# An Optimal Sparsification Lemma for Low-Crossing Matchings and Its Applications to Discrepancy and Approximations

**Mónika Csikós**
Université Paris Cité, IRIF, CNRS UMR 8243 and DI-ENS, Université PSL, France

**Nabil H. Mustafa**
Université Sorbonne Paris Nord, Laboratoire LIPN, CNRS 7030, France

**Abstract**

Matchings with low crossing numbers were originally introduced in the late 1980s in the seminal works of Welzl [35, 36] and Chazelle-Welzl [11]. They have since become fundamental structures in combinatorics, computational geometry, and algorithms.

In this paper, we study matchings with low crossing numbers and their relation to random samples. In particular, our main technical result states that, given a set system $(X, \mathcal{S})$ with dual VC-dimension $d$ and a parameter $\alpha \in (0, 1]$, a random set of $\tilde{\Theta}\left(n^{1+\alpha}\right)$ edges of $\binom{X}{2}$ contains a linear-sized matching with crossing number $O\left(n^{1-\alpha/d}\right)$.

Furthermore, we show that this bound is optimal up to a logarithmic factor.

By incorporating the above sampling step to existing algorithms, we obtain improved running times, by a factor of $\tilde{\Theta}(n)$, for computing matchings with low crossing numbers. This immediately implies new bounds for a number of well-studied problems, such as combinatorial discrepancy, $\varepsilon$-approximations and their applications.

To the best of our knowledge, these are the first near-linear time algorithms for general, non-geometric set systems, for a) matchings with *sub-linear* crossing numbers, and b) discrepancy beating the standard deviation bound. As an immediate consequence we get fast algorithms for computing $o(1/\varepsilon^2)$-sized $\varepsilon$-approximations.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** low-crossing matchings, uniform sampling, discrepancy, approximations

**Digital Object Identifier** 10.4230/LIPIcs.ICALP.2024.49

**Category** Track A: Algorithms, Complexity and Games

**Funding** Supported by the French ANR PRC grant ADDS (ANR-19-CE48-0005), ANR AlgoriDAM (ANR-19-CE48-0016), and Fondation Sciences Mathématiques de Paris (FSMP).

## 1 Introduction

A perfect matching of a set $X$ is a partition of $X$ into $|X|/2$ disjoint pairs[1]. Given a set system $(X, \mathcal{S})$, we say that a set $S \in \mathcal{S}$ *crosses* a pair $\{x, y\} \subseteq X$ iff $|S \cap \{x, y\}| = 1$. Then for a perfect matching $M$ of $X$, the *crossing number* of $M$ with respect to $\mathcal{S}$ is defined to be the maximum number of edges of $M$ crossed by any $S \in \mathcal{S}$.

Matchings with low crossing numbers were originally introduced by Welzl [35, 36] for the special case where $X$ is a set of points in $\mathbb{R}^d$ and $\mathcal{S}$ is induced on $X$ by half-spaces. His result was then improved and generalized by Chazelle and Welzl [11] to a broader class of set systems using the notion of the *dual shatter function* $\pi^*_{\mathcal{S}}$ of $(X, \mathcal{S})$, which is defined as follows:

[1] If $|X|$ is odd, then we partition $X$ into $\lfloor |X|/2 \rfloor$ disjoint pairs plus a singleton set.


51st International Colloquium on Automata, Languages, and Programming (ICALP 2024).
Editors: Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson;
Article No. 49; pp. 49:1–49:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

For any $k \leq |\mathcal{S}|$, $\pi^*_{\mathcal{S}}(k)$ denotes the maximum number of equivalence classes on $X$ defined by a $k$-element subfamily $\mathcal{R} \subseteq \mathcal{S}$, where $x, y \in X$ are equivalent w.r.t. $\mathcal{R}$ iff $x$ belongs to the same sets of $\mathcal{R}$ as $y$. The number of such equivalence classes, for a given $\mathcal{R} \subseteq \mathcal{S}$, is essentially the number of non-empty cells, w.r.t. $X$, in the Venn diagram of the sets of $\mathcal{R}$.

The family of set systems with polynomially bounded dual-shatter function includes set systems with bounded dual VC-dimension and most of the commonly-studied geometric cases – e.g., the primal and dual set systems induced by half-spaces, balls, intersections/unions of bounded complexity geometric objects and more generally, algebraic varieties [26].

The following theorem on the existence of matchings with low crossing number is a celebrated and fundamental result in computational geometry.

▶ **Theorem A** ([11, 20]). *Let $(X, \mathcal{S})$ be a set system with $n = |X|$, and dual shatter function $\pi^*_{\mathcal{S}}(k) = O(k^d)$. Then there exists a perfect matching on $X$ with crossing number $O\left(n^{1-1/d} + \ln |\mathcal{S}|\right)$.*

Theorem A has had numerous applications, including range searching, extremal results for hypergraphs with bounded VC dimension, low-discrepancy colorings in geometric hypergraphs, near-optimal sized epsilon-approximations, to name a few. We refer the reader to these books [10, 25, 29] for more information.

The power of Theorem A, as well as the difficulty in efficiently computing such matchings, come from the same source: a vanishingly small proportion of matchings are low-crossing matchings. Indeed, it is not difficult to see that a random matching will have crossing number $\Omega(n)$. For example, any set containing $n/2$ elements of $X$ will cross, in expectation, a linear number of edges of such a matching.

This contrasts sharply with the case for the related structures of $\varepsilon$-nets and $\varepsilon$-approximations: a large-enough random sample is an $\varepsilon$-net/$\varepsilon$-approximation with high probability and thus a constant fraction of all subsets of $X$ are $\varepsilon$-nets and $\varepsilon$-approximations. The use of randomness fails in our case since these $\varepsilon$-net/$\varepsilon$-approximation bounds rely on the fact that each set in the set system has *large* measure – at least $\varepsilon$-th measure – and the behavior of the random sample can be analyzed independently for each set, which are all known in advance.

Thus the ingenious proof of Theorem A takes a different route: it constructs a matching in $n/2$ iterations, where each iteration considers all remaining edges and picks one minimizing a certain function. This is computationally expensive, and consequently, all old and recent – with one exception – algorithms for building matchings with crossing numbers $o(n)$ in general set systems have $\Omega\left(n^2\right)$ running times, even when $|\mathcal{S}| = O(n)$ [11, 13, 18, 19].

The one exception is the algorithm proposed by Ducoffe *et al.* [16], designed specifically to get below the quadratic (in $|X| + |\mathcal{S}|$) running time barrier: they show that there is a universal constant $c > 2$ such that for any set system $(X, \mathcal{S})$ with VC-dimension $D$ and dual VC-dimension $d$, one can compute a spanning path with crossing number $\tilde{O}\left(n^{1-1/(c \cdot D \cdot d)}\right)$ in time $\tilde{O}\left(|\mathcal{S}| + n^{2-1/(c \cdot D \cdot d)}\right)$.

## 2 Our Results

Motivated by the above considerations, we revisit the key question, of independent interest, of the utility of random sampling for constructing matchings with low crossing numbers. Our main technical result is that although a random perfect matching is very far from a low-crossing matching, a slightly larger random sample of edges contains a linear-sized matching that is close to a low-crossing matching. More precisely:

▶ **Lemma 1** (Main lemma; Proof in Section 4). *Let $(X, \mathcal{S})$, $n = |X|$, be a set system with dual shatter function $\pi^*(k) = O(k^d)$, and let $\alpha \in (0, 1]$, $\delta \in (0, 1)$ be two given parameters. Let $E$ be a uniform random sample from $\binom{X}{2}$, where each edge is picked i.i.d. with probability*

$$p = \min\left\{\frac{2\ln n}{n^{1-\alpha}} + \frac{4\ln(2/\delta)}{n^{2-\alpha}},\ 1\right\}.$$

*Then with probability at least $1 - \delta$, $E$ contains a matching of size $n/4$ of crossing number*

$$O\left(n^{1-\alpha/d} + \ln|\mathcal{S}|\right).$$

Moreover, we show that the above is near-optimal.

▶ **Lemma 2** (Optimality; Proof in Section 5). *For any $d \geq 2$, $c \geq 2$, $\alpha > 0$, and $n_0 \in \mathbb{N}$, there is a set system $(X, \mathcal{S})$ with $|X| = n \geq n_0$, and dual shatter function $\pi^*_{\mathcal{S}}(k) = O(k^d)$, such that the following holds:*

> *let $E$ be a random edge-set obtained by selecting each edge in $\binom{X}{2}$ i.i.d. with probability $p(n) = o\left(n^{\alpha-1}\right)$. Then with probability at least $1/2$, every matching in $E$ of size $n/c$ has crossing number $\omega\left(n^{1-\alpha/d}\right)$ with respect to $\mathcal{S}$, where the constants in the asymptotic notation depend on $d$ and $c$.*

▶ Remark. The same asymptotic results hold for low-crossing spanning paths and spanning trees.

We find Lemma 1 surprising for the following reason: the classical proof of Theorem A assigns exponentially-increasing weights to the sets of $\mathcal{S}$, which then dictate the choice of the edge picked at each iteration. Thus a different choice of the edge at iteration $i$ could result in a changing of the weight distribution, which then influences the sequence of edges picked for all later iterations. At first glance, a random sample chosen once, and uniformly from $\binom{X}{2}$ cannot simply assure that it will contain many edges from all possible *exponentially* many paths possibly chosen by the algorithm.

