Learning optimal objective values for MILP

Karen Aardal, TU Delft

Joint with Lara Scavuzzo and Neil Yorke-Smith

(Q1): Can we predict the optimal objective value before we start branching?

(Q2): Can we predict, during the solution process, whether or not a solution is optimal?

The output from Q1 is used as input for the classifier that is used to answer Q2.

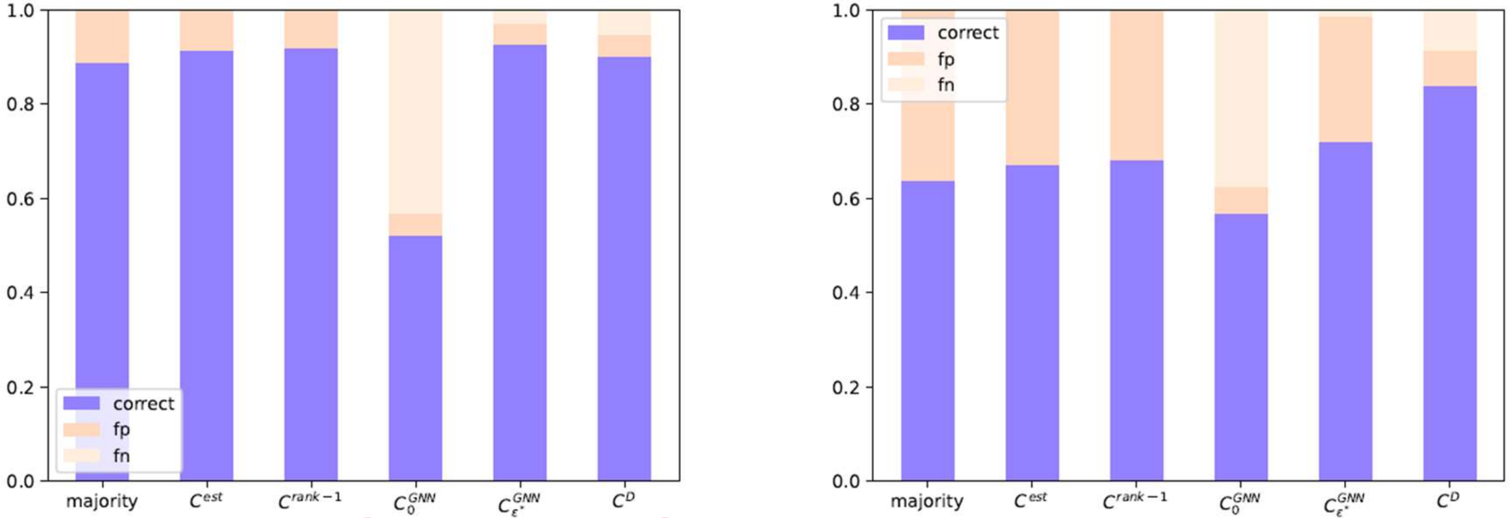
Previous work on predictions has focused on predicting the optimal solution: Ding et al. (2020), Nair et al. (2020), Khalil et al. (2022).

Our work is related to Berthold et al. (2018), and yields less “overly optimistic” predictions.

blue: fraction of correctly classified samples

dark yellow: fraction of false positives

light yellow: fraction of false negatives

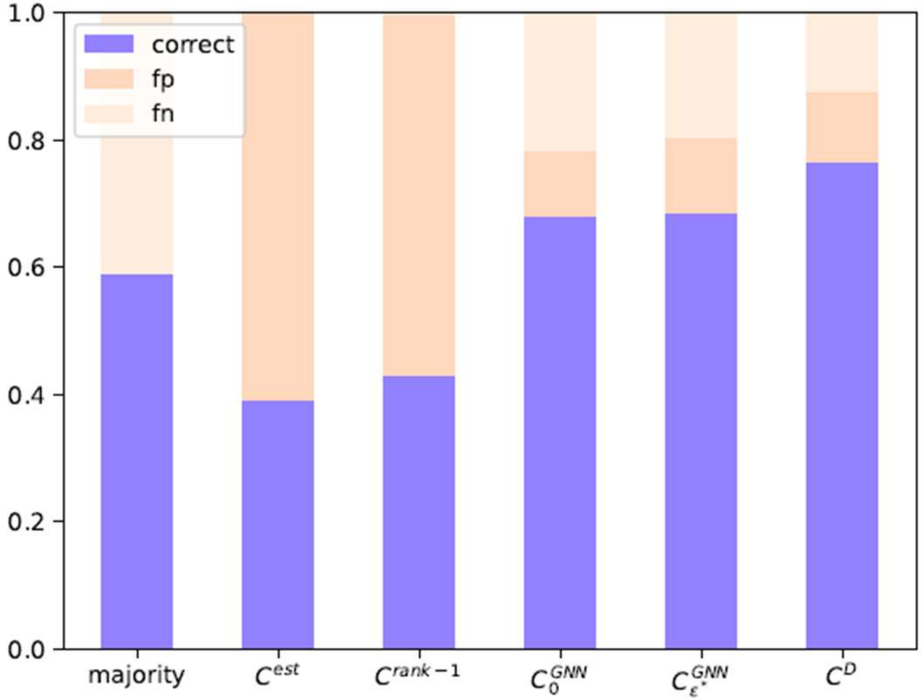


(a) Set covering

(b) Combinatorial auctions

Our classifiers

From Berthold et al. (2018)


1.0
0.8
0.6
0.4
0.2
0.0
correct
fp
fn
majority
C^{est}
C^{rank-1}
C_0^{GNN}
$C_{\varepsilon^*}^{GNN}$
C^D

(c) GISP

A Knapsack Game in Rounds

Given two agents each of them owning a set of n items.
Each item has a profit p and a weight w.

There is a knapsack with capacity c.
All information is public.

The **Knapsack Game** proceeds in rounds:

> *In each round the agents submit simultaneously one of their items (which must fit in the current knapsack).*
> *The item with higher efficiency p/w wins and is packed into the knapsack.*
> *Each agent wants to maximize the total profit of its packed items.*

Ulrich Pferschy, *joint work with Rosario Scatamacchia (Politecnico di Torino)*

A Knapsack Game in Rounds

Best Response of agent B against agent A:

- unknown strategy of $A \Longrightarrow$ outcome for B arbitrarily bad
- list strategy of A:
  A submits its items according to a predetermined list of items (known to B)
  $\Longrightarrow$ best response of B is a subset of items submitted by decreasing efficiencies
- this best response subset can be computed by Dynamic Programming and by an ILP model (specific versions if A follows a list sorted by decreasing efficiencies)
- no poly time response with bounded performance ratio, even if A sticks to an ordered list strategy
  (list sorted by decreasing efficiencies).

A Knapsack Game in Rounds

In progress:
Pure Nash Equilibrium, Subgame Perfect Equilibrium,

price of anarchy / price of stability arbitrarily large.

Variant of the problem:
losing items are permanently discarded

$\Longrightarrow$ additional difficulty: which item should be sacrificed?
$\Longrightarrow$ best response of B not necessarily sorted.

$\Longrightarrow$ ILP model for best response of B given a list of A.

Elevator pitch:
Ever wished a Lagrangian was as easy to use as a LP? Your wish is now granted

Antonio Frangioni[1]

[1] Dipartimento di Informatica, Università di Pisa

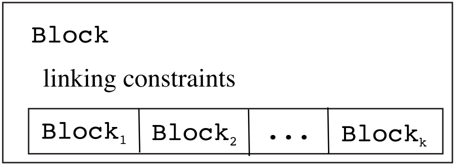
27th Combinatorial Optimization Workshop
Aussois (France), January 5 – 10, 2024

Have a block-structured program to solve

- . . . and always wondered if Lagrangian relaxation could be competitive

Have a block-structured program to solve

- . . . and always wondered if Lagrangian relaxation could be competitive
- Just write it as a `Block` with its sub-`Block` (recursively if needed)



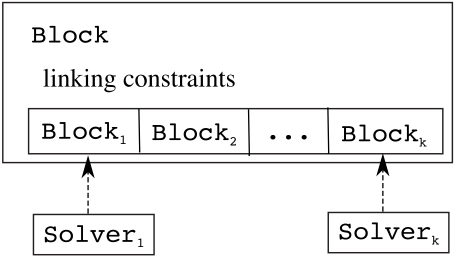
`Block`

linking constraints

`Block`$_1$ `Block`$_2$ . . . `Block`$_k$

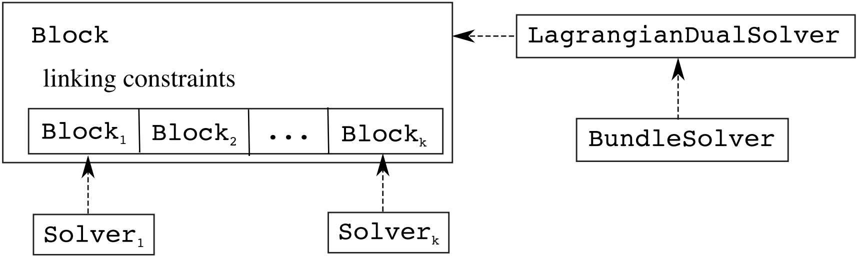
Have a block-structured program to solve

- …and always wondered if Lagrangian relaxation could be competitive
- Just write it as a `Block` with its sub-`Block` (recursively if needed)
- Attach `Solver` to each sub-`Block` (specialised if you have them)



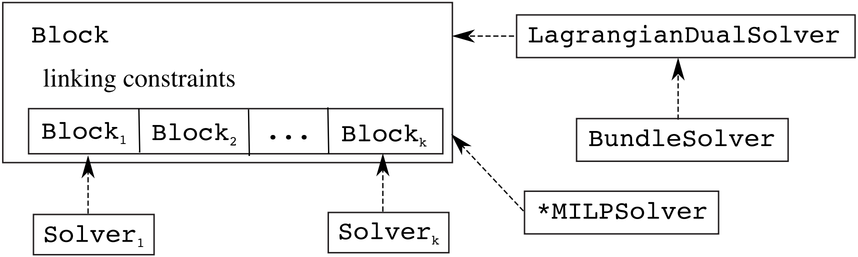
Have a block-structured program to solve

- …and always wondered if Lagrangian relaxation could be competitive
- Just write it as a `Block` with its sub-`Block` (recursively if needed)
- Attach `Solver` to each sub-`Block` (specialised if you have them)
- Attach `LagrangianDualSolver` to the `Block`, `compute()`, get optimal primal (convexified) and dual



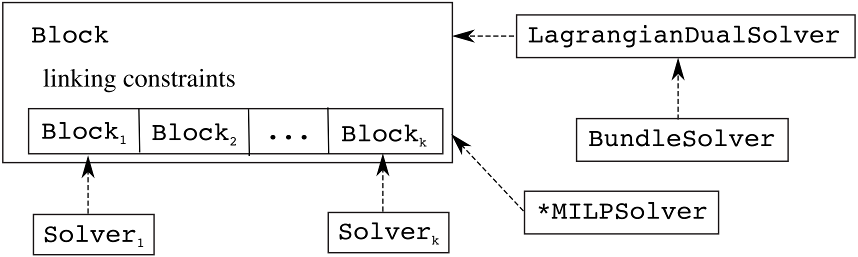
Have a block-structured program to solve

- …and always wondered if Lagrangian relaxation could be competitive
- Just write it as a `Block` with its sub-`Block` (recursively if needed)
- Attach `Solver` to each sub-`Block` (specialised if you have them)
- Attach `LagrangianDualSolver` to the `Block`, `compute()`, get optimal primal (convexified) and dual
- Attach `*MILPSolver` to the `Block`, `compute()`, get optimal primal and dual (of continuous relaxation) – together and in parallel if you want



Have a block-structured program to solve

- . . . and always wondered if Lagrangian relaxation could be competitive
- Just write it as a `Block` with its sub-`Block` (recursively if needed)
- Attach `Solver` to each sub-`Block` (specialised if you have them)
- Attach `LagrangianDualSolver` to the `Block`, `compute()`, get optimal primal (convexified) and dual
- Attach `*MILPSolver` to the `Block`, `compute()`, get optimal primal and dual (of continuous relaxation) – together and in parallel if you want
- Change anything in `Block`, `compute()` reoptimizes, get new stuff



One example: SDDP + Lagrange

- Mid-term (1y) stochastic energy reservoirs management, each a short-term (1w) unit commitment (≠ units, [HV]DC network, . . .)
- Perfect for Stochastic Dual Dynamic Programming, but need duals
- Either continuous relaxation of tight formulation or Lagrangian relaxation
- Cherry-picked result: 60 stages (1+ year), 37 scenarios, 168 instants (weekly UC), 83 thermals, 3 intermittent, 2 batteries, 1 hydro
- Out-of-sample simulation: all 37 scenarios to integer optimality

	Cont. relax.	Lag. relax.
Cost: Avg. / Std.	3.951e+11 / 1.608e+11	3.459e+11 / 8.903e+10
Time:	5h43m	7h54m

- Time OK using `ParallelBundleSolver` with 5 threads per scenario
- That's 14% just changing a few lines in the configuration

All this and much more awaits you in SMS++

https://gitlab.com/smspp/smspp-project

"For algorithm developers, from algorithm developers"

- Open source, extensive documentation https://smspp.gitlab.io (but only one User's Manual: me)
- Parallel features built-in the system, all three main OSs supported
- Now featuring 4 main MI*LP solvers (`Cplex`, `Gurobi`, `SCIP`, `HiGHS`) + a few specialised ones (flow, knapsack, 1UC, box, . . .)
- Community-oriented: easy to add your own project, join the fun

Capacitated Vehicle Routing with Fixed Order (CVRP-FOR)

Open Complexity on the line

Ekin Ergen (TU Berlin)
Steven Miltenburg (VU Amsterdam)
Rene Sitters (VU Amsterdam)

Leen Stougie (CWI and VU Amsterdam)

Problem setting

- n points in a metric space.
- A depot with sufficient vehicles
- Each vehicle has capacity c
- Vehicles return to the depot

after having served a subset of at most c points.

