

Minimum Eccentricity Shortest Paths in some Structured Graph Classes

Feodor F. Dragan¹ Arne Leitert¹

¹Department of Computer Science,
Kent State University, Kent, Ohio, USA

Abstract

We investigate the *Minimum Eccentricity Shortest Path* problem in some structured graph classes. It asks for a given graph to find a shortest path with minimum eccentricity. Although it is NP-hard in general graphs, we demonstrate that a minimum eccentricity shortest path can be found in linear time for distance-hereditary graphs (generalizing the previous result for trees) and give a generalised approach which allows to solve the problem in polynomial time for other graph classes. This includes chordal graphs, dually chordal graphs, graphs with bounded tree-length, and graphs with bounded hyperbolicity. Additionally, we give a simple algorithm to compute an additive approximation for graphs with bounded tree-length and graphs with bounded hyperbolicity.

Submitted: November 2015	Accepted: April 2016	Final: April 2016	Published: May 2016
	Article type: Regular paper	Communicated by: G. Liotta	

This work was partially supported by the NIH grant R01 GM103309.

E-mail addresses: dragan@cs.kent.edu (Feodor F. Dragan) aleitert@cs.kent.edu (Arne Leitert)

1 Introduction

The *Minimum Eccentricity Shortest Path* problem asks for a given graph $G = (V, E)$ to find a shortest path P such that for each other shortest path Q , $\text{ecc}(P) \leq \text{ecc}(Q)$ holds. Here, the eccentricity of a set $S \subseteq V$ in G is $\text{ecc}(S) = \max_{u \in V} d_G(u, S)$. This problem was introduced in [12]. It may arise in determining a “most accessible” speedy linear route in a network and can find applications in communication networks, transportation planning, water resource management and fluid transportation. It was also shown in [11, 12] that a minimum eccentricity shortest path plays a crucial role in obtaining the best to date approximation algorithm for a minimum distortion embedding of a graph into the line. Specifically, every graph G with a shortest path of eccentricity r admits an embedding f of G into the line with distortion at most $(8r + 2) \text{ld}(G)$, where $\text{ld}(G)$ is the minimum line-distortion of G (see [12] for details). Furthermore, if a shortest path of G of eccentricity r is given in advance, then such an embedding f can be found in linear time. Note also that every graph has a shortest path of eccentricity at most $\lfloor \text{ld}(G)/2 \rfloor$.

Those applications motivate investigation of the Minimum Eccentricity Shortest Path problem in general graphs and in particular graph classes. Fast algorithms for it will imply fast approximation algorithms for the minimum line distortion problem. Existence of low eccentricity shortest paths in structured graph classes will imply low approximation bounds for those classes. For example, all AT-free graphs (hence, all interval, permutation, cocomparability graphs) enjoy a shortest path of eccentricity at most 1 [7], all convex bipartite graphs enjoy a shortest path of eccentricity at most 2 [11].

In [12], the Minimum Eccentricity Shortest Path problem was investigated in general graphs. It was shown that its decision version is NP-complete (even for graphs with vertex degree at most 3). However, there are efficient approximation algorithms: a 2-approximation, a 3-approximation, and an 8-approximation for the problem can be computed in $\mathcal{O}(n^3)$ time, in $\mathcal{O}(nm)$ time, and in linear time, respectively. Furthermore, a shortest path of minimum eccentricity r in general graphs can be computed in $\mathcal{O}(n^{2r+2}m)$ time. Paper [12] initiated also the study of the Minimum Eccentricity Shortest Path problem in special graph classes by showing that a minimum eccentricity shortest path in trees can be found in linear time. In fact, every diametral path of a tree is a minimum eccentricity shortest path.

In this paper, we design efficient algorithms for the Minimum Eccentricity Shortest Path problem in distance-hereditary graphs, in chordal graphs, in dually chordal graphs, and in more general graphs with bounded tree-length or with bounded hyperbolicity. Additionally, we give a simple algorithm to compute an additive approximation for graphs with bounded tree-length and graphs with bounded hyperbolicity. Table 1 gives an overview of our results.

Note that our Minimum Eccentricity Shortest Path problem is close but different from the *Central Path* problem in graphs introduced in [21]. It asks for a given graph G to find a path P (not necessarily shortest) such that any other path of G has eccentricity at least $\text{ecc}(P)$. The Central Path problem generalizes

Graph class	Runtime	Approx.
distance-hereditary	linear	
chordal	$\mathcal{O}(n^2m)$ linear	+2
dually chordal	$\mathcal{O}(n^3m)$	
tree-length λ	$\mathcal{O}(n^{3\lambda+3})$ $\mathcal{O}(\lambda m)$	+2.5 λ
tree-breadth ρ	$\mathcal{O}(n^{6\rho+3})$	
δ -hyperbolic	$\mathcal{O}(n^{4\delta+4})$ $\mathcal{O}(\delta m)$	+ $\mathcal{O}(\delta \log n)$

Table 1: Runtime for solving the Minimum Eccentricity Shortest Path problem for some graph classes. Also, if the solution is not optimal, the maximal difference to an optimal solution is shown.

the Hamiltonian Path problem and therefore is NP-hard even for chordal graphs [19]. Our problem is polynomial time solvable for chordal graphs.

2 Notions and Notations

All graphs occurring in this paper are connected, finite, unweighted, undirected, loopless and without multiple edges. For a graph $G = (V, E)$, we use $n = |V|$ and $m = |E|$ to denote the cardinality of the vertex set and the edge set of G . $G[S]$ denotes the *induced subgraph* of G with the vertex set S .

The *length* of a path from a vertex v to a vertex u is the number of edges in the path. The *distance* $d_G(u, v)$ of two vertices u and v is the length of a shortest path connecting u and v . The distance between a vertex v and a set $S \subseteq V$ is defined as $d_G(v, S) = \min_{u \in S} d_G(u, v)$. The *eccentricity* $\text{ecc}(v)$ of a vertex v is $\max_{u \in V} d_G(u, v)$. For a set $S \subseteq V$, its eccentricity is $\text{ecc}(S) = \max_{u \in V} d_G(u, S)$. For a vertex pair s, t , a shortest (s, t) -path P has *minimal eccentricity*, if there is no shortest (s, t) -path Q with $\text{ecc}(Q) < \text{ecc}(P)$. Two vertices x and y are called *mutually furthest* if $d_G(x, y) = \text{ecc}(x) = \text{ecc}(y)$. A vertex u is *k-dominated* by a vertex v (by a set $S \subset V$), if $d_G(u, v) \leq k$ ($d_G(u, S) \leq k$, respectively).

The *diameter* of a graph G is $\text{diam}(G) = \max_{u, v \in V} d_G(u, v)$. The diameter $\text{diam}_G(S)$ of a set $S \subseteq V$ is defined as $\max_{u, v \in S} d_G(u, v)$. A pair of vertices x, y of G is called a *diametral pair* if $d_G(x, y) = \text{diam}(G)$. In this case, every shortest path connecting x and y is called a *diametral path*.

For a vertex $v \in V$, $N(v) = \{u \in V \mid uv \in E\}$ is called the *open neighborhood*, and $N[v] = N(v) \cup \{v\}$ the *closed neighborhood* of v . $N^r[v] = \{u \in V \mid d_G(u, v) \leq r\}$ denotes the *disk* of radius r around vertex v . Additionally, $L_r^{(v)} = \{u \in V \mid d_G(u, v) = r\}$ denotes the vertices with distance r from v . For two vertices u and v , $I(u, v) = \{w \mid d_G(u, v) = d_G(u, w) + d_G(w, v)\}$ is the *interval* between u and v . The set $S_i(s, t) = L_i^{(s)} \cap I(u, v)$ is called a *slice* of the interval from u to v . For any set $S \subseteq V$ and a vertex v ,

$\text{Pr}(v, S) = \{u \in S \mid d_G(u, v) = d_G(v, S)\}$ denotes the *projection* of v on S .

A *chord* in a path is an edge connecting two non-consecutive vertices of the path. A set of vertices S is a *clique* if all vertices in S are pairwise adjacent. A graph is *chordal* if every cycle with at least four vertices has a chord. A graph is *distance-hereditary* if the distances in any connected induced subgraph are the same as they are in the original graph. A graph is *dually chordal* if it is the intersection graph of maximal cliques of a chordal graph. For more definitions of these classes and relations between them see [4].

3 A Linear-Time Algorithm for Distance-Hereditary Graphs

Distance-hereditary graphs can be defined as graphs where each chordless path is a shortest path [16]. Several interesting characterizations of distance-hereditary graphs in terms of metric and neighborhood properties, and forbidden configurations were provided by BANDELT and MULDER [1], and by D'ATRI and MOSCARINI [8]. The following proposition lists the basic information on distance-hereditary graphs that is needed in what follows.

Proposition 1 ([1, 8]) *For a graph G the following conditions are equivalent:*

- (1) G is distance-hereditary;
- (2) The house, domino, gem (see Fig. 1) and the cycles C_k of length $k \geq 5$ are not induced subgraphs of G ;
- (3) For an arbitrary vertex x of G and every pair of vertices $u, v \in L_k^{(x)}$, that are in the same connected component of the graph $G[V \setminus L_{k-1}^{(x)}]$, we have $N(v) \cap L_{k-1}^{(x)} = N(u) \cap L_{k-1}^{(x)}$.
- (4) (*4-point condition*) For any four vertices u, v, w, x of G at least two of the following distance sums are equal: $d_G(u, v) + d_G(w, x)$; $d_G(u, w) + d_G(v, x)$; $d_G(u, x) + d_G(v, w)$. If the smaller sums are equal, then the largest one exceeds the smaller ones at most by 2.

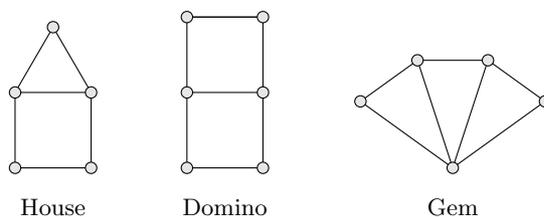


Figure 1: Forbidden induced subgraphs in a distance-hereditary graph.

As a consequence of statement (3) of Proposition 1 we get.

