# Modelling Gateway Placement in Wireless Networks: Geometric $k$-Centres of Unit Disc Graphs[☆]

S. Durocher[a,1,*], K. R. Jampani[b], A. Lubiw[b,1], L. Narayanan[c,1]

[a] *Department of Computer Science, University of Manitoba, Winnipeg, Canada*
[b] *Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada*
[c] *Department of Computer Science and Software Engineering, Concordia University, Montréal, Canada*

**Abstract**

Motivated by the gateway placement problem in wireless networks, we consider the *geometric $k$-centre* problem on unit disc graphs: given a set of points $P$ in the plane, find a set $F$ of $k$ points in the plane that minimizes the maximum graph distance from any vertex in $P$ to the nearest vertex in $F$ in the unit disc graph induced by $P \cup F$. We show that the vertex 1-centre provides a 7-approximation of the geometric 1-centre and that a vertex $k$-centre provides a 13-approximation of the geometric $k$-centre, resulting in an $O(kn)$-time 26-approximation algorithm. We describe $O(n^2m)$-time and $O(n^3)$-time algorithms, respectively, for finding exact and approximate geometric 1-centres, and an $O(mn^{2k})$-time algorithm for finding a geometric $k$-centre for any fixed $k$. We show that the problem is NP-hard when $k$ is an arbitrary input parameter. Finally, we describe an $O(n)$-time algorithm for finding a geometric $k$-centre in one dimension.

*Keywords:* unit disc graph, $k$-centre, intersection graph, facility location, gateway placement, wireless networks

## 1. Introduction

### 1.1. Motivation

In a wireless sensor network, sensor nodes collect and send data to *sink* nodes, which may either be the users of the data, or gateways to another (possibly wired) network through which a remote user can access the data. Sensor nodes perform a sensing function as well as a routing and forwarding function

[☆] Some of these results appeared in preliminary form at the ACM SIGACT-SIGOPS International Workshop on Foundations of Mobile Computing [19].

[*] Corresponding author

*Email addresses:* durocher@cs.umanitoba.ca (S. Durocher), krjampan@cs.uwaterloo.ca (K. R. Jampani), alubiw@uwaterloo.ca (A. Lubiw), lata@cse.concordia.ca (L. Narayanan)

[1] Work supported in part by NSERC.

to move data to sink nodes. Since sensor nodes are battery powered, conserving and making efficient use of energy is an important consideration for all network protocols. In particular, forwarding packets depletes battery power at all nodes on a routing path, a problem that is made worse if sink nodes are poorly positioned, resulting in longer path lengths to sink nodes. Similarly, much of the traffic in a wireless mesh network passes through *gateway* nodes that provide connectivity to exterior networks such as the Internet [3]. To optimize bandwidth usage, it is important to minimize the path length between nodes and gateways [3].

This motivates the problem of optimal sink placement in a wireless sensor network or gateway placement in a wireless mesh network. In this paper, we model these problems as a *facility location* problem, in which network nodes correspond to clients, and gateways or sink nodes correspond to facilities. A wireless network is often modelled by a unit disc graph (e.g., [7, 23, 26, 29, 30]) where the nodes are represented by points on the plane and a node $u$ is connected to every node located in the unit disc centred at $u$. Given a set of points $P$ in the plane, we consider the problem of finding a set $F$ of $k$ points in the plane that minimizes the maximum graph distance between any point in $P$ and the nearest point in $F$ in the unit disc graph induced by $P \cup F$. Although this problem is similar to the Euclidean $k$-centre and vertex $k$-centre problems (see Section 3), this version of the problem incorporates both geometric and graph-theoretic constraints, resulting in a new problem which we call the *geometric $k$-centre problem* for unit disc graphs[2].

In the geometric $k$-centre problem, facilities may be selected from anywhere in the plane (as in the Euclidean $k$-centre problem) whereas the distance between clients and facilities is measured by graph distance (as in the vertex $k$-centre problem). Thus the geometric $k$-centre problem is neither set solely in the host metric space nor on a graph. Given this new setting, existing solutions to the $k$-centre problem on graphs or in Euclidean space do not necessarily provide solutions to the geometric $k$-centre problem.

### *1.2. Overview of Results*

After establishing properties of arrangements of sets of unit discs (Section 4), we show that the vertex 1-centre provides a 7-approximation of the geometric 1-centre (Section 5.1). Next we show that a vertex $k$-centre provides a 13-approximation of the geometric $k$-centre, resulting in an $O(kn)$-time 26-approximation algorithm (Section 5.2). We describe $O(n^2m)$-time and $O(n^3)$-time algorithms, respectively, for finding exact and approximate geometric 1-centres (Sections 6.1 and 6.2). Our technique generalizes to an $O(mn^{2k})$-time algorithm for finding a geometric $k$-centre for any fixed $k$ (Section 6.3). When $k$ is an arbitrary input parameter, we show that the geometric $k$-centre problem

[2]In this paper, the term *geometric $k$-centre* refers exclusively to Definition 2 (see Section 2). In the literature, this term is sometimes used to refer to the Euclidean $k$-centre (Definition 4).

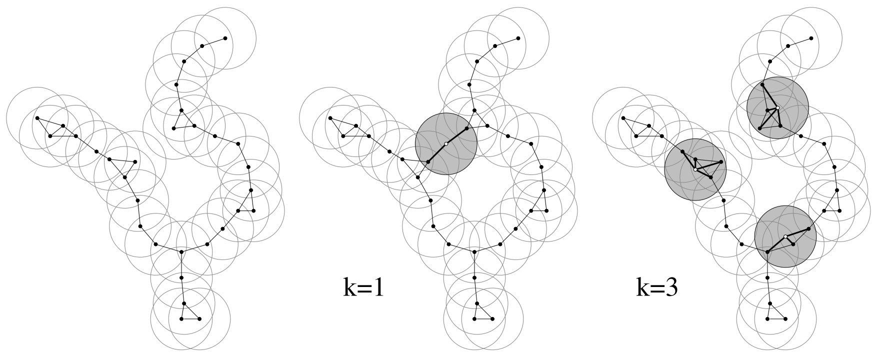


Figure 1: **(Left)** A set of points $P$, the corresponding set $\mathrm{Disc}(P)$, and $\mathrm{UDG}(P)$. **(Middle)** The point at the centre of the shaded unit disc is a geometric 1-centre of $P$. The corresponding graph $\mathrm{UDG}(P \cup F)$ is illustrated. **(Right)** The set of points at the centres of the three shaded unit discs is a geometric 3-centre of $P$. The corresponding graph $\mathrm{UDG}(P \cup F)$ is illustrated.

is NP-hard on unit disc graphs (Section 7). Finally, we describe an $O(n)$-time algorithm for finding a geometric $k$-centre in one dimension (Section 8).

## 2. Definitions

We employ standard graph-theoretic notation for a graph $G$, where $V(G)$ denotes the vertex set of $G$; $E(G)$ denotes the edge set of $G$; for each vertex $v \in V(G)$, $\mathrm{Adj}(v) = \{u \mid \{u, v\} \in E(G)\}$ denotes the set of vertices adjacent to $v$; and $\deg(v) = |\mathrm{Adj}(v)|$ denotes its degree. Let $\mathrm{dist}_G(u, v)$ denote the unweighted graph distance in $G$ between vertices $u$ and $v$ in $V(G)$. Let $\mathrm{dist}(p, q)$ denote the Euclidean ($\ell_2$) distance between points $p$ and $q$ in $\mathbb{R}^d$.

Given a point $p \in \mathbb{R}^2$, let $\mathrm{Disc}_r(p)$ denote the disc of radius $r$ centred at $p$. Similarly, given a set of points $P \subseteq \mathbb{R}^2$, let $\mathrm{Disc}_r(P)$ denote the corresponding set of discs. When $r = 1$ we omit the subscript $r$.

Although the geometric $k$-centre problem can be applied to several classes of geometrically-defined graphs, we focus primarily on graphs commonly used to model the topology of wireless networks: unit disc graphs.

**Definition 1 (Unit Disc Graph).** *Given a set of points $P$ in $\mathbb{R}^2$, the* unit disc graph *induced by $P$, denoted* $\mathrm{UDG}(P)$, *is an embedded graph with vertex set $P$ and edge set $\{\{u, v\} \mid \{u, v\} \subseteq P \text{ and } \mathrm{dist}(u, v) \leq 1\}$.*

That is, vertices $p$ and $q$ in $P$ are adjacent in $\mathrm{UDG}(P)$ if and only if $q \in \mathrm{Disc}(p)$. See the example in Figure 1. Equivalently, vertices $p$ and $q$ in $P$ are adjacent in $\mathrm{UDG}(P)$ if and only if $\mathrm{Disc}_{1/2}(p) \cap \mathrm{Disc}_{1/2}(q) \neq \varnothing$. Thus, a unit disc graph is an intersection graph. Note, a unit disc graph is sometimes defined as the intersection graph of a set of unit discs or of a set of equal-radius discs; all of these definitions are equivalent upon scaling.

If $P \subseteq \mathbb{Z}^2$, then $\mathrm{UDG}(P)$ is a *grid graph*. A unit disc graph is not necessarily planar and its maximum degree can be as large as $|P| - 1$. A grid graph, on the other hand, is planar and has maximum degree at most four. Naturally, the definition of a unit disc graph generalizes to three or higher dimensions as a *unit ball graph* and to one dimension as a *unit interval graph*, both of which can be considered with respect to the geometric $k$-centre problem.

Given a set of points $P \subseteq \mathbb{R}^2$ and $R = \mathrm{Disc}(P)$, the *arrangement* induced by $R$ and denoted $\mathcal{A}_R$, is a set of *cells*, each of which is a maximally connected region in the space formed by removing the boundaries of the discs in $R$ from $\mathbb{R}^2$. We define the *dual arrangement graph* of $R$ as the planar graph $G$ whose vertex set is $\mathcal{A}_R$ and whose edges connect adjacent cells in $\mathcal{A}_R$. We regard $G$ as a directed graph, with $(C_a, C_b) \in E(G)$ if and only if the set of discs in $R$ containing $C_a$ is a subset of the discs in $R$ containing $C_b$. See Figures 2A and 2B.

Next we define a geometric $k$-centre of a unit disc graph:

**Definition 2 (Geometric $k$-Centre).** *Given a set of points $P = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^2$ and a positive integer $k$, a* geometric $k$-centre *of $P$ is a set of points $F = \{f_1, \ldots, f_k\}$ in $\mathbb{R}^2$, such that $F$ minimizes* eccentricity *relative to $P$, denoted* $\mathrm{ecc}_G(P, F)$*, where*

$$\mathrm{ecc}_G(P, F) = \max_{p_i \in P} \min_{f_j \in F} \mathrm{dist}_{\mathrm{UDG}(P \cup F)}(p_i, f_j). \tag{1}$$

When $F$ is a geometric $k$-centre, we refer to the value of (1) as the *geometric $k$-radius* of $P$. In the facility location literature, $P$ typically represents a set of *clients* (the input defining a problem instance) and $F$ represents a set of *facilities* (a solution to the problem instance); we use these terms to differentiate between points in $P$ and those in $F$. With respect to our discussion of geometric $k$-centres on unit disc graphs, we identify the location of a client or facility by the point $p$ at the centre of the corresponding disc.