Problem setting

- n points in a metric space.
- A depot with sufficient vehicles
- Each vehicle has capacity c
- Vehicles return to the depot

after having served a subset of
at most c points.
Points are ordered!!!
On each vehicle the assigned
points must be served in
the order given

Problem setting

Problem setting

- n points in a metric space.
- A depot with sufficient vehicles
- Each vehicle has capacity c
- Vehicles return to the depot

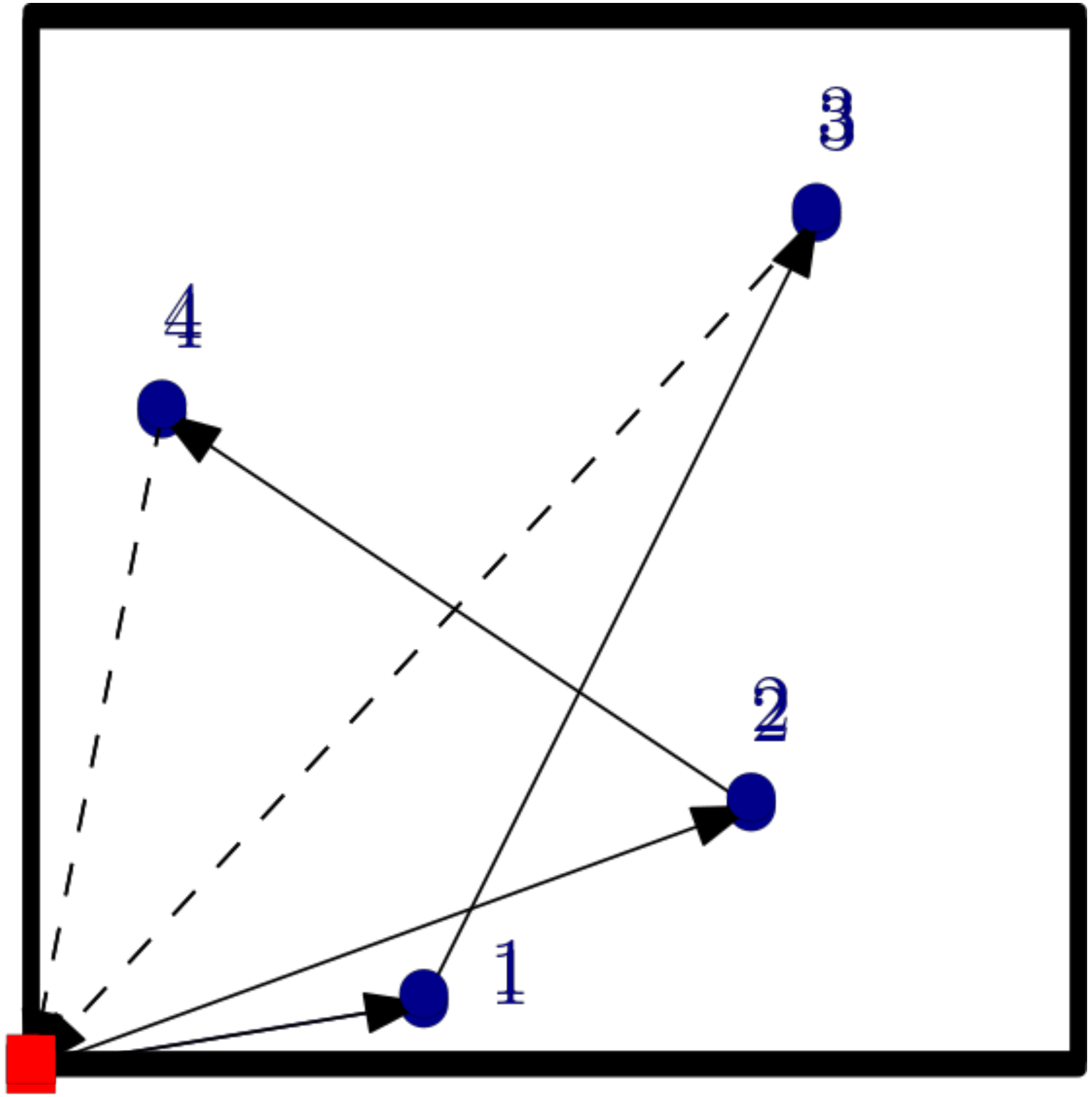
after having served a subset of at most c points.

Points are ordered!!!

Example with c=2, order {1,2,3,4}

Vehicle 1: {1,3}

Vehicle 2: {2,4}



Problem setting

Problem setting

- n points in a metric space.
- A depot with sufficient vehicles
- Each vehicle has capacity c
- Vehicles return to the depot

after having served a subset of
at most c points.
Points are ordered!!!

- Minimize Total distance travelled

Approximation on General Metric Spaces

- CVRP-FOR on general metric spaces
- Without FOR CVRP on general metric spaces is APX-hard and has a $1 + \alpha - \frac{1}{c}$ approximation scheme (Haimovich, Rinnooy Kan 1985) for $\alpha \approx \frac{3}{2}$ the TSP approximation.

  Improved very recently (Blauth, Traub, Vygen 2023) to 2.4997
- CVRP-FOR is still APX-hard (even if c=3), with a $1 + 1 - \frac{1}{c}$ approximation

Open Problem:

Can 2-1/c be improved?

Facts of Interest:

- Improvement of Blauth et al. does not work for CVRP-FOR

Complexity of CVRP-FOR on the line with fixed capacity $c = 3$?

Minimally Open Problem:

For $c = 3$ is CVRP-FOR on the line NP hard?

Facts of Interest:

- CVRP-FOR on the line is NP-hard for arbitrary c. Does a PTAS exist?

- For constant c a PTAS exists for CVRP-FOR on the line (extending to R^d).

Ergen & Miltenburg oral communication

A PTAS for CVRP-FOR on a tree?

Open Problem:

For constant $c \geq 3$ on a tree does a PTAS exist?

Facts of Interest:

- For constant c a PTAS exists for CVRP-FOR on the line.
- A PTAS exists for CVRP on a tree
  Mathieu & Zhou 2022

Thank You!

Theoretical properties of lower and upper bounds for the Bin Packing Problem with Setups

R. Baldacci [1], F. Ciccarelli [2], S. Coniglio [3], F. Furini [2]

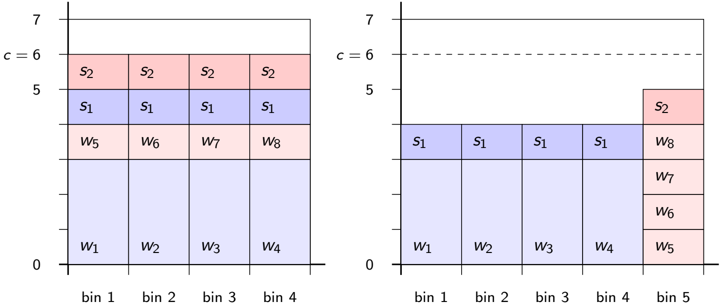
Hamad Bin Khalifa University [1], DIAG, Sapienza University of Rome [2], University of Bergamo [3]

January 6, 2025

The Bin Packing Problem with Setups (BPPS)

The BPPS is a generalization of the **Bin Packing Problem** (BPP), in which the item set N is partitioned into classes. Activating a class in a bin (i.e., packing at least one item of that class into it) incurs an additional capacity consumption as well as a setup cost. We refer to the sets of bins and classes as K and I, respectively.

Each item $j \in N$ has weight $w_j \in \mathbb{Z}^+$ and belongs to a class $t_j \in I$, while, for each class $i \in I$, we denote by $s_i \in \mathbb{Z}^+$ and $f_i \in \mathbb{Z}^+$ its *setup weight* and *setup cost*.



(a) high bin cost

(b) low bin cost

Natural Formulation for the BPPS

$$\min_{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})\in\{0,1\}} \quad \sum_{k\in K}\left(b\, z_k + \sum_{i\in I} f_i\, y_{ik}\right)$$

$$\sum_{k\in K} x_{jk} = 1 \qquad \forall j\in N,$$

$$c\, z_k - \sum_{j\in N} w_j\, x_{jk} - \sum_{i\in I} s_i\, y_{ik} \geq 0 \qquad \forall k\in K,$$

$$y_{t_j k} - x_{jk} \geq 0 \qquad \forall j\in N, k\in K.$$

Proposition

$$z(LP) = \sum_{i\in I} f_i + \frac{b}{c}\left(\sum_{j\in N} w_j + \sum_{i\in I} s_i\right)$$

Proposition

There exist BPPS instances for which:

$$\frac{z(LP)}{\mathrm{OPT}} \xrightarrow{|N|\to\infty} 0$$

The Minimum Class-occurrence Inequalities

We propose the following family of inequalities, which we refer to as **minimum class-occurrence inequalities (MCIs)**:

$$\sum_{k\in K} y_{ik} \geq \gamma_i \quad \text{where} \quad \gamma_i = \left\lceil \frac{\sum_{j\in N_i} w_j}{c - s_i} \right\rceil \quad \forall i \in I$$

Proposition

$$z(LP) = \sum_{i\in I} \gamma_i \, f_i + \frac{b}{c} \left(\sum_{j\in N} w_j + \sum_{i\in I} \gamma_i \, s_i \right)$$

Proposition

For all BPPS instances it holds that:

$$\frac{z(LP)}{\mathrm{OPT}} \geq \frac{1}{2}$$

Upper bounds to the optimal solution value of the BPPS

Proposition

Let $\mathcal{A}$ be an α-approximation algorithm (with $\alpha > 1$) for the `BPP`. It is then possible to derive a 2α-approximation algorithm for the `BPPS`.

Sketch of the algorithm:

1. Run $\mathcal{A}$ to pack the items of each class separately;
2. If possible, merge pairs of bins with free available capacity.

Open research paths

- Development of an approximation algorithm specifically tailored to the BPPS;
- New classes of lower bounds or bounds derived from new families of valid inequalities;
- Efficient set-partitioning formulations and column generation algorithms.

Thank you for your attention!

Questions?

