mathcal{F} pairwise intersect. By the Helly property, there must exist a vertex c in G which is at distance at most $\text{rad}(G) - 1$ from every $v \in V$. Necessarily, $e(c) \leq \text{rad}(G) - 1$, contradicting the definition of the radius of G .

Now consider a bipartite Helly graph $G = (X \cup Y, E)$ and assume, by way of contradiction, $\text{diam}(G) \leq 2 \text{rad}(G) - 3$. First assume that $\text{rad}(G)$ is even. Consider in G the system of half-balls $\mathcal{F}_X(G) := \{B[v, \text{rad}(G) - 2] \cap X : v \in X\} \cup \{B[v, \text{rad}(G) - 1] \cap X : v \in Y\}$. We claim that all half-balls of $\mathcal{F}_X(G)$ pairwise intersect. Indeed, let $u, v \in X \cup Y$ be arbitrary. Let $\ell = \lfloor \frac{d(u, v)}{2} \rfloor$, and let a, b be consecutive vertices on a shortest (u, v) -path such that $d(u, a) = d(u, b) - 1 = \ell$. Since G is bipartite, and vertices a and b are adjacent, one of the vertices a or b is in X . Therefore, $B[u, \text{rad}(G) - 1] \cap B[v, \text{rad}(G) - 1] \cap X \supseteq B[u, \ell + 1] \cap B[v, \ell + 1] \cap X \neq \emptyset$. If furthermore $u \in X$, then $B[u, \text{rad}(G) - 2] \cap X = B[u, \text{rad}(G) - 1] \cap X$ because $\text{rad}(G)$ is even. We proceed similarly if $v \in X$, thus proving the claim. By the Helly property, there must exist a vertex c in X which is at distance at most $\text{rad}(G) - 1$ from every $v \in X \cup Y$. Necessarily, $e(c) \leq \text{rad}(G) - 1$, contradicting the definition of the radius of G . If $\text{rad}(G)$ is odd, then consider in G the system of half-balls $\mathcal{F}_X(G) := \{B[v, \text{rad}(G) - 1] \cap X : v \in X\} \cup \{B[v, \text{rad}(G) - 2] \cap X : v \in Y\}$. Arguing the same way as above, and noting that for $v \in Y$, we have $B[v, \text{rad}(G) - 2] \cap X = B[v, \text{rad}(G) - 1] \cap X$ when $\text{rad}(G)$ is odd, we will get again a vertex c in X which is at distance at most $\text{rad}(G) - 1$ from every $v \in X \cup Y$ (i.e., with $e(c) \leq \text{rad}(G) - 1$), giving again a contradiction. \blacksquare

Proposition 17 *Every Helly graph $G = (V, E)$ has:*

- (1) *a radius certificate of size at most 2;*
- (2) *a diameter certificate of size at most $\omega(C(G))$ (the size of a largest clique in the subgraph induced by $C(G)$).*

Proof. By Proposition 3 and Proposition 16, we are done with (1). For (2), by Proposition 4 and Proposition 16, it remains only to consider the case when $\text{diam}(G) = 2 \text{rad}(G) - 1$. We will show that, in this case, there is a clique $K \subseteq C(G)$ such that, for every $v \in V$, $d(v, K) \leq \text{rad}(G) - 1$ holds. Consider in $C(G)$ a maximal by inclusion clique $K \subseteq C(G)$ and assume that, for some vertex u , $d(u, K) > \text{rad}(G) - 1$. As $K \subseteq C(G)$, necessarily, $d(u, c) = \text{rad}(G)$ holds for every $c \in K$. Consider in G the system of balls $\{B[v, \text{rad}(G)] : v \in V\} \cup \{B[c, 1] : c \in K\} \cup \{B[u, \text{rad}(G) - 1]\}$. Since, for every $u, v \in V$ and $c \in K$, $d(u, v) \leq \text{diam}(G) \leq 2 \text{rad}(G) - 1$ and $d(v, c) \leq \text{rad}(G)$, all these balls pairwise intersect. By the Helly property, there is a vertex c' in G which is adjacent to all $c \in K$, at distance $\text{rad}(G) - 1$ from u , and at distance at most $\text{rad}(G)$ from each $v \in V$. Necessarily, c' belongs to $C(G)$ and $K \cup \{c'\}$ is a clique from $C(G)$ with one more vertex than K has. This contradicts the maximality of K , proving that for every $v \in V$, $d(v, K) \leq \text{rad}(G) - 1$ holds. Hence, K is a diameter certificate of G since, for every $v \in V$, there is a vertex $c \in K$ such that $d(v, c) + e(c) \leq \text{rad}(G) - 1 + \text{rad}(G) = 2 \text{rad}(G) - 1 = \text{diam}(G)$. \blacksquare

The upper bound given in Proposition 17(2) is sharp. For example, in a complete graph, all vertices must be included in the diameter certificate.