In particular, if we fix an initial uniform sample of edges, and build the matching using this sample (by always choosing a light edge from the sample, as in previous algorithms), it introduces a bias (as the set of uncovered points depend on the initial sample of edges) and we cannot assume anymore that among the uncovered points, every edge is picked i.i.d. with a fixed probability. Indeed, with later iterations, the possible paths to be taken care of increase exponentially and the initial random sample has low probability of containing a good perfect matching.

Therefore, we take a different, more subtle, approach in the analysis, similar in spirit to the technique of quasi-uniform sampling of Varadarajan [34] and Chan et al.[9]:

1. instead of a perfect matching, we aim for a *linear-sized* partial matching with crossing number $O\left(n^{1-\alpha/d}\right)$. As we will prove, this requirement is weak-enough for a single sample to work, but strong-enough so that to compute a perfect matching, $O(\log n)$ adaptive uniform samples are sufficient.
2. instead of the classical algorithm, we propose a new randomized version where an edge from our initial, fixed, random sample of edges is picked in each iteration with a carefully chosen probability distribution that *does* depend on the changing weights in each iteration.

While Lemma 1 only guarantees the existence of a low-crossing *partial* matching, it can be used to speed up the fastest existing algorithms, by computing a linear sized matching and recursing on the uncovered points. This leads to the following result that improves the running times of previous-best algorithm of Csikós and Mustafa [13] by nearly a factor of $\Theta(n)$, at the cost of a higher, but still sub-linear crossing number.

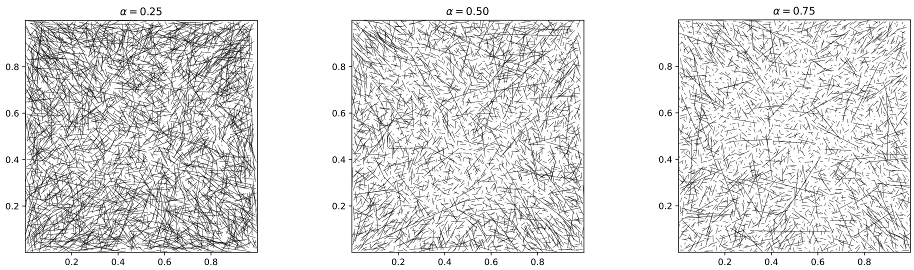
**Table 1** $(X, \mathcal{S})$ has dual-shatter function $\pi^*(k) = O(k^d)$, $n = |X|$, $m = |\mathcal{S}|$.

| Problem | Our Method: Guarantee | Our Method: Time | Previous-best: Guarantee | Previous-best: Time |
|---|---|---|---|---|
| matching with low crossing number | $O\left(n^{1-\alpha/d} + \ln m \ln n\right)$ | $\tilde{O}\left(n^{(1+\alpha)+\frac{2\alpha}{d}} + m \cdot n^{\frac{2\alpha}{d}}\right)$ | $O\left(n^{1-1/d} + \ln m \ln n\right)$ | $\tilde{O}\left(n^{2+\frac{2}{d}} + m \cdot n^{\frac{2}{d}}\right)$ [13] |
| | | | $O\left(n^{1-1/O(d \cdot 2^d)} + \ln m \ln n\right)$ | $\tilde{O}\left(m + n^{2-1/O(d \cdot 2^d)}\right)$ [16] |
| combinatorial discrepancy | $O\left(\sqrt{n^{1-\alpha/d} \ln m + \ln^2 m \ln n}\right)$ | $\tilde{O}\left(n^{(1+\alpha)+\frac{2\alpha}{d}} + m \cdot n^{\frac{2\alpha}{d}}\right)$ | $O\left(\sqrt{n^{1-1/d} \ln m}\right)$ | $\tilde{O}\left(n^{2+\frac{2}{d}} + m \cdot n^{\frac{2}{d}}\right)$ [13] |

▶ **Corollary 3** (Proof in Appendix A.1)**.** *Let $(X, \mathcal{S})$ be a set system with dual shatter function $\pi^*_{\mathcal{S}}(k) = O(k^d)$, $n = |X|$ and $\alpha \in (0, 1]$. Then there is a randomized algorithm which returns a perfect matching of expected crossing number $O\left(n^{1-\alpha/d} + \ln |\mathcal{S}| \ln n\right)$ in expected time $\tilde{O}\left(n^{1+\alpha+\frac{2\alpha}{d}} + |\mathcal{S}| \cdot n^{\frac{2\alpha}{d}}\right)$.*

▶ Remark. The algorithm of Corollary 3 can easily be modified to create a spanning path with the same crossing number guarantee.

**Figure 1** The matchings output by the algorithm of Corollary 3 on the set system induced by half-spaces in $\mathbb{R}^2$ are shown below, for $\alpha = 0.25, 0.5$ and $0.75$.




## Algorithmic Consequences

Besides a new trade-off between quality and speed of computation, Corollary 3 has several algorithmic applications.

**1. A better sub-quadratic time (in $|X| + |\mathcal{S}|$) algorithm.** Setting a low value for the parameter $\alpha \in (0, 1]$ allows us to get a running time that is close to linear, at the expense of the crossing number. At the other end of the spectrum, we can get below the quadratic running time barrier without sacrificing too much in the crossing number. This improves the result of Ducoffe *et al.* [16] – who computed a spanning path with crossing number $\tilde{O}\left(n^{1-1/(c \cdot D \cdot d)}\right)$ in time $\tilde{O}\left(m + n^{2-1/(c \cdot D \cdot d)}\right)$, where $D = \text{VCdim}(X, \mathcal{S})$ – by letting $\alpha = d/(d+3)$. Besides improving the crossing number, the new algorithm also does not depend on the primal VC dimension.

▶ **Corollary 4.** *Let $(X, \mathcal{S})$ be a set system with dual shatter function $\pi^*_{\mathcal{S}}(k) = O(k^d)$, $n = |X|$. Then there is a randomized algorithm which returns a perfect matching of expected crossing number $O\left(n^{1-1/(d+3)} + \ln |\mathcal{S}| \ln n\right)$ in expected time $\tilde{O}\left(n^{2-\frac{1}{d+3}} + |\mathcal{S}| \cdot n^{\frac{2}{d+3}}\right)$.*

▶ Remark. In the work of Ducoffe *et al.* [16], spanning paths were used as a key algorithmic tool for computing the diameter of graphs. Their main result is a $\tilde{O}\left(k\, n^{2-1/O(d\cdot 2^d)}\right)$ time algorithm to decide whether the diameter of a graph $G$ with distance VC-dimension $d$ is at most $k$. Using Corollary 4 in their algorithmic framework ([16, Lemma 6]), we get an algorithm that decides whether $G$ has diameter at most $k$, in time $O\left(k\, n^{2-1/(d+3)}\right)$.

**2. Discrepancy.** Using Corollary 3 in a standard iterative halving scheme [27] gives us faster approximation algorithms for combinatorial discrepancy, again improving the previous-best algorithms by nearly a factor of $\tilde{\Theta}(n)$, at the cost of a slightly higher discrepancy.

▶ **Corollary 5** (Proof in Appendix A.2). *Let $(X, \mathcal{S})$, $n = |X|$, $m = |\mathcal{S}|$, be a set system and $d$ be a constant such that $\pi^*_{\mathcal{S}}(k) = O\left(k^d\right)$. For any $0 < \alpha \leq 1$, there is a randomized algorithm which constructs a coloring $\chi$ of $X$ with expected discrepancy $O\left(\sqrt{n^{1-\alpha/d}\ln m + \ln^2 m \log n}\right)$, in expected time $\tilde{O}\left(n^{1+\alpha+\frac{2\alpha}{d}} + m \cdot n^{\frac{2\alpha}{d}}\right)$.*

This allows us to get the first near-linear time algorithm, for general non-geometric set systems, that beats the standard deviation discrepancy bound:

▶ **Corollary 6.** *Let $(X, \mathcal{S})$, $n = |X|$, $m = |\mathcal{S}|$, be a set system and $d$ be a constant such that $\pi^*_{\mathcal{S}}(k) = O\left(k^d\right)$. For any $0 < \varepsilon \leq 1$, there is a randomized algorithm which constructs a coloring $\chi$ of $X$ with expected discrepancy $O\left(\sqrt{n^{1-\frac{\varepsilon}{d+2}}\ln m + \ln^2 m \log n}\right)$, in expected time $\tilde{O}\left(n^{1+\varepsilon} + m^{1+\varepsilon}\right)$.*

**3. $\varepsilon$-Approximations.** The iterative application of Corollary 5 implies the following faster algorithm for computing $\varepsilon$-approximations [10, 29, 32, 24, 14].

▶ **Corollary 7.** *Let $\varepsilon \in (0, 1)$, $(X, \mathcal{S})$ be a set system and $c, d, D$ be constants such that $\pi^*_{\mathcal{S}}(k) \leq ck^d$ and $\mathrm{VCdim}(X, \mathcal{S}) \leq D$. Then for any $\alpha \in (0, 1)$, there is a randomized algorithm which returns an $\varepsilon$-approximation $A \subset X$ of size*

$$O\left(\left(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon}\right)^{\frac{d}{d+\alpha}}\right)$$

*in expected time*

$$\tilde{O}\left(\left(\frac{D}{\varepsilon^2}\right)^{1+\alpha+\frac{2\alpha}{d}} + \left(\frac{D}{\varepsilon^2}\right)^{D+\frac{2\alpha}{d}}\right).$$

## 3 Previous Results

### Matchings with low crossing numbers

The study of perfect matchings (along with spanning paths and spanning trees) with low crossing number was originally introduced for geometric range searching [35, 11]. Since then, they have found applications in various fields, for instance, discrepancy theory [27], learning theory [3], or algorithmic graph theory [16].