Asymptotically Optimal Hardness for k-Set Packing & k-Matroid Intersection

Theophile Thiery

Joint Work: *Euiwoong Lee* (U.Mich) & *Ola Svensson* (EPFL)

27th Aussois Combinatorial Workshop

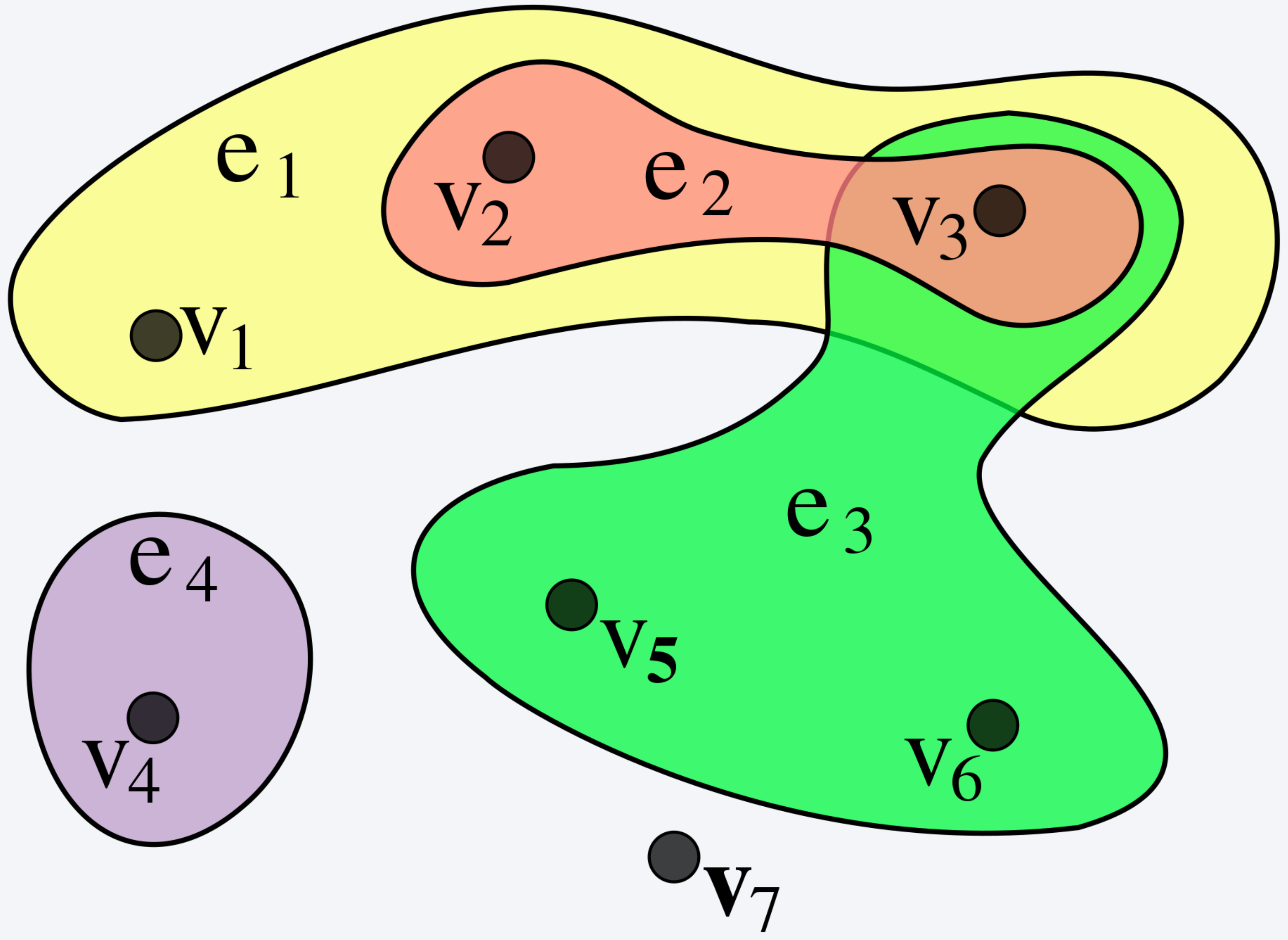
k-Set Packing

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

I. **Generalizes maximum matching** in graph (when, $k = 2$) and thus **models higher dependencies** in pratical applications.

II. **Benchmark problem**: listed in Karp's list of 21-NP complete problems for $k = 3$ and a special case of k-Matroid Intersection.



Result & Consequences

Main Theorem

For any $\varepsilon > 0$, and sufficiently large $k \geq k_\varepsilon$, the k-Set Packing problem is hard to approximate within a factor $k/(12+\varepsilon)$, unless $\mathbf{NP} \subseteq \mathbf{BPP}$.

Consequences

I. Improves over the $\Omega\left(\frac{k}{\log(k)}\right)$-hardness by Hazan, Safra and Schwartz'06 — consistently cited for maximizing linear and submodular function over k-Matroid Intersection, k-Matchoid, k-Matroid Parity.

II. **Asymptotically optimal** result and explains the lack of substantial progress beyond $O(k)$-approximation algorithms.

Brief History & Result

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

Known results over time

Brief History & Result

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

Known results over time

Brief History & Result

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

Known results over time

Brief History & Result

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

Known results over time

Brief History & Result

Problem Statement

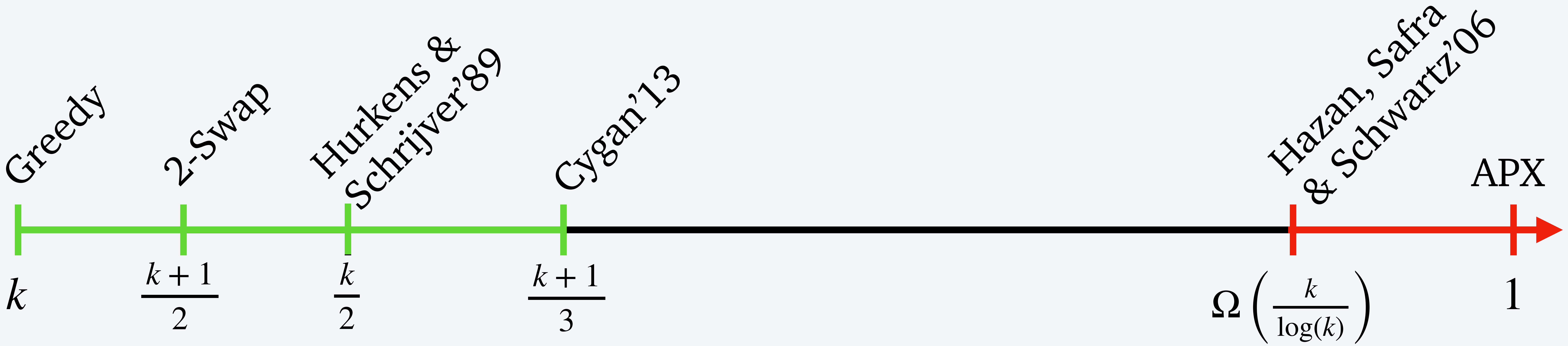
Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

Known results over time

Brief History & Result

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.

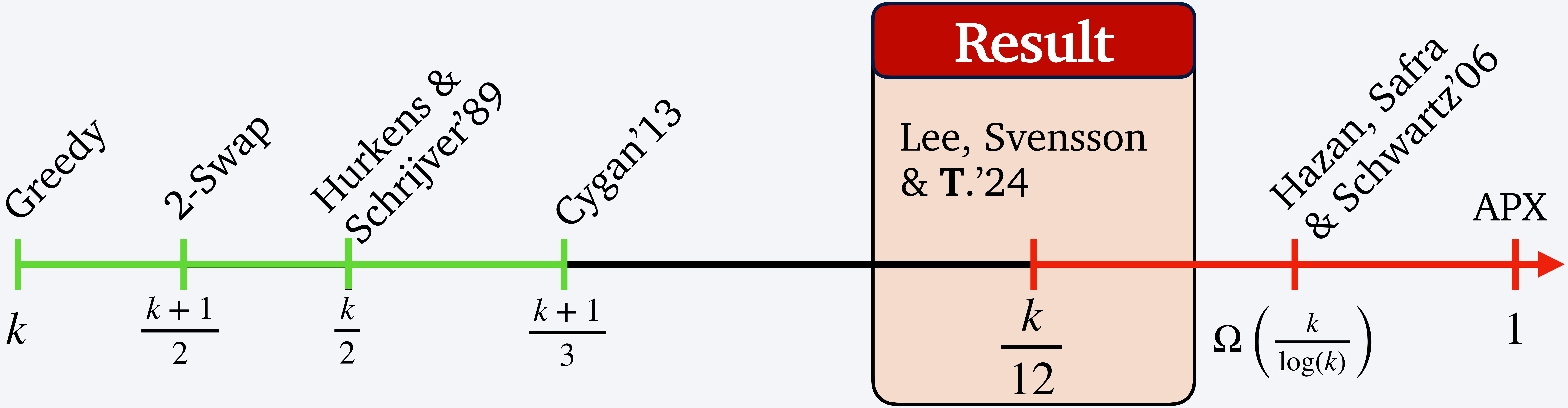


Known results over time

Brief History & Result

Problem Statement

Let $k \geq 3$ be some integer. Given a collection of sets, each containing up to k vertices. Find sub-collection of disjoint sets of maximum size.



Known results over time

Result & Brief Explaination

Main Theorem

For any $\varepsilon > 0$, and sufficiently large $k \geq k_\varepsilon$, the k-Set Packing problem is hard to approximate within a factor $k/(12 + \varepsilon)$, unless $\mathbf{NP} \subseteq \mathbf{BPP}$.

I. Following [HSS'06], we encode satisfying assignments of k-CSPs as large matchings. Our hyperedges correspond to constraints and labels that variables can take. **Invariant**: Two hyperedges should intersect if the assignment is not consistent.

II. The novelty in our approach is to **sparsify** CSP to **reduce the number of constraints** a variable appears in ($\leq$ alphabet size), which allows to design a *simple* gadget bypassing their tight construction.

Open Questions

Open Questions

I. **Close the Gap:** A better understanding of hardness of approximation of k-CSPs could lead to stronger hardness results.

II. What is the complexity of approximating a **monotone submodular function** subject to a k-set packing constraint?

III. New algo/hardness for k-SP in **online**/**streaming**/… settings.

Open Questions

Open Questions

I. **Close the Gap:** A better understanding of hardness of approximation of k-CSPs could lead to stronger hardness results.

II. What is the complexity of approximating a **monotone submodular function** subject to a k-set packing constraint?

III. New algo/hardness for k-SP in **online**/**streaming**/… settings.

I am on the job market from Fall 2025!

Model(s) for the homogeneous (tram) usage dispatching problem

Some remarks on work in progress

Michael Kahr[1] Markus Leitner[2] Rosario Paradiso[2]

[1]Department of Operations and Information Systems, University of Graz, Austria
[2]School of Business and Economics, Operations Analytics, Vrije Universiteit Amsterdam, The Netherlands

27th *Aussois* Combinatorial Optimization Workshop
Aussois, France, January 6, 2025

Problem description and Motivation

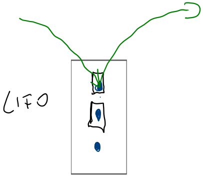
Problem description:

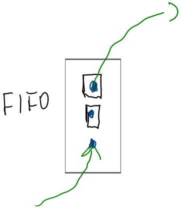
- Given a set of trams $\mathcal{T}$, a set of services $\mathcal{S}$ (or trips) they should perform, and a set of parking corridors $\mathcal{C}$ (with either LIFO, or FIFO queuing systems).
- The objective is to assign services to trams such that their utilization is (almost) balanced.

Motivation:

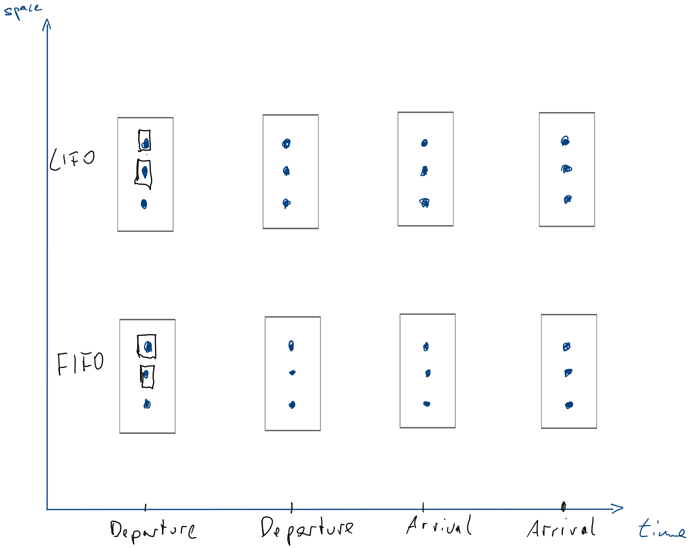
- Practical: improve planning (of maintenance and investment), reduce cost (by avoiding shunting).
- Scientific: real-world problem (data from Italy), modeling FIFO and LIFO queues, objective function structure.

