Maximizing Covered Area in the Euclidean Plane with Connectivity Constraint

Chien-Chung Huang
DI ENS, École Normale supérieure, Université PSL, Paris, France
CNRS, Paris, France
chien-chung.huang@ens.fr

Mathieu Mari
DI ENS, École Normale supérieure, Université PSL, Paris, France
mathieu.mari@ens.fr

Claire Mathieu
CNRS, Paris, France
clairemmathieu@gmail.com

Joseph S. B. Mitchell
Stony Brook University, Stony Brook, NY 11794, USA
joseph.mitchell@stonybrook.edu

Nabil H. Mustafa
Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, ESIEE Paris, France
mustafan@esiee.fr

Abstract

Given a set $\mathcal{D}$ of n unit disks in the plane and an integer $k \leq n$, the maximum area connected subset problem asks for a set $\mathcal{D}' \subseteq \mathcal{D}$ of size k that maximizes the area of the union of disks, under the constraint that this union is connected. This problem is motivated by wireless router deployment and is a special case of maximizing a submodular function under a connectivity constraint.

We prove that the problem is NP-hard and analyze a greedy algorithm, proving that it is a $\frac{1}{2}$-approximation. We then give a polynomial-time approximation scheme (PTAS) for this problem with resource augmentation, i.e., allowing an additional set of εk disks that are not drawn from the input. Additionally, for two special cases of the problem we design a PTAS without resource augmentation.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases approximation algorithm, submodular function optimisation, unit disk graph, connectivity constraint

Digital Object Identifier 10.4230/LIPIcs.APPROX-RANDOM.2019.32

Category APPROX

Related Version `https://www.di.ens.fr/~mmari/content/papers/MACS.pdf`

Funding *Joseph S. B. Mitchell*: Partially supported by the National Science Foundation (CCF-1540890), the US-Israel Binational Science Foundation (CCF-1540890), and DARPA (Lagrange).
Nabil H. Mustafa: The work of Nabil H. Mustafa in this paper has been supported by the grant ANR SAGA (JCJC-14-CE25-0016-01).

1 Introduction

Maximizing a submodular function[1] under constraints is a classical problem in computer science and operations research [8, 23]; the most commonly studied constraints are cardinality, knapsack and matroids constraints. A natural constraint that has received little attention is

[1] Given a set X, a function $f : 2^X \to \mathbb{R}$ is *submodular* if given any two subsets A, $B \subseteq X$, $f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$.

the *connectivity* constraint. In this paper, we study the following problem: given a set of n unit disks in the plane, select a subset of k disks that maximize the area of their union, under the constraint that this union is connected. We call this problem the *Maximum Area Connected Subset* problem (MACS). Notice that the area covered by a set of disks is a monotone submodular function.

The problem is motivated by wireless router deployment, first introduced in [14], where the goal is to install a given number of routers to maximize the number of clients covered. When the clients are uniformly spread in the plane, the number of clients in a region can be approximated by the area of that region, leading to our problem.

Our Contributions

We first analyze a variant of the greedy algorithm and show that it computes a $\frac{1}{2}$-approximation to MACS (Theorem 3); further, the analysis is tight. In contrast, the natural algorithm that greedily adds one disk at a time can end up with a solution with area a factor of $\Omega(k)$ worse than the optimal solution.

To improve upon the $\frac{1}{2}$-approximation ratio, we turn to the resource augmentation setting in which the algorithm is allowed to add a few additional disks that need not be drawn from the input. We design a PTAS for the resource augmentation version of the problem using Arora's shifted quadtree technique (Theorem 4)[2]. The proof hinges on the existence of a near-optimal solution with $O(\varepsilon k)$ additional disks and with additional structure that allows its computation by dynamic programming.

As a corollary, we show that for two special cases of MACS we can in fact design a PTAS without resource augmentation: *i*) when the Euclidean distances between centers of the input disks are well-approximated by shortest paths in their intersection graph (Corollary 6), and *ii*) when every point of the relevant region of the Euclidean plane is covered by at least one input disk (Corollary 9).

On the other hand, via a reduction from the Rectilinear Steiner Tree problem, we show that MACS is NP-hard (Theorem 3). We also show that MACS for the input of a set of quadrilaterals instead of disks, the problem is APX-hard (Theorem 12). We leave open the question whether MACS is APX-hard or admits a PTAS without resource augmentation.

Related work

Maximising a monotone submodular function under constraint(s) is a subject that has received a large amount of attention over the years. We refer the readers to [2, 5, 6, 8, 13, 15, 23] and the references therein. Our problem can be regarded as maximising a submodular function under a cardinality (knapsack) constraint and a connectivity constraint. Notice that the connectivity constraint is central to the difficulty of our problem: without connectivity constraints, MACS admits a PTAS even in the more general case of convex pseudodisks [4]. However even without the connectivity constraint the problem remains NP-hard[3].

Another motivation for studying the connectivity constraint is related to cancer genome studies. Suppose that a vertex represents an individual protein (and associated gene), an edge represents pairwise interactions, and each vertex has an associated set. Finding the connected subgraph of k genes that is mutated in the largest number of samples is equivalent to the problem of finding the connected subgraph with k nodes that maximizes the cardinality of the union of the associated sets (see [21]).

---

[2] We also develop an alternative algorithm using Mitchell's m-guillotine dissection technique. See the full version for details.

[3] The reduction is from Maximum independent set problem that is NP-hard in unit-disk graphs.

In the general (non-geometric) setting, there exists a $O\left(\frac{1}{\sqrt{k}}\right)$-approximation algorithm for maximizing a monotone submodular function [14]. Our results show that when the submodular function and the connectivity are induced by a geometric configuration, the approximation ratio can be significantly improved.

We next consider several related problems where the connectivity constraint plays an important role. The goal of the node-cost budget problem [20] is to find a connected set of vertices in a general graph to collect the maximum profit on the vertices while guaranteeing the total cost does not exceed a certain budget. Notice that in this setting the submodular function is a simple additive function of the profits. Another related problem [3] is to assign radii to a given set of points in the plane so that the resulting set of disks is connected and the objective is to minimize the sum of radii.

Khuller et al. [12] study the budgeted connected dominating set problem where given a general undirected graph, the goal is to select k vertices whose induced subgraph is connected and that maximizes the number of dominated vertices. It was pointed out to us that their algorithm can be used to give a $\frac{1}{13}\left(1 - \frac{1}{e}\right)$-approximate solution for MACS. The authors of [10] consider the problem of selecting k nodes of an input node-weighted graph to form a connected subgraph, with the aim of maximizing or minimizing the selected weight.

We now turn to the geometric setting. A logarithmic-factor approximation algorithm is known [9] for the connected sensor coverage problem in which one must select at most k sensors in the plane forming a connected communication network and covering the desired region, where the region covered by each sensor is convex [7, 11]. A $(1 - \varepsilon)$-approximation algorithm in time $n^{O(1/\varepsilon)}$ for the maximum independent set problem on unit disk graphs is known [17]. The authors of [16] present a constant-factor approximation algorithm for several problems on unit disk graphs, including maximum independent set. For the geometric set cover problem where the goal is to cover a given set of input points with a minimum number of given disks, a PTAS is possible [18].

2 Our results

The Euclidean distance between two points x and y is denoted by $\|x - y\|$. When there is no confusion, we will refer to a point x in the plane and the unit disk centered at x interchangeably.

▶ **Definition 1.** *Given a finite set S in the plane, the* unit disk intersection graph $\mathrm{UDG}(S)$ *is a graph on S where $\{x, y\} \subseteq S$ is an edge of $\mathrm{UDG}(S)$ if and only if $\|x - y\| \leq 2$.*

A set S of points in the plane are *connected* if $\mathrm{UDG}(S)$ is a connected graph.

▶ **Definition 2.** *The Maximum Area Connected Subset (*MACS*) problem is as follows.*
***Input**: a finite set of points $X \subseteq \mathbb{R}^2$ and a non-negative integer k, where $k \leq |X|$.*
***Output**: a subset $S \subseteq X$ of size at most k such that the unit-disk graph $\mathrm{UDG}(S)$ of S is connected.*
***Goal**: maximize the area of the union of the unit disks centered at the points of S.*
The optimal solution of MACS *on input (X, k) is denoted by $\boldsymbol{OPT}(X, k)$.*

When the context is clear, we refer to $\mathbf{OPT}(X, k)$ as $\mathbf{OPT}$, which is also used to denote the area covered by the optimal solution; observe that $\mathbf{OPT}$ is trivially upper-bounded by πk. Any $S \subseteq X$ with $|S| \leqslant k$ for which $\mathrm{UDG}(S)$ is connected is called a *feasible solution.*

We state our main results below. All omitted proofs and figures can be found in the appendix or in the full version.

▶ **Theorem 3** (Approximation). *There exists a polynomial-time algorithm that computes a $\frac{1}{2}$-approximation for* MACS *(Algorithm 1).*

With resource augmentation, we obtain a $(1-\varepsilon)$-approximation.

▶ **Theorem 4** (Resource augmentation). *Let $\varepsilon > 0$ be a given parameter. Given a set of points $X \subseteq \mathbb{R}^2$ and a non-negative integer k, there is an algorithm (Algorithm 2) that computes, in time $n^{\mathcal{O}(\varepsilon^{-3})}$, a subset $S \subseteq X$ of size at most k and a set $S_{add} \subseteq \mathbb{R}^2$ of at most εk points, such that* $\mathrm{UDG}(S \cup S_{add})$ *is connected and the area of the union of the unit disks centered at S is at least $(1-\varepsilon)\boldsymbol{OPT}(X,k)$.*

Theorem 5 can be obtained alternatively by a (deterministic) guillotine cut approach with a faster running time. We leave that for the full version of the paper.