Proposition 18 *Every bipartite Helly graph $G = (X \cup Y, E)$ has:*

- (1) *a radius certificate of size at most 4;*
- (2) *a diameter certificate of size at most $\omega_{\text{bip}}(C(G))$ (the size of a largest biclique in the subgraph induced by $C(G)$).*

Proof. (1) The case $\text{diam}(G) \geq 2\text{rad}(G) - 1$ is covered by Proposition 3. Therefore, by Proposition 16, we only need to consider the case when $\text{diam}(G) = 2\text{rad}(G) - 2$. Let $x, x' \in X$ be vertices of X maximizing $d(x, x')$. In the same way, let $y, y' \in Y$ be vertices of Y maximizing $d(y, y')$. We prove, in what follows, that $L = \{x, x', y, y'\}$ forms a radius certificate. For that, without loss of generality, assume $d(x, x') \geq d(y, y')$. Since G is bipartite and $\text{diam}(G)$ is even, we get $d(x, x') = \text{diam}(G) = 2\text{rad}(G) - 2$. In particular, for every vertex $v \in X \cup Y$, we have $\max\{d(v, x), d(v, x')\} \geq \text{rad}(G) - 1$. If furthermore $\text{rad}(G)$ is even and $v \in X$ ($\text{rad}(G)$ is odd and $v \in Y$, resp.), then we get $\max\{d(v, x), d(v, x')\} \geq \text{rad}(G)$. By symmetry, we thus deduce that if $d(y, y') = d(x, x')$, then L indeed forms a radius certificate. Thus, from now on, suppose $d(y, y') < d(x, x')$, which implies $d(y, y') = \text{diam}(G) - 2 = 2\text{rad}(G) - 4$. In order to complete the proof, it suffices to show that this case can never happen. Let H_Y be the graph with vertex set Y such that there is an edge between every two vertices of Y that are at distance two in G . We denote by d_Y the distance function in H_Y , while we keep denoting by d (without a subscript) the distance function in G . Since G is bipartite, we have $d_Y(u, v) = d(u, v)/2$ for every $u, v \in Y$. In particular, by the choice of y, y' we get $\text{diam}(H_Y) = d_Y(y, y') = d(y, y')/2 = \text{rad}(G) - 2$. It was observed in [32, Lemma 1] that H_Y is a Helly graph. If $\text{rad}(G)$ is even, then $\text{diam}(H_Y)$ is even, and so by Proposition 16 we get $\text{rad}(H_Y) = \text{diam}(H_Y)/2 = \text{rad}(G)/2 - 1$. But then, for any $c_Y \in C(H_Y)$, we would obtain $e_G(c_Y) \leq 2\text{rad}(H_Y) + 1 = \text{rad}(G) - 2 + 1 = \text{rad}(G) - 1 < \text{rad}(G)$, which is a contradiction. As a result, $\text{rad}(G)$ must be odd. By Proposition 16, it implies that $\text{rad}(H_Y) = (\text{diam}(H_Y) + 1)/2 = (\text{rad}(G) - 1)/2$. Let $K \subseteq C(H_Y)$ be a clique of H_Y such that $d_Y(v, K) \leq \text{rad}(H_Y) - 1$ for every $v \in Y$, whose existence follows from Proposition 17(2). Since K is a clique, the half-balls in $\{B[z, 1] \cap X : z \in K\}$ pairwise intersect in G . By the Helly property, there exists a vertex $c_K \in X$ such that $K \subseteq N(c_K)$. But then, $e_G(c_K) \leq 1 + \max\{d(v, K) : v \in X \cup Y\} \leq 2 + \max\{d(v, K) : v \in Y\} \leq 2 + 2(\text{rad}(H_Y) - 1) = 2\text{rad}(H_Y) = \text{rad}(G) - 1 < \text{rad}(G)$, which is again a contradiction.