Graph representation



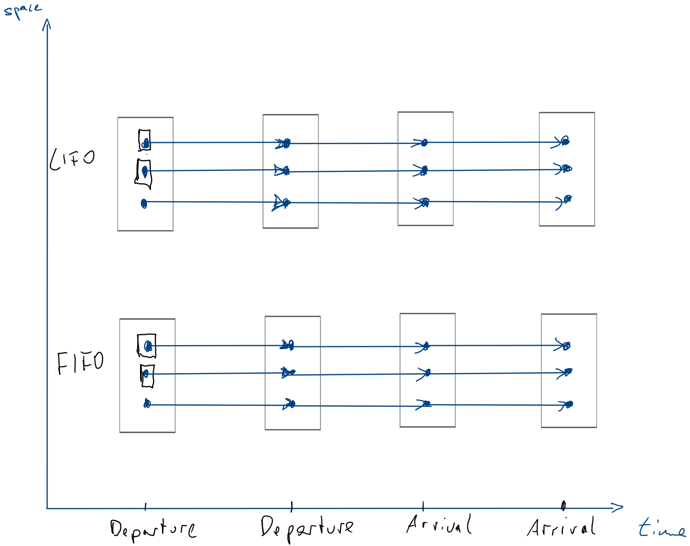


Graph representation

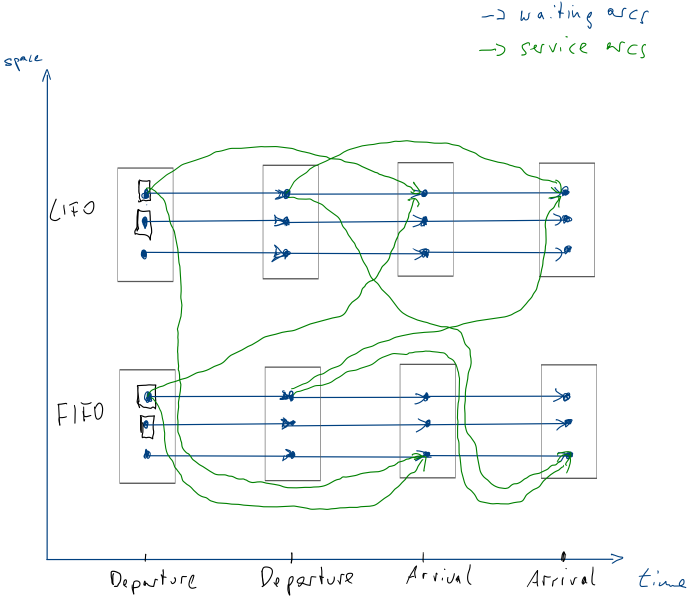


Graph representation

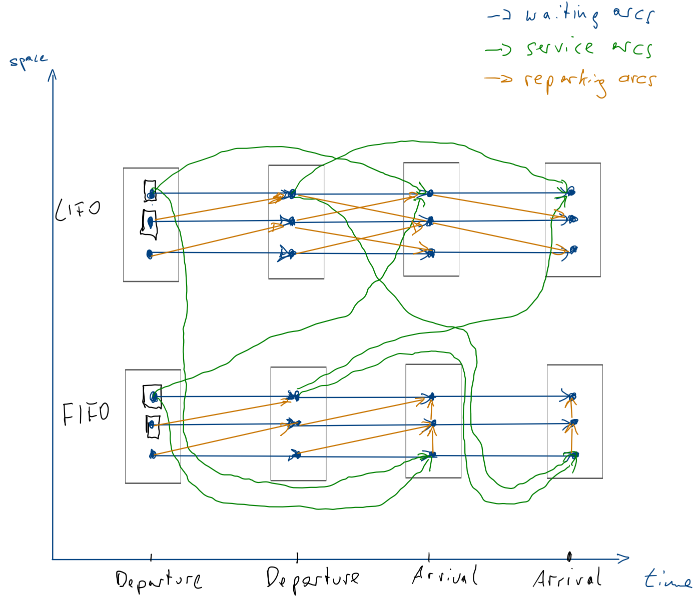
→ waiting arcs



Graph representation



Graph representation



Event based formulation

$$
\begin{aligned}
\text{s.t.} \quad & \sum_{c \in C} z_{ec} = 1 && \forall e \in \mathcal{E} \\
& z_{ec} \leq m_c - n_c + \sum_{f \in \mathcal{D}_e^<} z_{fc} - \sum_{f \in \mathcal{A}_e^<} z_{fc} && \forall e \in \mathcal{A},\ \forall c \in C \\
& z_{ec} \leq n_c + \sum_{f \in \mathcal{A}_e^<} z_{fc} - \sum_{f \in \mathcal{D}_e^<} z_{fc} && \forall e \in \mathcal{D}, \forall c \in C \\
& z_{ec} \in \{0, 1\} && \forall e \in \mathcal{E}, \forall c \in C
\end{aligned}
$$

Event based formulation

$$
\begin{aligned}
\min \quad & u_{\max} - u_{\min} & \\
\text{s.t.} \quad & \sum_{c \in C} z_{ec} = 1 & \forall e \in \mathcal{E} \\
& z_{ec} \leq m_c - n_c + \sum_{f \in \mathcal{D}_e^{<}} z_{fc} - \sum_{f \in \mathcal{A}_e^{<}} z_{fc} & \forall e \in \mathcal{A},\ \forall c \in C \\
& z_{ec} \leq n_c + \sum_{f \in \mathcal{A}_e^{<}} z_{fc} - \sum_{f \in \mathcal{D}_e^{<}} z_{fc} & \forall e \in \mathcal{D}, \forall c \in C \\
& z_{ec} \in \{0, 1\} & \forall e \in \mathcal{E}, \forall c \in C \\
& u_{\max} \geq u_{\max}(\bar{\mathbf{z}}) - \sum_{(e,c) \in \mathcal{E} \times C : \bar{z}_{ec} = 1} \Delta_{\max}(\bar{\mathbf{z}}, e)(1 - z_{ec}) & \forall \bar{\mathbf{z}} \in P(\mathbf{z}) \\
& u_{\min} \leq u_{\min}(\bar{\mathbf{z}}) + \sum_{(e,c) \in \mathcal{E} \times C : \bar{z}_{ec} = 1} \Delta_{\min}(\bar{\mathbf{z}}, e)(1 - z_{ec}) & \forall \bar{\mathbf{z}} \in P(\mathbf{z})
\end{aligned}
$$

Thank you!

A Linear Time Gap-ETH-Tight Approximation Scheme for TSP in the Euclidean Plane

Tobias Mömke
University of Augsburg

Joint work with Hang Zhou

27th Aussois Combinatorial Optimization Workshop, 2025

Euclidean TSP – Known Results

- **Arora [J. ACM 1998] and Mitchell [SICOMP 1999] (Gödel-Prize 2010):**

  Polynomial time $(1+\varepsilon)$-approximation algorithm, polynomial running time

- **Rao and Smith [STOC 1998]:**

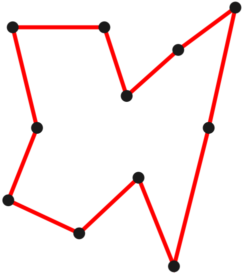
  Running time $(1/\varepsilon)^{O(1/\varepsilon)} n \log n$.

- **Bartal and Gottlieb [FOCS 2013]:**

  Running time $2^{(1/\varepsilon)^{O(1)}} n$, i.e., **linear** time

- **Kisfaludi-Bak, Nederlof, and Węgrzycki [FOCS 2021]:**

  Running time $2^{O(1/\varepsilon)} n \log n$, which is GAP-ETH tight



Our Result

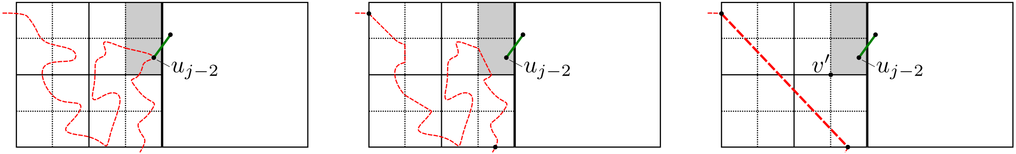
Theorem 1

There is a randomized $(1+\varepsilon)$*-approximation scheme for the Euclidean TSP in* $\mathbb{R}^2$ *that runs in time* $2^{O(1/\varepsilon)}n$ *in the real-RAM model with atomic floor operations.*

- Asymptotically tight unless the GAP-ETH is false
- Same machine model as Bartal and Gottlieb

Short Summary of Ideas

- Use sparsity-sensitive patching of Kisfaludi-Bak, Nederlof, and Węgrzycki if ≥ 2 crossings
- Ensure sufficient potential for single crossings:
    - Add portals of **2-approximate** solution
    - Long crossing edges: charge length of edge
    - Short crossing edges: charge approximate solution



`https://arxiv.org/abs/2411.02585`

Connectivity via convexity: Bounds on the edge expansion in graphs

Timotej Hrga, Melanie Siebenhofer, Angelika Wiegele

27th Aussois Combinatorial Optimization Workshop

January 2025

ALPEN-ADRIA
UNIVERSITÄT
KLAGENFURT

Edge Expansion

$$h(G) = \min_{S \subset V,\ 1 \leq |S| \leq \frac{n}{2}} \frac{|\partial S|}{|S|}$$

Edge Expansion

$$h(G) = \min_{S \subset V,\ 1 \leq |S| \leq \frac{n}{2}} \frac{|\partial S|}{|S|}$$

$$= \min\ \frac{x^\top L x}{e^\top x},\ \text{s.t.}\ \ 1 \leq e^\top x \leq \left\lfloor \frac{n}{2} \right\rfloor,\ x \in \{0,1\}^n.$$

Edge Expansion

$$h(G) = \min_{S\subset V,\ 1\le|S|\le\frac{n}{2}} \frac{|\partial S|}{|S|}$$

$$= \min \frac{x^\top Lx}{e^\top x}, \text{ s.t. } 1 \le e^\top x \le \left\lfloor \frac{n}{2} \right\rfloor, \ x \in \{0,1\}^n.$$

⇝ formulation as a completely positive program

$$\begin{aligned} \min \quad & \langle L, Y\rangle \\ \text{s.t.} \quad & (e_n^\top \quad 0_n^\top \quad 0_2^\top)y = 1 \\ & \operatorname{tr}(CYC^\top - Cyd^\top - dy^\top C^\top + \rho dd^\top) = 0 \\ & \operatorname{diag}(Y^{12}) = 0 \\ & \begin{pmatrix} Y & y \\ y^\top & \rho \end{pmatrix} \in \mathcal{CP}^{2n+3}. \end{aligned}$$

- doubly non-negative relaxation: $\mathcal{CP}$ constraint $\rightsquigarrow$ $\mathcal{DNN}$ constraint (non-negative & psd)

- doubly non-negative relaxation: $\mathcal{CP}$ constraint $\rightsquigarrow$ $\mathcal{DNN}$ constraint (non-negative & psd)
- facial reduction: reduce dimension from $2n+3$ to $n+1$.

- doubly non-negative relaxation: $\mathcal{CP}$ constraint $\rightsquigarrow$ $\mathcal{DNN}$ constraint (non-negative & psd)
- facial reduction: reduce dimension from $2n+3$ to $n+1$.
- strengthening the DNN relaxation by cutting planes

- doubly non-negative relaxation: $\mathcal{CP}$ constraint $\rightsquigarrow$ $\mathcal{DNN}$ constraint (non-negative & psd)
- facial reduction: reduce dimension from $2n + 3$ to $n + 1$.
- strengthening the DNN relaxation by cutting planes
- augmented Lagrangian algorithm with post-processing

- doubly non-negative relaxation: $\mathcal{CP}$ constraint $\rightsquigarrow$ $\mathcal{DNN}$ constraint (non-negative & psd)
- facial reduction: reduce dimension from $2n + 3$ to $n + 1$.
- strengthening the DNN relaxation by cutting planes
- augmented Lagrangian algorithm with post-processing
- relaxation yields strong lower bounds and is computationally efficient

code available at:

`github.com/melaniesi/CheegerConvexificationBounds.jl`

	n	time (sec)	gap (%)
moviegalaxies-52	59	21.5	0.3
highschool	70	52.4	1.7
sp-office	92	89.8	2.4
game-thrones	107	108.5	0.4
revolution	141	233.7	7.9
malariagenes-HVR1	307	2620.7	4.4

.

