# Partial optimality in Cubic Correlation Clustering

Silvia Di Gregorio
joint work with Bjoern Andres and David Stein

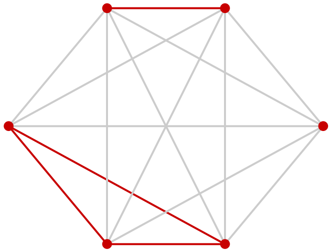
Faculty of Computer Science,
TU Dresden

May 22, 2023

# Cubic Correlation Clustering

Let $n \geq 3$, $c \in \mathbb{R}^{\binom{n}{3}+\binom{n}{2}}$, $S$ be the set containing all binary vectors inducing a clustering.

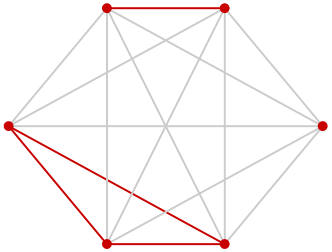
$$\begin{aligned} \min \quad & \sum_{pqr \in \binom{n}{3}} c_{pqr} x_{pq} x_{pr} x_{qr} + \sum_{pq \in \binom{n}{2}} c_{pq} x_{pq} \\ \text{s.t.} \quad & x \in S. \end{aligned}$$



# Cubic Correlation Clustering

Let $n \geq 3$, $c \in \mathbb{R}^{\binom{n}{3}+\binom{n}{2}}$, $S$ be the set containing all binary vectors inducing a clustering.

$$\begin{aligned} \min \quad & \sum_{pqr \in \binom{n}{3}} c_{pqr} x_{pq} x_{pr} x_{qr} + \sum_{pq \in \binom{n}{2}} c_{pq} x_{pq} \\ \text{s.t.} \quad & x \in S. \end{aligned}$$

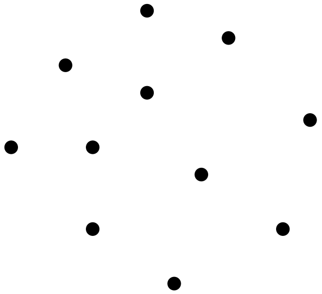


- Example of nonlinear combinatorial optimization problem
- NP-hard to solve

Goal: computing a partial solution to the problem efficiently

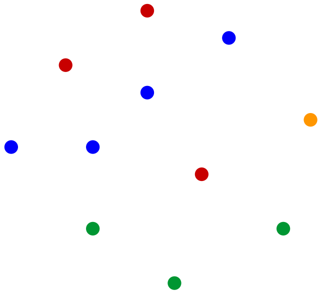
# Motivation: Correlation Clustering

- Goal: given $n$ points somehow related, cluster them
- No prior knowledge of optimal number of clusters (Bansal et al. '04)



# Motivation: Correlation Clustering

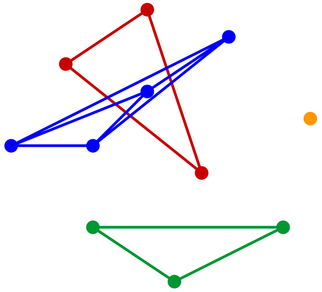
- Goal: given $n$ points somehow related, cluster them
- No prior knowledge of optimal number of clusters (Bansal et al. '04)



# Motivation: Correlation Clustering

- Goal: given $n$ points somehow related, cluster them
- No prior knowledge of optimal number of clusters (Bansal et al. '04)
- For any two points $p$, $q$, we introduce binary variable $x_{pq}$:

  $x_{pq} = 1 \iff p, q$ in same cluster

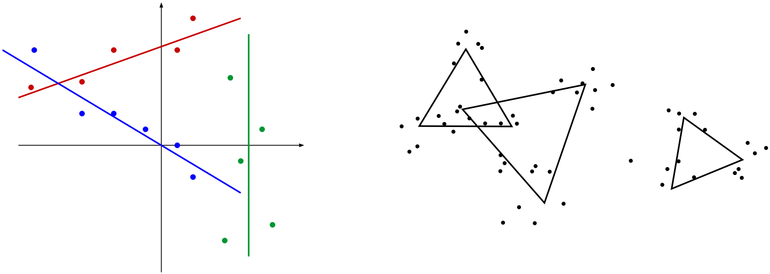


# Motivation: Cubic objective

Want to compare three points at the same time.