(2) The case when $\text{diam}(G) = 2\text{rad}(G)$ is covered by Proposition 4; any central vertex of G is a diameter certificate. Assume $\text{diam}(G) = 2\text{rad}(G) - 1$. We will show that, in this case, two adjacent central vertices form a diameter certificate. Note that since $\text{diam}(G) = 2\text{rad}(G) - 1$, i.e., the diameter is odd, for any two vertices u, w from X (or from Y), $d(u, w) \leq 2\text{rad}(G) - 2$ holds. Assume $\text{rad}(G)$ is odd. Consider in G the system of half-balls $\mathcal{F}_X(G) := \{B[v, \text{rad}(G) - 1] \cap X : v \in X\}$. These half-balls pairwise intersect. By the Helly property, there must exist a vertex c_X in X which is at distance at most $\text{rad}(G) - 1$ from every $v \in X$ and, hence, at distance at most $\text{rad}(G)$ from every $v \in X \cup Y$. Hence, $c_X \in C(G)$. Consider now a system of half-balls $\mathcal{F}_Y(G) := \{B[v, \text{rad}(G) - 1] \cap Y : v \in Y\}$ and a half-ball $B[c_X, 1] \cap Y$. All these half-balls pairwise intersect. By the Helly property, there must exist a vertex c_Y in Y which is adjacent to c_X , and which is at distance at most $\text{rad}(G) - 1$ from every $v \in Y$. Since c_Y is at distance at most $\text{rad}(G)$ from every $v \in X \cup Y$, $c_Y \in C(G)$. So, we found two adjacent vertices $c_X, c_Y \in C(G)$ such that, for every vertex $v \in X$, $d(v, c_X) \leq \text{rad}(G) - 1$ and, therefore, $d(v, c_X) + e(c_X) \leq \text{rad}(G) - 1 + \text{rad}(G) = \text{diam}(G)$, and, for every vertex $v \in Y$, $d(v, c_Y) \leq \text{rad}(G) - 1$ and, therefore, $d(v, c_Y) + e(c_Y) \leq \text{rad}(G) - 1 + \text{rad}(G) = \text{diam}(G)$. Hence, $\{c_X, c_Y\}$ is a diameter certificate. The case when $\text{rad}(G)$ is even is similar. We just need to consider first the system of half-balls $\mathcal{F}'_X(G) := \{B[v, \text{rad}(G) - 1] \cap X : v \in Y\}$ which will produce, by the Helly property, a central vertex $c_X \in X$ which is at distance at most $\text{rad}(G) - 1$ from every $v \in Y$. Then, we can consider the system of half-balls $\mathcal{F}'_Y(G) := \{B[v, \text{rad}(G) - 1] \cap Y : v \in X\}$ plus a half-ball $B[c_X, 1] \cap Y$ which together will produce a central vertex $c_Y \in Y$ which is at distance at most $\text{rad}(G) - 1$ from every $v \in X$. Again, $\{c_X, c_Y\} \subseteq C(G)$ forms a diameter certificate.

Finally assume $\text{diam}(G) = 2\text{rad}(G) - 2$. We will show that, in this case, a biclique consisting of central vertices of G forms a diameter certificate. Assume $\text{rad}(G)$ is odd (the case, when $\text{rad}(G)$ is even, is very similar as above). As in the case $\text{diam}(G) = 2\text{rad}(G) - 1$, we can get two adjacent