Implied Integrality in Mixed Integer Optimization

Rolf van der Hulst, Matthias Walter

- Presolving technique used by all major solvers
- Integrality of a variable is implied by the constraints and integrality of other variables.
- Existing methods detect implied integrality of one variable at a time.

$$\begin{aligned} &\cdots \\ 3x + 2y + z &= 4 \\ x, y &\in \mathbb{Z} \\ (z &\in \mathbb{Z}) \end{aligned}$$

Implied Integrality in Mixed Integer Optimization

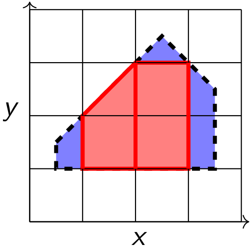
Rolf van der Hulst, Matthias Walter

- Presolving technique used by all major solvers
- Integrality of a variable is implied by the constraints and integrality of other variables.
- Existing methods detect implied integrality of one variable at a time.

$$\begin{aligned} &\dots \\ 3x + 2y + z &= 4 \\ x, y &\in \mathbb{Z} \\ (z &\in \mathbb{Z}) \end{aligned}$$

Implied integrality

For polyhedron $P \subseteq \mathbb{R}^N$ and $S, T \subseteq N$,
$\mathrm{conv}(P \cap (\mathbb{Z}^S \times \mathbb{R}^{N \setminus S})) = \mathrm{conv}(P \cap (\mathbb{Z}^{S \cup T} \times \mathbb{R}^{N \setminus (S \cup T)}))$

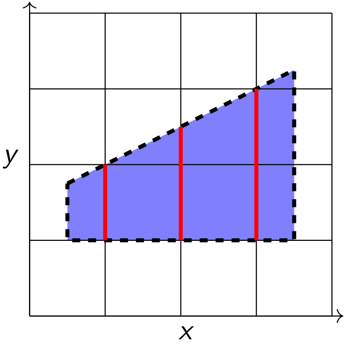


- Generalizes integer polyhedra ($S = \emptyset$, $T = N$)

Fibers and Totally Unimodular submatrices

Theorem (van der Hulst, Walter)

For $P \subseteq \mathbb{R}^N$, $S, T \subseteq N$, if each S-integral fiber is T-integral, then T is implied by S



- Fibers: $\{(\bar{x}, y) \mid By \leq b - A\bar{x}\}$ for fixed $\bar{x} \in \mathbb{Z}^S$.
- If and only if when S is binary
- Sufficient: B totally unimodular and b, A integral.
- Detect network matrices, ‘easy‘ subclass of TU

Results and Outlook

- MIPLIB 2017 benchmark set, statistics of presolved problems

Method	SCIP 9.0 default	TU detection
Mean % of i.i. variables	1.3%	**16.4%**
# affected instances (/240)	42	**162**

- Performance results are WIP, will be featured in SCIP 10

Future research and open questions:

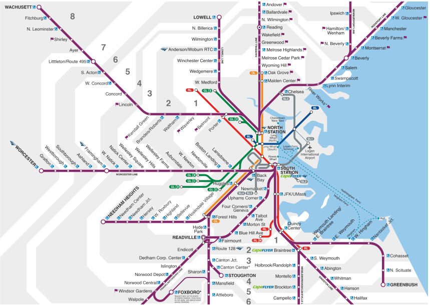
- Characterizations for relaxations of combinatorial optimization problems
- Complexity of recognizing implied integrality
  - At least co-NP hard, but no known certificate yet

Fare Zone Assignment

Lennart Kauther,
joint work with Sven Müller, Philipp Pabst, Britta Peis, and Khai Van Tran

January 6, 2025

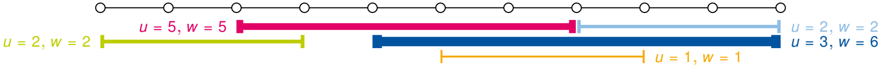
RWTH Aachen University



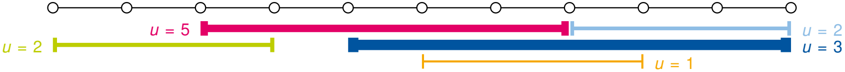
Problem Description

Input:

- Traffic network $G = (V, E)$
    - For this talk: G is a tree.
- For each commodity i:
    - Start- and endpoint s_i and t_i,
    - Maximum number of allowed tariff zone changes u_i,
    - Weight w_i

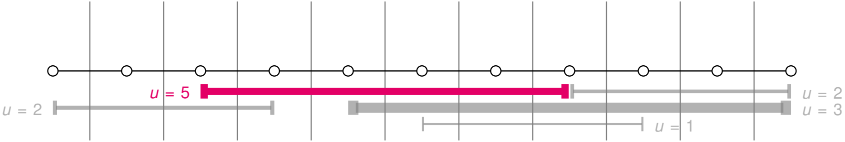


Goal: Find partition into fare zones that maximizes operator's revenue



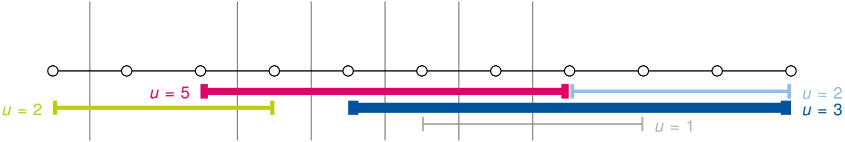
- u_i – upper bound on zone changes
- Revenue for commodity i: number of zones passed $\cdot$ w_i.
  - Revenue (no cut): $w_1 + w_2 + w_3 + w_4 + w_5$.

Goal: Find partition into fare zones that maximizes operator's revenue



- u_i – upper bound on zone changes
- Revenue for commodity i: number of zones passed $\cdot$ w_i.
  - Revenue (no cut): $w_1 + w_2 + w_3 + w_4 + w_5$.
  - Revenue (all cuts): $0 + 6 \cdot w_2 + 0 + 0 + 0$.

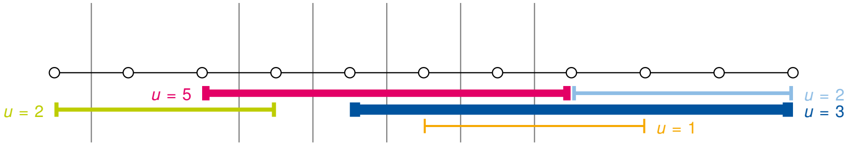
Goal: Find partition into fare zones that maximizes operator's revenue



- u_i – upper bound on zone changes
- Revenue for commodity i: number of zones passed $\cdot\ w_i$.
  - Revenue (no cut): $w_1 + w_2 + w_3 + w_4 + w_5$.
  - Revenue (all cuts): $0 + 6 \cdot w_2 + 0 + 0 + 0$.
  - Revenue (OPT): $3 \cdot w_1 + 6 \cdot w_2 + 4 \cdot w_3 + 0 + 1 \cdot w_5$.

FAREZONEASSIGNMENT

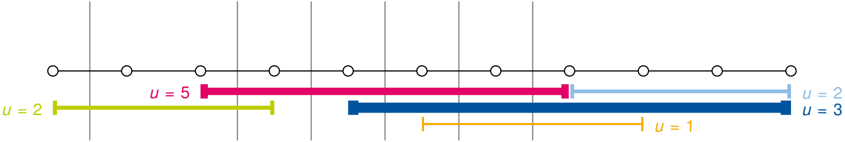
Goal: Find partition into fare zones that maximizes operator’s revenue



Results:

- NP-hard on paths.
- APX-hard on stars.
- Greedy arbitrarily bad.

Goal: Find partition into fare zones that maximizes operator’s revenue



Results:

- NP-hard on paths.
- APX-hard on stars.
- Greedy arbitrarily bad.

Open Problems:

- Constant-factor approximation?
- Greedy extension?

Support Vector Machines with ramp loss[1]

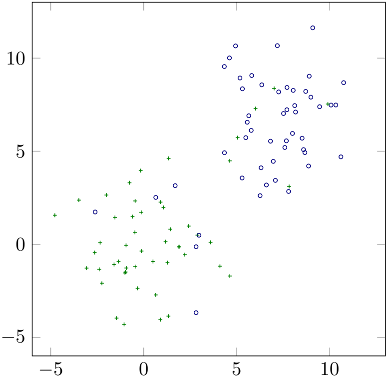
Data: $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$ for $i \in [n]$

---

[1] JP Brooks. "Support vector machines with the ramp loss and the hard margin loss." *Op. Res.* 59.2 (2011): 467-479

Support Vector Machines with ramp loss[1]

Data: $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$ for $i \in [n]$



---

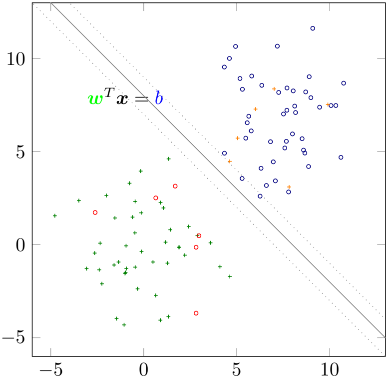
[1] JP Brooks. "Support vector machines with the ramp loss and the hard margin loss." *Op. Res.* 59.2 (2011): 467-479

Support Vector Machines with ramp loss[1]

Data: $\boldsymbol{x}_i \in \mathbb{R}^d, y_i \in \{-1, 1\}$ for $i \in [n]$
Find $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that:

$$\forall i : y_i = 1 \quad \boldsymbol{w}^T\boldsymbol{x}_i - b \geq 1$$
$$\forall i : y_i = -1 \quad \boldsymbol{w}^T\boldsymbol{x}_i - b \leq -1$$

$$\Rightarrow \forall i \in [n] \quad y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) \geq 1$$



---

[1] JP Brooks. "Support vector machines with the ramp loss and the hard margin loss." *Op. Res.* 59.2 (2011): 467-479

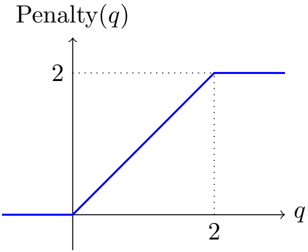
Support Vector Machines with ramp loss[1]

Data: $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$ for $i \in [n]$
Find $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that:

$$\begin{array}{ll} \forall i : y_i = 1 & \boldsymbol{w}^T \boldsymbol{x}_i - b \geq 1 \\ \forall i : y_i = -1 & \boldsymbol{w}^T \boldsymbol{x}_i - b \leq -1 \end{array}$$

$$\begin{array}{ll} \Rightarrow \ \forall i \in [n] & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 \\ \text{(Penalty)} & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i \end{array}$$

Penalize violation q: $\begin{cases} 0 & \text{if } q \leq 0 \\ q & \text{if } q \in (0, 2] \\ 2 & \text{if } q > 2. \end{cases}$



---

[1] JP Brooks. "Support vector machines with the ramp loss and the hard margin loss." *Op. Res.* 59.2 (2011): 467-479

Support Vector Machines with ramp loss[1]

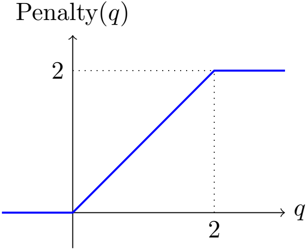
Data: $\boldsymbol{x}_i \in \mathbb{R}^d, y_i \in \{-1, 1\}$ for $i \in [n]$
Find $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that:

$$\begin{array}{ll} \forall i : y_i = 1 & \boldsymbol{w}^T \boldsymbol{x}_i - b \geq 1 \\ \forall i : y_i = -1 & \boldsymbol{w}^T \boldsymbol{x}_i - b \leq -1 \end{array}$$

$$\begin{array}{ll} \Rightarrow \ \forall i \in [n] & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 \\ \text{(Penalty)} & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i \end{array}$$

$$\text{Penalize violation } q\text{: } \begin{cases} 0 & \text{if } q \leq 0 \\ q & \text{if } q \in (0, 2] \\ 2 & \text{if } q > 2. \end{cases}$$

Minimize $\frac{C}{n}$(total penalty) $+\frac{1}{2}||\boldsymbol{w}||_2^2$.