Applications (Levinkov et al. '22):

- subspace clustering (affine lines in 2D or linear planes in 3D)
- scale-invariant recognition of symbols and rigid objects under scaling, rotation, translations



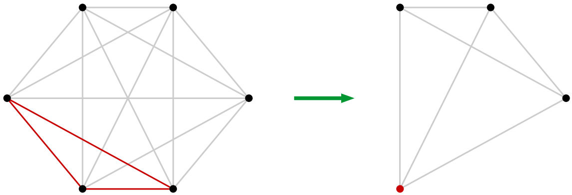
# Motivation: Partial optimality

- Helpful in reducing size of the instance: then either exact algorithm or heuristic
- Recent local search heuristics for several applications of higher-order correlation clustering (Levinkov et al. '17, '22)
- Successful approach for linear objective functions (Alush, Goldberger '12; Lange et al. '18, '19)

# Motivation: Partial optimality

- Helpful in reducing size of the instance: then either exact algorithm or heuristic
- Recent local search heuristics for several applications of higher-order correlation clustering (Levinkov et al. '17, '22)
- Successful approach for linear objective functions (Alush, Goldberger '12; Lange et al. '18, '19)

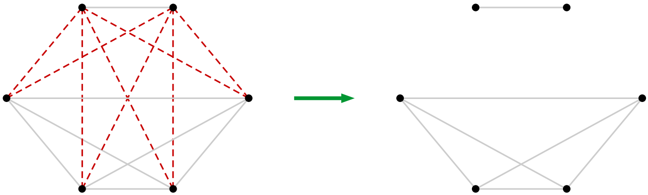
- Fixing variables to 1 leads to a smaller instance (join condition)



# Motivation: Partial optimality

- Helpful in reducing size of the instance: then either exact algorithm or heuristic
- Recent local search heuristics for several applications of higher-order correlation clustering (Levinkov et al. '17, '22)
- Successful approach for linear objective functions (Alush, Goldberger '12; Lange et al. '18, '19)

- Fixing variables to 0 leads potentially to smaller instances (cut condition)



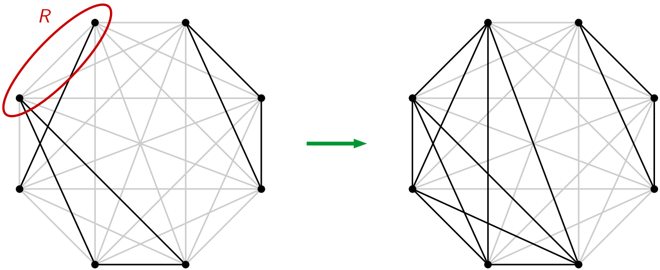
# Overview results

- In contrast to some usual approaches: we do not introduce additional variables and we do not employ a LP (or convex) relaxation (Adams et al. '98)
- Generalize all partial optimality for linear objective function and establish new conditions
- Total of 11 criteria: 3 cut, 8 join
- We can check all of them efficiently: either via an exact algorithm or through a heuristic
- Tested on two datasets
- Obtained by combining appropriately improving maps (Shekhovtsov '13)

# Improving maps: Join

Let $x \in S$, $R \subseteq [n]$, the *elementary join map* $\sigma_R$ is defined as

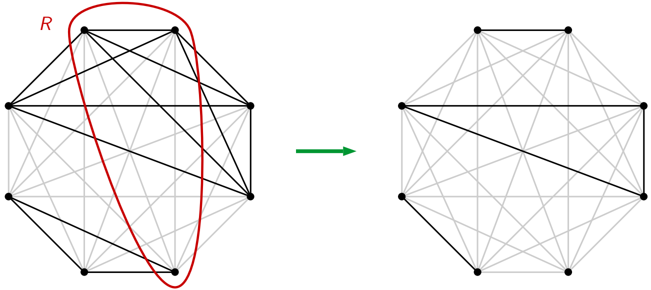
$$\sigma_R(x)_{pq} := \begin{cases} 1 & \text{if } pq \in \binom{R}{2} \\ 1 & \text{if } \forall p' \in \{p,q\} \setminus R \ \exists q' \in R\colon x_{p'q'} = 1 \\ x_{pq} & \text{otherwise} \end{cases}$$



# Improving maps: Cut

Let $x \in S$, $R \subseteq [n]$, the *elementary cut map* $\sigma_{\delta(R)}$ is defined as

$$\sigma_{\delta(R)}(x)_{pq} := \begin{cases} 0 & \text{if } pq \in \delta(R) \\ x_{pq} & \text{otherwise} \end{cases}$$