The proof of Theorem A is constructive, using the Multiplicative Weights Update (**MWU**) technique. Moreover it works for any (abstract) set system $(X, \mathcal{S})$ with polynomially bounded dual-shatter function. It builds a low-crossing matching iteratively, guided by a weight

**Algorithm 1** MatchingMWU$((X,\mathcal{S}))$.

```
ω_1(S) ← 1 for all S ∈ 𝒮
X_1 ← X
for i = 1, ..., n/2 do
    e_i ← the lightest edge in (X_i choose 2) w.r.t. ω_i
    Obtain ω_{i+1} from ω_i by doubling the weights of each set crossing e_i
    X_{i+1} ← X_i \ endpoints(e_i)
return {e_1, ..., e_{n/2}}
```

function $\omega$ on $\mathcal{S}$, with initial weights set to 1. At each iteration, the algorithm adds the "lightest" edge to the matching – that is, the edge that is crossed by sets of minimum total weight. At the end of an iteration, it updates $\omega$ by doubling the weight of each set crossing the picked edge. See Algorithm 1. The algorithmic bottleneck is in finding such an edge: for an abstract set system without additional structure, this takes $O(n^2m)$ time for each of the $n/2$ iterations, giving a total running time of $O\left(n^3m\right)$, where $n=|X|$ and $m=|\mathcal{S}|$. Using MWU together with ideas from linear programming duality have led to the current-best running time for computing matchings with crossing number $O\left(n^{1-1/d}\right)$ in time $\tilde{O}\left(n^{2+2/d}+mn^{2/d}\right)$ [13].

A different approach was proposed by Har-Peled [19] (see also [17]). His result implies that if $(X,\mathcal{S})$ has a spanning tree with crossing number $\kappa=\Theta(n^\gamma)$ for some $\gamma\in[1/\log n,1]$, then a spanning tree of crossing number $O(\kappa/\gamma)$ can be found by solving an LP on $\binom{n}{2}$ variables and $m+n$ constraints, see also [18]. Another result for general set systems having spanning trees with crossing number $\kappa$, is based on rounding fractional solutions of minimax integer programs with matroid constraints. This method gives a randomized algorithm that constructs a spanning tree with expected crossing number at most $\kappa+O(\sqrt{\kappa\log m})$ in time $\tilde{O}(mn^4+n^8)$ [12].

For the geometric case, where $X$ is a set of points in $\mathbb{R}^d$ and $\mathcal{S}$ is induced by half-spaces, Chan [8] gave a $O\left(n\log n\right)$ time algorithm to compute a matching with crossing number $O\left(n^{1-1/d}\right)$ using hierarchical cuttings. More generally, spatial partitioning by polynomials has also been extensively studied in the last decade [1, 2].

## Discrepancy

Spencer [31] showed that for any set system $(X,S)$, there exists a coloring of $X$ with discrepancy $O\left(\sqrt{n\ln(m/n)}\right)$, which is tight and improves the general bound for $m=O\left(n\right)$. A series of algorithms for its construction started with the breakthrough work of Bansal [5], who gave the first polynomial-time randomized algorithm (using SDP rounding) to compute a coloring with discrepancy $O\left(\sqrt{n}\ln(m/n)\right)$, which matches the bound of Spencer for $m=O(n)$. Later Lovett and Meka [23] gave a combinatorial randomized algorithm for constructing colorings with discrepancy $O\left(\sqrt{n\ln(m/n)}\right)$ and improved the expected running time to $\tilde{O}\left(n^3+m^3\right)$; see also [30] for a different proof. The algorithm of Bansal was de-randomized [7] (but still used a non-constructive method to prove the feasability of an underlying SDP), and later, Levy et al.[22] used the multiplicative weights update technique to give a deterministic $O\left(n^4m\right)$-time algorithm to compute a two-coloring with discrepancy $O\left(\sqrt{n\ln(m/n)}\right)$ for an arbitrary set system. See also [6] for a random-walk algorithm for Banaszczyk's discrepancy bound, with running time $O\left(n^{3.3728..}+nm^{2.3728..}\right)$. Most recently, Larsen proposed an

$O(n^2 m \ln(2+m/n)+n^3)$ time algorithm with hereditary discrepancy guarantees [21]. The discrepancy guarantees of [21] can also be achieved with an $\tilde{O}(\text{nnz}(A)+n^3)$ time algorithm [15], where $A$ is the membership matrix of $(X,\mathcal{S})$. We note that many of these algorithms can be applied in more general settings (e.g. for real-valued matrices), however none of them provide a sub-quadratic time algorithm for the combinatorial discepancy problem in structured set systems.

## 4 Proof of the Main Lemma

In this section, we prove Lemma 1. Given a weight function $\omega\colon \mathcal{S} \to \mathbb{R}^+$, for $e \in \binom{X}{2}$, set

$$\text{weight}(e,\omega) := \sum_{\substack{S\in\mathcal{S}:\\ e \text{ crosses } S}} \omega(S).$$

Then for any integer $k>0$, by *"the $k$ lightest edges of $\binom{X}{2}$ w.r.t. $\omega$"*, we refer to the $k$ edges with the smallest values of $\text{weight}(\cdot,\omega)$ (ties broken arbitrarily).

We show the required property of the random sample $E$ via the analysis of the randomized algorithm RelaxedMWU presented in Algorithm 2. In particular, Lemma 1 immediately follows from these two properties.

▶ **Lemma 8.** *Let $(X,\mathcal{S})$ be a set system with dual shatter function $\pi^*_{\mathcal{S}}(k) = O\left(k^d\right)$, $\alpha \in (0,1]$ and $E \subseteq \binom{X}{2}$. For any halting iteration $T=t$, the set of edges returned by RelaxedMWU$\big((X,\mathcal{S}),\alpha,E\big)$ have crossing number $O\left(t^{1-\alpha/d} + \ln|\mathcal{S}|\right)$.*

▶ **Lemma 9.** *If $E \subseteq \binom{X}{2}$ is an i.i.d. sample where each edge is picked with probability*

$$p = \min\left\{\frac{2\ln n}{n^{1-\alpha}} + \frac{4\ln(2/\delta)}{n^{2-\alpha}},\ 1\right\},$$

*then RelaxedMWU$((X,\mathcal{S}),\alpha,E)$ returns at least $n/4$ edges with probability at least $1-\delta$.*

▶ Remark. Lemma 8 does not use the fact that $E$ is chosen randomly, or the randomness used in the algorithm; these are only used by Lemma 9.

**Algorithm 2** RelaxedMWU$\big((X,\mathcal{S}),\alpha,E\big)$.

```
ω_1(S) ← 1 for all S ∈ 𝒮
X_1 ← X
for iteration i = 1,...,n/2 do
    ℰ_i ← the |X_i|^{2−α} lightest edges in (X_i choose 2) w.r.t. ω_i
    if E ∩ ℰ_i = ∅ then
        return {e_1,...,e_{i−1}} and set T = i − 1
    else
        Pick an edge e_i from E ∩ ℰ_i uniformly at random
        Compute ω_{i+1} from ω_i by doubling the weight of each set crossing e_i
        X_{i+1} ← X_i \ endpoints(e_i)
return {e_1,...,e_{n/2}} and set T = n/2
```

▶ Remark. The precise definition of $\mathcal{E}_i$ (line 4 of Algorithm 2) should consider the $\lfloor |X_i|^{2-\alpha}\rfloor$ lightest edges. For the simplicity of presentation, we replace $\lfloor |X_i|^{2-\alpha}\rfloor$ with $|X_i|^{2-\alpha}$ throughout the proof.

We now prove the two key properties of RelaxedMWU separately.

## 4.1 Proof of Lemma 8

For any function $f : \mathcal{S} \to \mathbb{R}$, define $f(\mathcal{S}) := \sum_{S \in \mathcal{S}} f(S)$.

We will use the following key lemma; its proof will be presented later.

▶ **Lemma 10.** *Let $(X, \mathcal{S})$ be a set system with dual shatter function $\pi^*_{\mathcal{S}}(k) \leq c_1 \cdot k^d$. Then given any $Y \subset X$, a weight function $w : \mathcal{S} \to \mathbb{Z}^+$, and an integer $\ell \in \left[|Y|, \binom{|Y|}{2}\right]$, there are at least $\ell$ distinct edges in $\binom{Y}{2}$ such that the total weight of the sets of $\mathcal{S}$ crossing each edge is at most*

$$(10c_1)^{1/d} \cdot \frac{w(\mathcal{S}) \cdot \ell^{1/d}}{|Y|^{2/d}}.$$

Let $\{e_1, \ldots, e_t\}$ be the output of RelaxedMWU and for each $i \in [1, t]$, set $\eta_i = \text{weight}(e_i, \omega_i)$.

