Computational Geometry 46 (2013) 394–399

Contents lists available at SciVerse ScienceDirect

Computational Geometry: Theory and Applications

www.elsevier.com/locate/comgeo

A note on multicovering with disks

Reuven Bar-Yehuda [a], Dror Rawitz [b,*]

[a] Department of Computer Science, Technion, Haifa 32000, Israel
[b] School of Electrical Engineering, Tel-Aviv University, Tel-Aviv 69978, Israel

ARTICLE INFO

Article history:
Received 30 December 2011
Accepted 18 October 2012
Available online 23 October 2012
Communicated by M. de Berg

Keywords:
Approximation algorithms
Disk cover
Multicovering

ABSTRACT

In the Disk Multicover problem the input consists of a set P of n *points* in the plane, where each point $p \in P$ has a covering requirement $k(p)$, and a set B of m *base stations*, where each base station $b \in B$ has a weight $w(b)$. If a base station $b \in B$ is assigned a radius $r(b)$, it *covers* all points in the disk of radius $r(b)$ centered at b. The weight of a radii assignment $r : B \to \mathbb{R}$ is defined as $\sum_{b \in B} w(b)r(b)^\alpha$, for some constant α. A feasible solution is an assignment such that each point p is covered by at least $k(p)$ disks, and the goal is to find a minimum weight feasible solution. The Polygon Disk Multicover problem is a closely related problem, in which the set P is a polygon (possibly with holes), and the goal is to find a minimum weight radius assignment that covers each point in P at least K times. We present a $3^\alpha k_{\max}$-approximation algorithm for Disk Multicover, where $k_{\max}$ is the maximum covering requirement of a point, and a $(3^\alpha K + \varepsilon)$-approximation algorithm for Polygon Disk Multicover.

1. Introduction

In the Disk Multicover problem we are given a set P of n *points* in the plane with a covering requirement function on the points $k : P \to \mathbb{N}$, and a set B of m *base stations* with a weight function on the base stations $w : B \to \mathbb{R}$. A solution is an assignment of radii to base stations, namely a function $r : B \to \mathbb{R}^+$. Given a solution r, a point p is said to be *covered* by a base station b, if the distance between p and b is at most $r(b)$. In other words, p is covered by b if $b \in \textsc{disk}[b, r(b)]$, where $\textsc{disk}[b, \rho]$ is the disk containing the points that are within distance ρ from base station b, namely $\textsc{disk}[b, \rho] = \{p\colon \|b, p\|_2 \leqslant \rho\}$. A point p is said to be *ℓ-covered* by r, if p is covered by at least ℓ base stations. A feasible solution is an assignment r that $k(p)$-covers each point $p \in P$. We assume throughout the paper that $|B| \geqslant k_{\max}$, where $k_{\max} \overset{\text{def}}{=} \max_p k(p)$, since otherwise no feasible solution exists. The weight of a disk centered at b with radius $r(b)$ is $w(b)r(b)^\alpha$, where $\alpha \geqslant 1$ is a constant, and the weight of an assignment r is $w(r) = \sum_{b \in B} w(b)r(b)^\alpha$. The goal in the Disk Multicover problem is to find a feasible assignment of minimum weight.

The special case in which $k(p) = K$, for every $p \in P$, is called Disk Multicover with *uniform* requirements. The case where $K = 1$ is called Disk Cover. An instance in which $w(b) = 1$, for every $b \in B$, is called unweighted.

In the Polygon Disk Multicover problem we are given a polygon P, possibly with holes, with n vertices, and an integer K (instead of a discrete set of points and a requirement function), and the goal is to find a minimum weight assignment that K-covers all points in the polygon P.

Disk covering problems naturally arise in the design phase of cellular and wireless networks, where m potential locations for base stations are identified. A base station at a potential location b may cover a disk centered at b at any radius, where the energy consumption of this disk is proportional to its area or to $r(b)^\alpha$, for some $\alpha \geqslant 1$. The weight $w(b)$ of the location

* Corresponding author.
E-mail addresses: reuven@cs.technion.ac.il (R. Bar-Yehuda), rawitz@eng.tau.ac.il (D. Rawitz).

0925-7721/$ – see front matter
http://dx.doi.org/10.1016/j.comgeo.2012.10.006

b is power parameter associated with b. Multicovering is used in cases where the network needs to be fault-tolerant. Each client (point) is associated with a covering requirement according to its importance or service level.

