# Computing optimal shortcuts for networks*

**Delia Garijo[1], Alberto Márquez[1], Natalia Rodríguez[2], and Rodrigo I. Silveira[3]**

1 **Departamento de Matemática Aplicada I, Universidad de Sevilla, Spain**
`dgarijo@us.es,almar@us.es`

2 **Departamento de Computación, Universidad de Buenos Aires, Argentina**
`nrodriguez@dc.uba.ar`

3 **Departament de Matemàtiques, Universitat Politècnica de Catalunya, Spain**
`rodrigo.silveira@upc.edu`

## Abstract

We augment a plane Euclidean network with a segment or *shortcut* to minimize the largest distance between any two points along the edges of the resulting network. In this continuous setting, the problem of computing distances and placing a shortcut is much harder as all points on the network, instead of only the vertices, must be taken into account. Our main result for general networks states that it is always possible to determine in polynomial time whether the network has an optimal shortcut and compute one in case of existence. We also improve this general method for networks that are paths, restricted to using two types of shortcuts: those of any fixed direction and shortcuts that intersect the path only on its endpoints.

## 1 Introduction

A *geometric network* of points in the plane is an undirected graph whose vertices are points in $\mathbb{R}^2$ and whose edges are straight-line segments connecting pairs of points. A *Euclidean network* is an edge-weighted geometric network: edges are assigned lengths equal to the Euclidean distance between their endpoints. When in addition there are no crossings between edges, the Euclidean network is said to be *plane*. In the following, we shall simply say network, it being understood as plane Euclidean network.

In this work we study a variant of the Diameter-Optimal-$k$-Augmentation problem that deals with inserting $k$ additional segments into a network, while minimizing the largest distance in the resulting network (see the survey [7] for more on augmentation problems over plane geometric graphs). Concretely, we consider a continuous version of the problem for $k = 1$: the endpoints of the inserted segment, called *shortcut*, are allowed to be any two points on the network (instead of only vertices), and we seek to minimize the largest distance between any two points on the edges of the augmented network. The complexity of the problem, which lies in the fact that all points must be considered in computing distances and placing the shortcut, motivates that there are very few results on this continuous version.

Yang [8] designed three different approximation algorithms to compute for certain types of paths an *optimal* shortcut which, informally, is a segment that attains the minimum of that largest distance. De Carufel et al. [3] gave an algorithm to determine in linear time optimal shortcuts for paths and optimal pairs of shortcuts ($k = 2$) for convex cycles. We want to stress that their definition of shortcut is simpler, as they do not consider the intersections between the shortcut and the network as points of the augmented network. The

* D.G. and R.S. were supported by project MTM2015-63791-R. R. S. was also supported by Gen. Cat. 2017SGR1640 and MINECO through the Ramón y Cajal program. A.M. was supported by project BFU2016-74975-P.

34th European Workshop on Computational Geometry, Berlin, Germany, March 21–23, 2018.
This is an extended abstract of a presentation given at EuroCG'18. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

same definition of shortcut was used in [4] to develop a study for trees, which includes the computation of an optimal shortcut for a tree of size $n$ in $O(n \log n)$ time.

The first approach for general networks was presented in [2] where the authors compute shortcuts (i.e., segments whose insertion improve the diameter) in polynomial time, but they do not obtain optimal shortcuts. In Section 2, we do decide existence and compute such optimal shortcuts in polynomial time. Section 3 focuses on paths: we first analyze how distances change by the insertion of a shortcut, and compute the largest distance between any two points on the augmented network in $\Theta(n)$ time. We also improve the method of Section 2 for shortcuts of any fixed direction and shortcuts that intersect the path only at its endpoints. Due to space limitations, proofs are briefly sketched.