At the start of iteration $i$, we have $|X_i| = n - 2i + 2$ and we pick one of the $|\mathcal{E}_i| = |X_i|^{2-\alpha}$ lightest edges of $\binom{X_i}{2}$ w.r.t. $\omega_i$. Applying Lemma 10 with $Y = X_i$, $\omega = \omega_i$ and $\ell = |\mathcal{E}_i|$, gives an upper bound on the weight of each edge in $\mathcal{E}_i$ w.r.t. $\omega_i$. In particular, as $e_i \in \mathcal{E}_i$, we have

$$\eta_i \leq (10c_1)^{1/d} \cdot \frac{\omega_i(\mathcal{S}) \cdot |X_i|^{(2-\alpha)/d}}{|X_i|^{2/d}} = (10c_1)^{1/d} \cdot \frac{\omega_i(\mathcal{S})}{|X_i|^{\alpha/d}} = \frac{(10c_1)^{1/d}\omega_i(\mathcal{S})}{(n - 2i + 2)^{\alpha/d}}. \tag{1}$$

Let $\kappa_t$ denote the maximum number of edges in $\{e_1, \ldots, e_t\}$ that are crossed by a set in $\mathcal{S}$. By the weight-update rule of the algorithm, we have

$$\omega_{t+1}(\mathcal{S}) \geq \max_{S \in \mathcal{S}} \omega_{t+1}(S) = 2^{\kappa_t},$$

and for any $j \in [1, t]$

$$\omega_{j+1}(\mathcal{S}) = \omega_j(\mathcal{S}) + \eta_j = \omega_j(\mathcal{S}) \left(1 + \frac{\eta_j}{\omega_j(\mathcal{S})}\right).$$

Applying the above equality for $j = t, \ldots, 1$ iteratively and using $\omega_1(\mathcal{S}) = |\mathcal{S}|$, we obtain

$$\omega_{t+1}(\mathcal{S}) = \omega_t(\mathcal{S}) \left(1 + \frac{\eta_t}{\omega_t(\mathcal{S})}\right) = \cdots = |\mathcal{S}| \cdot \prod_{j=1}^{t} \left(1 + \frac{\eta_j}{\omega_j(\mathcal{S})}\right) \leq |\mathcal{S}| \cdot \exp\left(\sum_{j=1}^{t} \frac{\eta_j}{\omega_j(\mathcal{S})}\right).$$

Combining the upper and lower bounds for $\omega_{t+1}(\mathcal{S})$, we get

$$2^{\kappa_t} \leq \omega_{t+1}(\mathcal{S}) \leq |\mathcal{S}| \cdot \exp\left(\sum_{j=1}^{t} \frac{\eta_j}{\omega_j(\mathcal{S})}\right) \quad \implies \quad \kappa_t \leq \frac{1}{\ln 2}\left(\ln|\mathcal{S}| + \sum_{j=1}^{t} \frac{\eta_j}{\omega_j(\mathcal{S})}\right). \tag{2}$$

Using the upper-bound on $\eta_j$ from Equation (1), we conclude that for any stopping time $t \in [1, n/2]$, the matching $\{e_1, \ldots, e_t\}$ returned by RelaxedMWU has crossing number at most

$$\begin{aligned}
\frac{1}{\ln 2}\left(\ln|\mathcal{S}| + \sum_{j=1}^{t} \frac{(10c_1)^{1/d}}{(n-2j+2)^{\alpha/d}}\right) &\leq \frac{1}{\ln 2}\left(\ln|\mathcal{S}| + \sum_{j=1}^{t} \frac{(10c_1)^{1/d}}{(2t-2j+2)^{\alpha/d}}\right) \quad \left(\textit{since } t \leq \frac{n}{2}\right) \\
&\leq \frac{1}{\ln 2}\left(\ln|\mathcal{S}| + \sum_{j=1}^{t} \frac{(10c_1)^{1/d}}{(t-j+1)^{\alpha/d}}\right) \\
&= \frac{1}{\ln 2}\left(\ln|\mathcal{S}| + \sum_{j=1}^{t} \frac{(10c_1)^{1/d}}{j^{\alpha/d}}\right) \\
&= \frac{\ln|\mathcal{S}|}{\ln 2} + O\left(t^{1-\alpha/d}\right) = O\left(t^{1-\alpha/d} + \ln|\mathcal{S}|\right).
\end{aligned}$$

This concludes the proof of Lemma 8 assuming Lemma 10. We now return to the proof of Lemma 10, for which we need the following two statements.

▶ **Theorem 11** (Turán's Theorem [33]). *Let $G = (V, E)$ be a graph with no clique of size $r + 1$. Then*

$$|E| \leq \left(1 - \frac{1}{r}\right) \frac{n^2}{2}.$$

▶ **Lemma 12** (Packing Lemma [20, 28, 29]). *Let $(X, \mathcal{S})$ be a set system with shatter function $\pi_{\mathcal{S}}(k) \leq c_1 \cdot k^d$ and let $\delta \in (1, |X|)$ be a parameter. Furthermore, let $\mathcal{P} \subset \mathcal{S}$ be a $\delta$-separated set; that is, $|S_1 \Delta S_2| \geq \delta$ for all $S_1, S_2 \in \mathcal{P}$ (where $S_1 \Delta S_2$ denotes the symmetric difference of $S_1$ and $S_2$). Then*

$$|\mathcal{P}| \leq 2c_1 \left(\frac{|X|}{\delta}\right)^d.$$

**Proof of Lemma 10.** Let $(\mathcal{S}_w, \mathcal{R}_Y)$ denote the set system dual to $(Y, \mathcal{S})$ with multiplicities given by $w(\cdot)$. That is, the base set $\mathcal{S}_w$ consists of sets of $\mathcal{S}$, where each $S \in \mathcal{S}$ has $w(S)$ copies in $\mathcal{S}_w$. And $\mathcal{R}_Y$ consists of $|Y|$ sets, one for each element of $Y$:

$$\mathcal{R}_Y = \left\{R_y : y \in Y\right\}, \quad \text{where} \quad R_y = \left\{S \in \mathcal{S}_w : y \in S\right\}.$$

Observe that

- for any $x, y \in Y$, the set $R_x \Delta R_y$ contains precisely the sets in $\mathcal{S}_w$ that cross the edge $xy$,
- $|\mathcal{S}_w| = w(\mathcal{S})$, and
- the shatter function of $(\mathcal{S}_w, \mathcal{R}_Y)$ is the dual shatter function of $(Y, \mathcal{S})$.

Consider a graph $G$ on $Y$, where there is an edge between two elements $x, y \in Y$ if and only if $xy$ is crossed by more than $\delta_\ell$ sets in $\mathcal{S}_w$, where we set

$$\delta_\ell = \left(10c_1 \cdot \frac{w(\mathcal{S})^d \ell}{|Y|^2}\right)^{1/d}.$$

Now the **Packing Lemma** implies that any $\delta_\ell$-separated subset of sets in $\mathcal{R}_Y$ has cardinality at most

$$C_\ell = 2c_1 \left(\frac{w(\mathcal{S})}{\delta_\ell}\right)^d = 2c_1 \frac{w(\mathcal{S})^d}{10c_1 \cdot \frac{w(\mathcal{S})^d \ell}{|Y|^2}} = \frac{|Y|^2}{5\ell}.$$

This implies that $G$ does not contain a clique on $C_\ell + 1$ vertices, and so by **Turán's Theorem**, the number of pairs that are *not* edges in $G$ is at least

$$\binom{|Y|}{2} - \left(1 - \frac{1}{C_\ell}\right) \frac{|Y|^2}{2} = \frac{|Y|^2}{2C_\ell} - \frac{|Y|}{2} = \frac{5\ell}{2} - \frac{|Y|}{2} \geq \ell,$$

where we used that $|Y| \leq \ell$. Thus, we have shown that there are at least $\ell$ edges which cross sets of total weight at most $\delta_\ell$. This concludes the proof of Lemma 10 and thus the proof of Lemma 8. ◀

## 4.2 Proof of Lemma 9

Our goal is to prove that if $E \subseteq \binom{X}{2}$ is a random set of edges, where each edge from $\binom{X}{2}$ is picked i.i.d. with probability

$$p = \min\left\{\frac{2\ln n}{n^{1-\alpha}} + \frac{4\ln(2/\delta)}{n^{2-\alpha}},\ 1\right\},$$

then the random halting time $T$ of $\textsc{RelaxedMWU}\big((X,\mathcal{S}),\alpha,E\big)$ satisfies

$$\mathbb{P}[\ T \le n/4\ ] \le \delta.$$

If $p = 1$, then the statement is trivially true, therefore assume that $p < 1$.

Note that we have two different sources of randomness: we run the algorithm on an initial sample $E$ of edges and at each iteration, if $E \cap \mathcal{E}_i \neq \emptyset$, we sample an edge $e_i \in E \cap \mathcal{E}_i$ uniformly at random. The notation $\mathbb{P}[A]$ denotes the probability of event $A$ under both randomness sources.