Let $d_G(x,y)$ denote the distance between two vertices x and y of G. A set X of points in the plane is called *α-well-distributed* if $\mathrm{UDG}(X)$ is an α-spanner for X [19]:

▶ **Definition 5.** *Given $\alpha > 0$, a finite set X of points in the plane is called* α-well-distributed *if for all $x, y \in X$, $d_{UDG(X)}(x,y) \leqslant \lceil \alpha \cdot \|x-y\| \rceil$.*

▶ **Corollary 6.** *There exists a* PTAS *for* MACS *on α-well-distributed inputs, where α is a fixed constant (Algorithm 3).*

▶ **Definition 7.** *A set X is called* pseudo-convex *if the convex-hull of X is covered by the union of the unit disks centered at points of X.*

▶ **Lemma 8.** *A pseudo-convex set X is* 3.82*-well-distributed.*

▶ **Corollary 9.** MACS *on pseudo-convex inputs admits a polynomial-time approximation scheme.*

We next turn to the hardness of MACS.

▶ **Theorem 10** (Hardness). MACS *is NP-hard.*

▶ **Definition 11.** *The* QUAD-CONNECTED-COVER *is defined as follows.*
***Input**: a set $\mathcal{T}$ of n convex quadrilaterals in the plane, and an integer k.*
***Output**: a subset T of $\mathcal{T}$ of size k such that the intersection graph of T is connected.*
***Goal**: Maximise the area covered by the union of quadrilaterals in T.*

▶ **Theorem 12.** QUAD-CONNECTED-COVER *is APX-hard.*

3 Proof of Theorem 3: the Two-by-two algorithm

In the section we present a simple $\frac{1}{2}$-approximation for MACS based on a greedy approach: we iteratively add two unit disks that maximize the area covered while maintaining feasibility. Interestingly, the algorithm that adds disks one at a time is not a constant approximation algorithm. See Figure 1 for an example. Moreover, trying all possible sets of s disks, for any $s \geq 3$, in the neighborhood of the current solution does not improve the approximation ratio. This can be seen on Figure 2 where the first disk chosen by the algorithm is not x, but x_s.

Let B_x denote the unit disk centered at $x \in \mathbb{R}^2$ and $B(S) = \bigcup_{x \in S} B_x$ denote their union. The area covered by a set $C \subset \mathbb{R}^2$ is denoted by $\mathcal{A}(C)$. When $C = B(S)$, its area is simply written as $\mathcal{A}(S)$. Given a graph G, $G[S]$ denotes the subgraph induced by a subset S of vertices. A subset of the vertices of a graph is a *dominating set* if every vertex belongs to the set or is adjacent to some vertex of it.

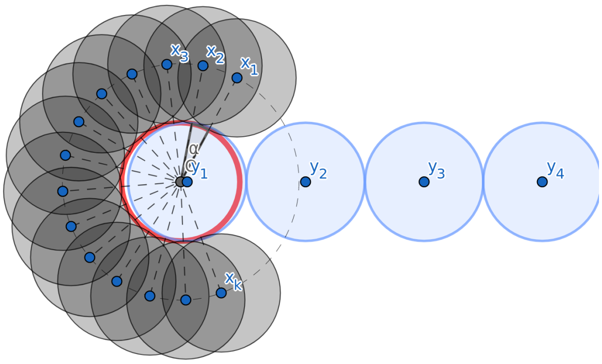


Figure 1 The greedy algorithm that adds only one connected disk maximising the marginal area covered is not a constant factor algorithm. For any $k \geqslant 0$ and $\varepsilon > 0$, consider the above input where $O = (0, 0)$, and $y_i = (2(i-1)+\varepsilon, 0)$ for all i. Then, put all $x_1, \dots, x_k$ evenly spaced (by an angle α) on a circle of radius 2 around O so that none of them intersect y_2. Each light grey regions are covered by only one disk x_i so the marginal gain of adding x_i to any solution is at least the area of one of these regions, say $a > 0$. If ε is chosen such that $\mathcal{A}(B_{y_1} \setminus B_O) < a$, then if the algorithm starts by picking disk O, it will then choose all x_j, so that the area covered by the solution is upper-bounded by the area of a radius 3 disk, 9π, while the optimal solution (disks y_i) has area πk.

One can find an example similar to Figure 2 to show that optimising the initial choice of the first disk(s) does not improve the approximation ratio.

▶ **Theorem 3** (Approximation). *There exists a polynomial-time algorithm that computes a $\frac{1}{2}$-approximation for* MACS *(Algorithm 1).*

We can assume w.l.o.g. that UDG(X) is connected; otherwise we return the maximum value over all connected components. The execution of Algorithm 1 is divided in two phases. An iteration belongs to the first phase as long as the current solution S is not a dominating set in the graph UDG(X).

During the first phase, in each iteration the area covered increases by at least π. During the second phase, since the current solution is a dominating set, any disk can be added while keeping the solution feasible. Therefore the algorithm reduces to a standard greedy algorithm to maximize a submodular function, and the analysis is similar to the proof that Nemhauser's algorithm is a $\left(1-\frac{1}{e}\right)$-approximation for classic submodular functions.

Algorithm 1 The Two-by-two algorithm for MACS.

Input: $X \subseteq \mathbb{R}^2, k \geq 0$, where X is finite and $k \leq |X|$.
Output: a feasible set of size k.

if k *is even* **then**
  $S \leftarrow$ any two intersecting disks of X;
else
  $S \leftarrow$ any one disk of X;
while $|S| \leqslant k-2$ **do**
  $\{x, x'\} \leftarrow \arg\max \{\mathcal{A}(S \cup \{x, x'\}) : x, x' \in X,\ S \cup \{x, x'\} \text{ is feasible }\}$;
  $S \leftarrow S \cup \{x, x'\}$;
return S;

Proof. We first analyze the even case where $k = 2\kappa$, and then we reduce the odd case to the even one. Let $S_\kappa = \{x_1, x_2, \dots, x_{2\kappa}\}$ be the solution returned by the algorithm. Let $S_i = \{x_1, \dots, x_{2i}\}$ be the set right before the i-th iteration and let d be the smallest integer such that S_d is a dominating set in $\mathrm{UDG}(X)$. If such an integer does not exist, i.e., S_κ is not a dominating set, then set $d = \kappa$.

▷ Claim 13. The area $\mathcal{A}(S_d)$ is at least πd.

Proof. For $i < d$, S_i is not a dominating set. Then there exist two disks y, y' such that $B(S_i) \cap B_y = \emptyset$ and $S \cup \{y, y'\}$ is connected. Adding such a pair increases the area covered by at least $\mathcal{A}(B_y) = \pi$. Since (x_{2i+1}, x_{2i+2}) is chosen to maximize $\mathcal{A}(S_i \cup \{x, x'\})$ among all feasible pairs, $\mathcal{A}(S_{i+1}) \geqslant \mathcal{A}(S_i \cup \{y, y'\}) \geqslant \mathcal{A}(S_i) + \pi$. By induction, $\mathcal{A}(S_d) \geqslant \pi d$. ◁

Note that when $d = \kappa$, Claim 13 immediately implies that $\mathcal{A}(S_\kappa) \geqslant \frac{\mathbf{OPT}}{2}$. Also regardless of the initial choice, the area covered by the first two disks is at least π. This observation will be useful when we prove the case where k is odd.

▷ Claim 14. For all $d \leqslant i \leqslant \kappa$, $\mathcal{A}(\mathbf{OPT}) \leqslant \mathcal{A}(S_i) + \kappa \cdot (\mathcal{A}(S_{i+1}) - \mathcal{A}(S_i))$.

Proof. It is easy to check that the function $\mathcal{A}(\cdot)$ satisfies the following properties for all $H \subseteq H' \subseteq X$:

positivity: $\mathcal{A}(H) \geqslant 0$.
monotonicity: $\mathcal{A}(H) \leqslant \mathcal{A}(H')$.
submodularity: $\forall H'' \subseteq X,\ \ \mathcal{A}(H' \cup H'') \leqslant \mathcal{A}(H \cup H'') - \mathcal{A}(H) + \mathcal{A}(H')$.

Let $\mathbf{OPT} = \{y_1, \dots, y_{2\kappa}\}$. We have for all $d \leqslant i \leqslant \kappa$:

$$\begin{aligned}
\mathcal{A}(\mathbf{OPT}) \leqslant\ & \mathcal{A}(S_i \cup \mathbf{OPT}) \\
&= \mathcal{A}(S_i) + (\mathcal{A}(S_i \cup \{y_1, y_2\}) - \mathcal{A}(S_i)) + \dots \\
&\qquad + (\mathcal{A}(S_i \cup \{y_1, \dots, y_{2\kappa}\}) - \mathcal{A}(S_i \cup \{y_1, \dots, y_{2\kappa-2}\})) \\
&\leqslant \mathcal{A}(S_i) + (\mathcal{A}(S_i \cup \{y_1, y_2\}) - \mathcal{A}(S_i)) + \dots + (\mathcal{A}(S_i \cup \{y_{2\kappa-1}, y_{2\kappa}\}) - \mathcal{A}(S_i)) \\
&\leqslant \mathcal{A}(S_i) + \kappa \cdot (\mathcal{A}(S_i \cup \{x_{2i+1}, x_{2i+2}\}) - \mathcal{A}(S_i)) \\
&= \mathcal{A}(S_i) + \kappa \cdot (\mathcal{A}(S_{i+1}) - \mathcal{A}(S_i)).
\end{aligned}$$

The first and the second inequality respectively come from *monotonicity* and *submodularity*, while the third one follows from the fact that for $i \geq d$, (x_{2i+1}, x_{2i+2}) is the pair of disks maximizing $\mathcal{A}(S_i \cup \{x, x'\})$ among **all pairs** (x, x') in X. As S_d is a connected dominating set in X, all pairs (y_{2j-1}, y_{2j}) for $1 \leqslant i \leqslant \kappa$ are considered. ◁

We can now re-write Claim 14 as

$$\text{For all } d \leqslant i \leqslant \kappa : \mathcal{A}(S_{i+1}) \geqslant \left(1 - \frac{1}{\kappa}\right) \mathcal{A}(S_i) + \frac{\mathbf{OPT}}{\kappa}.$$

Combined with Claim 13, simple algebra yields that for $d \leqslant i \leqslant \kappa$, we have

$$\mathcal{A}(S_i) \geqslant \left[1 - \left(1 - \frac{d}{2\kappa}\right)\left(1 - \frac{1}{\kappa}\right)^{i-d}\right] \mathbf{OPT}.$$

