The Maximum Exposure Problem

Neeraj Kumar
Department of Computer Science, University of California, Santa Barbara, USA
neeraj@cs.ucsb.edu

Stavros Sintos
Duke University, Durham, NC, USA
ssintos@cs.duke.edu

Subhash Suri
Department of Computer Science, University of California, Santa Barbara, USA
suri@cs.ucsb.edu

—— **Abstract** ——
Given a set of points P and axis-aligned rectangles $\mathcal{R}$ in the plane, a point $p \in P$ is called *exposed* if it lies outside all rectangles in $\mathcal{R}$. In the *max-exposure problem*, given an integer parameter k, we want to delete k rectangles from $\mathcal{R}$ so as to maximize the number of exposed points. We show that the problem is NP-hard and assuming plausible complexity conjectures is also hard to approximate even when rectangles in $\mathcal{R}$ are translates of two fixed rectangles. However, if $\mathcal{R}$ only consists of translates of a single rectangle, we present a polynomial-time approximation scheme. For general rectangle range space, we present a simple $O(k)$ bicriteria approximation algorithm; that is by deleting $O(k^2)$ rectangles, we can expose at least $\Omega(1/k)$ of the optimal number of points.

1 Introduction

Let $S = (P, \mathcal{R})$ be a geometric set system, also called a *range space*, where P is a set of points and each $R \in \mathcal{R}$ is a collection of subsets of P, also called a range. We are primarily interested in range spaces defined by a set of points in two dimensions and ranges defined by axis-aligned rectangles. We say that a point $p \in P$ is *exposed* if no range in $\mathcal{R}$ contains p. The *max-exposure* problem is defined as follows: given a range space $(P, \mathcal{R})$ and an integer parameter $k \geq 1$, remove k ranges from $\mathcal{R}$ so that a maximum number of points are exposed. That is, we want to find a subfamily $\mathcal{R}^* \subseteq \mathcal{R}$ with $|\mathcal{R}^*| = k$, so that the number of exposed points in the (reduced) range space $(P, \mathcal{R} \setminus \mathcal{R}^*)$ is maximized.

The max-exposure problem arises naturally in many geometric coverage settings. For instance, if points are the location of clients, and ranges are coverage of some facilities in the plane, then exposed points are those not covered by any facility. The max-exposure problem in this case gives a worst-case bound on the number of clients that can be exposed if an adversary disables k facilities. Similarly, in distributed sensor networks, ranges correspond to *sensing zones*, points correspond to physical assets being monitored by the network, and the max-exposure problem computes the number of assets exposed when k sensors are compromised.

More broadly, the max-exposure problem is related to the densest k-subgraph problem in *hypergraphs*. In the *densest k-subhypergraph* problem, we are given a hypergraph $H = (X, E)$, and we want to find a set of k vertices with a maximum number of induced hyperedges. In general hypergraphs, finding k-densest subgraphs is known to be (conditionally) hard to approximate within a factor of $n^{1-\epsilon}$, where n is the number of vertices. The max-exposure problem is equivalent to the densest k-subhypergraph problem on a *dual* hypergraph, the vertex set X corresponds to the ranges $\mathcal{R}$, and set of edges E of the dual hypergraph correspond to the set of points P. In the rest of the paper, we will use $n = |\mathcal{R}|$ for the number of ranges in $\mathcal{R}$ and $m = |P|$ to be the number of points. We show that if the range space is defined by *convex polygons*, then the max-exposure problem is just as hard as the densest k-subhypergraph problem. However, for ranges defined by *axis-aligned rectangles*, one can achieve much better approximation. In particular, we obtain the following results.

- We show that the max-exposure problem is NP-hard and assuming the *dense vs random* conjecture to be true, it is also hard to approximate better than a factor of $O(n^{1/4})$ even if the range space is defined by *only two types of rectangles* in the plane. (For range space defined by convex polygons, we show that max-exposure is equivalent to densest k-subhypergraph problem, which is hard to approximate within $O(n^{1-\epsilon})$).
- When ranges are defined by translates of a *single* rectangle, we give a polynomial-time approximation scheme (PTAS) for max-exposure. The PTAS stands in sharp contrast to the inapproximability of ranges defined by *two* types of rectangles. Moreover, as an easy consequence of this result, we obtain a constant approximation when the ratio of longest and smallest side of rectangles in $\mathcal{R}$ is bounded by a constant. However, we do not know if max-exposure with translates of a single rectangle can be solved in polynomial time or is NP-hard.
- For ranges defined by arbitrary rectangles, we present a simple greedy algorithm that achieves a bicriteria $O(k)$-approximation. No such approximation is possible for general hypergraphs. If rectangles in $\mathcal{R}$ have a bounded aspect ratio, the approximation improves to $O(\sqrt{k})$.

Related Work   Coverage and exposure problems have been widely studied in geometry and graphs. In the classical *set cover* problem, we want to select a subfamily of k sets that cover the maximum number of items (points) [14, 17]. For the set cover problem, the classical greedy algorithm achieves a factor $\log n$ approximation on the number of sets needed to cover all the items, or factor $(1 - 1/e)$ approximation on the number of items covered by using exactly k sets. Similarly, in geometry, the art gallery problems explore coverage of polygons using a minimum number of guards. Unlike coverage problems where greedy algorithms deliver reasonably good approximation, the exposure problems turn out to be much harder. Specifically, choosing k sets whose union is of *minimum size* is much harder to approximate with a conditional inapproximability of $O(n^{1-\epsilon})$ where n is the number of elements and $O(m^{1/4-\epsilon})$ where m is the number of sets [10]. This so-called *min-union* problem is essentially the densest k-subgraph problem on hypergraphs [9]. The densest k-subgraph problem for graphs has a long history [15, 3, 2, 6]. The classical coverage problems have been extensively studied for geometric set systems and significantly better approximation bounds have been achieved for them [1, 7, 20]. Several other variations such as the set multi-cover problem [8, 12] where each input point needs to be covered by more than one set have also been studied. Also closely related to max-exposure is the geometric constraint removal problem [4, 13], where given a set of ranges, the goal is to *expose* a path between two given points by deleting at most k ranges (a path is exposed if it lies in the

exterior of all ranges). Even for simple shapes such as unit disks (or unit squares) [5, 19], no PTAS is known for this problem.

The remainder of the paper is organized as follows. In Section 2, we discuss our hardness results followed by the bicriteria $O(k)$-approximation in Section 3. In Section 4, we study the case when $\mathcal{R}$ consists of translates of a fixed rectangle and describe a PTAS for it. Finally, in Section 5, we use these ideas to obtain a bicriteria $O(\sqrt{k})$-approximation when aspect ratio of rectangles in $\mathcal{R}$ is bounded by a constant.

2   Hardness of Max-Exposure

We show that max-exposure problem for geometric ranges is both NP-hard and inapproximable within a polynomial factor, under some well known hardness conjectures. In particular, we first show that the densest k-subgraph on bipartite graphs (*bipartite-DkS*) can be easily reduced to the max-exposure problem. In the *bipartite-DkS* problem, we are given a bipartite graph $G = (A, B, E)$, an integer k, and we want to compute a set of k vertices such that the induced subgraph on those k vertices has the maximum number of edges. Given an instance $G = (A, B, E)$ of bipartite-DkS, we will construct a max-exposure instance as follows.

Let $R_1 = [0, \epsilon] \times [0, n]$ be a thin vertical rectangle and $R_2 = [0, n] \times [0, \epsilon]$ be a thin horizontal rectangle. For each vertex $v_i \in A$, we create a copy R_i of R_1, and place it such that its lower-left corner is at $(i, 0)$. Similarly, for each vertex $v_j \in B$, we create a copy R_j of R_2, and place it such that its lower-left corner is at $(0, j)$. These $|A| + |B|$ rectangles create a checkerboard arrangement, with $|A| \times |B|$ cells of intersection. For each edge $(v_i, v_j) \in E$, we place a single point in the cell corresponding to intersection of R_i and R_j. It is now easy to see that G has a k-subgraph with m^* edges if and only if we can expose m^* points in this instance by removing k-rectangles: the removed rectangles are exactly the k vertices chosen in the graph, and each exposed point corresponds to the edge included in the output subgraph. (See also Figure 1.)

▶ **Lemma 1.** *The max-exposure problem is at least as hard as bipartite-DkS.*

Since bipartite-DkS is known to be NP-hard [16], we have the following.

▶ **Theorem 2.** *Max-exposure problem with axis-aligned rectangles is NP-hard.*

### 2.1   Hardness of Approximation

The construction in the preceding proof shows that max-exposure with rectangles is at least as hard as bipartite-DkS problem. Moreover, the geometric construction uses translates of *only two rectangles* R_1, R_2. In the following, we show that even with such a restricted range space, the problem is also hard to approximate. To that end we prove that bipartite-DkS cannot be approximated better than a factor $O(n^{1/4})$, where n is the number of vertices in this graph. More precisely, if the densest subgraph over k vertices has m^* edges, it is hard to find a subgraph over k vertices that contains at least $\Omega(m^*/n^{1/4-\epsilon})$ edges in polynomial time. This hardness of approximation is conditioned on the so-called *dense vs random* conjecture [10] being true. Roughly speaking, we are given a graph G, constants $0 < \alpha, \beta < 1$, and a parameter k, and we want to distinguish between the following two cases.
1. (RANDOM) $G = G(n, p)$ where $p = n^{\alpha-1}$, that is, G has average degree approximately n^α.
2. (DENSE) G is adversarially chosen so that the densest k-subgraph of G has average degree k^β.
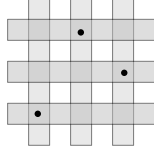
Figure 1 Reducing bipartite-DkS to max-exposure with axis-aligned rectangles.
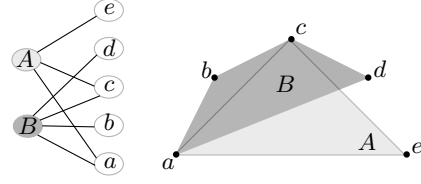
Figure 2 Reducing densest k-subhypergraph problem to max-exposure. Hypergraph vertices A, B shown as convex ranges.