## 1.1 Preliminaries

The *locus* of a network $\mathcal{N} = (V(\mathcal{N}), E(\mathcal{N}))$, denoted by $\mathcal{N}_\ell$, is the set of all points of the Euclidean plane that are on $\mathcal{N}$. Thus, $\mathcal{N}_\ell$ is treated indistinctly as a network or as a closed point set. We write $a \in \mathcal{N}_\ell$ for a point $a$ on $\mathcal{N}_\ell$, and $V(\mathcal{N}) \subset \mathcal{N}_\ell$. We will use $\mathcal{P}_\ell$ instead of $\mathcal{N}_\ell$ when $\mathcal{N}_\ell$ is a path. A *path* $P$ connecting two points $a, b$ on $\mathcal{N}_\ell$ is a sequence $au_1 \dots u_k b$ such that $u_1u_2, \dots, u_{k-1}u_k \in E(\mathcal{N})$, $a$ is a point on an edge ($\neq u_1u_2$) incident to $u_1$, and $b$ is a point on an edge ($\neq u_{k-1}u_k$) incident to $u_k$. We use $|P|$ to denote $P$'s *length*, i.e., the sum of the lengths of all edges $u_iu_{i+1}$ plus the lengths of the segments $au_1$ and $bu_k$ The length of a shortest path $P$ from $a$ to $b$ is the *distance* between $a$ and $b$ on $N_\ell$. This distance is written as $d_{N_\ell}(a, b)$ or $d(a, b)$ when the network is understood, and whenever $ab \notin E(N_\ell)$, it is different from the Euclidean distance between the points, denoted by $|ab|$.

The *eccentricity* of a point $a \in \mathcal{N}_\ell$ is $\mathrm{ecc}(a) = \max_{b \in \mathcal{N}_\ell} d(a, b)$, and the *diameter* of $\mathcal{N}_\ell$ is $\mathrm{diam}(\mathcal{N}_\ell) = \max_{a \in \mathcal{N}_\ell} \mathrm{ecc}(a)$. Two points $a, b \in \mathcal{N}_\ell$ are *diametral* whenever $d(a, b) = \mathrm{diam}(\mathcal{N}_\ell)$, and a shortest path between them is then called *diametral path*.

A *shortcut* for $\mathcal{N}_\ell$ is a segment $s$ with endpoints on $\mathcal{N}_\ell$ such that $\mathrm{diam}(\mathcal{N}_\ell \cup s) < \mathrm{diam}(\mathcal{N}_\ell)$. We say that shortcut $s$ is *simple* if its two endpoints are the only intersection points with $\mathcal{N}_\ell$, and $s$ is *maximal* if it is the intersection of a line and $(\mathcal{N}_\ell \cup s)$, i.e., $s = (\mathcal{N}_\ell \cup s) \cap \ell$, for some line $\ell$. A shortcut is *optimal* if it minimizes $\mathrm{diam}(\mathcal{N}_\ell \cup s)$ among all shortcuts $s$ for $\mathcal{N}_\ell$.

# 2 General networks

The main result in [2] states that one can always determine in polynomial time whether a network $\mathcal{N}_\ell$ has a shortcut, and compute one in case of existence. In this section, we first state the analogous result but for optimal shortcuts. Our proof mainly uses the ideas in [2], but some additional information is needed to capture the property of being optimal.

By Lemma 3.2 of [2], $\mathrm{diam}(\mathcal{N}_\ell)$ can be computed in polynomial time, and the diametral pairs of points on $\mathcal{N}_\ell$ are either two vertices, or two points on distinct non-pendant edges, or a pendant vertex and a point on a non-pendant edge (an edge is pendant if one of its vertices has degree one). Thus, with some abuse of notation, we say that a diametral pair may be vertex-vertex, edge-edge, or vertex-edge.

