Approximate Minimum-Weight Partial Matching under Translation

Pankaj Agarwal [1] Haim Kaplan [2] Geva Kipper [2]
Wolfgang Mulzer [3] Günter Rote [3] Micha Sharir [2] Allen Xiao [1]

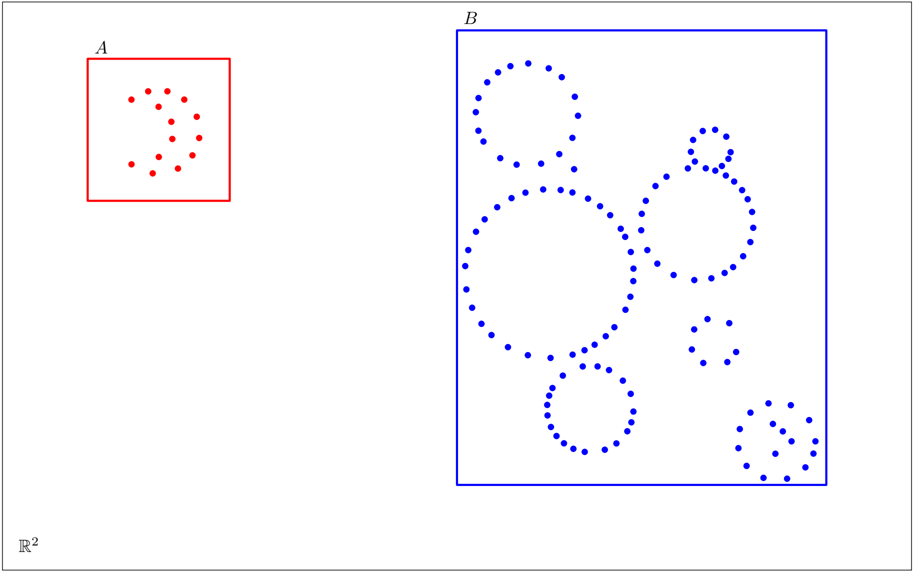
[1]Duke University

[2]Tel Aviv University

[3]Freie Universität Berlin

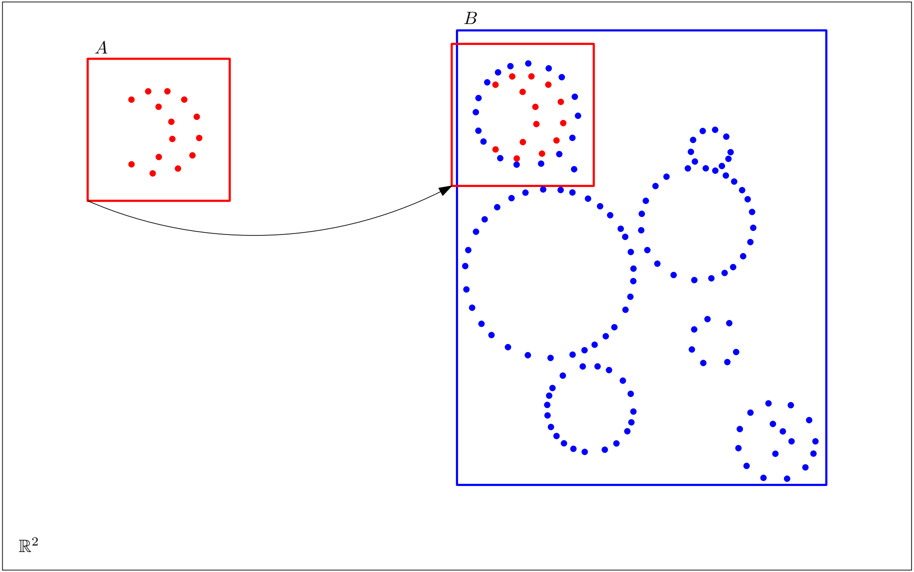
December 2018

Example



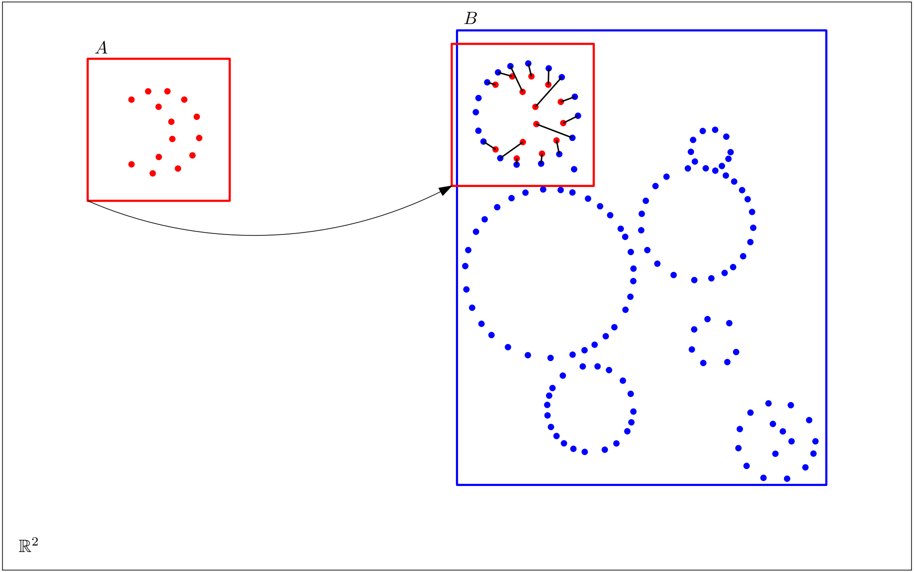
- Find the minimum cost matching over all translations

Example



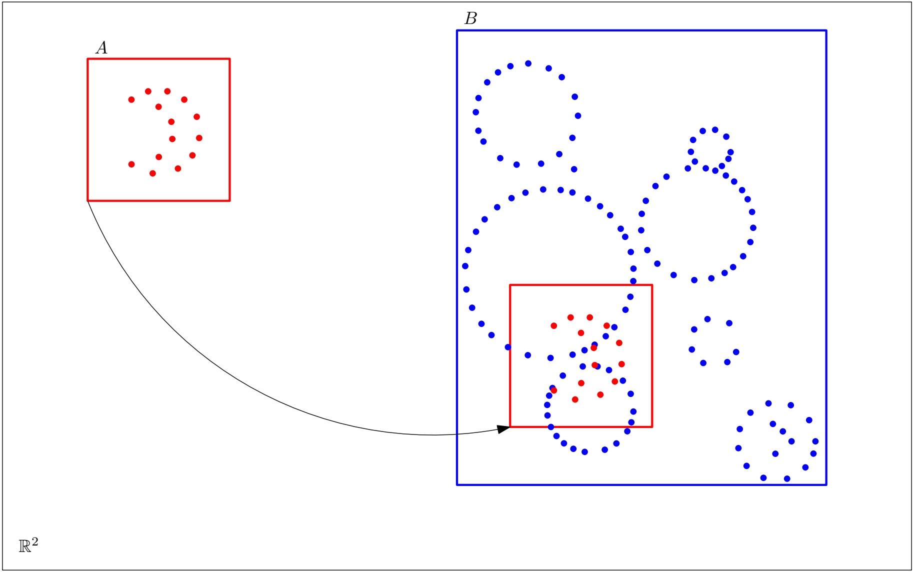
- Find the minimum cost matching over all translations

Example



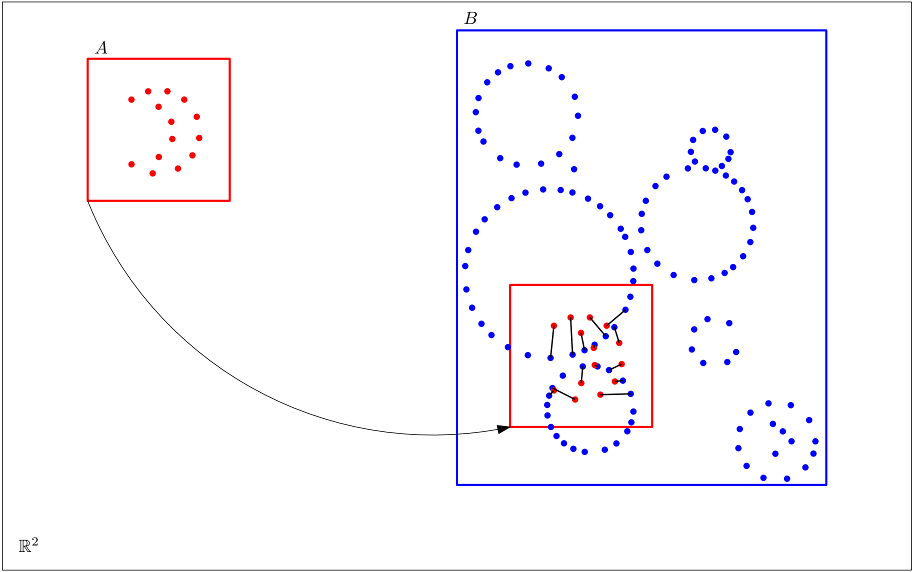
- Find the minimum cost matching over all translations

