A Greedy Algorithm for Low-Crossing Partitions for General Set Systems

Mónika Csikós[1] Alexandre Louvet[2] Nabil Mustafa[2]

[1]IRIF, Université Paris Cité

[2]LIPN, Université Sorbonne Paris Nord

January 13th 2025

Preliminaries

Let a set system be a ground set X and a collection of ranges (subsets) $\mathcal{F}$. We denote $|X| = n, |\mathcal{F}| = m$.

Preliminaries

Let a set system be a ground set X and a collection of ranges (subsets) $\mathcal{F}$. We denote $|X| = n, |\mathcal{F}| = m$.

The VC-dimension of a set system is a measure of its complexity.

Preliminaries

Let a set system be a ground set X and a collection of ranges (subsets) $\mathcal{F}$. We denote $|X| = n, |\mathcal{F}| = m$.

The VC-dimension of a set system is a measure of its complexity.

Set systems *spanned* by halfspaces in $\mathbb{R}^d$ has VC-dimension $d + 1$.

Preliminaries

Let a set system be a ground set X and a collection of ranges (subsets) $\mathcal{F}$. We denote $|X| = n, |\mathcal{F}| = m$.

The VC-dimension of a set system is a measure of its complexity.

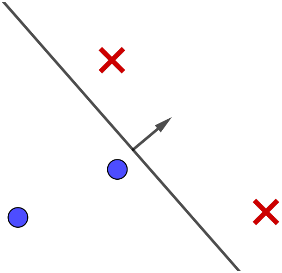
Set systems *spanned* by halfspaces in $\mathbb{R}^d$ has VC-dimension $d + 1$.
$\Rightarrow F \in \mathcal{F}$ iff $\exists \mathcal{H}$ s.t. $F = X \cap \mathcal{H}$

Preliminaries

Let a set system be a ground set X and a collection of ranges (subsets) $\mathcal{F}$. We denote $|X| = n, |\mathcal{F}| = m$.

The VC-dimension of a set system is a measure of its complexity.

Set systems *spanned* by halfspaces in $\mathbb{R}^d$ has VC-dimension $d+1$.
$\Rightarrow F \in \mathcal{F}$ iff $\exists \mathcal{H}$ s.t. $F = X \cap \mathcal{H}$

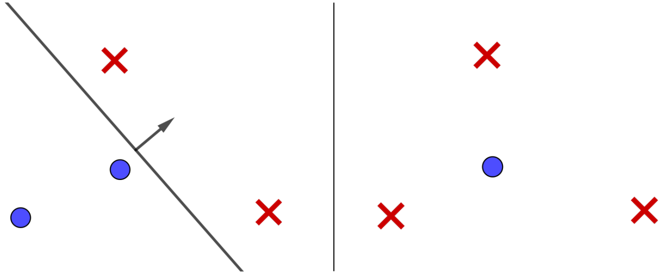


Preliminaries

Let a set system be a ground set X and a collection of ranges (subsets) $\mathcal{F}$. We denote $|X| = n, |\mathcal{F}| = m$.

The VC-dimension of a set system is a measure of its complexity.

Set systems *spanned* by halfspaces in $\mathbb{R}^d$ has VC-dimension $d + 1$.
$\Rightarrow F \in \mathcal{F}$ iff $\exists \mathcal{H}$ s.t. $F = X \cap \mathcal{H}$



Low-crossing partitions

Low-crossing Partitions

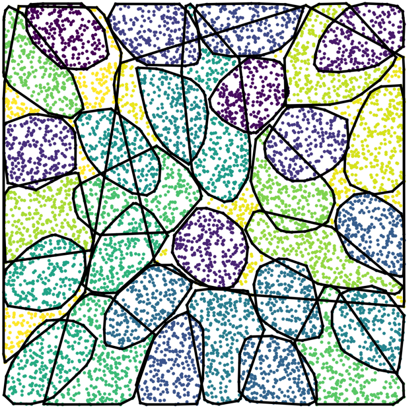
Given $(X, \mathcal{F})$ and a parameter $t \in [2..\frac{n}{2}]$, find a partition of X in t parts $P_1, .., P_t$ of size $O\left(\frac{n}{t}\right)$ to minimize the *crossing number*:

$$\max_{F \in \mathcal{F}} |\{P \in \{P_1, ...P_t\} \text{ s.t. } P \text{ crosses } F\}|$$