The conjecture states that for all $0 < \alpha < 1$, sufficiently small $\epsilon > 0$, and for all $k \leq \sqrt{n}$, one cannot distinguish between the *dense* and *random* cases in polynomial time (w.h.p), when $\beta \leq \alpha - \epsilon$.

In order to obtain hardness guarantees using the above conjecture, one needs to find the 'distinguishing ratio' r, that is the least multiplicative gap between the optimum solution for the problem on the dense and random instances. If there exists an algorithm with an approximation factor significantly smaller than r, then we would be able to use it to distinguish between the dense and random instances, thereby refuting the conjecture. We obtain the following result for densest k-subgraph problem on bipartite graphs. (See Appendix A.1 for a proof.)

▶ **Lemma 3.** *Assuming that* dense vs random *conjecture is true, the densest k-subgraph problem on bipartite graphs is hard to approximate better than a factor $O(n^{1/4})$ of optimum.*

Using the same construction as in Lemma 1, we obtain the following.

▶ **Corollary 4.** *Assuming the dense vs random conjecture, max-exposure with axis-aligned rectangles is hard to approximate better than a factor $O(n^{1/4})$ of optimum.*

Hardness of Max-exposure with Convex Polygons

If the range space $(P, \mathcal{R})$ consists of convex polygons, the max-exposure problem is *equivalent* to the densest k-subhypergraph problem for general hypergraphs. A max-exposure instance $(P, \mathcal{R})$ naturally corresponds to a hypergraph $H = (\mathcal{R}, P)$ whose vertices are the ranges and the edges correspond to points and are defined by the containment relationship. Clearly, the densest k-subhypergraph corresponds to the set of k ranges deleting which exposes maximum number of points. For the other direction, we have the following lemma. (See also Figure 2.)

▶ **Lemma 5.** *Given a hypergraph $H = (X, E)$, one can construct a max-exposure instance with convex ranges $\mathcal{R}$ and points P such that the densest k-subhypergraph of H corresponds to a solution of max-exposure.*

Proof. For each edge $e \in E$ of the hypergraph, add a point $p_e \in P$. We place all the points of P in convex position. Let $v \in X$ be a vertex and E_v be the set of hyperedges adjacent to v. Then for every $v \in X$, we add a convex polygon $R_v \in \mathcal{R}$ such that the corners of R_v is precisely the point set E_v. Note that this is possible since points of P are in convex position. It is easy to see that in order to include an edge e (expose p_e), we must include all vertices in E_v, which corresponds to removing all polygons corresponding to vertices in E_v. ◀

3 A Bicriteria $O(k)$-approximation Algorithm

In this section, we present a simple approximation algorithm for the max-exposure problem that achieves bicriteria $O(k)$-approximation for range spaces defined by arbitrary axis-aligned rectangles. Specifically, if the optimal number of points exposed is m^*, the algorithm picks a subset of k^2 rectangles such that the number of points exposed is at least m^*/ck, for some constant c. In fact, the results hold for any polygonal range with $O(1)$ complexity.

This bicriteria approximation should be contrasted with the fact that no such approximation is possible for for the densest k-subhypergraph problem: that is, one cannot compute a set of $O(k^b)$ vertices for any constant b such that the number of edges in the induced subhypergraph is at least optimal. Thus the geometric properties of the range space have a significant impact on the problem complexity. In particular, if $\mathcal{R}$ consists of rectangle ranges, we show that the following strategy picks a subset of αk ranges such that the number of points exposed is at least $\alpha m^*/ck^2$, for a parameter $1 \le \alpha \le k$ and constant c that will be fixed later. Choosing $\alpha = k$ gives us the claimed bound.

Our algorithm is essentially greedy. We divide the points into maximal equivalence classes, where each class is the maximal subset of points belonging to the same subset of ranges. We define $\mathcal{R}(p)$ as the set of ranges that contain a point $p \in P$, and remove all points that are contained in more than k ranges, since they can be never exposed in the optimal solution. Therefore, without loss of generality, we can assume that $|\mathcal{R}(p)| \le k$ for all points $p \in P$.

Algorithm 1 Greedy-Bicriteria

1. Partition P into a set $\mathcal{G}$ of groups where each group $G_i \in \mathcal{G}$ is an equivalence class of points that are contained in the same set of ranges. That is, for any $p \in G_i$, $p' \in G_j$, we have $\mathcal{R}(p) = \mathcal{R}(p')$ if $i = j$ and $\mathcal{R}(p) \ne \mathcal{R}(p')$, otherwise.
2. Sort the groups in $\mathcal{G}$ by decreasing order of their size $|G_i|$ and select the first α groups. Return $m' = \sum_{1 \le i \le \alpha} |G_i|$ as the number of points exposed.

Observe that every point $p \in G_i$ is contained in the same set of ranges $\mathcal{R}_i = \mathcal{R}(p)$ and $|\mathcal{R}_i| \le k$. Therefore, the total number of ranges that we remove is at most αk. It remains to show that the number of points exposed m' is at least $\alpha m^*/ck^2$.

▶ **Lemma 6.** *Let m' be the number of points exposed by the algorithm* Greedy-Bicriteria, *and let m^* be the optimal number of exposed points, Then, $m' \ge \alpha m^*/ck^2$.*

Proof. Consider the optimal set $\mathcal{R}^*$ of k ranges that are deleted, and let P^* be the set of exposed points. We partition the set of points P^* into groups $\mathcal{G}^*$ as before, such that each group $G_i^* \in \mathcal{G}^*$ is identified by the range set $\mathcal{R}_i^* = \mathcal{R}(p)$, for any $p \in G_i^*$. Since $P^* \subseteq P$, we must have that $\mathcal{G}^* \subseteq \mathcal{G}$. This holds because for every group $G_i^* \in \mathcal{G}^*$ there must be a group $G_i \in \mathcal{G}$ such that $\mathcal{R}_i^* = \mathcal{R}_i$. Moreover since P^* is the maximum set of points that can be exposed, we must have that $G_i^* = G_i$. Finally, we note that the number of groups $|\mathcal{G}^*|$ is bounded by the number of cells in the arrangement of ranges in $\mathcal{R}^*$ which is at most ck^2 for some fixed constant c, for all $O(1)$-complexity ranges.

If the groups in $\mathcal{G}$ are arranged by decreasing order of their sizes, we have that

$$m^* = \sum_{1 \le i \le |\mathcal{G}^*|} |G_i^*| \le \sum_{1 \le i \le |\mathcal{G}^*|} |G_i| \le \sum_{1 \le i \le ck^2} |G_i| \le \frac{ck^2}{\alpha} \sum_{1 \le i \le \alpha} |G_i| = \frac{ck^2}{\alpha} \cdot m'$$

◀

The parameter α can be tuned to improve the approximation guarantee with respect to one criterion (say the number of exposed points) at the cost of other. With $\alpha = k$, the algorithm exposes at least $\Omega(m^*/k)$ by removing k^2 ranges. If the range space $\mathcal{R}$ consists of pseudodisk of *bounded-ply* (no point in the plane is incident to more than a constant number ρ of pseudodisks), then the algorithm Greedy-Bicriteria achieves an $O(\rho)$ approximation. This holds because the number of cells in an arrangement of k pseudodisks with depth at most ρ is $O(\rho k)$ [11].

4   A PTAS for Unit Square Ranges

We have seen that max-exposure is hard to approximate even if the ranges are translates of two types of rectangles. We now describe an approximation scheme when the ranges are translates of a *single* rectangle. In this case, we can scale the axes so that the rectangle becomes a *unit square* without changing any point-rectangle containment. Therefore, we can assume that our ranges are all unit squares. The problem is non-trivial even for unit square ranges, and as a warmup we first solve the following special case: *all the points lie inside a unit square*. We develop a dynamic programming algorithm to solve this case exactly, and then use it to design an approximation for the general set of points.

4.1   Exact Solution in a Unit Square

We are given a max-exposure instance consisting of unit square ranges $\mathcal{R}$ and a set of points P in a unit square C. Without loss of generality, we can assume that the lower left corner of C lies at origin $(0, 0)$ and all ranges in $\mathcal{R}$ intersect C. We classify the ranges in $\mathcal{R}$ to be one of the two types: (See also Figure 3).

Type-0 : Unit square ranges that intersect $x = 0$.
Type-1 : Unit square ranges that intersect $x = 1$.

(A unit square range coincident with both $x = 0$ and $x = 1$ is assumed to be Type-0). We draw two parallel horizontal lines $\ell_0 : y = 0$ and $\ell_1 : y = 1$ coincident with bottom and top horizontal sides of C respectively. We say that a range $R \in \mathcal{R}$ is *anchored* to a line ℓ if it intersects ℓ. Note that every $R \in \mathcal{R}$ is anchored to exactly one of ℓ_0 or ℓ_1. (When R is coincident with both ℓ_0 and ℓ_1, we say that it is anchored to ℓ_0). Moreover, for the rest of our discussion, let $x = x_i$ be a vertical line and define $P_i \subseteq P$ to be the set of points that have x-coordinate at least x_i. Similarly, define $\mathcal{R}_i \subseteq \mathcal{R}$ to be the set of ranges that have at least one corner to the right of $x = x_i$. That is a range $R \in \mathcal{R}_i$ either intersects $x = x_i$ or lies completely to the right of it.
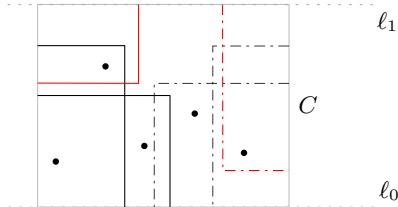
Figure 3 Max-exposure in a unit square C. Type 0 ranges are drawn with solid lines, Type 1 ranges are dash-dotted.
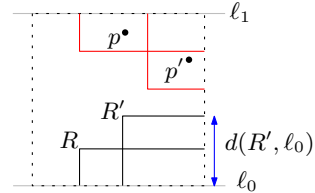
Figure 4 An example of *closer* relationship. Point p is *closer* to ℓ_1 than p'. R is closer to ℓ_0 than R'.