Let $\alpha, \beta \in V(\mathcal{N}) \cup E(\mathcal{N})$, and let $e = uv$ and $e' = u'v'$ be two edges of $\mathcal{N}$. When $\alpha$ is an edge, we use $\mathrm{ecc}(u, \alpha)$ to indicate the maximum distance from $u$ to the points on $\alpha$ (analogous for $\beta$ and the remaining endpoints of $e$ and $e'$); if $\alpha$ is a vertex, $\mathrm{ecc}(u, \alpha) = d(u, \alpha)$. In general, $\mathrm{ecc}(\alpha, \beta) = \max_{t \in \alpha, z \in \beta} d(t, z)$.

▶ **Lemma 2.1.** *Let $y = ax + b$ be a line intersecting edges $e = uv$ and $e' = u'v'$ on, respectively, points $p$ and $q$, and let $\alpha, \beta \in V(\mathcal{N}) \cup E(\mathcal{N})$. For each pair $(w, z)$ with $w \in \{u, v\}$ and*

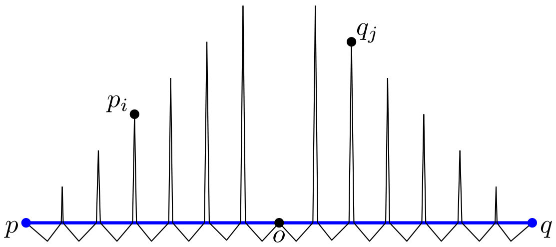


**Figure 1** $\Theta(n)$ spikes placed symmetrically with respect to the midpoint $o$ of a shortcut $pq$. The spikes are spaced by one unit each, while their heights are set such that the distance from $o$ to the top of the spike is always the same, namely $|pq|/2$. Thus, the distance between the top of one spike on the left of $o$ and one on its right, like $p_i$ and $q_j$, is $|pq|$, and equals the diameter of $\mathcal{P}_\ell \cup pq$.

*$z \in \{u', v'\}$, function $f^{w,z}_{\alpha,\beta}(a,b) = \mathrm{ecc}(w,\alpha) + d(w,p) + |pq| + d(q,z) + \mathrm{ecc}(z,\beta)$ is linear in $b$.*

Given a line $m$ that crosses two fixed edges $e, e' \in E(\mathcal{N})$, let $\mathcal{P}_{e,e'}(m)$ be the set of equivalent lines to $m$ that intersect edges $e$ and $e'$.[1] Consider a line $r \equiv y = ax + b$ in $\mathcal{P}_{e,e'}(m)$ and the set $I = \{e = e_0, e_1, \ldots, e_k, e_{k+1} = e'\}$ of edges that it intersects in between $e$ and $e'$; let $e_i = u_i v_i$ and $p_i = r \cap e_i$. For a fixed diametral pair $\alpha, \beta \in V(\mathcal{N}) \cup E(\mathcal{N})$, a function of the type $f^{w,z}_{\alpha,\beta}(a,b)$ with $w \in \{u_i, v_i\}$ and $z \in \{u_j, v_j\}$, $0 \le i \ne j \le k+1$, computes $\mathrm{ecc}(\alpha,\beta)$ when using a path that passes through vertices $w, z$ and contains segment $p_i p_j$. For a fixed value of $a$, each function $f^{w,z}_{\alpha,\beta}(a,b)$ becomes a linear function on $b$. Thus, geometrically, an optimal shortcut for $\mathcal{N}_\ell$ in $\mathcal{P}_{e,e'}(m)$ is given by the minimum of the upper envelope of the set of lines $f^{w,z}_{\alpha,\beta}(a,b)$. Note that any shortcut $s$ satisfies that $\mathrm{ecc}(t) < \mathrm{diam}(\mathcal{N}_\ell)$ for every $t \in s$, and so all segments $p_i p_j$ must be included in the set of diametral pairs $\alpha, \beta$. Applying the same argument to the $O(n^2)$ regions $\mathcal{P}_{e,e'}(m)$, we obtain the following theorem.