1.1. Related work

Lev-Tov and Peleg [9] considered (unweighted) Disk Cover with $\alpha = 1$, namely where the cost of the solution is the sum of radii. They obtained a polynomial-time algorithm for the case where all points and base stations are located on a line, and a PTAS for the general case. Bilò et al. [5] showed that Disk Cover is NP-hard for $\alpha \geqslant 2$. This result was extended to $\alpha > 1$ by Alt et al. [2]. This implies that Disk Multicover is also NP-hard.[1] A 9-approximation algorithm for Disk Cover with $\alpha = 2$ appears in [8]. The approximation ratio of this algorithm is 3^α in general.

Chekuri et al. [6] considered the set multi-cover problem in geometric settings. In this problem the input is a set of points P and a collection of geometric shapes (or sets) $\mathcal{F}$, and the goal is to find a minimum cardinality subset of $\mathcal{F}$ such that each point $p \in P$ is covered by at least $k(p)$ sets. They presented an $O(\log \text{OPT})$-approximation algorithm for set systems of bounded VC-dimension and an $O(1)$-approximation algorithm for covering points by half-spaces in three dimensions.

Abu-Affash et al. [1] studied unweighted Disk Multicover with uniform requirements and $\alpha = 2$. They presented an approximation algorithm whose approximation guarantee is $23.02 + 63.91(K - 1)$. They also presented an approximation algorithm for Polygon Disk Multicover that is based on a reduction to Disk Multicover. This reduction increases the approximation ratio by a factor of $\frac{25}{9}$. Hence, using their algorithm for Disk Multicover they obtained an approximation ratio of $63.94 + 177.64(K - 1)$. In the unit requirements case they used the algorithm from [8] and obtained a 25-approximation algorithm.

1.2. Our results

We present a $3^\alpha k_{\max}$-approximation algorithm for Disk Multicover. Our algorithm extends the algorithm for Disk Cover from [8], and its analysis is based on the local ratio technique [4,3]. Note that this already leads to a $25K$-approximation algorithm for unweighted Polygon Disk Multicover with $\alpha = 2$ by using our algorithm as a subroutine of the algorithm from [1]. However, we obtain a better approximation ratio by using a different reduction to multicovering of points with a given set of disks that increases the approximation ratio by a factor of $(1 + \varepsilon)$. More specifically, we show that one may compute a set of points and a set of disks whose optimal cover is at most $(1 + \varepsilon)$ times larger than the optimum of the given Polygon Disk Multicover instance. Using this reduction we obtain a $(3^\alpha K + \varepsilon)$-approximation algorithm for Polygon Disk Multicover with $\alpha \geqslant 1$, for every $\varepsilon > 0$. This algorithm assumes integer (or rational) coordinates and weights, and its running time is polynomial in the number of bits in the input. The same approach can be used for covering the boundary of a given polygon.

2. Preliminaries

Definitions. Observe that given a Disk Cover instance, the set of radii can be discretized, since there are at most n interesting radii per base station, one per point in P. Therefore, at most nm disks should be considered. We denote the set of these disks by $\mathcal{D}$. Given a disk $D \in \mathcal{D}$, we denote the center of D by $b(D)$ and its radius by $r(D)$. The weight of a disk D is $w(D) = w(b(D)) \cdot r(D)^\alpha$. Given a set of disks $\mathcal{S} \subseteq \mathcal{D}$, we define $w(\mathcal{S}) \stackrel{\text{def}}{=} \sum_{D \in \mathcal{S}} w(D)$. Given a set of $\mathcal{S}$ of disks and a point p, we define $\mathcal{S}(p) = \{D \in \mathcal{S}\colon\ p \in D\}$. A set of disks $\mathcal{D}' \subseteq \mathcal{D}$ is called *ℓ-independent* if (i) $|\mathcal{D}'(p)| \leqslant \ell$, for every point $p \in P$, and (ii) $\mathcal{D}'$ does not contain disks centered at the same location.

Local-ratio technique. The local-ratio technique is based on the Local-Ratio Lemma, which applies to minimization problems of the following type. (See [3] for the maximization version.) The input is a non-negative weight vector $w \in \mathbb{R}^n$ and a set $\mathcal{F}$ of feasibility constraints. The problem is to find a solution vector $x \in \{0, 1\}^n$ that minimizes the inner product $w \cdot x$ subject to the constraints $\mathcal{F}$. (The proof of the lemma is given for completeness.)