The geometric $k$-centre problem is related to the vertex $k$-centre problem:

**Definition 3 (Vertex $k$-Centre).** *Given a graph $G$ and a positive integer $k$, a* vertex $k$-centre *of $G$ is a set of vertices $F = \{v_1, \ldots, v_k\} \subseteq V(G)$ that minimizes*

$$\max_{u \in V(G)} \min_{v_j \in F} \mathrm{dist}_G(u, v_j). \tag{2}$$

When $F$ is a vertex $k$-centre, we refer to the value of (2) as the *vertex $k$-radius* of $G$. A vertex $k$-centre is often called simply a *$k$-centre*; we include the prefix "vertex" to distinguish it from a geometric $k$-centre. The vertex $k$-centre problem has been studied extensively (see Section 3). Although the vertex $k$-centre problem can be applied to a unit disc graph, these two $k$-centre problems differ in that the set of facilities need not be a subset of the set of clients under the geometric version of the problem. As we show in Section 5, a vertex 1-centre and $k$-centre provide respective 7- and 13-approximations of the geometric 1-centre and $k$-centre.

Finally, the vertex and geometric $k$-centre problems are related to the Euclidean $k$-centre problem:

**Definition 4 (Euclidean $k$-Centre).** *Given a set $P$ of points in $\mathbb{R}^2$ and a positive integer $k$, a* Euclidean $k$-centre *of $G$ is a set of points $F = \{f_1, \ldots, f_k\} \subseteq \mathbb{R}^2$ that minimizes*

$$\max_{p \in P} \min_{f_i \in F} \operatorname{dist}(p, f_i), \tag{3}$$

*where* $\operatorname{dist}(\cdot, \cdot)$ *denotes the Euclidean ($\ell_2$) metric.*

A Euclidean 1-centre need not lie within unit distance of any point in $P$. Consequently, unlike the vertex $k$-centre problem, a solution to the Euclidean $k$-centre problem does not guarantee any approximation to the geometric $k$-centre problem.

## 3. Related Work

### 3.1. Vertex k-Centre

Given a graph $G$, Hakimi and Kariv [28] describe an algorithm to find a vertex 1-centre in $O(mn + n^2 \log n)$ time, where $n = |V(G)|$ and $m = |E(G)|$. A vertex 1-centre can also be found by calculating the unweighted all-pairs shortest path distances and identifying the vertex for which the maximum distance is minimized; as shown by Chan [11], this can be done in $O(mn/\log n)$ time if $m > \log^2 n$, $O(mn \log\log n/\log n)$ time if $m > n \log\log n$, and $O(n^2 \log^2 \log n/\log n)$ time if $m \leq n \log\log n$. When $k$ is fixed, a vertex $k$-centre can be found in $O(m^k n^k \log n)$ time [39]. When $k$ is an arbitrary input parameter, the problem is NP-hard [28]. Finding a $(2-\epsilon)$-approximation remains NP-hard for any $\epsilon > 0$, even for unweighted planar graphs of maximum degree 3 [25, 36]; an $O(kn)$-time 2-approximation algorithm exists [22, 24] using a greedy approach: select an arbitrary vertex as the first centre. Then for each $i \in \{2, \ldots, k\}$, let the $i$th centre be a vertex whose minimum distance to the previous centres is maximized.

### 3.2. Unit Disc Graphs

Clark et al. [16] give hardness results for several problems on unit disc graphs, including the minimum dominating set problem (which we use as the basis for our hardness reduction in Section 7). They mention an earlier result by Masuyama et al. [32] regarding hardness of the vertex $k$-centre problem on unit disc graphs. Marathe et al. [31] describe approximation algorithms for NP-hard problems on unit disc graphs, including a 5-approximation for the minimum dominating set problem. In addition, they demonstrate the following property:

**Lemma 1 (Marathe et al. [31]).** *Given any finite set of points $P$ and any $p \in P$, every independent set of* $\operatorname{Adj}(p)$ *in* $\operatorname{UDG}(P)$ *has cardinality at most five.*

Given $P \subseteq \mathbb{R}^2$, Breu [8] describes an $O(m + n \log n)$-time algorithm for constructing $\operatorname{UDG}(P)$ and an $O(n \log n)$-time algorithm for enumerating the connected components of $\operatorname{UDG}(P)$. Breu and Kirkpatrick [9] show it is NP-hard to decide whether a graph is a unit disc graph. That is, given only the combinatorial description for a UDG, it is NP-hard to find a unit disc embedding in the plane.

The difficulty in finding a geometric $k$-centre of a unit disc graph arises from the geometric constraints implied by an embedding; given only a combinatorial description for a graph, the addition of a universal vertex trivially solves the problem. As such, we assume knowledge of the graph's planar embedding in a problem instance.

### *3.3. Euclidean k-Centre*

A Euclidean 1-centre can be found in linear time [2, 13]. Welzl [43] gives a simpler randomized algorithm with $O(n)$ expected time. At present, the fastest algorithm for finding a Euclidean 2-centre requires $O(n \log^2 n \log^2 \log n)$ time in the worst case [10]. When $k$ is fixed, a Euclidean $k$-centre can be found in $n^{O(\sqrt{k})}$ time [27]. When $k$ is an arbitrary input parameter, the problem is NP-hard [33]. Feder and Greene [21] show that finding an $\epsilon$-approximation remains NP-hard for any $\epsilon < (1+\sqrt{7})/2 \approx 1.8229$. An $O(n \log k)$-time 2-approximation algorithm exists [21] using a greedy approach similar to the 2-approximation algorithm for the vertex $k$-centre. The above results refer to the Euclidean $k$-centre in the plane; see [1, 18] for a discussion of the Euclidean $k$-centre in higher dimensions.

### *3.4. Geometric Sink/Relay Placement*

Similar to the geometric $k$-centre problem in which $k$ is fixed and the objective is to minimize the geometric $k$-radius, Mihandoust and Narayanan [34] consider the related $h$-hop covering set problem on a unit disc graph, in which the maximum $k$-radius is fixed and the objective is to minimize $k$. They provide PTASs for several variants of this problem. Aoun et al. [3] follow a similar approach for gateway placement in wireless mesh networks. Efrat et al. [20] consider the related relay placement problem, in which the objective is to add the minimum number of facilities (relays) such that the resulting network is connected. They consider a more general model in which the range of communication of relays and network nodes may differ.

## 4. The Arrangement of a Set of Unit Discs

We begin by examining properties of arrangements of unit discs. Throughout Sections 4 to 7, $P$ denotes an arbitrary set of points in $\mathbb{R}^2$, $R = \text{Disc}(P)$, $\mathcal{A}_R$ denotes the arrangement induced by $\text{Disc}(P)$, $n = |P|$, and $m = |E(\text{UDG}(P))|$.

Definition 1, the definition of an arrangement, and (1) imply the following observation:

**Observation 2.** *Given a set $P \subseteq \mathbb{R}^2$ and points $f_1$ and $f_2$ in the same cell of the arrangement of* $\text{Disc}(P)$*,*

$$\text{ecc}_{\text{UDG}(P\cup\{f_1\})}(P, \{f_1\}) = \text{ecc}_{\text{UDG}(P\cup\{f_2\})}(P, \{f_2\}).$$

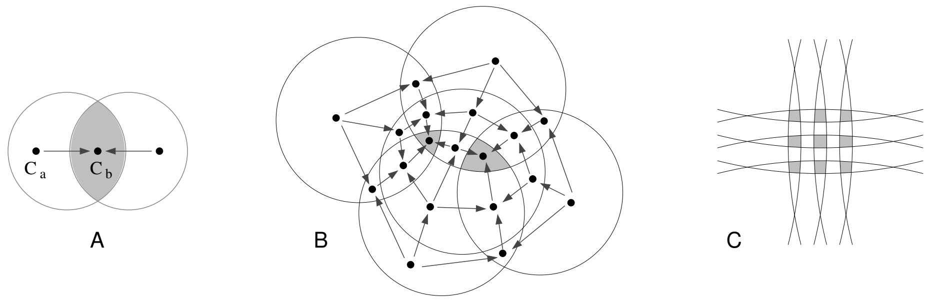


Figure 2: **A.** The edges of the dual arrangement graph can be directed such that $(C_a, C_b) \in E(G)$ if and only if for every $p \in P$, $C_a \subseteq \text{Disc}(p) \Rightarrow C_b \subseteq \text{Disc}(p)$. **B.** The arrangement induced by these five unit discs partitions the plane into twenty cells, including the exterior face. The partial order of the corresponding dual graph has four sources and two sinks. To select locations for a facility, it suffices to consider the sinks, which correspond precisely to convex cells (shaded). **C.** This example due to Tóth [42] shows an arrangement induced by $n$ unit discs that has $\Omega(n^2)$ convex cells.

Therefore, if point $f_1$ is a geometric 1-centre of $P$, then any point in the same cell as $f_1$ is also a geometric 1-centre of $P$. Consequently, to identify a geometric 1-centre of $P$ it suffices to consider one point from every cell in $\mathcal{A}_R$.

By Propositions 3 and 4, the number of cells in any arrangement of discs in the plane is $\Theta(n^2)$ in the worst case; this value is directly proportional to the running times of algorithms we describe in Sections 6.1 and 6.3.

**Proposition 3 (Steiner 1881 [38]).** *An arrangement of $n$ discs in $\mathbb{R}^2$ contains at most $n^2 - n + 2$ cells. This bound is tight.*

Recall that the edges of the dual arrangement graph $G$ can be directed such that for any cells $C_a$ and $C_b$ in $\mathcal{A}_R$, $(C_a, C_b) \in E(G)$ if and only if for every $p \in P$, $C_a \subseteq \text{Disc}(p) \Rightarrow C_b \subseteq \text{Disc}(p)$. Since a facility not located in any unit disc will be disconnected from $\text{UDG}(P)$, we omit the exterior face from $V(G)$. See Figures 2A and 2B. Observe that $G$ is a partial order relation. Furthermore, for any cells $\{C_a, C_b\} \subseteq \mathcal{A}_R$ and any points $f_a \in C_a$ and $f_b \in C_b$, if $(C_a, C_b) \in E(G)$, then $\text{UDG}(P \cup \{f_a\})$ is a subgraph of $\text{UDG}(P \cup \{f_b\})$. Hence,

$$\text{ecc}_{\text{UDG}(P \cup \{f_b\})}(P, \{f_b\}) \leq \text{ecc}_{\text{UDG}(P \cup \{f_a\})}(P, \{f_a\}).$$

Consequently, when selecting a position for a 1-centre, it suffices to consider only cells in $\mathcal{A}_R$ that are sinks with respect to the partial order induced by $\mathcal{A}_R$. The sinks correspond exactly to the convex cells in $\mathcal{A}_R$. One might hope that the number of sinks is asymptotically less than the total number of cells; this is not always the case, as shown by the following proposition based on an example suggested by Tóth [42]. See Figure 2C.