▶ **Theorem 2.2.** *It is always possible to determine in polynomial time whether a network $\mathcal{N}_\ell$ admits an optimal shortcut, and compute one in case of existence.*

It would be interesting to characterize the networks $\mathcal{N}_\ell$ that have an optimal shortcut, even restricted to simple shortcuts. The following proposition is a first approach to this question. Note that one must distinguish between an *optimal simple shortcut* and a *simple optimal shortcut*. The first is a shortcut that is optimal in the set of simple shortcuts; this is different of being optimal in the set of all shortcuts and, in addition, to be simple.

▶ Proposition 1. Let $\mathcal{N}$ be a network whose locus $\mathcal{N}_\ell$ admits a simple shortcut, and let $\overline{\mathcal{N}}$ be the network resulting from adding to $\mathcal{N}$ all edges of the convex hull of $V(\mathcal{N})$. If all faces of $\overline{\mathcal{N}}$ are convex, then $\mathcal{N}_\ell$ has an optimal simple shortcut.

## 3 Path networks

We begin by noting that the insertion of a shortcut to a path can create a quadratic number of diametral pairs; as illustrated in the construction in Figure 1.

### 3.1 Diameter after inserting a shortcut

The diameter of $\mathcal{P}_\ell$ can be immediately computed in linear time, however, the addition of a shortcut $s$ can create a linear number of new bounded faces, thus in principle it is not clear

[1] Two lines are *equivalent* if the half-planes to the right (left) they define contain the same vertices of $\mathcal{N}$

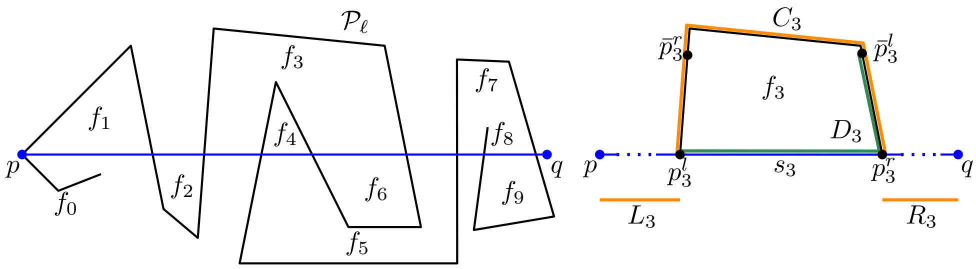


**Figure 2** Left: faces created by $s$; $f_0$ and $f_8$ are degenerate faces. Right: detail for $f_3$ (note that the chain bounding $f_4$ is not considered for $f_3$). Thick lines are used here to denote distances.

whether $\operatorname{diam}(\mathcal{P}_\ell \cup \mathrm{s})$ can be computed in linear time, i.e., without computing the diameter between each pair of faces. The main result in this section is that this is still possible.

Suppose, without loss of generality, that $s = pq$ is horizontal and maximal. Assume (bounded) faces are numbered in the order of their left endpoints from left to right along $s$ (using right endpoints to disambiguate). If the first vertex of $\mathcal{P}_\ell$ is not on $s$, we consider the path from its first vertex to the first intersection of $\mathcal{P}_\ell$ with $s$ as a (degenerate) face (analogous for the last vertex of $\mathcal{P}_\ell$), see Figure 2(left). A face $f_i$ can be bounded by several chains of $\mathcal{P}_\ell$. However, we are only interested in associating $f_i$ to its chain with leftmost endpoint on $s$. Thus a face $f_i$ will be defined by a subsegment $s_i$ of $s$ from point $p_i^l$ to $p_i^r$, and a polygonal chain $C_i$ on one side of $s$, see Figure 2(right). Let $|C_i|$ be the length of $C_i$, and let $L_i = |pp_i^l|$ and $R_i = |p_i^r q|$ be the distances to the leftmost and rightmost endpoints of $s$. Finally, let $D_i$ be the distance on $\mathcal{P}_\ell \cup s$ from $p_i^l$ to its furthest point $\bar{p}_i^l$ on $f_i$. Note that this is identical to the distance from $p_i^r$ to the analogously defined point $\bar{p}_i^r$, and also to the semiperimeter of $f_i$, which is equal to $(|C_i| + |s_i|)/2$.