Low-crossing Partitions

Given $(X, \mathcal{F})$ and a parameter $t \in [2..\frac{n}{2}]$, find a partition of X in t parts $P_1, .., P_t$ of size $O\left(\frac{n}{t}\right)$ to minimize the *crossing number*:

$$\max_{F \in \mathcal{F}} |\{P \in \{P_1, ...P_t\} \text{ s.t. } P \text{ crosses } F\}|$$



A partition of points in the unit cube and random halfspaces

Previous work

- [Mat92] showed the existence of partitions with crossing number $O(t^{1-1/d})$ for set systems spanned by halfspaces in $\mathbb{R}^d$ using cuttings.

Previous work

- [Mat92] showed the existence of partitions with crossing number $O(t^{1-1/d})$ for set systems spanned by halfspaces in $\mathbb{R}^d$ using cuttings.
- [Cha93, HP00] provided implementations of cuttings algorithms.

Previous work

- [Mat92] showed the existence of partitions with crossing number $O(t^{1-1/d})$ for set systems spanned by halfspaces in $\mathbb{R}^d$ using cuttings.
- [Cha93, HP00] provided implementations of cuttings algorithms.
- [MP18] extended these to provide an implementation of partitions.

Previous work

- [Mat92] showed the existence of partitions with crossing number $O(t^{1-1/d})$ for set systems spanned by halfspaces in $\mathbb{R}^d$ using cuttings.
- [Cha93, HP00] provided implementations of cuttings algorithms.
- [MP18] extended these to provide an implementation of partitions.

There are neither fast algorithms in practice in dimension > 3 to compute small crossing number partitions for geometric set systems nor for abstract set systems in any dimension.

A greedy approach

A greedy approach

An Ordering Lemma (simplified)

A greedy approach

An Ordering Lemma (simplified)

Let $(X, \mathcal{F})$ be a set system such that $\forall Y \subseteq X$ and $s \leq |Y|, (Y, \mathcal{F}|_Y)$ admits a partition of size s with crossing number $s^{1-1/d}$.

A greedy approach

An Ordering Lemma (simplified)

Let $(X, \mathcal{F})$ be a set system such that $\forall Y \subseteq X$ and $s \leq |Y|, (Y, \mathcal{F}|_Y)$ admits a partition of size s with crossing number $s^{1-1/d}$.

For any fixed $t \in [2, n/2], \mathcal{P} = \{P_1, \ldots, P_t\}$ be t disjoint subsets of X, where $|P_i| = n/t$ for all $i \in [t]$.

A greedy approach

An Ordering Lemma (simplified)

Let $(X, \mathcal{F})$ be a set system such that $\forall Y \subseteq X$ and $s \leq |Y|, (Y, \mathcal{F}|_Y)$ admits a partition of size s with crossing number $s^{1-1/d}$.

For any fixed $t \in [2, n/2], \mathcal{P} = \{P_1, \ldots, P_t\}$ be t disjoint subsets of X, where $|P_i| = n/t$ for all $i \in [t]$.

Let $P_l \in \mathcal{P}$ be selected uniformly at random.

A greedy approach

An Ordering Lemma (simplified)

Let $(X, \mathcal{F})$ be a set system such that $\forall Y \subseteq X$ and $s \leq |Y|, (Y, \mathcal{F}|_Y)$ admits a partition of size s with crossing number $s^{1-1/d}$.

For any fixed $t \in [2, n/2], \mathcal{P} = \{P_1, \ldots, P_t\}$ be t disjoint subsets of X, where $|P_i| = n/t$ for all $i \in [t]$.

Let $P_l \in \mathcal{P}$ be selected uniformly at random.

Then there exists an ordering of the elements of P_l, say $\langle x_1, x_2, \ldots, x_{n/t} \rangle$, such that w.h.p.,

A greedy approach

An Ordering Lemma (simplified)

Let $(X, \mathcal{F})$ be a set system such that $\forall Y \subseteq X$ and $s \leq |Y|, (Y, \mathcal{F}|_Y)$ admits a partition of size s with crossing number $s^{1-1/d}$.

For any fixed $t \in [2, n/2], \mathcal{P} = \{P_1, \ldots, P_t\}$ be t disjoint subsets of X, where $|P_i| = n/t$ for all $i \in [t]$.