Corollary 1 *Let $P := P(s, t)$ be a shortest path in a distance-hereditary graph G connecting vertices s and t , and w be an arbitrary vertex of G . Let a be a vertex of $\text{Pr}(w, P)$ that is closest to s , and let b be a vertex of $\text{Pr}(w, P)$ that is closest to t . Then $d_G(a, b) \leq 2$ and there must be a vertex w' in G adjacent to both a and b and at distance $d_G(w, P) - 1$ from w .*

As a consequence of statement (4) of Proposition 1 we get.

Corollary 2 *Let x, y, v, u be arbitrary vertices of a distance-hereditary graph G with $v \in I(x, u)$, $u \in I(y, v)$, and $d_G(u, v) > 1$, then $d_G(x, y) = d_G(x, v) + d_G(v, u) + d_G(u, y)$. That is, if two shortest paths share ends of length at least 2, then their union is a shortest path.*

Proof: Consider distance sums $S_1 := d_G(x, v) + d_G(u, y)$, $S_2 := d_G(x, y) + d_G(u, v)$ and $S_3 := d_G(x, u) + d_G(v, y)$. Since $d_G(x, u) + d_G(v, y) = d_G(x, v) + d_G(u, y) + 2d_G(u, v)$, we have $S_3 > S_1$. Then, either $S_2 = S_3$ or $S_1 = S_2$ and $S_3 - S_1 \leq 2$. If the latter is true, then $2 \geq S_3 - S_1 = d_G(x, v) + d_G(u, y) + 2d_G(u, v) - d_G(x, v) - d_G(u, y) = 2d_G(u, v) > 2$ and a contradiction arises. Thus, $S_2 = S_3$ and we get $d_G(x, y) = d_G(x, v) + d_G(v, u) + d_G(u, y)$. \square

Lemma 1 *Let x, y be a diametral pair of vertices of a distance-hereditary graph G , and k be the minimum eccentricity of a shortest path in G . If for some shortest path $P = P(x, y)$, connecting x and y , $\text{ecc}(P) > k$ holds, then $\text{diam}(G) = d_G(x, y) \geq 2k$. Furthermore, if $d_G(x, y) = 2k$ then there is a shortest path P^* between x and y with $\text{ecc}(P^*) = k$.*

Proof: Consider a vertex v with $d_G(v, P) > k$. Let x' be a vertex of $\text{Pr}(v, P)$ closest to x , and y' be a vertex of $\text{Pr}(v, P)$ closest to y . By Corollary 1, $d_G(x', y') \leq 2$ and there must be a vertex v' in G adjacent to both x' and y' and at distance $d_G(v, P) - 1$ from v . Let $P(x, x')$ and $P(y', y)$ be subpaths of P connecting vertices x, x' and vertices y, y' , respectively. Consider also an arbitrary shortest path $Q(v, v')$ connecting v and v' in G . By choices of x' and y' , no chords in G exist in paths $P(x, x') \cup Q(v', v)$ and $P(y, y') \cup Q(v', v)$. Hence, those paths are shortest in G . Since x, y is a diametral pair, we have $d_G(x, x') + d_G(x', y') + d_G(y', y) = d_G(x, y) \geq d_G(x, v) = d_G(x, x') + 1 + d_G(v', v)$. That is, $d_G(y', y) \geq d_G(v', v) + 1 - d_G(x', y')$. Similarly, $d_G(x', x) \geq d_G(v', v) + 1 - d_G(x', y')$. Combining both inequalities and taking into account that $d_G(v, v') \geq k$, we get $d_G(x, y) = d_G(x, x') + d_G(x', y') + d_G(y', y) \geq 2k + 2 - d_G(x', y') \geq 2k$. Furthermore, we have $d_G(x, y) \geq 2k + 1$ if $d_G(x', y') = 1$ and $d_G(x, y) \geq 2k + 2$ if $d_G(x', y') = 0$. Also, if $d_G(x, y) = 2k$ then $d_G(x', y') = 2$, $d_G(v, v') = k$, $d_G(x, x') = d_G(y, y') = k - 1$ and $d_G(v, x) = d_G(v, y) = 2k$.

Now assume that $d_G(x, y) = 2k$. Consider sets $S = \{w \in V \mid d_G(x, w) = d_G(y, w) = k\}$ and $F_{x,y} = \{u \in V \mid d_G(u, x) = d_G(u, y) = 2k\}$. Let $c \in S$ be a vertex of S that k -dominates the maximum number of vertices in $F_{x,y}$. Consider a shortest path P^* connecting vertices x and y and passing through vertex c . We will show that $\text{ecc}(P^*) = k$. Let x' (y') be the neighbor of c in subpath of P^* connecting c with x (with y , respectively).

Assume there is a vertex v in G such that $d_G(v, P^*) > k$. As in the first part of the proof, one can show that $d_G(v, x') = d_G(v, y') = k + 1$, i. e., $x', y' \in \text{Pr}(v, P^*)$ and $d_G(v, P^*) = k + 1$. Furthermore, $d_G(v, x) = d_G(v, y) = 2k$, i. e., $v \in F_{x,y}$. Also, vertex v' , that is adjacent to x', y' and at distance k from v , must belong to S . Since $d_G(v, c) > k$ but $d_G(v, v') = k$, by choice of c , there must exist a vertex $u \in F_{x,y}$ such that $d_G(u, c) \leq k$ and $d_G(u, v') > k$. Since $d_G(u, y) = d_G(u, x) = 2k$, $d_G(u, c)$ must equal k and both $d_G(u, x')$ and $d_G(u, y')$ must equal $k + 1$.

Since $d_G(v, u) \leq \text{diam}(G) = 2k$ and $d_G(v, y') = d_G(v, x') = k + 1 = d_G(u, x') = d_G(u, y')$, we must have a chord between vertices of a shortest path $P(v, v')$ connecting v with v' and vertices of a shortest path $P(u, c)$ connecting u with c . If no chords exist or only chord cv' is present, then $d_G(v, u) \geq 2k + 1$, contradicting with $\text{diam}(G) = 2k$. So, consider a chord ab with $a \in P(v, v')$, $b \in P(u, c)$, $ab \neq cv'$, and $d_G(a, v') + d_G(b, c)$ is minimum. We know that $d_G(a, v') = d_G(b, c)$ must hold since $d_G(u, v') > k = d_G(u, c)$ and $d_G(v, c) > k = d_G(v, v')$. To avoid induced cycles of length $k \geq 5$, $d_G(a, v') = d_G(b, c) = 1$ must hold. But then, vertices a, b, c, x', v' form either an induced cycle C_5 , when c and v' are not adjacent, or a house, otherwise. Note that, by distance requirements, edges $bv', ca, bx',$ and ax' are not possible.

Contradictions obtained show that such a vertex v with $d_G(v, P^*) > k$ is not possible, i. e., $\text{ecc}(P^*) = k$. \square

Lemma 2 *In every distance-hereditary graph there is a minimum eccentricity shortest path $P(s, t)$ where s and t are two mutually furthest vertices.*

Proof: Let k be the minimum eccentricity of a shortest path in G . Let $Q := Q(s, t) = (s = v_0, v_1, \dots, v_i, \dots, v_q = t)$ be a shortest path of G of eccentricity k with maximum q , that is, among all shortest paths with eccentricity k , Q is a longest one. Assume, without loss of generality, that t is not a vertex most distant from s . Let $i \leq q$ be the smallest index such that subpath $Q(s, v_i) = (v_0, v_1, \dots, v_i)$ of Q has also the eccentricity k . By choice of i , there must exist a vertex v in G which is k -dominated only by vertex v_i of $Q(s, v_i)$, i. e., $\text{Pr}(v, Q(s, v_i)) = \{v_i\}$ and $d_G(v, Q(s, v_i)) = k$. Let $P(v, v_i)$ be an arbitrary shortest path of G connecting v with v_i . By choice of i , no vertex of $P(v, v_i) \setminus \{v_i\}$ is adjacent to a vertex of $Q(s, v_i) \setminus \{v_i\}$. Hence, path obtained by concatenating $Q(s, v_i)$ with $P(v_i, v)$ is chordless and, therefore, shortest in G , and has eccentricity k , too. Note that v is now a most distant vertex from s , i. e., $d_G(s, v) = \text{ecc}(s)$. Since $d_G(s, v) > d_G(s, t)$, a contradiction with maximality of q arises. \square

The main result of this section is the following.

Theorem 1 *Let x, y be a diametral pair of vertices of a distance-hereditary graph G , and k be the minimum eccentricity of a shortest path in G . Then, there is a shortest path P between x and y with $\text{ecc}(P) = k$.*

Proof: We may assume that for some shortest path P' connecting x and y , $\text{ecc}(P') > k$ holds (otherwise, there is nothing to prove). Then, by Lemma 1, we have $d_G(x, y) \geq 2k$.

Let $Q := Q(s, t) = (s = v_0, v_1, \dots, v_i, \dots, v_q = t)$ be a shortest path of G of eccentricity k such that s and t are two mutually furthest vertices (see Lemma 2). Consider projections of x and y to Q . We distinguish between three cases: $\text{Pr}(x, Q)$ is completely on the left of $\text{Pr}(y, Q)$ in Q ; $\text{Pr}(x, Q)$ and $\text{Pr}(y, Q)$ have a common vertex w ; and the remaining case (see Corollary 1) when $\text{Pr}(x, Q) = \{v_{i-1}, v_{i+1}\}$ and $\text{Pr}(y, Q) = \{v_i\}$ for some index i .

Case 1: $\text{Pr}(x, Q)$ is completely on the left of $\text{Pr}(y, Q)$ in Q .

Let x' be a vertex of $\text{Pr}(x, Q)$ closest to t and y' a vertex of $\text{Pr}(y, Q)$ closest to s . Consider an arbitrary shortest path $P(x, x')$ of G connecting vertices x and x' , an arbitrary shortest path $P(y', y)$ of G connecting vertices y' and y , and a subpath $Q(x', y')$ of $Q(s, t)$ between vertices x' and y' . We claim that the path P of G obtained by concatenating $P(x, x')$ with $Q(x', y')$ and then with $P(y', y)$ is a shortest path of eccentricity k .

Indeed, by choice of x' , no edge connecting a vertex in $P(x, x') \setminus \{x'\}$ with a vertex in $Q(x', y') \setminus \{x'\}$ can exist in G . Similarly, no edge connecting a vertex in $P(y', y) \setminus \{y'\}$ with a vertex in $Q(x', y') \setminus \{y'\}$ can exist in G . Since we also have $d_G(x, y) \geq 2k$, $d_G(x, Q) \leq k$ and $d_G(y, Q) \leq k$, no edge connecting a vertex in $P(y', y) \setminus \{y'\}$ with a vertex in $P(x, x') \setminus \{x'\}$ can exist in G . Hence, chordless path $P = P(x, x') \cup Q(x', y') \cup P(y', y)$ is a shortest path of G .