Therefore, for $i = \kappa$ we have

$$\mathcal{A}(S) = \mathcal{A}(S_\kappa) \geqslant \left[1 - \left(1 - \frac{d}{2\kappa}\right)\left(1 - \frac{1}{\kappa}\right)^{\kappa-d}\right] \mathbf{OPT} = \left[1 - \frac{1}{2}(1+t)\left(1 - \frac{1}{\kappa}\right)^{\kappa t}\right] \mathbf{OPT}$$

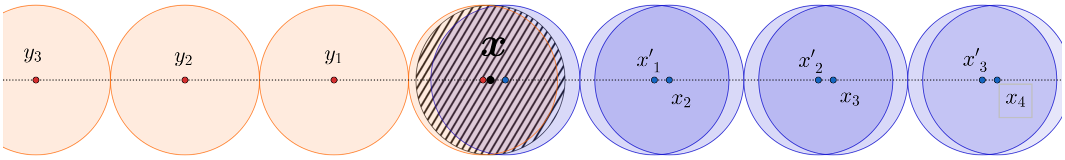


Figure 2 A tight example for Algorithm 1. For any $\varepsilon > 0$, X contains $x = (0, 0)$ (stripe-shaded), $x_i = (2(i-1) + i\varepsilon, 0)$ and $x'_i = ((2+\varepsilon)i, 0)$ for $1 \leqslant i \leqslant k$ (blue) and $y_i = (-2i - \varepsilon/2, 0)$ for $0 \leqslant i \leqslant k$ (orange). Suppose that $k = 1 + 2\kappa$ is odd and the algorithm starts with $S_0 := \{x, x\}$. Then the algorithm will add $\{x_i, x'_i\}$ in iteration i since it covers more additional area than $\{y_0, y_1\}$. The solution returned (blue disks) covers an area of $\pi + \kappa(\pi + f(\varepsilon)) \approx \frac{k}{2}\pi$, for some function $f(\cdot)$ with $\lim_{\varepsilon\to 0} f(\varepsilon) = 0$, while **OPT** (orange disks) covers an area $k\pi$.

where $t = \dfrac{\kappa - d}{\kappa} \in [0, 1]$. As $1 + x \leqslant e^x$ for all $x \in \mathbb{R}$, we get

$$\mathcal{A}(S) \geqslant \left(1 - \frac{1}{2}(1+t)e^{-t}\right) \mathbf{OPT} \geqslant \left(1 - \frac{1}{2}e^{t}e^{-t}\right) \mathbf{OPT} = \frac{1}{2}\mathbf{OPT},$$

concluding the proof of the case when k is an even number.

For the odd case $k = 2\kappa - 1$: in the first iteration, instead of adding two disks to S_1, we add a single disk of X to S_1. This is equivalent to adding two copies of the same disk. This iteration belongs to the first phase, and the only properties we used in the first phase is that each iteration adds an area of π, and keeps the solution feasible; these are clearly true for the first iteration even with one disk. ◀

Figure 2 shows a tight example.

4 Proof of Theorem 4: PTAS with resource augmentation

▶ **Theorem 4** (Resource augmentation). *Let $\varepsilon > 0$ be a given parameter. Given a set of points $X \subseteq \mathbb{R}^2$ and a non-negative integer k, there is an algorithm (Algorithm 2) that computes, in time $n^{\mathcal{O}(\varepsilon^{-3})}$, a subset $S \subseteq X$ of size at most k and a set $S_{add} \subseteq \mathbb{R}^2$ of at most εk points, such that* UDG $(S \cup S_{add})$ *is connected and the area of the union of the unit disks centered at S is at least* $(1-\varepsilon)\boldsymbol{OPT}(X,k)$.

We first summarise the high level ideas; the details are then presented in subsequent sections. Let (X, k) denote an input of MACS and **OPT** be the optimal solution of MACS on input (X, k). When the context is clear **OPT** can also denote the total area covered by the union of the unit disks centered in points of **OPT**.

We start by guessing a bounding box of size $\Theta(k) \times \Theta(k)$ that contains **OPT**. Next, another square of size $L \times L$, where $L = \Theta(k)$, is randomly shifted so that it always contains the bounding box. We remove all disks that are outside the square. That square is then recursively partitioned into smaller squares until they have (large) constant size. This hierarchical dissection induces a grid.

We remove all disks that intersect the lines of the grid. In contrast, we deploy some new disks (X_{add}) in some strategic *portal* positions along the lines and near the boundary of all the smallest squares.

Next, we use dynamic programming to build a solution from the smallest squares upwards. The difficulty lies in having to guarantee the connectivity when combining solutions from smaller squares into larger squares using additional disks, while controlling the time complexity and the number of disks added.

The key of our approach lies in Lemma 20, in which we argue that with constant probability, there exists a well-structured near-optimal solution that uses at most εk additional disks.

4.1 The grid

The first step is to reduce significantly the size of the input by guessing the position of the optimal solution.

▶ **Lemma 15.** *There exists a point $c \in X$ such that $\boldsymbol{OPT}$ is contained in an axis-parallel square of side length $4k$ and centered in c.*

Proof. For c, take any disk in $\mathbf{OPT}$ and recall that $\mathbf{OPT}$ is connected and has at most k disks, so all the disks in $\mathbf{OPT}$ are contained in the square centered at c and with side length $4k$. ◀

Given the randomly shifted hierarchical dissection, we use the same terminology as Vazirani [22, Chapter 11] to define the *root square*, the *shift* of the dissection, the *horizontal* and *vertical lines*, the *levels* of squares and of lines of the dissection, and the *portals*. The recursive dissection stops when a square has side length $L_0 = \Theta(\varepsilon^{-1})$ (*leaf square*). Portals are either at the intersection of grid lines or distributed along the grid lines (with varying density). We make some observations here (all details and proofs are in the following section and the appendix). First, the distance between two consecutive portals on a line at level ℓ is $\mathcal{O}(L/(m2^{\ell}))$, where m represents the density of portals on the grid. The greater this parameter, the greater the accuracy of the solution and higher the running time. Choosing $m = \Theta(\varepsilon^{-1}\log(L/L_0)) = \mathcal{O}(\varepsilon^{-1}\log(\varepsilon k))$ allows us to compute a near-optimal solution in polynomial time.

▶ **Observation 16.** *If an horizontal line of level ℓ crosses a vertical line of level greater than or equal to ℓ then the intersection point is a horizontal portal.*

We define a set $\mathcal{P}$ of *portal disks* which we position at or near the portals. If a portal (i, j) is on exactly one line of the grid then we add the portal disk (i, j) to $\mathcal{P}$. If a portal (i, j) is at the intersection of two lines of the grid, then *i*) if it is a horizontal portal then we add to $\mathcal{P}$ two portal disks $(i, j+2)$ and $(i, j-2)$, and *ii*) if it is a vertical portal then we add to $\mathcal{P}$ two portal disks $(i-2, j)$ and $(i+2, j)$.

Given a square C of the dissection, the *potential portal disks* of C, denoted by $\mathcal{P}_C$, are the portal disks on the boundary of C.

▶ **Observation 17.** *For any square, the number of potential portal disks is $\mathcal{O}(m) = \mathcal{O}(\varepsilon^{-1}\log(\varepsilon k))$.*

The *border* of a leaf square C, denoted as ∂C, is the set of points in C within distance 1 from C's boundary. The remaining points of C are called the *core* of C, written as *core*(C). A unit disk with its center in C intersects the boundary if and only if its center lies in the border. If two disks are in the core of two different leaf squares, then they do not intersect. We refer to the union of the core of all *leaf squares* as the *core*. In a leaf square $C = [a, b] \times [c, d]$, the set of points formed by the boundary of the square $[a+2, b-2] \times [c+2, d-2]$ is called the *fence*. We cover the fence of C by *fence disks*, aligned such that each corner of this square is the center of a fence disk. See Figure 4. We denote by $\mathcal{F}$ the set of all fence disks for all leaf squares. The set of portal disks and fence disks form the set of *additional disks* $X_{add} = \mathcal{P} \cup \mathcal{F}$.

4.2 Detailed construction of the grid

Let L' be the sidelength of the box given by the Lemma 15, and set X' be the set of points of X lying inside this box. Let L be the smallest power of 2 greater than $2L'$. The *root square* is defined to be the axis-parallel $L \times L$ square with the same left-bottom corner as the bounding-box.

A *shift* is an non-negative integer a smaller than or equal to $L/2$. We say that the root square is *shifted* by a if it is translated by the vector $(-a, -a)$. Notice that any shifted root square contains the bounding-box.

Given a shifted root square, we can define its *dissection* as a recursive partitioning into smaller squares. The $L \times L$ root square is divided into four squares of size $L/2 \times L/2$. Each of these squares is again divided into four $L/4 \times L/4$ squares, so forth. The process stops when the side length of a square is equal to $L_0 = \Theta(\varepsilon^{-1})$. Let $d = \log(L/L_0) = \mathcal{O}(\log(\varepsilon k))$. We can think of this partitioning as 4-ary tree, where each node at level ℓ corresponds to a $L_0 2^\ell \times L_0 2^\ell$ square and has four children corresponding to four $L_0 2^{\ell-1} \times L_0 2^{\ell-1}$ squares. The root square is at level 0 and the *leaf squares* are at level d. Given two squares of level ℓ and level ℓ', $\ell > \ell'$, we say the former is of higher level than the latter. So the leaf square is the one with the highest level.

This dissection defines a *grid* composed of $2 \cdot (2^d - 1)$ horizontal and vertical lines of length L. We say that a line is at *level* $\ell \in \{1, \ldots, d\}$ if it was added on the grid to divide a square at level $l - 1$ into four squares at level ℓ. There are 2^ℓ horizontal (*resp.* vertical) lines at level ℓ. See figure 3.