---

[1] JP Brooks. "Support vector machines with the ramp loss and the hard margin loss." *Op. Res.* 59.2 (2011): 467-479

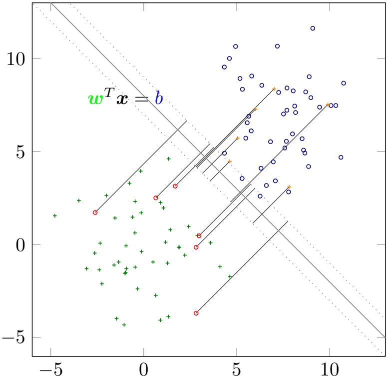
Support Vector Machines with ramp loss[1]

Data: $\boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$ for $i \in [n]$

Find $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that:

$$\begin{array}{ll} \forall i : y_i = 1 & \boldsymbol{w}^T \boldsymbol{x}_i - b \geq 1 \\ \forall i : y_i = -1 & \boldsymbol{w}^T \boldsymbol{x}_i - b \leq -1 \end{array}$$

$$\begin{array}{ll} \Rightarrow \ \forall i \in [n] & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 \\ \text{(Penalty)} & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i \end{array}$$



$$\text{Penalize violation } q\text{: } \begin{cases} 0 & \text{if } q \leq 0 \\ q & \text{if } q \in (0, 2] \\ 2 & \text{if } q > 2. \end{cases}$$

Minimize $\frac{C}{n}$(total penalty) $+\frac{1}{2}||\boldsymbol{w}||_2^2$.

---

[1] JP Brooks. "Support vector machines with the ramp loss and the hard margin loss." *Op. Res.* 59.2 (2011): 467-479

A Mixed Integer Quadratic Optimization model

$$\begin{array}{lll} \min_{\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{z}} & \frac{1}{2}||\boldsymbol{w}||_2^2 + \frac{C}{n} \sum_{i \in [n]} (\xi_i + 2z_i) & \\ \text{s.t.} & y_i(\boldsymbol{w}^T \boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i & \forall i \in [n] \\ & \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}, \boldsymbol{\xi} \in [0, 2]^n, \boldsymbol{z} \in \{0, 1\}^n. & \end{array}$$

Tight bounds $[\boldsymbol{\ell}, \boldsymbol{u}]$ on $(\boldsymbol{w}, b)$ $\Rightarrow$ small M_i's $\Rightarrow$ tighter formulation.

A Mixed Integer Quadratic Optimization model

$$\begin{array}{lll} \min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{z}} & \frac{1}{2}||\boldsymbol{w}||_2^2 + \frac{C}{n}\sum_{i\in[n]}(\xi_i + 2z_i) & \\ \text{s.t.} & y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i & \forall i \in [n] \\ & \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}, \boldsymbol{\xi} \in [0,2]^n, \boldsymbol{z} \in \{0,1\}^n. & \end{array}$$

Tight bounds $[\boldsymbol{\ell}, \boldsymbol{u}]$ on $(\boldsymbol{w}, b)$ ⇒ small M_i's ⇒ tighter formulation.

$$\text{Coeff. matrix: } \left(\begin{array}{c|c|ccc|ccc} \overbrace{\quad\quad}^{\boldsymbol{w}} & \overbrace{\quad\quad}^{b} & & \overbrace{\quad\quad\quad\quad}^{\boldsymbol{\xi}} & & & \overbrace{\quad\quad\quad\quad}^{\boldsymbol{z}} & \\ y_1\boldsymbol{x}_1^T & -y_1 & 1 & & & M_1 & & \\ \vdots & & & \ddots & & & \ddots & \\ y_n\boldsymbol{x}_n^T & -y_n & & & 1 & & & M_n \end{array}\right)$$

A Mixed Integer Quadratic Optimization model

$$\begin{array}{lll} \min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{z}} & \frac{1}{2}||\boldsymbol{w}||_2^2 + \frac{C}{n}\sum_{i\in[n]}(\xi_i + 2z_i) & \\ \text{s.t.} & y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i & \forall i \in [n] \\ & \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}, \boldsymbol{\xi} \in [0,2]^n, \boldsymbol{z} \in \{0,1\}^n. & \end{array}$$

Tight bounds $[\boldsymbol{\ell}, \boldsymbol{u}]$ on $(\boldsymbol{w}, b)$ ⇒ small M_i's ⇒ tighter formulation.

$$\text{Coeff. matrix:} \quad \begin{array}{c|c|ccc|ccc} \overbrace{\quad\;\;}^{\boldsymbol{w}} & \overbrace{\quad\;\;}^{b} & & \overbrace{\qquad\qquad}^{\boldsymbol{\xi}} & & & \overbrace{\qquad\qquad}^{\boldsymbol{z}} & \\ y_1\boldsymbol{x}_1^T & -y_1 & 1 & & & M_1 & & \\ \vdots & & & \ddots & & & \ddots & \\ y_n\boldsymbol{x}_n^T & -y_n & & & 1 & & & M_n \end{array} \Rightarrow \boxed{\begin{array}{c}\text{Branch} \\ \text{on } \boldsymbol{w}, b\end{array}}$$

A Mixed Integer Quadratic Optimization model

$$\begin{array}{lll} \min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{z}} & \frac{1}{2}||\boldsymbol{w}||_2^2 + \frac{C}{n}\sum_{i\in[n]}\left(\xi_i + 2z_i\right) & \\ \text{s.t.} & y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i & \forall i \in [n] \\ & \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}, \boldsymbol{\xi} \in [0,2]^n, \boldsymbol{z} \in \{0,1\}^n. & \end{array}$$

Tight bounds $[\boldsymbol{\ell}, \boldsymbol{u}]$ on $(\boldsymbol{w}, b)$ $\Rightarrow$ small M_i's $\Rightarrow$ tighter formulation.

Coeff. matrix:

$$\begin{array}{cccc} \boldsymbol{w} & b & \boldsymbol{\xi} & \boldsymbol{z} \\ \left(\begin{array}{c|c|ccc|ccc} y_1\boldsymbol{x}_1^T & -y_1 & 1 & & & M_1 & & \\ \vdots & & & \ddots & & & \ddots & \\ y_n\boldsymbol{x}_n^T & -y_n & & & 1 & & & M_n \end{array}\right) \end{array} \Rightarrow \boxed{\begin{array}{l}\text{Branch} \\ \text{on } \boldsymbol{w}, b\end{array}}$$

- Select w_j or b and $\tau \in (\ell_j, u_j)$
- Branching rule: $w_j \leq \tau \vee w_j \geq \tau$ (or $b \leq \tau \vee b \geq \tau$)

A Mixed Integer Quadratic Optimization model

$$\begin{array}{lll} \min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{z}} & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \frac{C}{n}\sum_{i\in[n]}(\xi_i + 2z_i) & \\ \text{s.t.} & y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i & \forall i \in [n] \\ & \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}, \boldsymbol{\xi} \in [0,2]^n, \boldsymbol{z} \in \{0,1\}^n. & \end{array}$$

Tight bounds $[\boldsymbol{\ell}, \boldsymbol{u}]$ on $(\boldsymbol{w}, b) \Rightarrow$ small M_i's $\Rightarrow$ tighter formulation.

Coeff. matrix:

$$\left(\begin{array}{c|c|ccc|ccc} \overbrace{\qquad}^{\boldsymbol{w}} & \overbrace{\qquad}^{b} & & \overbrace{\qquad\qquad}^{\boldsymbol{\xi}} & & & \overbrace{\qquad\qquad}^{\boldsymbol{z}} & \\ y_1\boldsymbol{x}_1^T & -y_1 & 1 & & & M_1 & & \\ \vdots & & & \ddots & & & \ddots & \\ y_n\boldsymbol{x}_n^T & -y_n & & & 1 & & & M_n \end{array}\right) \Rightarrow \boxed{\text{Branch on } \boldsymbol{w}, b}$$

- Select w_j or b and $\tau \in (\ell_j, u_j)$
- Branching rule: $w_j \leq \tau \vee w_j \geq \tau$ (or $b \leq \tau \vee b \geq \tau$)
- Heuristic based on LP solution to find good (w_j, τ) or (b, τ)

A Mixed Integer Quadratic Optimization model

$$
\begin{array}{lll}
\min_{\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{z}} & \frac{1}{2}||\boldsymbol{w}||_2^2 + \frac{C}{n}\sum_{i\in[n]}(\xi_i + 2z_i) & \\
\text{s.t.} & y_i(\boldsymbol{w}^T\boldsymbol{x}_i - b) \geq 1 - \xi_i - M_i z_i & \forall i \in [n] \\
 & \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}, \boldsymbol{\xi} \in [0,2]^n, \boldsymbol{z} \in \{0,1\}^n. &
\end{array}
$$

Tight bounds $[\boldsymbol{\ell}, \boldsymbol{u}]$ on $(\boldsymbol{w}, b) \Rightarrow$ small M_i's $\Rightarrow$ tighter formulation.

$$
\text{Coeff. matrix:} \quad
\begin{array}{c}
\begin{array}{cccc} \overbrace{\qquad}^{\boldsymbol{w}} & \overbrace{\qquad}^{b} & \overbrace{\qquad\qquad\qquad}^{\boldsymbol{\xi}} & \overbrace{\qquad\qquad\qquad\qquad}^{\boldsymbol{z}} \end{array} \\
\left(\begin{array}{c|c|ccc|ccc}
y_1\boldsymbol{x}_1^T & -y_1 & 1 & & & M_1 & & \\
\vdots & & & \ddots & & & \ddots & \\
y_n\boldsymbol{x}_n^T & -y_n & & & 1 & & & M_n
\end{array}\right)
\end{array}
\Rightarrow \boxed{\begin{array}{c}\text{Branch} \\ \text{on } \boldsymbol{w}, b\end{array}}
$$

- Select w_j or b and $\tau \in (\ell_j, u_j)$
- Branching rule: $w_j \leq \tau \vee w_j \geq \tau$ (or $b \leq \tau \vee b \geq \tau$)
- Heuristic based on LP solution to find good (w_j, τ) or (b, τ)
- Apply branching rule + **all tighter big-M constraints**

Computational tests[2] on known instances[3]

- $n = 100, d = 2$
- 2hr time limit
- Xpress 9.2, C API
- branching callbacks

	Binary branch (z)		Branch on w, b	
Inst.	t(BB)	Nodes	t(BB)	Nodes
1	1470.3	13M	**8.1**	6491
2	88.7	834k	**14.2**	13991
3	613.8	6909k	**24.0**	25241
4	388.5	3255k	**7.1**	5743
5	16.8	68058	**11.5**	11228
6	120.8	1086k	**21.3**	21813
7	181.0	1660k	**6.6**	5315
8	107.3	683k	**10.8**	10040
9	131.2	1515k	**18.2**	17929
10	84.0	929k	**6.1**	4931
11	11.8	153k	**8.6**	7831
12	121.0	1083k	**15.0**	14221
13	53.5	422k	**5.2**	4505
14	18.1	151k	**7.5**	6191
15	40.6	324k	**11.1**	10947
16	22.3	115k	**4.9**	3949
17	**6.1**	66173	6.1	5231
18	18.7	214k	**8.3**	8230

---

[2] PB, "Spatial branching for a special class of convex MIQO problems", *Optimization Letters* 18.8 (2024): 1757-1770

[3] PB, P Bonami, M Fischetti, A Lodi, M Monaci, A Nogales-Gómez, D Salvagnin. "On handling indicator constraints in mixed integer programming." *Computational Optimization and Applications* 65 (2016): 545-566.