In order to gain some intuition, we will first consider the following two natural dynamic programming formulations for the problem.

DP-template-0   Suppose that the points in P are ordered by their increasing x-coordinates and let x_i be the x-coordinate of the ith point p_i. We define a subproblem as $S(i, k', \mathcal{R}_d)$ which represents the maximum number of points in P_i that can be exposed by removing k' ranges from the set $\mathcal{R}_i \setminus \mathcal{R}_d$. If we define $x_0 = 0$, then $S(0, k, \emptyset)$ gives the optimal number of exposed points for our problem.

Let $k_i = |\mathcal{R}(p_i) \setminus \mathcal{R}_d|$ be the number of ranges of $\mathcal{R}_i \setminus \mathcal{R}_d$ that contain p_i. Then, we can can express the subproblems at i in terms of subproblems at $i + 1$ as follows.

$$S(i,\ k',\mathcal{R}_d) = \max \begin{cases} S(i + 1,\ k' - k_i,\ \mathcal{R}_d \cup \mathcal{R}(p_i))\ +\ 1 & \text{expose } p_i \\ S\,(i + 1,\ k',\ \mathcal{R}_d) & \text{otherwise} \end{cases}$$

Roughly speaking, at $x = x_i$ which is the *event* corresponding to a point $p_i \in P$, we have two choices : *expose p_i* or *do not expose p_i*. If we expose p_i, we pay for deleting the ranges in $\mathcal{R}_i \setminus \mathcal{R}_d$ that contain p_i and mark them as deleted by adding to the *deleted range set $\mathcal{R}_d$*. Moreover, since we only delete ranges from $\mathcal{R}_i \setminus \mathcal{R}_d$, we can assume that $\mathcal{R}_d = \mathcal{R}_d \cap \mathcal{R}_i$ at each x_i. It is easy to see that this correctly computes the optimal number of exposed points. However, there is one complication: a priori it is not clear how to bound the number of range subset $\mathcal{R}_d$ used by this dynamic program. We later argue that the geometry of range space for Type-0 ranges allows us to use only a polynomial number of choices.

DP-template-1   An alternative approach is to consider both *point* and *begin-range* events. That is, $x = x_i$ is either incident to a point $p_i \in P$ or to the left vertical side of a range $R_i \in \mathcal{R}$. Then, we can define a subproblem by the tuple $S(i, k', P_f)$ which represents the maximum number of points in $(P_i \setminus P_f)$ that can be exposed by removing k' ranges in $\mathcal{R}_i$. If we define $x_0 = 0$, then $S(0, k, \emptyset)$ gives the optimal number of exposed points. Let $P(R_i) \subseteq P$ be the set of points contained in the range R_i, then we have the following recurrence.

$$\begin{aligned}
S(i,\ k', P_f)\ &=\ \max \begin{cases} S(i + 1,\ k' - 1,\ P_f) & \text{delete range } R_i \\ S(i + 1,\ k',\ P_f \cup P(R_i)) & \text{otherwise} \end{cases} \\
&\qquad \textit{(event } x = x_i \textit{ was beginning of a range } R_i \in \mathcal{R}_i\textit{)} \\
&=\ \max \begin{cases} S(i + 1,\ k', P_f) & \text{if } p_i \in P_f, \text{ cannot expose } p_i \\ S(i + 1,\ k', P_f) + 1 & \text{otherwise, expose } p_i \end{cases} \\
&\qquad \textit{(otherwise, event } x = x_i \textit{ was a point } p_i \in P_i\textit{)}
\end{aligned}$$

In the above formulation, at each *begin-range* event for some $R_i \in \mathcal{R}_i$, we have two choices: *delete R_i* or *do not delete R_i*. If R_i was deleted, we reduce the budget k' by one. Otherwise, if R_i was not deleted, we can never expose the points in $P(R_i)$, and therefore we add $P(R_i)$ to the *forbidden point set P_f*. The correctness of the dynamic program follows from the fact that for every point p_i, all the ranges containing it must begin before $x = x_i$, and we expose p_i only if those ranges were *deleted*. Finally, since we only expose points in $P_i \setminus P_f$, we can assume that $P_f = P_f \cap P_i$ at each x_i. Again, it is not obvious how many different subsets P_f are needed by the dynamic program. However, we will later show that by keeping track of polynomial number of sets P_f, we can solve max-exposure with Type-1 ranges.

We note that the Type-0 and Type-1 ranges may superficially seem symmetric but once we fix the order of computing subproblems, they become structurally different. Therefore, we would need slightly different techniques to handle each type. For the ease of exposition, we present dynamic programs for Type-0 and Type-1 ranges separately and finally combine them.

We first define the following ordering relations that will be useful. Let ℓ be a horizontal line, and let $d(p, \ell)$ denote the orthogonal distance of $p \in P$ from ℓ. If $p, p' \in P$ are two points, we say that p is *closer* to ℓ than p' if $d(p, \ell) < d(p', \ell)$. Similarly, for a range $R \in \mathcal{R}$ that is anchored to ℓ, let $d(R, \ell)$ be the vertical distance *inside the unit square C* between ℓ and the side of R parallel to ℓ. If $R, R' \in \mathcal{R}$ are two ranges, we say that R is *closer* (or equivalently R' is *farther*) from ℓ if both R, R' are anchored to ℓ and $d(R, \ell) < d(R', \ell)$. (See Figure 4.)

4.1.1 Max-exposure with Type-0 Ranges

Recall that Type-0 ranges intersect the vertical lines $x = 0$ and are anchored to either ℓ_0 or ℓ_1. We will apply the formulation discussed in *DP-template-0*. The key challenge here is to bound the number of possible deleted range sets $\mathcal{R}_d$. Towards that end, we make the following claim.

▶ **Lemma 7.** *Let q_0, q_1 be the two exposed points strictly to the left of $x = x_i$ that are closest to ℓ_0 and ℓ_1 respectively. Then our dynamic program only needs to consider the set of deleted ranges $\mathcal{R}_d = \mathcal{R}(q_0) \cup \mathcal{R}(q_1)$ at $x = x_i$ conditioned on q_0, q_1.*

Proof. Observe that since $\mathcal{R}$ consists of Type-0 ranges, every range in $\mathcal{R}_i$ must intersect the vertical line $x = x_i$. Suppose we partition $\mathcal{R}_i$ into ranges $\mathcal{R}_i^0$ that are anchored to ℓ_0 and $\mathcal{R}_i^1$ that are anchored to ℓ_1. Let $P' \subseteq P$ be the set of all *exposed points* strictly to the left of $x = x_i$. Observe that for all $p \in P'$, any range $R \in \mathcal{R}_i^0$ that contains p must also contain q_0. Therefore, we must have $\mathcal{R}_i^0 \cap \mathcal{R}(p) \subseteq \mathcal{R}_i^0 \cap \mathcal{R}(q_0)$, for all $p \in P'$. Similarly, $\mathcal{R}_i^1 \cap \mathcal{R}(p) \subseteq \mathcal{R}_i^1 \cap \mathcal{R}(q_1)$, for all $p \in P'$. Hence, $\bigcup_{p \in P'} \mathcal{R}_i \cap \mathcal{R}(p) = \mathcal{R}(q_0) \cup \mathcal{R}(q_1)$. Recall that $\mathcal{R}_d$ is precisely the set of ranges at $x = x_i$ that contain any exposed point to the left of $x = x_i$, so we have $\mathcal{R}_d = \mathcal{R}(q_0) \cup \mathcal{R}(q_1)$. ◀

Therefore, if our dynamic program remembers the exposed points q_0, q_1, then we can compute the deleted range set $\mathcal{R}_d = \mathcal{R}(q_0) \cup \mathcal{R}(q_1)$ at $x = x_i$. There are $O(m^2)$ choices for the pair q_0, q_1, so the number of possible sets $\mathcal{R}_d$ is also $O(m^2)$. We can therefore identify our subproblems by the tuple $S(i, k', q_0, q_1)$ which represents the maximum number of exposed points with x-coordinates x_i or higher using k' rectangles from the set $\mathcal{R}_i \setminus \mathcal{R}_d$. With $k_i = |\mathcal{R}(p_i) \setminus \mathcal{R}_d|$, we obtain the following recurrence:

$$
S(i, k', q_0, q_1) = \max \begin{cases} S(i+1, k' - k_i, closer(p_i, q_0), closer(p_i, q_1)) + 1 & \text{expose } p_i \\ S(i+1, k', q_0, q_1) & \text{otherwise} \end{cases}
$$

where the function $closer(p_i, q_0)$ returns whichever of p_i, q_0 is closer to ℓ_0, and $closer(p_i, q_1)$ returns whichever of p_i, q_1 is closer to ℓ_1. The optimal solution is given by $S(0, k, q_0^*, q_1^*)$, where $q_0^* = (0, 1)$ and $q_1^* = (0, 0)$ are two artificial points with $\mathcal{R}(q_0^*) = \mathcal{R}(q_1^*) = \emptyset$ (not contained in any range). The base case is defined by the vertical line $x = 1$ and is initialized with zeroes for all q_0, q_1 and $k' \geq 0$. Any subproblem with $k' < 0$ has value $-\infty$.