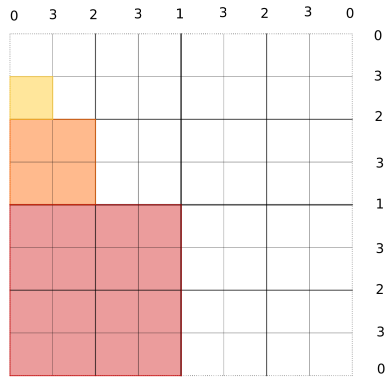


Figure 3 An illustration of the grid with $d = 3$. Numbers on the top and the right are the level of the corresponding lines and the red, orange and yellow are respectively the example of square of the dissection at level $1, 2$ and 3.

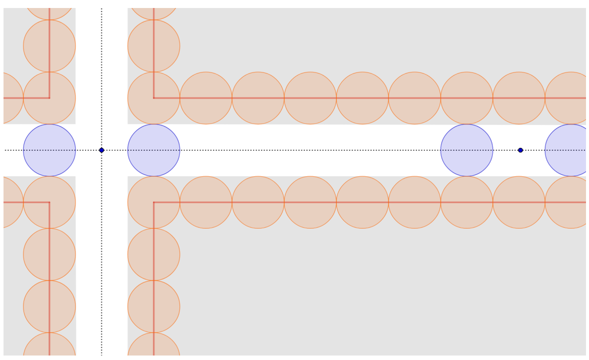


Figure 4 The grey and white area are respectively the *core* and the *border*. Dotted lines are from the grid while the orange lines represent the *fence* and orange disks are the *fence disks*. Blue points are (vertical) *portals* and blue disks are *portal disks*.

On each horizontal line of level $\ell \geqslant 1$, we will place a set of *vertical* – notice the naming asymmetry – *portals* of level ℓ, near which (not exactly on which) we will deploy the portal disks to facilitate the connection of disks on both sides of this line. We define a set of horizontal portals for each vertical line in an analogous manner. Notice that it is possible that a point is both a vertical portal and a horizontal portal. Let $m = \mathcal{O}(\varepsilon^{-1} d)$ be a power of two. Along a line of level ℓ, there are $m2^\ell + 1$ portals evenly spaced so that the distance between two neighboring portals have distance exactly $\dfrac{L}{m2^\ell}$.

4.3 Dynamic program

The algorithm uses dynamic programming. The dynamic programming table is indexed by *configurations*.

▶ **Definition 18.** *A* configuration *is a 5-tuple $\mathcal{C} = [C, t, t_{add}, P, \sim]$, where:*

- *C is a square of the dissection.*
- *$0 \le t \le k$ is an integer, denoting the number of disks of S used by the solution inside C.*
- *$0 \le t_{add} \le \varepsilon k$ is an integer, denoting the number of additional disks used by the solution inside C.*
- *$P \subseteq \mathcal{P}_C$ is a subset of potential portal disks of C, those that are used by the solution.*
- *$\sim$ is a planar connectivity relation on P (described below), representing the connectivity achieved so far by the part of the solution inside C.*

In the following, to facilitate discussion, we will refer to portals disks as simply portals. An equivalence relation $\sim$ on P is a *planar connectivity relation* if each equivalence class has an associated tree with the portals at the leaves, and there exists a planar embedding of those trees inside the square, such that the trees do not intersect.

The content of the dynamic programming table, the *value* of a configuration $\mathcal{C} = [C, t, t_{add}, P, \sim]$, denoted by $\mathcal{A}(\mathcal{C})$, is the maximum area that can be covered by a set $S \subseteq X$ of t disks in $C \cap core$[4], such that there is a set $S_{add} \subseteq X_{add}$ of t_{add} additional disks such that any $p, p' \in P$ with $p \sim p'$ are in the same connected component induced by $S \cup S_{add} \cup P$. We say that p and p' are *connected in* $\mathcal{C}$. If such sets $\{S, S_{add}, P\}$ do not exist for configuration $\mathcal{C}$, the value $\mathcal{A}(\mathcal{C})$ is set to $-\infty$.

4.4 Computing leaf entries of the dynamic programming

We first explain how to fill the entries of the table corresponding to the leaf squares. For each leaf square C, we enumerate

1. all possible subsets $S \subseteq X' \cap core(C)$ of at most k_0 disks, for a parameter $k_0 = \mathcal{O}(\varepsilon^{-3})$ (see Lemma 20).
2. all possible subsets $S_f \subseteq \mathcal{F} \cap C$,
3. all possible subsets $P \subseteq \mathcal{P}_C$, and
4. all possible planar connectivity relations $\sim$ on P.

We say that $(S, S_f, P, \sim)$ is a *guess* in C and that it is *usable* if one of the following two conditions holds:

Case 1. if $P = \emptyset$, then $S \cup S_f$ is connected, otherwise

Case 2. every connected component of $S \cup S_f \cup P$ contains at least one portal disk in P.

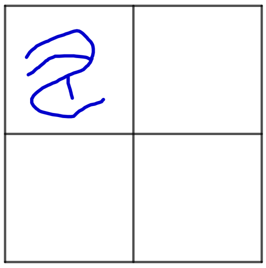
Each usable guess (S, S_f, P) in C corresponds to a configuration $\mathcal{C} := [C, |S|, |S_f|, P, \sim]$, where $\sim$ is the planar connectivity relation on P induced by the connected components of $S \cup S_f \cup P$.

Several usable guesses (S, S_f, P) can potentially correspond to the same configuration $\mathcal{C}$. The value of $\mathcal{C}$ is computed[5] as the maximum value $\mathcal{A}(S)$ over all such guesses S.

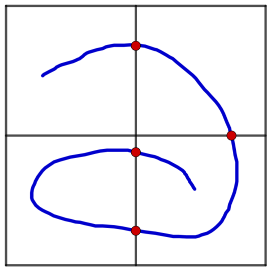
---

[4] Recall that *core* is the union of the $core(C)$ of all leaf squares C.

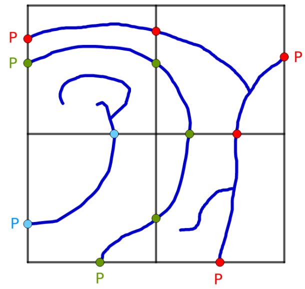
[5] The area covered by the union of a set of disks is a real number that can be computed exactly. When the desired accuracy is a fixed constant (for instance ε), one can give an approximation of this area with the desired precision in polynomial time.



(a) the top-left configuration is closed while other configurations are empty.



(b) there is unique connected component independent from the outside world.



(c) Each connected component contains a portal in P.

Figure 5 Illustration of cases (a)-(b)-(c) of point 6. in Definition 19.

4.5 Computing all entries

It remains to show how to compute the solution of a configuration, say $\mathcal{C} = [C, t, t_{add}, P, \sim]$, for a square C at level ℓ, by combining the solutions $\left[C^i, t^i, t^i_{add}, P^i, \sim^i\right]$ of the four child squares C^i, $i = 1, 2, 3, 4$, at level $\ell + 1$. Recall that connectivity relations $\sim^i$ capture the information about connectivity in the squares C^i. Let $P = \{p_0, \ldots, p_s\}$ be the subset of potential portal disks. We define $\sim'$ as the *transitive closure* of all $\sim^i$: $p \sim' p'$ if and only if there exists a sequence of squares $i_1, \ldots, i_s \in \{1, 2, 3, 4\}$ and a sequence of portals $p = p_0, \ldots, p_s = p'$ such that for all $1 \leqslant j \leqslant s$, p_j is a common portal of $P^{i_{j-1}}$ and P^{i_j}. Further, p_{j-1} and p_j must be connected in $\mathcal{C}^{i_j}$. We call $\mathcal{C}$ *empty* if $P = \emptyset$ and $t = 0$, and *closed* if $P = \emptyset$ and $t > 0$.

We now define the notion of compatibility of configurations.

▶ **Definition 19.** *Five configurations* $\left(\mathcal{C}, \mathcal{C}^1, \mathcal{C}^2, \mathcal{C}^3, \mathcal{C}^4\right)$ *with* $\mathcal{C} = [C, t, t_{add}, P, \sim]$ *and* $\mathcal{C}^i = [C^i, t^i, t^i_{add}, P^i, \sim^i]$ *are* compatible *if all the following properties are satisfied.*

1. *all* C^i *have the same level and their union is the square* C*.*
2. $P = \bigcup_{i=1}^4 P^i \cap \partial C$*.*
3. $\sim$ *is the restriction of the transitive closure* $\sim'$ *of* $\left(\sim^i\right)_{1 \leqslant i \leqslant 4}$ *to* P*.*
4. $t = t^1 + t^2 + t^3 + t^4$ *and* $t \leqslant k$*.*
5. $t_{add} = t^1_{add} + t^2_{add} + t^3_{add} + t^4_{add} + |\bigcup_{i=1}^4 P^i \setminus P|$ *and* $t_{add} \leqslant \varepsilon k$*.*
6. *exactly one of following three conditions holds.*
   (a) $\mathcal{C}^i$*,* $i \in \{1, 2, 3, 4\}$*, is closed and all* $\mathcal{C}^j$*,* $j \neq i$ *are empty.*
   (b) $\mathcal{C}$ *is closed and there is exactly one equivalence class for* $\sim'$*.*
   (c) *all equivalence classes of* $\sim'$ *contain a portal in* P*.*

▶ Remark. By condition 2, the set P of portals used by C is obtained by removing from $\bigcup_{i=1}^4 P^i$ the portals not on the border of C. Notice that these removed portals in $\bigcup_{i=1}^4 P^i \setminus P$ are now counted as additional disks (in condition 5). Condition 6 attempts to capture all possible situations – either we have a single connected component not connected to the "outside world", which is a feasible solution by itself, (see Condition (6a) and Condition (6b)), or we have several connected components, each of which must be further connected to the outside world in a later stage (see Condition (6c)). See Figure 5. Finally, it is easy to see that if all $\sim^i$ satisfy the connectivity relation, then so does $\sim$.

Let a be a shift chosen uniformly at random in $\left\{0, \frac{L}{2}\right\}$. We consider the grid associated to this shift and the set of additional disks on this grid as defined in the previous section. The following lemma is essential to our main theorem. Recall that $\mathcal{P}$ denotes the set of portal disks and $\mathcal{F}$ the set of fence disks.