Consider now an arbitrary vertex v of G . We want to show that $d_G(v, P) \leq k$. Since $\text{ecc}(Q) = k$, $d_G(v, Q) \leq k$. Consider the projection of v to Q . We may assume that $\text{Pr}(v, Q) \cap Q(x', y') = \emptyset$ and, without loss of generality, that vertices of $\text{Pr}(v, Q)$ are closer to s than vertex x' . Let v' be a vertex of $\text{Pr}(v, Q)$ closest to x' . As before, by choices of v' and y' , paths $P(y, y') \cup Q(y', v')$ and $P(v, v') \cup Q(y', v')$ are chordless and, therefore, are shortest paths of G (here $P(v, v')$ is an arbitrary shortest path of G connecting v with v'). Since $d_G(v', y') \geq 2$, by Corollary 2, $d_G(v, y) = d_G(v, v') + d_G(v', y') + d_G(y', y)$. Hence, from $d_G(x, y) \geq d_G(y, v)$, $d_G(x, y) = d_G(x, x') + d_G(x', y)$ and $d_G(v, y) = d_G(v, x') + d_G(x', y)$, we obtain $d_G(v, x') \leq d_G(x, x') \leq k$.

Case 2: $\text{Pr}(x, Q)$ and $\text{Pr}(y, Q)$ have a common vertex w .

In this case, we have $d_G(x, y) \leq d_G(x, w) + d_G(y, w) \leq k + k = 2k$. Earlier we assumed also that $d_G(x, y) \geq 2k$. Hence, $\text{diam}(G) = d_G(x, y) = 2k$ and the statement of the theorem follows from Lemma 1.

Case 3: Remaining case when $\text{Pr}(x, Q) = \{v_{i-1}, v_{i+1}\}$ and $\text{Pr}(y, Q) = \{v_i\}$ for some index i .

In this case, we have $d_G(x, y) \leq d_G(x, v_{i-1}) + 1 + d_G(v_i, y) \leq 2k + 1$. By Lemma 1, we can assume that $\text{diam}(G) = d_G(x, y) = 2k + 1$, i. e., $d_G(x, v_{i-1}) = d_G(x, v_{i+1}) = d_G(v_i, y) = k$.

Let $Q(s, v_{i-1})$ and $Q(t, v_{i+1})$ be subpaths of Q connecting vertices s and v_{i-1} and vertices t and v_{i+1} , respectively. Pick an arbitrary shortest path $P(y, v_i)$ connecting y with v_i . Since no chords are possible between $Q(s, v_i) \setminus \{v_i\}$ and $P(y, v_i) \setminus \{v_i\}$ and between $Q(t, v_i) \setminus \{v_i\}$ and $P(y, v_i) \setminus \{v_i\}$, we have $d_G(y, t) = d_G(y, v_i) + d_G(v_i, t) = k + d_G(v_i, t)$ and $d_G(y, s) = d_G(y, v_i) + d_G(v_i, s) =$

$k + d_G(v_i, s)$. Inequalities $d_G(x, y) \geq d_G(y, t)$ and $d_G(x, y) \geq d_G(y, s)$ imply $d_G(v_{i+1}, t) \leq d_G(v_{i+1}, x) = k$ and $d_G(v_{i-1}, s) \leq d_G(v_{i-1}, x) = k$. If both $d_G(v_{i+1}, t)$ and $d_G(v_{i-1}, s)$ equal k , then $d_G(s, t) = 2k + 2$ contradicting with $\text{diam}(G) = 2k + 1$. Hence, we may assume, without loss of generality, that $d_G(v_{i-1}, s) \leq k - 1$. We will show that shortest path $P := P(x, v_{i+1}) \cup P(v_i, y)$ has eccentricity k (here, $P(x, v_{i+1})$ is an arbitrary shortest path of G connecting x with v_{i+1}).

Consider a vertex v in G and assume that $\text{Pr}(v, Q)$ is strictly contained in $Q(t, v_{i+1})$. Denote by v' the vertex of $\text{Pr}(v, Q)$ that is closest to s . Let $P(v, v')$ be an arbitrary shortest path connecting v and v' . As before, $P(v, v') \cup Q(v', s)$ is a chordless path and therefore $d_G(v, s) = d_G(v, v_{i+1}) + d_G(v_{i+1}, s)$. Since t is a most distant vertex from s , $d_G(s, v) \leq d_G(s, t)$. Hence, $d_G(v, v_{i+1}) + d_G(v_{i+1}, s) = d_G(s, v) \leq d_G(s, t) = d_G(s, v_{i+1}) + d_G(v_{i+1}, t)$, i. e., $d_G(v, v_{i+1}) \leq d_G(v_{i+1}, t) \leq k$.

Consider a vertex v in G and assume now that $\text{Pr}(v, Q)$ is strictly contained in $Q(s, v_{i-1})$. Denote by v' the vertex of $\text{Pr}(v, Q)$ that is closest to t . Let $P(v, v')$ be an arbitrary shortest path connecting v and v' . Again, $P(v, v') \cup Q(v', t)$ is a chordless path and therefore $d_G(v, t) = d_G(v, v_i) + d_G(v_i, t)$. Since s is a most distant vertex from t , $d_G(t, v) \leq d_G(s, t)$. Hence, $d_G(v, v_i) + d_G(v_i, t) = d_G(t, v) \leq d_G(s, t) = d_G(s, v_i) + d_G(v_i, t)$, i. e., $d_G(v, v_i) \leq d_G(v_i, s) \leq k$.

Thus, all vertices of G are k -dominated by $P(x, v_{i+1}) \cup P(v_i, y)$. \square

It is known [13] that a diametral pair of a distance-hereditary graph can be found in linear time. Hence, according to Theorem 1, to find a shortest path of minimum eccentricity in a distance-hereditary graph in linear time, one needs to efficiently extract a best eccentricity shortest path for a given pair of end-vertices. In what follows, we demonstrate that, for a distance-hereditary graph, such an extraction can be done in linear time as well.

We will need few auxiliary lemmas.

Lemma 3 *In a distance-hereditary graph G , for each pair of vertices s and t , if x is on a shortest path from v to $\Pi_v = \text{Pr}(v, I(s, t))$ and $d_G(x, \Pi_v) = 1$, then $\Pi_v \subseteq N(x)$.*

Proof: Let p and q be two vertices in Π_v and $d_G(v, \Pi_v) = r$. By statement (3) of Proposition 1, $N(p) \cap L_{r-1}^{(v)} = N(q) \cap L_{r-1}^{(v)}$. Thus, each vertex x on a shortest path from v to Π_v with $d_G(x, \Pi_v) = 1$ (which is in $N(p) \cap L_{r-1}^{(v)}$ by definition) is adjacent to all vertices in Π_v , i. e., $\Pi_v \subseteq N(x)$. \square

Lemma 4 *In a distance-hereditary graph G , let $S_i(s, t)$ and $S_{i+1}(s, t)$ be two consecutive slices of an interval $I(s, t)$. Each vertex in $S_i(s, t)$ is adjacent to each vertex in $S_{i+1}(s, t)$.*

Proof: Consider statement (3) of Proposition 1 from perspective of t . Thus, $S_i(s, t) \subseteq N(v)$ for each vertex $v \in S_{i+1}(s, t)$. Additionally, from perspective of s , $S_{i+1}(s, t) \subseteq N(u)$ for each vertex $u \in S_i(s, t)$. \square

Lemma 5 *In a distance-hereditary graph G , if a projection $\Pi_v = \text{Pr}(v, I(s, t))$ intersects two slices of an interval $I(s, t)$, each shortest (s, t) -path intersects Π_v .*

Proof: Because of Lemma 3, there is a vertex x with $N(x) \supseteq \Pi_v$ and $d_G(v, x) = d_G(v, \Pi_v) - 1$. Thus, Π_v intersects at most two slices of interval $I(s, t)$ and those slices have to be consecutive, otherwise x would be a part of the interval. Let $S_i(s, t)$ and $S_{i+1}(s, t)$ be these slices. Note that $d_G(s, x) = i + 1$. Thus, by statement (3) of Proposition 1, $N(x) \cap S_i(s, t) = N(u) \cap S_i(s, t)$ for each $u \in S_{i+1}(s, t)$. Therefore, $S_i(s, t) \subseteq \Pi_v$, i.e., each shortest path from s to t intersects Π_v . \square

From the lemmas above, we can conclude that, for determining a shortest (s, t) -path with minimal eccentricity, a vertex v is only relevant if $d_G(v, I(s, t)) = \text{ecc}(I(s, t))$ and the projection of v on the interval $I(s, t)$ only intersects one slice. Algorithm 1 uses this.

Algorithm 1: Computes a shortest (s, t) -path P with minimal eccentricity for a given distance-hereditary graph G and a vertex pair s, t .

Input: A distance-hereditary graph $G = (V, E)$ and two distinct vertices s and t .

Output: A shortest path P from s to t with minimal eccentricity.