Example



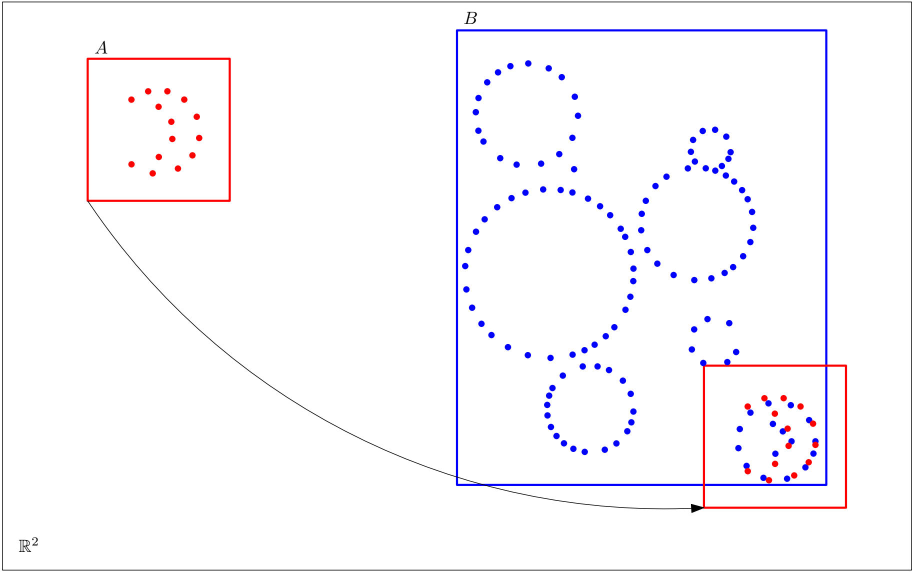
▶ Find the minimum-cost matching over all translations

Example



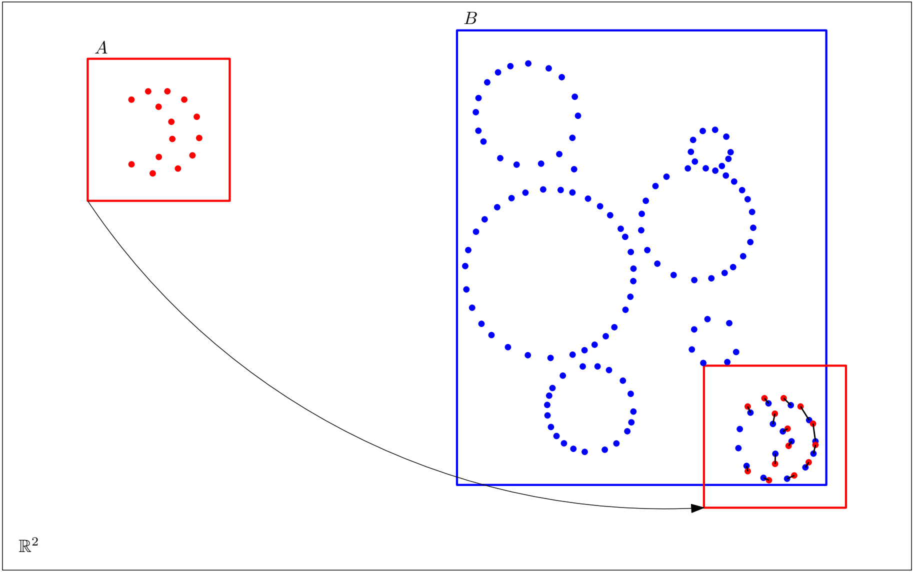
▶ Find the minimum cost matching over all translations

Example



▶ Find the minimum-cost matching over all translations

Example



- Find the minimum cost matching over all translations.

Cost function

- Given point sets A, B, with $|A| = m$ and $|B| = n$, match A into an m-subset of B after translation t.
- For matching M and translation t, define the root-mean-sqaure (RMS) cost of the matching:

$$\operatorname{cost}(M, t) := \left[\frac{1}{m} \sum_{(a,b) \in M} \|a + t - b\|^p \right]^{1/2}$$

- Can generalize to p-th power/p-th root (versus 2), and matching of size k (versus m).
- Best matching cost at each translation:

$$\operatorname{cost}^*(t) := \min_{\text{matching } M} \operatorname{cost}(M, t)$$

Cost function

- Given point sets A, B, with $|A| = m$ and $|B| = n$, match A into an m-subset of B after translation t.
- For matching M and translation t, define the root-mean-sqaure (RMS) cost of the matching:

$$\operatorname{cost}(M, t) := \left[\frac{1}{m} \sum_{(a,b) \in M} \|a + t - b\|^p \right]^{1/2}$$

- Can generalize to p-th power/p-th root (versus 2), and matching of size k (versus m).
- Best matching cost at each translation:

$$\operatorname{cost}^*(t) := \min_{\text{matching } M} \operatorname{cost}(M, t)$$

Cost function

- Given point sets A, B, with $|A| = m$ and $|B| = n$, match A into an m-subset of B after translation t.
- For matching M and translation t, define the root-mean-sqaure (RMS) cost of the matching:

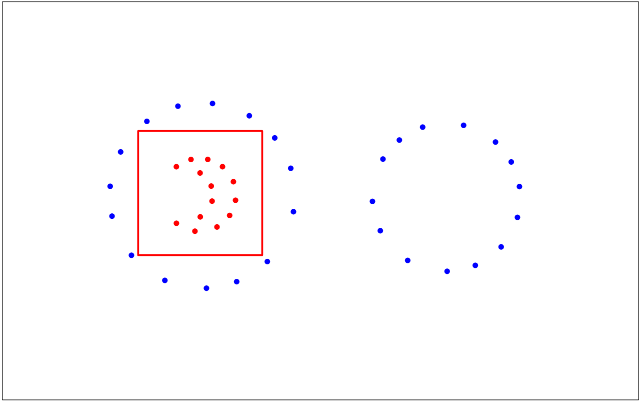
$$\operatorname{cost}(M, t) := \left[\frac{1}{m} \sum_{(a,b) \in M} \|a + t - b\|^p \right]^{1/2}$$

- Can generalize to p-th power/p-th root (versus 2), and matching of size k (versus m).
- Best matching cost at each translation:

$$\operatorname{cost}^*(t) := \min_{\text{matching } M} \operatorname{cost}(M, t)$$

Cost function under perturbations

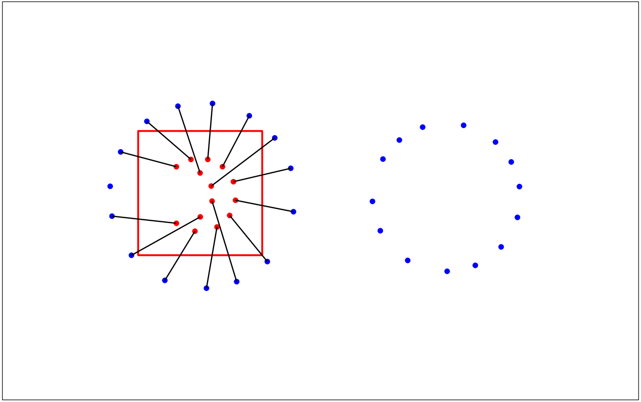
$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



- How many distinct matchings appear in $\mathrm{cost}^*(t)$?

Cost function under perturbations

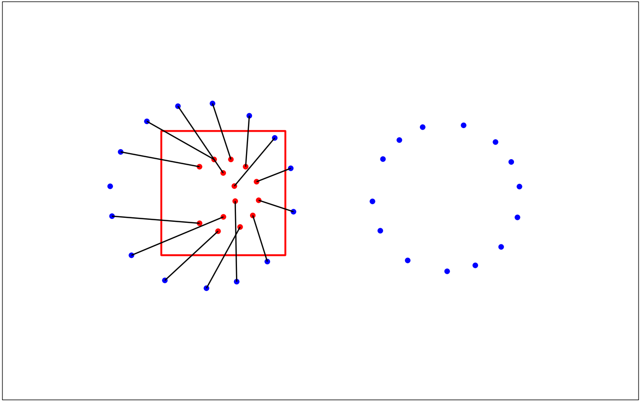
$$\text{cost}^*(t) := \min_{\text{matching } M} \text{cost}(M, t)$$



- How many distinct matchings appear in $\text{cost}^*(t)$?

Cost function under perturbations

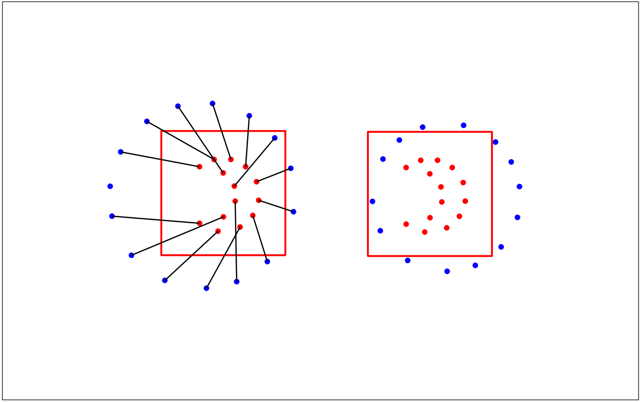
$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



- How many distinct matchings appear in $\mathrm{cost}^*(t)$?

Cost function under perturbations

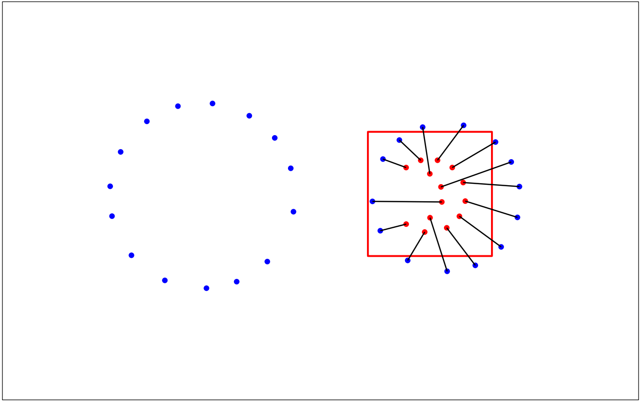
$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



- How many distinct matchings appear in $\mathrm{cost}^*(t)$?