▶ **Lemma 20** (Structural Lemma). *Given a fixed parameter $\varepsilon > 0$, there exists a subset $S \subseteq core$ of input disks and a set $S_{add} \subseteq \mathcal{P} \cup \mathcal{F}$ of* additional disks*, such that with probability at least* $1/3$*,*

(i) *(feasibility) $|S| \leqslant k$ and $S \cup S_{add}$ is connected,*
(ii) *(bounded resource augmentation) $|S_{add}| \leqslant \varepsilon k$,*
(iii) *(near-optimality) $\mathcal{A}(S) \geqslant (1-\varepsilon)\boldsymbol{OPT}$,*
(iv) *(bounded local size) For each leaf square C, $|C \cap S| = \mathcal{O}(\varepsilon^{-3})$.*

Our dynamic programming aims at finding a solution satisfying all conditions of this Structural Lemma. We show that such a solution can be computed in time $n^{\mathcal{O}(\varepsilon^{-3})}$. The *bounded local size* property ensures that we can try all possible configurations in the leaf squares in polynomial time. We also prove that for any square, the number of different planar connectivity relations is upper-bounded by the *Catalan number* of the number of potential portal disks of the square. It follows from Observation 17 that this number is polynomially bounded.

4.6 Proof of the structural Lemma

We construct S and S_{add} from $\mathbf{OPT}$ in two steps. In the first step, we build sets S' and S_{add} that satisfy properties *(i)-(iii)*; and in the second step, we construct $S \subseteq S'$ by removing some disks from S' so as to satisfy property *(iv)* while maintaining the validity of the three first properties.

4.6.1 Part 1: Construction of the set of additional disks

Fix any shift, consider its associated grid and dissection and the corresponding set of additional disks $X_{add} = \mathcal{P} \cup \mathcal{F}$. Let S' be the union of disks in $\mathbf{OPT}$ that are located in the core of a leaf square of the dissection, namely

$$S' = \mathbf{OPT} \cap core.$$

Observe that S' might be disconnected since we have removed from $\mathbf{OPT}$ all the disks that were intersecting the grid. Letting *border* denote $\bigcup_{C \text{ is leaf}} \partial(C)$, we show how to replace the set of input disks $\mathbf{OPT} \cap border$ by a subset $S_{add} \subseteq \mathcal{F} \cup \mathcal{P}$ of additional disks.

Each leaf square $[a,b] \times [c,d]$ has an associated fence that is the boundary of the square $[a+2, b-2] \times [c+2, d-2]$. For each vertical (*resp.* horizontal) portal disk (x,y), we define a *connection line*, which is $\{x\} \times [y-2, y+2]$ (*resp.* $[x-2, x+2] \times \{y\}$). The set of fences and connection lines naturally partition the set of points which are at distance at most 2 from the lines of the grid into a set of *rectangles* $\mathcal{R}$. See Figure 6. Notice that all connections and fences are covered by the union of additional disks. Given a rectangle $R \in \mathcal{R}$, we define $\text{disk}(R) \subseteq X_{add}$ as the minimal set of additional disks that contain R.

We construct S_{add} as the union of $\text{disk}(R)$, over all rectangles R that intersect a disk $x \in \mathbf{OPT} \cap border$.

$$S_{add} = \bigcup \{\text{disk}(R) \colon R \in \mathcal{R}, \exists x \in \mathbf{OPT} \cap border \text{ such that } B_x \cap R \neq \emptyset\}$$

Notice that each disk $x \in \mathbf{OPT} \cap border$ intersects at most two rectangles. Furthermore, such a disk does not intersect with any fence and can intersect at most one connection line.

▷ Claim 21. Sets S' and S_{add} are such that $S' \cup S_{add}$ is connected, S' has size at most k and with probability at least $1/3$: $|S_{add}| \leqslant \mathcal{O}(\varepsilon k)$ and $\mathcal{A}(S') \geqslant (1 - \mathcal{O}(\varepsilon))\mathbf{OPT}$.

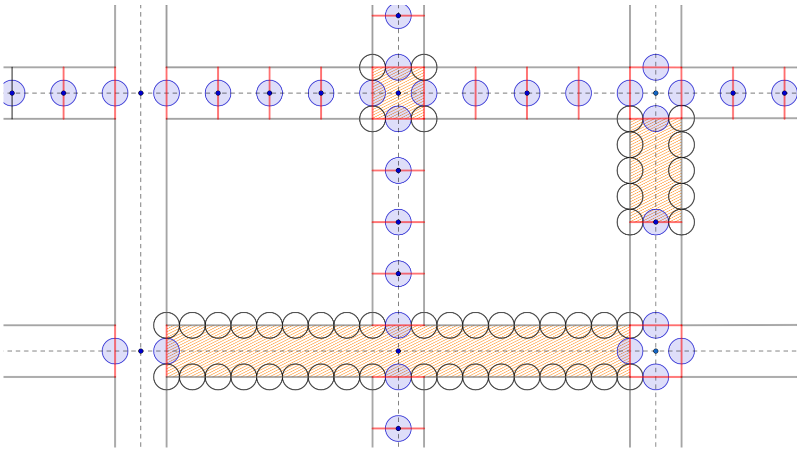


Figure 6 Dotted lines are the grid lines. The bottom and top horizontal lines have respectively level 8 and 10, and the vertical lines from left to right have level $5, 10$ and 9. Grey continuous line are the *fence*, and the red ones, the *connection lines*. Points and blue disks are *portals* and *portal disks*. Striped orange areas illustrate some rectangles $R \in \mathcal{R}$, and other disks are *fence disks* of the corresponding sets $\text{disk}(R)$.

The proof is in the appendix (the argument is similar to the one of Arora [1]). We first upper-bound the expectation of $|S_{add}|$ and $\mathcal{A}(\mathbf{OPT}) - \mathcal{A}(S')$, and then use Markov's inequality. To bound the expectation of $|S_{add}|$, we observe that the number of additional disks added in S_{add} for each disk in **OPT** intersecting a line at level ℓ is $\mathcal{O}(L/(m2^\ell))$ while the probability that a disk intersects a line at level ℓ is $\mathcal{O}(2^\ell/L)$.

4.6.2 Part 2: Sparsification of S'

The sets $S' \cup S_{add}$ obtained so far may not satisfy the last property (*bounded local size*). In this section, we show how to remove some disks from S' to guarantee this property while still maintaining the other required properties in Lemma 20.

Suppose that there exists a leaf square C such that $S'_C := S' \cap C$ has size greater than $k_0 := (1 + \beta^{-1})L_0^2 = \mathcal{O}(\varepsilon^{-3})$, where $\beta = \min\{\varepsilon/12, 1\}$. Then the core of C is "overcrowded" and we show how to construct a non-overcrowded subset maintaining connectivity while losing only an $\varepsilon/2$-th fraction of the covered area.

Define a set S to be initially equal to S'. Consider each overcrowded leaf square C one by one, and define $S_C = S \cap C$. Start with an empty set H and for each disk $x \in S_C$, add x in H if $\mathcal{A}(H \cup \{x\}) - \mathcal{A}(H) \geqslant \beta$. Define $\overline{H} = S_C \setminus H$ as the complement of H and then apply Claim 22 to $G = UDG(S \cup S_{add})$ and $D = S \cup S_{add} \setminus \overline{H}$ to define $D' \subseteq \overline{H}$. Finally update S to $(S \setminus \overline{H}) \cup D'$.

▷ Claim 22. Let $G = (V, E)$ be a connected graph and D a dominating set with μ connected components. There exists a subset $D' \subseteq V \setminus D$ of size at most $2(\mu - 1)$ such that $G[D \cup D']$ is connected.

Proof. Let H and H' be two connected components in D that minimize $d_G(H, H')$. Then, $d_G(H, H') \leqslant 3$. Indeed, if $d_G(H, H') \geqslant 4$, then there exists a vertex x on a shortest path from H to H' that is not dominated by D. This implies that we can find two vertices that connect H and H'. We repeat this operation until there is only one connected component. This requires at most $2(\mu - 1)$ vertices. ◁

The following claim, together with Claim 21 ensures that sets S and S_{add} built in Part 1 and Part 2 satisfy the expected properties of our structural Lemma.

▷ Claim 23. The constructed sets S and S_{add} satisfy

(i) $S \cup S_{add}$ is connected,

(ii) for each leaf square C, $|S \cap C| \leqslant k_0$, and

(iii) $\mathcal{A}(S) \geqslant (1 - \varepsilon/2)\mathcal{A}(S')$.

This Claim might not be true if the radius of disks considered are arbitrary. The proof of this fact follows from geometrical observations about unit disks.

Algorithm 2 PTAS for MACS with resource augmentation.

Input: X, k, ε.
Output: a real number $maxi \geqslant (1 - \varepsilon)\mathbf{OPT}$.
forall $c \in X$ **do**
  let $\mathcal{B}'$ be the $4k \times 4k$ square centered at c;
  $X' \leftarrow X \cap \mathcal{B}'$;
  $L \leftarrow$ the smallest power of 2 such that $L \geqslant 8k$;
  forall *shift* $a \in \{0, \dots, L/2\}$ **do**
    Create a table *tab*;
    foreach *configuration* $\mathcal{C}$ **do**
      $tab[\mathcal{C}] \leftarrow -\infty$;
    `/* Initialization */`
    foreach $\mathcal{C}$ *at level* d *(leaf square)* **do**
      $tab[\mathcal{C}] \leftarrow \max\{\mathcal{A}(S) : (S, S_f, P)$ is usable and corresponds to $\mathcal{C}\}$;
    `/* Fusion */`
    foreach *level* $0 \leqslant i \leqslant d - 1$ *in decreasing order* **do**
      foreach *configuration* $\mathcal{C}$ *at level* i **do**
        $tab[\mathcal{C}] \leftarrow \max \left\{ \sum_{i=1}^{4} tab[\mathcal{C}^i] : (\mathcal{C}, \mathcal{C}^1, \mathcal{C}^2, \mathcal{C}^3, \mathcal{C}^4) \text{ are compatible} \right\}$;
return $maxi = \max_{\substack{\text{configuration } \mathcal{C} \\ \text{for root square}}} tab\,[\mathcal{C}]$;

Notice that since the root square has no potential portals (portals are only placed on lines at level at least 1), any configuration that corresponds to the root square has only one connected component. We can easily add information in the table so that the algorithm also outputs the corresponding sets S and S_{add}.