**Proposition 4.** *For any $n \in \mathbb{Z}^+$, there exists an arrangement of $n$ unit discs in $\mathbb{R}^2$ for which the number of convex cells is at least $\lfloor n/4 \rfloor^2$.*

**Proof.** Choose any $n$.

CASE 1. Suppose $n \bmod 4 = 0$. Position two unit discs such that their centres are distance $2 - 64/(16+n^2)$ apart. It is straightforward to show that their intersection is a lune of width $64/(16+n^2)$ and height $16n/(16+n^2)$. Observe that the height is $n/4$ times the width. Therefore, $n$ discs can be positioned such that $n/4$ vertical lunes each intersect $n/4$ horizontal lunes. See Figure 2C. Each lune is convex and, therefore, the intersection of two lunes is also convex, resulting in at least $n^2/16$ convex cells.

CASE 2. Suppose $n = 4j + i$ for some $j \in \mathbb{Z}$ and some $i \in \{1, 2, 3\}$. Given any sets of unit discs $R_1$ and $R_2$, the number of convex cells in $\mathcal{A}_{R_1 \cup R_2}$ is greater than or equal to the number of convex cells in $\mathcal{A}_{R_1}$. The result follows by Case 1 since $\lfloor (4j+i)/4 \rfloor^2 = \lfloor (4j)/4 \rfloor^2$. $\square$

## 5. Approximating by a Vertex $k$-Centre

A geometric $k$-centre of a unit disc graph can be approximated by a vertex $k$-centre in the corresponding unit disc graph. Facilities in a geometric $k$-centre can be positioned anywhere in the plane while facilities in a vertex $k$-centre must coincide with clients. Consequently, the geometric $k$-radius of a unit disc graph is at most the vertex $k$-radius. Of course, the geometric $k$-radius can be less than the vertex $k$-radius. Theorems 5 and 8 bound the ratio between the two radii when $k = 1$ and when $k$ is arbitrary, respectively. As such, an algorithm that returns a vertex $k$-centre can be used to approximate a geometric $k$-centre if the underlying unit disc graph is connected.

### 5.1. Approximating a Geometric 1-Centre

**Theorem 5.** *If* $\mathrm{UDG}(P)$ *is connected, then the vertex 1-radius of* $\mathrm{UDG}(P)$ *is at most* $5r + 2$*, where* $r$ *denotes its geometric 1-radius. This bound is tight asymptotically.*

**Proof.** Choose any finite set $P \subseteq \mathbb{R}^2$. Let $f \in \mathbb{R}^2$ be a geometric 1-centre of $P$ and let $r$ denote the corresponding geometric 1-radius. Let $A_f = \mathrm{Adj}(f)$ and let $C_f$ be a maximal independent set of $A_f$ in $\mathrm{UDG}(P)$. By Lemma 1, $|C_f| \leq 5$. For every $d \in A_f$, there exists a $c \in C_f$ such that $\mathrm{dist}_{\mathrm{UDG}(P)}(c, d) \leq 1$.

We begin by showing that $C_f$ is a vertex 5-centre of $\mathrm{UDG}(P)$ with radius $r$. For every $p \in P$, every shortest path from $f$ to $p$ in $\mathrm{UDG}(P \cup \{f\})$ must pass through some vertex $d \in A_f$. Therefore, for all $p \in P$, there exists a $d \in A_f$ such that

$$\begin{aligned}
\mathrm{dist}_{\mathrm{UDG}(P\cup\{f\})}(f, p) &= \mathrm{dist}_{\mathrm{UDG}(P\cup\{f\})}(f, d) + \mathrm{dist}_{\mathrm{UDG}(P\cup\{f\})}(d, p) \\
&= 1 + \mathrm{dist}_{\mathrm{UDG}(P\cup\{f\})}(d, p) \\
&= 1 + \mathrm{dist}_{\mathrm{UDG}(P)}(d, p) \\
&\leq r,
\end{aligned}$$

since $f$ is a geometric 1-centre of $P$ with radius $r$. Therefore,

$$\forall p \in P,\ \exists d \in A_f \ \text{ s.t. } \operatorname{dist}_{\mathrm{UDG}(P)}(d, p) \leq r - 1. \tag{4}$$

Furthermore, by (4), for all $p \in P$, there exist a $c \in C_f$ and a $d \in A_f$ such that

$$\begin{aligned} \operatorname{dist}_{\mathrm{UDG}(P)}(c, p) &\leq \operatorname{dist}_{\mathrm{UDG}(P)}(c, d) + \operatorname{dist}_{\mathrm{UDG}(P)}(d, p) \\ &\leq 1 + \operatorname{dist}_{\mathrm{UDG}(P)}(d, p) \\ &\leq r. \end{aligned} \tag{5}$$

By (5), $C_f$ is a vertex 5-centre of UDG($P$) of radius at most $r$.

Next, we show there exists a vertex $a \in C_f$ that is a vertex 1-centre of UDG($P$) with radius at most $5r+2$. In other words, we reduce the cardinality of the set of facilities at the expense of increasing the radius. Construct a graph $G$ with vertex set $C_f$ such that nodes $u$ and $v$ are adjacent in $G$ if and only if $\operatorname{dist}(u, v)_{\mathrm{UDG}(P)} \leq 2r + 1$. We claim that $G$ is connected. Suppose otherwise. By assumption, UDG($P$) is connected. Consider a shortest path in UDG($P$) between two vertices in $C_f$ that lie in separate connected components of $G$. This path has length at least $2r+2$. Thus, the midpoint of the path has distance at least $r+1$ from every $c \in C_f$, contradicting (5). Therefore $G$ must be connected.

Since $G$ is a connected graph and $|V(G)| \leq 5$, $G$ has a vertex 1-centre $a$ of radius at most 2. Therefore,

$$\begin{aligned} &\forall c \in C_f,\ \operatorname{dist}_G(c, a) \leq 2 \\ \Rightarrow\ &\forall c \in C_f,\ \operatorname{dist}_{\mathrm{UDG}(P)}(c, a) \leq 2(2r + 1). \end{aligned} \tag{6}$$

By (5) and (6),

$$\begin{aligned} \forall p \in P,\ \exists c \in C_f \text{ s.t. } \operatorname{dist}_{\mathrm{UDG}(P)}(a, p) &\leq \operatorname{dist}_{\mathrm{UDG}(P)}(a, c) + \operatorname{dist}_{\mathrm{UDG}(P)}(c, p) \\ &\leq 5r + 2. \end{aligned} \tag{7}$$

Therefore, $a$ is a vertex 1-centre of UDG($P$) with vertex 1-radius at most $5r+2$.

This bound is realized in the limit as $s \to \infty$ by the unit disc graph $G_s$ illustrated in Figure 3. Graph $G_s$ consists of a facility $f$ with five independent neighbours, each of which is adjacent to two paths of length $s$. For any $s \geq 2$, $G_s$ has geometric 1-radius $2 + \lfloor s/2 \rfloor$ (realized by the geometric 1-centre located at $f$) and vertex 1-radius $\lfloor 5s/2 \rfloor$. $\square$

Observe that $5r + 2 \leq 7r$ if $r \geq 1$. If $r = 0$, then $|P| \leq 1$ and $P$ is both a geometric 1-centre and a vertex 1-centre of $P$. Consequently, a vertex 1-centre of $P$ provides a 7-approximation of a geometric 1-centre when UDG($P$) is connected. See Section 3.1 for a discussion of algorithms for finding a vertex 1-centre.

### *5.2. Approximating a Geometric k-Centre*

In this section we generalize the result of Theorem 5 to an arbitrary number of facilities $k$. Theorem 8 shows that if UDG($P$) is connected, then the vertex

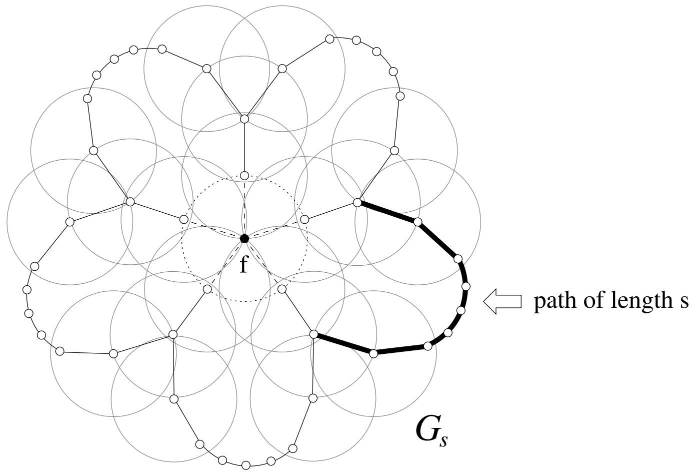


Figure 3: illustration in support of Theorem 5

$k$-radius of UDG($P$) is at most $9r + 4$, where $r$ denotes the geometric $k$-radius of $P$. The proof follows by an argument similar to that of Theorem 5. Starting with a geometric $k$-centre $F$ of radius $r$, we construct a maximal independent set $C$ of the set of neighbours of $F$ in UDG($P \cup F$). Set $C$ is a vertex $(5k)$-centre of UDG($P$) with radius at most $r$. We show there exists a subset of $C$ that is a vertex $k$-centre of UDG($P$) with radius at most $9r + 4$. Our proof refers to Lemmas 6 and 7 that establish a trade off between $k$, the cardinality of the set of facilities, and the corresponding vertex $k$-radius.

**Lemma 6.** *Given any positive integers $n$ and $d$, and any connected graph $G$ on $n$ vertices, $G$ has a vertex $\lceil n/d \rceil$-centre of radius at most $d - 1$.*

**Proof.** We use induction on $n$. If $n = 1$, $d = 1$, or $d \geq n$, then the result holds trivially. Choose any $n > 1$ and assume the result holds for all $d$ and all connected graphs on fewer than $n$ vertices. Choose any connected graph $G$ on $n$ vertices and any $d$. Let $T$ denote any rooted spanning tree of $G$ and let $l$ denote a deepest leaf of $T$. Let $v$ denote the $d$th vertex along the path from $l$ to the root or let $v$ be the root if this path has length less than $d$. Let $T_v$ be the subtree of $T$ rooted at $v$. Every vertex in $T_v$ has depth at most $d - 1$; that is,

$$\forall u \in V(T_v),\ \operatorname{dist}_G(u, v) \leq d - 1. \tag{8}$$

Let $G'$ denote the subgraph of $G$ induced by $V(T) \setminus V(T_v)$. Let $n' = |V(G')|$.

Case 1. Suppose $n' = 0$. That is, $T_v$ spans $G$. Therefore, $v$ is a 1-centre of $G$ with radius at most $d - 1$ and the result holds.