Lemma 1 *(Local-Ratio). (See [4].) Let $\mathcal{F}$ be a set of constraints and let w, w_1, and w_2 be weight vectors such that $w = w_1 + w_2$. Let γ be a number. Then, if x is γ-approximate both with respect to $(\mathcal{F}, w_1)$ and with respect to $(\mathcal{F}, w_2)$, then x is also γ-approximate with respect to $(\mathcal{F}, w)$.*

Proof. Let x^* denote an optimal solution with respect to w, and let x_i denote an optimal solution with respect to w_i, $i \in \{1, 2\}$. Then

$$wx = w_1 \cdot x + w_2 \cdot x \leqslant \gamma(w_1 \cdot x_1) + \gamma(w_2 \cdot x_2) \leqslant \gamma\big(w_1 \cdot x^* + w_2 \cdot x^*\big) = \gamma\big(w \cdot x^*\big),$$

and we are done. □

[1] It is not hard to show a reduction from Disk Cover to Disk Multicover with uniform requirements.

We use of the Local-Ratio Lemma in the following manner. Given a Disk Cover instance with non-negative input weight function w, we find a non-negative weight function $w_1 \leqslant w$ such that every $k_{\max}$-independent set is γ-approximate with respect to w_1. Then, we solve the problem with respect to $w - w_1$. Since our algorithm computes a $k_{\max}$-independent γ-approximate set with respect to $w - w_1$, this set is also γ-approximate with respect to w.

3. Multicovering points with disks

In this section we present an $3^{\alpha}k_{\max}$-approximation algorithm for the Disk Multicover problem.

Algorithm **DiskMC** works as follows. The first step is to compute a feasible solution $\mathcal{D}_0 \subseteq \mathcal{D}$. The set $\mathcal{D}_0$ may contain several disks centered at the same base station, but it is feasible in the sense that any point p in P is covered by at least $k(p)$ disks that are centered at different locations. Then, a $k_{\max}$-independent set $\mathcal{I} \subseteq \mathcal{D}_0$ is computed. Note that $\mathcal{I}$ may be infeasible. Finally, a cover $\mathcal{C}$ is computed by increasing the radii of the disks in $\mathcal{I}$ by a factor of 3.

Algorithm 1 DiskMC $(P, \mathcal{D}, w)$.

1: $\mathcal{D}_0 \leftarrow \{D \in \mathcal{D}\colon\ w(D) = 0\}$
2: **while** $P \neq \emptyset$ **do**
3: $\quad \varepsilon \leftarrow \min_{D \in \mathcal{D} \setminus \mathcal{D}_0} \frac{w(D)}{|D \cap P|}$
4: $\quad$ Define $w_1(D) = \begin{cases} \varepsilon \cdot |D \cap P| & w(D) > 0, \\ 0 & w(D) = 0 \end{cases}$
5: $\quad w \leftarrow w - w_1$
6: $\quad \mathcal{D}_0 \leftarrow \{D \in \mathcal{D}\colon\ w(D) = 0\}$
7: $\quad P \leftarrow P \setminus \{p \in P\colon\ p \text{ is } k(p)\text{-covered by } \mathcal{D}_0\}$
8: **end while**
9: $\mathcal{I} \leftarrow \emptyset$
10: **while** $\mathcal{D}_0 \neq \emptyset$ **do**
11: $\quad D_{\max} \leftarrow \operatorname{argmax}_{D \in \mathcal{D}_0} r(D)$
12: $\quad$ **if** $\mathcal{I} \cup \{D_{\max}\}$ is $k_{\max}$-independent **then** $\mathcal{I} \leftarrow \mathcal{I} \cup \{D_{\max}\}$
13: $\quad \mathcal{D}_0 \leftarrow \mathcal{D}_0 \setminus \{D_{\max}\}$
14: **end while**
15: **return** $\mathcal{C} = \{\textsc{disk}[b(D), 3r(D)]\colon\ D \in \mathcal{I}\}$

The first while loop of Algorithm **DiskMC** has at most nm iterations, since at least one disk joins $\mathcal{D}_0$ during each iteration. The running time of each iteration of this loop is $O(nm)$. The second loop can easily be implemented in $O(nm)$ time. Hence, the overall running time is polynomial.

Let $\mathcal{D}_0^i$ be the set $\mathcal{D}_0$ at the end of the ith iteration of the first while loop (lines 2–8), and let t be the number of iterations of this loop. Similarly, let P^i and w^i denote the set P and the weight vector w at the end of the ith iteration.