4.1.2   Max-exposure with Type-1 Ranges

Next we consider the case when we only have Type-1 ranges in $\mathcal{R}$. Unfortunately in this case, our previous dynamic program does not work and we need to remember a different set of parameters. More precisely, we will apply the formulation discussed in *DP-template-1*, and bound the number of possible forbidden point sets P_f.

▶ **Lemma 8.** *Let Q_0, Q_1 be two ranges that begin to the left of $x = x_i$ and were* not *deleted. Moreover, Q_0 is anchored to and is farthest from ℓ_0. Similarly Q_1 is anchored to and is farthest from ℓ_1 (Figure 5). Then the forbidden point set at $x = x_i$ is given by $P_f = P(Q_0) \cup P(Q_1)$, where $P(Q)$ is the set of points contained in range Q.*

Proof. Recall that the set $\mathcal{R}_i$ consists of ranges that have at least one corner to the right of the vertical line $x = x_i$. Since we are dealing with Type-1 ranges, every range that begins to the left of $x = x_i$ lies in $\mathcal{R}_i$. Now let $\mathcal{R}' \subseteq \mathcal{R}_i$ be the set of ranges that begin to the left of $x = x_i$ and were *not deleted*. Recall that P_i is the set of points in P that have x-coordinate x_i or higher. Now consider any range $R \in \mathcal{R}'$. Observe that if R was anchored to ℓ_0, then every point of P_i that lies in R also lies in Q_0. Otherwise, if R was anchored to ℓ_1, every point of P_i that lies in R also lies in Q_1. Therefore, we must have $\bigcup_{R \in \mathcal{R}'} (P_i \cap P(R)) = P(Q_0) \cup P(Q_1)$. Recall that P_f was precisely the set of points in P_i contained in ranges that begin to the left of $x = x_i$ and were not deleted. Therefore, we have that $P_f = P(Q_0) \cup P(Q_1)$.   ◀
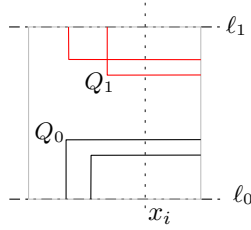
Figure 5 Undeleted ranges Q_0 and Q_1 farthest from ℓ_0 and ℓ_1 respectively.
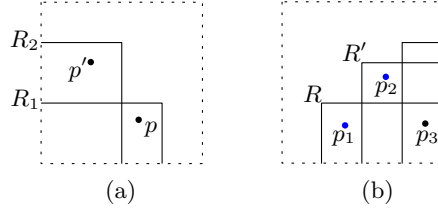
Figure 6 Remembering one of R_1, R_2 in (a) or one of p_1, p_2 in (b) is not sufficient.

Therefore, if our dynamic program remembers the ranges Q_0 and Q_1, we can compute the forbidden point set $P_f = P(Q_0) \cup P(Q_1)$ at $x = x_i$. Since there are $O(n^2)$ choices for the pair Q_0, Q_1, the number of possible sets P_f is also $O(n^2)$. We can now identify the subproblems by the tuple $S(i, k', Q_0, Q_1)$ which represents the maximum number of points in $P_i \setminus P_f$ that are exposed by deleting k' ranges that begin on or after $x = x_i$. This gives us the following recurrence.

$$S(i, k', Q_0, Q_1) =$$

$$\max \begin{cases} S(i+1, k'-1, Q_0, Q_1) & \text{delete range } R_i \\ S(i+1, k', farther(R_i, Q_0), Q_1) & \text{otherwise, } R_i \text{ is not deleted and anchored to } \ell_0 \\ S(i+1, k', Q_0, farther(R_i, Q_1)) & \text{otherwise, } R_i \text{ is not deleted and anchored to } \ell_1 \end{cases}$$

$\qquad$ (*event $x = x_i$ was beginning of a range $R_i \in \mathcal{R}$*)

$$\max \begin{cases} S(i+1, k', Q_0, Q_1) & \text{if } p_i \in P_f, \text{ cannot expose } p_i \\ S(i+1, k', Q_0, Q_1) + 1 & \text{otherwise, expose } p_i \end{cases}$$

$\qquad$ (*otherwise, event $x = x_i$ was a point $p_i \in P$*)

Here, $farther(R_i, Q_0)$ returns whichever of R_i, Q_0 is farther from ℓ_0; and $farther(R_i, Q_1)$ returns whichever of R_i, Q_1 is farther from ℓ_1. The optimal solution is given by $P(0, k, Q_0^*, Q_1^*)$, where Q_0^*, Q_1^* are two artificial ranges of zero-width : Q_0^* is anchored to ℓ_0 and is defined by corners $(0, 0)$ and $(0, 1)$; similarly, Q_1^* is anchored to ℓ_1 and is defined by corners $(0, 1)$ and $(1, 1)$.

▶ **Remark 9.** We note that remembering constant number of exposed points q_0, q_1 or a constant number of undeleted ranges Q_1, Q_2 by themselves cannot solve *both* Type-0 and Type-1 ranges. For instance, in Figure 6(a) with Type-0 ranges, if R_1, R_2 were both *not deleted* but we remembered one of them, then we will incorrectly expose one of p, p'. Similarly in Figure 6(b) with Type-1 ranges, if p_1, p_2 were both exposed but we only remembered one of them, we will pay for one of the ranges R, R' again when we expose p_3. However, since the previous dynamic programs for Type-0 and Type-1 ranges express subproblems at event i in terms of subproblems at event $i + 1$, we can easily combine them with minor adjustments.

4.1.3   Combining them together

In the following, we combine the dynamic programs for Type-0 and Type-1 ranges to obtain a dynamic program for max-exposure in a unit square C. We will need a couple of changes. First, the events at $x = x_i$ are now defined by either a point $p_i \in P$ or beginning of a Type-1 range R_i. Next, the *deleted range set* $\mathcal{R}_d$ at $x = x_i$ will only consist of Type-0 ranges and is defined as $\mathcal{R}_d = \mathcal{R}_{i0} \cap (\mathcal{R}(q_0) \cup \mathcal{R}(q_1))$ where $\mathcal{R}_{i0} \subseteq \mathcal{R}_i$ is the set of Type-0 ranges that intersect the vertical line $x = x_i$, The *forbidden point set* $P_f = P(Q_0) \cup P(Q_1)$ stays the same. Here q_0, q_1, Q_0, Q_1 are same as defined before. The subproblems represent the maximum number of points in $P_i \setminus P_f$ that can be exposed by deleting k' ranges from $\mathcal{R}_i \setminus \mathcal{R}_d$. If $k_i = |\mathcal{R}(p_i) \setminus \mathcal{R}_d|$, then we obtain the following combined recurrence.

$$
\begin{aligned}
&S(i, \ k', \ q_0, \ q_1, Q_0, Q_1) \ = \\
&\quad \max \begin{cases} S(i+1, \ k', \ q_0, \ q_1, \ Q_0, \ Q_1) & \text{if } p_i \in P_f, \text{ cannot expose } p_i \\ S(i+1, \ k', \ q_0, \ q_1, \ Q_0, \ Q_1) & \text{choose to not expose } p_i \\ S(i+1, \ k'-k_i, \ closer(q_0, p_i), \ closer(q_1, p_i) \ ) \ + \ 1 & \text{otherwise, expose } p_i \end{cases} \\
&\qquad (\textit{event } x = x_i \textit{ was a point } p_i \in P_i) \\[6pt]
&\quad \max \begin{cases} S(i+1, \ k'-1, \ q_0, \ q_1, \ Q_0, \ Q_1) & \text{delete Type-1 range } R_i \\ S(i+1, \ k', \ q_0, \ q_1, \ farther(R_i, Q_0), \ Q_1) & R_i \text{ not deleted and anchored to } \ell_0 \\ S(i+1, \ k', \ q_0, \ q_1, \ Q_0, \ farther(R_i, Q_1)) & R_i \text{ not deleted and anchored to } \ell_1 \end{cases} \\
&\qquad (\textit{event } x = x_i \textit{ was beginning of a Type-1 range } R_i \in \mathcal{R}_i)
\end{aligned}
$$

The optimal solution is given by $S(0, \ k, \ q_0^*, \ q_1^*, \ Q_0^*, \ Q_1^*)$. The correctness of the above formulation follows from the fact that when we choose to expose p_i, we are guaranteed that all Type-1 ranges in $\mathcal{R}(p_i)$ have already been deleted, and the expression k_i only charges for Type-0 ranges containing p_i. As for the running time, for each event $x = x_i$, we compute $O(kn^2m^2)$ entries and computing each entry takes constant time. Since there are $O(n + m)$ events, we obtain the following.

▶ **Lemma 10.** *Given a set P of m points in a unit square C and a set of n unit square ranges $\mathcal{R}$, we can compute their max-exposure in $O(k(n+m)n^2m^2)$ time.*

4.2 A Constant Factor Approximation

We now use the preceding algorithm to solve the max-exposure problem for general set of points and unit square ranges within a factor 4 of optimum. In particular, we compute a set of $4k$ ranges in $\mathcal{R}$ such that the number of points exposed in P by deleting them is at least the optimal number of points. Suppose we embed the ranges $\mathcal{R}$ on a uniform unit-sized grid G, and define $\mathcal{C}$ as the collection of all cells in G that contain at least one point of P. We have the following approximation algorithm.

Algorithm 2 DP-Approx

1. Apply Lemma 10 to solve max-exposure locally in every cell $C_i \in \mathcal{C}$ for all $0 \leq k_i \leq k$. Call this a *local* solution denoted by $local(P(C_i), \mathcal{R}(C_i), k_i)$, where $P(C_i) \subseteq P$ is the set of points contained in cell C_i and $\mathcal{R}(C_i)$ is the set of ranges intersecting C_i.