Case 2. Suppose $n' \geq 1$. Since $T_v$ is a rooted subtree of $T$, $G'$ is connected. Since $|V(T_v)| \geq d$, therefore $n' + d \leq n$ and

$$\left\lceil \frac{n'}{d} \right\rceil + 1 \leq \left\lceil \frac{n}{d} \right\rceil. \tag{9}$$

By our inductive hypothesis, $G'$ has a vertex $\lceil n'/d \rceil$-centre of radius $d-1$; let $F$ denote the corresponding set of facilities. By (8) and (9), the set $F \cup \{v\}$ is a $\lceil n/d \rceil$-centre of $G$ with radius $d-1$. $\square$

**Lemma 7.** *For any positive integers $\kappa$ and $d$, if a connected graph $G$ has a vertex $\kappa$-centre of radius $r$, then $G$ has a vertex $\lceil \kappa/d \rceil$-centre of radius at most $r(2d-1)+(d-1)$.*

**Proof.** Let $F$ denote a vertex $\kappa$-centre of $G$ and let $r$ denote its radius. Therefore,

$$\forall v \in V(G),\ \exists f \in F \text{ s.t. } \operatorname{dist}_G(v, f) \leq r. \tag{10}$$

Let $G'$ denote a graph with vertex set $V(G') = F$, where edge $\{u, v\} \in E(G')$ if and only if $\operatorname{dist}_G(u, v) \leq 2r+1$.

Suppose $G'$ is disconnected. Consider a shortest path $S$ in $G$ connecting two components of $G'$, where the endpoints of $S$ are vertices in $F$. Since $S$ has length at least $2r+2$, there must be some vertex of $S$ whose distance to any vertex in $F$ is at least $r+1$, deriving a contradiction. Therefore, $G'$ is connected.

By Lemma 6, $G'$ has a vertex $\lceil \kappa/d \rceil$-centre of radius $d-1$; let $F'$ denote the corresponding set of facilities. Therefore,

$$\begin{aligned}
& \forall f \in F,\ \exists f' \in F' \text{ s.t. } \operatorname{dist}_{G'}(f, f') \leq d-1, \\
\Rightarrow\ & \forall f \in F,\ \exists f' \in F' \text{ s.t. } \operatorname{dist}_{G}(f, f') \leq (d-1)(2r+1) \text{ by definition of } G', \\
\Rightarrow\ & \forall v \in V(G),\ \exists f' \in F' \text{ s.t. } \operatorname{dist}_{G}(v, f') \leq r + (d-1)(2r+1) \text{ by (10).}
\end{aligned}$$

Therefore, $F'$ is a vertex $\lceil \kappa/d \rceil$-centre of $G$ with radius at most $r(2d-1)+(d-1)$. $\square$

We now establish our result on the approximation of a geometric $k$-centre by a vertex $k$-centre.

**Theorem 8.** *If $\operatorname{UDG}(P)$ is connected, then the vertex $k$-radius of $\operatorname{UDG}(P)$ is at most $9r+4$, where $r$ denotes its geometric $k$-radius.*

**Proof.** Choose any finite set $P \subseteq \mathbb{R}^2$ and any positive integer $k$ such that $\operatorname{UDG}(P)$ is connected. Let $F = \{f_i, \ldots, f_k\} \subseteq \mathbb{R}^2$ be a geometric $k$-centre of $P$ and let $r$ denote the corresponding geometric $k$-radius. If $k \geq n$, then $F = P$ is both a geometric $k$-centre and a vertex $k$-centre of $\operatorname{UDG}(P)$; in this case the result holds trivially. Therefore, suppose $k < n$.

For every $f_i \in F$, let $C_i$ be a maximal independent set of $\operatorname{Adj}(f_i)$ in $\operatorname{UDG}(P \cup F)$. By Lemma 1, $|C_i| \leq 5$. By an argument analogous to (5) in the proof of Theorem 5, the set $\bigcup_{i=1}^{k} C_i$ is a vertex $(5k)$-centre of $\operatorname{UDG}(P)$ with radius at most $r$. The result follows by Lemma 7. $\square$

Observe that $9r+4 \leq 13r$ if $r \geq 1$. If $r = 0$, then $|P| \leq k$ and set $P$ is both a geometric $k$-centre and a vertex $k$-centre of $P$. Therefore, a vertex $k$-centre of $P$ provides a 13-approximation of a geometric $k$-centre when $\operatorname{UDG}(P)$

is connected. As noted in Section 3.1, finding a vertex $k$-centre is NP-hard, but there is a simple greedy 2-approximation algorithm [22] that runs in time $O(kn)$. Consequently, Corollary 9 follows from Theorem 8:

**Corollary 9.** *If* UDG($P$) *is connected, then a 26-approximation to the geometric $k$-centre can be found in $O(kn)$ time, where $n = |P|$.*

Theorems 5 and 8 and Corollary 9 require that UDG($P$) be connected. These results do not necessarily hold when this condition is not met. In particular, a disconnected graph consisting of greater than $k$ components does not have a vertex $k$-centre; a geometric $k$-centre might exist, however, since disconnected components in UDG($P$) can be covered by a single facility in UDG($P \cup F$). This gives the following observation:

**Observation 10.** *If* UDG($P$) *is disconnected, then a vertex $k$-centre cannot guarantee any bounded approximation of a geometric $k$-centre.*

We now consider a lower bound on the approximation factor of the vertex $k$-centre. We begin with a specific example for $k = 9$, in which the geometric 9-radius is $d$, the vertex 9-radius is $9d - 4$, and the greedy 2-approximation algorithm on the unit disc graph gives a 9-radius of $18d - 9$. The graph is based on a star with ten arms, each of length $9d - 4$. In order to realize this graph as a unit disc graph, we must add a cycle of edges connecting vertices adjacent to the hub. See Figure 4. A vertex 9-centre misses at least one of the arms, and therefore the leaf of that arm has distance at least $9d - 4$ from any centre. The greedy 2-approximation algorithm for the vertex $k$-centre, with initial facility placement at a leaf, places all nine centres at leaves, and therefore the final remaining leaf has distance $2(9d - 4) - 1 = 18d - 9$ from any centre; one unit is subtracted from the value $2(9d - 4)$ due to the cycle of edges around the hub, allowing a shortcut on the shortest path between leaves of adjacent arms.

It remains to show how to arrange the discs in the plane to achieve a geometric 9-radius of $d$. Distinguish four *petal* nodes along each arm, dividing the arm into subpaths of lengths $2d$, $2d-1$, $2d-1$, $2d-1$, and $d-1$, respectively, as shown for one arm in Figure 4. Arrange these $4 \times 10$ petals into eight *flowers* of five petals each in such a way that one new disc can intersect all five petals of a flower. See Figure 5. This accounts for eight centres; place the ninth centre at the hub of the star. This gives a 9-radius of $d$: every petal has distance one to a centre, and any vertex either has distance at most $d$ to the hub, or has distance at most $d - 1$ to a petal, hence distance at most $d$ to a centre.

Although the arrangement shown in Figure 5 cannot be realized for very small values of $d$, we claim that it is feasible for larger values of $d$.

We also claim that the example can be generalized to show that for any fixed $k \geq 9$, there is a family of examples (with $d$ growing) that realizes an asymptotic ratio of 9 between the geometric $k$-radius and the vertex $k$-radius, and an asymptotic ratio of 18 between the geometric $k$-radius and the vertex $k$-radius found by the greedy 2-approximation algorithm. The general example is based on a star graph with $t$ arms each of length $9d-4$, where $t = \lfloor 5(k-1)/4 \rfloor$.

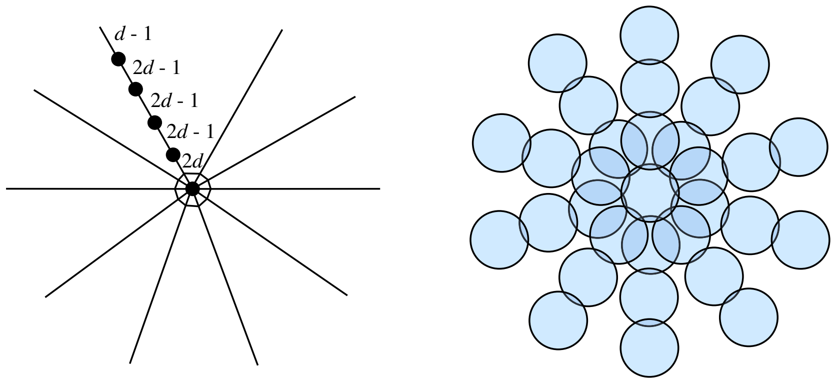


Figure 4: **(Left)** The star graph for $k = 9$ realizes the worst-case ratio between the geometric $k$-radius and the vertex $k$-radius of the corresponding unit disc graph. Each of the ten arms of the star has length $9d - 4$. **(Right)** A close-up view of the discs forming the hub of the star is shown (illustrated as the intersection graph of a set of discs of radius 1/2).

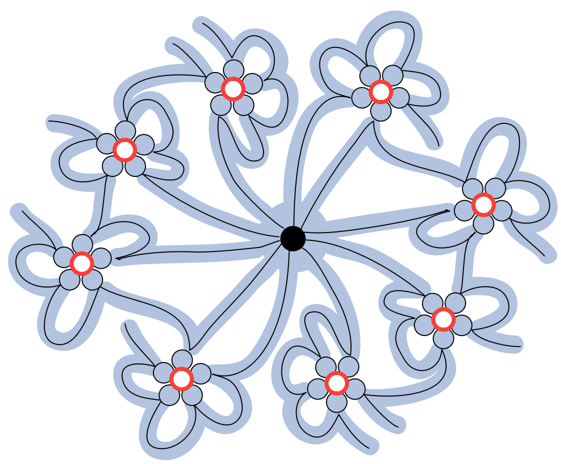

Figure 5: The geometric arrangement of the star from Figure 4. The large grey disc at the hub and the grey paths indicate multiple unit discs. The hub and the 40 petals are drawn as distinct discs.

Note that for $k \geq 9$ we have $t \geq k+1$. The argument is basically the same as before, so we only point out the differences. When the star is constructed of unit discs, the arms will intersect near the hub, but only within a disc of radius $O(t)$. This shortens the distance from a leaf to another arm of the star by $O(t)$ but for $k$ (and hence $t$) fixed, and $d$ growing, this does not affect the asymptotic behaviour.

## 6. Algorithmic Results

Building on our observations from Section 4, we describe algorithms for finding a geometric 1-centre in $O(n^2 m)$ worst-case time and a nearly-optimal approximate geometric 1-centre in $O(n^3)$ time; the resulting approximate solution has eccentricity at most one greater than the geometric 1-radius, corresponding to an additive approximation factor of at most one. Finally, we describe a generalization of our algorithm to find a geometric $k$-centre for any fixed $k$ in $O(mn^{2k})$ worst-case time.