- 1 Compute the sets $V_i = \{v \mid d_G(v, I(s, t)) = i\}$ for $1 \leq i \leq \text{ecc}(I(s, t))$.
 - 2 Each vertex $v \notin I(s, t)$ gets a pointer $g(v)$ initialised with $g(v) := v$ if $v \in V_1$, and $g(v) := \emptyset$ otherwise.
 - 3 **for** $i := 2$ **to** $\text{ecc}(I(s, t))$ **do**
 - 4 **[** For each $v \in V_i$, select a vertex $u \in V_{i-1} \cap N(v)$ and set $g(v) := g(u)$.
 - 5 **foreach** $v \in V_{\text{ecc}(I(s, t))}$ **do**
 - 6 **[** If $N(g(v))$ intersects only one slice of $I(s, t)$, flag $g(v)$ as *relevant*.
 - 7 Set $P := \{s, t\}$.
 - 8 **for** $i := 1$ **to** $d_G(s, t) - 1$ **do**
 - 9 **[** Find a vertex $v \in S_i(s, t)$ for which the number of *relevant* vertices in $N(v)$ is maximal.
 - 10 **[** Add v to P .
-

Lemma 6 *For a distance-hereditary graph G and an arbitrary vertex pair s, t , Algorithm 1 computes a shortest (s, t) -path with minimal eccentricity in linear time.*

Proof: The loop in line 3 determines a *gate vertex* $g(v)$ for each vertex v outside of the interval $I(s, t)$ such that $N(g(v)) \supseteq \text{Pr}(v, I(s, t))$ and $d_G(v, I(s, t)) = d_G(v, g(v)) + 1$ (see Lemma 3). From Lemma 5 and Lemma 4, it follows that for a vertex v which is not in $V_{\text{ecc}(I(s, t))}$ or its projection to $I(s, t)$ is intersecting two slices of $I(s, t)$, $d_G(v, P(s, t)) \leq \text{ecc}(I(s, t))$ for every shortest path $P(s, t)$ between s and t . Therefore, line 6 only marks $g(v)$ if $v \in V_{\text{ecc}(I(s, t))}$ and its

projection $\text{Pr}(v, I(s, t))$ intersects only one slice. Because only one slice is intersected and each vertex in a slice is adjacent to all vertices in the consecutive slice (see Lemma 4), in each slice the vertex of an optimal (of minimum eccentricity) path P can be selected independently from the preceding vertex. If a vertex x of a slice $S_i(s, t)$ has the maximum number of *relevant* vertices in $N(x)$, then x is good to put in P . Indeed, if x dominates all relevant vertices adjacent to vertices of $S_i(s, t)$, then x is a perfect choice to put in P . Else, any vertex y of a slice $S_i(s, t)$ is a good vertex to put in P . Hence, P is optimal if the number of *relevant* vertices adjacent to P is maximal. Thus, the path selected in line 8 to line 10 is optimal. \square

Running Algorithm 1 for a diametral pair of vertices of a distance-hereditary graph G , by Theorem 1, we get a shortest path of G with minimum eccentricity. Thus, we have proven the following result.

Theorem 2 *A shortest path with minimum eccentricity of a distance-hereditary graph $G = (V, E)$ can be computed in $\mathcal{O}(|V| + |E|)$ time.*

4 A Polynomial-Time Algorithm for Tree-Structured Graphs

4.1 Projection Gap

In a graph G , consider a shortest path P which starts in a vertex s . Each vertex x has a projection $\Pi_x = \text{Pr}(x, P)$. In case of a tree this is a single vertex. However, in general, Π_x can contain multiple vertices and does not necessarily induce a connected subgraph. In this case, there are two vertices u and w in Π_x such that all vertices v in the subpath Q between u and w are not in Π_x . Formally, $u, w \in \Pi_x$, $Q = \{v \in P \mid d_G(s, u) < d_G(s, v) < d_G(s, w)\}$, and $Q \cap \Pi_x = \emptyset$.

Now, assume the cardinality of Q is at most γ , i. e., $d_G(u, w) \leq \gamma + 1$ for each P, x, u and w . We refer to γ as the *projection gap* of G .

Definition 1 (Projection Gap) *In a graph G , let $P = \{v_0, \dots, v_l\}$ be a shortest path with $d_G(v_0, v_i) = i$. The projection gap of G is γ , $\text{pg}(G) = \gamma$ for short, if, for every vertex x of G and every two vertices $v_i, v_k \in \text{Pr}(x, P)$, $d_G(v_i, v_k) > \gamma + 1$ implies that there is a vertex $v_j \in \text{Pr}(x, P)$ with $i < j < k$.*

Based on this definition, we can make the following observation.

Lemma 7 *In a graph G with $\text{pg}(G) = \gamma$, let P be a shortest path starting in s , Q be a subpath of P , $|Q| > \gamma$, u and w be two vertices in $P \setminus Q$ such that $d_G(s, u) < d_G(s, Q) < d_G(s, w)$, and x be an arbitrary vertex in G . If $d_G(x, u) < d_G(x, Q)$, then $d_G(x, w) \geq d_G(x, Q)$.*

Proof: Assume that $d_G(x, u) < d_G(x, Q)$ and $d_G(x, w) < d_G(x, Q)$. Without loss of generality, let $d_G(x, u) = d_G(x, w) < d_G(x, v)$ for all $v \in P$ with

$d_G(s, u) < d_G(s, v) < d_G(s, w)$. Let P' be the subpath of P from u to w . Note that $\text{Pr}(x, P') = \{u, w\}$ and $Q \subset P'$. Thus, $d_G(u, w) \geq |Q| + 1 > \gamma + 1$. This contradicts with $\text{pg}(G) = \gamma$. \square

Informally, Lemma 7 says that, when exploring a shortest path P , if the distance to a vertex x did not decrease during the last $\gamma + 1$ vertices of P , it will not decrease when exploring the remaining subpath. Based on this, we will show that a minimum eccentricity shortest path can be found in polynomial time if $\text{pg}(G)$ is bounded by some constant.

For the rest of this section, we assume we are given a graph G with $\text{pg}(G) = \gamma$ containing a vertex s . We will need the following notions and notations:

- Q_i and Q_j are subpaths of length γ of some shortest paths starting in s . They do not need to be subpath of the same shortest path. Let $v_i \in Q_i$ and $v_j \in Q_j$ be the two vertices such that $d_G(s, Q_i) = d_G(s, v_i)$ and $d_G(s, Q_j) = d_G(s, v_j)$. Without loss of generality, let $d_G(s, v_i) \leq d_G(s, v_j)$. We say, Q_i is *compatible* with Q_j (with respect to s) if $|Q_i \cap Q_j| = \gamma - 1$, v_i is adjacent to v_j , and $d_G(s, v_i) < d_G(s, v_j)$. Let $\mathcal{C}_s(Q_j)$ denote the set of subpaths compatible with Q_j .
- $R_s(Q_j) = \{w \mid Q_j \subseteq I(s, w)\} \cup Q_j$ is the set of vertices w such that there is a shortest path from s to w containing Q_j (or $w \in Q_j$).
- $I(s, Q_j) = I(s, v_j) \cup Q_j$ are the vertices that are on a shortest path from s to Q_j (or in Q_j).
- $V_s^\downarrow(Q_j) = \{x \mid d_G(x, Q_j) = d_G(x, R_s(Q_j))\}$ is the set of vertices x which are closer to Q_j than to all other vertices in $R_s(Q_j)$. Thus, given a shortest path P containing Q_j and starting in s , expanding P beyond Q_j will not decrease the distance from x to P .

Note that $Q_j \subseteq V_s^\downarrow(Q_j)$ and $Q_j = I(s, Q_j) \cap R_s(Q_j)$.

4.2 Algorithm

Lemma 8 For each vertex x in G , $d_G(x, Q_j) = d_G(x, I(s, Q_j))$ or $d_G(x, Q_j) = d_G(x, R_s(Q_j))$.

Proof: Assume, $d_G(x, Q_j) > d_G(x, I(s, Q_j))$ and $d_G(x, Q_j) > d_G(x, R_s(Q_j))$. Then, there is a vertex $u_i \in I(s, Q_j)$ and a vertex $u_r \in R_s(Q_j)$ with $d_G(x, u_i) < d_G(x, Q_j)$ and $d_G(x, Q_j) > d_G(x, u_r)$. Because u_i , Q_j , and u_r are on a shortest path starting in s and $|Q_j| > \gamma$, this contradicts Lemma 7. \square

Lemma 9 If Q_i is compatible with Q_j , then $V_s^\downarrow(Q_i) \subseteq V_s^\downarrow(Q_j)$.

Proof: Assume that $V_s^\downarrow(Q_i) \not\subseteq V_s^\downarrow(Q_j)$, i.e., there is a vertex $x \in V_s^\downarrow(Q_i) \setminus V_s^\downarrow(Q_j)$. Then, $d_G(x, Q_j) > d_G(x, R_s(Q_j))$. Thus, by Lemma 8, $d_G(x, Q_j) = d_G(x, I(s, Q_j))$. Because $Q_i \subseteq I(s, Q_j)$, $d_G(x, Q_i) \geq d_G(x, I(s, Q_j)) = d_G(x, Q_j)$. Since $x \in V_s^\downarrow(Q_i)$, $d_G(x, Q_i) = d_G(x, R_s(Q_i))$. Also, because $x \notin V_s^\downarrow(Q_i)$,

$d_G(x, Q_j) > d_G(x, R_s(Q_j))$. Thus, $d_G(x, R_s(Q_i)) > d_G(x, R_s(Q_j))$. On the other hand, because $R_s(Q_i) \supseteq R_s(Q_j)$, $d_G(x, R_s(Q_i)) \leq d_G(x, R_s(Q_j))$, and a contradiction arises. \square

For a subpath Q_j , let $\mathcal{P}_s(Q_j)$ denote the set of shortest paths P which start in s such that $Q_j \subseteq P \subseteq I(s, Q_j)$. Then, we define $\varepsilon_s(Q_j)$ as follows:

$$\varepsilon_s(Q_j) = \min_{P \in \mathcal{P}_s(Q_j)} \max_{x \in V_s^\downarrow(Q_j)} d_G(x, P)$$

Consider a subpath Q_j for which $R_s(Q_j) = Q_j$, i. e., a shortest path containing Q_j cannot be extended any more. Then, $V_s^\downarrow(Q_j) = V$. Therefore, for any path $P \in \mathcal{P}_s(Q_j)$, $\max_{x \in V_s^\downarrow(Q_j)} d_G(x, P) = \text{ecc}(P)$.

Lemma 10 *If $\mathcal{C}_s(Q_j)$ is not empty, then*

$$\varepsilon_s(Q_j) = \min_{Q_i \in \mathcal{C}_s(Q_j)} \max \left[\max_{x \in V_s^\downarrow(Q_j) \setminus V_s^\downarrow(Q_i)} \min(d_G(x, Q_i), d_G(x, Q_j)), \varepsilon_s(Q_i) \right].$$

Proof: By definition,

$$\varepsilon_s(Q_j) = \min_{P \in \mathcal{P}_s(Q_j)} \max_{x \in V_s^\downarrow(Q_j)} d_G(x, P).$$

Let Q_i be compatible with Q_j . Because, by Lemma 9, $V_s^\downarrow(Q_i) \subseteq V_s^\downarrow(Q_j)$, we can partition $V_s^\downarrow(Q_j)$ into $V_s^\downarrow(Q_j) \setminus V_s^\downarrow(Q_i)$ and $V_s^\downarrow(Q_i)$. For simplicity, we will write $V_s^\downarrow(Q_j) \setminus V_s^\downarrow(Q_i)$ as $V_s^\downarrow[Q_j|Q_i]$. Thus, $\varepsilon_s(Q_j) =$

$$\min_{Q_i \in \mathcal{C}_s(Q_j)} \min_{P \in \mathcal{P}_s(Q_i)} \max \left[\max_{x \in V_s^\downarrow[Q_j|Q_i]} d_G(x, P \cup Q_j), \max_{x \in V_s^\downarrow(Q_i)} d_G(x, P \cup Q_j) \right].$$

Note that we changed the definition of P from $P \in \mathcal{P}_s(Q_j)$ to $P \in \mathcal{P}_s(Q_i)$, i. e., P may not contain the last vertex of Q_j any more.