Cost function under perturbations

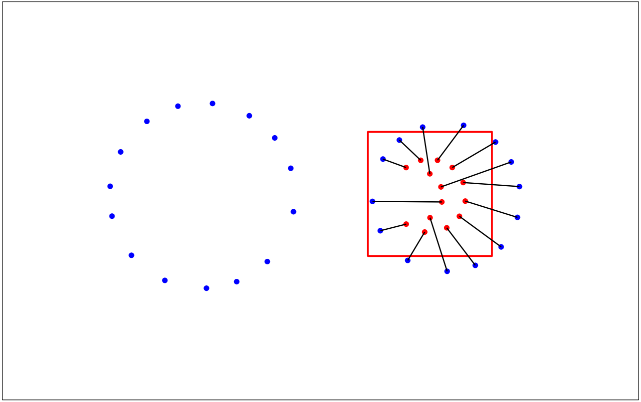
$$\text{cost}^*(t) := \min_{\text{matching } M} \text{cost}(M, t)$$



- How many distinct matchings appear in $\text{cost}^*(t)$?

Cost function under perturbations

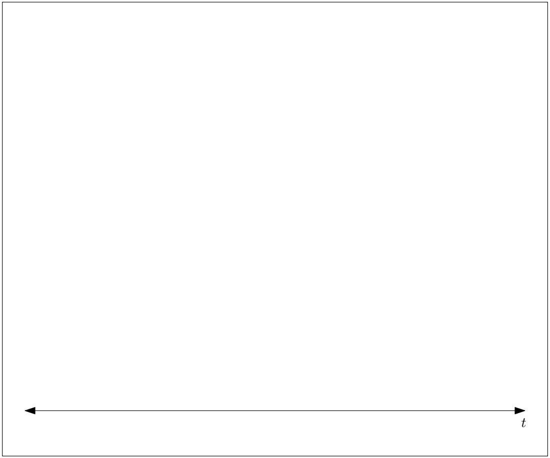
$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



- How many distinct matchings appear in $\mathrm{cost}^*(t)$?

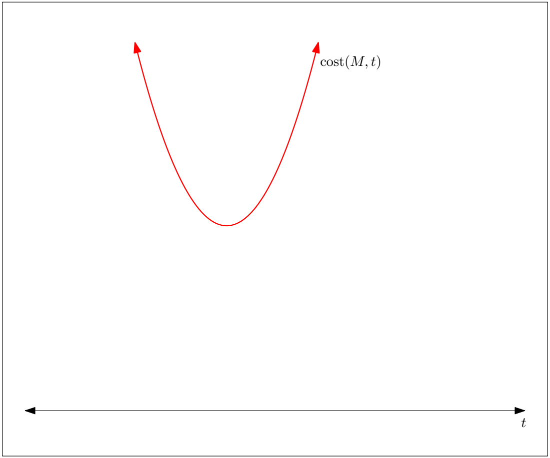
How many distinct matchings? (1-dimensional)

$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



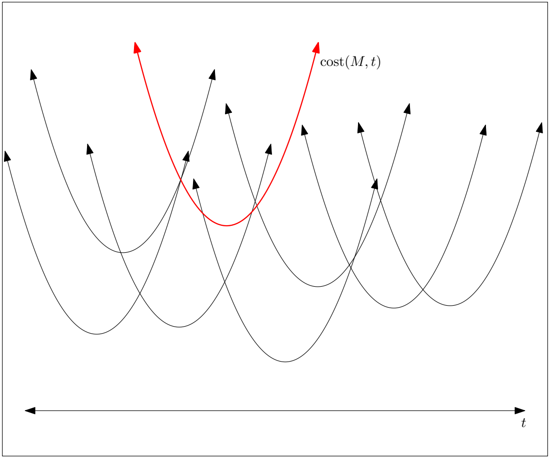
How many distinct matchings? (1-dimensional)

$$\text{cost}^*(t) := \min_{\text{matching } M} \text{cost}(M, t)$$



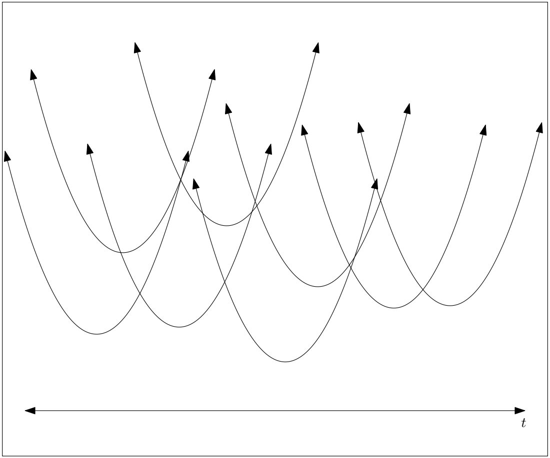
How many distinct matchings? (1-dimensional)

$$\text{cost}^*(t) := \min_{\text{matching } M} \text{cost}(M, t)$$



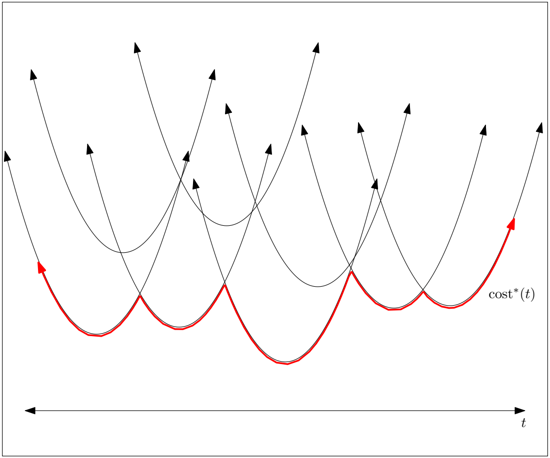
How many distinct matchings? (1-dimensional)

$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



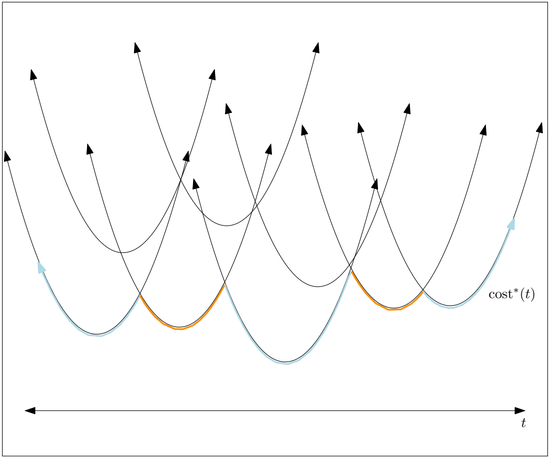
How many distinct matchings? (1-dimensional)

$$\text{cost}^*(t) \coloneqq \min_{\text{matching } M} \text{cost}(M, t)$$



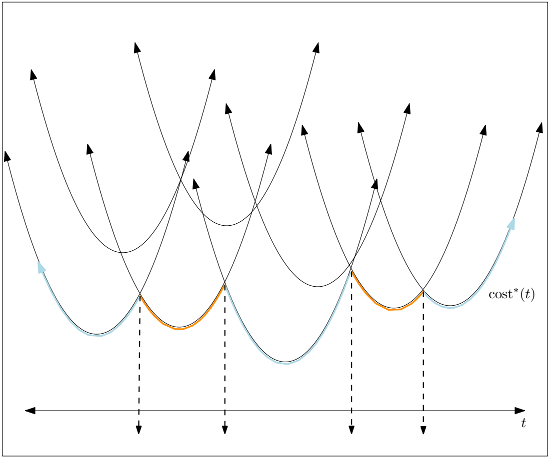
How many distinct matchings? (1-dimensional)

$$\text{cost}^*(t) := \min_{\text{matching } M} \text{cost}(M, t)$$



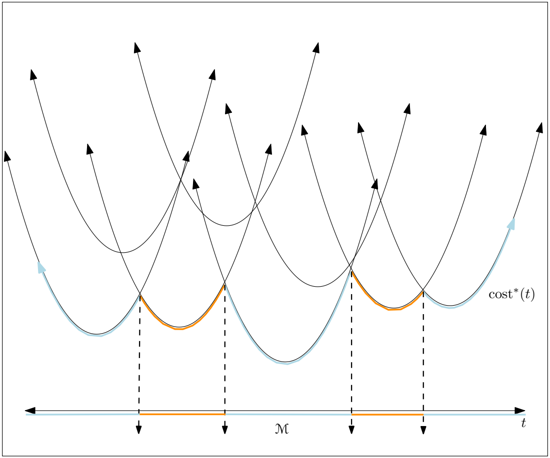
How many distinct matchings? (1-dimensional)

$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$

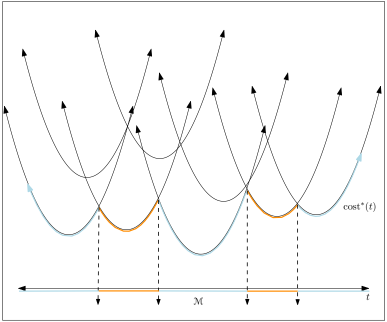


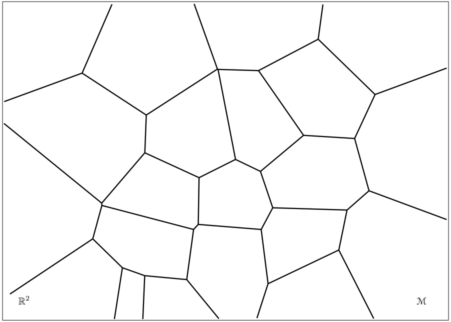
How many distinct matchings? (1-dimensional)

$$\mathrm{cost}^*(t) := \min_{\text{matching } M} \mathrm{cost}(M, t)$$



How many distinct matchings? (2-dimensional)





- ~~How many distinct matchings appear in $\mathrm{cost}^*(t)$?~~
- What is the combinatorial complexity of $\mathcal{M}$?

Questions and prior results

1. How quickly can we compute t^*, a global minimum of $\mathrm{cost}^*(t)$?
2. What is the combinatorial complexity of $\mathcal{M}$?
   - [Rote 10]: in 1D, at most $m(n-m)+1$.
   - [Ben-Avraham *et al.* 14]: $O(n^2 m^{3.5}(e \ln m + e)^m)$

- Open to find t^* in polytime, and whether $\mathcal{M}$ has polynomial complexity.
- Approximation?