Let $P_I \in \mathcal{P}$ be selected uniformly at random.

Then there exists an ordering of the elements of P_I, say $\langle x_1, x_2, \ldots, x_{n/t} \rangle$, such that w.h.p.,

$\forall k \leq \frac{n}{t}$, the prefix set $\{x_1, \ldots, x_k\}$ is crossed by at most $\dfrac{4mk^{1/d}}{n^{1/d}}$ sets.

A greedy approach

An Ordering Lemma (simplified)

Let $(X, \mathcal{F})$ be a set system such that $\forall Y \subseteq X$ and $s \leq |Y|, (Y, \mathcal{F}|_Y)$ admits a partition of size s with crossing number $s^{1-1/d}$.

For any fixed $t \in [2, n/2], \mathcal{P} = \{P_1, \ldots, P_t\}$ be t disjoint subsets of X, where $|P_i| = n/t$ for all $i \in [t]$.

Let $P_l \in \mathcal{P}$ be selected uniformly at random.

Then there exists an ordering of the elements of P_l, say $\langle x_1, x_2, \ldots, x_{n/t} \rangle$, such that w.h.p.,

$\forall k \leq \frac{n}{t}$, the prefix set $\{x_1, \ldots, x_k\}$ is crossed by at most $\frac{4mk^{1/d}}{n^{1/d}}$ sets.

An algorithm that picks elements following this potential function builds partitions with crossing number $O(t^{1-1/d} \ln(t))$.

A greedy algorithm

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

- We maintain weights on ranges initially all equal to 1.

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

- We maintain weights on ranges initially all equal to 1.
- To build a part P_i:

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

- We maintain weights on ranges initially all equal to 1.
- To build a part P_i:
  - $\rightarrow$ We select a root $r \in X \backslash (P_1 \cup \ldots \cup P_{i-1})$ at random.

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

- We maintain weights on ranges initially all equal to 1.
- To build a part P_i:
  - → We select a root $r \in X \backslash (P_1 \cup ... \cup P_{i-1})$ at random.
  - → We compute the cost of elements $x \in X \backslash (P_1 \cup ... \cup P_{i-1})$ that is the sum of weight of ranges the edge r, x crosses.

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

- We maintain weights on ranges initially all equal to 1.
- To build a part P_i:
  - → We select a root $r \in X \backslash (P_1 \cup ... \cup P_{i-1})$ at random.
  - → We compute the cost of elements $x \in X \backslash (P_1 \cup ... \cup P_{i-1})$ that is the sum of weight of ranges the edge r, x crosses.
  - → We select an arbitrary element that keeps the cost of part (sum of costs of elements of the part) below the potential function.

A greedy algorithm

We reuse the Multiplicative Weight idea from Matousek:

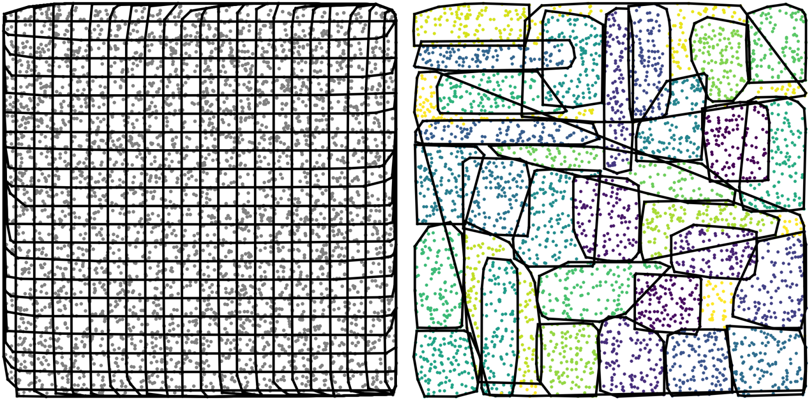
- We maintain weights on ranges initially all equal to 1.
- To build a part P_i:
  - → We select a root $r \in X \backslash (P_1 \cup \ldots \cup P_{i-1})$ at random.
  - → We compute the cost of elements $x \in X \backslash (P_1 \cup \ldots \cup P_{i-1})$ that is the sum of weight of ranges the edge r, x crosses.
  - → We select an arbitrary element that keeps the cost of part (sum of costs of elements of the part) below the potential function.
- We double the weight of ranges crossing the part built.