Robust optimization approaches for the Multiple Suppliers Purchase Planning Problem under Uncertainty

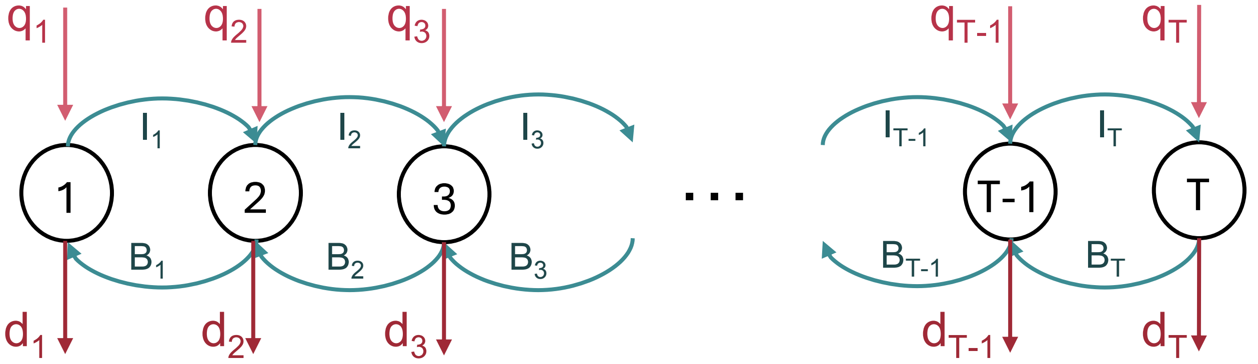
Gentile C.[1], **Giancola F.**[1,2], Mattia S.[1]

[1] Institute for System Analysis and Computer Science "Antonio Ruberti" (IASI), National Research Council of Italy, Via dei Taurini 19, 00185, Rome, Italy

[2] Department of Computer, Control and Management Engineering Antonio Ruberti (DIAG),Sapienza University of Rome, Italy

Introduction: Multiple Supplier Selection and Purchase Planning Problem

- **The problem:** Definition of a purchase plan to meet ***dynamic demand*** by minimizing costs in a ***multiperiod*** model, which balances supply (q_t^s) from **multiple suppliers** (S) with demand (d_t) over a planning horizon (T) while managing inventory levels (I_t) and backorders (B_t).



Single supplier s

- **Key challenge:** Uncertainty in suppliers lead times
- **Our approach:** Robust optimization algorithms to find valuable solutions even in the worst-case scenario

Robust optimization approaches

Comparison of three robust optimization models:

1. **Adjustable Unrestricted Model (Two-stage)**
   - Fully adjustable decisions (max flexibility).
   - High computational complexity (NP hard).

2. **Static Model (Single-stage)**
   - Fixed decisions for all scenarios (worst case).
   - Computationally efficient but very conservative.

3. **Partially Adjustable Model**
   - Hybrid approach that combines the static and the adjustable methods.
   - The planning horizon is divided into two phases, each with a different level of adaptability.
   - Balanced approach (tradeoff between flexibility and complexity).

Key Contribution:

- ❖ Comparative analysis of these models in terms of solution quality and computational feasibility.
- ❖ Practical insights for supply chain decision makers on managing uncertainty in supplier lead times.

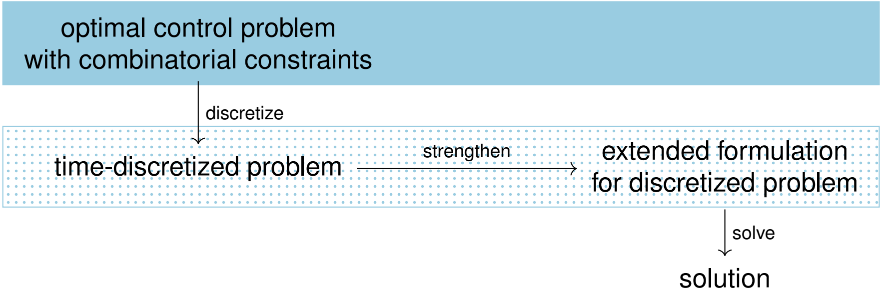
Thank you!

Extended Formulations for Control Languages Defined by Finite-State Automata

Maximilian Merkert[1], Christoph Buchheim[2], 27th Aussois COW, January 6, 2025

[1]TU Braunschweig, [2]TU Dortmund

Extended Formulations for the Set of Feasible Controls



optimal control problem
with combinatorial constraints

discretize

time-discretized problem

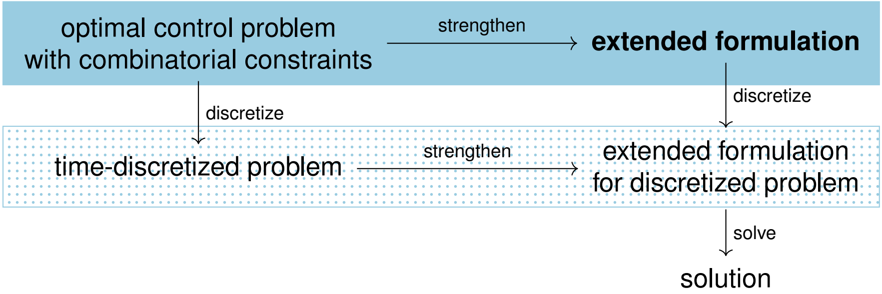
strengthen

extended formulation
for discretized problem

solve

solution

Extended Formulations for the Set of Feasible Controls



optimal control problem
with combinatorial constraints

strengthen

extended formulation

discretize

discretize

time-discretized problem

strengthen

extended formulation
for discretized problem

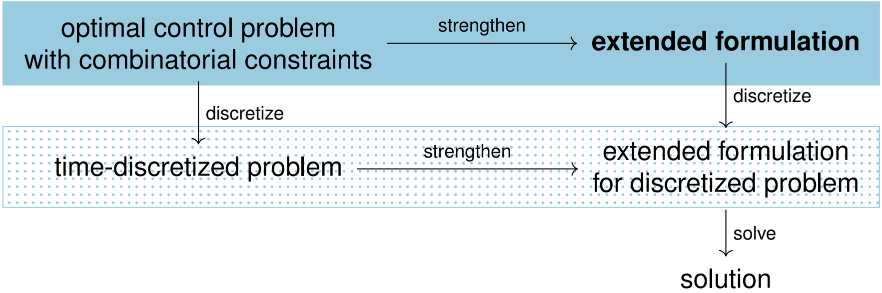
solve

solution

Advantages:

- Convex-hull formulation in the space of controls is independent of the discretization.
- Methods such as combinatorial integral approximation [Sager, Jung, Kirches, 2011] benefit from strong continuous relaxations.

Extended Formulations for the Set of Feasible Controls



optimal control problem
with combinatorial constraints

strengthen

extended formulation

discretize

discretize

time-discretized problem

strengthen

extended formulation
for discretized problem

solve

solution

Advantages:

- Convex-hull formulation in the space of controls is independent of the discretization.
- Methods such as combinatorial integral approximation [Sager, Jung, Kirches, 2011] benefit from strong continuous relaxations.

But: Very few such formulations known; started only recently with [Buchheim, 2024].

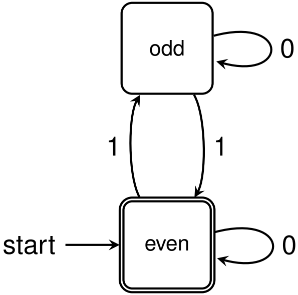
Finite-State Automata & Extended Formulations

Proposition (Fiorini, Pashkovich, 2015)

Let $\mathcal{L}$ denote a language over $\Sigma = \{0, 1\}$ and $M = (Q, \delta, \Sigma, q_0, F)$ be any deterministic finite-state automaton recognizing the language $\mathcal{L}$. Then for each $n \in \mathbb{N}_+$, there exists an extended formulation of

$$\operatorname{conv}\{x \in \{0, 1\}^n \mid x \in \mathcal{L}\}$$

with size at most $2|Q|n$.



Example: Even Parity

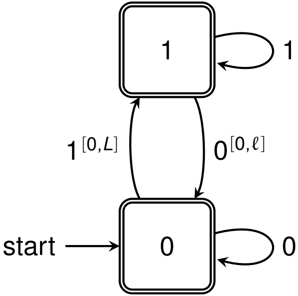
Finite-State Automata & Extended Formulations

Theorem (Buchheim, M., 2024)

Let $\mathcal{L}$ denote a language over $\Sigma \subseteq \mathbb{R}^n$ and $M = (Q, \delta, \Sigma, q_0, F)$ be any ***finite-state control automaton*** *recognizing the language $\mathcal{L}$. Then for every $T \in \mathbb{Q}_+$ there exists an extended formulation of*

$$\overline{\operatorname{conv}}(u \in \mathrm{BV}([0,T],\Sigma) \mid u \in \mathcal{L})$$

with polynomially many controls and linear constraints.



Example: Min-Up/Down

Summary

- Main result transfers large class of extended formulations to function space.
- We provide tools for non-representability proofs.
- Some surprises, e.g. any discretization regular $\neq$ regular as a control language

- Preprint on finite-state control automata and convex-hull descriptions in function space
  → [Buchheim, M.: *Extended Formulations for Control Languages Defined by Finite-State Automata*, Preprint (Optimization Online), 2024.]

Summary

- Main result transfers large class of extended formulations to function space.
- We provide tools for non-representability proofs.
- Some surprises, e.g. any discretization regular $\neq$ regular as a control language

- Preprint on finite-state control automata and convex-hull descriptions in function space
  → [Buchheim, M.: *Extended Formulations for Control Languages Defined by Finite-State Automata*, Preprint (Optimization Online), 2024.]

Thank you for your attention!

Analyzing the Sensitivity of Integer Linear Programs via Optimization Oracles

Erik Jansen

joint work with Marc E. Pfetsch

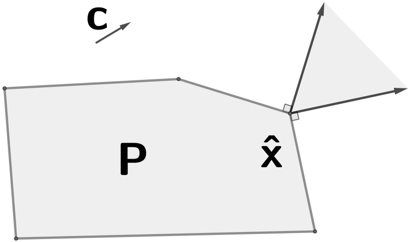
Funding by the Hessian Ministry of Higher Education, Research, Science and the Arts — cluster project Clean Circles

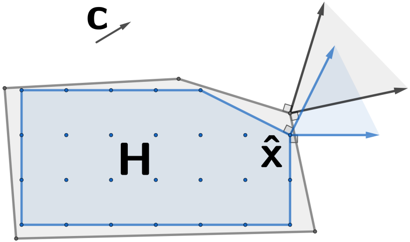
Motivation

Integer Program

$$\begin{aligned} \max_{x} \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b, \\ & x \in \mathbb{Z}^n \end{aligned} \tag{IP}$$

How much can we change the objective *c* without changing the optimal solution(s) $\hat{x}$?



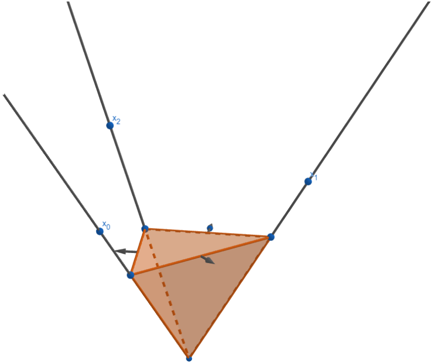


Oracle-Based Radial Cone Algorithm

built upon IPO (Walther, 2016)

Input: Optimization oracle $O(\cdot)$, Vertex of Interest $\hat{x}$
Output: Incidence graph T of radial cone at $\hat{x}$

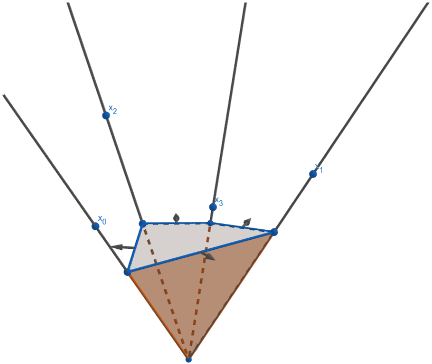
```
$T \leftarrow$ Initial-Conical-Hull($O(\cdot), \hat{x}$)
$F \leftarrow$ Set of active Facets

while $F \neq \emptyset$ do
    Select $f \in F$
    $x \leftarrow O(c_f)$                     // Run the Oracle with $c_f$
    if $c_f x > \delta_f$ then
        $(T, F) \leftarrow$ Cone-Update($T, x, F, f$) // Update Cone
    else
        $F \leftarrow F \setminus \{f\}$           // Set facet $f$ inactive
end
return $T$
```



Oracle-Based Radial Cone Algorithm

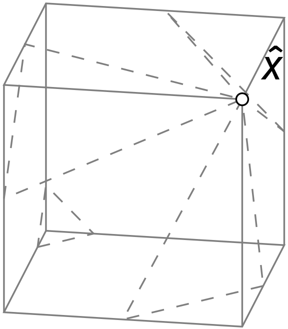
built upon IPO (Walther, 2016)

```
Input: Optimization oracle $O(\cdot)$, Vertex of Interest $\hat{x}$
Output: Incidence graph $T$ of radial cone at $\hat{x}$

$T \leftarrow$ Initial-Conical-Hull($O(\cdot), \hat{x}$)
$F \leftarrow$ Set of active Facets

while $F \neq \emptyset$ do
    Select $f \in F$
    $x \leftarrow O(c_f)$                   // Run the Oracle with $c_f$
    if $c_f x > \delta_f$ then
        $(T, F) \leftarrow$ Cone-Update($T, x, F, f$) // Update Cone
    else
        $F \leftarrow F \setminus \{f\}$          // Set facet $f$ inactive
end
return $T$
```

