Hitting Sets when the Shallow Cell Complexity is Small *

Sander $Aarts^{1[0000-0003-1852-9116]}$

David B. Shmoys^{1[0000-0003-3882-901X]}

September 26, 2023

Abstract. The hitting set problem is a well-known NP-hard optimization problem in which, given a set of elements and a collection of subsets, the goal is to find the smallest selection of elements, such that each subset contains at least one element in the selection. Many geometric set systems enjoy improved approximation ratios, which have recently been shown to be tight with respect to the shallow cell complexity of the set system. The algorithms that exploit the cell complexity, however, tend to be involved and computationally intensive. This paper shows that a slightly improved asymptotic approximation ratio for the hitting set problem can be attained using a much simpler algorithm: solve the linear programming relaxation, take one initial random sample from the set of elements with probabilities proportional to the LP-solution, and, while there is an unhit set, take an additional sample from it proportional to the LP-solution. Our algorithm is a simple generalization of the elegant net-finder algorithm by Nabil Mustafa. To analyze this algorithm for the hitting set problem, we generalize the classic Packing Lemma, and the more recent Shallow Packing Lemma, to the setting of weighted epsilon-nets.

Keywords: Hitting set \cdot Set cover \cdot Approximation algorithms \cdot Computational geometry \cdot Shallow cell complexity \cdot Wireless coverage

1 Introduction

The input to the hitting set problem is a finite set system – a ground set X of m elements, or *points*, and a collection \mathcal{R} of n subsets, or *ranges*, of X. This can also be understood as a hypergraph, with vertices X and hyper-edges \mathcal{R} . A *hitting set* is a subset of elements $H \subseteq X$ such that every set $R \in \mathcal{R}$ is hit by H,

^{*}This material is based on work supported by the NSF under Grant CNS-1952063.

i.e. $R \cap H \neq \emptyset$, for all $R \in \mathcal{R}$. This is a vertex cover under the hypergraph view. The set system can be encoded as a set-element incidence matrix $A \in \{0, 1\}^{n \times m}$, in which the (i, j)th entry a_{ij} is 1 if range R_i contains point x_j , and 0 otherwise. The IP of the minimum hitting set problem is

$$\min_{y} \sum_{j:x_{j} \in X} y_{j}$$
s.t.
$$\sum_{j:x_{j} \in X} a_{ij}y_{j} \ge 1, \qquad \forall i: R_{i} \in \mathcal{R}; \qquad (1)$$

$$y_{j} \in \{0,1\}, \qquad \forall j: x_{j} \in X,$$

where variable $y_j \in \{0, 1\}$ indicates whether element x_j is in the solution H.

Hitting sets and set covers are intimately connected; a hitting set for A is a set cover of A^T . Both problems' decision versions are NP-complete [10]. There exists an $\mathcal{O}(\log m)$ -approximation algorithm, and this bound is tight unless P = NP [9,13]. However, there are algorithms that exploit additional structure in A to attain improved approximation ratios¹. Indeed, our work is motivated by the problem of exploiting structure when covering large numbers of wireless LoRaWAN transmitters with wireless receivers. Transmitters can be viewed as points, which are considered to be covered if they are in the line of sight of a wireless receiver, which in turn drives transmission quality in LoRaWAN [22]. The area in the line of sight of a receiver roughly resembles a simple shape.

Many geometric set systems enjoy better approximation ratios via *epsilon*nets, or ϵ -nets. A set system is said to be geometric whenever its elements can be encoded as points in Euclidean space, and sets are derived from containment of the points in geometric shapes, such as half-spaces, balls or rectangles². The seminal work of Brönnimann and Goodrich [3], and Even *et al.* [8], connects the approximability of a hitting set instance to the size of weighted ϵ -nets. Given non-negative weights on the points, $\mu : X \to \mathbb{R}_{\geq 0}$, a weighted ϵ -net with respect to weights μ is a subset $H \subseteq X$ that hits all ϵ -heavy sets:

$$\forall R \in \mathcal{R} \text{ with } \mu(R) \ge \epsilon \cdot \mu(X) : \quad R \cap H \neq \emptyset, \tag{2}$$

where the weight of any subset $S \subseteq X$ is defined as $\mu(S) = \sum_{x \in S} \mu(x)$. Even et al. [8] reduce the problem of finding a small hitting set to finding a small ϵ -net via a reformulation of the linear programming relaxation of the hitting set problem (1). The reformulated LP (3) is a program for finding the largest ϵ , and corresponding weights μ , subject to the constraint that an ϵ -net with respect to

¹ For example when A has bounded row or column sums [2,6].

² Some definition allow for uncountably many geometric shapes in \mathcal{R} , e.g. all squares. However, because the number of points X is finite, there are nevertheless a finite number of unique sets induced by these shapes.

weights μ is a hitting set.

$$\max_{\substack{\epsilon,\mu\\ \epsilon,\mu}} \epsilon$$
s.t.
$$\sum_{\substack{j:x_j \in X\\ j:x_j \in X}} a_{ij}\mu_j \ge \epsilon, \qquad \forall i: R_i \in \mathcal{R};$$

$$\sum_{\substack{j:x_j \in X\\ \mu_i \ge 0,}} \mu_j = 1; \qquad (3)$$

The first constraint requires that each set R is ϵ -heavy; the second constraint normalizes the weights. Let (ϵ^*, μ^*) denote an optimal solution to LP (3), with $\mu^* = (\mu_1^*, \ldots, \mu_n^*)$. Let z^* be the optimal value to the LP relaxation of the original program (1). The first constraint ensures that an ϵ^* -net with respect to weights μ^* is a hitting set. Moreover, the reciprocal optimal value $1/\epsilon^*$ is equal to the optimal LP value z^* [8]. In particular, an ϵ^* -net of size $g(1/\epsilon^*)$ for some function $g(\cdot)$ is a hitting set of size of $g(z^*)$. Hence, to find a small hitting set it suffices to solve LP (3) and find a small ϵ^* -net with respect to weights μ^* .

Haussler and Welzl [12] show that set systems with bounded VC-dimension admit small ϵ -nets, and develop a simple algorithm to find them. The VCdimension is a measure of the set system's complexity. Given a subset $S \subseteq X$, the projection of \mathcal{R} to S is the set system formed by elements S and sets $\mathcal{R}|_S = \{R \cap S : R \in \mathcal{R}\}$. The VC-dimension of \mathcal{R} is the size of the largest subset $S \subseteq X$ such that $\mathcal{R}|_S$ shatters S, i.e. the largest set S such that $\mathcal{R}|_S$ contains all subsets of S. In particular, Clarkson [7], and Haussler and Welzl [12], show that any set system with VC-dimension d has a weighted ϵ -net of size $\mathcal{O}\left(\frac{d}{\epsilon}\log\frac{1}{\epsilon}\right)$. This is remarkable, as the size is independent of both the size of X and \mathcal{R} . Moreover, the algorithm for finding such an ϵ -net is simple: Select a subset $H \subseteq X$ by sampling each element x in X independently.

Theorem 1 (c-net Theorem [12,14]). Let (X, \mathcal{R}) be a set system with VCdimension d, and let $\mu : X \to \mathbb{R}_{\geq 0}$ be element weights with $\mu(X) = 1$. Then for any $\epsilon, \gamma \in (0, 1)$:

$$H \leftarrow pick \ each \ x \in X \ with \ probability \ \min\left\{1, \frac{2\mu(x)}{\epsilon} \cdot \max\left\{\log \frac{1}{\gamma}, d\log \frac{1}{\epsilon}\right\}\right\}$$

is a weighted ϵ -net with respect to weights μ with probability at least $1 - \gamma$.

Throughout, we define $\mu(S) = \sum_{x \in S} \mu(x)$ for all subsets $S \subseteq X$. For general set systems of VC-dimension d, this bound is tight in expectation [14]. However, there are alternative ways to parameterize the complexity of set systems.