If $x \in V_s^\downarrow[Q_j|Q_i]$, then $d_G(x, Q_i) > d_G(x, R_s(Q_i))$. Thus, by Lemma 8, $d_G(x, Q_i) = d_G(x, I(s, Q_i))$. By definition of P , $d_G(x, Q_i) \geq d_G(x, P) \geq d_G(x, I(s, Q_i))$. Therefore, $d_G(x, P) = d_G(x, Q_i)$ and

$$\max_{x \in V_s^\downarrow[Q_j|Q_i]} d_G(x, P \cup Q_j) = \max_{x \in V_s^\downarrow[Q_j|Q_i]} \min(d_G(x, Q_i), d_G(x, Q_j)).$$

For simplicity, we define

$$\varepsilon_s(Q_i, Q_j) := \max_{x \in V_s^\downarrow[Q_j|Q_i]} \min(d_G(x, Q_i), d_G(x, Q_j)).$$

Note that $\varepsilon_s(Q_i, Q_j)$ does not depend on P . Thus, because $\min_u \max[c, f(u)] = \max[c, \min_u f(u)]$,

$$\varepsilon_s(Q_j) = \min_{Q_i \in \mathcal{C}_s(Q_j)} \max \left[\varepsilon_s(Q_i, Q_j), \min_{P \in \mathcal{P}_s(Q_i)} \max_{x \in V_s^\downarrow(Q_i)} d_G(x, P \cup Q_j) \right].$$

If $x \in V_s^\downarrow(Q_i)$, then $d_G(x, Q_i) = d_G(x, R_s(Q_i)) \leq d_G(x, R_s(Q_j)) = d_G(x, Q_j)$. Therefore,

$$\min_{P \in \mathcal{P}_s(Q_i)} \max_{x \in V_s^\downarrow(Q_i)} d_G(x, P \cup Q_j) = \min_{P \in \mathcal{P}_s(Q_i)} \max_{x \in V_s^\downarrow(Q_i)} d_G(x, P) = \varepsilon_s(Q_i).$$

Thus,

$$\varepsilon_s(Q_j) = \min_{Q_i \in \mathcal{C}_s(Q_j)} \max \left[\max_{x \in V_s^\downarrow(Q_j) \setminus V_s^\downarrow(Q_i)} \min(d_G(x, Q_i), d_G(x, Q_j)), \varepsilon_s(Q_i) \right].$$

□

Based on Lemma 10, Algorithm 2 computes a shortest path starting in s with minimal eccentricity. The algorithm has two parts. First, it computes the pairwise distance of all vertices and $d_G(x, R_s(v))$ for each vertex pair x and v where, similarly to $R_s(Q_j)$, $R_s(v) = \{z \in V \mid v \in I(s, z)\}$. This allows to easily determine if a vertex x is in $V_s^\downarrow(Q_j)$. Second, it computes $\varepsilon_s(Q_j)$ for each subpath Q_j . For this, the algorithm uses dynamic programming. After calculating $\varepsilon_s(Q_i)$ for all subpaths with distance i to s , the algorithm uses Lemma 10 to calculate $\varepsilon_s(Q_j)$ for all subpaths Q_j which Q_i is compatible with.

Theorem 3 *For a given graph G with $\text{pg}(G) = \gamma$ and a vertex s , Algorithm 2 computes a shortest path starting in s with minimal eccentricity. It runs in $\mathcal{O}(n^{\gamma+3})$ time if $\gamma \geq 2$, in $\mathcal{O}(n^2m)$ time if $\gamma = 1$, and in $\mathcal{O}(nm)$ time if $\gamma = 0$.*

Proof: [Correctness] The algorithm has two parts. The first part (line 1 to line 8) is a preprocessing which computes $d_G(x, R_s(v))$ for each vertex pair x and v . The second part computes $\varepsilon_s(Q_j)$ which is then used to determine a path with minimal eccentricity.

For the first part, without loss of generality, let $d_G(s, v) = i$, $N_s^\uparrow(v) = N(v) \cap L_{i+1}^{(s)}$, and let x be an arbitrary vertex. By definition of R_s , $N_s^\uparrow(v) = \emptyset$ implies $R_s(v) = \{v\}$, i. e., $d_G(x, R_s(v)) = d_G(x, v)$. Therefore, $d_G(x, R_s(v))$ is correct for all vertices v with $N_s^\uparrow(v) = \emptyset$ after line 3. By induction, assume that $d_G(x, R_s(w))$ is correct for all vertices $w \in N_s^\uparrow(v)$. Because $R_s(v) = \bigcup_{w \in N_s^\uparrow(v)} R_s(w) \cup \{v\}$, $d_G(x, R_s(v)) = \min(\min_{w \in N_s^\uparrow(v)} d_G(x, R_s(w)), d_G(x, v))$. Therefore, line 8 correctly computes $d_G(x, R_s(v))$.

The second part of Algorithm 2 iterates over all subpaths Q_j in increasing distance to s . Line 12 checks if a given vertex x is in $V_s^\downarrow(Q_j)$. By definition, $R_s(Q_j) = Q_j \cup R_s(z_j)$ where z_j is the vertex in Q_j with the largest distance to s . Thus, $d_G(x, R_s(Q_j)) = \min(d_G(x, R_s(z_j)), d_G(x, Q_j))$. By definition of V_s^\downarrow , $x \in V_s^\downarrow(Q_j)$ if and only if $d_G(x, Q_j) = d_G(x, R_s(Q_j))$. Therefore, $x \in V_s^\downarrow(Q_j)$ if and only if $d_G(x, Q_j) \leq d_G(x, R_s(z_j))$, i. e., line 12 computes $V_s^\downarrow(Q_j)$ correctly.

Recall the definition of $\varepsilon_s(Q_j)$:

$$\varepsilon_s(Q_j) = \min_{P \in \mathcal{P}_s(Q_j)} \max_{x \in V_s^\downarrow(Q_j)} d_G(x, P)$$

Algorithm 2: Determines, for a given graph G with $\text{pg}(G) \leq \gamma$ and a vertex s , a minimal eccentricity shortest path starting in s .

Input: A graph $G = (V, E)$, an integer γ , and a vertex $s \in V$.

Output: A shortest path P starting in s with minimal eccentricity.

```

1 Determine the pairwise distances of all vertices.
2 foreach  $v, x \in V$  do
3    $\lfloor$  Set  $d_G(x, R_s(v)) := d_G(x, v)$ .
4 for  $i = \text{ecc}(s) - 1$  downto 0 do
5   foreach  $v \in L_i^{(s)}$  do
6     foreach  $w \in N(v) \cap L_{i+1}^{(s)}$  do
7       foreach  $x \in V$  do
8          $\lfloor$  Set  $d_G(x, R_s(v)) := \min [d_G(x, R_s(v)), d_G(x, R_s(w))]$ .
9 for  $j = 0$  to  $\text{ecc}(s) - \gamma$  do
10  foreach  $Q_j$  with  $d_G(s, Q_j) = j$  do
11    foreach  $x \in V$  do
12       $\lfloor$  Let  $z_j$  be the vertex in  $Q_j$  with the largest distance to  $s$ . If
13         $d_G(x, Q_j) \leq d_G(x, R_s(z_j))$ , add  $x$  to  $V_s^\downarrow(Q_j)$  and store
14         $d_G(x, Q_j)$ .
15      if  $j = 0$  then
16         $\lfloor$   $\varepsilon_s(Q_j) := \max_{x \in V_s^\downarrow(Q_j)} d_G(x, Q_j)$ 
17      else
18         $\lfloor$   $\varepsilon_s(Q_j) := \infty$ 
19      foreach  $Q_i \in \mathcal{C}_s(Q_j)$  do
20         $\lfloor$   $\varepsilon'_s(Q_j) := \max_{x \in V_s^\downarrow(Q_j) \setminus V_s^\downarrow(Q_i)} \min (d_G(x, Q_i), d_G(x, Q_j)), \varepsilon_s(Q_i)$ 
21        if  $\varepsilon'_s(Q_j) < \varepsilon_s(Q_j)$  then
22           $\lfloor$  Set  $\varepsilon_s(Q_j) := \varepsilon'_s(Q_j)$  and  $p(Q_j) := Q_i$ .
21 Find a subpath  $Q_j$  such that a shortest path containing  $Q_j$  cannot be
    extended any more and for which  $\varepsilon_s(Q_j)$  is minimal.
22 Construct a path  $P$  from  $Q_j$  to  $s$  using the  $p()$ -pointers and output it.

```

If $d_G(s, Q_j) = 0$, $d_G(x, P) = d_G(x, Q_j)$. Thus, $\varepsilon_s(Q_j) = \max_{x \in V_s^\downarrow(Q_j)} d_G(x, Q_j)$ as computed in line 14. Note that there is no subpath Q_i which is compatible with Q_j , if $d_G(s, Q_j) = 0$. Therefore, the loop starting in line 17 is skipped for these Q_j . Thus, the algorithm correctly computes $\varepsilon_s(Q_j)$, if $d_G(s, Q_j) = 0$.

By induction, assume that $\varepsilon_s(Q_i)$ is correct for each $Q_i \in \mathcal{C}(Q_j)$. Thus, Lemma 10 can be used to compute $\varepsilon_s(Q_j)$. This is done in the loop starting in

line 17. Therefore, at the beginning of line 21, $\varepsilon_s(Q_j)$ is computed correctly for each subpath Q_j .