vertices $c_X \in C(G) \cap X$ and $c_Y \in C(G) \cap Y$ such that, for every vertex $v \in X$, $d(v, c_X) \leq \text{rad}(G) - 1$ and, for every vertex $v \in Y$, $d(v, c_Y) \leq \text{rad}(G) - 1$. Let $K_{p,q}$ be a maximal by inclusion biclique in $C(G)$ containing edge $c_X c_Y$. We will show that $d(v, K_{p,q}) \leq \text{rad}(G) - 2$ holds for every $v \in X \cup Y$. Assume there is a vertex v' such that $d(v', K_{p,q}) > \text{rad}(G) - 2$. Without loss of generality, say $v' \in X$. Since $d(v', c_X) \leq \text{rad}(G) - 1$, we have $d(v', K_{p,q}) = \text{rad}(G) - 1$. Consider a system of half-balls $\{B[v, \text{rad}(G) - 1] \cap Y : v \in Y\} \cup \{B[c, 1] \cap Y : c \in K_{p,q} \cap X\} \cup \{B[v', \text{rad}(G) - 2] \cap Y\}$. Since the diameter $\text{diam}(G) = 2 \text{rad}(G) - 2$ is even and, for every $u \in Y$, $d(v', u)$ is odd, necessarily, $d(v', u) \leq 2 \text{rad}(G) - 3$ and $B[u, \text{rad}(G) - 1]$ intersects $B[v', \text{rad}(G) - 2]$. Furthermore, since for every $c \in K_{p,q} \cap X$, we have $d(v', c) \leq \text{rad}(G)$, and $d(v', c)$ is even while $\text{rad}(G)$ is odd, we necessarily have $d(v', c) \leq \text{rad}(G) - 1$, and $B[c, 1]$ intersects $B[v', \text{rad}(G) - 2]$. Since we have $c_Y \in B[v, \text{rad}(G) - 1] \cap B[c, 1]$ for all $v \in Y$ and $c \in K_{p,q} \cap X$, we conclude that all half-balls pairwise intersect. By the Helly property, there must exist a vertex c' in Y which is adjacent to every $c \in K_{p,q} \cap X$, at distance at most $\text{rad}(G) - 2$ from v' , and at distance at most $\text{rad}(G) - 1$ from every $y \in Y$. That is, $c' \in C(G) \cap Y$. Moreover, $K_{p,q} \cup \{c'\}$ is a biclique in $C(G)$ containing one more vertex than $K_{p,q}$. This contradicts the maximality of $K_{p,q}$, proving that $d(v, K_{p,q}) \leq \text{rad}(G) - 2$ holds for every $v \in X \cup Y$. Therefore, the biclique $K_{p,q}$ is a diameter certificate as for every $v \in X \cup Y$ there is a vertex c in $K_{p,q}$ such that $d(v, c) + e(c) \leq \text{rad}(G) - 2 + \text{rad}(G) = \text{diam}(G)$. ■

Again, we can observe that the upper bound stated in Proposition 18(2) is sharp. For instance, in a complete bipartite graph, all vertices must be included in the diameter certificate.

Corollary 4 *For every (bipartite) Helly graph G , we can compute its radius in $O(m \log^3 n)$ time with high probability.*

Proof. It directly follows from Theorem 6, in combination with Propositions 17 and 18. ■

22 Graphs of bounded asteroidal number

We finally give an application of this paper's framework to parameterized complexity. Namely, an *asteroidal set* in a graph $G = (V, E)$ is an independent set $A \subseteq V$ such that, for every $a \in A$, all vertices of $A \setminus \{a\}$ must be in a same connected component of $G \setminus B[a, 1]$. The *asteroidal number* of G is the largest cardinality of its asteroidal sets. In particular, the graphs of asteroidal number one are exactly the complete graphs. The graphs of asteroidal number at most two are called AT-free graphs, and they contain cocomparability graphs, interval graphs and permutation graphs amongst their interesting subclasses. In [33], a deterministic $O(m^{3/2})$ -time algorithm for computing all eccentricities in an AT-free graph is presented. In [34], a deterministic algorithm for computing the diameter of graphs with asteroidal number k in $O(k^3 m^{3/2})$ time is given. However, the complexity of computing the radius in graphs of asteroidal number $k \geq 3$ was open until this work. We prove the following result in this section.

Theorem 17 *For every graph $G = (V, E)$ with asteroidal number at most k , its radius and a central vertex can be computed in $O(km^{3/2})$ time.*

Roughly, the result follows from replacing in Algorithm 1 the notion of antipode with that of *extremity*. A vertex v in a graph G is called an extremity if and only if $G \setminus B[v, 1]$ is connected. There are graphs with no extremities, such as complete bipartite graphs. Therefore, in what follows, we need an additional assumption on the graphs considered. Namely, a module in a graph $G = (V, E)$

is a vertex subset $M \subseteq V$ such that every vertex of $V \setminus M$ is either adjacent to all of M , or nonadjacent to all of it. A graph $G = (V, E)$ is prime if and only if it has no other modules but \emptyset, V and $\{v\}$, for every $v \in V$.