1.1 Shallow Cell Complexity

The *shallow cell complexity* (SCC) is a finer parameterization of the complexity of set systems. [1,4,21]. Readers are referred to Mustafa and Varadarajan [20]

for more background. A *cell* in a binary matrix A is a collection of identical rows. A cell has *depth* k if the number of 1's in any of its rows is exactly k, i.e., if each set in the cell contains k elements. For a non-decreasing function $\varphi(\cdot, \cdot)$ we say binary matrix A has *shallow cell complexity* (SCC) $\varphi(\cdot, \cdot)$ if, for all $1 \leq k \leq l \leq m$, the number of cells of depth at most k in any submatrix A^* of Aof at most l columns, is at most $\varphi(l, k)$. A set system (X, \mathcal{R}) is said to have SCC $\varphi(l, k)$ if its set-element incidence matrix A does. Often $\varphi(l, k) = \mathcal{O}(\varphi(l) k^c)$ for some constant c > 0 and single-variable function $\varphi(\cdot)$, in which case the dependence on k is can be dropped and the SCC denoted by $\varphi(l)$. Examples of geometric set systems with small shallow cell complexity are discs in the plane with $\varphi(l, k) = \mathcal{O}(k)$, and axis-parallel rectangles with $\varphi(l, k) = \mathcal{O}(lk^2)$ [18].

As is true for VC-dimension, there are algorithms that find hitting sets or ϵ -nets with sizes bounded in terms of the shallow cell complexity. A prominent example is the quasi-uniform sampling algorithm of Chan *et al.* [4]. Given non-negative weights $\mu : X \to \mathbb{R}_{\geq 0}$, and a value $\epsilon > 0$, the algorithm finds a hitting set while maintaining an upper bound on the probability of selecting any given element.

Theorem 2 (Quasi-uniform sampling [4]). Suppose a set system defined by A has $SCC \varphi(l, k) = \varphi(l) k^c$ for some c > 0. Then there is a randomized polytime algorithm that returns a hitting set of expected size $\mathcal{O}(\max\{1, \log(\varphi(m))\})$ times the LP optimum.

The algorithm attains the optimal approximation ratio with respect to the SCC³. However, the sampling procedure is involved, and may require enumeration over all sets \mathcal{R} , of which there can be $n = \Omega(m^c)$ for some constant c > 0 [17].

Taking a different approach, Mustafa and colleagues [16,17,19] develop a net-finder for asymptotically optimal-sized unweighted ϵ -nets with respect to the SCC. The algorithm is remarkably simple: Take an initial sample from X, and while there are unhit sets, choose an unhit set arbitrarily, and add $\mathcal{O}(1)$ randomly chosen elements from this set to the original sample. The algorithm assumes access to an oracle that returns an unhit set. This oracle is called at most $\mathcal{O}(1/\epsilon)$ times in expectation. While the size of the returned ϵ -net is asymptotically on par with the quasi-uniform sampling algorithm, there are large constants in the upper bound [17].

This algorithm is not directly applicable to the hitting set problem via the LP-reduction above, although it can be used via a standard reduction. The analysis of the algorithm applies to only uniform weights, and the optimal weights μ^* of the LP-formulation (3) are not generally uniform. Nevertheless, it is possible to reduce the problem of finding a weighted ϵ -net to that of finding a uniform ϵ' -net following a standard reduction, in which an expanded instance is generated by copying each element $x_j \in X$ a number of times roughly proportional to its weight $\mu^*(x_j)$ [3,4]. This can generate $\Omega(m)$ copies of each element, which can

³ In addition, it is worth noting that this algorithm can solve the more general *weighted* hitting set problem, in which each element has a given weight, and the goal is to find the minimum weight hitting set.

have notable consequences. First, to achieve a weighted ϵ^* -net in the original instance, one must use a smaller value ϵ' for the expanded instance, on the order of $\mathcal{O}(\epsilon^*/m)$. This results in an approximation ratio of $\mathcal{O}(\log \varphi(\mathcal{O}(m)))$. Secondly, generating copies can increase the number of elements from m to $\Omega(m^2)$. This can increase the runtime considerably. In particular, repeatedly sampling from sets of size $\Theta(m^2)$ can become prohibitive on large instances such as the wireless coverage problem motivating our work.

1.2 Our Contributions

This paper generalizes the elegant net-finder algorithm of Mustafa [17] to the setting of weighted ϵ -nets, in order to produce a fast and simple algorithm for the hitting set problem, which attains asymptotically optimal approximation ratios with respect to the shallow cell complexity. The algorithm enjoys a faster runtime that makes solving larger instances, such as LoRaWAN receiver placement at scale, feasible. This is achieved by combining the weighted ϵ -net finder with the reduction of Even *et al.* [8]. In doing so, we also improve on the asymptotic approximation ratio from max{1,log $\varphi(m)$ } to max{1, $\mathcal{O}(\log \varphi(\mathcal{O}(z^*)))$ } where z^* is the optimal value to the linear relaxation of the hitting set program (1). While in the worst case $z^* = m$, it is often the case that $z^* \ll m$. However, the multiplicative constants in our analysis are relatively large, matching those of Mustafa [17]. In addition to the algorithm, our analysis generalizes the classic Packing Lemma of Haussler [11], as well as the Shallow Packing Lemma of Mustafa *et al.* [19], to the weighted setting, which may be of independent interest.

Key to our approach are adaptations of Mustafa's [19] Shallow Packing Lemma and Haussler's [11] classic Packing Lemma that accommodate nonuniform weights. Our main technical contribution is to allow a notion of weighted packings. Consider any non-negative weights $\mu: X \to \mathbb{R}_{\geq 0}$ with $\sum_{x \in X} \mu(X) = 1$, and extend it to element subsets via $\mu(S) = \sum_{x \in S} \mu(S)$.⁴ A (k, δ) -packing with respect to weights μ is a collection of sets $\mathcal{P} \subseteq \mathcal{R}$ in which (i) all sets R in \mathcal{P} are at most k-heavy, i.e., have bounded weight $\mu(R) \leq k$; and (ii) all pairs of sets have symmetric differences of weight at least δ . (See Definition 1). Our weighted shallow packing lemma upper bounds the number of sets in \mathcal{P} as a function of the SCC. Our approach accommodates weights μ by sampling elements from a distribution with probability mass proportional to the weights, rather than from a uniform distribution as in the original proofs. Moreover, our proof uses sampling with replacement rather than without replacement to simplify the analysis. While more generally applicable, our result yields the same bound on the size of \mathcal{P} as in the unweighted setting. An analogous sampling approach is used in proving Theorem 1 [14]. Equipped with our generalized lemma, it is straightforward to adapt Mustafa's [17] analysis to a weighted net-finder. A proof of our Weighted Packing Lemma is included in the extended online version.

⁴ Any non-negative weights $w : X \to \mathbb{R}_{\geq 0}$ with w(X) > 0 can be normalized as $\mu(x) = w(x)/w(X)$.

2 Algorithm and Main Result

Our algorithm combines the LP-relaxation of Even *et al.* [8] with the generalized sampling approach of Mustafa [17]. Our procedure is summarized in Algorithm 1. The algorithm makes use of two global constants, β and γ . These are assumed to be positive, and to satisfy $\gamma \leq 1/4$ and $\beta + \gamma \leq 1$.