### *6.1. Finding a Geometric 1-Centre*

Recall our discussion of properties of arrangements of unit discs in Section 4. Chazelle and Lee [12] describe how to build the arrangement graph of a set of $n$ unit discs (and its dual) in $O(n^2)$ time. As the graph is constructed, for each cell $C$ we maintain a list of discs within which $C$ is contained; these correspond to the neighbours of $f$ in $\mathrm{UDG}(P \cup \{f\})$, where $f$ is any point in $C$. Since a disc's centre can be contained in $\Theta(n)$ other discs, this increases the running time to $O(n^3)$. A traversal of this graph can be used to enumerate the cells of $\mathcal{A}_R$ (faces of the graph) in $O(|\mathcal{A}_R|)$ time. A geometric 1-centre of $P$ can be found by considering one point $f$ from each cell in $\mathcal{A}_R$ and using breadth-first search to compute the eccentricity of $f$ in $\mathrm{UDG}(P \cup \{f\})$. The minimum such value is the geometric 1-radius of $\mathrm{UDG}(P)$ and the corresponding point $f$ is a geometric 1-centre. In the pseudocode below, BFS-DEPTH$(G, v)$ calls a standard queue-based breadth-first search algorithm to calculate the distance from $v$ to the furthest vertex in $G$.

GEOMETRIC 1-CENTRE$(P)$
  $radius \leftarrow \infty$
  **for** each cell $C \in \mathcal{A}_R$
      $f \leftarrow$ any point in $C$
      $ecc \leftarrow$ BFS-DEPTH$(\mathrm{UDG}(P \cup \{f\}), f)$
      **if** $ecc < radius$
          $radius \leftarrow ecc$
          $centre \leftarrow f$
  **return** $centre$

Adding vertex $f$ increases the number of edges in $\mathrm{UDG}(P)$ by at most $n$. Therefore, each breadth-first search on $\mathrm{UDG}(P \cup \{f\})$ takes $\Theta(n+m)$ time. By Proposition 3, $|\mathcal{A}_R| \in O(n^2)$. Therefore, Algorithm GEOMETRIC 1-CENTRE has worst-case running time $O(n^2(m+n))$. Recall that $\mathrm{UDG}(P \cup \{f\})$ must be connected for a geometric 1-centre to exist. Therefore, $m \geq n-1$ and the running time simplifies to $O(n^2m)$. In the worst case, therefore, this algorithm is quartic in $n$.

Although it suffices to consider only convex cells in $\mathcal{A}_R$, the number of such cells remains $\Omega(n^2)$ in the worst case by Proposition 4. Therefore, the worst-case running time of Algorithm GEOMETRIC 1-CENTRE is not improved by considering only convex cells.

We believe there should be an $o(mn^2)$ time algorithm, but we have been unable to find one. An initial idea is to compute the all-pairs shortest distance matrix $M$ for UDG($P$) and to consider iteratively placing a facility $f$ in each cell $C \in \mathcal{A}_R$. Upon visiting $C$, we examine the eccentricity of each neighbour of $f$ in UDG($P$) using $M$ (these eccentricity values can be precomputed for constant-time table reference). The motivation is that a facility $f$ that minimizes the maximum eccentricity of its neighbours might also minimize the geometric 1-radius of $P$. Unfortunately, this technique fails to account for edges in $E(\mathrm{UDG}(P \cup \{f\})) \setminus E(\mathrm{UDG}(P))$ that provide shorter paths between many pairs of vertices, invalidating the corresponding distances stored in $M$. Thus, eccentricity in UDG($P$) is not necessarily related to eccentricity in UDG($P \cup \{f\}$). We describe a related technique in Section 6.2 that provides an approximate solution.

A more sophisticated idea is to traverse the cells in the arrangement of discs in a sensible order, updating all-pairs shortest distance information each time the search moves to a neighbouring cell using a dynamic shortest path data structure (e.g., [17, 37, 40, 41]). Unfortunately, our efforts in this direction have resulted in a prohibitively expensive increase in running time. In particular, the product of the number of updates required in the worst case and the worst-case cost per update for existing dynamic all-pairs shortest path data structures is high. A strategy combining an efficient traversal of the arrangement with a tailored dynamic all-pairs shortest path algorithm and careful cost analysis may result in improved running time. As noted in Section 9.4, $O(n^3)$ running time is a natural goal for solving the geometric 1-centre problem since the fastest known algorithms for finding a vertex 1-centre require nearly $\Theta(nm)$ time.

### *6.2. Finding an Approximate Geometric 1-Centre*

As we now show, a faster algorithm is possible if we relax constraints on optimality and allow the eccentricity of a solution to exceed the geometric 1-radius by at most one. In brief, Lemma 1 implies that we need only consider a constant number of neighbours of $f$ to measure the eccentricity of $f$ within an additive approximation factor of at most one.

As with the previous algorithm, we begin by constructing $\mathcal{A}_R$ and the corresponding lists of discs (and their centres) in which each cell is contained. A point $f$ is selected within each cell and the corresponding list of disc centres is partitioned according to the regions $R_1(f)$ through $R_6(f)$. These regions correspond to six symmetric sectors whose union forms the unit disc centred at $f$. See Figure 6A. The algorithm computes the approximate eccentricity by iteratively calculating

$$\min_{C \in \mathcal{A}_R} \max_{p \in P} \min_{i \in \{1,\ldots,6\}} \left(1 + \mathrm{dist}_{\mathrm{UDG}(P)}(q_i, p)\right),$$

where $f$ is any point in cell $C$ and $q_i$ is any vertex in $P \cap R_i(f)$. By Lemma 11, to compute the approximate eccentricity of $f$ it suffices to iterate over all $p \in P$ and compare the graph distance between $p$ and a vertex $q_i$ in $P \cap R_i(f)$ for each nonempty region $R_i(f)$. Adding one to the minimum of these (at most) six distances gives either $\mathrm{dist}_{\mathrm{UDG}(P \cup \{f\})}(f, p)$ or $\mathrm{dist}_{\mathrm{UDG}(P \cup \{f\})}(f, p) + 1$, depending on whether a shortest path from $f$ to $p$ passes through the vertex $q_i$ that was selected. The algorithm makes use of unweighted all-pairs shortest-path distances on the vertices of $\mathrm{UDG}(P)$. This distance function can be precomputed in $o(mn)$ time (e.g., see [11]).

```
Approximate Geometric 1-Centre(P)
approxRadius ← ∞
for each cell C ∈ A_R
    f ← any point in C
    approxEcc ← 0
    for each point p ∈ P
        dist ← ∞
        for i ← 1 to 6
            q_i ← any point in R_i(f) ∩ P
            if dist_UDG(P)(q_i, p) + 1 < dist
                dist ← dist_UDG(P)(q_i, p) + 1
        if dist > approxEcc
            approxEcc ← dist
    if approxEcc < approxRadius
        approxRadius ← approxEcc
        approxCentre ← f
return approxCentre
```

**Lemma 11.** *For any set of points $P$ in $\mathbb{R}^2$, any point $f \in \mathbb{R}^2$, and any point $p \in P$,*

$$\left(1 + \min_{i \in \{1,\ldots,6\}} \mathrm{dist}_{\mathrm{UDG}(P)}(q_i, p) - \mathrm{dist}_{\mathrm{UDG}(P \cup \{f\})}(f, p)\right) \in \{0, 1\},$$

*where $q_i$ is any point in $R_i(f) \cap P$.*

**Proof.** For any $i \in \{1, \ldots, 6\}$, any two points $a$ and $b$ in $R_i(f)$ are at most unit distance apart. Consequently, $a$ and $b$ are adjacent in $\mathrm{UDG}(P)$ and

$$\forall p \in P,\ \mathrm{dist}_{\mathrm{UDG}(P)}(a, p) \leq \mathrm{dist}_{\mathrm{UDG}(P)}(b, p) + 1.$$

See Figures 6B and 6C. Any shortest path from $f$ to $p$ must pass through a vertex in $P \cap R_i(f)$, for some $i \in \{1, \ldots, 6\}$. The result follows. □

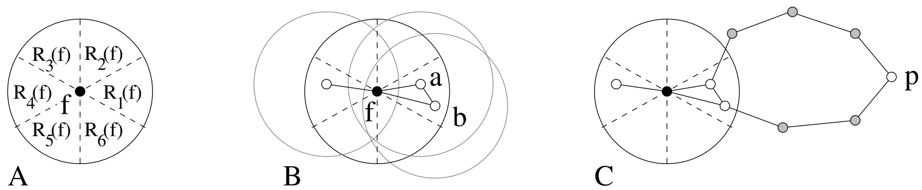


Figure 6: If $a$ and $b$ are in the same sector, $\mathrm{dist}_{\mathrm{UDG}(P)}(a,p)$ and $\mathrm{dist}_{\mathrm{UDG}(P)}(b,p)$ differ by at most one since $\mathrm{dist}_{\mathrm{UDG}(P)}(a,b) = 1$.

For every point $f$, the sets $R_1(f) \cap P$ through $R_6(f) \cap P$ are precomputed in $O(n^3)$ time. Thus, a point can be selected from each set in $O(1)$ time, giving the following theorem:

**Theorem 12.** *Given a set of points $P$ in $\mathbb{R}^2$, Algorithm* Approximate Geometric 1-Centre *identifies a point $f \in \mathbb{R}^2$ in $O(n^3)$ time such that*

$$\mathrm{ecc}_{\mathrm{UDG}(P\cup\{f\})}(P, \{f\}) \leq r + 1,$$

*where $r$ denotes the geometric 1-radius of $P$ and $n = |P|$.*

*6.3. Finding a Geometric $k$-Centre for a Fixed $k$*

When $k$ is fixed, Algorithm Geometric 1-Centre generalizes to give an $O(mn^{2k})$-time algorithm for finding a geometric $k$-centre of a unit disc graph. We begin with the following observation:

**Observation 13.** *Given a set of points $P \subseteq \mathbb{R}^2$ and a set of points $F \subseteq \mathbb{R}^2$ that forms a geometric $k$-centre of $P$, for every client $p \in P$, some shortest path in $\mathrm{UDG}(P \cup F)$ from $p$ to a facility $f \in F$ nearest to $p$ does not contain any facility $f' \in F$, where $f' \neq f$.*

An analogous property also holds for a vertex $k$-centre of any graph. As a consequence of Observation 13, edges connecting two facilities need not be considered when selecting locations for a geometric $k$-centre. Any two or more facilities located in a cell of $\mathcal{A}_R$ serve the same set of clients in $P$, resulting in redundant facilities. Therefore, by Proposition 3, it suffices to consider at most $\binom{n^2-n+2}{k}$ combinations for assigning $k$ facilities to cells in $\mathcal{A}_R$. For each combination of cells, we calculate the corresponding eccentricity. Thus, Algorithm Geometric 1-Centre is modified such that the outer loop considers all combinations of $k$ cells. In this case, BFS-Depth$(G, V)$ begins breadth-first search at the vertices in the set $V$, returning the eccentricity of $V$ in graph $G$. The corresponding running time is at most

$$(n+m)\binom{n^2-n+2}{k} \in O(mn^{2k}).$$

This gives the following theorem:

**Theorem 14.** *For any fixed $k \in \mathbb{Z}^+$, a geometric $k$-centre of a set of $n$ unit discs in $\mathbb{R}^2$ can be found in $O(mn^{2k})$ time.*

We refer to this algorithm simply as GEOMETRIC $k$-CENTRE and provide pseudocode below.

```
GEOMETRIC k-CENTRE(P)
radius ← ∞
for each combination of cells C = {C_1, ..., C_k} ⊆ A_R
    F ← ∅
    for each C_i ∈ C
        f_i ← any point in C_i
        F ← F ∪ f_i
    ecc ← BFS-DEPTH(UDG(P ∪ F), F)
    if ecc < radius
        radius ← ecc
        Centres ← F
return Centres
```

## 7. Hardness Results

In Section 6.3 we described an $O(mn^{2k})$-time algorithm for finding a geometric $k$-centre of a unit disc graph. Of course this running time is exponential if $k$ is an arbitrary input parameter to the problem. In this section we show that GEOMETRIC $k$-CENTRE is NP-hard on unit disc graphs when $k$ is not fixed. This implies NP-hardness for the more general problem, that is, on intersection graphs of sets of regions in two or more dimensions.