We will upper bound the probabilities $\mathbb{P}[\ T = i\ ]$ for each $i = 0, \ldots, n/4$. For the case $i = 0$, since $E$ is an i.i.d. uniform random sample of $\binom{X}{2}$, we have

$$\mathbb{P}[\ T = 0\ ] = \mathbb{P}[\ E \cap \mathcal{E}_1 = \emptyset\ ] = (1-p)^{|\mathcal{E}_1|} = (1-p)^{n^{2-\alpha}}.$$

Now consider the case $i \ge 1$. Observe that the edge-set $\mathcal{E}_i$ depends on the edges chosen in earlier iterations $j < i$. To signify this, for any sequence of $i-1$ edges $(e^1, \ldots, e^{i-1})$, let $\mathcal{E}_i(e^1, \ldots, e^{i-1})$ denote the set of $\big(n - 2(i-1)\big)^{2-\alpha}$ lightest edges *assuming* that $e^1, \ldots, e^{i-1}$ were added to the matching and the weights of the sets of $\mathcal{S}$ were adjusted multiplicatively accordingly.

We say that a sequence $(e^1, \ldots, e^i)$ is *feasible* if $e^1 \in \mathcal{E}_1,\ e^2 \in \mathcal{E}_2(e^1),\ \ldots,\ e^i \in \mathcal{E}_i(e^1, \ldots, e^{i-1})$. We denote the set of all feasible sequences of length $i$ by $\mathcal{C}^i$.

For a $\mathbf{c} \in \mathcal{C}^i$, we use the notation $\mathbf{c}^0 = \emptyset$, and $\mathbf{c}^j = (e^1, \ldots, e^j)$ for all $j \in [1, i]$; note that $\mathbf{c}^i = \mathbf{c}$. Let $\mathbf{e}_i = (e_1, \ldots, e_i)$ denote the sequence of *random variables* representing the edges actually chosen at each step by the algorithm up to iteration $i$, with $\mathbf{e}_0 = \emptyset$ and $\mathbf{e}_j = (e_1, \ldots, e_j)$ for all $j \in [1, i]$.

In the analysis, we will apply the law of total probability over the events that the algorithm picks the edges given by a certain feasible sequence $\mathbf{c} \in \mathcal{C}^i$, that is, $\mathbf{e}_i = \mathbf{c}^i$.

We break the analysis into three steps.

### 1. Unfolding the probability $\mathbb{P}[\ T = i\ ]$

Given the above notation, we have

$$\begin{aligned}
&\mathbb{P}[\ T = i\ ] = \mathbb{P}[\ E \cap \mathcal{E}_1 \neq \emptyset,\ \ldots,\ E \cap \mathcal{E}_i \neq \emptyset,\ E \cap \mathcal{E}_{i+1} = \emptyset\ ] \\
&= \sum_{\mathbf{c} \in \mathcal{C}^i} \mathbb{P}\left[\ E \cap \mathcal{E}_1(\mathbf{c}^0) \neq \emptyset,\ \ldots,\ E \cap \mathcal{E}_i(\mathbf{c}^{i-1}) \neq \emptyset,\ E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \ \middle|\ \mathbf{e}_i = \mathbf{c}^i\ \right] \cdot \mathbb{P}\left[\ \mathbf{e}_i = \mathbf{c}^i\ \right] \\
&\le \sum_{\mathbf{c} \in \mathcal{C}^i} \mathbb{P}\left[\ E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \ \middle|\ \mathbf{e}_i = \mathbf{c}^i\ \right] \cdot \mathbb{P}\left[\ \mathbf{e}_i = \mathbf{c}^i\ \right].
\end{aligned}$$

Note that $\mathcal{E}_{i+1}(\mathbf{c}^i)$ is a fixed set once we are given $\mathbf{c}^i = (e^1, \ldots, e^i)$. Using Bayes' theorem, we can express the conditional probabilities on the R.H.S. of the above inequality as

$$\mathbb{P}\left[\ E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \ \middle|\ \mathbf{e}_i = \mathbf{c}^i\ \right] = \frac{\mathbb{P}\left[\ \mathbf{e}_i = \mathbf{c}^i \ \middle|\ E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset\ \right] \cdot \mathbb{P}\left[\ E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset\ \right]}{\mathbb{P}[\ \mathbf{e}_i = \mathbf{c}^i\ ]}.$$

Thus, we get

$$\mathbb{P}\left[\, T = i \,\right] \le \sum_{\mathbf{c} \in \mathcal{C}^i} \mathbb{P}\left[\, \mathbf{e}_i = \mathbf{c}^i \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \,\right] \cdot \mathbb{P}\left[\, E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \,\right].$$

Fixing $\mathbf{e}_i = \mathbf{c}^i$ completely determines the set of edges in $\mathcal{E}_{i+1}(\mathbf{c}^i)$, and so

$$= \sum_{\mathbf{c} \in \mathcal{C}^i} \mathbb{P}\left[\, \mathbf{e}_i = \mathbf{c}^i \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \,\right] \cdot (1-p)^{|\mathcal{E}_{i+1}(\mathbf{c}^i)|}.$$

We now proceed by bounding the probability $\mathbb{P}\left[\, \mathbf{e}_i = \mathbf{c}^i \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \,\right]$, iteration by iteration:

$$\begin{aligned} &\mathbb{P}\left[\, \mathbf{e}_i = \mathbf{c}^i \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \,\right] \\ &= \mathbb{P}\left[\, e_i = e^i \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{i-1} = \mathbf{c}^{i-1} \,\right] \cdot \mathbb{P}\left[\, \mathbf{e}_{i-1} = \mathbf{c}^{i-1} \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset \,\right] \\ &= \ldots = \prod_{j=1}^{i} \mathbb{P}\left[\, e_j = e^j \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset, \mathbf{e}_{j-1} = \mathbf{c}^{j-1} \,\right], \end{aligned}$$

recalling that $\mathbf{e}_0 = \mathbf{c}^0 = \emptyset$. We conclude that

$$\mathbb{P}\left[\, T = i \,\right] \le \sum_{\mathbf{c} \in \mathcal{C}^i} (1-p)^{|\mathcal{E}_{i+1}(\mathbf{c}^i)|} \cdot \prod_{j=1}^{i} \mathbb{P}\left[\, e_j = e^j \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset, \mathbf{e}_{j-1} = \mathbf{c}^{j-1} \,\right]. \quad (3)$$

## 2. Bounding the probabilities $\mathbb{P}[\, e_j = e^j \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{j-1} = \mathbf{c}^{j-1}\,],\ j \in [1, i]$

Note that the condition $\mathbf{e}_{j-1} = \mathbf{c}^{j-1}$ fixes the set $\mathcal{E}_j(\mathbf{c}^{j-1})$. As $e_j$ was picked uniformly from $E \cap \mathcal{E}_j(\mathbf{c}^{j-1})$, we further condition on all possible choices of $E \cap \mathcal{E}_j(\mathbf{c}^{j-1})$, with the constraint that $E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset$:

$$\begin{aligned} &\mathbb{P}\left[\, e_j = e^j \mid E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{j-1} = \mathbf{c}^{j-1} \,\right] \\ &= \sum_{S' \subseteq \mathcal{E}_j(\mathbf{c}^{j-1})} \Big( \mathbb{P}\left[\, e_j = e^j \;\middle|\; E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{j-1} = \mathbf{c}^{j-1},\ E \cap \mathcal{E}_j(\mathbf{c}^{j-1}) = S' \,\right] \Big) \\ &\qquad \cdot \Big( \mathbb{P}\left[\, E \cap \mathcal{E}_j(\mathbf{c}^{j-1}) = S' \;\middle|\; E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{j-1} = \mathbf{c}^{j-1} \,\right] \Big) \end{aligned}$$

Note that if $S'$ is such that $e^j \notin S'$, then the first probability in the above product is 0. Similarly, if $S' \cap \mathcal{E}_{i+1}(\mathbf{c}^i) \neq \emptyset$, then the second probability is equal to 0. Continuing,

$$\begin{aligned} &= \sum_{\substack{S' \subseteq \mathcal{E}_j(\mathbf{c}^{j-1}) \setminus \mathcal{E}_{i+1}(\mathbf{c}^i): \\ e^j \in S'}} \Big( \mathbb{P}\left[\, e_j = e^j \;\middle|\; E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{j-1} = \mathbf{c}^{j-1},\ E \cap \mathcal{E}_j(\mathbf{c}^{j-1}) = S' \,\right] \Big) \\ &\qquad \cdot \Big( \mathbb{P}\left[\, E \cap \mathcal{E}_j(\mathbf{c}^{j-1}) = S' \;\middle|\; E \cap \mathcal{E}_{i+1}(\mathbf{c}^i) = \emptyset,\ \mathbf{e}_{j-1} = \mathbf{c}^{j-1} \,\right] \Big) \\ &= \sum_{\substack{S' \subseteq \mathcal{E}_j(\mathbf{c}^{j-1}) \setminus \mathcal{E}_{i+1}(\mathbf{c}^i): \\ e^j \in S'}} \left( \frac{1}{|S'|} \right) \cdot \left( p^{|S'|} \cdot (1-p)^{|\mathcal{E}_j(\mathbf{c}^{j-1}) \setminus \mathcal{E}_{i+1}(\mathbf{c}^i)| - |S'|} \right). \end{aligned}$$

Rearranging the sum by the sizes of the $S'$'s containing $e^j$:

$$= \sum_{\ell=1}^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|} \binom{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|-1}{\ell-1}\cdot\frac{1}{\ell}\cdot p^\ell\cdot(1-p)^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|-\ell}$$

$$= \sum_{\ell=1}^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|} \frac{\ell}{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\binom{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}{\ell} \cdot\frac{1}{\ell}\cdot p^\ell\cdot(1-p)^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|-\ell}$$

$$= \frac{1}{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\sum_{\ell=1}^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\binom{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}{\ell} \cdot p^\ell\cdot(1-p)^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|-\ell}.$$

Using the Binomial theorem, and adjusting for the case $\ell = 0$,

$$= \frac{1}{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\left((p+(1-p))^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}-(1-p)^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\right)$$

$$= \frac{1}{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\left(1-(1-p)^{|\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|}\right).$$

We can thus conclude that, setting $a = |\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|$,

$$\mathbb{P}\left[\ e_j = e^j \mid E\cap\mathcal{E}_{i+1}(\mathbf{c}^i)=\emptyset,\ \mathbf{e}_{j-1}=\mathbf{c}^{j-1}\ \right] = \frac{1}{a}\left(1-(1-p)^a\right)\le p, \tag{4}$$

where the last bound follows from Bernoulli's inequality $(1+x)^r \ge 1+rx$ for $x\ge -1$ and $r\ge 1$, which holds in our case since $x=-p\ge -1$ and $r = |\mathcal{E}_j(\mathbf{c}^{j-1})\setminus\mathcal{E}_{i+1}(\mathbf{c}^i)|\ge 1$.