Algorithm	1:	А	simple	hitting	set	algorithm	with	details
-----------	----	---	--------	---------	----------------------	-----------	------	---------

Data: A matrix A with VC-dim(A) $\leq d$ and SCC $\varphi(\cdot, \cdot)$, constants $\gamma, \beta > 0$ $\epsilon^*, (\mu_1^*, \dots, \mu_n^*) \leftarrow$ solve LP {max $\epsilon : A\mu \geq \epsilon, \mu^T \mathbf{1} = 1, \mu \geq \mathbf{0}$ }; $H \leftarrow \emptyset$; **for** $x_j \in X$ **do** $H \leftarrow H \cup \{x_j\}$ with probability $\min \left\{ 1, \frac{2\mu_j^*}{(\frac{3}{4} - \frac{\beta}{2}) \epsilon^*} \cdot \max \left\{ \log \left(d^2 \varphi \left(\frac{8d}{\beta \epsilon^*}, \frac{48d}{\beta} \right)^2 \right), d \log \left(\frac{1}{(\frac{3}{4} - \frac{\beta}{2}) \epsilon^*} \right) \right\} \right\}$ **end while** there is a set $R \in \mathcal{R}$ not hit by H **do** Independently add each $x_j \in R$ to H with probability $\min \left\{ 1, \frac{2\mu_j^*}{\gamma \mu^*(R)} \max\{\log 2, d \log \frac{1}{\gamma}\} \right\}$ **end return** H

In the while loop, the weights $\mu^*(R) = \sum_{j:x_j \in R} \mu_j^*$ denote the weight of set R under the LP optimal weights $\mu^* = (\mu_1^*, \dots, \mu_n^*)$.

Conceptually, the algorithm is simple; it randomly selects an initial set of elements H from X, and proceeds to add additional random subsets of elements to H until this is a hitting set. The algorithm relies on an oracle that returns an arbitrary unhit set. This oracle is treated as a black box. Our main result is twofold: we bound the expected size of the solution hitting set H as a function of the cell complexity, and bound the expected number of oracle calls.

Theorem 3. Let A be a binary matrix encoding a hitting set instance with shallow cell complexity $\varphi(\cdot, \cdot)$ and VC-dim $(\mathcal{R}) \leq d$. Let z^* be LP optimal value. Then the algorithm returns a hitting set of expected size

 $\mathcal{O}\left(z^* \cdot \max\left\{1, \log \varphi\left(\mathcal{O}\left(z^*\right), \mathcal{O}\left(d\right)\right)\right\}\right).$

Furthermore it makes $\mathcal{O}(z^*)$ oracle calls in expectation.

Note that the algorithm always returns a hitting set; the randomness is in the size of the solution and the runtime. This is in contrast with the net-finder in Theorem 1. Both algorithms require knowing the VC dimension d; ours must additionally know the shallow cell complexity $\varphi(\cdot, \cdot)$. If unknown, these can be searched for using a standard doubling trick [17].

6

3 The Weighted Shallow Packing Lemma

The Weighted Shallow Packing Lemma is key to proving Theorem 3. This section formally defines weighted shallow packings, states the lemma, and proves it. To this end, fix non-negative weights μ over X, and define the weight of a subset of elements $S \subseteq X$ as $\mu(S) = \sum_{j \in S} \mu_j$. Assume that $\mu(X) = 1$. To contrast, let card(S) denote the cardinality of any set S. Note that the weights μ induce a probability distribution over the elements X. Throughout, whenever an element u of X is randomly sampled, it is assumed to follow a distribution proportional to $\mu(\cdot)$, in which case we say u is sampled from $\mu(\cdot)$, and denote this by $u \sim \mu(\cdot)$. Note that an element $u \sim \mu(\cdot)$ sampled this way lies in subset $S \subseteq X$ with probability $\mu(S)$.

The main purpose of the weighted shallow packing lemma is to bound the number of sets in a set system in terms of its shallow cell complexity. Clearly, an arbitrary set systems can contain large numbers of sets. Instead, we focus on a particular kind of set system called a *weighted packing*. A set system is a packing if all its sets are "light", and each pair of sets are sufficiently different from each other. Critically, we define "light" and "different" in reference to the weights.

Definition 1. Let (X, \mathcal{P}) be a set system with weights μ , and let $k, \delta \in (0, 1)$ be constants. If all sets S in \mathcal{P} satisfy $\mu(S) \leq k$, and all pairs of distinct sets S, R in \mathcal{P} have symmetric difference of weight at least δ , i.e.

$$\mu\left(\Delta(S,R)\right) = \mu\left(\left(S\backslash R\right) \cup \left(R\backslash S\right)\right) \ge \delta,\tag{4}$$

then we say (X, \mathcal{P}) is a weighted (k, δ) -packing with respect to μ .

We omit the "with respect to μ "-statement whenever this is clear from context. The shallow packing lemma bounds the number of sets in a packing as a func-

tion of the constants (k, δ) , the VC-dimension, and the shallow cell complexity.

Lemma 1 (Weighted shallow packing lemma). Let (X, \mathcal{P}) be a set system on *m* elements, equipped with weights μ , and let (X, \mathcal{P}) be a (k, δ) -packing with respect to μ for constants $k, \delta > 0$. Assume the set system has VC-dim $(\mathcal{P}) \leq d$, and shallow cell complexity $\varphi(\cdot, \cdot)$. Then

$$\operatorname{card}(\mathcal{P}) \leq \frac{24d}{\delta} \cdot \varphi\left(\frac{8d}{\delta}, \frac{48dk}{\delta}\right).$$

The proof to this lemma makes use of our weighted Packing Lemma. The unweighted Packing Lemma is a classic result by Haussler [11] that bounds the number of sets in a packing. We generalize this to nonuniform weights.

Lemma 2 (Weighted packing lemma). Let (X, \mathcal{P}) be a set system with n sets and m elements, equipped with weights μ . Let $\text{VC-dim}(\mathcal{P}) \leq d$ for some integer $d \geq 1$ and assume there is a constant $\delta \in (0, 1)$ such that $\mu(\Delta(S_i, S_k)) \geq \delta$ for all $1 \leq i < k \leq n$. Then

$$\operatorname{card}(\mathcal{P}) \leq 2\mathbb{E}\left[\operatorname{card}(\mathcal{P}|_{Y})\right],$$

where Y is the set of unique elements in a random sample $U = (u_1, u_2, \ldots, u_s)$ of size $s = \lceil \frac{8d}{\delta} \rceil - 1$, in which each element u_k is sampled iid $u_k \sim \mu(\cdot)$ with replacement.

The proof of the latter lemma is in the appendix to the extended online version of the paper; Lemma 1 is proved next.

3.1 Proof of the Weighted Shallow Packing Lemma

Proof. Fix a (k, δ) -packing \mathcal{P} and let $U = (u_1, u_2, \ldots, u_s)$ be a random sample of length s, in which each element is sampled $u_k \sim \mu(\cdot), k = 1, \ldots, s$, independently and with replacement. The number of elements sampled is $s = \lceil \frac{8d}{\delta} \rceil - 1$. Let $Y \subseteq X$ be the set of unique elements in U. For every set $R \in \mathcal{P}$, let M(R, U) := $\sum_{i=k}^{s} \mathbf{1}[u_k \in R]$ denote number of (copies of) elements in U that are in R. Define $\mathcal{P}_L \subseteq \mathcal{P}$ as the sub-collection of "large" sets in packing \mathcal{P} that contain at least $6 \left(\frac{8dk}{\delta} \right)$ (copies of) elements in the random sample U:

$$\mathcal{P}_L = \left\{ R \in \mathcal{P} : M(R, U) \ge 6 \cdot \frac{8dk}{\delta} \right\}.$$

It follows that the probability of a given range R in \mathcal{P} being a member of \mathcal{P}_L is

$$\mathbb{P}[R \in \mathcal{P}_L] = \mathbb{P}\left[M(R, U) \ge 6 \cdot \frac{8dk}{\delta}\right].$$