Questions and prior results

1. How quickly can we compute t^*, a global minimum of $\mathrm{cost}^*(t)$?
2. What is the combinatorial complexity of $\mathcal{M}$?
   - [Rote 10]: in 1D, at most $m(n-m)+1$.
   - [Ben-Avraham *et al.* 14]: $O(n^2 m^{3.5}(e \ln m + e)^m)$

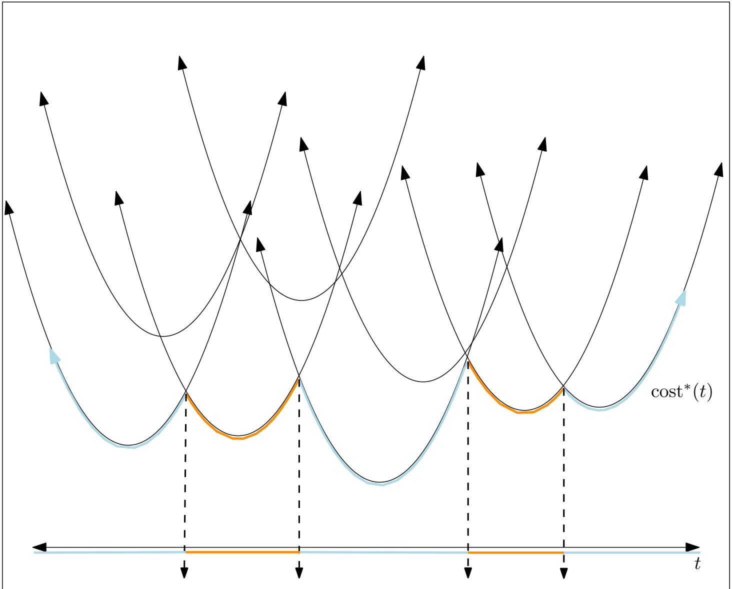
- Open to find t^* in polytime, and whether $\mathcal{M}$ has polynomial complexity.
- Approximation?

Questions and prior results

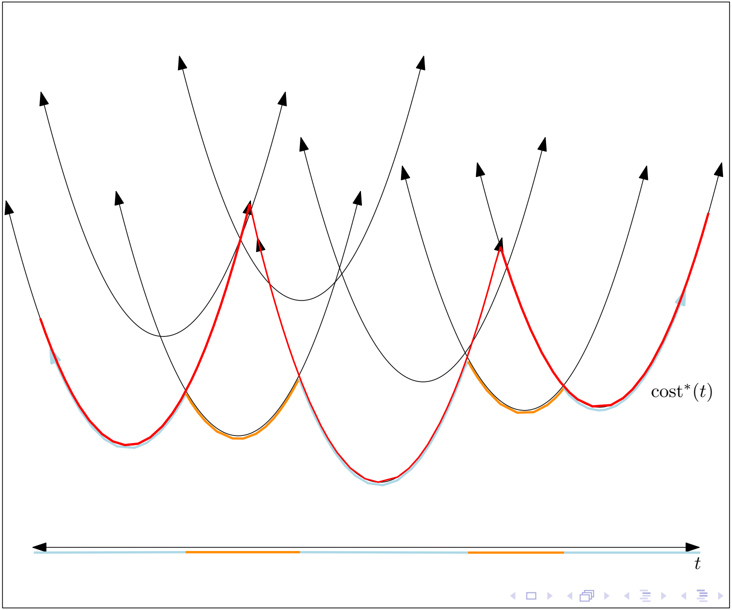
1. How quickly can we compute t^*, a global minimum of $\mathrm{cost}^*(t)$?
2. What is the combinatorial complexity of $\mathcal{M}$?
    - [Rote 10]: in 1D, at most $m(n-m)+1$.
    - [Ben-Avraham *et al.* 14]: $O(n^2 m^{3.5}(e \ln m + e)^m)$

- Open to find t^* in polytime, and whether $\mathcal{M}$ has polynomial complexity.
- Approximation?

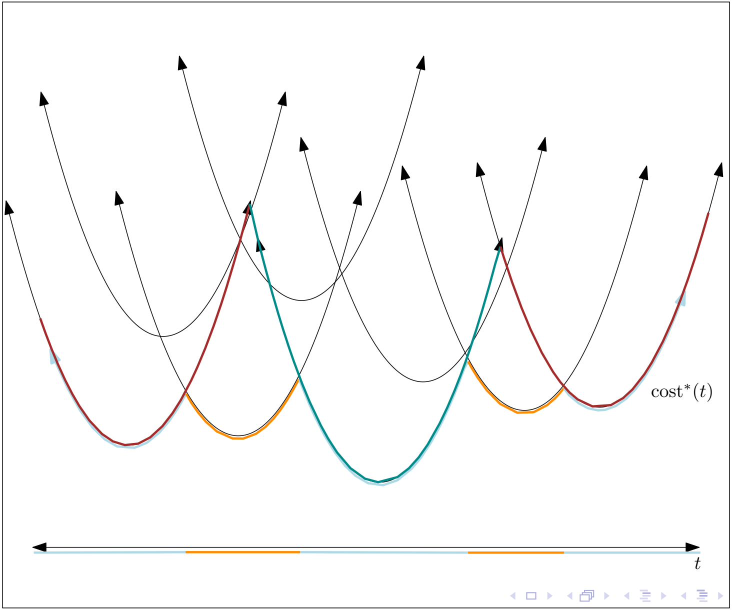
Approximating $\mathcal{M}$ (1-dimensional)



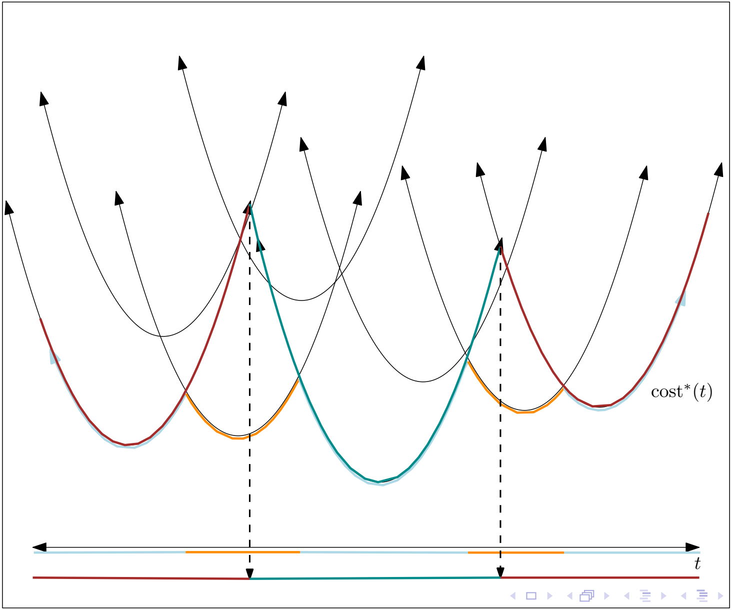
Approximating $\mathcal{M}$ (1-dimensional)



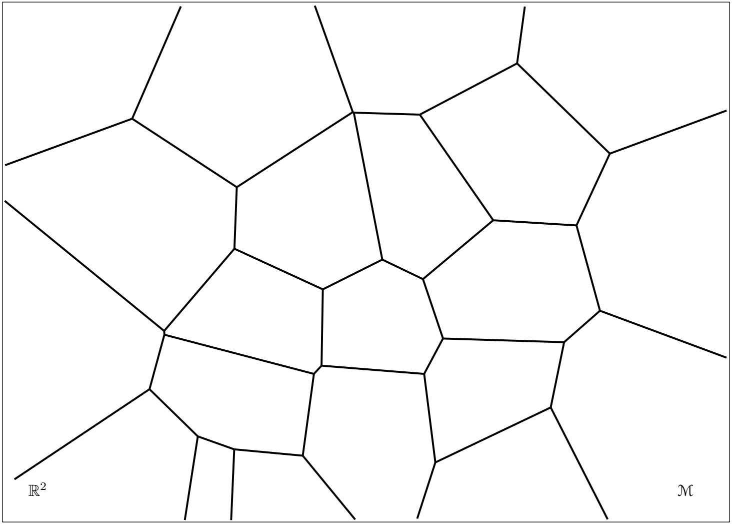
Approximating $\mathcal{M}$ (1-dimensional)



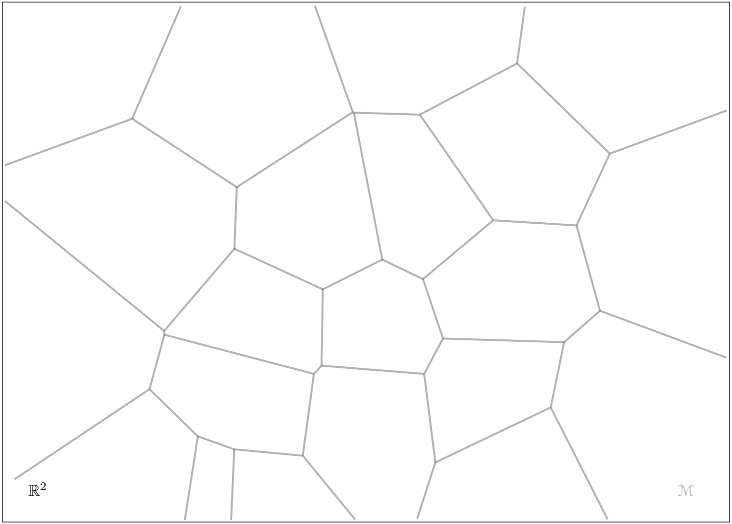
Approximating $\mathcal{M}$ (1-dimensional)