▶ **Observation 3.1** (Disjoint faces). *Let $f_i, f_j$ be two faces of $(\mathcal{P}_\ell \cup s)$ with $s_i \cap s_j = \emptyset$ and $s_i$ to the left of $s_j$. The diameter of $C_i \cup p_i^l p_j^r \cup C_j$ is $D_i + |p_i^r p_j^l| + D_j = D_i + R_i - R_j - |s_j| + D_j$ and is achieved by $\bar{p}_i^r$ and $\bar{p}_j^l$.*

▶ **Observation 3.2** (Nested faces). *Let $f_i, f_j$ be two faces of $(\mathcal{P}_\ell \cup s)$ with $s_j \subset s_i$. The diameter of $C_i \cup s_i \cup C_j$ is $\frac{1}{2}(|C_i| + |p_i^l p_j^l| + |p_i^r p_j^r| + |C_j|) = \frac{1}{2}(|C_i| + L_j - L_i + R_j - R_i + |C_j|)$.*

▶ **Observation 3.3** (Overlapping faces). *Let $f_i, f_j$ be two faces of $(\mathcal{P}_\ell \cup s)$ with $s_i \cap s_j \neq \emptyset$, $p_i^l \notin s_j$ and $p_j^r \notin s_i$. The diameter of $C_i \cup p_i^l p_j^r \cup C_j$ is $\frac{1}{2}(|C_i| + |p_i^l p_j^l| + |p_i^r p_j^r| + |C_j|) = \frac{1}{2}(|C_i| + L_j - L_i + R_i - R_j + |C_j|)$.*

The preceding observations reveal a key property: the linear ordering between faces induced by $s$ defines uniquely how the diameter between two faces is achieved. Thus, the algorithm for computing $\operatorname{diam}(\mathcal{P}_\ell \cup \mathrm{s})$ in linear time starts by going along $\mathcal{P}_\ell$ and computing all intersections with $s$ in the order of $\mathcal{P}_\ell$. Then we apply a linear-time algorithm for Jordan sorting [6] to obtain the intersections in the order along $s$, say, from right to left. Within the same running time we can obtain the necessary information of each face created by the insertion of $s$. Next we compute and store certain information for each face $f_i$: its furthest faces, respectively, to the right and to the left, its furthest face nested inside $f_i$, and its furthest face with one endpoint in $f_i$ and the other one outside. When sweeping the faces along $s$, this information allows us to find in $O(1)$ time, for each face $f_i$, its furthest face from the ones seen so far, so the maximum distance between any two faces can be found in total linear time.

▶ **Theorem 3.4.** *Given a path $\mathcal{P}_\ell$ with $n$ vertices and a shortcut $s$, the diameter of $(\mathcal{P}_\ell \cup s)$ can be computed in $\Theta(n)$ time.*

## 3.2 Optimal horizontal shortcuts

The observations in Section 3.1 also give us a way to compute an optimal horizontal shortcut for a path considerably faster than using the general method in Section 2. After a suitable rotation, this allows to find optimal shortcuts of any fixed orientation.