Recall, if $P \in \mathcal{P}(Q_j)$ and $R_s(Q_j) = Q_j$, then $V_s^\downarrow(Q_j) = V$ and, therefore, $\max_{x \in V_s^\downarrow(Q_j)} d_G(x, P) = \text{ecc}(P)$. Thus, $R_s(Q_j) = Q_j$ implies that $\varepsilon_s(Q_j)$ is the minimal eccentricity of all shortest paths starting in s and containing Q_j . Therefore, if Q_j is picked by line 21, then $\varepsilon_s(Q_j)$ is the minimal eccentricity of all shortest paths starting in s . \square

Proof: [Complexity] First, we will analyse line 1 to line 8. Line 1 runs in $\mathcal{O}(nm)$ time. This allows to access the distance between two vertices in constant time. Thus, the total running time for line 3 is $\mathcal{O}(n)$. Because line 8 is called at most once for each vertex x and edge vw , implementing line 4 to line 8 can be done in $\mathcal{O}(nm)$ time.

For the second part of the algorithm (starting in line 9), if $\gamma \geq 2$, let all subpaths be stored in a trie as follows: There are $\gamma + 1$ layers of internal nodes. Each internal node is an array of size n (one entry for each vertex) and each entry points to an internal node of the next layer representing n subtrees. This requires $\mathcal{O}(n^{\gamma+1})$ memory. Leafs are objects representing a subpath.

If $\gamma = 1$, a subpath is a single edge, and, if $\gamma = 0$, a subpath is a single vertex. Thus, no extra data structure is needed for these cases. In all three cases, a subpath can be accessed in $\mathcal{O}(\gamma)$ time.

Next, we analyse the runtime of line 11 to line 16 for a single subpath Q_j . Accessing Q_j can be done in $\mathcal{O}(\gamma)$ time. Line 12 requires at most $\mathcal{O}(\gamma)$ time for a single call and is called at most $\mathcal{O}(n)$ times. Line 14 requires $\mathcal{O}(n\gamma)$ time and line 16 runs in constant time. Therefore, for a given subpath, line 11 to line 16 require $\mathcal{O}(\gamma n + n)$ time.

For line 18 to line 20, consider a given pair of compatible subpaths Q_i and Q_j . Accessing both subpaths can be done in $\mathcal{O}(\gamma)$ time. Assuming the vertices in $V_s^\downarrow(Q_i)$ and $V_s^\downarrow(Q_j)$ are sorted and stored with their distance to Q_i and Q_j , line 18 requires at most $\mathcal{O}(n)$ time. Note that Q_i and Q_j intersect in $\gamma - 1$ vertices. Thus, $\min(d_G(x, Q_i), d_G(x, Q_j)) = \min(d_G(x, v_i), d_G(x, Q_j))$ where v_i is the vertex in Q_i closest to s . Line 19 and line 20 run in constant time. Therefore, for a given pair of compatible subpaths, line 18 to line 20 require $\mathcal{O}(n)$ time.

Let ϕ be the number of subpaths and ψ be the number of pairs of compatible subpaths. Then, the overall runtime for line 9 to line 20 is $\mathcal{O}(\phi(\gamma n + n) + \psi n)$ time, $\mathcal{O}(\phi)$ time for line 21, and $\mathcal{O}(n)$ time for line 22. Together with the first part of the algorithm, the total runtime of Algorithm 2 is $\mathcal{O}(mn + \phi(\gamma n + n) + \psi n)$.

Because a subpath contains $\gamma + 1$ vertices, there are up to $\mathcal{O}(n^{\gamma+1})$ subpaths and up to $\mathcal{O}(n^{\gamma+2})$ compatible pairs if $\gamma \geq 2$, i. e., $\phi \leq n^{\gamma+1}$ and $\psi \leq n^{\gamma+2}$. Therefore, if $\gamma \geq 2$, Algorithm 2 runs in $\mathcal{O}(n^{\gamma+3})$ time.

If $\gamma = 1$, a subpath is a single edge and there are at most mn compatible pairs of subpaths, i. e., $\phi \leq m$ and $\psi \leq nm$. For the case when $\gamma = 0$, a subpath is a single vertex ($\phi \leq n$) and a pair of compatible subpaths is an edge ($\psi \leq m$). Therefore, Algorithm 2 runs in $\mathcal{O}(n^2 m)$ time if $\gamma = 1$, and in $\mathcal{O}(nm)$ time if $\gamma = 0$. \square

Note that Algorithm 2 computes a shortest path starting in a given vertex s . Thus, a shortest path with minimum eccentricity among all shortest paths in G can be determined by running Algorithm 2 for all start vertices s , resulting in the following:

Theorem 4 *For a given graph G with $\text{pg}(G) = \gamma$, a minimum eccentricity shortest path can be found in $\mathcal{O}(n^{\gamma+4})$ time if $\gamma \geq 2$, in $\mathcal{O}(n^3m)$ time if $\gamma = 1$, and $\mathcal{O}(n^2m)$ time if $\gamma = 0$.*

4.3 Projection Gap for some Graph Classes

Above, we have shown that a minimum eccentricity shortest path can be found in polynomial time if the projection gap is bounded by a constant. In this subsection, we will determine the projection gap for some graph classes.

4.3.1 Chordal Graphs and Dually Chordal Graphs.

The class of chordal graphs is a well known class which can be recognised in linear time [22]. Due to the strong tree structure of chordal graphs, they have the following property known as m -convexity:

Lemma 11 ([14]) *Let G be a chordal graph. If, for two distinct vertices u, v in a disk $N^r[x]$, there is a path P connecting them with $P \cap N^r[x] = \{u, v\}$, then u and v are adjacent.*

Lemma 12 *If G is a chordal graph, then $\text{pg}(G) = 0$.*

Proof: Assume $\text{pg}(G) \geq 1$. Then, there is a shortest path $P = \{u, \dots, w\}$ and a vertex x with $\text{Pr}(x, P) = \{u, w\}$ and $d_G(u, w) > 1$. By Lemma 11, u and w are adjacent. This contradicts with $d_G(u, w) > 1$. \square

Corollary 3 *For chordal graphs, a minimum eccentricity shortest path can be found in $\mathcal{O}(n^2m)$ time.*

Dually chordal graphs were introduced in [3]. They are closely related to chordal graphs.

Lemma 13 *If G is a dually chordal graph, then $\text{pg}(G) \leq 1$.*

Proof: Assume there is a shortest path $P = \{u, v_1, \dots, v_i, w\}$ and a vertex x with $\text{Pr}(x, P) \supseteq \{u, w\}$. To show that $\text{pg}(G) \leq 1$, we will show that $d_G(u, w) = i + 1 > 2$ implies there is a vertex $v_k \in \text{Pr}(x, P)$ with $1 \leq k \leq i$.

Consider a family of disks $\mathcal{D} = \{N[u], N[v_1], \dots, N[v_i], N[w], N^r[x]\}$ where $r = d_G(x, P) - 1$. Let H be the intersection graph of \mathcal{D} , a be the vertex in H representing $N[u]$, b_k representing $N[v_k]$ (for $1 \leq k \leq i$), c representing $N[w]$, and z representing $N^r[x]$. Because the intersection graph of disks of a dually chordal graph is chordal [3], H is chordal, too. H contains the edges za and zc , ab_1 , cb_i , and $b_k b_{k+1}$ for all $1 \leq k < i$. Note that, if $d_G(u, w) > 2$, a and c are not

adjacent in H . However, the path $\{a, b_1, \dots, b_k, c\}$ connects a and c . Therefore, because H is chordal and by Lemma 11, there is a k with $1 \leq k \leq i$ such that z is adjacent to b_k in H . Thus, $d_G(x, v_k) \leq r + 1$, i. e., $v_k \in \text{Pr}(x, P)$. \square

Corollary 4 *For dually chordal graphs, a minimum eccentricity shortest path can be found in $\mathcal{O}(n^3m)$ time.*

4.3.2 Graphs with bounded Tree-Length or Tree-Breadth.

As defined by ROBERTSON and SEYMOUR [20], a *tree-decomposition* of a graph $G = (V, E)$ is a tree \mathcal{T} with the vertex set \mathcal{B} where each vertex of \mathcal{T} , called bag, is a subset of V such that: (i) $V = \bigcup_{B \in \mathcal{B}} B$, (ii) for each edge $uv \in E$, there is a bag $B \in \mathcal{B}$ with $u, v \in B$, and (iii) for each vertex $v \in V$, the bags containing v induce a subtree of \mathcal{T} .

The *length* of a tree decomposition is smaller than or equal to λ if for each bag B , $\text{diam}_G(B) \leq \lambda$. A graph G has *tree-length* λ , if there exist a tree-decomposition \mathcal{T} for G such that \mathcal{T} has length λ . Similarly, the *breadth* of a tree decomposition is smaller than or equal to ρ if for each bag B there is a vertex $v \in V$ with $N^\rho[v] \supseteq B$. A graph G has *tree-breadth* ρ , if there exist a tree-decomposition \mathcal{T} for G such that \mathcal{T} has breadth λ .

For these graphs, we use a concept called *layering partition*. It was introduced in [2, 5]. The idea is to first partition the vertices of a given graph in distance layers $L_i^{(x)}$ with respect to a given vertex x . Second, partition each layer $L_i^{(x)}$ into *clusters* in such a way that two vertices u and v are in the same cluster if they are connected by a path P such that $d_G(x, P) = d_G(x, u)$, i. e., P does not contain vertices of layers closer to x than u and v .

Unfortunately, computing the tree-length of a graph is an NP-hard problem [18]. However, for our needs, an approximation of it would suffice.

Lemma 14 *If G has tree-length λ or tree-breadth ρ , a factor $\gamma \geq \text{pg}(G)$ can be computed in $\mathcal{O}(n^3)$ time such that $\gamma \leq 3\lambda - 1$ or $\gamma \leq 6\rho - 1$, respectively.*

Proof: To compute γ , first determine the pairwise distances of all vertices. Then, compute a layering partition for each vertex x . Let $\gamma + 1$ be the maximum diameter of all clusters of all layering partitions.

The diameter of each cluster is at most 3λ if G has tree-length λ and at most 6ρ if G has tree-breadth ρ [9, 10]. Therefore, for each shortest path P , $\text{diam}(\text{Pr}(x, P)) \leq 3\lambda$ and $\text{diam}(\text{Pr}(x, P)) \leq 6\rho$, respectively. Thus, $\text{pg}(G) \leq \gamma \leq 3\lambda - 1$ and $\text{pg}(G) \leq \gamma \leq 6\rho - 1$.