2. Process cells in $\mathcal{C}$ in any order $C_1, C_2, \ldots, C_g$, and define $global(i,\ k')$ as the maximum number of points exposed in the cells C_i through C_g using k' ranges. Combine local solutions to obtain $global(i,\ k')$ as follows.

$$global(i,\ k')\ =\ \max_{0 \leq k_i \leq k'}\ global(i+1,\ k'-k_i)\ +\ local(P(C_i),\ \mathcal{R}(C_i),\ k_i)$$

3. Return $global(1, 4k)$ as the number of exposed points.

We have the following lemma. (See Section A.2 in the Appendix for a proof.)

▶ **Lemma 11.** *If $P^* \subseteq P$ is the optimal set of exposed points, then $global(1, 4k) \geq |P^*|$, that is , the algorithm* DP-Approx *achieves a 4-approximation and runs in $O(k(n+m)n^2m^2)$ time.*

4.3 Towards a PTAS

We now consider the max-exposure instance in a horizontal strip of unit width. That is, all points in P lie in a horizontal strip bounded by lines ℓ_0, ℓ_1 and $\mathcal{R}$ consists of unit square ranges. Suppose, we subdivide the strip into unit square cells $C_1, C_2, \ldots, C_r \in \mathcal{C}$ ordered from left to right. We make the following simple observation.
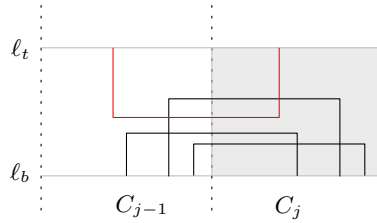
Figure 7 Max-exposure instance in a strip. $C_{j-1}, C_j \in \mathcal{C}$ are two consecutive cells.

▶ **Lemma 12.** *Let $R \in \mathcal{R}$ be a unit square range and C_{j-1} be the first cell from left which it intersects. Then the only other cell that R can intersect is C_j. Moreover, R is Type-1 with respect to C_{j-1} and Type-0 with respect to C_j. (See Figure 7.)*

Observe that the set of points exposed in cell C_j will also depend on the set of Type-0 objects of C_j that were already deleted in C_{j-1}. So we need to ensure that we do not double count the set of ranges that were already deleted in C_{j-1}. To do this, we again use a dynamic

program similar to that for max-exposure within a cell where we express the subproblems at $x = x_i$ in terms of subproblems to the right of $x = x_i$. However, there are some important differences in how we define our subproblems. First, events at a vertical line $x = x_i$ are one of three types:

1. *cell-boundary*: $x = x_i$ is coincident with left-boundary of a cell $C_j \in \mathcal{C}$,
2. *begin-range*: $x = x_i$ is coincident with left-vertical side of a range $R_i \in \mathcal{R}$
3. *point*: $x = x_i$ is incident to an input $p_i \in P$

Moreover for a given cell C_j, in addition to the points q_0, q_1, and ranges Q_0, Q_1, we will also need to remember two additional ranges : L_0 (anchored to ℓ_0) and L_1 (anchored to ℓ_1) that begin in C_{j-1}, were *not deleted* and are farthest from ℓ_0, ℓ_1 respectively. For the sake of clarity, we will use $Z_0 = (q_0, Q_0, L_0)$ to denote the triplets corresponding to ℓ_0 and $Z_1 = (q_1, Q_1, L_1)$ to denote the triplets corresponding to ℓ_1.

Suppose $x = x_i$ lies in the cell C_j. Then we show that the set of deleted ranges $\mathcal{R}_d$ consisting of Type-0 ranges in C_j, and the set of forbidden points P_f can be uniquely identified using the triples Z_0, Z_1.

- *Deleted Type-0 range-set $\mathcal{R}_d$* Let $\mathcal{R}_{j-1}$ be the set of ranges that begin in cell C_{j-1}, and therefore are Type-1 with respect to C_{j-1}. Suppose we define $\mathcal{L}_{>0} \subseteq \mathcal{R}_{j-1}$ to be the set consisting of ranges anchored to ℓ_0 and farther from ℓ_0 than L_0. Similarly, $\mathcal{L}_{>1} \subseteq \mathcal{R}_{j-1}$ consists of ranges anchored to ℓ_1 and farther from ℓ_1 than L_1. Then, we define $\mathcal{R}_d = (\mathcal{R}(q_0) \cup \mathcal{R}(q_1) \cup \mathcal{L}_{>0} \cup \mathcal{L}_{>1})$.

- *Forbidden point-set P_f* We define $P_f = (P(L_0) \cup P(L_1) \cup P(Q_0) \cup P(Q_1))$.

Finally, we say that a range R *dominates* another range R', if both R, R' begin in the same cell C_j and $R' \cap C_j \subseteq R \cap C_j$. That is, R completely contains the part of R' that lies in cell C_j. Note that the key difference from earlier formulations is that at a begin-range event for a Type-1 range R_i in cell C_j, we choose to *ignore* R_i if it is dominated by ranges Q_0 or Q_1, because the points of R_i contained in C_j already lie in the forbidden set P_f. With $k_i = |\mathcal{R}(p_i) \setminus \mathcal{R}_d|$, we obtain the following recurrence.

$$S(i, \ k', \ Z_0, \ Z_1) \ = \ S(i+1, \ k, \ \mathcal{U}(Z_0, C_j), \ \mathcal{U}(Z_1, C_j))$$

$$(\textit{event } x = x_i \textit{ is left-boundary of cell } C_j)$$

$$\max \begin{cases} S(i+1, \ k', \ Z_0, \ Z_1) & \text{if } p_i \in P_f, \text{ cannot expose } p_i \\ S(i+1, \ k', \ Z_0, \ Z_1) & \text{otherwise, choose to not expose } p_i \\ S(i+1, \ k'-k_i, \ \mathcal{U}(Z_0, p_i), \ \mathcal{U}(Z_1, p_i) \ ) + 1 & \text{otherwise, expose } p_i \end{cases}$$

$$(\textit{otherwise, event } x = x_i \textit{ was a point } p_i \textit{ in cell } C_j)$$

$$\max \begin{cases} S(i+1, \ k', \ Z_0, \ Z_1) & \text{if either } Q_0 \text{ or } Q_1 \text{ dominates } R_i, \textit{ ignore } R_i \\ S(i+1, \ k'-1, \ Z_0, \ Z_1) & \text{otherwise, delete Type-1 range } R_i \\ S(i+1, \ k', \ \mathcal{U}(Z_0, R_i), Z_1) & \text{otherwise if } R_i \text{ is not deleted and anchored to } \ell_0 \\ S(i+1, \ k', \ Z_0, \ \mathcal{U}(Z_1, R_i)) & \text{otherwise, } R_i \text{ is not deleted and anchored to } \ell_1 \end{cases}$$

$$(\textit{otherwise, event } x = x_i \textit{ was beginning of a Type-1 range } R_i \textit{ in cell } C_j.)$$

The function $\mathcal{U}(Z, \ E)$ used above is defined as follows. Roughly speaking, it updates the triplets $Z \in \{Z_0, Z_1\}$ based on the event E and returns an updated triplet. We have the following three cases.

- For a cell-boundary event C_j, if we have $Z_0 = (q_0, Q_0, L_0)$, the function $\mathcal{U}(Z_0, C_j) = (q_0^*, \ Q_0^*, \ Q_0)$. Similarly, $\mathcal{U}(Z_0, C_j) = (q_1^*, \ Q_1^*, \ Q_1)$. This corresponds to resetting the

points q_0, q_1, rectangles Q_0, Q_1 for the current cell C_j, and remembering the rectangles L_0, L_1 from the previous cell C_{j-1}.

- For a point event p_i, we have $\mathcal{U}(Z_0, p_i) = (closer(p_i, q_0),\ Q_0,\ L_0)$ and similarly $\mathcal{U}(Z_1, p_i) = (closer(p_i, q_1),\ Q_1,\ L_1)$. Recall that the function $closer(p_i, q_0)$ returns whichever of p_i, q_0 is closer to ℓ_0, and $closer(p_i, q_1)$ returns whichever of p_i, q_1 is closer to ℓ_1.

- Finally for a begin-rectangle event R_i, we have $\mathcal{U}(Z_0, R_i) = (q_0,\ farther(R_i, Q_0),\ L_0)$ and $\mathcal{U}(Z_1, R_i) = (q_1,\ farther(R_i, Q_1),\ L_1)$. Recall that the function $farther(R_i, Q_0)$ returns whichever of R_i, Q_0 is farther from ℓ_0, and $farther(R_i, Q_1)$ returns whichever of R_i, Q_1 is farther from ℓ_1.

The optimal solution is given by $W(0,\ k,\ Z_0^\emptyset,\ Z_1^\emptyset)$ where $Z_0^\emptyset = (q_0^*,\ Q_0^*,\ Q_0^*)$ and $Z_1^\emptyset = (q_1^*,\ Q_1^*,\ Q_1^*)$. In order to establish the correctness of the above formulation, we make the following claim.

▶ **Lemma 13.** *Let $P^* \subseteq P$ be the optimal set of exposed points. Then, for every point $p_i \in P^*$, we count the range $R \in \mathcal{R}(p_i)$ towards the total number of deleted ranges exactly once.*

Proof. We begin by noting that R intersects at most two cells : C_{j-1} as a Type-0 range and C_j as a Type-1 range. It suffices to show that we count R towards the total number of deleted ranges in exactly one of these two cells. Alternatively, it suffices to show that we count R in cell C_j if and only if we have not already counted R in C_{j-1}. Recall that we can only count for R in C_{j-1} by deleting it at a begin-range event. Moreover, we can only count for R in C_j when a point $p_i \notin P_f$ that lies in cell C_j is exposed. Without loss of generality, assume that R is anchored to ℓ_0. The case when R is anchored to ℓ_1 is symmetric.