Lemma 10 (see Theorem 14 in [21]) *Computing the radius of any graph G can be reduced in linear time to computing the radius of one of its prime induced subgraphs G' .*

Hence, we only need to consider prime graphs of bounded asteroidal number. We need the following additional results:

Lemma 11 (see Lemma 19 in [34]) *If $G = (V, E)$ is prime and $n \geq 3$, then for every vertex v , there exists an extremity u such that $d(u, v) = e(v)$, which can be computed in linear time.*

Lemma 12 (see Lemma 7 in [34]) *If $G = (V, E)$ has asteroidal number at most k , then it contains at most $O(k\sqrt{m})$ extremities.*

We are now ready to prove the main result of this section, namely:

Proof.[of Theorem 17] We may assume that G is prime (by Lemma 10) and that $n \geq 3$. We modify Algorithm 1 as follows: at every iteration, we add in L an extremity a such that, for the vertex u considered, $d(u, a) = e(u)$. By Lemma 11, such an extremity a always exists, and it can be computed in linear time. Doing so, we still have the invariant $e_L(u) = e(u)$ for every $u \in K$. In particular, we can prove as before (see the proof of Theorem 11) that this algorithm correctly computes the radius and a central vertex. By Lemma 12, the number of iterations is in $O(k\sqrt{m})$ at most. Therefore, the runtime of the algorithm is in $O(km^{3/2})$. ■

23 Conclusion

In this paper we extensively study this idea of small certificates for radius, diameter and all-eccentricities. It gives us another view point on existing algorithms, somehow explaining why they are practically so efficient. It also leads to new algorithmic ideas to overcome the quadratic barrier for radius, diameter and all eccentricities. We are convinced that the celebrated notion of certificate can still lead to fruitful developments in the study of other problems in P and their complexity barrier.

References

- [1] Amir Abboud, Virginia Vassilevska Williams, and Joshua R. Wang. Approximation and fixed parameter subquadratic algorithms for radius and diameter in sparse graphs. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 377–391. SIAM, 2016. URL: <https://doi.org/10.1137/1.9781611974331.ch28>, doi:10.1137/1.9781611974331.CH28.
- [2] Ittai Abraham, Daniel Delling, Andrew V. Goldberg, and Renato Fonseca F. Werneck. Hierarchical hub labelings for shortest paths. In Leah Epstein and Paolo Ferragina, editors, *Algorithms - ESA 2012 - 20th Annual European Symposium, Ljubljana, Slovenia, September 10-12, 2012. Proceedings*, volume 7501 of *Lecture Notes in Computer Science*, pages 24–35. Springer, 2012. doi:10.1007/978-3-642-33090-2_4.

- [3] Muad Abu-Ata and Feodor F. Dragan. Metric tree-like structures in real-world networks: an empirical study. *Networks*, 67(1):49–68, 2016. URL: <https://doi.org/10.1002/net.21631>, doi:10.1002/NET.21631.
- [4] Takuya Akiba, Yoichi Iwata, and Yuki Kawata. An exact algorithm for diameters of large real directed graphs. In Evripidis Bampis, editor, *Experimental Algorithms - 14th International Symposium, SEA 2015, Paris, France, June 29 - July 1, 2015, Proceedings*, volume 9125 of *Lecture Notes in Computer Science*, pages 56–67. Springer, 2015. doi:10.1007/978-3-319-20086-6_5.
- [5] Eyad Alkassar, Sascha Böhme, Kurt Mehlhorn, and Christine Rizkallah. Verification of certifying computations. In Ganesh Gopalakrishnan and Shaz Qadeer, editors, *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings*, volume 6806 of *Lecture Notes in Computer Science*, pages 67–82. Springer, 2011. doi:10.1007/978-3-642-22110-1_7.
- [6] Lars Backstrom, Paolo Boldi, Marco Rosa, Johan Ugander, and Sebastiano Vigna. Four degrees of separation. In Noshir S. Contractor, Brian Uzzi, Michael W. Macy, and Wolfgang Nejdl, editors, *Web Science 2012, WebSci '12, Evanston, IL, USA - June 22 - 24, 2012*, pages 33–42. ACM, 2012. doi:10.1145/2380718.2380723.
- [7] Manuel Blum, Robert W. Floyd, Vaughan R. Pratt, Ronald L. Rivest, and Robert Endre Tarjan. Time bounds for selection. *J. Comput. Syst. Sci.*, 7(4):448–461, 1973. doi:10.1016/S0022-0000(73)80033-9.
- [8] Béla Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *Eur. J. Comb.*, 1(4):311–316, 1980. doi:10.1016/S0195-6698(80)80030-8.
- [9] Michele Borassi, Pierluigi Crescenzi, Michel Habib, Walter A. Kosters, Andrea Marino, and Frank W. Takes. Fast diameter and radius bfs-based computation in (weakly connected) real-world graphs: With an application to the six degrees of separation games. *Theor. Comput. Sci.*, 586:59–80, 2015. URL: <https://doi.org/10.1016/j.tcs.2015.02.033>, doi:10.1016/J.TCS.2015.02.033.
- [10] Michele Borassi, Pierluigi Crescenzi, and Luca Trevisan. An axiomatic and an average-case analysis of algorithms and heuristics for metric properties of graphs. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 920–939. SIAM, 2017. doi:10.1137/1.9781611974782.58.
- [11] Massimo Cairo, Roberto Grossi, and Romeo Rizzi. New bounds for approximating extremal distances in undirected graphs. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 363–376. SIAM, 2016. URL: <https://doi.org/10.1137/1.9781611974331.ch27>, doi:10.1137/1.9781611974331.CH27.
- [12] Marco L. Carmosino, Jiawei Gao, Russell Impagliazzo, Ivan Mihajlin, Ramamohan Paturi, and Stefan Schneider. Nondeterministic extensions of the strong exponential time hypothesis and consequences for non-reducibility. In Madhu Sudan, editor, *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science, Cambridge, MA, USA, January 14-16, 2016*, pages 261–270. ACM, 2016. doi:10.1145/2840728.2840746.