Computing the pairwise distances of all vertices can be done in $\mathcal{O}(nm)$ time. A layering partition can be computed in linear time [5]. For a given layering partition, the diameter of each cluster can be computed in $\mathcal{O}(n^2)$ time if the pairwise distances of all vertices are known. Thus, γ can be computed in $\mathcal{O}(n^3)$ time. \square

Note that it is not necessary to know the tree-length or tree-breadth of G to compute γ . Thus, by computing γ and then running Algorithm 2 for each vertex in G , we get:

Corollary 5 *For graphs with tree-length λ or tree-breadth ρ , a minimum eccentricity shortest path can be found in $\mathcal{O}(n^{3\lambda+3})$ time or $\mathcal{O}(n^{6\rho+3})$ time, respectively.*

4.3.3 δ -Hyperbolic Graphs.

A graph has *hyperbolicity* δ if for any four vertices u, v, w , and x , the two larger of the sums $d_G(u, v) + d_G(w, x)$, $d_G(u, w) + d_G(v, x)$, and $d_G(u, x) + d_G(v, w)$ differ by at most 2δ .

Lemma 15 ([6]) *Let u, v, w , and x be four vertices in a δ -hyperbolic graph. If $d_G(u, w) > \max\{d_G(u, v), d_G(v, w)\} + 2\delta$, $d_G(v, x) < \max\{d_G(x, u), d_G(x, w)\}$.*

Lemma 16 *If G is δ -hyperbolic, then $\text{pg}(G) \leq 4\delta$.*

Proof: Consider two vertices u and w such that $u, w \in \text{Pr}(x, P)$ for some vertex x and shortest path P . Let $v \in P$ be a vertex such that $d_G(u, v) - d_G(v, w) \leq 1$ and $d_G(u, v) \geq d_G(v, w)$, i. e., v is a middle vertex on the subpath from u to w .

Assume, $d_G(u, w) > 4\delta + 1$. Thus, $d_G(u, v) \geq d_G(v, w) \geq 2\delta + 1$ and $d_G(u, w) > d_G(u, v) + 2\delta$. Therefore, by Lemma 15, $d_G(v, x) < \max\{d_G(x, u), d_G(x, w)\}$. This contradicts that $u, w \in \text{Pr}(x, P)$. Hence, the diameter of a projection is at most $4\delta + 1$ and, therefore, $\text{pg}(G) \leq 4\delta$. \square

Corollary 6 *For δ -hyperbolic graphs, a minimum eccentricity shortest path can be found in $\mathcal{O}(n^{4\delta+4})$ time.*

5 Approximation for Graphs with Bounded Hyperbolicity and Bounded Tree-Length

In the last sections, we have shown how to find a shortest path with minimum eccentricity k for several graph classes. For graphs with tree-length λ , this can require up to $\mathcal{O}(n^{3\lambda+3})$ time. In this section, we will show that, for graphs with tree-length λ , we can find a shortest path with eccentricity at most $k + 2.5\lambda$ in at most $\mathcal{O}(\lambda m)$ time and, for graphs with hyperbolicity δ , we can find a shortest path with eccentricity at most $k + \mathcal{O}(\delta \log n)$ in at most $\mathcal{O}(\delta m)$ time.

Lemma 17 *Let G be a graph with hyperbolicity δ . Two vertices x and y in G with $\text{ecc}(x) = \text{ecc}(y) = d_G(x, y)$ can be found in $\mathcal{O}(\delta m)$ time.*

Proof: Let u and v be two vertices in G such that $d_G(u, v) = \text{diam}(G)$. For an arbitrary vertex x_0 and for $i \geq 0$, let $y_i = x_{i+1}$ be vertices in G such that $d_G(x_i, y_i) = \text{ecc}(x_i)$ and $d_G(x_i, y_i) < d_G(x_{i+1}, y_{i+1})$. To prove Lemma 17, we will show that there is no vertex $y_{2\delta+1}$.

Because $d_G(x_0, y_0) = \text{ecc}(x_0)$, $d_G(x_0, y_0) \geq \max\{d_G(x_0, u), d_G(x_0, v)\}$. Thus, by Lemma 15, $d_G(u, v) \leq \max\{d_G(u, y_0), d_G(v, y_0)\} + 2\delta$ and, hence, $\text{diam}(G) \leq$

$\text{ecc}(x_1) + 2\delta$. Since $d_G(x_i, y_i) < d_G(x_{i+1}, y_{i+1})$, there is no vertex y_j with $j \geq 2\delta + 1$, otherwise $d_G(x_j, y_j) > \text{diam}(G)$. Therefore, a vertex pair x, y with $\text{ecc}(x) = \text{ecc}(y) = d_G(x, y)$ can be found in $\mathcal{O}(\delta m)$ time as follows: Pick an arbitrary vertex x_0 and find a vertex x_1 with $d_G(x_0, x_1) = \text{ecc}(x_0)$ using a BFS. Next, find a vertex x_2 such that $d_G(x_1, x_2) = \text{ecc}(x_1)$. Repeat this (at most 2δ times) until $d_G(x_i, x_{i+1}) = \text{ecc}(x_i) = \text{ecc}(x_{i+1})$. \square

Note that, if a graph has tree-length λ , its hyperbolicity is at most λ [6]. Thus, it follows:

Corollary 7 *Let G be a graph with tree-length λ . Two vertices x and y in G with $\text{ecc}(x) = \text{ecc}(y) = d_G(x, y)$ can be found in $\mathcal{O}(\lambda m)$ time.*

The next lemma will show that, in a graph with bounded tree-length, a shortest path between two mutually furthest vertices gives an approximation for the MESP-problem.

Lemma 18 *Let G be a graph with tree-length λ having a shortest path with eccentricity k . Also, let x and y be two mutually furthest vertices, i. e., $\text{ecc}(x) = \text{ecc}(y) = d_G(x, y)$. Each shortest path from x to y has eccentricity less than or equal to $k + 2.5\lambda$.*

Proof: Let P be a shortest path from s to t with eccentricity k and Q be a shortest path from x to y . Consider a tree-decomposition \mathcal{T} for G with length λ . We distinguish between two cases: (1) There is a bag in \mathcal{T} containing a vertex of P and a vertex of Q and (2) there is no such bag in \mathcal{T} .

Case 1: There is a bag in \mathcal{T} containing a vertex of P and a vertex of Q . We define bags B_x and B_y as follows: Both contain a vertex of P and a vertex of Q , B_x is a bag closest to a bag containing x , B_y is a bag closest to a bag containing y , and the distance between B_x and B_y in \mathcal{T} is maximal. Let $\{B_0, B_1, \dots, B_l\}$ be a subpath of the shortest path from B_x to B_y in \mathcal{T} such that B_0 is a bag closest to a bag containing s , B_l is a bag closest to a bag containing t , B_i is adjacent to B_{i+1} in \mathcal{T} ($0 \leq i < l$), and the distance l between B_0 and B_l is maximal. Without loss of generality, let $d_{\mathcal{T}}(B_x, B_0) \leq d_{\mathcal{T}}(B_x, B_l)$. Let p_s be the vertex in $B_0 \cap P$ which is closest to s in G and let p_t be the vertex in $B_l \cap P$ which is closest to t in G . Figure 2 gives an illustration.

Claim. For each vertex $p \in P$ with $d_G(s, p_s) \leq d_G(s, p) \leq d_G(s, p_t)$, $d_G(p, Q) \leq 1.5\lambda$.

Proof (Claim): There is a vertex set $\{p_s = p_0, p_1, \dots, p_l, p_{l+1} = p_t\} \subseteq P$, where $p_i \in B_{i-1} \cap B_i$ for all positive $i \leq l$. Because $p_i, p_{i+1} \in B_i$ for $0 \leq i \leq l$, $d_G(p_i, p_{i+1}) \leq \lambda$. Thus, because P is a shortest path, for all $p' \in P$ with $d_G(s, p_s) \leq d_G(s, p') \leq d_G(s, p_t)$ there is a vertex p_i with $0 \leq i \leq l + 1$ such that $d_G(p_i, p') \leq 0.5\lambda$. By definition of \mathcal{T} , each bag B_i ($0 \leq i \leq l$) contains a vertex $q \in Q$, i. e., $d_G(p_i, Q) \leq \lambda$ ($0 \leq i \leq l + 1$). Therefore, for all $p' \in P$ with $d_G(s, p_s) \leq d_G(s, p') \leq d_G(s, p_t)$ there is a vertex p_i with $0 \leq i \leq l + 1$ such that $d_G(p', Q) \leq d_G(p_i, p') + d_G(p_i, Q) \leq 1.5\lambda$. \diamond

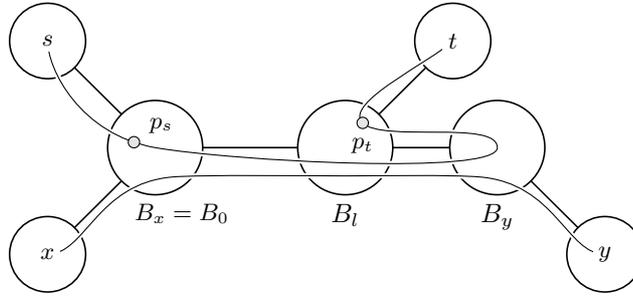


Figure 2: Example for a possible tree-decomposition.

Consider an arbitrary vertex v in G . Let v' be a vertex in P closest to v and let P_v be a shortest path from v to v' . If v' is between p_s and p_t , i. e., $d_G(s, p_s) \leq d_G(s, v') \leq d_G(s, p_t)$, by the claim above, $d_G(v, Q) \leq d_G(v, v') + d_G(v', Q) \leq k + 1.5\lambda$. If P_v intersects a bag containing a vertex $q \in Q$, $d_G(v, Q) \leq k + \lambda$.

Next, consider the case when P_v does not intersect a bag containing a vertex of Q and (without loss of generality) $d_G(s, v') > d_G(s, p_t)$. In this case, each path from x to v intersects B_l .

Claim. There is a vertex $u \in B_l$ such that $d_G(u, y) \leq k + 0.5\lambda$.

Proof (Claim): Let y' be a vertex in P that is closest to y and let P_y be a shortest path from y to y' . If P_y intersects B_l , there is a vertex $u \in P_y \cap B_l$ with $d_G(y, u) \leq k$.