### 3. Putting everything together

Let $k_i = \left|\mathcal{E}_i(\mathbf{c}^{i-1})\right| = |X_i|^{2-\alpha} = (n-2(i-1))^{2-\alpha}$.

Continuing Equation (3) together with the bound from Equation (4), we get

$$\mathbb{P}[\ T=i\ ] \le (1-p)^{k_{i+1}}\cdot\sum_{\mathbf{c}\in\mathcal{C}^i}\prod_{j=1}^{i} p \ = \ (1-p)^{k_{i+1}}\cdot k_1\cdot k_2\cdots k_i\cdot p^i.$$

Summing the above over all iterations,

$$\mathbb{P}[\ T\le i\ ] \le (1-p)^{k_1} + \sum_{\ell=1}^{i}(1-p)^{k_{\ell+1}}\cdot k_1\cdot k_2\cdots k_\ell\cdot p^\ell$$

Using that $k_1\ge k_2\ge\cdots\ge k_{i+1}$,

$$\mathbb{P}[\ T\le i\ ] \le (1-p)^{k_{i+1}} + \sum_{\ell=1}^{i}(1-p)^{k_{i+1}}\cdot k_1^\ell\cdot p^\ell = (1-p)^{k_{i+1}}\cdot\sum_{\ell=0}^{i}(k_1\cdot p)^\ell.$$

Thus, for $i=n/4$, using that $pk_1 = pn^{2-\alpha}\ge 2n\ln n\ge 2$, we obtain

$$\begin{aligned}\mathbb{P}[\ T\le n/4\ ] &\le (1-p)^{k_{n/4+1}}\sum_{\ell=0}^{n/4}(pk_1)^\ell \\ &= (1-p)^{k_{n/4+1}}\frac{(pk_1)^{n/4+1}-1}{pk_1-1} < (1-p)^{k_{n/4+1}}\frac{(pk_1)^{n/4+1}}{pk_1-1} \\ &= (1-p)^{k_{n/4+1}}\cdot\frac{(pk_1)}{pk_1-1}\cdot(pk_1)^{n/4} \le (1-p)^{k_{n/4+1}}\cdot 2\cdot(pk_1)^{n/4} \qquad \left(pk_1>2\right) \\ &\le \exp(-pk_{n/4+1})\cdot 2\cdot k_1^{n/4}.\end{aligned}$$

Substituting $k_1 = n^{2-\alpha}$, $k_{n/4+1} = (n/2)^{2-\alpha} \geq n^{2-\alpha}/4$ and $p = \frac{2\ln n}{n^{1-\alpha}} + \frac{4\ln(2/\delta)}{n^{2-\alpha}}$, we conclude

$$\mathbb{P}[\, T \leq n/4 \,] \leq 2 \exp\left(-\frac{n\ln n}{2} - \ln\frac{2}{\delta}\right) \cdot \left(n^{2-\alpha}\right)^{n/4} = 2 \cdot \frac{1}{n^{n/2}} \cdot \frac{\delta}{2} \cdot n^{n/2-\alpha n/4}$$
$$= n^{-\alpha n/4} \cdot \delta \leq \delta.$$

Therefore, with probability at least $1-\delta$, RelaxedMWU returns a matching of size at least $n/4$, which concludes the proof of Lemma 9. ◀

▶ Remark. In the last equation, to get failure probability at most $\delta$, we crucially need the fact that at the final iteration $i$, we still have $k_i \geq n^{2-\alpha}/4$, which limits the range of $i$. This is the technical reason why we can only guarantee that $E$ contains a good partial matching, but the analysis breaks for perfect matchings.

## 5 Proof of Optimality

In this section we present the proof of Lemma 2. Let $X$ be the set of $n = \left\lceil n_0^{1/d} \right\rceil^d$ points defined as $\left[\, 1, \left\lceil n_0^{1/d} \right\rceil \,\right] \times \cdots \times \left[\, 1, \left\lceil n_0^{1/d} \right\rceil \,\right] \subset \mathbb{Z}^d$, and let $\mathcal{S}$ consist of the $d \cdot \left\lfloor n_0^{1/d} \right\rfloor$ subsets of $X$ induced by half-spaces of the form

$$H_{i,j} = \left\{ x \in \mathbb{R}^d \ : \ x_i \leq j + 1/2 \right\}, \quad i = 1, \dots, d, \ j = 1, \dots \left\lfloor n_0^{1/d} \right\rfloor.$$

Observe that for any edge $\{x, y\} \in \binom{X}{2}$, the number of sets in $\mathcal{S}$ that crosses $\{x, y\}$ is precisely the $\ell_1$-distance[2] of $x$ and $y$. Using this observation, it is easy to see that for any $k \in \mathbb{N}^+$ and $x \in X$, the number of edges $\{x, y\}$ that are crossed by at most $k$ sets from $\mathcal{S}$ is $O(k^d)$. Thus, there is an absolute constant $c_0$ (depending on $d$) such that the total number of edges in $\binom{X}{2}$ crossed by at most $k$ sets from $\mathcal{S}$ is at most $c_0 \cdot nk^d$. We refer to these edges as *$k$-good* and denote their set with $\mathcal{G}_k$.

Let $p(n) = o(n^{\alpha-1})$ and $E$ be a uniform random sample of edges, where each edge of $\binom{X}{2}$ is picked with probability $p(n)$. Setting $k_{p,c}(n) = \left(\frac{1}{4c \cdot c_0 p(n)}\right)^{1/d}$, the expected number of $k_{p,c}(n)$-good edges in $E$ is

$$\mathbb{E}\left[\, |E \cap \mathcal{G}_{k_{p,c}(n)}| \,\right] \leq c_0 n \left(k_{p,c}(n)\right)^d \cdot p(n) = \frac{n}{4c}\ .$$

Thus, by Markov's inequality, we have $|E \cap \mathcal{G}_{k_{p,c}(n)}| \leq \frac{n}{2c}$ with probability at least $1/2$. Assume that $|E \cap \mathcal{G}_{k_{p,c}(n)}| \leq \frac{n}{2c}$ holds and let $M \subset E$ be any subset of size $\frac{n}{c}$. Then $M$ contains at least $\frac{n}{2c}$ edges which are not $k_{p,c}(n)$-good. Therefore, the number of crossings between the edges of $M$ and the sets of $\mathcal{S}$ is at least

$$\frac{n}{2c} \cdot \left(\frac{1}{4c \cdot c_0 \cdot p(n)}\right)^{1/d}\ .$$

Recall that $|\mathcal{S}| = d \cdot \left\lfloor n_0^{1/d} \right\rfloor \leq dn^{1/d}$ and so by the pigeonhole principle, we get that there is a set in $\mathcal{S}$ that crosses at least

[2] The $\ell_1$-distance of $x$ and $y$ is defined as $\ell_1(x, y) = \sum_{i=1}^{d} |x_i - y_i|$, where $x_i$ is the $i$-th coordinate of $x$.

$$\frac{\frac{n}{2c}\cdot\left(\frac{1}{4c\cdot c_0 p(n)}\right)^{1/d}}{|\mathcal{S}|} \geq \frac{\frac{n}{2c}\cdot\left(\frac{1}{4c\cdot c_0 p(n)}\right)^{1/d}}{dn^{1/d}} = \frac{1}{2c\cdot d\cdot(4c\cdot c_0)^{1/d}}\cdot n^{1-1/d}\underbrace{\left(\frac{1}{p(n)}\right)^{1/d}}_{\omega\left(n^{(1-\alpha)/d}\right)}$$
$$= \omega_{d,c}\cdot\left(n^{1-\alpha/d}\right)$$

edges of $M$. This concludes the proof of Lemma 2. ◀

## References


1 P. K. Agarwal and J. Matoušek. On range searching with semialgebraic sets. *Discrete & Computational Geometry*, 11(4):393–418, 1994.

2 P. K. Agarwal, J. Matoušek, and M. Sharir. On range searching with semialgebraic sets. II. *SIAM Journal on Computing*, 42(6):2039–2062, 2013.