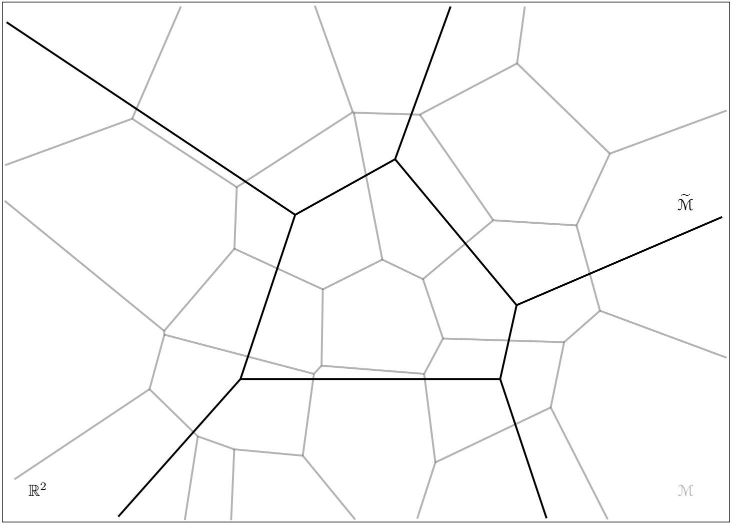
Approximating $\mathcal{M}$ (2-dimensional)



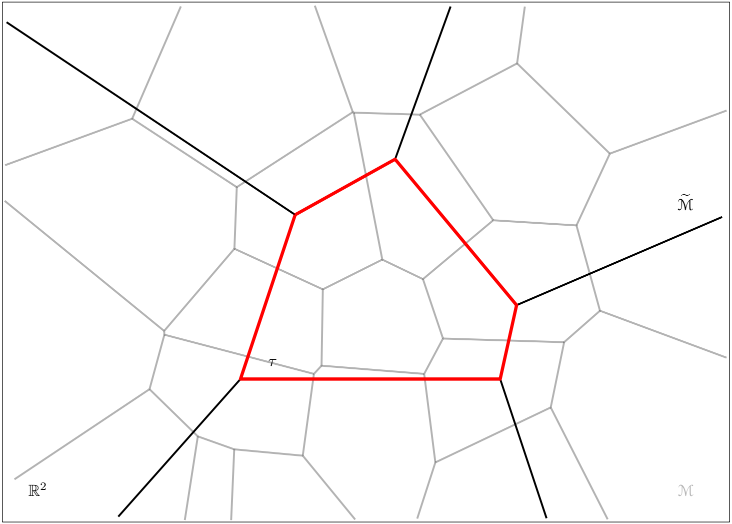
Approximating $\mathcal{M}$ (2-dimensional)



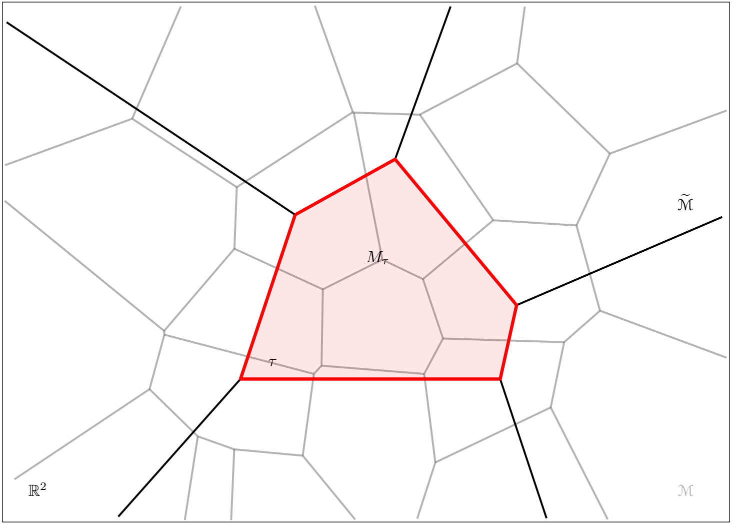
Approximating $\mathcal{M}$ (2-dimensional)



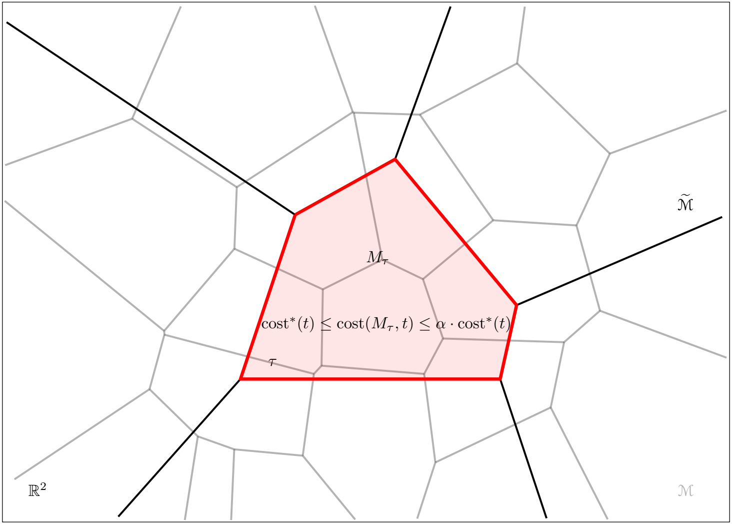
Approximating $\mathcal{M}$ (2-dimensional)



Approximating $\mathcal{M}$ (2-dimensional)



Approximating $\mathcal{M}$ (2-dimensional)



Our results (approximation helps)

1. How quickly can we compute t^*, a global minimum of $\text{cost}^*(t)$?

Theorem

In $\text{poly}(m, n, \varepsilon^{-1})$ *time, can compute a* $(1+\varepsilon)$ *approximation to* $\text{cost}^*(t^*)$ *by exploring the faces of* $\widetilde{\mathcal{M}}$.

2. What is the combinatorial complexity of $\mathcal{M}$?

Theorem

In $\text{poly}(m, n, \varepsilon^{-1})$ *time, can construct a* $(1+\varepsilon)$ *approximate diagram* $\widetilde{\mathcal{M}}$ *of complexity* $O(n\varepsilon^{-2}\log\varepsilon^{-1})$.

Overview

1. The set of point-to-point translations give a constant approximate diagram of size mn.
2. Using exponential grids, constant $\rightarrow (1+\varepsilon)$ approximation of size $O(mn\varepsilon^{-2}\log\varepsilon^{-1})$.
3. Reduce size to $O(n\varepsilon^{-2}\log\varepsilon^{-1})$ by clustering.

Overview

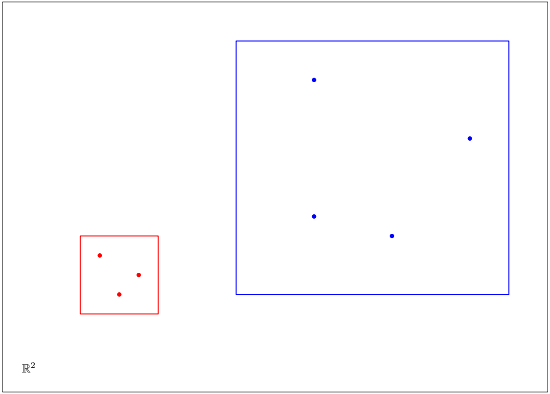
1. The set of point-to-point translations give a constant approximate diagram of size mn.
2. Using exponential grids, constant $\rightarrow$ $(1+\varepsilon)$ approximation of size $O(mn\varepsilon^{-2}\log\varepsilon^{-1})$.
3. Reduce size to $O(n\varepsilon^{-2}\log\varepsilon^{-1})$ by clustering.

Overview

1. The set of point-to-point translations give a constant approximate diagram of size mn.
2. Using exponential grids, constant $\to$ $(1+\varepsilon)$ approximation of size $O(mn\varepsilon^{-2}\log\varepsilon^{-1})$.
3. Reduce size to $O(n\varepsilon^{-2}\log\varepsilon^{-1})$ by clustering.

Point-to-point translations

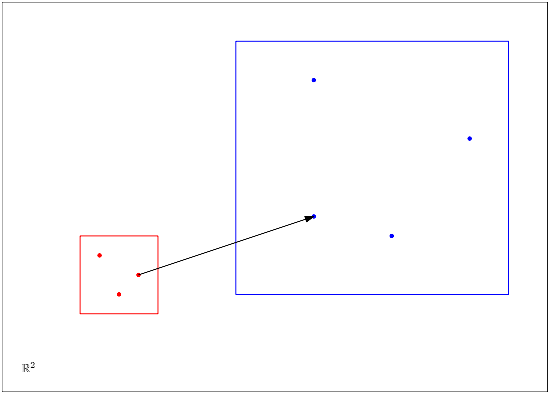
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

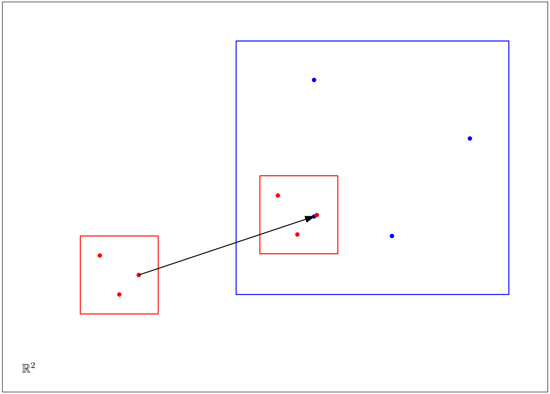
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

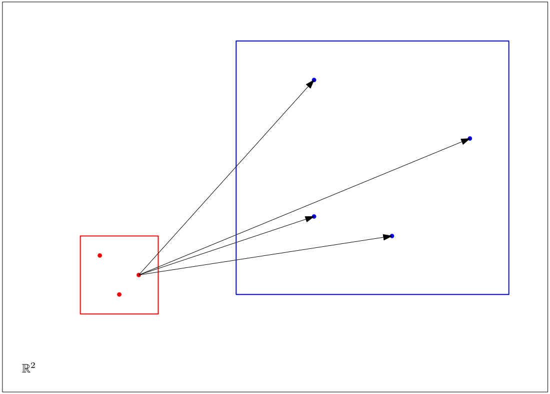
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