**Theorem 15.** *When $k$ is an arbitrary input parameter, the geometric $k$-centre problem on unit disc graphs is NP-hard.*

**Proof.** Given a graph $G$ and an integer $k$, the DOMINATING SET decision problem is to determine whether there exists a set $D \subseteq V(G)$ such that $|D| \leq k$ and every vertex in $V(G)$ is adjacent to some vertex in $D$. DOMINATING SET remains NP-hard if $G$ is a grid graph [16, 32]. We describe a polynomial-time reduction from DOMINATING SET on grid graphs to GEOMETRIC $k$-CENTRE on unit disc graphs.

Choose any finite set of points $P \subseteq \mathbb{Z}^2$ and any integers $k \geq 1$ and $i \geq 0$. Let $s = 2i+1$ and $r = 3i+1$. Let $f : \mathbb{Z}^2 \to \mathbb{Z}^2$ denote the uniform scaling function defined by $f((p_x, p_y)) = (sp_x, sp_y)$. Similarly, let $f^{-1}((p_x, p_y)) = (p_x/s, p_y/s)$. If $A$ is a set, let $f(A) = \{f(p) \mid p \in A\}$. Let

$$\begin{aligned} P' = f(P) \\ &\cup \{(sx+i, sy) \mid 1 \leq i \leq s-1 \text{ and } \{(x,y),(x+1,y)\} \subseteq P\} \\ &\cup \{(sx, sy+i) \mid 1 \leq i \leq s-1 \text{ and } \{(x,y),(x,y+1)\} \subseteq P\}. \end{aligned}$$

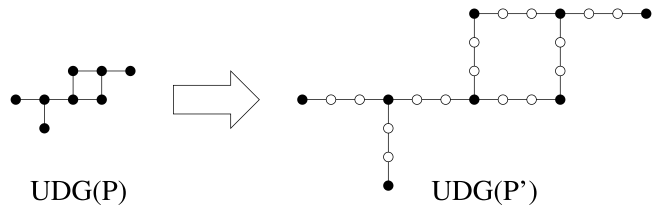


Figure 7: $\mathrm{UDG}(P)$ has a dominating set of cardinality $k$ if and only if $\mathrm{UDG}(P')$ has a geometric $k$-centre of radius $r$. In this example $s = 3$ and $r = 4$.

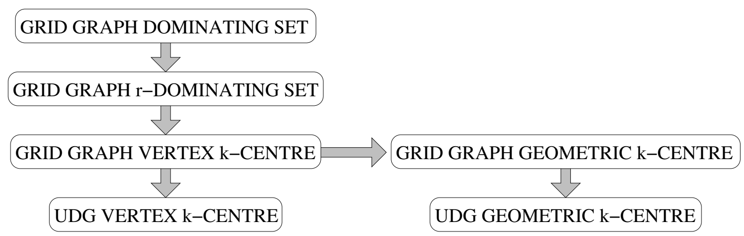


Figure 8: In Theorem 15 we describe a reduction from Dominating Set on grid graphs to Geometric $k$-Centre on unit disc graphs. The hardness of other problems in this hierarchy can be derived by a reduction corresponding to a subset of the steps described in our proof of Theorem 15.

For each $p \in P'$, let $g(p)$ denote the unique point in $f(P)$ that is nearest to $p$ in $\mathrm{UDG}(P')$ by graph distance. Therefore,

$$\forall p \in P',\ \mathrm{dist}_{\mathrm{UDG}(P')}(p, g(p)) \leq \lfloor s/2 \rfloor. \tag{11}$$

Since the points of $P'$ lie on the unit grid, $\mathrm{UDG}(P')$ is a grid graph. Furthermore, $\mathrm{UDG}(P)$ is a minor of $\mathrm{UDG}(P')$; that is, $\mathrm{UDG}(P)$ is equal to $\mathrm{UDG}(P')$ upon scaling the grid by a factor of $s$ and replacing each edge by a path of length $s$. See Figure 7. We claim that $\mathrm{UDG}(P)$ has a dominating set of cardinality at most $k$ if and only if $\mathrm{UDG}(P')$ has a geometric $k$-centre of radius $r$.

Case 1. ($\Rightarrow$) Suppose $\mathrm{UDG}(P)$ has a dominating set, denoted by $D$, of cardinality at most $k$. Observe that $f(D) \subseteq P'$. Furthermore,

$$\forall q \in f(P),\ \exists t \in f(D) \text{ such that } \mathrm{dist}_{\mathrm{UDG}(P')}(q, t) \leq s. \tag{12}$$

By the triangle inequality, (11), and (12),

$$\forall p \in P',\ \exists t \in f(D) \text{ such that } \mathrm{dist}_{\mathrm{UDG}(P')}(p, t) \leq s + \lfloor s/2 \rfloor = r.$$

Therefore, $f(D)$ is a geometric $k$-centre of $P'$ with radius $r$.

Case 2. ($\Leftarrow$) Suppose $F \subseteq \mathbb{R}^2$ is a geometric $k$-centre of $\mathrm{UDG}(P')$ with radius $r$. For any point $t \in \mathbb{R}^2$, there exists some vertex $q \in P'$ such that $\mathrm{Adj}(t) \subseteq \mathrm{Adj}(q)$ in $\mathrm{UDG}(P' \cup \{t\})$. By Observation 13, no two facilities need to be adjacent in $\mathrm{UDG}(P' \cup F)$. Consequently, there exists a set $F' \subseteq P'$ such that $|F'| \leq k$ and

$$\forall p \in P',\ \exists q \in F' \text{ such that } \mathrm{dist}_{\mathrm{UDG}(P')}(p, q) \leq r. \tag{13}$$

By the triangle inequality, (11), and (13),

$$\forall p \in P',\ \exists q \in F' \text{ such that } \operatorname{dist}_{\mathrm{UDG}(P')}(p, g(q)) \leq r + \lfloor s/2 \rfloor < 2s. \tag{14}$$

Observe that

$$\forall \{p_1, p_2\} \subseteq f(P),\ \operatorname{dist}_{\mathrm{UDG}(P')}(p_1, p_2) \bmod s = 0. \tag{15}$$

Therefore, by (14), (15), and since $g(q) \in f(P)$,

$$\forall p \in f(P),\ \exists q \in F' \text{ such that } \operatorname{dist}_{\mathrm{UDG}(P')}(p, g(q)) \leq s.$$

Consequently,

$$\forall p \in P,\ \exists q \in F' \text{ such that } \operatorname{dist}_{\mathrm{UDG}(P)}(p, f^{-1}(g(q))) \leq 1.$$

Let $D = f^{-1}(g(F'))$. Since $|F'| \leq k$, therefore $|D| \leq k$ and $D$ is a dominating set of $\mathrm{UDG}(P)$ whose cardinality is at most $k$. $\square$

## 8. Interval Graphs

Until now we have considered the geometric $k$-centre problem on unit disc graphs, that is, in two-dimensional Euclidean space. We now consider the geometric $k$-centre problem in one dimension and present an algorithm that finds a solution in $O(n)$ time for any arbitrary $k$.

### *8.1. Introduction*

We have examined the geometric $k$-centre problem on unit disc graphs. The one-dimensional analog of a unit disc graph is simply a unit interval graph, i.e., the intersection graph of a set of unit intervals. We can relax the unit-radius restriction and consider the geometric $k$-centre problem on an interval graph, denoted $\mathrm{IG}(I)$, of a finite set of closed intervals $I$ in $\mathbb{R}$.

**Definition 5 (Geometric $k$-Centre of an Interval Graph).** *Given a set of intervals $I = \{I_1, \ldots, I_n\}$ and a positive integer $k$, a* geometric $k$-centre *of $I$ is a set of unit intervals $F = \{F_1, \ldots, F_k\}$ that minimizes*

$$\max_{I_i \in I} \min_{F_j \in F} \operatorname{dist}_{\mathrm{IG}(I \cup F)}(I_i, F_j). \tag{16}$$

Even when $k = 1$, a geometric $k$-centre of an interval graph $\mathrm{IG}(I)$ cannot be determined exclusively by the combinatorial description of $\mathrm{IG}(I)$. If geometry is omitted, then adding a universal vertex (an interval that intersects all intervals in $I$) provides a trivial solution. As with the unit disc graph induced by a given set of discs, we consider the interval graph induced by a given set of intervals $I$. In general, no unit interval is a universal vertex of $\mathrm{IG}(I)$.

*8.2. Related Work: Vertex k-Centre on Interval Graphs*

Olariu [35] gives an $O(n)$-time algorithm for finding a vertex 1-centre of an interval graph, where $n$ denotes the number of intervals. Bespamyatnikh et al. [4] present an $O(nk)$-time algorithm for finding a vertex $k$-centre of a circular-arc graph (and therefore also for any interval graph). Cheng et al. [15] improve the running time with an $O(n)$-time algorithm for finding a vertex $k$-centre of an interval graph. Each of these algorithms requires that the input list of intervals be sorted (e.g., by left endpoints).

*8.3. Finding a Geometric k-Centre of a Set of Intervals*

Given any finite set of intervals $I$ and an arbitrary positive integer $k$, we describe an algorithm for finding a geometric $k$-centre of $I$ in $O(n)$ time if $\mathrm{IG}(I)$ is connected, where $n = |I|$. Our algorithm is straightforward to generalize to $O(n + k \log n)$ time if $\mathrm{IG}(I)$ is disconnected.