Notice that Algorithm 2 tries all possible shift a. Our structural Lemma 20 ensures that there exists at least one shift such that the output satisfies all expected properties of Theorem 4.

▶ **Theorem 24.** *Algorithm 2 has a running time $n^{\mathcal{O}(\varepsilon^{-3})}$.*

The key ingredient in order to prove that our algorithm is polynomial follows from Observation 17. We show that the number of connectivity relations of a set of $\mathcal{O}(m)$ portals corresponds to its *Catalan number* which is polynomial when $m = \mathcal{O}(\varepsilon^{-1} \log(\varepsilon k))$.

Algorithm 3 PTAS for MACS for well-distributed inputs.

Input: X an α-well-distributed input, $k \geqslant 0$, $\varepsilon > 0$.
Output: A feasible solution to MACS(X, k).

1. Choose $\varepsilon' > 0$ and $k' \leqslant k$ such that $(1-\varepsilon')(1-10(22\alpha+4)\varepsilon') \geqslant (1-\varepsilon)$ and $k'(1+(22\alpha+4)\varepsilon') = k$;
2. Let S, S_{add} be the solution of Algorithm 2 on input (X, k', ε') ;
3. Let S' be the set obtained from S_{add} by Lemma 25;
4. **return** $S \cup S'$;

5 A PTAS for well-distributed inputs

Let us recall the definition of well-distributed input.

▶ **Definition 5.** *Given $\alpha > 0$, a finite set X of points in the plane is called* α-well-distributed *if for all $x, y \in X$, $d_{UDG(X)}(x, y) \leqslant \lceil \alpha \cdot \|x-y\| \rceil$.*

Here $\lceil \cdot \rceil$ is the ceiling function. This ensures that the right-hand side is always at least one. Notice that a well-distributed set is necessarily connected.

One intuitive view of a well-distributed input is to look at the shape of the "holes" of the input, that are the different connected components of the complement of the union of the input disks in the plane. The assumption of well-distribution means that these holes are roughly *fat*.

One particular interesting case arises when there is no hole at all. We call these sets *pseudo-convex*, and we prove that this is a particular case of well-distributed inputs.

▶ **Definition 7.** *A set X is called* pseudo-convex *if the convex-hull of X is covered by the union of the unit disks centered at points of X.*

▶ **Lemma 8.** *A pseudo-convex set X is* 3.82*-well-distributed.*

Our Corollary 6 states that the restriction of MACS to well-distributed inputs admits a PTAS. The algorithm works as follows. Given a parameter $0 < \varepsilon \leqslant 1/2$, and an input (X, k) of MACS, we run Algorithm 2 on input (X, k', ε') for suitable values k' and ε' specified below. Next, we transform the set of additional disks obtained into a set of input disks that has roughly the same size while maintaining the connectivity of the solution. See Lemma 25 and Algorithm 3 for details. This algorithm naturally applies to pseudo-convex inputs (Corollary 9).

▶ **Lemma 25.** *Given an α-well-distributed input X and two finite sets $S \subseteq X$ and $S_{add} \subseteq \mathbb{R}^2$ such that $UDG(S \cup S_{add})$ is connected, there exists a set $S' \subseteq X$ of size at most $(22\alpha+4)|S_{add}|$ such that $UDG(S \cup S')$ is connected. Moreover, such a set can be computed in polynomial time.*

In the previous lemma, the set S_{add} is not supposed to be a set of additional disks as defined in Section 4.

Since $\varepsilon' = \Theta(\varepsilon/\alpha)$, the previous algorithm runs in polynomial time when ε and α are fixed constants.

▷ Claim 26. The solution returned by Algorithm 3 on input (X, k, ε) is a feasible solution to MACS(X, k) and covers an area at least $(1-\varepsilon)\mathbf{OPT}(X, k)$.

In order to prove this result we need to state the following "stability" property over optimal solutions.

▶ **Lemma 27.** *Let $\eta < \frac{1}{2}$. Then $\boldsymbol{OPT}(X, k) \geqslant (1-10\eta) \cdot \boldsymbol{OPT}(X, k(1+\eta))$.*

References

1 Sanjeev Arora. Polynomial Time Approximation Schemes for Euclidean Traveling Salesman and Other Geometric Problems. *J. ACM*, 45(5):753–782, September 1998. doi:10.1145/290179.290180.

2 Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a Monotone Submodular Function Subject to a Matroid Constraint. *SIAM J. Comput.*, 40(6):1740–1766, 2011.

3 Erin W. Chambers, Sándor P. Fekete, Hella-Franziska Hoffmann, Dimitri Marinakis, Joseph S.B. Mitchell, Venkatesh Srinivasan, Ulrike Stege, and Sue Whitesides. Connecting a set of circles with minimum sum of radii. *Computational Geometry*, 68(1-3):62–76, January 1991. special issue in memory of Ferran Hurtado.

4 Steven Chaplick, Minati De, Alexander Ravsky, and Joachim Spoerhase. Approximation Schemes for Geometric Coverage Problems. In *ESA*, volume 112 of *LIPIcs*, pages 17:1–17:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

5 Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. Submodular Function Maximization via the Multilinear Relaxation and Contention Resolution Schemes. *SIAM J. Comput.*, 43(6):1831–1879, 2014.

6 Yuval Filmus and Justin Ward. Monotone Submodular Maximization over a Matroid via Non-Oblivious Local Search. *SIAM J. Comput.*, 43(2):514–542, 2014. doi:10.1137/130920277.

7 Stefan Funke, Alex Kesselman, Fabian Kuhn, Zvi Lotker, and Michael Segal. Improved Approximation Algorithms for Connected Sensor Cover. *Wirel. Netw.*, 13(2):153–164, April 2007. doi:10.1007/s11276-006-3724-9.

8 L. A. Wolsey G.L. Nemhauser and M.L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 14(1):265–294, 1978.

9 Himanshu Gupta, Zongheng Zhou, Samir R. Das, and Quinyi Gu. Connected Sensor Cover: Self-organization of Sensor Networks for Efficient Query Execution. *IEEE/ACM Trans. Netw.*, 14(1):55–67, February 2006. doi:10.1109/TNET.2005.863478.

10 Dorit S. Hochbaum and Anu Pathria. Node-Optimal Connected k-Subgraphs, 1994.

11 Koushik Kar and Suman Banerjee. Node Placement for Connected Coverage in Sensor Networks, 2003.

12 Samir Khuller, Manish Purohit, and Kanthi K. Sarpatwar. Analyzing the Optimal Neighborhood: Algorithms for Budgeted and Partial Connected Dominating Set Problems. In *Proceedings of the Twenty-fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 1702–1713, Philadelphia, PA, USA, 2014. Society for Industrial and Applied Mathematics. URL: http://dl.acm.org/citation.cfm?id=2634074.2634197.

13 Ariel Kulik, Hadas Shachnai, and Tami Tamir. Maximizing submodular set functions subject to multiple linear constraints. In *SODA*, pages 545–554. SIAM, 2009.

14 Tung-Wei Kuo, Kate Ching-Ju Lin, and Ming-Jer Tsai. Maximizing Submodular Set Function with Connectivity Constraint: Theory and Application to Networks. *IEEE/ACM Trans. Netw.*, 23(2):533–546, April 2015. doi:10.1109/TNET.2014.2301816.

15 Jon Lee, Maxim Sviridenko, and Jan Vondrák. Submodular Maximization over Multiple Matroids via Generalized Exchange Properties. *Math. Oper. Res.*, 35(4):795–806, 2010.

16 M. V. Marathe, H. Breu, H. B. Hunt III, S. S. Ravi, and D. J. Rosenkrantz. Simple heuristics for unit disk graphs. *NETWORKS*, 1995.

17 Tomomi Matsui. Approximation Algorithms for Maximum Independent Set Problems and Fractional Coloring Problems on Unit Disk Graphs. In Jin Akiyama, Mikio Kano, and Masatsugu Urabe, editors, *Discrete and Computational Geometry*, pages 194–200, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

18 Nabil H. Mustafa and Saurabh Ray. Improved Results on Geometric Hitting Set Problems. *Discrete & Computational Geometry*, 44(4):883–895, 2010.

19 Giri Narasimhan and Michiel Smid. *Geometric Spanner Networks*. Cambridge University Press, New York, NY, USA, 2007.

20 Yuval Rabani and Gabriel Scalosub. Bicriteria approximation tradeoff for the node-cost budget problem. *ACM Trans. Algorithms*, 5(2):19:1–19:14, 2009.
21 Fabio Vandin, Eli Upfal, and Benjamin J. Raphael. Algorithms for Detecting Significantly Mutated Pathways in Cancer. *Journal of Computational Biology*, 18(3):507–522, 2011.
22 Vijay V. Vazirani. *Euclidean TSP*, pages 84–89. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. doi:10.1007/978-3-662-04565-7_11.
23 L. Wolsey. Maximising real-valued submodular functions: primal and dual heuristics for location problems. *Mathematics of Operations Research*, 7(3):410–425, 1982.

A Omitted proofs

Proof of Claim 21. Clearly $|S'| \leqslant |\mathbf{OPT}| \leqslant k$. We now prove that $S' \cup S_{add}$ is connected. Suppose that there exists a disk $x \in \mathbf{OPT} \cap border$ such that $\mathbf{OPT} \setminus \{x\}$ is split into several connected components. We know that x intersects only one rectangle $R_1 \in \mathcal{R}$ or two rectangles $R_1, R_2 \in \mathcal{R}$. Since $\mathbf{OPT}$ is connected, and B_x is contained in the set $U = R_1$ or $U = R_1 \cup R_2$, each connected component intersects the boundary of U. Then, B_x intersects a disk in $\text{disk}(R_1)$ or $\text{disk}(R_2)$. Therefore, $\mathbf{OPT} \setminus \{x\} \cup \text{disk}(R_1) \cup \text{disk}(R_2)$ is connected. By doing so for each $x \in \mathbf{OPT} \cap border$, it follows that $S' \cup S_{add}$ is connected.