Our goal is to show that the collection of large sets \mathcal{P}_L has few members in expectation. To do so, it suffices to bound the probability that a fixed set R is a member of \mathcal{P}_L . This is achieved using Markov's inequality. Recalling that all sets $R \in \mathcal{P}$ have bounded weight $\mu(R) \leq k$ gives

$$\mathbb{E}[M(R,U)] = \sum_{k=1}^{s} \mathbb{P}[u_k \in R] = \sum_{k=1}^{s} \mu(R) \le s \cdot k \le \frac{8dk}{\delta},$$

where we used the fact that we sample from $\mu(\cdot)$, which implies that $\mathbb{P}[u_k \in R] = \mu(R)$. Now, Markov's inequality bounds the probability of R being in \mathcal{P}_L :

$$\mathbb{P}[R \in \mathcal{P}_L] = \mathbb{P}\left[M(R, U) \ge 6 \cdot \frac{8dk}{\delta}\right]$$
$$\le \mathbb{P}\left[M(R, U) \ge 6 \cdot \mathbb{E}[M(R, U)]\right] \le 1/6.$$

Finally, because $\mathcal{P}_L \subseteq \mathcal{P}$, we conclude that

$$\mathbb{E}[\operatorname{card}(\mathcal{P}|_{Y})] \leq \mathbb{E}[\operatorname{card}(\mathcal{P}_{L})] + \mathbb{E}[\operatorname{card}((\mathcal{P}\setminus\mathcal{P}_{L})|_{Y})]$$

$$\leq \sum_{R\in\mathcal{P}} \mathbb{P}[R\in\mathcal{P}_{L}] + \operatorname{card}(Y) \cdot \varphi\left(\operatorname{card}(Y), 6 \cdot \frac{8dk}{\delta}\right)$$

$$\leq \frac{1}{6}\operatorname{card}(\mathcal{P}) + \frac{8d}{\delta} \cdot \varphi\left(\frac{8d}{\delta}, \frac{48dk}{\delta}\right),$$

where the second-to-last inequality uses the shallow cell complexity of \mathcal{P} ; the system $(Y, (\mathcal{P} \setminus \mathcal{P}_L)|_Y)$ has at most $\operatorname{card}(Y) \leq s$ elements, and sets have depth at most $\left(6 \cdot \frac{8dk}{\delta}\right)$, as the system consists only of cells that are not "large". The final inequality holds because $\mathbb{P}[R \in \mathcal{P}_L] \leq 1/6$. Finally, applying Lemma 2 completes the proof. \Box

4 Proof of the Main Theorem

Equipped with the Weighted Shallow Packing Lemma, we follow a similar strategy as Mustafa [17]. We state and prove three key lemmas, and finally prove Theorem 3.

4.1 Key lemmas

The proof of our main theorem relies on all sets having similar weight. Let ϵ and $\mu = (\mu_1, \ldots, \mu_n)$ be a feasible solution to the LP relaxation (3). By the constraints of the LP, each set $R \in \mathcal{R}$ has weight $\mu(R) = \sum_{j:x_j \in R} \mu_j \geq \epsilon$. Partition the collection of sets \mathcal{R} into groups $\ell = 0, 1, \ldots, \lceil \log \epsilon \rceil$ of sets of similar weight; set R belongs to group ℓ if and only if $2^{-\ell-1}\epsilon \leq \mu(R) < 2^{-\ell}\epsilon$. Because the algorithm exclusively takes independent samples, we can view one run of the algorithm as multiple parallel, independent runs on each group of sets. All our bounds scale on the order $\mathcal{O}(1/(2^{-\ell}\epsilon))$, so summing over the groups gives a final bound on the order of $\mathcal{O}(1/\epsilon)$. Hence, we assume henceforth that all sets $R \in \mathcal{R}$ have weight $\epsilon \leq \mu(R) \leq 2\epsilon$.

The key idea of the proof is to amortize the elements added from each processed unhit set throughout the run of the algorithm. We say a set is processed each time it is flagged as unhit by the oracle, and a sample is taken from it. We bound the total number of elements sampled using weighted (k, δ) -packings on two levels. The first-level packing is an arbitrary maximal packing \mathcal{P} of sets in \mathcal{R} . There are a bounded number of sets in \mathcal{P} . Next, each processed set R_i is assigned to a set in the first-level packing \mathcal{P} . For a fixed set P^j in the first-level packing, given that it has been assigned processed sets, we show that the collection of sets R_i assigned to P^j forms a second-level packing. Each second-level packing also has a bounded number of sets. Finally, by bounding the probability that a set in the first-level packing has any sets assigned to it, the total expected number of times the algorithm processes a set is bounded. Note that the assignments are only a tool for analysis; they need not be computed by the algorithm.

We begin by defining the first-level packing. Fix a maximal $(2\epsilon, \beta\epsilon)$ -packing $\mathcal{P} = \{P^1, \ldots, P^p\}$, where p denotes the number of sets in the packing. The Shallow Packing Lemma 1 upper bounds the number of sets in the packing by

$$p \le \frac{24d}{\beta\epsilon} \cdot \varphi\left(\frac{8d}{\beta\epsilon}, \frac{96d}{\beta}\right). \tag{5}$$

Now, suppose the algorithm runs for T steps, processing sets (R_1, \ldots, R_T) in sequence. One given set may be processed multiple times. Denote the sets of

sampled elements H_{R_1}, \ldots, H_{R_T} . The processed sets R_i are assigned to sets P^j in the first-level packing \mathcal{P} as follows. Arbitrarily assign each set R_i to any index $j \in \{1, 2, \ldots, p\}$ satisfying $\mu \left(\Delta(R_i, P^j) \right) < \beta \epsilon$. Such an index j exists because \mathcal{P} is a maximal $(2\epsilon, \beta\epsilon)$ -packing. It may be the case that $R_i = P^j$. The next task is to bound the number of sets R_i assigned to any set P^j in the first-level packing.

Let n_j denote the number of processed sets in (R_1, \ldots, R_T) assigned to $P^j \in \mathcal{P}$. For now, condition on first-level packing set P^j having at least one set assigned to it, i.e. $n_j \geq 1$. We study the probability of this event later. Relabel the sets and consider them in the order in which they were processed by the algorithm,

$$\mathcal{S}^j = (R_1^j, \dots, R_{n_i}^j).$$

Claim. For all $j \in \{1, 2, ..., p\}, i \in \{1, 2, ..., n_j\}$ we have

$$\mu\left(P^{j} \cap R_{i}^{j}\right) > \frac{\mu\left(P^{j}\right) + \mu\left(R_{i}^{j}\right) - \beta\epsilon}{2}.$$
(6)

Proof. Fix $j \in \{1, 2, ..., p\}$. For all $i \in \{1, 2, ..., n_j\}$ we have

$$\mu\left(P^{j}\right) + \mu\left(R_{i}^{j}\right) = \mu\left(P^{j}\backslash R_{i}^{j}\right) + \mu\left(R_{i}^{j}\backslash P^{j}\right) + 2\mu\left(P^{j}\cap R_{i}^{j}\right)$$
$$= \mu\left(\Delta(P^{j}, R_{i}^{j})\right) + 2\mu\left(P^{j}\cap R_{i}^{j}\right) < \beta\epsilon + 2\mu\left(P^{j}\cap R_{i}^{j}\right).$$

The first equality follows from straightforward accounting, and the second from the definition of symmetric difference. The inequality follows from the manner in which set R_i^j is matched to the packing set P^j . Finally, a simple rearrangement of terms yields the result.

This proves that the intersection of each set R_i^j with its corresponding first-level packing set P^j is heavy. This lets us define a second-level packing using the intersections $R_i^j \cap P^j$.

Rather than directly bounding the the number of processed sets assigned to a first-level packing set, it is easier to first bound the length of a random subsequence of the assigned sets S^j . For any $j \in \{1, \ldots, p\}$, define the subsequence S'^j as the subsequence of processed sets R in S^j whose corresponding samples H_R form γ -nets for the system $(R, \mathcal{R}|_R)$:

$$\mathcal{S}^{\prime j} = \left(R \in \mathcal{S}^j : H_R \text{ is a } \gamma \text{-net for } (R, \mathcal{R}|_R) \right).$$

We proceed by bounding the length of the subsequence $\mathcal{S}^{\prime j}$, and by choosing γ so as to make it likely for a set R in \mathcal{S}^{j} to be in $\mathcal{S}^{\prime j}$, using the ϵ -net Theorem 1. We use this to upper bound the expected number of sets in \mathcal{S}^{j} . Let len(\mathcal{S}) denote the length of a sequence \mathcal{S} .