Assume again that shortcuts are horizontal and maximal, so they can be treated as horizontal lines. Now, consider the vertices in $\mathcal{P}_\ell$ sorted increasingly by $y$-coordinate, and let $y_a, y_b$, with $y_a < y_b$, be the $y$-coordinates of two consecutive vertices in that order. By Observations 3.1–3.3, the distance between any two faces $f_i$ and $f_j$ is a linear function $d_{ij}(y)$ for $y_a \leq y \leq y_b$. Thus, each face is associated with $k-1$ lines in 2D where $k$ is the total number of faces, leading to a set $\mathcal{L}$ of $\Theta(k^2)$ lines (note that $k = O(n)$). The optimal shortcut over all $y \in [y_a, y_b]$ is given by the minimum of the upper envelope of $\mathcal{L}$, which can be computed in $O(k^2 \log k)$ time [5]. If this is done with each of the $n-1$ horizontal strips formed by consecutive vertices of $\mathcal{N}_\ell$, the optimal horizontal shortcut is obtained in total $O(n^3 \log n)$ time. Now, this method can be improved if, instead of computing from scratch the upper envelope of $\mathcal{L}$ at each horizontal strip, we maintain the upper envelope between consecutive strips and only add or remove the lines that change when going from one strip to the next one. The changes between two consecutive strips are of three types: (i) one of the two line segments bounding a face within the strip changes; (ii) a face ends; (iii) a new face appears. In the worst case, $n-1$ lines are removed from $\mathcal{L}$ and another $n-1$ lines are added to $\mathcal{L}$. Maintaining the upper envelope of $n$ lines is equivalent to maintaining the convex hull of $n$ points in 2D, which can be done in $O(n \log n)$ amortized time and using $O(n^2)$ space [1].

▶ **Theorem 3.5.** *For every path $\mathcal{P}_\ell$ with $n$ vertices, it is possible to find an optimal horizontal shortcut in $O(n^2 \log n)$ time, using $O(n^2)$ space.*

## 3.3 Optimal simple shortcuts

Consider now a simple shortcut $s = pq$ for a path $\mathcal{P}_\ell$ with endpoints $u, v$. Suppose that point $p$ is closer to $u$ than $q$ along $P_\ell$; let $x = d(u,p)$ and $y = d(v,q)$. There is only one bounded face in $P_\ell \cup s$ whose boundary is a cycle $C(p,q)$. Let $\overline{p}$ and $\overline{q}$ be the farthest points from, respectively, $p$ and $q$ on $C(p,q)$, and let $z = (d_{P_\ell}(p,q) - |pq|)/2$. Note that $d(\overline{p}, \overline{q}) = |pq|$ and $z = d(p, \overline{q}) = d(\overline{p}, q)$. See Figure 3(left). There are three candidates for diametral path in $\mathcal{P}_\ell \cup s$ (see [3]): (1) the path from $u$ to $v$ via $s$ is diametral if and only if $z = \min\{x,y,z\}$, (2) the path from $u$ to $\overline{p}$ via $s$ is diametral whenever $y = \min\{x,y,z\}$, (3) the path from $v$ to $\overline{q}$ via $s$ is diametral if and only if $x = \min\{x,y,z\}$. Thus, $\mathrm{diam}(\mathcal{P}_\ell \cup s) \in \{x+y+|pq|, x+z+|pq|, y+z+|pq|\}$. Further, in [3] it is proved that $\mathcal{P}_\ell$ has an optimal shortcut satisfying $x = y$, which allows to compute it in linear time. Their method does not apply here because, as explained in the Introduction, their definition of shortcut leads to a much simpler situation. Nevertheless, in the same fashion, we can prove the following lemma.

▶ **Lemma 3.6.** *Let $pq$ be an optimal simple shortcut for $\mathcal{P}_\ell$. The following statements hold.*

1. *If neither $p$ nor $q$ are vertices of $\mathcal{P}_\ell$ then $x = y = z$.*
2. *If $p$ or $q$ are vertices of $\mathcal{P}_\ell$ then the two smallest values among $x, y, z$ are equal.*

With Lemma 3.6 in hand, we first compute the points $p, q$ where $x = y = z$ by solving $O(n)$ quadratic equations, and obtain $O(n)$ candidates for optimal simple shortcut such that