We first consider the easy case when R was not deleted in C_{j-1}. Observe that since R is Type-0 with respect to C_j, similar to the earlier cases, the terms $\mathcal{R}(q_0) \cup \mathcal{R}(q_1)$ in the expression for $\mathcal{R}_d$ will correctly charge for R in cell C_j.
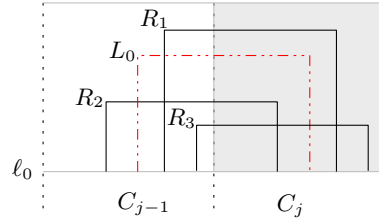
Figure 8 Three cases for the proof: $R_1 \in \mathcal{L}_{>0}$, and $R_2, R_3 \notin \mathcal{L}_{>0}$. R_2 begins before L_0 and R_3 begins after L_0.

Now, we move to the second case where we are currently in cell C_j and we have already counted R by deleting it at a begin-range event in cell C_{j-1}. In this case, we show that we will not count R again in C_j. More precisely, we show that if R contains a point p that lies in cell C_j but *is not contained* in the forbidden point set P_f, then the deleted range set $\mathcal{R}_d$ contains R, and therefore the expression $k_i = \mathcal{R}(p) \setminus \mathcal{R}_d$ will not charge for R again. We have three cases.

1. $R \in \mathcal{L}_{>0}$. This case is straightforward as $\mathcal{R}_d$ contains all ranges in $\mathcal{L}_{>0}$.
2. $R \notin \mathcal{L}_{>0}$ *and R begins before L_0*. This case is not possible because any point that is contained in $(R \cap C_j)$ is also contained in L_0. This holds because R and L_0 have the same width, so if R begins before L_0 in C_{j-1}, it must end before L_0 in C_j. Since every point contained in L_0 is contained in the forbidden set P_f, we must have $p \in P_f$ which is a contradiction. (See Figure 8 with $R = R_2$.)

3. $R \notin \mathcal{L}_{>0}$ *and R begins after L_0.* This case is also not possible because if this were true L_0 would have dominated R. Therefore, we would have *ignored R* in C_{j-1} and would not have deleted it. (See Figure 8 with $R = R_3$.)

◀

▶ **Lemma 14.** *The restricted max-exposure instance such that all points in P lie within a unit-width horizontal strip bounded by lines ℓ_0, ℓ_1 and $\mathcal{R}$ consists of unit squares can be solved in $O(k(n+m)n^4m^2)$ time, where $m = |P|$ and $n = |\mathcal{R}|$.*

Using similar ideas as Lemma 11, the above lemma readily gives a 2-approximation for max-exposure. More precisely, we can embed the input instance on to a unit-sized grid as before, but instead of solving max-exposure in a cell, we use the above algorithm to solve max-exposure locally in a row of the grid. Since each range $R \in \mathcal{R}$ can intersect at most two rows, R is split into two sub-ranges R_1, R_2 contained in at most two rows. Since these new sub-ranges in two different rows are disjoint, there exists an optimal solution with $2k$ sub-ranges. Therefore, if we have already computed the local solutions for each row i, using the algorithm *DP-Approx* we can compute *global*$(1, 2k)$ which exposes at least optimal number of points using at most $2k$ ranges.

▶ **Corollary 15.** *There exists a 2-approximation algorithm for max-exposure with unit square ranges running in $O(k(n+m)n^4m^2)$ time.*

Generalizing to h anchor lines   The dynamic program for max-exposure in a horizontal strip bounded by two anchor lines ℓ_0, ℓ_1 can be generalized to the case when we have h anchor lines $\ell_1, \ell_2, \ldots, \ell_h$. However, there is a minor technical change required. Observe that for a given anchor line ℓ_i, there can be points and anchored ranges on either side of ℓ_i. Therefore, we will need to remember the closest exposed points and the farthest undeleted ranges on both sides of ℓ_i. So for each anchor line ℓ_i, we will need the triplet $Z_i^+ = (q_i^+, Q_i^+, L_i^+)$ for points and ranges above ℓ_i and the triplet $Z_i^- = (q_i^-, Q_i^-, L_i^-)$ for points and ranges below ℓ_i. The dynamic program will now need to remember at most $4h$ ranges and $2h$ points which gives a running time of $O(k(n+m)n^{4h}m^{2h})$. If we denote a collection of h consecutive anchor lines by a *bundle* of width h, then we have the following.

▶ **Lemma 16.** *Max-exposure in a bundle of width h can be solved in $O(k(n+m)n^{4h}m^{2h})$ time.*

4.4   An $(1 + \epsilon)$-Approximation Algorithm

We are now ready to describe our PTAS for the problem. Suppose the anchor lines correspond to the horizontal lines of the uniform unit-sized grid G. Since we have already solved max-exposure exactly for h consecutive rows in G, we can now apply standard shifting techniques [18] to obtain an $(1 + \epsilon)$-approximation. If P^* is the optimal set of exposed points, then we show how to compute a set of $(1 + \epsilon)k$ ranges deleting which will expose at least $|P^*|$ points. Note that using similar ideas, it is also possible to expose at least $(1 - \epsilon)|P^*|$ points by deleting exactly k ranges (See Appendix B).

Suppose that anchor lines $\ell_1, \ell_2, \ldots, \ell_z$ are ordered by increasing y-coordinates. We define a *bundle B_j* to be a set of h consecutive anchor lines, identified by the lowest index anchor ℓ_j. We also define *bundle-set* to be a sequence of consecutive bundles, identified by the index of the lowest bundle. For instance the bundle B_1 comprises of anchor lines ℓ_1 through ℓ_h (inclusive). And the bundle-set $\mathcal{B}_1$ comprises of bundles $B_1, B_h, B_{2h}, \ldots B_{\lceil z/h \rceil}$. The lines $\ell_1, \ell_h, \ldots, \ell_{\lceil z/h \rceil}$ form the *bundle boundaries* $\partial \mathcal{B}_1$ of bundle-set $\mathcal{B}_1$.

For each bundle $B_j \in \mathcal{B}_1$, we can use the dynamic program from Lemma 16 to solve max-exposure locally. Using the exact solution for each bundle as local solution, we can use the algorithm DP-Approx (from Section 4.2) to combine them into a global solution for the bundle-set $\mathcal{B}_1$ given by $P(\mathcal{B}_1) = global(1, (k + k/h))$. We repeat this for each bundle-set $\mathcal{B}_i$ for all $i \in \{1, 2, \ldots, h\}$, and return the point set $P(\mathcal{B}_i)$ that has maximum cardinality over all $i \in \{1, 2, \ldots, h\}$.

It remains to show that this achieves a good approximation. To see this, we observe that the only ranges that may be double counted are the ones that are anchored to bundle boundaries of $\partial \mathcal{B}_i$. In the following, we show that this number is a small fraction of the optimum solution. (Proof in Appendix A.3.)

▶ **Lemma 17.** *The bundle boundaries $\partial \mathcal{B}_i, \partial \mathcal{B}_j$ for any two bundle-set $\mathcal{B}_i, \mathcal{B}_j$ are disjoint, and therefore the set of ranges anchored to lines in $\partial \mathcal{B}_i$ are also disjoint. Then, there exists a bundle-set $\mathcal{B}_{\min}$ such that the number of ranges of the optimal solution anchored to lines in $\partial \mathcal{B}_{\min}$ is at most k/h.*

Choosing $\epsilon = 1/h$ gives us a set of $(1+\epsilon)k$ objects such that the number of points exposed by selecting these objects is at least the optimum number of points.

▶ **Theorem 18.** *There exists an $(1 + \epsilon)$-approximation algorithm for max-exposure with unit square ranges running in $O(k(n + m)n^{4/\epsilon}m^{2/\epsilon})$ time.*

5   Extensions and Applications

In this section, we discuss some extensions and applications of our the results from previous section. We say that the range family $\mathcal{R}$ consists of *fat rectangles* if every range $R \in \mathcal{R}$ is a rectangle of bounded *aspect ratio*. Moreover, we say that $\mathcal{R}$ consists of *similar and fat rectangles*, if ranges in $\mathcal{R}$ are rectangles and the ratio of the largest to the smallest side in $\mathcal{R}$ is constant. We show that if $\mathcal{R}$ consists of *similar and fat* rectangles, one can achieve a constant approximation. Moreover, if $\mathcal{R}$ consists of *fat rectangles* one can achieve a bicriteria $O(\sqrt{k})$-approximation.

5.1   Approximation for Similar and Fat Rectangles

Let a, b be the length of smallest and largest sides of rectangles in $\mathcal{R}$ such that $b/a = c$ is constant. Then we can modify the input instance as follows. Replace each range $R \in \mathcal{R}$ by tiling it with at most c^2 squares of sidelength a such that the area occupied by R and its replacements are the same. Now, we have a modified set of ranges $\mathcal{R}'$ consisting of squares that have the same sidelength. Consider the optimal solution with k ranges $\mathcal{R}^*$ that exposes m^* points. It is easy to see that the set $\mathcal{R}^*$ corresponds to at most c^2k ranges in the modified instance, and therefore deleting c^2k ranges from $\mathcal{R}'$ exposes at least m^* points. Therefore, we can run the polynomial-time 2-approximation algorithm (Corollary 15) to obtain a set of at most $2c^2k$ ranges that expose at least m^* points.