It remains to show that, under a uniform random shift a, with probability at least one third we have $|S_{add}| \leqslant \mathcal{O}(\varepsilon k)$ **and** $\mathcal{A}(S') = \mathcal{A}(\mathbf{OPT} \cap core) \geqslant (1 - \mathcal{O}(\varepsilon))\mathbf{OPT}$. The proof is very similar to Arora's approach, we first upper-bound the expectation of $|S_{add}|$ and $\mathcal{A}(\mathbf{OPT}) - \mathcal{A}(S')$, and then use *Markov inequality* to conclude.

We first upper-bound the expected number of additional disks. For each $x \in \mathbf{OPT}$ intersecting a line at level ℓ, we have added at most two sets of additional disks associated to rectangles with side length smaller than the distance between two consecutive portals of this line. It follows that $\mathcal{O}(L/(m2^\ell))$ additional disks have been added to S_{add} for each disk in $\mathbf{OPT}$ intersecting a line of level ℓ. This can be observed in Figure 7. Moreover, the probability that a disk intersects a line at level ℓ is $\mathcal{O}(2^\ell/L)$. Then,

$$\begin{aligned}\mathbb{E}(|S_{add}|) &\leqslant \sum_{x \in \mathbf{OPT}} \sum_{\ell=0}^{d-1} P(x \text{ intersects exactly one line at level } \ell)\mathcal{O}(\frac{L}{m2^\ell}) \\ &= \sum_{x \in \mathbf{OPT}} \sum_{\ell=0}^{d-1} \mathcal{O}(\frac{2^\ell}{L} \cdot \frac{L}{m2^\ell}) = \mathcal{O}(\frac{dk}{m}) = \mathcal{O}(\varepsilon k)\end{aligned}$$

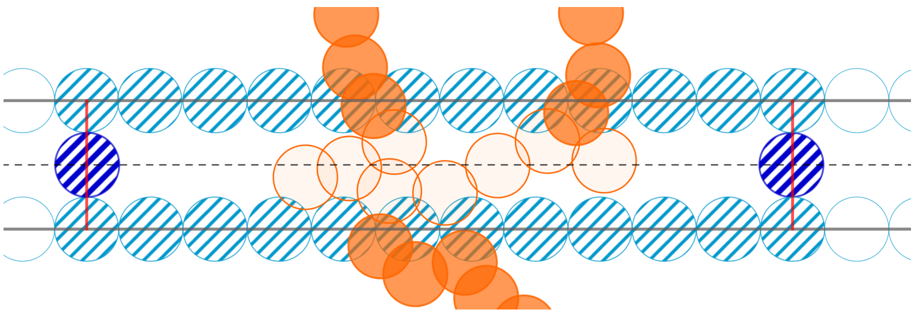


Figure 7 $\mathbf{OPT}$ is represented by orange disks. Disks of $\mathbf{OPT}$ that intersect the grid (dotted line) are replaced by additional disks (striped blue disks). This operation maintains the connectivity of the set.

We now upper-bound the expectation of $\mathcal{A}(\mathbf{OPT}) - \mathcal{A}(S')$. First we have $\mathcal{A}(\mathbf{OPT}) - \mathcal{A}(S') \leqslant \mathcal{A}(\mathbf{OPT} \cap border)$, and the probability that a point $p \in B(\mathbf{OPT})$ is in $B(\mathbf{OPT} \cap border)$ is smaller that p is at distance 2 from the lines of the grid. Therefore

$$\begin{aligned}\mathbb{E}\left(\mathcal{A}(\mathbf{OPT}) - \mathcal{A}(S')\right) &\leqslant \mathbb{E}(\mathcal{A}(\mathbf{OPT} \cap border)) \\ &\leqslant \int_{p\in B(\mathbf{OPT})} P(p \text{ is at distance at most 4 from the grid})\mathrm{d}p \\ &\leqslant \int_{p\in B(\mathbf{OPT})} 2 \cdot \frac{4}{L_0}\mathrm{d}p \\ &\leqslant \frac{8 \cdot \mathbf{OPT}}{L_0} = \mathcal{O}(\varepsilon \mathbf{OPT})\end{aligned}$$

By choosing the constant properly in the big O notation and using the Markov inequality, we can show that the probability of $|S_{add}| > O(\varepsilon k)$ and the probability of $A(OPT) - A(S) > O(\varepsilon OPT)$ are both upper bounded by $\frac{1}{3}$. Thus, by a union bound, we conclude the proof. $\lhd$

Proof of Claim 23. For (i), we just need to argue that for each leaf square C, after $\overline{H}$ is defined, $S \cup S_{add} \setminus \overline{H}$ is a dominating set in $UDG(S \cup S_{add})$ (then the proof follows from Claim 22). Indeed if a disk x is in $\overline{H}$ then it means that $\mathcal{A}(H \cup \{x\}) - \mathcal{A}(H) < \beta \leqslant 1$. In particular, it implies that there exists a disk in $H \subseteq S \cup S_{add} \setminus \overline{H}$ that intersects x.

For (ii), observe that the size of $S \cap C$ is the sum of the size of the corresponding sets H and D' built during the "sparsification" of C. Since all disks in H increases the area covered by at least β and are contained in a square of area L_0^2, the number of disks in H is upper-bounded by $\beta^{-1}L_0^2$. Moreover, each connected component of $S \cup S_{add} \setminus \overline{H}$ had a disk contained in C so that the number μ of connected component is upper-bounded by $L_0^2/\pi < L_0^2/2$. Therefore $|D'| < L_0^2$. Finally $|H \cup D'| < (1 + \beta^{-1})L_0^2 = k_0$.

For (iii), we start by observing that the union $B(S')$ of disks in S' is contained in the set $B^+(S)$, which is defined as

$$B^+(S) := \{z \in \mathbb{R}^2 \mid \exists x \in S \text{ such that } ||z - x|| \leqslant 1 + \beta\}$$

Indeed, if there exists a point p covered by a disk x in S' but at distance at least $1 + \beta$ from any disk of S then adding x to S would increase the area covered by S by more that β.

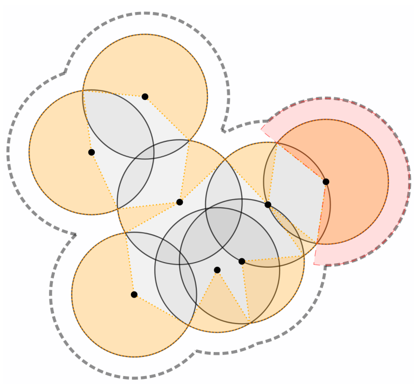


Figure 8 S consists of grey disks. The boundary of $B^+(S)$ is the dotted curve. Circular sectors are in orange while the red one represents a circular sector in $B^+(S)$.

Therefore, we have the following inclusion

$$B(S) \subseteq B(S') \subseteq B^+(S), \tag{1}$$

and if the following geometrical claim holds, our proof of (iii) will be complete.

▷ Claim 28. $\mathcal{A}(B(S)) \geqslant (1-\varepsilon/2)\mathcal{A}(B^+(S))$

The result follows from the fact that $B(S)$ is a union of unit-disks. See Figure 8. The boundary of $B(S)$ is made of *circular arcs* and each of these arcs is associated with a *circular sector* θ_i. Circular sectors intersect with other circular sectors only on the extreme points of their corresponding arcs, thus $\mathcal{A}(\cup_i \theta_i) = \sum_i \mathcal{A}(\theta_i)$.

We can associate with each circular sector θ_i (of a disk of radius 1) its "dilation" θ_i^+ which corresponds to the same circular sector in a disk of radius $1+\beta$. We have $\mathcal{A}(\theta_i^+) = (1+\beta)^2\mathcal{A}(\theta_i)$ and can see that $B^+(S) \setminus B(S) \subseteq \bigcup_i(\theta_i^+ \setminus \theta_i)$. Then

$$\begin{aligned}
\mathcal{A}(B^+(S)) - \mathcal{A}(B(S)) = \mathcal{A}(B^+(S) \setminus B(S)) &= \mathcal{A}\left(\bigcup_i (\theta_i^+ \setminus \theta_i)\right) \\
&\leqslant \sum_i \mathcal{A}(\theta_i^+ \setminus \theta_i) = \sum_i \mathcal{A}(\theta_i^+) - \mathcal{A}(\theta_i) \\
&\leqslant \sum_i (1+\beta)^2 \mathcal{A}(\theta_i) - \mathcal{A}(\theta_i) \\
&\leqslant \sum_i 3\beta\mathcal{A}(\theta_i) = 3\beta\mathcal{A}\left(\bigcup_i \theta_i\right) \leqslant 3\beta\mathcal{A}(B(S))
\end{aligned}$$

Therefore, $\mathcal{A}(B(S)) \geqslant \dfrac{\mathcal{A}(B^+(S))}{1+3\beta} \geqslant (1-\varepsilon/2)\mathcal{A}(B^+(S))$. This concludes the proofs of Claims 28 and 23. ◁

Proof of Theorem 24.

Size of tab. There exists 4^i squares at level i so the total of squares is $\sum_{i=0}^d 4^i = \mathcal{O}(4^{d+1})$. For any square C, the number of potential portal disks is at most $4m$. To see this, observe that if C is of level i, it is of size $L/2^i \times L/2^i$. Furthermore, it is surrounded by lines of level at most i and two adjacent portals on such a line has distance $\Omega(L/(m2^i))$.
Therefore, the number of possible sets $P \subseteq \mathcal{P}_C$ is 2^{4m}, and for each set P of size r the total number of planar connectivity relations is equal to the r-th *Catalan number* : $P(r) = \dfrac{1}{r-1}\dbinom{2r}{r} = \mathcal{O}\left(\dfrac{1}{m-1}\dbinom{8m}{m}\right)$ and then by Stirling formula we get $P(r) = \mathcal{O}(4^{4m})$. To see that $P(r)$ is the r-th Catalan number, we check that it satisfies the same recurrence relation :

$$P(r) = \sum_{k=1}^{r} P(k-1) \cdot P(r-k) \tag{2}$$

with $P(0) = 1$. Indeed, if k denotes the index of the first portal p_k that is on the connected component of the r-th portal disk p_r, then the portal disk p_i with $1 \leqslant i \leqslant k-1$ cannot be equivalent to a portal p_j disk with $k \leqslant j \leqslant n$, and then the equivalence relation can be restricted to the set $\{p_i, 1 \leqslant i \leqslant k-1\}$ and there are $P(k-1)$ possible distinct choices. Next observe that since p_n and p_k are connected (i.e. $p_n \sim p_k$), it is enough to count the number of different equivalence relations in $\{p_j, k+1 \leqslant j \leqslant r\}$, which is $P(r-k)$.