- [13] Victor Chepoi. Some d -convexity properties in triangulated graphs. *Mathematical Research*, 87:164–177, 1986. Știința, Chișinău (Russian).
- [14] Victor Chepoi and Feodor F. Dragan. A linear-time algorithm for finding a central vertex of a chordal graph. In Jan van Leeuwen, editor, *Algorithms - ESA '94, Second Annual European Symposium, Utrecht, The Netherlands, September 26-28, 1994, Proceedings*, volume 855 of *Lecture Notes in Computer Science*, pages 159–170. Springer, 1994. URL: <https://doi.org/10.1007/BFb0049406>, doi:10.1007/BFB0049406.
- [15] Victor Chepoi, Feodor F. Dragan, Bertrand Estellon, Michel Habib, and Yann 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, College Park, MD, USA, June 9-11, 2008*, pages 59–68. ACM, 2008. doi:10.1145/1377676.1377687.
- [16] Victor Chepoi, Feodor F. Dragan, Michel Habib, Yann Vaxès, and Hend Alrasheed. Fast approximation of eccentricities and distances in hyperbolic graphs. *J. Graph Algorithms Appl.*, 23(2):393–433, 2019. URL: <https://doi.org/10.7155/jgaa.00496>, doi:10.7155/JGAA.00496.
- [17] Victor Chepoi and Bertrand Estellon. Packing and covering δ -hyperbolic spaces by balls. In Moses Charikar, Klaus Jansen, Omer Reingold, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 10th International Workshop, APPROX 2007, and 11th International Workshop, RANDOM 2007, Princeton, NJ, USA, August 20-22, 2007, Proceedings*, volume 4627 of *Lecture Notes in Computer Science*, pages 59–73. Springer, 2007. doi:10.1007/978-3-540-74208-1_5.
- [18] Victor D Chepoi. Centers of triangulated graphs. *Mathematical Notes of the Academy of Sciences of the USSR*, 43:82–86, 1988. URL: https://pageperso.lis-lab.fr/~victor.chepoi/centers_triang.pdf.
- [19] Derek G. Corneil, Feodor F. Dragan, Michel Habib, and Christophe Paul. Diameter determination on restricted graph families. *Discret. Appl. Math.*, 113(2-3):143–166, 2001. doi:10.1016/S0166-218X(00)00281-X.
- [20] Derek G. Corneil, Feodor F. Dragan, and Ekkehard Köhler. On the power of BFS to determine a graph’s diameter. *Networks*, 42(4):209–222, 2003. URL: <https://doi.org/10.1002/net.10098>, doi:10.1002/NET.10098.
- [21] David Coudert, Guillaume Ducoffe, and Alexandru Popa. Fully polynomial FPT algorithms for some classes of bounded clique-width graphs. *ACM Trans. Algorithms*, 15(3):33:1–33:57, 2019. doi:10.1145/3310228.
- [22] Pilu Crescenzi, Roberto Grossi, Michel Habib, Leonardo LANZI, and Andrea Marino. On computing the diameter of real-world undirected graphs. *Theor. Comput. Sci.*, 514:84–95, 2013. URL: <https://doi.org/10.1016/j.tcs.2012.09.018>, doi:10.1016/J.TCS.2012.09.018.
- [23] Mina Dalirrooyfard, Ray Li, and Virginia Vassilevska Williams. Hardness of approximate diameter: Now for undirected graphs. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 1021–1032. IEEE, 2021. doi:10.1109/FOCS52979.2021.00102.