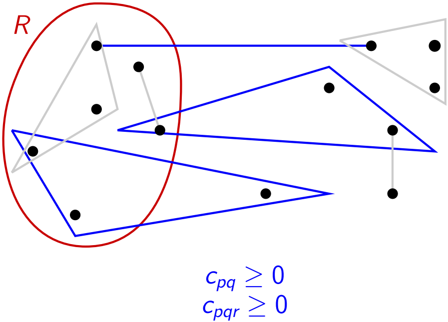
# First cut criterion

## Proposition

*If there exists* $R \subseteq [n]$ *such that*

$$c_{pq} \geq 0 \qquad \forall pq \in \delta(R)$$
$$c_{pqr} \geq 0 \qquad \forall pqr \in T_{\delta(R)}$$

*then there is an optimal solution* $x^*$ *such that* $x^*_{ij} = 0$ *for all* $ij \in \delta(R)$.




- Can be tested exactly by greedy algorithm
- Split instance in independent smaller instances

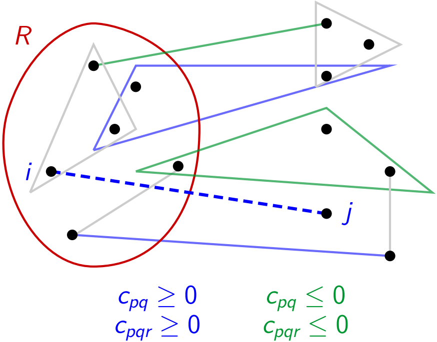
# Second cut criterion

## Proposition

*Let $ij \in \binom{n}{2}$. If there exists $R \subseteq [n]$ with $ij \in \delta(R)$ and*

$$c^+_{ij} \geq \sum_{pqr \in T_{\delta(R)}} c^-_{pqr} + \sum_{pq \in \delta(R)} c^-_{pq},$$

*then there is an optimal solution $x^*$ such that $x^*_{ij} = 0$.*




- Can be tested exactly by reducing it to a min *st*-cut problem
- Does not divide the instance in independent smaller instances

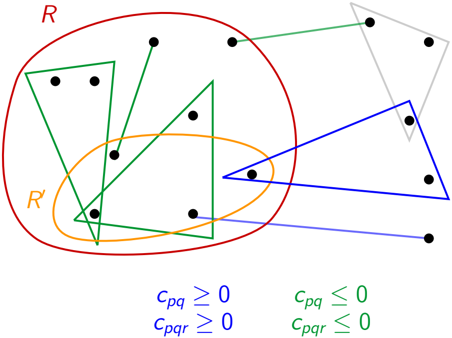
# Join criterion

## Proposition

*If there exists $R \subseteq [n]$ such that $c_{pq} \leq 0$, $c_{pqr} \leq 0$ inside of $R$, and*

$$\max_{\substack{R' \subset R \\ R' \neq \emptyset}} \left\{ \sum_{pqr \in T_{\delta(R')} \cap \binom{R}{3}} c_{pqr} + \sum_{pq \in \delta(R', R \setminus R')} c_{pq} \right\} \leq \sum_{pqr \in T_{\delta(R)} \cap T^-} c_{pqr} + \sum_{pq \in \delta(R) \cap P^-} c_{pq}$$

*then there is an optimal solution $x^*$ such that $x^*_{ij} = 1$, for all $ij \in \binom{R}{2}$.*

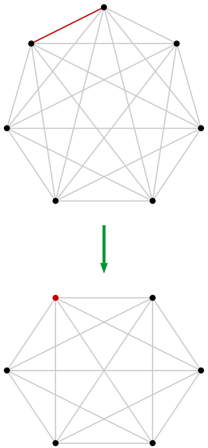



- Can be tested with a heuristic: combination of a greedy region growing and min *st*-cut problem
- Leads to one smaller instance

# Practical impact

Goal: examine effectiveness empirically by computing percentage of fixed optimal values

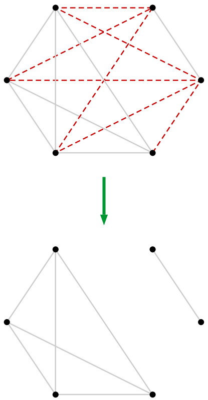
- Combine partial optimality criteria in a recursive algorithm
- Start with join criteria



# Practical impact

Goal: examine effectiveness empirically by computing percentage of fixed optimal values

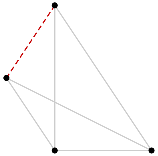
- Combine partial optimality criteria in a recursive algorithm
- Start with join criteria
- Then move to cut criteria
  - First the one that divides instance in connected components