If P_y does not intersect B_l , there is a subpath of P starting at p_t , containing y' , and ending in a vertex $p_l \in B_l$. Because $d_G(p_t, p_l) \leq \lambda$, $d_G(y', \{p_t, p_l\}) \leq 0.5\lambda$. Therefore, $d_G(y, \{p_t, p_l\}) \leq d_G(y, y') + d_G(y', \{p_t, p_l\}) \leq k + 0.5\lambda$. \diamond

Let u, v' , and z be vertices in B_l such that $d_G(u, y) \leq k + 0.5\lambda$, v' is on a shortest path from x to v , and $z \in Q$. Because $d_G(x, y) = \text{ecc}(x)$, $d_G(x, v') + d_G(v', v) \leq d_G(x, y)$. Also, by the triangle inequality, $d_G(x, y) \leq d_G(x, v') + d_G(v', y)$ and $d_G(v', y) \leq d_G(v', u) + d_G(u, y)$. Because $\{u, v', z\} \subseteq B_l$ and $d_G(u, y) \leq k + 0.5\lambda$, $d_G(v', v) \leq k + 1.5\lambda$ and therefore $d_G(z, v) \leq k + 2.5\lambda$.

Thus, if there is a bag in \mathcal{T} containing a vertex of P and a vertex of Q , $d_G(v, Q) \leq k + 2.5\lambda$ for all vertices v in G .

Case 2: There is no bag in \mathcal{T} containing vertices of P and Q . Because there is no such bag, \mathcal{T} contains a bag B such that each path from x and y to P intersects B and there is a vertex $z \in B \cap Q$.

Consider an arbitrary vertex v . If there is a shortest path P_v from v to P which intersects B , then $d_G(z, v) \leq k + \lambda$. If there is no such path, each path from x to v intersects B . Let $v' \in B$ be a vertex on a shortest path from x to v and let $u \in B$ be a vertex on a shortest path from y to P . Note that $d_G(u, y) \leq k$.

Because $d_G(x, y) = \text{ecc}(x)$, $d_G(x, v') + d_G(v', v) \leq d_G(x, y)$. Also, by the triangle inequality, $d_G(x, y) \leq d_G(x, v') + d_G(v', y)$ and $d_G(v', y) \leq d_G(v', u) + d_G(u, y)$. Because $\{u, v', z\} \subseteq B$ and $d_G(u, y) < k$, $d_G(v', v) < k + \lambda$ and therefore $d_G(z, v) < k + 2\lambda$.

Thus, if there is no bag in \mathcal{T} containing vertices of P and Q , $d_G(v, Q) < k + 2\lambda$ for all vertices v in G . \square

In [6], it was shown that an n -vertex δ -hyperbolic graph has tree-length at most $\mathcal{O}(\delta \log n)$.

Corollary 8 *Let G be a graph with hyperbolicity δ having a shortest path with eccentricity k . Also, let x and y be two mutually furthest vertices, i. e., $\text{ecc}(x) = \text{ecc}(y) = d_G(x, y)$. Each shortest path from x to y has eccentricity less than or equal to $k + \mathcal{O}(\delta \log n)$.*

Lemma 17, Lemma 18, Corollary 7, and Corollary 8 imply our main result of this section:

Theorem 5 *Let G be a graph having a shortest path with eccentricity k . If G has tree-length λ , a shortest path with eccentricity at most $k + 2.5\lambda$ can be found in $\mathcal{O}(\lambda m)$ time. If G has hyperbolicity δ , a shortest path with eccentricity at most $k + \mathcal{O}(\delta \log n)$ can be found in $\mathcal{O}(\delta m)$ time.*

A graph is chordal if and only if it has tree-length 1 [15].

Corollary 9 *If G is a chordal graph and has a shortest path with eccentricity k , a shortest path in G with eccentricity at most $k + 2$ can be found in linear time.*

Figure 3 gives an example that, for chordal graphs, $k + 2$ is a tight upper bound for the eccentricity of the determined shortest path.

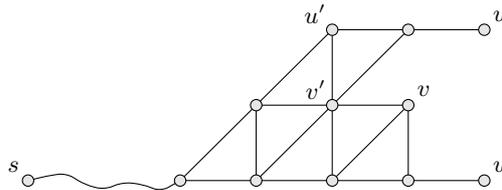


Figure 3: A chordal graph G . A shortest path from s to v passing v' has eccentricity 2 which is the minimum for all shortest paths in G . The diametral path from s to u passing u' has eccentricity 4 because of its distance to w .

6 Conclusion

We have investigated the Minimum Eccentricity Shortest Path problem for some structured graph classes. For these classes, we were able to present linear or

polynomial time algorithms. Additionally, we presented a simple algorithm which gives an additive approximation in linear time for chordal graphs, in $\mathcal{O}(\lambda m)$ time for graphs with tree-length λ , and in $\mathcal{O}(\delta m)$ time for graphs with hyperbolicity δ .

One reason why the runtime to find an optimal path for distance-hereditary graphs is linear is that we can determine the start and end vertices of an optimal path in linear time for these graphs. For the other classes, the algorithm iterates over all possible start vertices s . We know that, for general graphs, the problem remains NP-complete even if a start-end vertex pair is given (see the reduction in [12]). Also, we have shown that there is a shortest path with minimum eccentricity between every diametral pair of vertices of a distance-hereditary graph (Theorem 1). This leads to the following question: How hard is it to determine the start and end vertices of an optimal path? This question applies to general graphs as well as to special graph classes like chordal graphs.

Another interesting question is, for which other graph classes the problem remains NP-complete or can be solved in polynomial time. The NP-completeness proof in [12] uses a reduction from SAT. There is a planar version of 3-SAT (see [17]). Does this imply that the problem remains NP-complete for planar graphs?

References

- [1] H.-J. Bandelt and H. M. Mulder. Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B*, 41(2):182–208, 1986. doi:10.1016/0095-8956(86)90043-2.
- [2] A. Brandstädt, V. Chepoi, and F. Dragan. Distance approximating trees for chordal and dually chordal graphs. *Journal of Algorithms*, 30(1):166–184, 1999. doi:10.1006/jagm.1998.0962.
- [3] A. Brandstädt, F. Dragan, V. Chepoi, and V. Voloshin. Dually chordal graphs. *SIAM Journal on Discrete Mathematics*, 11(3):437–455, 1998. doi:10.1137/S0895480193253415.
- [4] A. Brandstädt, V. Le, and J. Spinrad. *Graph Classes: A Survey*. Society for Industrial and Applied Mathematics, 1999. doi:10.1137/1.9780898719796.
- [5] V. Chepoi and F. Dragan. A note on distance approximating trees in graphs. *European Journal of Combinatorics*, 21(6):761–766, 2000. doi:10.1006/eujc.1999.0381.
- [6] V. Chepoi, F. Dragan, B. Estellon, M. Habib, and Y. Vaxès. Diameters, centers, and approximating trees of delta-hyperbolic geodesic spaces and graphs. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry, SCG '08*, pages 59–68, New York, NY, USA, 2008. ACM. doi:10.1145/1377676.1377687.
- [7] D. G. Corneil, S. Olariu, and L. Stewart. Linear time algorithms for dominating pairs in asteroidal triple-free graphs. *SIAM Journal on Computing*, 28(4):1284–1297, 1999. doi:10.1137/S0097539795282377.
- [8] A. D’Atri and M. Moscarini. Distance-hereditary graphs, steiner trees, and connected domination. *SIAM Journal on Computing*, 17(3):521–538, 1988. doi:10.1137/0217032.
- [9] Y. Dourisboure, F. F. Dragan, C. Gavaille, and C. Yan. Spanners for bounded tree-length graphs. *Theoretical Computer Science*, 383(1):34–44, 2007. doi:10.1016/j.tcs.2007.03.058.
- [10] F. F. Dragan and E. Köhler. An approximation algorithm for the tree t-spanner problem on unweighted graphs via generalized chordal graphs. *Algorithmica*, 69(4):884–905, 2013. doi:10.1007/s00453-013-9765-4.
- [11] F. F. Dragan, E. Köhler, and A. Leitert. Line-distortion, bandwidth and path-length of a graph. In *Proceedings of 14th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2014*, pages 158–169. Springer, 2014. doi:10.1007/978-3-319-08404-6_14.

- [12] F. F. Dragan and A. Leitert. On the minimum eccentricity shortest path problem. In *Proceedings of 14th International Symposium on Algorithms and Data Structures*, WADS 2015, pages 276–288. Springer, 2015. doi:10.1007/978-3-319-21840-3_23.
- [13] F. F. Dragan and F. Nicolai. Lexbfs-orderings of distance-hereditary graphs with application to the diametral pair problem. *Discrete Applied Mathematics*, 98(3):191–207, 2000. doi:10.1016/S0166-218X(99)00157-2.
- [14] M. Farber and R. E. Jamison. Convexity in graphs and hypergraphs. *SIAM Journal on Algebraic Discrete Methods*, 7(3):433–444, 1986. doi:10.1137/0607049.
- [15] F. Gavril. The intersection graphs of subtrees in trees are exactly the chordal graphs. *Journal of Combinatorial Theory, Series B*, 16(1):47–56, 1974. doi:10.1016/0095-8956(74)90094-X.
- [16] E. Howorka. A characterization of distance-hereditary graphs. *The Quarterly Journal of Mathematics*, 28(4):417–420, 1977. doi:10.1093/qmath/28.4.417.
- [17] D. Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982. doi:10.1137/0211025.
- [18] D. Lokshtanov. On the complexity of computing tree-length. *Discrete Applied Mathematics*, 158(7):820–827, 2010. doi:10.1016/j.dam.2009.10.007.
- [19] H. Müller. Hamiltonian circuits in chordal bipartite graphs. *Discrete Mathematics*, 156(1–3):291–298, 1996. doi:10.1016/0012-365X(95)00057-4.
- [20] N. Robertson and P. Seymour. Graph minors. ii. algorithmic aspects of tree-width. *Journal of Algorithms*, 7(3):309–322, 1986. doi:10.1016/0196-6774(86)90023-4.
- [21] P. J. Slater. Locating central paths in a graph. *Transportation Science*, 16(1):1–18, 1982. doi:10.1287/trsc.16.1.1.
- [22] R. E. Tarjan and M. Yannakakis. Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM Journal on Computing*, 13(3):566–579, 1984. doi:10.1137/0213035.