- [24] Fabien de Montgolfier, Mauricio Soto, and Laurent Viennot. Treewidth and hyperbolicity of the internet. In *Proceedings of The Tenth IEEE International Symposium on Networking Computing and Applications, NCA 2011, August 25-27, 2011, Cambridge, Massachusetts, USA*, pages 25–32. IEEE Computer Society, 2011. doi:10.1109/NCA.2011.11.
- [25] Irit Dinur and David Steurer. Analytical approach to parallel repetition. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 624–633. ACM, 2014. doi:10.1145/2591796.2591884.
- [26] Feodor F. Dragan. *Centers of graphs and the Helly property*. PhD thesis, Moldova State University, 1989. (in Russian).
- [27] Feodor F. Dragan. Conditions for coincidence of local and global minima for eccentricity function on graphs and the helly property. *Applied Mathematics and Information Science*, pages 49–56, 1990.
- [28] Feodor F. Dragan, Guillaume Ducoffe, and Heather M. Guarnera. Fast deterministic algorithms for computing all eccentricities in (hyperbolic) helly graphs. In Anna Lubiw and Mohammad R. Salavatipour, editors, *Algorithms and Data Structures - 17th International Symposium, WADS 2021, Virtual Event, August 9-11, 2021, Proceedings*, volume 12808 of *Lecture Notes in Computer Science*, pages 300–314. Springer, 2021. doi:10.1007/978-3-030-83508-8_22.
- [29] Feodor F. Dragan and Heather M. Guarnera. Helly-gap of a graph and vertex eccentricities. *Theor. Comput. Sci.*, 867:68–84, 2021. URL: <https://doi.org/10.1016/j.tcs.2021.03.022>, doi:10.1016/J.TCS.2021.03.022.
- [30] Feodor F. Dragan, Falk Nicolai, and Andreas 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, 22nd International Workshop, WG ’96, Cadenabbia (Como), Italy, June 12-14, 1996, Proceedings*, volume 1197 of *Lecture Notes in Computer Science*, pages 166–180. Springer, 1996. doi:10.1007/3-540-62559-3_15.
- [31] Andreas WM Dress. Trees, tight extensions of metric spaces, and the cohomological dimension of certain groups: a note on combinatorial properties of metric spaces. *Advances in Mathematics*, 53(3):321–402, 1984. URL: [https://doi.org/10.1016/0001-8708\(84\)90029-X](https://doi.org/10.1016/0001-8708(84)90029-X).
- [32] Guillaume Ducoffe. Beyond helly graphs: The diameter problem on absolute retracts. In Lukasz Kowalik, Michal Pilipczuk, and Pawel Rzazewski, editors, *Graph-Theoretic Concepts in Computer Science - 47th International Workshop, WG 2021, Warsaw, Poland, June 23-25, 2021, Revised Selected Papers*, volume 12911 of *Lecture Notes in Computer Science*, pages 321–335. Springer, 2021. doi:10.1007/978-3-030-86838-3_25.
- [33] Guillaume Ducoffe. The diameter of at-free graphs. *J. Graph Theory*, 99(4):594–614, 2022. URL: <https://doi.org/10.1002/jgt.22754>, doi:10.1002/JGT.22754.
- [34] Guillaume Ducoffe. Obstructions to faster diameter computation: Asteroidal sets. In Holger Dell and Jesper Nederlof, editors, *17th International Symposium on Parameterized and Exact Computation, IPEC 2022, September 7-9, 2022, Potsdam, Germany*, volume 249 of *LIPICs*, pages 10:1–10:24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. URL: <https://doi.org/10.4230/LIPICs.IPEC.2022.10>, doi:10.4230/LIPICs.IPEC.2022.10.