3 N. Alon, S. Moran, and A. Yehudayoff. Sign rank versus VC dimension. In *COLT*, 2016.

4 N. Alon and J. H. Spencer. *The probabilistic method.* John Wiley & Sons, 2016.

5 N. Bansal. Constructive algorithms for discrepancy minimization. In *Proceedings of Symposium on Foundations of Computer Science, FOCS*, pages 3–10. IEEE Computer Society, 2010.

6 N. Bansal, D. Dadush, S. Garg, and S. Lovett. The Gram-Schmidt walk: a cure for the Banaszczyk blues. In *Proceedings of the Symposium on Theory of Computing, STOC*, pages 587–597, 2018.

7 N. Bansal and J. H. Spencer. Deterministic discrepancy minimization. *Algorithmica*, 67(4):451–471, 2013.

8 T. M. Chan. Optimal partition trees. *Discrete Comput. Geom.*, 47(4):661–690, 2012.

9 T. M. Chan, E. Grant, J. Könemann, and M. Sharpe. Weighted capacitated, priority, and geometric set cover via improved quasi-uniform sampling. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA 2012)*, pages 1576–1585, 2012.

10 B. Chazelle. *The Discrepancy Method: Randomness and Complexity.* Cambridge University Press, New York, NY, USA, 2000.

11 B. Chazelle and E. Welzl. Quasi-optimal range searching in spaces of finite VC-dimension. *Discrete Comput. Geom.*, pages 467–489, 1989.

12 C. Chekuri, J. Vondrák, and R. Zenklusen. Dependent randomized rounding for matroid polytopes and applications. *arXiv*, abs/0909.4348, 2009.

13 M. Csikós and N. H. Mustafa. Escaping the Curse of Spatial Partitioning: Matchings with Low Crossing Numbers and Their Applications. In *Proceedings of Symposium on Computational Geometry (SoCG 2021)*, volume 189, pages 28:1–28:17, 2021.

14 M. Csikós and N. H. Mustafa. Optimal approximations made easy. *Inf. Process. Lett.*, 176:106250, 2022.

15 Y. Deng, Z. Song, and O. Weinstein. Discrepancy minimization in input-sparsity time. *arXiv*, abs/2210.12468, 2022. doi:10.48550/arXiv.2210.12468.

16 G. Ducoffe, M. Habib, and L. Viennot. Diameter computation on h-minor free graphs and graphs of bounded (distance) VC-dimension. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1905–1922, 2020.

17 S. P. Fekete, M. E. Lübbecke, and H. Meijer. Minimizing the stabbing number of matchings, trees, and triangulations. *Discrete & Computational Geometry*, 40(4):595–621, 2008.

18 P. Giannopoulos, M. Konzack, and W Mulzer. Low-crossing spanning trees: an alternative proof and experiments. In *Proceedings of EuroCG*, 2014.

19 S. Har-Peled. Approximating spanning trees with low crossing number. *arXiv*, abs/0907.1131, 2009. arXiv:0907.1131.

**20** D. Haussler. 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.

**21** K. G. Larsen. Fast discrepancy minimization with hereditary guarantees. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023*, pages 276–289. SIAM, 2023. doi:10.1137/1.9781611977554.CH11.

**22** A. Levy, H. Ramadas, and T. Rothvoss. Deterministic discrepancy minimization via the multiplicative weight update method. In *Integer Programming and Combinatorial Optimization (IPCO)*, pages 380–391, 2017.

**23** S. Lovett and R. Meka. Constructive discrepancy minimization by walking on the edges. *SIAM J. Comput.*, 44(5):1573–1582, 2015.

**24** M. Matheny and J. M. Phillips. Practical low-dimensional halfspace range space sampling. In *Annual European Symposium on Algorithms (ESA)*, volume 112, pages 62:1–62:14, 2018.

**25** J. Matoušek. *Geometric Discrepancy: An Illustrated Guide*. Springer Berlin Heidelberg, 1999.

**26** J. Matoušek. *Lectures on discrete geometry*, volume 212. Springer Science & Business Media, 2013.

**27** J. Matoušek, E. Welzl, and L. Wernisch. Discrepancy and approximations for bounded VC-dimension. *Combinatorica*, 13(4):455–466, 1993.

**28** N. H. Mustafa. A simple proof of the shallow packing lemma. *Discret. Comput. Geom.*, 55(3):739–743, 2016.

**29** N. H. Mustafa. *Sampling in Combinatorial and Geometric Set Systems*. American Mathematical Society (AMS), 2022.

**30** T. Rothvoss. Constructive discrepancy minimization for convex sets. *SIAM J. Comput.*, 46(1):224–234, 2017.

**31** J. H. Spencer. Six standard deviations suffice. *Transactions of the American Mathematical Society*, 289:679–706, 1985.

**32** M. Talagrand. Sharper bounds for Gaussian and empirical processes. *Annals of Probability*, 22:28–76, 1994.

**33** P. Turán. On an external problem in graph theory. *Mat. Fiz. Lapok*, 48:436–452, 1941.

**34** K. R. Varadarajan. Weighted geometric set cover via quasi-uniform sampling. In *Proceedings of ACM Symposium on Theory of Computing (STOC 2010)*, pages 641–648, 2010.

**35** E. Welzl. Partition trees for triangle counting and other range searching problems. In *Proceedings of Annual Symposium on Computational Geometry (SoCG 1988)*, pages 23–33, 1988.

**36** E. Welzl. On spanning trees with low crossing numbers. In *Data Structures and Efficient Algorithms, Final Report on the DFG Special Joint Initiative*, pages 233–249, 1992.


# A Appendix

## A.1 Proof of Corollary 3

The algorithm achieving the guarantees of Corollary 3 is presented in MatchingPresampled. It is essentially the algorithm presented in [13] run on an initial random sample of edges with a small modification: to incorporate the pre-sampling step in the analysis, we need to recurse slightly more often (after $n/16$ steps instead of $n/4$).

We use the following key lemma for PartialMatching.

▶ **Lemma 13** ([13]). *Let $\tilde{E} \subset E$ denote the set of edges that have non-zero weight when* PartialMatching$\big((X, \mathcal{S}), E, \kappa\big)$ *terminates. Then*

$$\mathbb{E}\left[\max_{S \in \mathcal{S}} \sum_{i=1}^{n/16} \mathrm{I}\left(e_i, S\right)\right] \leq 2 \cdot \mathbb{E}\left[\min_{e \in \tilde{E}} \sum_{i=1}^{n/16} \mathrm{I}\left(e, S_i\right)\right] + O\left(\kappa + \ln|E| + \ln|\mathcal{S}|\right). \tag{5}$$

**Algorithm 3** MatchingPresampled$\big((X,\mathcal{S}),d,\alpha\big)$.

```
M ← ∅
while |X| > 16 do
    n ← |X|
    E ← sample of O(n^{1+α} ln n) edges from (X choose 2)
    {e_1, ..., e_{n/16}} ← PartialMatching((X,S), E, (n/16)^{1-α/d})
    M ← M ∪ {e_1, ..., e_{n/16}}
    X ← X \ endpoints(M)
match the remaining elements of X randomly and add the edges to M
return M
```

**Algorithm 4** PartialMatching$\big((X,\mathcal{S}),E,\kappa\big)$.

```
ω_1(e) ← 1,  π_1(S) ← 1    ∀e ∈ E, S ∈ 𝒮
p ← min{106 · |X|/κ^2 · ln(|E| · |X|/16), 1}
q ← min{39 · |X|/κ^2 · ln(|𝒮| · |X|/16), 1}
for i = 1, ..., |X|/16 do
    ω_i(E) ← Σ_{e∈E} ω_i(e)
    π_i(𝒮) ← Σ_{S∈𝒮} π_i(S)
    choose e_i ~ ω_i ;                           // ℙ[e_i = e] = ω_i(e)/ω_i(E)   ∀e ∈ E
    choose S_i ~ π_i ;                           // ℙ[S_i = S] = π_i(S)/π_i(𝒮)   ∀S ∈ 𝒮
    E_i ← sample from E with probability p ;     // ℙ[e ∈ E_i] = p   ∀e ∈ E
    𝒮_i ← sample from 𝒮 with probability q ;     // ℙ[S ∈ 𝒮_i] = q   ∀S ∈ 𝒮
    ;                  // I(e,S) = 1 if e crosses S, I(e,S) = 0 otherwise
    for e ∈ E_i do
        ω_{i+1}(e) ← ω_i(e)(1 − ½ I(e,S_i)) ;    // halve weight if S_i crosses e
    for S ∈ 𝒮_i do
        π_{i+1}(S) ← π_i(S)(1 + I(e_i,S)) ;      // double weight if S crosses e_i
    set the weight in ω_{i+1} of e_i and of each edge adjacent to e_i to zero
return {e_1, ..., e_{|X|/16}}
```

In MatchingPresampled, the subroutine PartialMatching is called with the parameter $\kappa = (n/16)^{1-\alpha/d}$, thus we get the following bound on the expected crossing number of $\{e_1,\ldots,e_{n/16}\}$:

$$\mathbb{E}\left[\max_{S\in\mathcal{S}}\sum_{i=1}^{n/16}\mathrm{I}(e_i,S)\right] \le \frac{1}{2}\cdot\mathbb{E}\left[\min_{e\in\tilde{E}}\sum_{i=1}^{n/16}\mathrm{I}(e,S_i)\right] + O\left(n^{1-\alpha/d}\right). \tag{6}$$