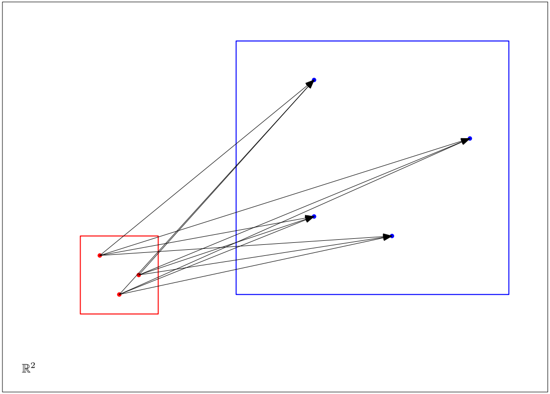
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

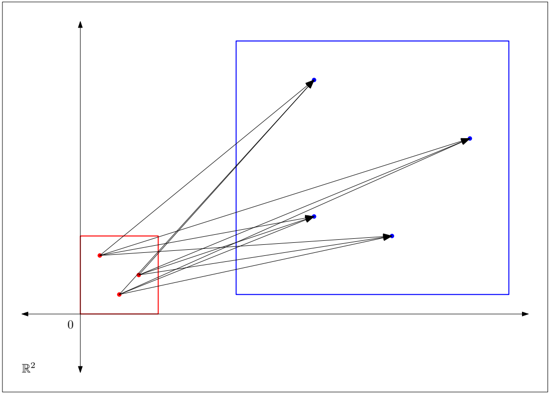
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

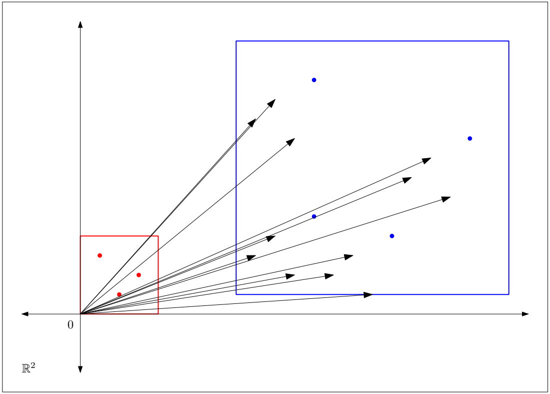
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

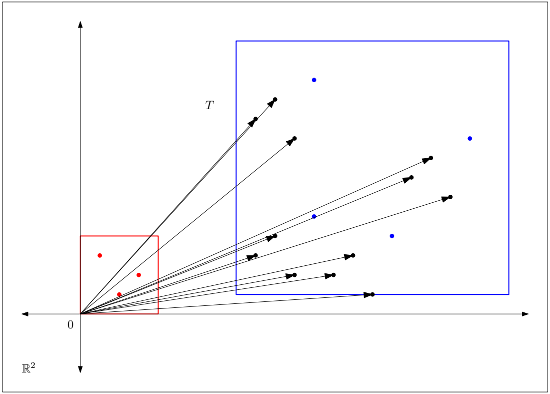
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

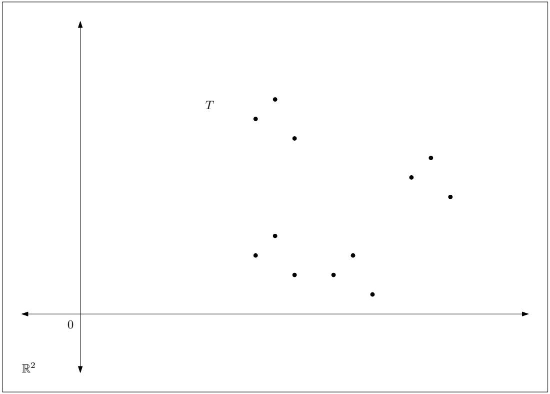
- Point-to-point translations [Cabello *et al.* 08]:
$T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

Point-to-point translations

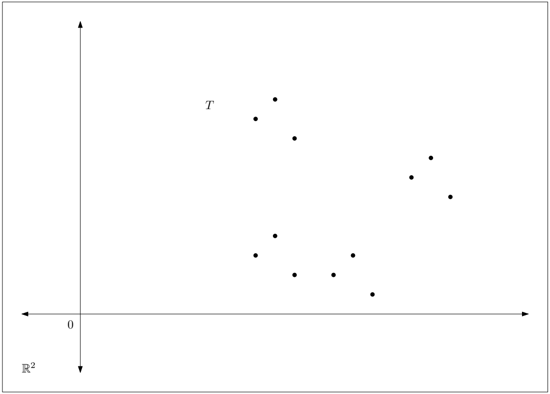
- Point-to-point translations [Cabello *et al.* 08]:
  $T := \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



- Enough to look at the optimal matching of each $t_{ba} \in T$.

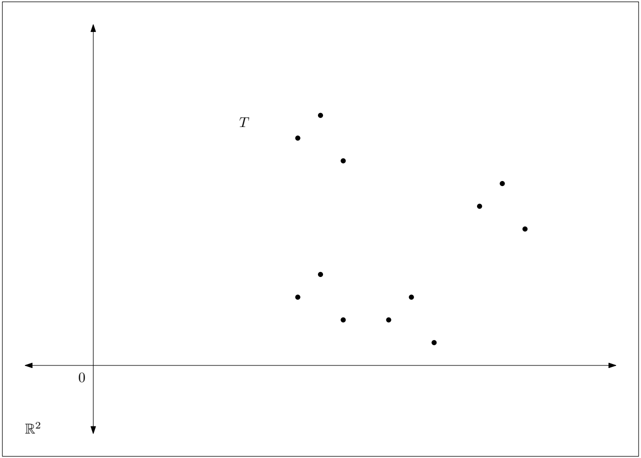
Point-to-point translations

- Point-to-point translations [Cabello *et al.* 08]:
  $T \coloneqq \{t_{ba} = (b - a) \mid a \in A, b \in B\}$



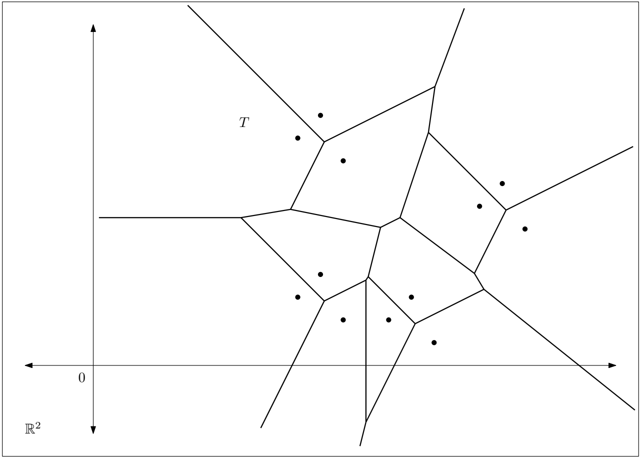
- Enough to look at the optimal matching of each $t_{ba} \in T$.

$O(1)$-approximate diagram from point-to-point translations



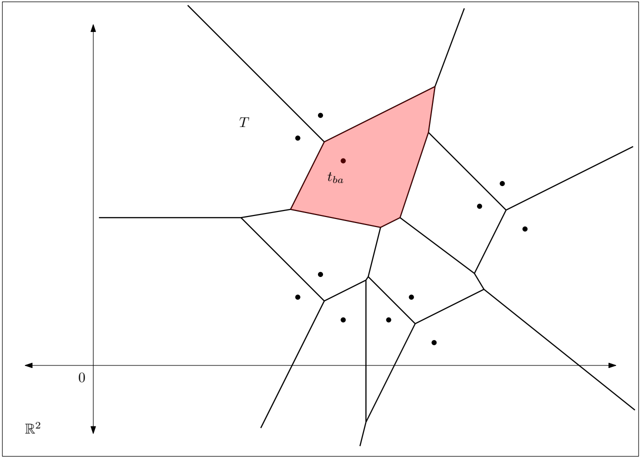
- $\widetilde{\mathcal{M}}$ is a constant-approximate diagram of size $|T| = mn$.

$O(1)$-approximate diagram from point-to-point translations



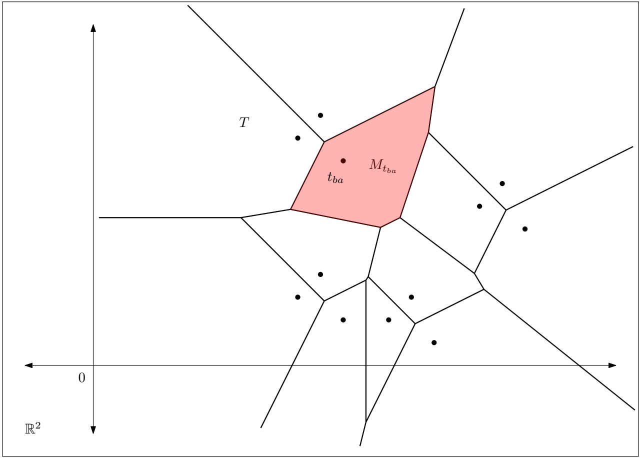
- $\widetilde{\mathcal{M}}$ is a constant-approximate diagram of size $|T| = mn$.

$O(1)$-approximate diagram from point-to-point translations



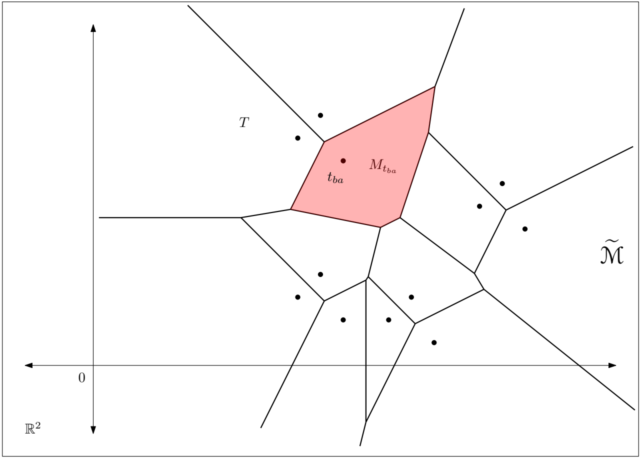
- $\widetilde{\mathcal{M}}$ is a constant-approximate diagram of size $|T| = mn$.