Lemma 2. *Algorithm **DiskMC** computes feasible solutions.*

Proof. First, observe that $\mathcal{D}_0^t$ $k(p)$-covers, every point $p \in P$, due to the termination condition of the first while loop. To show that $\mathcal{C}$ is a feasible solution, consider a point $p \in P$. If $|\mathcal{I}(p)| \geqslant k(p)$, then we are done. Otherwise, there exists at least one disk $D \in \mathcal{D}_0^t \setminus \mathcal{I}$ such that $p \in D$. Moreover, there must exist a point $p' \neq p$ due to which D was not added to $\mathcal{I}$, namely a point $p' \in D$ such that $\mathcal{I}(p') \geqslant k_{\max}$. More specifically, there are $k_{\max}$ disks $D_1, \ldots, D_{k_{\max}}$ such that $p' \in D_i$ and $r(D_i) \geqslant r(D)$, for every i. Since $p, p' \in D$, it follows that the distance between p and p' is at most $2r(D)$. Hence, p is $k_{\max}$-covered by $\mathcal{C}$, since $p \in \textsc{disk}[b(D_i), 3r(D_i)]$, for every i. $\square$

We use the local ratio technique [4,3] to prove that $\mathcal{I}$ is $k_{\max}$-approximate. (This can be shown for any $k_{\max}$-independent set $\mathcal{I}$.)

Lemma 3. *The set $\mathcal{I}$ that is computed by Algorithm **DiskMC** is $k_{\max}$-approximate.*

Proof. We prove that $\mathcal{I}$ is $k_{\max}$-approximate by (backward) induction on the number of iterations of the first while loop. At the induction base, $w^t(\mathcal{I}) = 0$, since $\mathcal{I} \subseteq \mathcal{D}_0^t$. For the induction step, consider the ith iteration of the first while loop, and assume that $\mathcal{I}$ is $k_{\max}$-approximate with respect to w^i. We show that $\mathcal{I}$ is $k_{\max}$-approximate with respect to $w_1^i = w^{i-1} - w^i$, and it follows by the Local Ratio Lemma that $\mathcal{I}$ is $k_{\max}$-approximate with respect to w. Since $\mathcal{I}$ is $k_{\max}$-independent, we have that

$$w_1^i(\mathcal{I}) = \sum_{D \in \mathcal{I}} w_1^i(D) = \sum_{D \in \mathcal{I} \setminus \mathcal{D}_0^{i-1}} \varepsilon \cdot \left|P^{i-1} \cap D\right| = \varepsilon \sum_{p \in P^{i-1}} \left|\left\{D \in \mathcal{I} \setminus \mathcal{D}_0^{i-1}\colon\ p \in D\right\}\right| \leqslant \varepsilon k_{\max} \cdot \left|P^{i-1}\right|.$$

On the other hand, any feasible solution $\mathcal{S}$ must cover any point $p \in P^{i-1}$ at least once. Moreover, even if $\mathcal{D}_0^{i-1} \subseteq S$, $\mathcal{S}$ must cover any point $p \in P^{i-1}$ with at least one disk that does not belong to $\mathcal{D}_0^{i-1}$, otherwise p would have been removed from P prior to the ith iteration. Hence, $w_1^i(\mathcal{S}) \geqslant \varepsilon \cdot |P^{i-1}|$. The lemma follows. □

Since $\mathcal{C}$ is feasible due to Lemma 2 and $w(\mathcal{C}) = 3^\alpha \cdot w(\mathcal{I})$ we obtain the following result.

Theorem 1. *Algorithm* **DiskMC** *is a* $3^\alpha k_{\max}$*-approximation algorithm for* Disk Multicover.

4. Multicovering polygons with disks

In this section we present our algorithm for Polygon Disk Multicover. The algorithm is based on a reduction to Disk Multicover that incurs a factor of $(1+\varepsilon)$ in the approximation ratio.

4.1. Discretizing the radii

Consider an instance of Polygon Disk Multicover and a feasible solution r. First, observe that one may assume that $r(b) \leqslant \max_{p \in P} \|b, p\|_2$, for every b. Otherwise we can decrease the radius of any base station b for which $r(b) > \max_{p \in P} \|b, p\|_2$ without affecting feasibility. We define a possibly larger upper bound $\rho_1(b) \stackrel{\text{def}}{=} \max_{p \in P} \|b, p\|_1$, for every base station b. The advantage of ρ_1 is that its binary representation is polynomial in the input size.