▶ **Theorem 19.** *Given a set of points P, a set of rectangle ranges $\mathcal{R}$ such that the ratio of largest to smallest side in $\mathcal{R}$ is bounded by a constant, then there exists a polynomial time $O(1)$-approximation algorithm for max-exposure.*

5.2   Approximation for Fat Rectangles

We now consider the case when rectangles in $\mathcal{R}$ have bounded aspect ratio. That is for all rectangles $R \in \mathcal{R}$, the ratio of its two sides is bounded by a constant c. We transform the

input ranges $\mathcal{R}$ to obtain a modified set of ranges $\mathcal{R}'$ as follows. For each rectangle $R \in \mathcal{R}$, let x be the length of the smaller side of R. Then we replace R by at most $\lceil c \rceil$ *squares* each of sidelength x. If m^* is the optimal number of points exposed by deleting k ranges from $\mathcal{R}$, then there exists a set of $O(k)$ ranges in $\mathcal{R}'$ deleting which will expose at least m^* points. Observe that the set $\mathcal{R}'$ consists of *square* ranges, of possibly different sizes. Therefore, if we can obtain an f-approximation for square ranges, we can easily obtain $O(f)$-approximation with fat rectangles.

5.2.1   A Bicriteria $O(\sqrt{k})$-approximation for Squares

We will describe an approximation algorithm for the case when the set of ranges $\mathcal{R}$ consists of axis-aligned squares. We achieve an approximation algorithm in three steps. First, we partition the point set by assigning them to one of the input squares. Next, we solve the problem exactly for a fixed square. Finally, we combine these solutions to achieve a good approximation to the optimal solution.

We define $\mathcal{A} : P \to \mathcal{R}$ to be a function that assigns a point in P to exactly one range in $\mathcal{R}$. If $\mathcal{R}(p_i)$ is the set of squares that contain p_i, then $\mathcal{A}(p_i)$ is the smallest square in $\mathcal{R}(p_i)$. This assignment scheme ensures the following property.

▶ **Lemma 20.** *Let $R \in \mathcal{R}$ be a square and let $P(R) = \mathcal{A}^{-1}(R)$ be the set of points* assigned *to it. Moreover, let $\mathcal{R}' \subseteq \mathcal{R}$ be the set of squares that intersect R and contain at least one point in $P(R)$. Then, every square $R' \in \mathcal{R}'$ must have sidelength bigger than that of R, and therefore contains at least one corner of R.*

Now suppose we fix a square R, and consider a restricted max-exposure instance with the set of its assigned points $P(R)$. Since, ranges that contain a point in $P(R)$ are all bigger then R, this case is essentially the same as points inside a unit square, and therefore Lemma 10 can be easily extended to solve it exactly. This gives us the following algorithm. Here $1 \leq \alpha \leq k$ is a parameter.

Algorithm 3 Greedy-Squares

1. For every square $R \in \mathcal{R}$, apply Lemma 10 over the point set $P(R)$ to expose the maximum set of points $P(R, k) \subseteq P(R)$ by deleting k ranges.
2. Order squares in $\mathcal{R}$ by decreasing $|P(R, k)|$ values, and pick the set $\mathcal{S} \subseteq \mathcal{R}$ of first α squares. Return $\bigcup_{R \in \mathcal{S}} P(R, k)$ as the set of exposed points.

▶ **Lemma 21.** *Let m^* be the optimal number of points exposed using k squares, then algorithm* Greedy-Squares *computes a set of at most αk squares that expose at least $\alpha m^*/k$ points.*

For $\alpha = \sqrt{k}$, the above algorithm achieves a bicriteria $O(\sqrt{k})$-approximation. Since an f-approximation for square ranges gives an $O(f)$-approximation for fat rectangles, we obtain the following.

▶ **Theorem 22.** *Given a set of points P and a set of ranges $\mathcal{R}$ consisiting of rectangles of bounded aspect ratio, then one can obtain a bicriteria $O(\sqrt{k})$-approximation for max-exposure in polynomial time.*

6   Conclusion

In this paper, we introduced the max-exposure problem over the range space $(P, \mathcal{R})$ and presented approximation algorithms for rectangle range spaces. We showed that the problem

is hard to approximate even when $\mathcal{R}$ consists of two types of rectangles, and therefore focused on the complexity of the problem for the case when $\mathcal{R}$ consists of translates of a single rectangle. We show that in this case, the geometry of ranges can be exploited to obtain a PTAS. A natural question to consider is how does the complexity of the problem change with more general shapes. In particular, does there exist a constant approximation when $\mathcal{R}$ consists of axis-aligned squares?

References

1 P. K Agarwal and J. Pan. Near-linear algorithms for geometric hitting sets and set covers. In *Proceedings of 30th SoCG*, page 271. ACM, 2014.

2 S. Arora, D. Karger, and M. Karpinski. Polynomial time approximation schemes for dense instances of np-hard problems. *Journal of computer and system sciences*, 58(1):193–210, 1999.

3 Y. Asahiro, K. Iwama, H. Tamaki, and T. Tokuyama. Greedily finding a dense subgraph. *Journal of Algorithms*, 34(2):203–221, 2000.

4 S. Bandyapadhyay, N. Kumar, S. Suri, and K. Varadarajan. Improved approximation bounds for the minimum constraint removal problem. In *Proceedings of 21st APPROX*, pages 2:1–2:19, 2018.

5 S. Bereg and D. G. Kirkpatrick. Approximating barrier resilience in wireless sensor networks. In *Proceedings of 5th ALGOSENSORS*, pages 29–40, 2009.

6 A. Bhaskara, M. Charikar, E. Chlamtac, U. Feige, and A. Vijayaraghavan. Detecting high log-densities: an $O(n^{1/4})$ approximation for densest k-subgraph. In *Proceedings of the 42nd STOC*, pages 201–210. ACM, 2010.

7 H. Brönnimann and M. T. Goodrich. Almost optimal set covers in finite vc-dimension. *Discrete & Computational Geometry*, 14(4):463–479, 1995.

8 C. Chekuri, K. L. Clarkson, and S. Har-Peled. On the set multi-cover problem in geometric settings. *ACM Transactions on Algorithms (TALG)*, 9(1):9, 2012.

9 E. Chlamtac, M. Dinitz, C. Konrad, G. Kortsarz, and G. Rabanca. The densest k-subhypergraph problem. In *Proceedings of 19th APPROX*, pages 6:1–6:19, 2016.

10 E. Chlamtáč, M. Dinitz, and Y. Makarychev. Minimizing the union: Tight approximations for small set bipartite vertex expansion. In *Proceedings of the 28th SODA*, pages 881–899, 2017.

11 K. L. Clarkson and P. W. Shor. Application of random sampling in computational geometry, II. *Discrete & Computational Geometry*, 4:387–421, 1989.

12 M. Cygan, F. Grandoni, S. Leonardi, M. Mucha, M. Pilipczuk, and P. Sankowski. Approximation algorithms for union and intersection covering problems. In *Proceedings of 31st FSTTCS*, page 28, 2011.

13 E. Eiben, J. Gemmell, I. Kanj, and A. Youngdahl. Improved results for minimum constraint removal. In *Proceedings of 32nd AAAI Conference on Artificial Intelligence*, 2018.

14 U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

15 U. Feige, D. Peleg, and G. Kortsarz. The dense k-subgraph problem. *Algorithmica*, 29(3):410–421, 2001.

16 U. Feige and M. Seltser. On the densest k-subgraph problems. Technical report, Weizmann Institute of Science, Jerusalem, Israel, 1997.

17 R. J. Fowler, M. S. Paterson, and S. L. Tanimoto. Optimal packing and covering in the plane are np-complete. *Information processing letters*, 12(3):133–137, 1981.

18 D. S. Hochbaum and W Maass. Approximation schemes for covering and packing problems in image processing and vlsi. *Journal of the ACM (JACM)*, 32(1):130–136, 1985.

19 M. Korman, M. Löffler, R. I. Silveira, and D. Strash. On the complexity of barrier resilience for fat regions and bounded ply. *Comput. Geom.*, 72:34–51, 2018.

20 N. H. Mustafa, R. Raman, and S. Ray. Settling the apx-hardness status for geometric set cover. In *Proceedings of 55th FOCS*, pages 541–550. IEEE, 2014.

A   Missing Proofs

A.1   Proof of Lemma 3

Given a graph $G' = (V', E')$ sampled from one of the dense or random instances, we first construct a bipartite graph $G = (A, B, E)$ as follows. For every vertex $v \in V'$, we add a vertex v_a to A and v_b to B. Now for every edge $e = (u, v) \in E'$, we add the pair of edges $e_1 = (u_a, v_b)$ and $e_2 = (v_a, u_b)$ to E. That is, every edge $e \in E'$ is mapped to two copies $e_1, e_2 \in E$ and we can define $par(e_1) = par(e_2) = e$. Similarly, we define $par(u_a) = par(u_b) = u$. We say that G is dense if the underlying graph G' was sampled from the dense case, otherwise we say that G is random.

Consider a set of $k^* = 2k$ vertices in G. If G came from the dense case, there must be a set of $2k$ vertices that have $2k^{\beta+1}$ edges between them. So the number of edges in dense case $m_d^* \geq 2k^{\beta+1}$. Otherwise, we are in the random case. Consider the optimal set of $2k$ vertices V^* and let E^* be the set of edges in the induced subgraph $G[V^*]$. Now consider the corresponding set of vertices $V_p = \{par(v) \mid v \in V^*\}$ of the original graph G' and the set of edges E_p in the induced subgraph $G'[V_p]$). We have that $|V_p| \leq |V^*| = 2k$ and $|E_p| \geq |E^*|/2$ because for each edge $e = (u, v) \in E^*$, we will have the edge $par(e) = (par(u), par(v)) \in E_p$. We can now bound the number of edges E_p over $2k$ vertices in the random case to be $\tilde{O}(\max(2k, 4k^2 n^{\alpha-1}))$ w.h.p, and therefore the optimum number of edges in the random case is $m_r^* = |E^*| \leq 2|E_p| = \tilde{O}(\max(k, k^2 n^{\alpha-1}))$ w.h.p.