*8.3.1. Range of the Geometric k-Radius*

Since a linear-time algorithm exists for finding a vertex $k$-centre, a simple attempt at finding a geometric $k$-centre might be to position a unit interval on every facility of a vertex $k$-centre, leading to the following observation:

**Observation 16.** *Given any set of intervals $I$, the geometric $k$-radius of $I$ is at most $r_v + 1$, where $r_v$ denotes the vertex $k$-radius of $\mathrm{IG}(I)$. This bound is tight.*

Due to their embedding in one dimension, the vertices of an interval graph can be partitioned into $k$ contiguous clusters, each of which has diameter at most $\lceil n/k \rceil - 1$ if $\mathrm{IG}(I)$ is connected. Specifically, Cheng et al. obtain the following tight bound:

**Lemma 17 (Cheng et al. 2007 [15]).** *Given any set of $n$ intervals $I$, if $\mathrm{IG}(I)$ is connected, then the vertex $k$-radius, $r_v$, of $I$ is bounded by*

$$\left\lceil \frac{\lceil d/k \rceil - 1}{2} \right\rceil \leq r_v \leq \left\lceil \frac{\lceil d/k \rceil - 1}{2} \right\rceil + 1, \tag{17}$$

*where $d$ denotes the diameter of $\mathrm{IG}(I)$.*

The algorithm of Cheng et al. [15] finds a vertex $k$-centre in $O(n)$ time by checking whether there exists a solution of radius $r_v$ for each of the two possible integer values for $r_v$ bounded by (17). By Observation 16 and Lemma 17, the geometric $k$-radius of a connected interval graph is at most

$$\left\lceil \frac{\lceil d/k \rceil - 1}{2} \right\rceil + 2.$$

Unlike the vertex $k$-radius, however, the geometric $k$-radius can be any integer in the range

$$\left[0, \left\lceil \frac{\lceil d/k \rceil - 1}{2} \right\rceil + 2\right]. \tag{18}$$

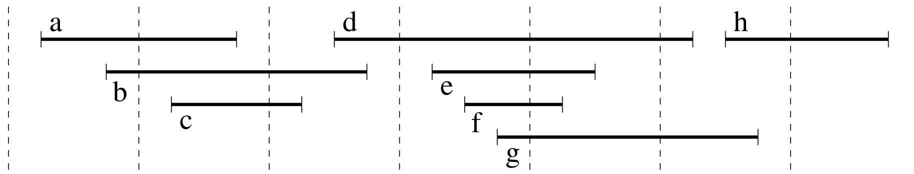


Figure 9: Consider the set of intervals $I = \{a, \ldots, h\}$. Adjacent dotted lines denote a distance of one unit. Examples: $\text{nextInt}(a, 1) = \text{nextInt}(a) = b$, $\text{nextInt}(a, 2) = \text{nextInt}(\text{nextInt}(a)) = \text{nextInt}(b) = d$, $\text{nextInt}(a, 3) = g$, $\text{nextInt}(a, 4) = h$, $\text{nextInt}(a, 5) = h$, $\text{nextInt}^{+1}(a) = d$, $\text{nextInt}^{+1}(f) = g$, $\text{nextInt}^{+1}(h) = h$, $\text{nextDisj}(a) = f$, $\text{nextDisj}(b) = f$, and $\text{nextDisj}(d) = h$.

This is because each facility (consisting of a unit interval) can cover a sequence of intervals in $\text{IG}(I)$, forming a bridge that can potentially reduce the diameter of each cluster $\text{IG}(I)$ significantly. Our solution is to search the range of possible values for the geometric $k$-radius.

*8.3.2. Preliminary Computation*

Given a finite set of intervals $I = \{I_1, \ldots, I_n\}$ such that $\text{IG}(I)$ is connected, we define the following functions for any interval $I_i = [a_i, b_i] \in I$. See Figure 9.

- Let $\text{nextInt}(I_i)$ denote the interval in $I$ whose right endpoint is rightmost among all intervals that intersect $I_i$.
- As defined by Cheng et al. [15], let $\text{nextInt}(I_i, j)$ denote the iterated application of $\text{nextInt}(\cdot)$, such that for any interval $I_i \in I$ and any positive integer $j$,
$$\text{nextInt}(I_i, j) = \begin{cases} \text{nextInt}(\text{nextInt}(I_i, j-1)) & \text{if } j \geq 2, \\ \text{nextInt}(I_i) & \text{if } j < 2. \end{cases}$$
- Let $\text{nextInt}^{+1}(I_i)$ denote the interval in $I$ whose right endpoint is rightmost among all intervals that intersect the interval $[b_i, b_i + 1]$. That is, $\text{nextInt}^{+1}(I_i) = \text{nextInt}(I'_i)$, where $I'_i = [b_i, b_i + 1]$.
- Let $\text{nextDisj}(I_i)$ denote the interval in $I$ whose right endpoint is leftmost among all intervals entirely contained in $(b_i, \infty)$.
- Let $\text{dist}_{\text{right}}(I_i) = \text{dist}_{\text{IG}(I)}(I_i, I_n)$, where $I_n$ denotes the interval in $I$ whose left endpoint is rightmost.

**Lemma 18 (Chen et al. 1998 [14]).** *A set $I$ of $n$ intervals can be preprocessed in $O(n)$ time to support $O(1)$-time query for* $\text{nextInt}(I_i)$ *and* $\text{nextInt}(I_i, j)$ *for any $I_i \in I$ and any positive integer $j$.*

Lemma 18 requires that set $I$ be presorted. In particular, we require one list of intervals sorted by left endpoints and a second list sorted by right endpoints. Precomputing function $\text{nextInt}^{+1}(\cdot)$ can be achieved in $O(n)$ time by scanning the list of intervals in non-increasing order by their right endpoints as follows:

COMPUTE-nextInt$^{+1}(I)$ // precondition: $\forall i < j,\ b_i \leq b_j$

$i \leftarrow n$<br>
**for** $j \leftarrow n$ **to** 1<br>
    **while** $a_i > b_j + 1$<br>
        $i \leftarrow i - 1$<br>
    $Rint[j] \leftarrow I_i$<br>
**return** $Rint[1:n]$

Similarly, function nextDisj$(\cdot)$ can be precomputed in $O(n)$ time by scanning two ordered lists of $I$ in parallel, one sorted by left endpoints and one sorted by right endpoints. Function $\text{dist}_{\text{right}}(\cdot)$ can be computed recursively in $O(n)$ time. The values computed for each function can be stored in a table of size $O(n)$ for constant-time reference.

*8.3.3. Algorithm*

Let $I = \{I_1, \ldots, I_n\}$ denote an input set of intervals such that IG$(I)$ is connected, where for each $i$, $I_i = [a_i, b_i]$. Suppose the intervals are sorted by right endpoints, i.e., for all $i < j$, $b_i \leq b_j$. Furthermore, if $b_i = b_{i+1}$ for two intervals $I_i$ and $I_{i+1}$, suppose $a_i \leq a_{i+1}$. Compute functions nextInt$(\cdot)$, nextInt$(\cdot,\cdot)$, nextInt$^{+1}(\cdot)$, nextDisj$(\cdot)$, and $\text{dist}_{\text{right}}(\cdot)$ as described in Section 8.3.2.

The algorithm for finding a geometric $k$-centre of $I$ consists of a binary search on the integers in the range given in Expression (18), where for each integer $r$ in the search sequence we check whether there exists a geometric $k$-centre of radius at most $r$. For a given $r$, this check is achieved by examining a set of $k$ unit intervals, $\{F_1, \ldots, F_k\}$, defined by a sequence of calls to nextInt$(\cdot,\cdot)$, nextInt$^{+1}(\cdot)$, and nextDisj$(\cdot)$, to determine whether $r$ is sufficiently large.

For a given $r$, start at the first interval $I_1$, and let the first facility be positioned at $F_1 = [c_1, c_1 + 1]$, where the last interval in $I$ before $F_1$ is $I_{f_1} = \text{nextInt}(I_1, r-1) = [a_{f_1}, b_{f_1}]$ and $c_1 = b_{f_1}$. Each interval in $\{I_1, \ldots, I_{f_1}\}$ is within distance $r$ of $F_1$ in IG$(I \cup \{F_1\})$ and lies to the left of $F_1$. Next we identify intervals within distance $r$ to the right of $F_1$. Let $I'_{f_1} = \text{nextInt}^{+1}(I_{f_1})$ denote the interval whose right endpoint is rightmost among all intervals that intersects $F_1$. Let $I''_{f_1} = \text{nextInt}(I'_{f_1}, r-2)$. Observe that $\text{dist}_{\text{IG}(I \cup \{F_1\})}(I''_{f_1}, F_1) = r - 1$. The first interval not within distance $r$ of $F_1$ is nextDisj$(I''_{f_1})$. Therefore, for every interval $I_i = [a_i, b_i]$, if $a_i < a''_i$, where $I''_{f_1} = [a''_i, b''_i]$, then $\text{dist}_{\text{IG}(I \cup \{F_1\})}(I_i, F_1) \leq r$. See Figure 10.

The procedure repeats starting with interval nextDisj$(I''_{f_1})$. The set of intervals $F = \{F_1, \ldots, F_k\}$ is determined after $k$ iterations. If $\text{dist}_{\text{right}}(I'_{f_k}) \leq r - 1$, then $\text{dist}_{\text{IG}(I \cup F)}(F_k, I_n) \leq r$, and the set $F$ is a geometric $k$-centre of $I$ of radius at most $r$. Otherwise, the geometric $k$-radius of $I$ must be strictly greater than $r$.

This procedure is repeated for each value of $r$ in the binary search sequence. Upon termination, the algorithm identifies the value $r$ such that $I$ has a geometric $k$-centre of radius $r$, but not $r - 1$.

Precomputation requires $O(n)$ time. For each value of $r$ examined, checking whether $I$ has a geometric $k$-centre of radius $r$ requires $O(k)$ time. The binary

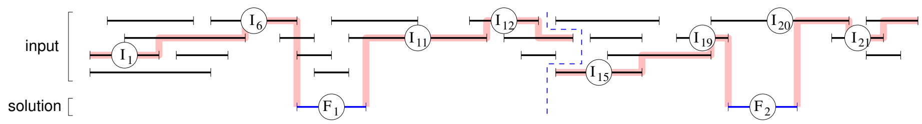


Figure 10: **Geometric $k$-centre algorithm in one dimension.** Suppose $k = 2$ and $r = 3$. $I_{f_1} = \text{nextInt}(I_1, r-1) = I_6$. The right endpoint of $I_6$ determines the left endpoint of the unit interval $F_1$. $I'_{f_1} = \text{nextInt}^{+1}(I_6) = I_{11}$. $I''_{f_1} = \text{nextInt}(I_{11}, r-2) = I_{12}$. The first interval not within distance $r$ of $F_1$ is $\text{nextDisj}(I_{12}) = I_{15}$. Thus, the intervals $I_1, \ldots, I_{14}$ are within distance $r$ of $F_1$. The procedure repeats starting with $I_{15}$. $I_{f_2} = \text{nextInt}(I_{15}, r-1) = I_{19}$. The right endpoint of $I_{19}$ determines the left endpoint of the unit interval $F_2$. $I'_{f_2} = \text{nextInt}^{+1}(I_{19}) = I_{20}$. Since $\text{dist}_{\text{right}}(I'_{f_2}) \leq r - 1$, the intervals $I_{15}, \ldots, I_n$ are within distance $r$ of $F_2$. Therefore, the set $\{F_1, F_2\}$ is a geometric 2-centre of $I$ of radius $r = 3$. At this point the procedure begins again as the binary search continues examining values $r \in [0, 3]$.

search sequence examines $O(\log(n/k))$ values for $r$. Therefore, if $\text{IG}(I)$ is connected, then the total running time is $O(n + k\log(n/k))$. If $k \geq n$, then $F = I$ is a geometric $k$-centre of $I$ (i.e., no computation is required). If $k < n$, then $O(n + k\log(n/k)) \in O(n)$. Since each client must be examined, a lower bound of $\Omega(n)$ applies. The algorithm is straightforward to generalize to $O(n+k\log n)$ time if $\text{IG}(I)$ is disconnected. This gives the following theorem:

**Theorem 19.** *For any arbitrary $k \in \mathbb{Z}^+$, a geometric $k$-centre of a set of $n$ sorted intervals in $\mathbb{R}$ can be found in $\Theta(n)$ time if $\text{IG}(I)$ is connected.*

## 9. Directions for Future Research

### *9.1. Intersection Graphs*

Motivated by gateway placement in wireless networks, we have examined the problem of finding a geometric $k$-centre in unit disc graphs. Of course, unit disc graphs are not the only model for representing wireless networks. In addition to the results described in this paper, we have partial results for generalizations to the setting of disc graphs (intersection graphs of discs of differing radii), to three dimensions (intersection graphs of balls), and to rectangle intersection graphs.