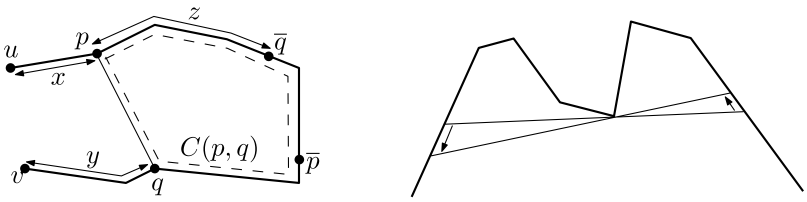


**Figure 3** Left: Inserting a simple shortcut $pq$. Right: Shortcut that is pivoting on a vertex.

$p, q \notin V(\mathcal{P}_\ell)$. We then classify them into three sets: $\mathcal{S}$ of simple shortcuts, $\mathcal{L}$ of limit cases (the segment intersects $\mathcal{P}_\ell$ on three points), and shortcuts that intersect $\mathcal{P}_\ell$ on four points. Candidate segments with at least one endpoint in $V(\mathcal{P}_\ell)$ must then be included in $\mathcal{S}$ and $\mathcal{L}$; this last set also contains those segments that are pivoting on a vertex of $\mathcal{P}_\ell$ (see Figure 3(right)) and such that the two smallest values among $x, y, z$ are equal. Finally, we obtain the minimum value of $\mathrm{diam}(\mathcal{P}_\ell \cup s)$ over $s \in \mathcal{S} \cup \mathcal{L}$; there exists an optimal simple shortcut whenever the minimum is attained by a segment in $\mathcal{S}$.

▶ **Theorem 3.7.** *It is always possible to decide whether a path $\mathcal{P}_\ell$ with $n$ vertices has an optimal simple shortcut and compute one (in case of existence) in $O(n^2)$ time.*

## 4 Conclusion

We compute optimal shortcuts for general networks and improve our method for paths but restricted to simple shortcuts and those of any fixed direction. This is an ongoing research and our first priority is to investigate techniques that allow us to design a more efficient algorithm for computing an optimal shortcut (with no restriction) for a path. It would be then interesting to develop a similar algorithmic study for more general networks and to consider the analogous problems when augmenting the network with more than one shortcut.

### References


1 G. S. Brodal and R. Jacob. Dynamic planar convex hull. In *Proc. 43rd Symposium on Foundations of Computer Science*, FOCS '02, pages 617–626, 2002.

2 J. Cáceres, D. Garijo, A. González, A. Márquez, M. L. Puertas, and P. Ribeiro. Shortcut sets for the locus of plane Euclidean networks, 2016. arXiv:1604.03878. Extended Abstract in ENDM, 54:163-168, 2016.

3 J. L. De Carufel, C. Grimm, A. Maheshwari, and M. Smid. Minimizing the continuous diameter when augmenting paths and cycles with shortcuts. In *Proc. 15th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2016*, pages 27:1–27:14, 2016.

4 J. L. De Carufel, C. Grimm, S. Schirra, and M. Smid. Minimizing the continuous diameter when augmenting a tree with a shortcut. In *Algorithms and Data Structures. WADS 2017. LNCS 10389*, pages 301– 312, 2017.

5 M. de Berg, O. Cheong, M. J. van Kreveld, and M. H. Overmars. *Computational geometry: algorithms and applications, 3rd Edition.* Springer, 2008.

6 K. Y. Fung, T. M. Nicholl, R. E. Tarjan, and C. J. Van Wyk. Simplified linear-time Jordan sorting and polygon clipping. *Inf. Proc. Letters*, 35(2):85 – 92, 1990.

7 F. Hurtado and C. D. Tóth. Plane geometric graph augmentation: A generic perspective. In *Pach J. (eds) Thirty Essays on Geometric Graph Theory*, pages 327–354. Springer, 2013.

8 B. Yang. Euclidean chains and their shortcuts. *Theor. Comput. Sci.*, 497:55–67, 2013.