The left-hand side of Equation (5) is precisely the expected crossing number of the edges returned by PartialMatching. It remains to bound the expectation on the right-hand side of Equation (6). By Lemma 1, with probability at least $1-\frac{1}{n}$, the initial sample $E$ contains a matching $M_0$ of size $\lceil n/4\rceil$ with crossing number

$$c_0\cdot\left(n^{1-\alpha/d}+\ln|\mathcal{S}|\right)$$

for some fixed constant $c_0$. Assume that it happens, then clearly $M_0\cap\tilde{E}$ also has crossing number at most $c_0\cdot\left(n^{1-\alpha/d}+\ln|\mathcal{S}|\right)$ with respect to $\mathcal{S}$. Moreover, since we only zeroed the weights of edges adjacent to $2\cdot n/16$ distinct vertices of $X$, there are at least $n/8$ edges of

$M_0$ with positive weight when PartialMatching terminates. That is, $|M_0 \cap \tilde{E}| \geq n/8$ and $n/8 > 0$ since $n > 16$ at each call of PartialMatching . By the pigeonhole principle, there is an edge in $M_0 \cap \tilde{E}$ which is crossed by at most

$$\frac{c_0 \cdot (n^{1-\alpha/d} + \ln|\mathcal{S}|) \cdot n/16}{n/8} = O\left(n^{1-\alpha/d} + \ln|\mathcal{S}|\right)$$

sets from $S_1, \ldots, S_{n/16}$. Therefore, we have

$$\mathbb{E}\left[\max_{S\in\mathcal{S}} \sum_{i=1}^{n/16} \mathrm{I}(e_i, S)\right] \leq \frac{1}{2}\cdot\mathbb{E}\left[\min_{e\in\tilde{E}} \sum_{i=1}^{n/16} \mathrm{I}(e, S_i)\right] + O\left(n^{1-\alpha/d}\right) = O\left(n^{1-\alpha/d} + \ln|\mathcal{S}|\right),$$

where the last bound holds with with probability at least $1 - \frac{1}{n}$. Since the crossing number of any matching of $X$ is at most $n/2$, the expected crossing number of the matching returned by the subroutine PartialMatching$\big((X, \mathcal{S}), E, (n/16)^{1-\alpha/d}\big)$ is $O\left(n^{1-\alpha/d} + \ln|\mathcal{S}|\right)$.

The bottleneck algorithmic step in PartialMatching is to update the weights of edges and sets belonging to $E_i$ and $\mathcal{S}_i$ at each iteration $i = 1, \ldots, n/16$.

For any $i$, we have $\mathbb{E}[\,|E_i| + |\mathcal{S}_i|\,] = \tilde{O}\left(n^{1+\alpha}\mathbf{p} + m\mathbf{q}\right)$, thus in expectation, the total running time is $O\left(n\left(n^{1+\alpha}\cdot\min\{n/\kappa^2\ln n,\, 1\} + m\cdot\min\{n/\kappa^2\ln m,\, 1\}\right)\right) = \tilde{O}\left(n^{1+\alpha+\frac{2\alpha}{d}} + mn^{\frac{2\alpha}{d}}\right)$.

The algorithm MatchingPresampled makes $\log n$ calls to PartialMatching with exponentially decreasing input sizes. It can easily be deduced that MatchingPresampled returns a matching with expected crossing number $O\left(n^{1-\alpha/d} + \ln|\mathcal{S}|\ln n\right)$ in expected time $\tilde{O}\left(n^{1+\alpha+\frac{2\alpha}{d}} + mn^{\frac{2\alpha}{d}}\right)$. This concludes the proof of Corollary 3. ◀

## A.2 Proof of Corollary 5

Now we deduce how Corollary 3 implies Corollary 5. The randomized algorithm that achieves the guarantees of Corollary 5 is presented in Algorithm 5.

**Algorithm 5** LowDiscColorPresampled$\big((X, \mathcal{S}), d, \alpha\big)$.

```
n ← |X|
{e_1, ..., e_⌈n/2⌉} ← MatchingPresampled((X, S), d, α)
for i = 1, ..., ⌈n/2⌉ do
    {x_i, y_i} ← endpoints(e_i)
    χ(x_i) = { 1   with probability 1/2
             { −1  with probability 1/2
    χ(y_i) = −χ(x_i) ;                       // we skip this step if y_i = x_i
return χ
```

▶ **Lemma 14.** *Let $(X, \mathcal{S})$ be a set system, $n = |X|$, $m = |\mathcal{S}| \geq 34$, and let $M$ be a perfect matching of $X$ with crossing number $\kappa$ with respect to $\mathcal{S}$ and for each edge $\{x, y\} \in M$ define*

$$\chi_M(x) = \begin{cases} 1 & \textit{with probability } 1/2 \\ -1 & \textit{with probability } 1/2 \end{cases}$$

*and $\chi_M(y) = -\chi_M(x)$. Then the expected discrepancy of $\chi_M$ is at most $\sqrt{3\kappa\ln m}$.*

▶ Remark. A "high probability version" of Lemma 14 is well-known [27, Lemma 2.5] and implies the above bound.

Corollary 3 and Lemma 14 immediately imply that the algorithm LowDiscColorPresampled constructs a coloring with expected discrepancy $O\left(\sqrt{n^{1-\alpha/d}\ln m + \ln^2 m \log n}\right)$, in time $\tilde{O}\left(n^{1+\alpha+2\alpha/d} + |\mathcal{S}| \cdot n^{2\alpha/d}\right)$. This concludes the proof of Corollary 5. ◀

**Proof of Lemma 14.** Let $S \in \mathcal{S}$ be a fixed range. We express the sum $\chi_M(S)$ of colors over elements of $S$ as

$$\chi_M(S) = \sum_{\{x,y\}\in M; x,y\in S} (\chi_M(x) + \chi_M(y)) + \sum_{x\in \mathrm{cr}(S,M)} \chi_M(x) = \sum_{x\in \mathrm{cr}(S,M)} \chi_M(x) \ ,$$

where $\mathrm{cr}(S,M) = \{x \in S : \{x,y\} \in M, y \notin S\}$. Since $\mathrm{cr}(S,M) \le \kappa$ for any $S \in \mathcal{S}$, $\mathrm{disc}(S, \chi_M)$ is a sum of at most $\kappa$ *independent* random variables. We use the following concentration bound from [4].

▷ Claim 15 (Theorem A.1.1 from [4]). Let $X_1, \ldots, X_k$ be independent $\{-1,1\}$-valued random variables with $\mathbb{P}[X_i = -1] = \mathbb{P}[X_i = 1] = 1/2$. Then for any $\alpha \ge 0$

$$\mathbb{P}\left[\ \left|\sum_{i=1}^{k} X_i\right| > \alpha\ \right] \le 2e^{-\alpha^2/2k}.$$

Applying Claim 15, we get that for any fixed $S \in \mathcal{S}$ and $\alpha > 0$,

$$\mathbb{P}[\ |\chi_M(S)| > \alpha\ ] \le 2e^{-\alpha^2/2\kappa}.$$

By the union bound, we get

$$\mathbb{P}[\ \mathrm{disc}_{\mathcal{S}}(\chi_M) > \alpha\ ] = \mathbb{P}\left[\ \max_{S\in\mathcal{S}} |\chi_M(S)| > \alpha\ \right] \le m \cdot 2e^{-\alpha^2/2\kappa}.$$

Finally, we bound the expected discrepancy by applying Fubini's theorem

$$\mathbb{E}[\ \mathrm{disc}_{\mathcal{S}}(\chi_M)\ ] \stackrel{\text{def}}{=} \int_0^\infty \mathbb{P}[\ \mathrm{disc}_{\mathcal{S}}(\chi_M) > \alpha\ ]\, d\alpha \le \int_0^\infty \min\left\{2m \cdot e^{-\alpha^2/2\kappa},\ 1\right\} d\alpha$$

$$= \int_0^{\sqrt{2\kappa\ln(2m)}} 1 d\alpha + \int_{\sqrt{2\kappa\ln(2m)}}^\infty 2m \cdot e^{-\alpha^2/2\kappa} d\alpha = \sqrt{2\kappa\ln(2m)} + 2m\sqrt{2\kappa} \int_{\sqrt{\ln(2m)}}^\infty e^{-t^2}\ dt$$

$$= \sqrt{2\kappa\ln(2m)} + 2m\sqrt{2\kappa} \int_{\sqrt{\ln(2m)}}^\infty \frac{t}{t} \cdot e^{-t^2}\ dt \le \sqrt{2\kappa\ln(2m)} + 2m\sqrt{\frac{2\kappa}{\ln(2m)}} \int_{\sqrt{\ln(2m)}}^\infty te^{-t^2}\ dt$$

$$= \sqrt{2\kappa\ln(2m)} + 2m\sqrt{\frac{2\kappa}{\ln(2m)}} \left[\ -\frac{e^{-t^2}}{2}\ \right]_{\sqrt{\ln(2m)}}^\infty = \sqrt{2\kappa\ln(2m)} + \sqrt{\frac{\kappa}{2\ln(2m)}} \le \sqrt{3\kappa\ln m},$$

if $m \ge 34$. This concludes the proof of Lemma 14. ◀