The following claim bounds the length of the subsequence above.

Claim. For any $j \in \{1, 2, ..., p\}$:

$$\operatorname{len}(\mathcal{S}'^{j}) \leq \begin{cases} \frac{24d}{3/2 - \beta - \gamma} \cdot \varphi\left(\frac{8d}{3/2 - \beta - \gamma}, \frac{48d}{3/2 - \beta - \gamma}\right), & \text{if } \beta + \gamma \ge 1/2; \\ \mathcal{O}(1), & \text{otherwise.} \end{cases}$$
(7)

Proof. Let $n'_j = \operatorname{len}(\mathcal{S}'^j)$ and relabel the sets so that $\mathcal{S}'^j = \left(R_1^j, \ldots, R_{n'_j}^j\right)$. Now consider an auxiliary sequence of sets based on intersecting the entries R_i^j in \mathcal{S}'^j with P^j :

$$\mathcal{T}^{\prime j} = \left(S_1^j, \dots, S_{n'_j}^j\right) \quad \text{with} \quad S_i^j = R_i^j \cap P^j \text{ for each } i \in \{1, \dots, n'_i\}.$$

This sequence of sets is used to generate a second-level packing. To do this, consider two distinct set-indices $1 \leq k < l \leq n'_j$. The points $H_{R_k^j}$ are added before set R_l^j is considered, so $H_{R_k^j}$ is a γ -net for $\left(R_k^j, \mathcal{R}|_{R_k^j}\right)$, whereas the set R_l^j – because it is subsequently considered by the algorithm – is not hit by this net. This implies that the intersection of R_k^j and R_l^j is of bounded weight, as it would be hit by the γ -net otherwise:

$$\mu\left(R_k^j \cap R_l^j\right) < \gamma \cdot \mu\left(R_k^j\right).$$

This implies that the weight of the intersection of S_k^j and S_l^j is bounded above:

$$\mu\left(S_{k}^{j}\cap S_{l}^{j}\right) = \mu\left(R_{k}^{j}\cap R_{l}^{j}\cap P^{j}\right) \leq \mu\left(R_{k}^{j}\cap R_{l}^{j}\right) < \gamma \cdot \mu\left(R_{k}^{j}\right).$$
(8)

The fact that sets in $\mathcal{T}^{\prime j}$ have pairwise intersections of small weight implies that their symmetric differences are heavy:

$$\begin{split} \mu\left(\Delta(S_k^j, S_l^j)\right) &= \mu\left(S_k^j\right) + \mu\left(S_l^j\right) - 2\mu\left(S_k^j \cap S_l^j\right) \\ &= \mu\left(R_k^j \cap P^j\right) + \mu\left(R_l^j \cap P^j\right) - 2 \cdot \mu\left(S_k^j \cap S_l^j\right) \\ &> \frac{\mu\left(P^j\right) + \mu\left(R_k^j\right) - \beta\epsilon}{2} + \frac{\mu\left(P^j\right) + \mu\left(R_l^j\right) - \beta\epsilon}{2} - 2 \cdot \mu\left(S_k^j \cap S_l^j\right) \\ &> \frac{\mu\left(P^j\right) + \mu\left(R_k^j\right) - \beta\epsilon}{2} + \frac{\mu\left(P^j\right) + \mu\left(R_l^j\right) - \beta\epsilon}{2} - 2\gamma \cdot \mu\left(R_k^j\right) \\ &= \mu\left(P^j\right) - \beta\epsilon + \frac{1}{2}\mu\left(R_l^j\right) + (1/2 - 2\gamma)\mu\left(R_k^j\right) \\ &\geq (3/2 - \beta - \gamma) \cdot \mu\left(P^j\right), \end{split}$$

where the first inequality uses Eq. (6), the second Eq. (8), and the last exploits the fact that sets R_k^j , R_l^j , and P^j are each of measure at least ϵ and at most 2ϵ , and that $\gamma \leq 1/4$. Thus, depending on the constants, the sequence \mathcal{T}'^j may form a weighted packing.

Finally, reviewing two cases for the constants β and γ makes the above more precise. First, if $\beta + \gamma < 1/2$, the inequality above implies that the symmetric difference of S_k^j and S_l^j is strictly larger than $\mu(P^j)$. This cannot be the case as both sets are subsets of P^j . Thus, the only sequence \mathcal{S}'^j for which $\beta + \gamma$ can be less than a half is if there are no two unique indices, implying that $\operatorname{len}(\mathcal{S}'^j) \leq 1$. Secondly, if $\beta + \gamma \geq 1/2$, the sets in $\mathcal{T}^{\prime j}$ form a $(\mu(P^j), (3/2 - \beta - \gamma)\mu(P^j))$ packing over P^j ; all sets have measure at most $\mu(P^j)$, and every symmetric difference is at least $(3/2 - \beta - \gamma)\mu(P^j)$. This is our second-level packing. Now, the Shallow Packing Lemma 1 implies:

$$\operatorname{len}(\mathcal{S}'^{j}) = \operatorname{len}(\mathcal{T}'^{j}) \leq \frac{24d}{3/2 - \beta - \gamma} \cdot \varphi\left(\frac{8d}{3/2 - \beta - \gamma}, \frac{48d}{3/2 - \beta - \gamma}\right),$$

where we have used the fact that $\varphi(\cdot, \cdot)$ is non-decreasing and that $\mu(P^j) \leq 1$. \Box

We can now bound the length of the full sequence of sets assigned to the packing set P^j . Taking expectations sidesteps any dependencies in the sequences. For instance, a set R can only be in S^j if previous samples failed to hit it. However, for each fixed set $R \in \mathcal{R}$, the probability of the sampled points H_S forming a γ -net for $(R, R|_{\mathcal{R}})$ is independent of previous sampling. Indeed, by Theorem 1, the probability that H_R is a γ -net is at least $1 - \gamma \geq 1/2$.

Lemma 3 (Mustafa, Lemma 5 [17]).

$$\mathbb{E}\left[\ln(\mathcal{S}^{j}) \mid n_{j} \geq 1\right] \leq \frac{48}{3/2 - \beta - \gamma} \cdot \varphi\left(\frac{8d}{3/2 - \beta - \gamma}, \frac{48d}{3/2 - \beta - \gamma}\right)$$

Proof. We use a simple application of linearity of expectation, and Theorem 1:

$$\mathbb{E}[\operatorname{len}(\mathcal{S}'^{j}) | n_{j} \geq 1] = \sum_{R \in \mathcal{S}^{j}} \mathbb{P}[H_{R} \text{ is a } \gamma \text{-net for } (R, \mathcal{R}|_{R})] \geq \frac{1}{2} \cdot \operatorname{len}(\mathcal{S}^{j})$$

where we drop the conditioning on $n_j \geq 1$ because the event that a particular sample H_R is a γ -net is independent of the number of previous samples. On the other hand, Eq. (7) upper bounds the size of $\operatorname{len}(\mathcal{S}'^j)$. Piecing these together yields the inequality:

$$\frac{1}{2}\operatorname{len}(\mathcal{S}^{j}) \leq \mathbb{E}\left[\operatorname{len}(\mathcal{S}^{\prime j}) \mid n_{j} \geq 1\right] \leq \operatorname{len}(\mathcal{S}^{\prime j}) \leq \frac{24d}{3/2 - \beta - \gamma}\varphi\left(\frac{8d}{3/2 - \beta - \gamma}, \frac{48d}{3/2 - \beta - \gamma}\right)$$

Thus far we have conditioned on a set in the first-level packing being assigned at least one processed set. We now bound the probability of this being the case. Later, this probability is used to compute the expected number of processed sets assigned to a first-level packing set.