- [35] Guillaume Ducoffe and Feodor F. Dragan. A story of diameter, radius, and (almost) helly property. *Networks*, 77(3):435–453, 2021. URL: <https://doi.org/10.1002/net.21998>, doi:10.1002/NET.21998.
- [36] Robert Geisberger, Peter Sanders, Dominik Schultes, and Daniel Delling. Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In Catherine C. McGeoch, editor, *Experimental Algorithms, 7th International Workshop, WEA 2008, Provincetown, MA, USA, May 30-June 1, 2008, Proceedings*, volume 5038 of *Lecture Notes in Computer Science*, pages 319–333. Springer, 2008. doi:10.1007/978-3-540-68552-4_24.
- [37] Mikhael Gromov. Hyperbolic groups. In *Essays in group theory*, pages 75–263. Springer, 1987.
- [38] Heather M. Guarnera, Feodor F. Dragan, and Arne Leitert. Injective hulls of various graph classes. *Graphs Comb.*, 38(4):112, 2022. URL: <https://doi.org/10.1007/s00373-022-02512-z>, doi:10.1007/S00373-022-02512-Z.
- [39] Gabriel Y. Handler. Minimax location of a facility in an undirected tree graph. *Transportation Science*, 7(3):287–293, 1973. URL: <https://pubsonline.informs.org/doi/abs/10.1287/trsc.7.3.287>.
- [40] John R. Isbell. Injective envelopes of banach spaces are rigidly attached. *Commentarii mathematici Helvetici*, 39:65–76, 1964. URL: <https://projecteuclid.org/journals/bulletin-of-the-american-mathematical-society/volume-70>
- [41] W. Sean Kennedy, Iraj Saniee, and Onuttom Narayan. On the hyperbolicity of large-scale networks and its estimation. In James Joshi, George Karypis, Ling Liu, Xiaohua Hu, Ronay Ak, Yinglong Xia, Weijia Xu, Aki-Hiro Sato, Sudarsan Rachuri, Lyle H. Ungar, Philip S. Yu, Rama Govindaraju, and Toyotaro Suzumura, editors, *2016 IEEE International Conference on Big Data (IEEE BigData 2016), Washington DC, USA, December 5-8, 2016*, pages 3344–3351. IEEE Computer Society, 2016. URL: <https://doi.org/10.1109/BigData.2016.7840994>, doi:10.1109/BIGDATA.2016.7840994.
- [42] Marvin Künnemann. On nondeterministic derandomization of freivalds’ algorithm: Consequences, avenues and algorithmic progress. In Yossi Azar, Hannah Bast, and Grzegorz Herman, editors, *26th Annual European Symposium on Algorithms, ESA 2018, August 20-22, 2018, Helsinki, Finland*, volume 112 of *LIPICs*, pages 56:1–56:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPICs.ESA.2018.56.
- [43] Clémence Magnien, Matthieu Latapy, and Michel Habib. Fast computation of empirically tight bounds for the diameter of massive graphs. *ACM J. Exp. Algorithmics*, 13, 2008. doi:10.1145/1412228.1455266.
- [44] Ross M. McConnell, Kurt Mehlhorn, Stefan Näher, and Pascal Schweitzer. Certifying algorithms. *Comput. Sci. Rev.*, 5(2):119–161, 2011. doi:10.1016/J.COSREV.2010.09.009.
- [45] R. Nandakumar and K. Parthasarathy. Eccentricity-preserving spanning trees. *J. Math. Phys. Sci.*, 24(1):33–35, 1990.
- [46] Liam Roditty and Virginia Vassilevska Williams. Fast approximation algorithms for the diameter and radius of sparse graphs. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *Symposium on Theory of Computing Conference, STOC’13, Palo Alto, CA, USA, June 1-4, 2013*, pages 515–524. ACM, 2013. doi:10.1145/2488608.2488673.

- [47] Sandeep Sen and V. N. Muralidhara. The covert set-cover problem with application to network discovery. In Md. Saidur Rahman and Satoshi Fujita, editors, *WALCOM: Algorithms and Computation, 4th International Workshop, WALCOM 2010, Dhaka, Bangladesh, February 10-12, 2010. Proceedings*, volume 5942 of *Lecture Notes in Computer Science*, pages 228–239. Springer, 2010. doi:10.1007/978-3-642-11440-3_21.
- [48] Frank W. Takes and Walter A. Kosters. Determining the diameter of small world networks. In Craig Macdonald, Iadh Ounis, and Ian Ruthven, editors, *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*, pages 1191–1196. ACM, 2011. doi:10.1145/2063576.2063748.
- [49] Frank W. Takes and Walter A. Kosters. Computing the eccentricity distribution of large graphs. *Algorithms*, 6(1):100–118, 2013. URL: <https://doi.org/10.3390/a6010100>, doi:10.3390/A6010100.