Experiments

Experimental Setup

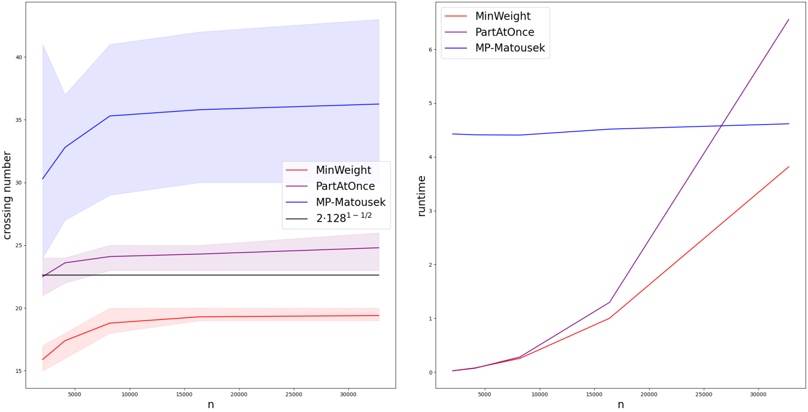
- We propose two variations of the greedy algorithms:
  - → MinWeight: We pick the element minimizing cost at each iteration.
  - → PartAtOnce: We estimate weights with range sampling and pick the n/t elements with lowest estimated weight as a part.
- Experiments results are obtained from average over 10 runs.
- Experiments have been performed on AMD Ryzen 7 5800X (16 cores) @ 4.85 GHz (Home computer).

Experiments



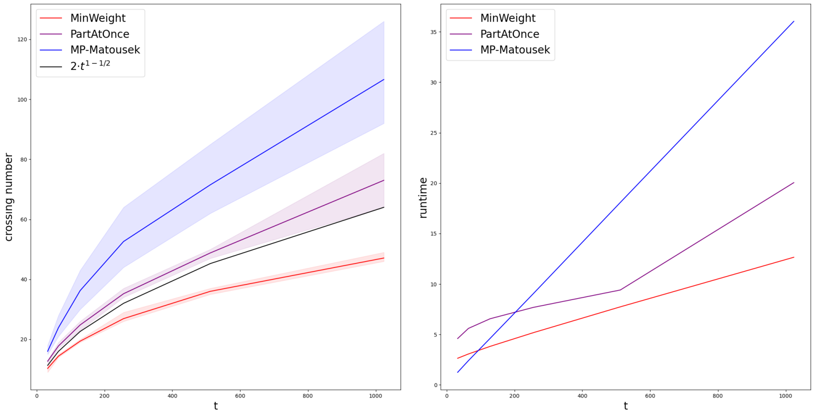
$n = 8192$, $t = 32$, crossing number : 10

Comparison with previous methods (n)



Comparison of crossing number and runtime of our algorithms for varying n and $t = 128, d = 2$.

Comparison with previous methods (t)



Comparison of crossing number and runtime of our algorithms for varying t and $n = 32768, d = 2$.

Real world datasets: ArXiv co-authorship graph

Real world datasets: ArXiv co-authorship graph

Input t	$t^{4/5}$	MinWeight $\kappa_{\mathcal{F}}$	MinWeight runtime (s)	PartAtOnce $\kappa_{\mathcal{F}}$	PartAtOnce runtime (s)
50	23	12	0.973	21	0.182
100	40	16	1.08	28	0.256
200	69	19	1.17	34	0.373
500	144	26	1.46	41	0.74

VC-dimension: 5 [CCDV24]

Summary:

Summary:

- We prove the existence of a potential function to build low-crossing partitions iteratively.

Summary:

- We prove the existence of a potential function to build low-crossing partitions iteratively.
- We build the first practically fast algorithm to compute low-crossings partitions of abstract set systems.

Summary:

- We prove the existence of a potential function to build low-crossing partitions iteratively.
- We build the first practically fast algorithm to compute low-crossings partitions of abstract set systems.
- It has performances matching the performances of previous algorithm on set systems we could compare them.

Summary:

- We prove the existence of a potential function to build low-crossing partitions iteratively.
- We build the first practically fast algorithm to compute low-crossings partitions of abstract set systems.
- It has performances matching the performances of previous algorithm on set systems we could compare them.