Lemma 4. Let H_0 be the initial sample taken by the algorithm. Then for any $j \in \{1, \ldots, p\}$:

$$\mathbb{P}[n_j \ge 1] = \mathcal{O}\left(\frac{1}{d^2\varphi\left(\frac{8d}{\beta\epsilon}, \frac{48d}{\beta}\right)^2}\right)$$

Proof. Fix an index $j \in \{1, \ldots, p\}$. Suppose that $n_j \ge 1$. By Eq. (6), for any $i \in \{1, \ldots, n_j\}$:

$$\mu\left(P^{j} \cap R_{i}^{j}\right) > \frac{\mu\left(P^{j}\right) + \mu\left(R_{i}^{j}\right) - \beta\epsilon}{2}$$
$$\geq \frac{\mu\left(P^{j}\right) + \mu\left(P^{j}\right)/2 - \beta\mu\left(P^{j}\right)}{2} = \left(\frac{3}{4} - \frac{\beta}{2}\right) \cdot \mu\left(P^{j}\right)$$

The second inequality above follows from the assumption that all sets have weights within a factor 2 of each other. The above implies that, if H_0 is a $\left(\frac{3}{4} - \frac{\beta}{2}\right)$ -net for $\left(P^j, \mathcal{R}|_{P^j}\right)$, then any $R \in S^j$ would be hit by H_0 . In other words, $n_j \geq 1$ only if H_0 is not a $\left(\frac{3}{4} - \frac{\beta}{2}\right)$ -net for $\left(P^j, \mathcal{R}|_{P^j}\right)$:

$$\mathbb{P}[n_j \ge 1] \le \mathbb{P}\left[H_0 \text{ is not a } \left(\frac{3}{4} - \frac{\beta}{2}\right) \text{-net for } (P^j, \mathcal{R}|_{P^j})\right].$$

Beacuse $\frac{\mu_j}{\epsilon} \geq \frac{\mu_j}{\mu(R)}$, the initial sample includes each element with sufficient probability to apply Theorem 1 to the RHS above, completing the proof. \Box

4.2 Proof of Theorem 3

Proof of Theorem 3. At this stage, the analysis closely follows Mustafa's [17]. Clearly, the algorithm proceeds until H is an ϵ^* -net with respect to measure μ^* , i.e., a hitting set. It suffices to bound the expected size of the hitting set H, as well as the expected number of oracle calls. These quantities are related, since the number of points added depends on the number of times a set is processed.

First, consider the expected size of the hitting set. There are two contributions to the set: the initial sample H_0 , and the samples from the processed sets H_{R_1}, \ldots, H_{R_T} . We bound the expected size of the initial sample first.

Claim. The expected size of the initial sample, $\mathbb{E}[\operatorname{card}(H_0)]$ is bounded by

$$\mathcal{O}\left(\frac{1}{\left(\frac{3}{2}-\frac{\beta}{2}\right)\epsilon}\max\left\{\log\left(d\varphi\left(\frac{8d}{\beta\epsilon},\frac{48d}{\beta}\right)\right),\frac{d}{\left(\frac{3}{2}-\frac{\beta}{2}\right)}\log\frac{1}{\left(\frac{3}{2}-\frac{\beta}{2}\right)\epsilon}\right\}\right).$$
 (9)

This follows by summing the probability of sampling x for each $x \in X$. An analogous result is used for the number of points added during the processing of a set $R \in \mathcal{R}$, provided it is processed:

Claim. For any fixed set $R \in \mathcal{R}$, conditional on being processed, the expected number of points added each time it is processed is

$$\mathbb{E}\left[\operatorname{card}(H_R)\right] \le 2\left(\frac{\log 2}{\gamma} + \frac{d}{\gamma}\log\frac{1}{\gamma}\right) = \mathcal{O}\left(1\right).$$
(10)

This bound applies irrespective of whether or not a set was processed previously.

The number of points added during processing, and the number of oracle calls, can be bounded together. Recalling that R_1, \ldots, R_T are the processed sets, and using the claim above, the number of added elements is at most

$$\mathbb{E}\left[\sum_{i=1}^{T} \operatorname{card}(H_{R_i})\right] \le \mathbb{E}\left[\sum_{i=1}^{T} 2\left(\frac{\log 2}{\gamma} + \frac{d}{\gamma}\log\frac{1}{\gamma}\right)\right] = \mathbb{E}[T] \cdot 2\left(\frac{\log 2}{\gamma} + \frac{d}{\gamma}\log\frac{1}{\gamma}\right).$$
(11)

Thus, it suffices to bound the expected number of oracle calls $\mathbb{E}[T]$. This is where we employ both the first-, and second-level packings. In particular

$$\mathbb{E}[T] = \mathbb{E}\left[\sum_{j=1}^{p} \operatorname{len}(\mathcal{S}^{j})\right] = \underbrace{\sum_{j=1}^{p}}_{(\mathrm{i})} \cdot \underbrace{\mathbb{E}[\operatorname{len}(\mathcal{S}^{j}) \mid n_{j} \ge 1]}_{(\mathrm{i}\mathrm{i})} \cdot \underbrace{\mathbb{P}[n_{j} \ge 1]}_{(\mathrm{i}\mathrm{i}\mathrm{i})}.$$
 (12)

The terms (i), (ii) and (iii) are bounded using Eq. (5), Lemma 3, and Lemma 4, respectively. In addition, using $\frac{3}{2} - \beta - \gamma \ge \frac{1}{2} \ge \max\{\beta\epsilon, \beta/2\}$:

(i)
$$p \leq \frac{24d}{\beta\epsilon}\varphi\left(\frac{8d}{\beta\epsilon},\frac{24d}{\beta}\right)$$

(ii) $\mathbb{E}\left[\operatorname{len}(\mathcal{S}^{j}) \mid n_{j} \geq 1\right] \leq \frac{48d}{3/2-\beta-\gamma}\varphi\left(\frac{8d}{3/2-\beta-\gamma},\frac{24d}{3/2-\beta-\gamma}\right) \leq \frac{48d}{3/2-\beta-\gamma}\varphi\left(\frac{8d}{\beta\epsilon},\frac{24d}{\beta}\right);$
(iii) $\mathbb{P}[n_{j} \geq 1] \leq \left(d^{2}\varphi\left(\frac{8d}{\beta\epsilon},\frac{24d}{\beta}\right)^{2}\right)^{-1}.$

Combining the right-hand-side terms, we obtain the bound

$$\mathbb{E}[T] \le \frac{24 \cdot 48}{\beta \cdot (3/2 - \beta - \gamma)} \frac{1}{\epsilon} = \mathcal{O}\left(\frac{1}{\epsilon}\right).$$
(13)

This is minimized by choosing a small γ , e.g. $\gamma = 1/100$, and setting $\beta = 3/4$.

Finally, summing over the ℓ groups of sets, and adding the expected number of initial samples to the expected number added points completes the proof. Note that Eq. (13) also bounds the expected number of oracle calls made during the run of the algorithm.