The next step is to look for lower bounds on the radii. Let u and v be two vertices on the boundary of the polygon P. The segment between u and v must be covered at least once by disks. Hence, the optimal disk cover of this segment is a lower bound on the optimum. Moreover, the optimum with respect to the weight function $w'(b) = w_{\min}$, for every b, where $w_{\min} = \min_b w(b)$, is not larger than the optimum with respect to w. Now, assuming that we are allowed to move base stations and since $\alpha \geqslant 1$, the cheapest way to cover the segment is to place all base stations on the segment, such that all have radius of $\frac{1}{2m}\|u, v\|_2$. Hence,

$$\sum_{b \in B} w_{\min} \rho^\alpha \leqslant \sum_b w_{\min} \left(\frac{1}{2m} \|u, v\|_2 \right)^\alpha \leqslant \text{OPT}, \tag{1}$$

where $\rho \stackrel{\text{def}}{=} \frac{1}{2m}\|u, v\|_\infty$.

Given a parameter $\varepsilon \in (0, 1)$, we define a radius assignment ρ_0 as follows: $\rho_0(b) = \varepsilon\rho \cdot \frac{w_{\min}}{w(b)}$, for every $b \in B$. Notice that the binary representation of ρ_0 is polynomial in the input size.

Lemma 4. $w(\rho_0) \leqslant \varepsilon \cdot \text{OPT}$.

Proof. Since $\alpha \geqslant 1$ we have that

$$w(\rho_0) = \sum_{b \in B} w(b)\rho_0(b)^\alpha = \sum_{b \in B} w(b) \left(\varepsilon\rho \cdot \frac{w_{\min}}{w(b)} \right)^\alpha = \varepsilon^\alpha \sum_{b \in B} \rho^\alpha \frac{w_{\min}^\alpha}{w(b)^{\alpha-1}} \leqslant \varepsilon \sum_{b \in B} w_{\min} \rho^\alpha,$$

and the lemma follows from (1). □

We now define a set of disks $\mathcal{D}$. The set of disks centered at b are

$$\mathcal{D}(b) = \left\{ \text{DISK}\left[b, \rho_0(1+\varepsilon)^\ell\right] \colon \ \ell \in \mathbb{N}, \ \rho_0(1+\varepsilon)^\ell < \rho_1 \right\},$$

and $\mathcal{D} = \bigcup_b \mathcal{D}(b)$.

Observation 5. $|\mathcal{D}|$ *is polynomial in the input size.*

Proof. This is implied by $|\mathcal{D}(b)| \leqslant \log_{1+\varepsilon} \frac{\rho_1(b)}{\rho_0(b)}$, for every $b \in B$. □

By limiting ourselves to $\mathcal{D}$ we increase the optimal solution by a small factor.

Lemma 6. $\text{OPT}(D) \leqslant [(1+\varepsilon)^\alpha + \varepsilon]\text{OPT}$.

Proof. Consider an optimal solution r. If $r(b) \geqslant \rho_0(b)$, then there exists a disk $D \in \mathcal{D}(b)$ whose radius is at most $r(b)(1+\varepsilon)$. Hence, $w(D) \leqslant (1+\varepsilon)^\alpha w(b) r(b)^\alpha$. If $0 < r(b) < \rho_0(b)$, then we take the disk $\text{DISK}[b, \rho_0(b)]$. By Lemma 4 we have that the total weight of all disks of the form $\text{DISK}[b, \rho_0(b)]$, for some b, is at most $\varepsilon \cdot \text{OPT}$. □

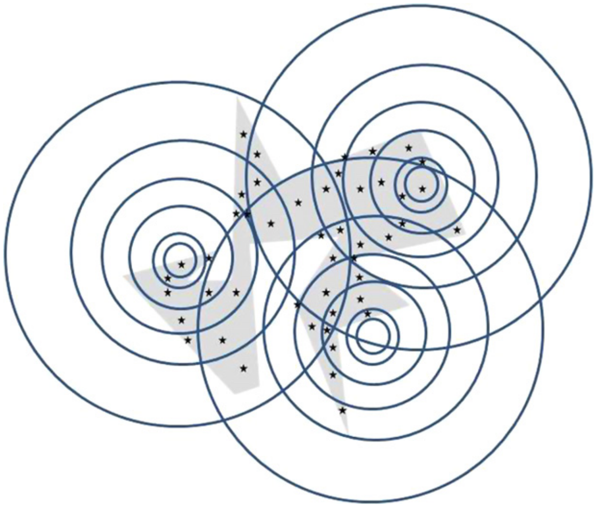

Fig. 1. An example of an arrangement. Representatives are shown as stars.