Finally, observe that k can be from 1 to r ($k = r$ means that p_r is alone in its connected component.) We thus concludes (2). Therefore, creating *tab* in line 6 can be done in time $\mathcal{O}(4^{d+1}\varepsilon k^2 8^{4m}) = k^{\mathcal{O}(1/\varepsilon)}$.

Initialization. There exists 4^d leaf squares and for each of them, we try all possible guesses. This can be done in time $n^{\mathcal{O}(\varepsilon^{-3})}$.

Fusion. Trying all possible combinations can be done in time $k^{\mathcal{O}(1/\varepsilon)}$ ◀

Proof of Lemma 27. Let X be a set of points of the plane, k a positive integer and $\eta \leqslant 1/2$ a parameter. We prove a stronger result. Given any solution feasible solution S to $\mathrm{MACS}(X, k(1+\eta))$, there exists a subset S' of S that is a feasible solution to $\mathrm{MACS}(X, k)$ with value at least $(1 - 10\eta)\mathcal{A}(S)$. Obviously Lemma 27 follows when S is optimal. If $\mathcal{A}(S) \geqslant k/3$, then remove ηk disks from S without disconnecting S. For instance, consider a spanning tree on $UDG(S)$ and remove the nodes from the leaves to the root until you reach the desired size. Let S' denote the subset obtained.

$$\mathcal{A}(S') \geqslant \mathcal{A}(S) - \eta\pi k \geqslant (1 - 3\pi\eta)\mathcal{A}(S) \geqslant (1 - 10\eta)\mathcal{A}(S)$$

If $\mathcal{A}(S) < k/3$, let I be a maximal independent set in S. We have $|I|\pi = \mathcal{A}(I) \leqslant \mathcal{A}(S) < \dfrac{k}{3}$. According to claim 22, there exists a connected dominating set $I \subseteq D \subseteq S$ in S of size at most $3|I| - 2 < k/\pi < \dfrac{k}{3}$. Consider a set $H \subseteq S \setminus D$ of size $k - |D| > \frac{2}{3}k$ built by greedily adding a disk $h \in S \setminus (D \cup H)$ maximising the marginal area $\mathcal{A}(D \cup H \cup \{h\}) - \mathcal{A}(D \cup H)$. Since D is a connected dominating set, the set $S' := D \cup H$ is connected. Since all disk where added greedily in H, for all $H \in S \setminus S'$, we have

$$\mathcal{A}(S' \cup \{h\}) - \mathcal{A}(S')) \leqslant \frac{\mathcal{A}(S) - \mathcal{A}(D)}{|H|} \leqslant \frac{2\mathcal{A}(S)}{k}$$

.

By submodularity, we deduce that $\mathcal{A}(S) - \mathcal{A}(S') \leqslant \eta k \cdot \dfrac{2\mathcal{A}(S)}{3k}$. That implies $\mathcal{A}(S') \geqslant (1 - \frac{3}{2}\eta)\mathcal{A}(S)$. This concludes the proof of lemma 27. ◀

Remark that this proof is constructive and it is easy to check that finding S' from any given set S can be done in polynomial time.

Proof of Lemma 25. Let us use the same notation as in the statement of Lemma 25. We prove how to build S' from S_{add} such that $|S'| \leqslant (22\alpha+4)|S_{add}|$ while preserving connectivity.

Let Y be a connected component of S_{add}. We prove that we can find a set $Y' \subseteq X$ of input disks such that $|Y'| \leqslant (4 + 22\alpha)|Y|$ and $(S_{add} \setminus Y) \cup (S \cup Y')$ is connected. Removing Y might split the solution into several connected components $F_1, \dots, F_s$. For each connected component F_i, pick one disk x_i in $F_i \cap X$ that intersects Y.

Step 1. Each additional disk y in Y is adjacent to at most 6 disks x_i. We can connect the corresponding connected component by using 20α disks of the input. Indeed, any two x_i and x_j adjacent to y has a Euclidean distance at most 4. Since X is well-distributed their distance in $UDG(X)$ is at most 4α. Then, we can find $\lceil 4\alpha - 1 \rceil$ disks in X which connect x_i and x_j. In order to connect all the x_i that are adjacent to y, it is sufficient to repeat this operation 5 times, which asks at most 20α disks. We can perform this operation for each additional disk that was not already considered. Then, in total for this first step we need to use at most $20\alpha|Y|$ disks.

Step 2. During step 1, we may have connected some disks x_i, so that the number of connected components has decreased. The number of connected components is $s' \leqslant s$, each of them corresponds to a disk x_i, and without loss of generality we can assume that the corresponding indexes are such that $1 \leqslant i \leqslant s'$. Let T be a spanning tree on $UDG(Y)$. Without loss of generality, we can suppose that indexes i are such that the sequence $(x_1, \dots, x_{s'})$ correspond to a T transversal. Note that after step 1, each x_i can be associated to a different y in Y. Then, we reconnect each x_i to x_{i+1} for $1 \leqslant i \leqslant s-1$. If x_i and x_{i+1} are respectively associated to y_i and y_{i+1}, then $||x_i - x_{i+1}|| \leqslant 2 + 2d_T(y_i, y_{i+1})$ and thus $d_{UDG(X)}(x_i, x_{i+1}) \leqslant \lceil \alpha(2 + 2d_T(y_i, y_{i+1}) \rceil$. Then, we can find $\lceil \alpha(2 + 2d_T(y_i, y_{i+1}) \rceil - 1$ disks in X to connect x_i and x_{i+1}. In order to connect all x_i we need to use at most

$$\sum_{i=1}^{s'-1} \lceil \alpha(2 + 2d_T(y_i, y_{i+1}) \rceil - 1 \leqslant 2(s'-1)\alpha + 2\sum_{i=1}^{s'-1} d_T(y_i, y_{i+1})$$

input disks. Since the order corresponds to a T transversal, each edge is visited at most twice and then $\sum_{i=1}^{s'-1} d_T(y_i, y_{i+1}) \leqslant 2(|Y| - 1)$. Therefore the total number of disks that were added during this second step is bounded by $|Y|(4 + 2\alpha)$.

We proved that there exists a subset $Y' \subseteq X$ of size at most $(4 + 22\alpha)|Y|$ such that $(S \cup S_{add} \setminus Y) \cup Y'$ is connected. By doing so for each connected component of S_{add}, we get the result claimed. ◀

Proof of Lemma 8. Let X be a pseudo-convex set, G its unit-disk-graph, and x and y be any two disks in X at distance $L = \|x - y\|$. We show that $d_G(x, y) \leqslant \lceil \alpha L \rceil$ where $\alpha = 12/\pi < 3.82$.

If $L < 2$ then the two unit disks associated to x and y overlap so that $d_G(x, y) = 1 \leqslant \lceil \alpha L \rceil$. Otherwise suppose that $L \geqslant 2$. Since X is pseudo-convex, it is connected and any point in the line segment $[x, y]$ is covered by a disk in X. Let $S = \{z \in X \mid B_z \cap [x, y] \neq \emptyset, \|x - z\| > 2 \text{ and } \|y - z\| > 2\}$ and let I be any maximal independent set in $S \cup \{x, y\}$. Since S is at distance at least 2 from x and y, we deduce that $x, y \in I$ and all disks in $I \setminus \{x, y\}$ are inside a $L \times 4$ rectangle and then $|I| \leqslant 4L/\pi$. Since I is maximal, it is a dominating set in S. Therefore, claim 22 implies that there exists a connected subset $D \subseteq X$ such that $I \subseteq D$ and $|D| \leqslant 3|I| - 2 \leqslant 12L/\pi - 2$. We deduce that $d_G(x, y) \leqslant (12L/\pi - 2) + 1 \leqslant \lceil \alpha L \rceil$. ◀

Proof of Claim 26. The solution output by Algorithm 2 on input (X, k', ε') verifies the following properties: $S \cup S_{add}$ is connected, the size of S and S_{add} are respectively upper-bounded by k' and $\varepsilon' k'$ and $\mathcal{A}(S) \geqslant (1 - \varepsilon')\mathbf{OPT}(X, k')$. Therefore, the set S' given by Lemma 25 has size at most $(22\alpha + 4)|S_{add}| \leqslant (22\alpha + 4)\varepsilon' k'$, and then $|S \cup S'| \leqslant k' + (22\alpha + 4)\varepsilon' k' \leqslant (1 + (22\alpha + 4)\varepsilon')k' = k$. Since $S \cup S'$ is connected, this set is a feasible solution to MACS(X, k).

Finally, from Lemma 27 with parameter $\eta = (22\alpha + 4)\varepsilon'$, we get that the area covered by this solution is

$$\begin{aligned}
\mathcal{A}(S \cup S') \geqslant \mathcal{A}(S) &\geqslant (1 - \varepsilon')\mathbf{OPT}(X, k') \geqslant (1 - \varepsilon')(1 - 10\eta)\mathbf{OPT}(X, k'(1 + \eta)) \\
&\geqslant (1 - \varepsilon')(1 - 10(22\alpha + 4)\varepsilon')\mathbf{OPT}(X, k'(1 + (22\alpha + 4)\varepsilon')) \\
&\geqslant (1 - \varepsilon)\mathbf{OPT}(X, k)
\end{aligned}$$

which concludes the proof. ◁