### *9.2. Visibility Graphs*

Another possible direction is to model obstacles and interference in wireless networks by applying the geometric $k$-centre problem to the setting of visibility graphs. Given a set of points $P$ (clients) in a polygonal region $R$, the objective is to select a set $F$ of $k$ points (facilities) in $R$ such that the maximum graph distance between any client and its nearest facility is minimized in the visibility graph of $P \cup F$ in $R$; a pair of nodes is connected in the visibility graph if and only if the line segment between them is unobstructed by polygon $R$. By applying observations similar to those made in Section 4, a solution can be found discretely and, furthermore, the corresponding partition of the plane into visibility regions is a partial order relation for which it suffices to consider the sinks. Thus, visibility graphs seems like a natural setting to which to apply some of the ideas developed in this paper. See [5] and [6] for results on properties of visibility regions and the corresponding partial order.

### *9.3. Geometric k-Median*

One might consider generalizations of the optimization function that is minimized in selecting positions for gateways. In particular, two fundamental problems of facility location are the $k$-centre and $k$-median problems. In this paper we restrict attention to the first of these. The two problems are defined analogously, with the exception that the maximum over all $p_i \in P$ in (1) is replaced by a summation over all $p_i \in P$. Whereas a geometric $k$-centre minimizes the maximum node-to-gateway distance, a geometric $k$-median minimizes the average node-to-gateway distance. The algorithms for finding a geometric 1-centre and a geometric $k$-centre for a fixed $k$ presented in this paper are straightforward to adapt to the problems of identifying a geometric 1-median or a geometric $k$-median, respectively. In this case, each call to BFS-Depth is replaced by a call to BFS-Sum, which returns the corresponding sum of the distances from every node to the nearest gateway. The resulting running times remain $O(mn^2)$ and $O(mn^{2k})$, respectively.

### *9.4. Improved Running Time*

Finally, can a geometric 1-centre of a unit disc graph be found in $O(n^3)$ worst-case time? To the authors' knowledge, the $O(n^2m)$-time algorithm presented in Section 6.1 is the first solution to this problem; can the $O(n^2m)$-time be improved? $O(nm)$ or $O(n^3)$ running times are natural goals for solving this problem since the fastest known algorithms for finding a vertex 1-centre require nearly $\Theta(nm)$ time. See Section 6.1 for a brief overview of promising strategies that failed to achieve $o(n^2m)$ time, as well as one possible direction for a future algorithm which the authors believe has the potential to succeed.


## Acknowledgements

The authors would like to thank the following individuals whose ideas and suggestions have helped shape various aspects of this work: Ian Munro and Matthew Skala with whom we discussed convexity in continuous metric spaces, Csaba Tóth who suggested the example in Figure 2C, Rolf Klein with whom we discussed this example, Christian Knauer with whom we discussed Theorem 5, David Bremner, Sándor Fekete, Wendy Myrvold, and Sue Whitesides with whom we discussed Theorem 8 during the 2009 McGill-INRIA Bellairs Workshop, and finally the anonymous reviewers who assisted us in improving this paper with numerous helpful suggestions.

## References


[1] P. K. Agarwal and M. Sharir. Efficient algorithms for geometric optimization. *ACM Computing Surveys*, 30:412–458, 1998.

[2] P. K. Agarwal, M. Sharir, and S. Toledo. An efficient multi-dimensional searching technique and its applications. Technical Report CS-1993-20, Department of Computer Science, Duke University, 1993.

[3] B. Aoun, R. Boutaba, and Y. Iraqi. Gateway placement optimization in WMN with QoS constraints. *IEEE Journal on Selected Areas in Communications*, 11:2127–2136, 2006.

[4] S. Bespamyatnikh, B. Bhattacharya, M. Keil, D. Kirkpatrick, and M. Segal. Efficient algorithms for centers and medians in interval and circular-arc graphs. *Networks*, 39:144–152, 2002.

[5] P. Bose. Visibility in simple polygons. Master's thesis, University of Waterloo, 1991.

[6] P. Bose, A. Lubiw, and J. I. Munro. Efficient visibility queries in simple polygons. *Computational Geometry: Theory and Applications*, 23(3):313–335, 2002.

[7] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, 7:609–616, 2001.

[8] H. Breu. *Algorithmic Aspects of Constrained Unit Disk Graphs*. PhD thesis, University of British Columbia, 1996.

[9] H. Breu and D. Kirkpatrick. Unit disk graph recognition is NP-hard. *Computational Geometry: Theory and Applications*, 9(1–2):3–24, 1998.

[10] T. M. Chan. More planar two-center algorithms. *Computational Geometry: Theory and Applications*, 13(3):189–198, 1999.

[11] T. M. Chan. All-pairs shortest paths for unweighted undirected graphs in $o(mn)$ time. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 514–523, 2006.

[12] B. M. Chazelle and D. T. Lee. On a circle placement problem. *Computing*, 36(1–2):1–16, 1986.

[13] B. M. Chazelle and J. Matoušek. On linear-time deterministic algorithms for optimization problems in fixed dimensions. *Journal of Algorithms*, 21:579–597, 1996.

[14] D. Z. Chen, D.T. Lee, R. Sridhar, and C. N. Sekharan. Solving the all-pair shortest path query on interval and circular-arc graphs. *Networks*, 31(4):249–258, 1998.

[15] T.C.E. Cheng, L. Kang, and C.T. Ng. An improved algorithm for the p-center problem on interval graphs with unit lengths. *Computers and Operations Research*, 34(8):2215–2222, 2007.

[16] B. Clark, C. Colbourn, and D. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1–3):165–177, 1991.

[17] C. Demetrescu and G. F. Italiano. A new approach to dynamic all pairs shortest paths. *Journal of the ACM*, 51(6):968–992, 2004.

[18] S. Durocher. *Geometric Facility Location under Continuous Motion.* PhD thesis, University of British Columbia, 2006.

[19] S. Durocher, K. R. Jampani, A. Lubiw, and L. Narayanan. Modelling gateway placement in wireless networks: Geometric $k$-centres of unit disc graphs. In *Proceedings of the ACM SIGACT-SIGOPS International Workshop on Foundations of Mobile Computing (DIAL M-POMC)*, volume 5, pages 79–86. ACM, 2008.

[20] A. Efrat, S. P. Fekete, P. R. Gaddehosur, J. S. B. Mitchell, V. Polishchuk, and J. Suomela. Improved approximation algorithms for relay placement. In *Proceedings of the European Symposium on Algorithms (ESA)*, Lecture Notes in Computer Science, pages 356–367. Springer-Verlag, 2008.

[21] T. Feder and D. Greene. Optimal algorithms for approximate clustering. In *Proceedings of the Symposium on Theory of Computing (STOC)*, pages 434–444. ACM, 1988.

[22] T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.

[23] W. K. Hale. Frequency assignment: Theory and applications. *Proceedings of the IEEE*, 68(12):1497–1514, 1980.

[24] D. S. Hochbaum and D. B. Shmoys. A unified approach to approximation algorithms for bottleneck problems. *Journal of the ACM*, 33(3):533–550, 1986.

[25] W. L. Hsu and G. L. Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Mathematics*, 1:209–216, 1979.

[26] M. L. Huson and A. Sen. Broadcast scheduling algorithms for radio networks. In *Proceedings of the IEEE Military Communications Conference (MILCOM)*, volume 2, pages 647–651, 1995.

[27] R. Z. Hwang, R. Chia-Tung Lee, and R. C. Chang. The slab dividing approach to solve the Euclidean $p$-center problem. *Algorithmica*, 9:1–22, 1993.

[28] O. Kariv and S. L. Hakimi. An algorithmic approach to network location problems I: The $p$-centers. *SIAM Journal of Applied Mathematics*, 37(3):513–538, 1979.

[29] X.-Y. Li. Applications of computational geometry in wireless ad hoc networks. In X. Cheng, X. Huang, and D.-Z. Du, editors, *Ad Hoc Wireless Networking*, pages 197–264. Kluwer, 2003.

[30] X.-Y. Li and Y. Wang. Wireless sensor networks and computational geometry. In M. Ilyas, I. Mahgoub, and L. Kelly, editors, *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, pages 39.1–39.42. CRC Press, 2004.

[31] M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz. Simple heuristics for unit disk graphs. *Networks*, 25(2):59–68, 1995.

[32] S. Masuyama, T. Ibaraki, and T. Hasegawa. The computational complexity of the $m$-center problems in the plane. *Transactions of IEICE*, E64:57–64, 1981.

[33] N. Megiddo and K. J. Supowit. On the complexity of some common geometric location problems. *SIAM Journal on Computing*, 13(1):182–196, 1984.

[34] S. Mihandoust and L. Narayanan. Multiple sink positioning in sensor networks. In *Proceedings of the ACM SIGACT-SIGOPS International Workshop on Foundations of Mobile Computing (DIAL M-POMC)*, volume 4. ACM, 2007.

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

[36] J. Plesník. On the computational complexity of centers locating in a graph. *Applications of Mathematics*, 25:445–452, 1980.

[37] L. Roditty and U. Zwick. On dynamic shortest paths problems. In *Proceedings of the European Symposium on Algorithms (ESA)*, volume 3221 of *Lecture Notes in Computer Science*, pages 580–591. Springer-Verlag, 2004.

[38] J. Steiner. *Gesammelte Werke*, volume 1. Reimer, Berlin, 1881.

[39] A. Tamir. Improved complexity bounds for center location problems on networks by using dynamic data structures. *SIAM Journal on Discrete Mathematics*, 1(3):377–396, 1988.

[40] M. Thorup. Fully-dynamic all-pairs shortest paths: Faster and allowing negative cycles. In *Proceedings of the Scandanavian Workshop on Algorithm Theory (SWAT)*, volume 3111 of *Lecture Notes in Computer Science*, pages 384–396. Springer-Verlag, 2004.

[41] M. Thorup. Worst-case update times for fully-dynamic all-pairs shortest paths. In *Proceedings of the Symposium on Theory of Computing (STOC)*, pages 112–119, 2005.

[42] C. Tóth, 2007. Personal communication.

[43] E. Welzl. Smallest enclosing disks (balls and ellipsoids). In H. Maurer, editor, *New Results and New Trends in Computer Science*, volume 555 of *Lecture Notes in Computer Science*, pages 359–370. Springer-Verlag, 1991.