$O(1)$-approximate diagram from point-to-point translations



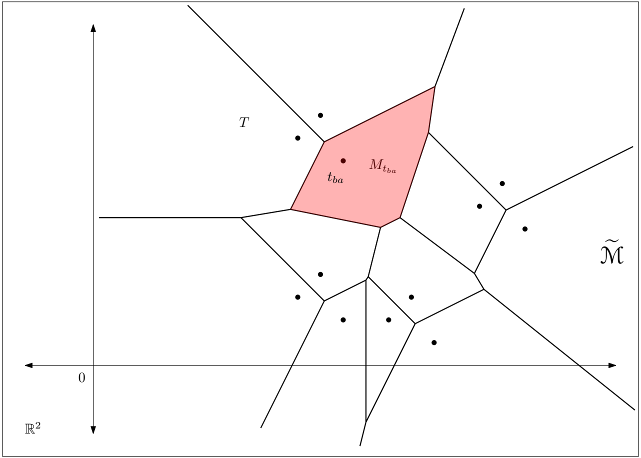
- $\widetilde{\mathcal{M}}$ is a constant-approximate diagram of size $|T| = mn$.

$O(1)$-approximate diagram from point-to-point translations



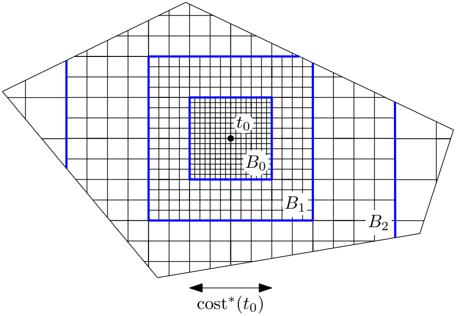
- $\widetilde{\mathcal{M}}$ is a constant-approximate diagram of size $|T| = mn$.

$O(1)$-approximate diagram from point-to-point translations



- $\widetilde{\mathcal{M}}$ is a constant-approximate diagram of size $|T| = mn$.

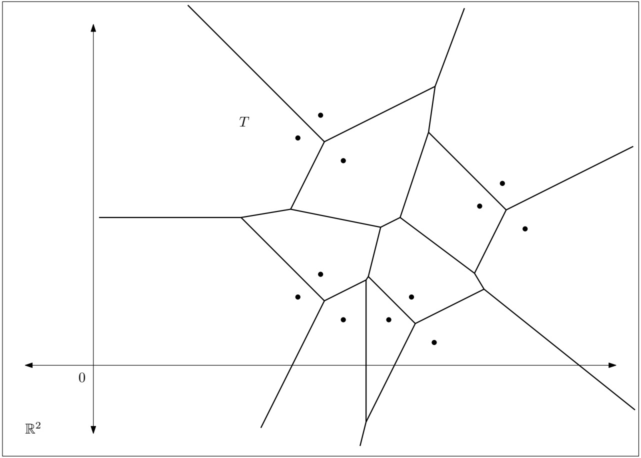
$O(1) \to (1+\varepsilon)$ approximation



- $(1+\varepsilon)$ approximate diagram of size $O(|T|\varepsilon^{-2}\log\varepsilon^{-1}) = O((mn)\varepsilon^{-2}\log\varepsilon^{-1})$

Near-linear size: clustering T

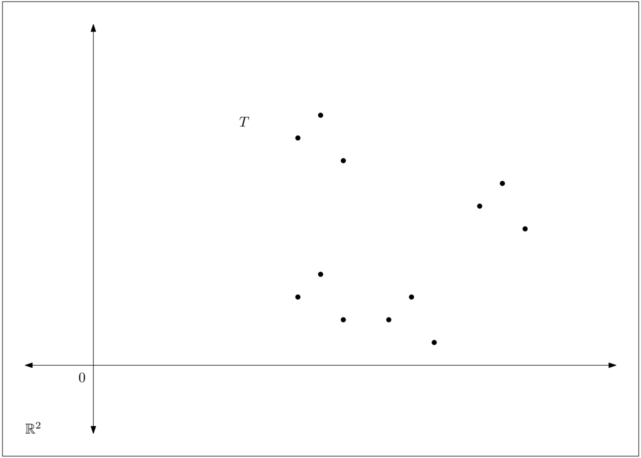
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

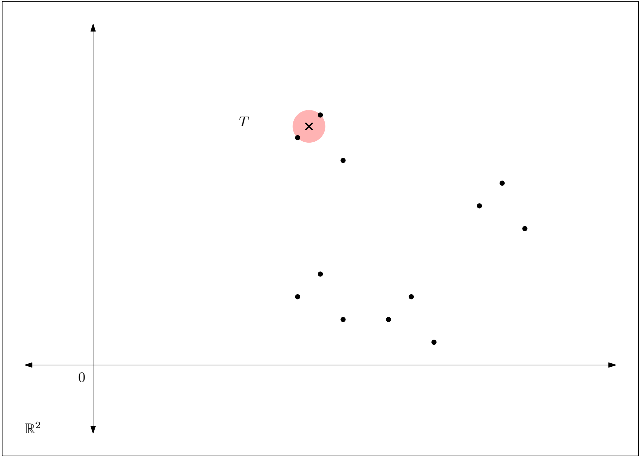
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

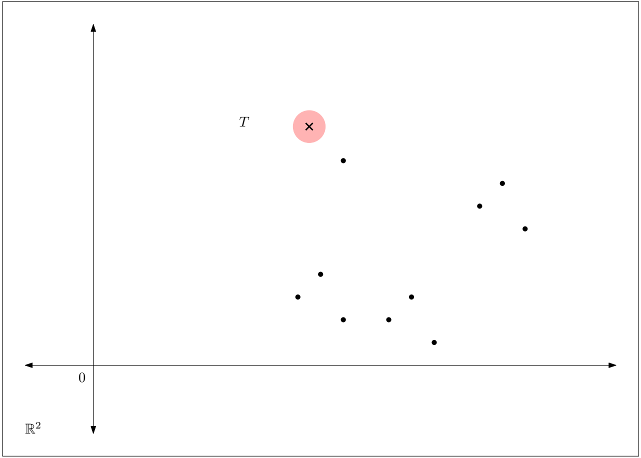
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

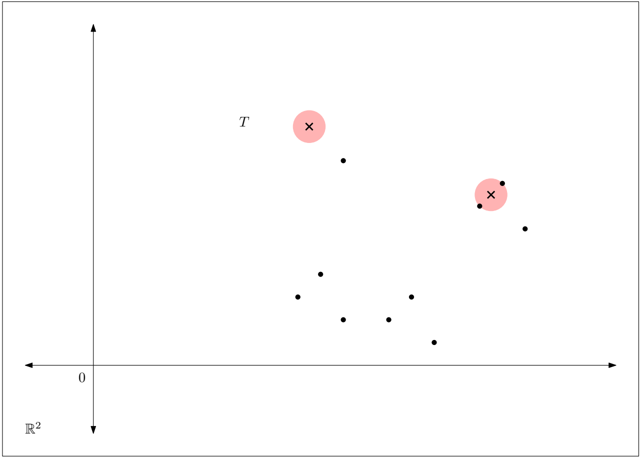
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

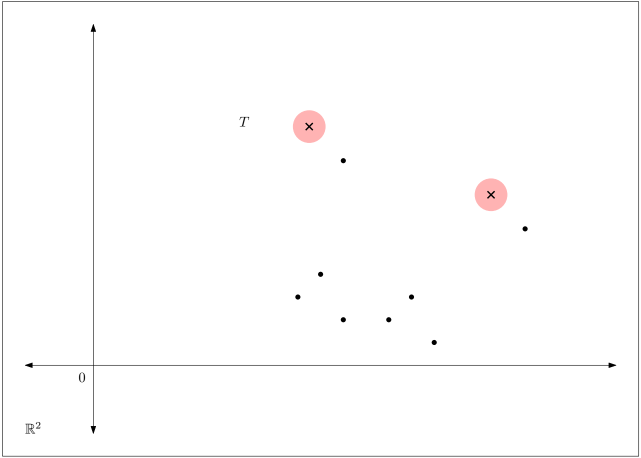
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

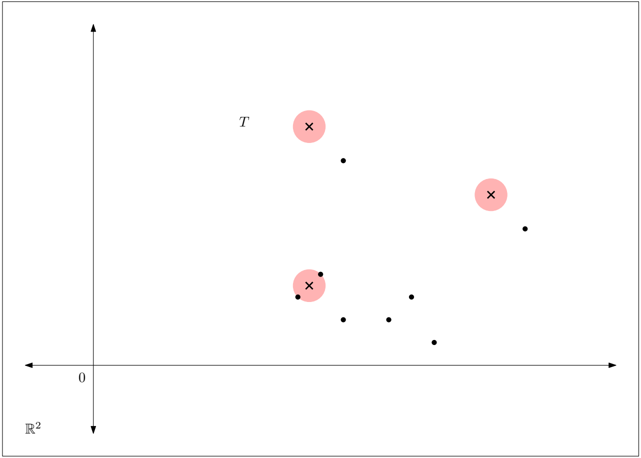
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