Choosing $k = n^{1/2}$, $\alpha = \frac{1}{2}$, $\beta = \frac{1}{2} - \epsilon$, gives us $m_r^* = \tilde{O}(n^{1/2})$ w.h.p. and $m_d^* = \tilde{\Omega}(n^{\frac{3-2\epsilon}{4}})$. Suppose, we could approximate this problem within a factor $O(n^{1/4-\epsilon})$, then in the dense case, the number of edges computed by this approximation algorithm is $\tilde{\Omega}(n^{\frac{1+\epsilon}{2}})$ which is strictly more than the maximum possible edges in the random case. Therefore, we would be able to distinguish between dense and random cases, and thereby refuting the conjecture for these values of α, β and k.

A.2   Proof of Lemma 11
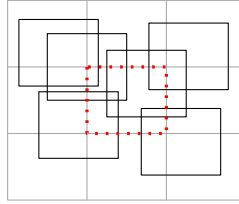
Figure 9 Embedding a max-exposure instance with unit square ranges on a unit-sized grid. Optimal solution in each grid cell can be computed exactly using Lemma 10.

Consider the optimal set of ranges $\mathcal{R}^* \subseteq \mathcal{R}$. Observe that each range $R \in \mathcal{R}^*$ intersects at most four grid cells. Let $R_i = R \cap C_i$ be the rectangular region defined by intersection of R and C_i. Clearly, there are at most four regions R_i for each $R \in \mathcal{R}^*$ and therefore $4k$ in total. At this point, the regions in cell C_i are disjoint from regions in some other cell $C_j \in \mathcal{C}$. Therefore, optimal solution exposes $|P^*|$ points over a set of cells $\mathcal{C}^*$ such that the set $\mathcal{R}^*$ has at most $4k$ disjoint components in the cells $\mathcal{C}^*$. Since we can solve the problem exactly for each cell and can combine them using the above dynamic program, we have that $global(1, 4k) \geq |P^*|$ and we achieve a 4-approximation.

For the running time, we observe that solving max-exposure locally in a cell C_i takes

$O(k(n_i + m_i)n_i^2 m_i^2)$ time, where n_i is the number of ranges that intersect C_i and m_i is the number of points in P that lie in C_i. Summed over all cells, we get the following bound.

$$\sum_i k(n_i + m_i)n_i^2 m_i^2 \leq k \sum_i (n_i + m_i) \sum_i n_i^2 \sum_i m_i^2$$

$$\leq k(n + m) \left(\sum_i n_i\right)^2 \left(\sum_i m_i\right)^2 = O(k(n+m)n^2 m^2)$$

Once the local solutions are computed, the dynamic program that merges them into a global solution has $O(k|\mathcal{C}|)$ subproblems and computing each subproblem takes $O(k)$ time. Recall that every cell in $\mathcal{C}$ contains at least one point, so $|\mathcal{C}| \leq n$ and the merge step takes an additional $O(k^2 n)$ time.

A.3   Proof of Lemma 17

Let $\mathcal{R}^* \subseteq \mathcal{R}$ be the optimal set of ranges, and let $\mathcal{R}_i^* \subseteq \mathcal{R}^*$ be the set of ranges anchored to lines in $\partial \mathcal{B}_i$. Since $\bigcup_{i \in \{1, \dots h\}} \partial \mathcal{B}_i$ is the set of all anchor lines, we have

$$\bigcup_{i \in \{1, \dots h\}} \mathcal{R}_i^* = \mathcal{R}^* \qquad \implies \qquad \sum_{i \in \{1, \dots h\}} |\mathcal{R}_i^*| = k$$

$$\implies \sum_{i \in \{1, \dots h\}} |\mathcal{R}_{\min}^*| \leq k \qquad \implies \qquad |\mathcal{R}_{\min}^*| \leq k/h$$

A.4   Proof of Lemma 21

It is easy to see that the number of squares is at most αk. To show the bound on number of points exposed, consider the optimal solution $\mathcal{R}^*$ and let the optimal set of points exposed by $\mathcal{R}^*$ to be P^*. We will now use the same assignment procedure $\mathcal{A}^* : P^* \to \mathcal{R}^*$ to assign points in P^* to a square in $\mathcal{R}^*$. That is, $\mathcal{A}^*(p_i)$ is the smallest square in $\mathcal{R}^*$ that contains p_i. We claim that $\mathcal{A}^*(p_i) = \mathcal{A}(p_i)$ for all $p_i \in P^*$ since every square that contains p_i lies in $\mathcal{R}^*$. Moreover, let $P^*(R)$ denote the set of points of P^* assigned to R.

Let m^* be the optimal number of points that are exposed, and m' be the number of points exposed by the algorithm. Now assume that the squares in $\mathcal{R}$ are ordered such that $|P(R_i, k)| \geq |P(R_j, k)|$ for all $i < j$. Then, we have the following.

$$m^* = \left| \bigcup_{R \in \mathcal{R}^*} P^*(R) \right| = \sum_{R \in \mathcal{R}^*} |P^*(R)|$$

$$\leq \sum_{1 \leq i \leq k} |P(R_i, k)| \leq \frac{k}{\alpha} \sum_{1 \leq i \leq \alpha} |P(R_i, k)|$$

$$= \frac{k}{\alpha} m'$$

B   PTAS for Unit Square Ranges on Number of Exposed Points

Given a set of points P, unit square ranges $\mathcal{R}$, we will now show that the PTAS for unit square ranges can be modified so that we can compute a set of k ranges that expose at least $(1 - \epsilon)$ fraction of the maximum possible number of points. For simplicity we assume that h is odd. The basic setup is the same: we have the anchor lines $\ell_1, \ell_2, \dots, \ell_z$ that are unit distance apart. However, there is one important change, we will only use the

odd-numbered lines $\ell_1, \ell_3, \ldots, \ell_h, \ell_{h+2}, \ldots, \ell_z$ to define bundles. For instance, the bundle B_1 now consists of the anchor lines $\ell_1, \ell_3, \ldots, \ell_h$, while the bundle-set $\mathcal{B}_1$ now comprises of bundles $B_1, B_h, B_{2h}, \ldots, B_{z/h}$. Same as before, the lines $\ell_1, \ell_h, \ldots, \ell_{z/h}$ form the boundary $\partial \mathcal{B}_1$. We have the following algorithm.

Algorithm 4 PTAS-Exposed-Points

---

1. Assign each point $p \in P$ to the closest line among $l_1, l_3, \ldots l_z$.
2. For each $i \in \{1, 3, \ldots, h\}$, process bundle set $\mathcal{B}_i$ as follows.
   - Let P_i be the set of points assigned to anchor lines $l_j \in \partial \mathcal{B}_i$, boundaries of $\mathcal{B}_i$.
   - Using the exact algorithm for each bundle $B \in \mathcal{B}_i$ as local solutions, we run the algorithm DP-Approx (from Section 4.2) over the point set $P \setminus P_i$ to obtain global solutions given by $global(1, k)$. Let $P(\mathcal{B}_i)$ be the set of exposed points returned by DP-Approx.
3. Return the set $P(\mathcal{B}_i)$ that has maximum cardinality over all $i \in \{1, 3, \ldots, h\}$.

Clearly, the number of ranges used by the above algorithm is k. It remains to show that the number of points m' exposed by the algorithm is also close to m^*, the optimal number of exposed points. Let $P^* \subseteq P$ be the optimal set of exposed points.

▶ **Lemma 23.** *The bundle boundaries $\partial \mathcal{B}_i, \partial \mathcal{B}_j$ for any two bundle-set $\mathcal{B}_i, \mathcal{B}_j$ are disjoint, and therefore the set of points assigned to lines in $\partial \mathcal{B}_i$ are also disjoint. Then, there exists a bundle-set $\mathcal{B}_{\min}$ such that the number of points of P^* assigned to its boundaries $\partial \mathcal{B}_{\min}$ is at most $\frac{2m^*}{h-1}$.*

Proof. let $P_i^* \subseteq P^*$ be the set of points in P^* that are assigned to lines in boundaries $\partial \mathcal{B}_i$ of some bundle $\mathcal{B}_i$. Since $\bigcup_{i \in \{1,3,\ldots,h\}} \partial \mathcal{B}_i$ is the set of all anchor lines to which we assign points, we have

$$\bigcup_{i \in \{1,3,\ldots h\}} P_i^* \;=\; P^* \qquad \implies \sum_{i \in \{1,3\ldots h\}} |P_i^*| \;=\; m^*$$

$$\implies \sum_{i \in \{1,3,\ldots h\}} |P_{\min}^*| \;\leq\; m^* \qquad \implies \left(\frac{h-1}{2}\right)|P_{\min}^*| \;\leq\; m^*$$

$$\implies |P_{\min}^*| \;\leq\; \frac{2m^*}{h-1}$$

◀

Observe that for the bundle-set $\mathcal{B}_{\min}$, we may have removed $P_{\min}$ points, but the remaining set $P \setminus P_{\min}$ consists at least $m^* - \frac{2m^*}{h-1} = (1 - \frac{2}{h-1})m^*$ points of the optimal set P^*. Moreover, observe that we have removed points that are within a unit distance on either side of anchor line $\ell_j \in \partial \mathcal{B}_{\min}$, the set of ranges deleted in each bundle are disjoint from another. Therefore, the value $P(\mathcal{B}_{\min})$ returned by the algorithm DP-Approx exposes at least $P \setminus P_{\min} = (1 - \frac{2}{h-1})m^*$ points by deleting k ranges. If we set $h = 2/\epsilon + 1$ we have the following result.

▶ **Theorem 24.** *There exists an $(1 - \epsilon)$-approximation on the number of exposed points for max-exposure with unit-square ranges running in $k(nm)^{O(1/\epsilon)}$ time.*