The code is available online
https://github.com/alex-louvet/partitions

Summary:

- We prove the existence of a potential function to build low-crossing partitions iteratively.
- We build the first practically fast algorithm to compute low-crossings partitions of abstract set systems.
- It has performances matching the performances of previous algorithm on set systems we could compare them.

The code is available online
https://github.com/alex-louvet/partitions

Thank You!

More results: ε-approximations

ε − *approximation*

Given $(X, \mathcal{F})$, an ε-approximation is a set A such that:

$$\forall F \in \mathcal{F}, \left| \frac{|F|}{|X|} - \frac{|F \cap A|}{|A|} \right| \leq \varepsilon$$

$\varepsilon - approximation$

Given $(X, \mathcal{F})$, an ε-approximation is a set A such that:

$$\forall F \in \mathcal{F}, \left| \frac{|F|}{|X|} - \frac{|F \cap A|}{|A|} \right| \leq \varepsilon$$

[Tal94, LLS01] proved that for set systems with VC-dimension $\leq d$, a uniform random sample of X of size $O\left(\frac{d}{\varepsilon^2}\right)$ is an ε-approximation.

ε − *approximation*

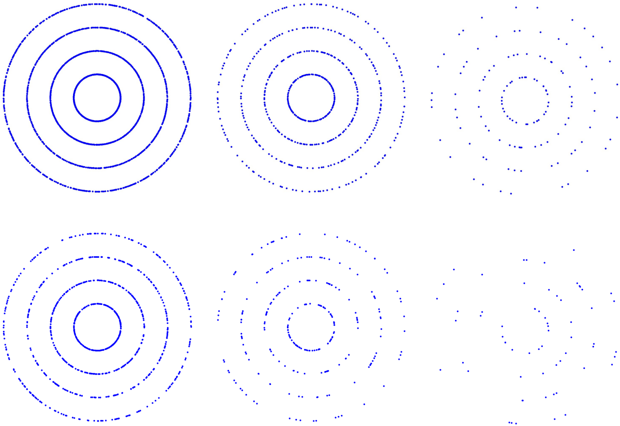
Given $(X, \mathcal{F})$, an ε-approximation is a set A such that:

$$\forall F \in \mathcal{F}, \left| \frac{|F|}{|X|} - \frac{|F \cap A|}{|A|} \right| \leq \varepsilon$$

[Tal94, LLS01] proved that for set systems with VC-dimension $\leq d$, a uniform random sample of X of size $O\left(\frac{d}{\varepsilon^2}\right)$ is an ε-approximation.

[STZ06] proved that it is possible to construct an ε-approximation of size $O\left(\frac{d}{\varepsilon^{\frac{2d}{d+1}}}\right)$ using low-crossing partitions for set systems with VC-dim $\leq d$.

ε-approximation experiments



our algorithm

random sample [Tal94]

$|A| = \frac{n}{4}$ $|A| = \frac{n}{16}$ $|A| = \frac{n}{64}$

References

References I

David Coudert, Mónika Csikós, Guillaume Ducoffe, and Laurent Viennot.
Practical Computation of Graph VC-Dimension.
In *Symposium on Experimental Algorithms (SEA)*, July 2024.

B. Chazelle.
Cutting hyperplanes for divide-and-conquer.
Discrete & Computational Geometry, 9, February 1993.

S. Har-Peled.
Constructing planar cuttings in theory and practice.
SIAM J. Comput., 29, 2000.

References II

Yi Li, Philip M. Long, and Aravind Srinivasan.
Improved Bounds on the Sample Complexity of Learning.
Journal of Computer and System Sciences, 62(3):516–527, May 2001.

Jiří Matoušek.
Efficient partition trees.
Discrete & Computational Geometry, 8:315–334, 1992.

Michael Matheny and Jeff M Phillips.
Practical low-dimensional halfspace range space sampling.
In *26th Annual European Symposium on Algorithms (ESA)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

References III

Subhash Suri, Csaba D Toth, and Yunhong Zhou.
Range counting over multidimensional data streams.
Discrete & Computational Geometry, 36:633–655, 2006.

Michel Talagrand.
Sharper bounds for gaussian and empirical processes.
The Annals of Probability, pages 28–76, 1994.