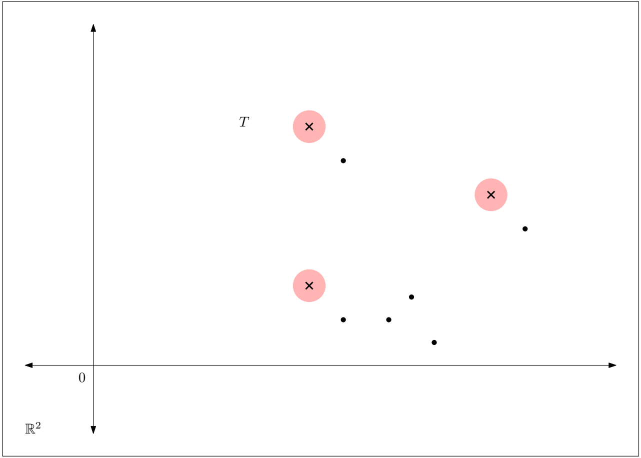
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

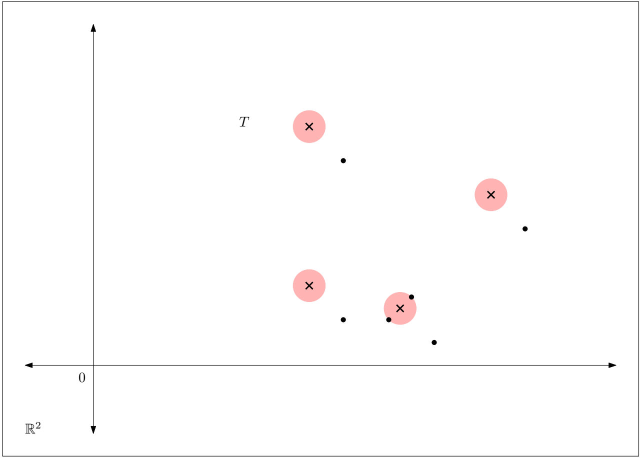
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

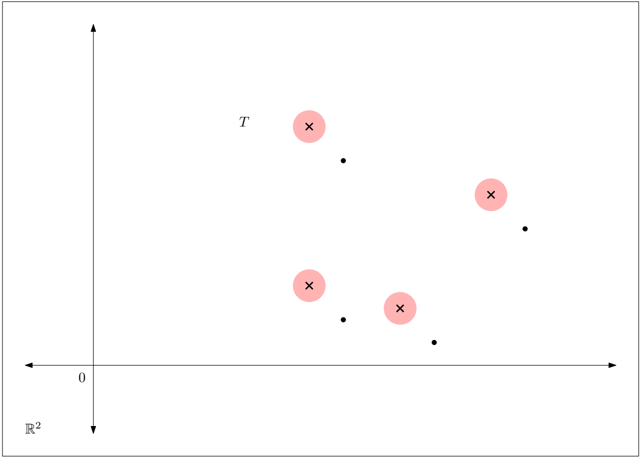
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

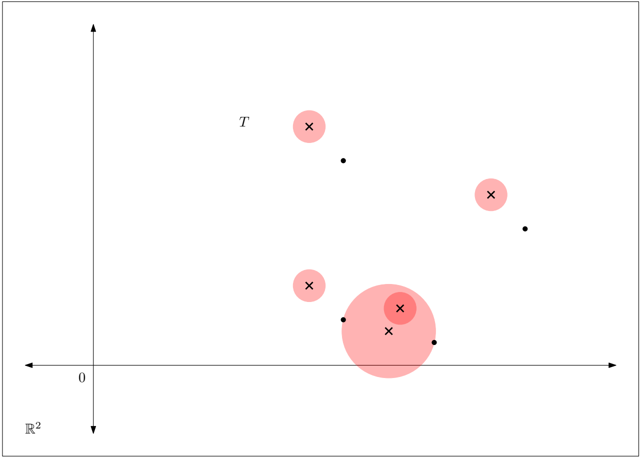
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

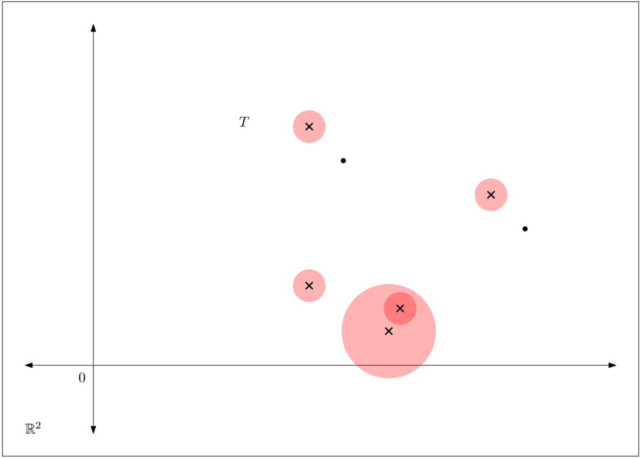
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

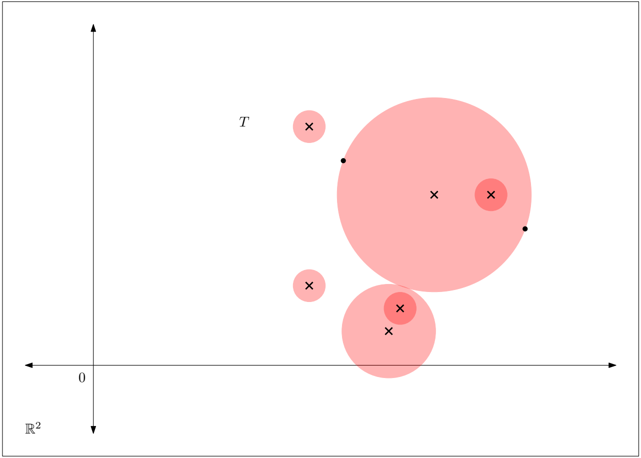
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

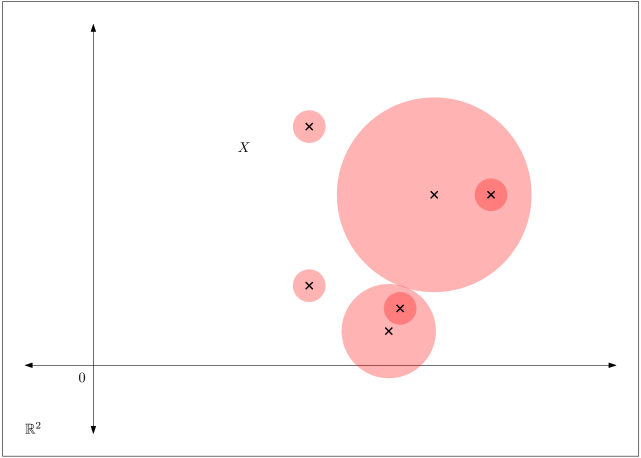
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

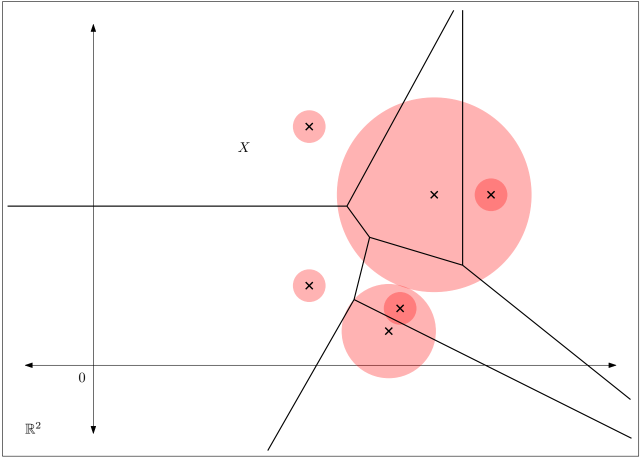
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

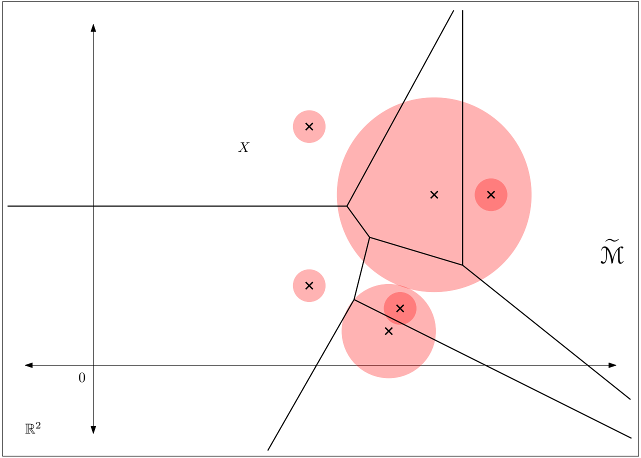
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

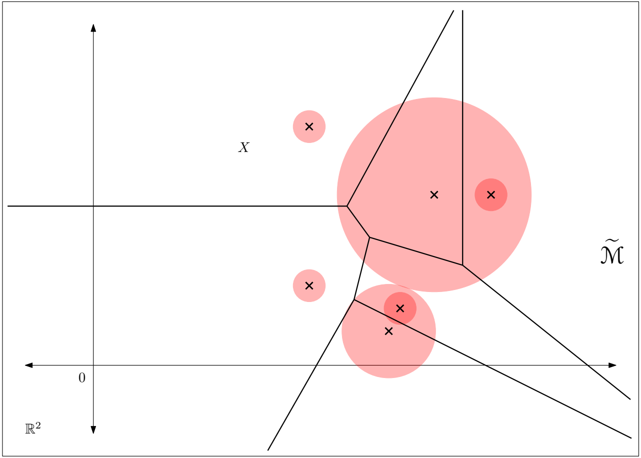
- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Near-linear size: clustering T

- Any $O(1)$ approx. diagram is enough for the $(1+\varepsilon)$ diagram.
- Partition T into clusters of size at least $m/2$.



- Reduces diagram size from $mn \to O(n)$.

Open questions

1. Is the complexity of $\mathcal{M}$ polynomial?
2. Approximate diagrams for rotations? Rigid transforms?

The End

Thank you.