# Practical impact

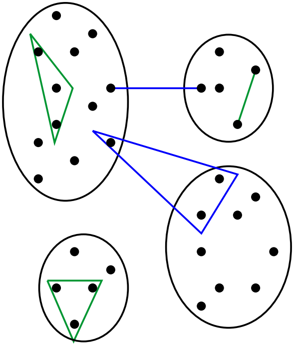
Goal: examine effectiveness empirically by computing percentage of fixed optimal values

- Combine partial optimality criteria in a recursive algorithm
- Start with join criteria
- Then move to cut criteria
    - First the one that divides instance in connected components
    - Lastly the remaining ones

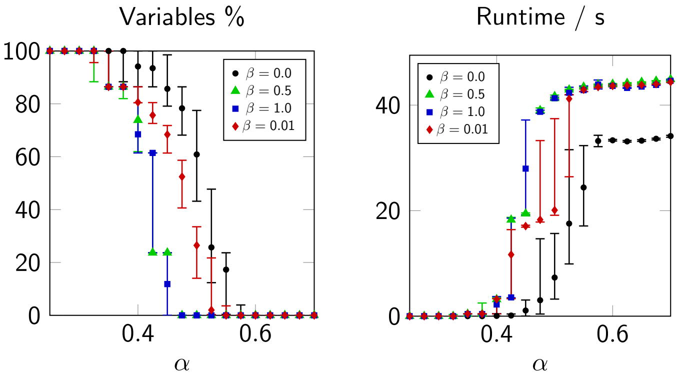


# Partition dataset: Description

- Instances defined with respect to a partition into four sets
- $\alpha \in [0, 1]$: similarity between intra- and inter-clusters' costs
- $\beta \in [0, 1]$: quantity of triples' costs relative to quantity of pairs' costs



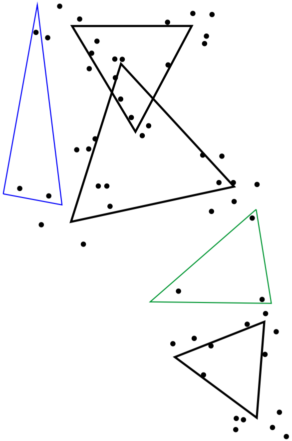
# Partition dataset: Results



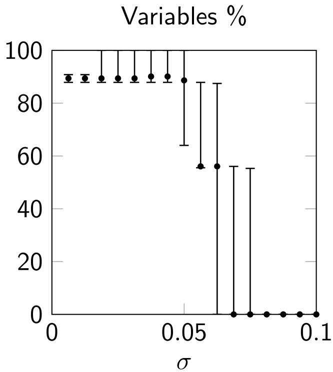

- 30 repetitions, number of points fixed to 48
- The percentage of fixed variables decreases with increasing $\alpha$, while $\beta$ has no big effect
- $\alpha$ increases, runtime increases ($<$ 1 minute)

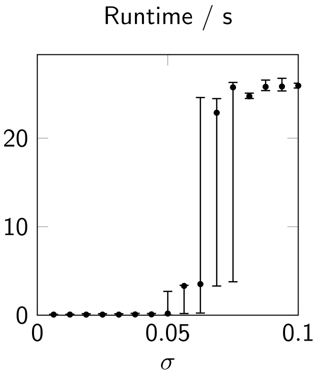
# Triangles dataset: Description

- Geometric problem of finding equilateral triangles in a noisy point cloud
- We fix three equilateral triangles in the plane
- For each vertex of a triangle, we draw points around it from a Gaussian distribution with standard deviation $\sigma$



# Triangles dataset: Results




- 30 repetitions, number of points fixed to 45
- The percentage of fixed variables decreases with increasing $\sigma$
- $\sigma$ increases, runtime increases ($< 40$ seconds)

# Conclusions

- Generalized all partial optimality criteria for linear objectives to the cubic setting, and developed new ones
- Devised exact or heuristic algorithms to test each condition
- Tested them on two datasets

# Conclusions

- Generalized all partial optimality criteria for linear objectives to the cubic setting, and developed new ones
- Devised exact or heuristic algorithms to test each condition
- Tested them on two datasets

Next steps:

- Currently working on a linearization approach and a branch-and-cut algorithm: using partial optimality conditions as a preprocessing
- Instances encoded by sparse (hyper)graphs

# Thanks for your attention!

Questions? email: silvia.di_gregorio@tu-dresden.de