4.2. Going from polygon to points

Consider the arrangement of $\mathcal{D}$ and the polygon P. The number of intersections in the arrangement of $\mathcal{D}$ and P is bounded by $2\binom{n+|\mathcal{D}|}{2}$. Due to Observation 5, it follows that the number of faces in the arrangement is polynomial. Let q_f be any point that lies strictly in the face f, for every face f in the arrangement that is contained in P. Such a point q_f is called the *representative* of f. Define $Q \stackrel{\text{def}}{=} \{q_f\colon\ f \text{ is a face}\}$. See example of an arrangement and with its representatives in Fig. 1.

Lemma 7. *Let $\mathcal{S} \subseteq \mathcal{D}$. $\mathcal{S}$ is a multicover of Q if and only if it is a multicover of P.*

Proof. Since $Q \subseteq P$, if $\mathcal{S}$ is a multicover of P, it must be also a multicover of Q. For the other direction, consider a face f and a disk $D \in \mathcal{D}$. Either $f \subseteq D$ or $f \cap D = \emptyset$. Hence, a cover of q_f by a disk D means that D also covers f. □

By Lemma 7 we have that by computing a set Q of representatives and a multicover of Q by disks from $\mathcal{D}$, we may obtain an approximate solution. The set of faces in the arrangement can be computed using the sweep line method [11,7]. More specifically, the algorithm from [10] with a few minor modifications gives us an $O(N \log N)$ time algorithm that computes a set Q that contains a representative q_f for every face $f \subseteq P$, where N is the number of intersections in the arrangement.

This leads to the following result:

Theorem 2. *There exists a polynomial time $(3^\alpha K + \varepsilon)$-approximation algorithm for* Polygon Disk Multicover *with $\alpha \geqslant 1$, for every constant $\varepsilon > 0$.*

Proof. Due to Lemma 7 we can use Algorithm **DiskMC** to find an approximate multicover of the point set Q using the disk set $\mathcal{D}$. Lemma 6 implies that the computed solution is $(3^\alpha K + \varepsilon)$-approximate. □

We note that a similar result can be obtained for the problem of multicovering the boundary of the polygon P. In this case we need to pick a representative for every edge in the arrangement that is contained in the boundary of the polygon.

References

[1] A.K. Abu-Affash, P. Carmi, M.J. Katz, G. Morgenstern, Multi cover of a polygon minimizing the sum of areas, in: 5th International Workshop on Algorithms and Computation, in: LNCS, vol. 6552, 2011, pp. 134–145.

[2] H. Alt, E.M. Arkin, H. Brönnimann, J. Erickson, S.P. Fekete, C. Knauer, J. Lenchner, J.S.B. Mitchell, K. Whittlesey, Minimum-cost coverage of point sets by disks, in: 22nd ACM Symposium on Computational Geometry, 2006, pp. 449–458.

[3] R. Bar-Yehuda, K. Bendel, A. Freund, D. Rawitz, Local ratio: a unified framework for approximation algorithms, ACM Computing Surveys 36 (4) (2004) 422–463.

[4] R. Bar-Yehuda, S. Even, A local-ratio theorem for approximating the weighted vertex cover problem, Annals of Discrete Mathematics 25 (1985) 27–46.

[5] V. Bilò, I. Caragiannis, C. Kaklamanis, P. Kanellopoulos, Geometric clustering to minimize the sum of cluster sizes, in: 13th Annual European Symposium on Algorithms, in: LNCS, vol. 3669, 2005, pp. 460–471.

[6] C. Chekuri, K.L. Clarkson, S. Har-Peled, On the set multi-cover problem in geometric settings, in: 25th ACM Symposium on Computational Geometry, 2009, pp. 341–350.
[7] M. de Berg, O. Cheong, M. van Kreveld, M. Overmars, Computational Geometry: Algorithms and Applications, 3rd edition, Springer-Verlag, 2008.
[8] A. Freund, D. Rawitz, Combinatorial interpretations of dual fitting and primal fitting, in: 1st Workshop on Approximation and Online Algorithms, in: LNCS, vol. 2909, 2003, pp. 137–150.
[9] N. Lev-Tov, D. Peleg, Polynomial time approximation schemes for base station coverage with minimum total radii, Computer Networks 47 (4) (2005) 489–501.
[10] J. Nievergelt, F.P. Preparata, Plane-sweep algorithms for intersecting geometric figures, Communications of the ACM 25 (10) (1982) 739–747.
[11] F.P. Preparata, M.I. Shamos, Computational Geometry: An Introduction, Springer-Verlag, 1985.