References

- Aronov, B., Ezra, E., Sharir, M.: Small-size ε-nets for axis-parallel rectangles and boxes. In: Proceedings of the Forty-First Qnnual ACM Symposium on Theory of Computing. pp. 639–648 (2009)
- Bar-Yehuda, R., Even, S.: A linear-time approximation algorithm for the weighted vertex cover problem. Journal of Algorithms 2(2), 198–203 (1981)
- Brönnimann, H., Goodrich, M.T.: Almost Optimal Set Covers in Finite VC-Dimension. Discrete Comput. Geom. 14, 263–279 (1995)

- Chan, T.M., Grant, E., Könemann, J., Sharpe, M.: Weighted capacitated, priority, and geometric set cover via improved quasi-uniform sampling. In: Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms. p. 1576–1585. SODA '12, Society for Industrial and Applied Mathematics, USA (2012)
- 5. Chazelle, B.: A note on Haussler's packing lemma (1992), a note on Haussler's packing lemma
- Chvatal, V.: A greedy heuristic for the set-covering problem. Mathematics of Operations Research 4(3), 233–235 (1979)
- Clarkson, K.L.: A randomized algorithm for closest-point queries. SIAM Journal on Computing 17(4), 830–847 (1988)
- Even, G., Rawitz, D., Shahar, S.M.: Hitting sets when the VC-dimension is small. Information Processing Letters 95(2), 358–362 (2005)
- Feige, U.: A threshold of ln n for Approximating Set Cover. Journal of the ACM (JACM) 45(4), 634–652 (1998)
- Garey, M.R., Johnson, D.S.: Computers and Intractability, vol. 174. Freeman San Francisco (1979)
- Haussler, D.: Sphere packing numbers for subsets of the Boolean n-cube with bounded Vapnik-Chervonenkis dimension. Journal of Combinatorial Theory, Series A 69(2), 217–232 (1995)
- 12. Haussler, D., Welzl, E.: Epsilon-nets and simplex range queries. In: Proceedings of the Second Annual Symposium on Computational Geometry. pp. 61–71 (1986)
- Johnson, D.S.: Approximation algorithms for combinatorial problems. In: Proceedings of the Fifth Annual ACM Symposium on Theory of Computing. pp. 38–49 (1973)
- Komlós, J., Pach, J., Woeginger, G.J.: Almost tight bounds for epsilon-nets. Discret. Comput. Geom. 7, 163–173 (1992)
- 15. Matoušek, J.: Geometric discrepancy: An illustrated guide. Springer (1999)
- Mustafa, N.H.: A simple proof of the shallow packing lemma. Discret. Comput. Geom. 55(3), 739–743 (2016)
- Mustafa, N.H.: Computing optimal epsilon-nets is as easy as finding an unhit set. In: Baier, C., Chatzigiannakis, I., Flocchini, P., Leonardi, S. (eds.) 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece. LIPIcs, vol. 132, pp. 87:1–87:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2019)
- Mustafa, N.H.: Sampling in Combinatorial and Geometric Set Systems, vol. 265. American Mathematical Society (2022)
- Mustafa, N.H., Dutta, K., Ghosh, A.: A simple proof of optimal epsilon nets. Combinatorica 38(5), 1269–1277 (2018)
- Mustafa, N.H., Varadarajan, K.: Epsilon-approximations & epsilon-nets. In: Handbook of Discrete and Computational Geometry, pp. 1241–1267. Chapman and Hall/CRC (2017)
- Varadarajan, K.: Epsilon nets and union complexity. In: Proceedings of the Twenty-Fifth Annual Symposium on Computational Geometry. pp. 11–16 (2009)
- Yousuf, A.M., Rochester, E.M., Ghaderi, M.: A low-cost LoRaWAN testbed for IoT: Implementation and measurements. In: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). pp. 361–366 (2018)

A Proof of the Weighted Packing Lemma

This section proves the Weighted Packing Lemma 2. Our proof closely follows the original in Haussler [11]. The reader is referred to Matoušek [15] (Sec 5.3) for an excellent treatment of the unweighted proof. We begin by restating the weighted lemma.

Lemma 5 (Weighted Packing Lemma). Let (X, \mathcal{P}) be a set system with m elements and n sets $\mathcal{P} = \{S_1, \ldots, S_n\}$, equipped with weights $\mu : X \to \mathbb{R}_{\geq 0}$ with $\mu(X) > 0$. Let VC-dim $(\mathcal{P}) \leq d$ for some integer $d \geq 1$, and let $\delta > 0$ be a constant such that $\mu(\Delta(S_i, S_j)) \geq \delta$ for all $1 \leq i < j \leq n$. Then

$$\operatorname{card}(\mathcal{P}) \leq 2\mathbb{E}\left[\operatorname{card}(\mathcal{P}|_{Y})\right]$$

where Y is the set of unique elements in an random sample $U = (u_1, u_2, \ldots, u_s)$ of size $s = \lceil \frac{8d}{\delta} \rceil - 1$, in which each element u_k is sampled iid $u_k \sim \mu(\cdot)$ with replacement.

The proof strategy is the following. First, we consider a random sample $U = (u_1, \ldots, u_s)$ sampled from $\mu(\cdot)$ with replacement. The sample U may contain repeated elements; let $Y \subseteq X$ denote the set of unique elements in U. Next, we generate a unit-distance graph; a weighted graph that depends on the random set Y, and derive three claims about the total weight of its edges: (i) an upper bound on the total weight, (ii) a partial lower bound, and (iii) a complete lower bound. Combining the bounds completes the proof. The main differences between our approach and that of Chazelle, Haussler, and Mustafa are twofold [5,11,16]. Firstly, we permit non-uniform weights μ as opposed to weighing each set by its cardinality, and for we sample from a probability distribution proportional to μ . Secondly we use sampling with replacement as opposed to without replacement. This makes the analysis more straightforward under non-uniform sampling using μ .

A weighted unit-distance graph over the sampled set-system takes a central stage in the proof. In this graph, sets are viewed as vertices, and edges are drawn between any two sets at unit-distance of each other.

Definition 2 (Unit distance graph). Let (X, \mathcal{P}) be a set system. The unit distance graph of (X, \mathcal{P}) is a graph $G = (\mathcal{P}, E_{\mathcal{P}})$ with vertex set \mathcal{P} and edges

$$E_{\mathcal{P}} = \{\{S_i, S_j\} \in \mathcal{P} \times \mathcal{P} : \operatorname{card}(\Delta(S_i, S_j)) = 1 \text{ and } i \neq j\}$$

In other words, edges represents pairs of sets that differ on exactly one element. The following lemma connects the number of edges with the VC-dimension.

Lemma 6 (Haussler [11]). Fix a set system (X, \mathcal{P}) with VC-dim $(\mathcal{P}) = d$. Let $G = (\mathcal{P}, E_{\mathcal{P}})$ be its unit distance graph. Then $\operatorname{card}(E_{\mathcal{P}}) \leq d \operatorname{card}(\mathcal{P})$.

We use this lemma to bound the total edge weight in our unit-distance graph. But first, we define our particular unit-distance graph and its edge weights. We construct a graph that depends on the random set $Y \subseteq X$. Consider the projection of \mathcal{P} to Y, denoted $\mathcal{P}|_Y$. Let $G_Y = (\mathcal{P}|_Y, E_{\mathcal{P}|_Y})$ be a unit distance graph over the projected system. For each vertex $S' \in \mathcal{P}|_Y$, define the *vertex* weight as the number of sets $S \in \mathcal{P}$ that are projected to S' in $\mathcal{P}|_Y$, that is

$$w(S') = \operatorname{card}(\{S \in \mathcal{P} : S \cap Y = S'\})$$

Moreover, define the weight of an edge $\{S'_i, S'_j\} \in E_{\mathcal{P}|_Y}$ as the minimum over the weights of its two vertices, $w\left(\{S'_i, S'_j\}\right) = \min\{w(S'_i), w(S'_j)\}$. Finally, let the total edge weight be $W = \sum_{e \in E_{\mathcal{P}|_Y}} w(e)$. Note that the weights are random variables because they depend on the random selection Y. We proceed by bounding the total edge weight. This will allow us to bound the size of the packing in a way that "looks like a magician's trick" [15].

First, we find an upper bound for the total edge weight. This is naturally also an upper bound on the expected total edge weight.

Claim. The total edge weight is upper-bounded by $W \leq 2d \operatorname{card}(\mathcal{P})$.

Proof. This is the proof of Haussler and Chazelle [11,5]. Lemma 6 implies that:

$$\sum_{S' \in \mathcal{P}|_{Y}} \deg(S') = 2 \operatorname{card}(E_{\mathcal{P}|_{Y}}) \le 2d \operatorname{card}(\mathcal{P}|_{Y}).$$

Hence there exists a vertex S' in $\mathcal{P}|_Y$ with degree at most 2*d*. Each edge incident to S' has weight at most w(S'), so the vertex S' is responsible for edges of total weight at most 2dw(S'). Applying this inductively proves the bound on the total weight.

$$W \leq \sum_{S' \in \mathcal{P}|_{Y}} (\text{Weight due to } S') \leq 2d \sum_{S' \in \mathcal{P}|_{Y}} w(S') = 2d \operatorname{card}(\mathcal{P}) = 2dn$$

Next we derive a lower bound. It suffices to derive the bound for a reduced problem. Let U' be the subsequence of sample $U = (u_1, \ldots, u_s)$ containing all but the last element u_s . Similarly, let Y' be the set of unique elements in the subsample U', just as Y denotes the unique elements in the full sample U. Now, when the final element u_s is added to U', some vertices in $\mathcal{P}|_Y$ may form an edge due to u_s . This occurs exactly when the last sampled element u_s falls in the symmetric difference of two sets that were previously equal in the projection $\mathcal{P}|_{Y'}$. Let W_s be the sum of the weights of the edges due to element u_s . In other words, the weight W_s is the weight of the edges generated by adding a random element $u_s \sim \mu(\cdot)$ to the random sequence $U' = (u_1, \ldots, u_{s-1})$. Because the samples are iid, given a squence, the ordering of the random elements in the sequence is uniform over all permutations, so we can apply symmetry of expectation:

$$\mathbb{E}[W] = s\mathbb{E}[W_s].$$

Hence, it suffices to derive an upper bound on $\mathbb{E}[W_s]$.

Following Haussler [15], an intermediate step is to lower bound the expectation of the weight W_s due to the *s*th element u_s conditional on the previous s-1 elements U'. The lower bound follows from considering pairs of vertices that lack edges in $E_{\mathcal{P}|_{Y'}}$ but that may share an edge after adding u_s . This happens exactly when the new element u_s falls in their symmetric difference. This event occurs with probability at least δ , because we assume $\mu(\Delta(S,)) \geq \delta$ for all pairs $S \neq R$ in our packing \mathcal{P} . This argument uses the fact that we sample elements proportional to the weight μ .

Claim. For any set $Y' \subseteq X$ generated by a fixed sequence $u' = (u_1, \ldots, u_{s-1})$, with $u_k \in X$ for $k = 1, \ldots, s - 1$, it holds that

$$\mathbb{E}_{\mu}[W_s \mid U' = u'] \ge \frac{\delta}{2} \left(n - \operatorname{card}(\mathcal{P}|_{Y'}) \right)$$

where \mathbb{E}_{μ} denotes expectation over is over a single random element $u_s \sim \mu(\cdot)$.

Proof. Fix a set $Y' \subseteq X$ corresponding to the unique elements in the partial selection $u' = (u_1, \ldots, u_{s-1})$. Consider an arbitrary set Q in the projection $\mathcal{P}|_{Y'}$. There may be many sets in \mathcal{P} that map to Q in $\mathcal{P}|_{Y'}$. Let \mathcal{P}_Q be the collection of these sets, and let b denote the number of such sets. Note that for any pair of sets $S_i, S_j \in \mathcal{P}_Q, Y'$ cannot contain any element in the symmetric difference $\Delta(S_i, S_j)$, or else the two sets would not map to the same Q. However, when an additional element u_s is sampled and added to Y' (with the possibility that u_s is already in Y') the collection \mathcal{P}_Q is partitioned into two groups: (i) sets that contain u_s , and (ii) sets that do not contain u_s . Let b_1 and b_2 denote the number of sets in these groups, respectively, with $b = b_1 + b_2$. These two groups are at a unit-distance in $\mathcal{P}|_Y$. The weight of the resulting edges in the unit-distance graph is exactly $\min\{b_1, b_2\}$.

By adding up the expected weights due to each pair we get $\mathbb{E}[W_s]$. For every pair $S'_1, S'_2 \in \mathcal{P}_Q$, the probability that u_s hits $\Delta(S'_1, S'_2)$ is $\mu(\Delta(S'_1, S'_2)) \geq \delta$. Thus, the expected contribution of each pair to the product b_1b_2 is at least δ . Note that the sum $b_1 + b_2 = b$ depends on Y', however is independent of u_s . Hence

$$\mathbb{E}[\min\{b_1, b_2\}] \ge \mathbb{E}\left[\frac{b_1 b_2}{b}\right]$$

$$= \sum_{\substack{S'_1, S'_2 \in \mathcal{P}_Q \\ 0 \text{ card}(\mathcal{P}_Q) \text{ (card}(\mathcal{P}_Q) - 1) \delta \\ 2 \text{ card}(\mathcal{P}_Q)}} \frac{\mathbb{E}\left[U_s \in \Delta(S'_1, S'_2,)\right]}{b}$$

The first inequality follows from the fact that $\min\{b_1, b_2\} \ge b_1 b_2/b$. The first equality uses the fact that there \mathcal{R}_Q is partitioned into two groups, each at unit distance, so $b_1 b_2$ is the total number of edges between the two groups. Taking

the above inequality and summing over all vertices $Q \in \mathcal{P}|_{Y'}$ gives the result

$$\mathbb{E}[W_s \mid U' = u'] \ge \sum_{Q \in \mathcal{P}|_{Y'}} \frac{\delta}{2} \left(\operatorname{card}(\mathcal{P}_Q) - 1 \right)$$
$$= \frac{\delta}{2} \left(\sum_{Q \in \mathcal{P}|_{Y'}} \operatorname{card}(\mathcal{P}_Q) - \sum_{Q \in \mathcal{P}|_{Y'}} 1 \right) = \frac{\delta}{2} \left(n - \operatorname{card}(\mathcal{P}|_{Y'}) \right).$$

By using the claim above it is now straightforward to produce a lower bound on the expected total edge weight.

Claim. $\mathbb{E}[W] \geq 4dn - 4d\mathbb{E}[\operatorname{card}(\mathcal{P}|_{Y'})].$

Proof. We employ the reduction from above as well as the partial lower bound. Let X^k denote the Cartesian product of element set X. For any length sequence $u' = (u_1, \ldots, u_{s-1})$ of elements in X^{s-1} , let $Y'(u') \subseteq X$ be the set of unique elements in u'. Analogously let let Y'(U') denote the random set of unique elements in a random sequence U' in X^{s-1} . It then follows that:

$$\begin{split} \mathbb{E}[W] &= s\mathbb{E}[W_s] \\ &= s\sum_{u' \in X^{s-1}} \mathbb{E}[W_s \mid Y'(U') = Y(u')] \cdot \mathbb{P}[U' = u'] \\ &\geq s\sum_{u' \in X^{s-1}} \frac{\delta}{2} \left(n - \operatorname{card}\left(\mathcal{P}|_{Y(u')}\right) \right) \cdot \mathbb{P}[U' = u'] \\ &= \frac{s\delta}{2} \left(n\sum_{u' \in X^{s-1}} \mathbb{P}[U' = u'] - \sum_{u' \in X^{s-1}} \operatorname{card}\left(\mathcal{P}|_{Y(u')}\right) \cdot \mathbb{P}[U' = u'] \right) \\ &= \frac{s\delta}{2} \left(n - \mathbb{E} \left[\operatorname{card}\left(\mathcal{P}|_{Y'(U')}\right) \right] \right), \end{split}$$

where the first inequality uses the partial lower bound, and the last equality follows from the definition the random set Y(U').

Finally, all the pieces are in place to prove the Packing Lemma. Using the lower and upper bounds it follows that cardinality of \mathcal{P} is bounded above by twice the expected size of $\mathcal{P}|_{Y'}$:

$$4dn - 4d\mathbb{E}[\operatorname{card}(\mathcal{P}|_{Y'})] \le \mathbb{E}[W] \le 2dn.$$

This yields the statement of the weighted packing lemma, completing the proof. \Box