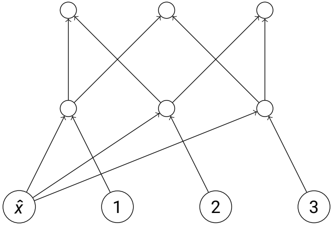


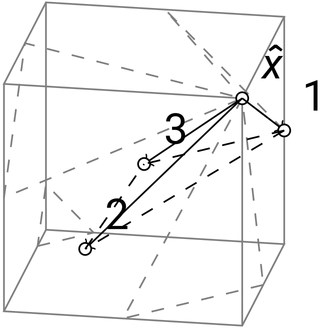
Small Example

$$
\begin{array}{llllll}
 & \min\limits_{x} & & \mathbb{1}^T x & & \\
\text{s.t.} & -0.5x & +y & +z & \leq & 1.5 \\
 & x & -0.5y & +z & \leq & 1.5 \\
 & x & +y & -0.5z & \leq & 1.5 \\
 & -x & -y & -z & \leq & -0.2 \\
 & & & 0 \leq x, y, z \leq 1 & &
\end{array}
$$

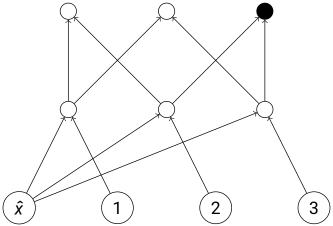


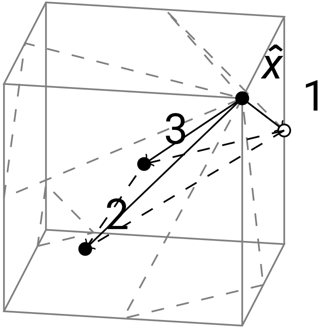
Small Example



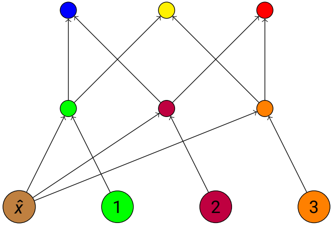


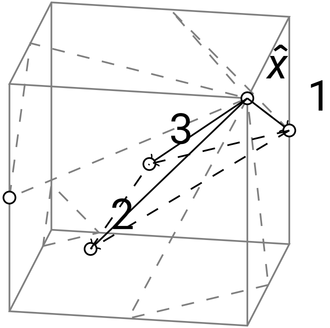
Small Example



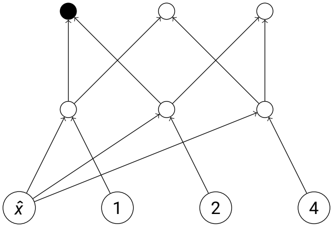


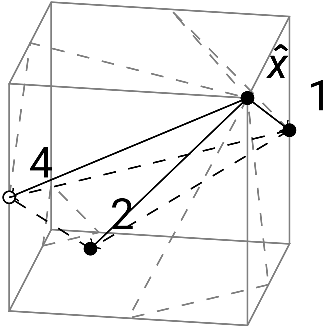
Small Example



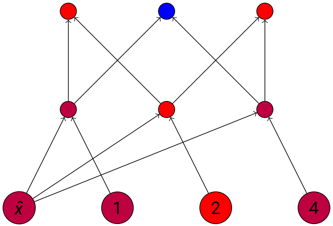


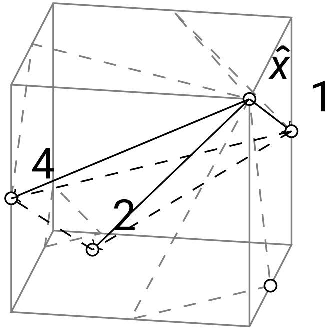
Small Example



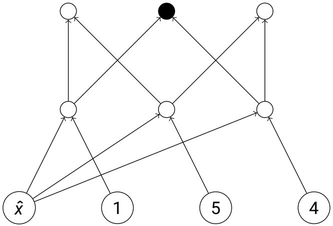


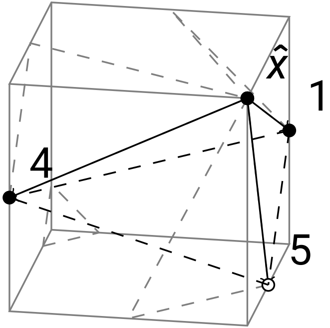
Small Example



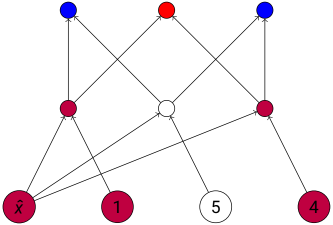


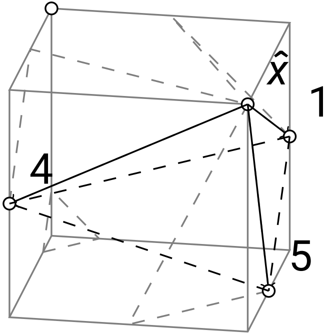
Small Example



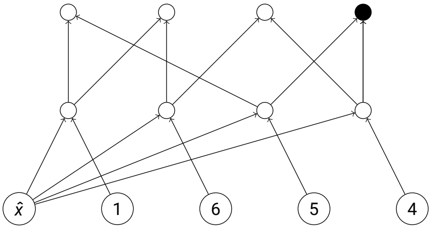


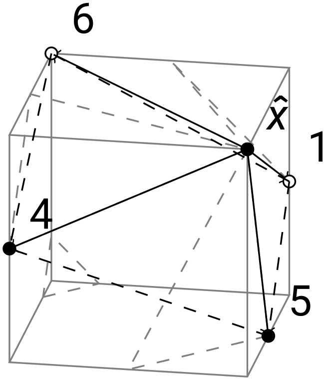
Small Example



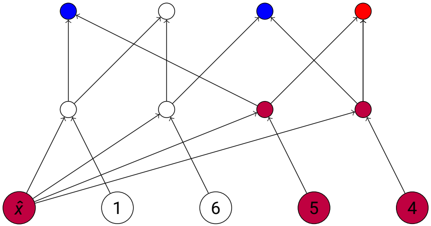


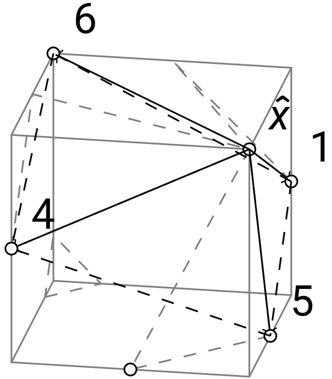
Small Example



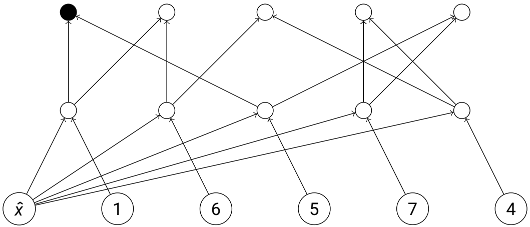


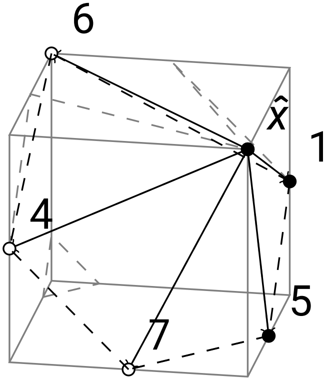
Small Example



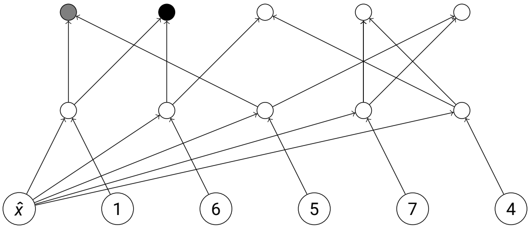


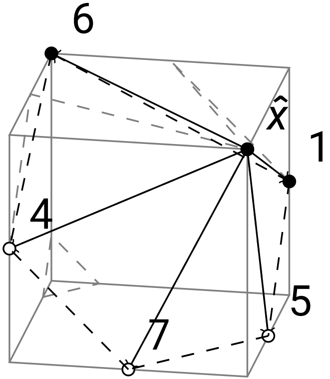
Small Example



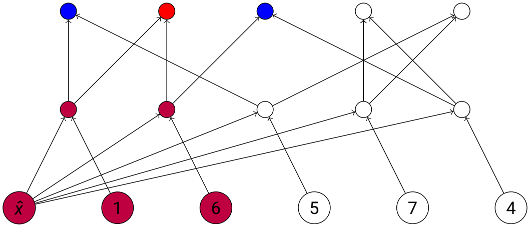


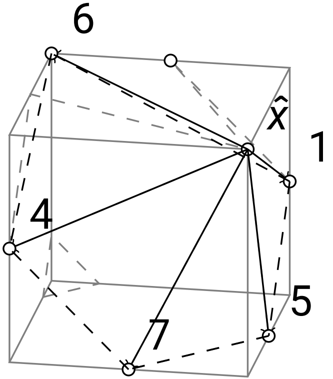
Small Example



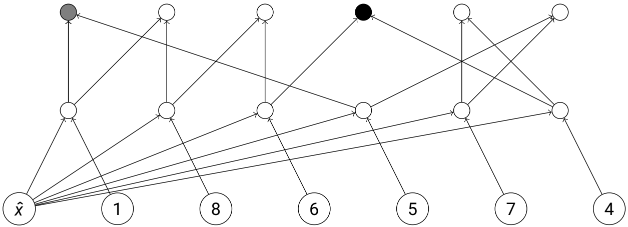


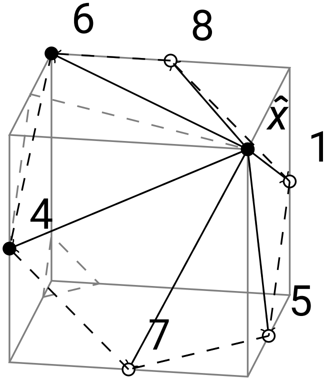
Small Example



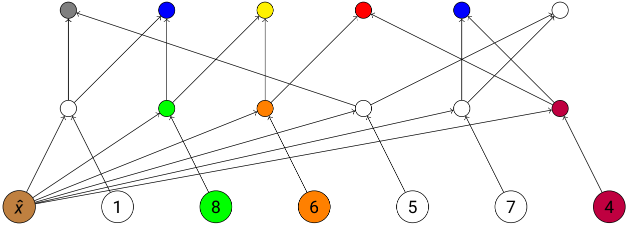


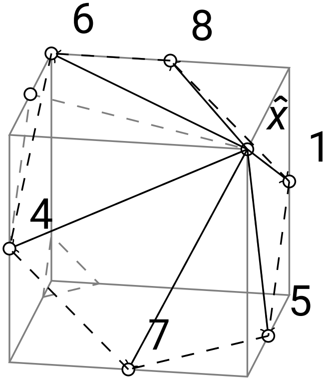
Small Example



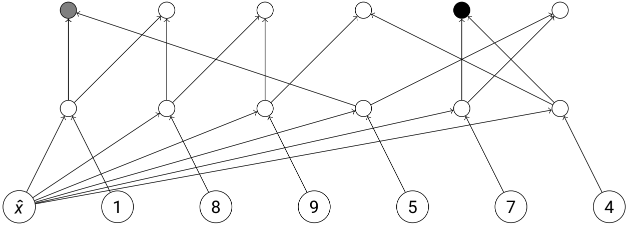


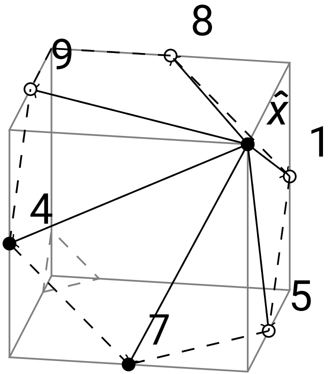
Small Example



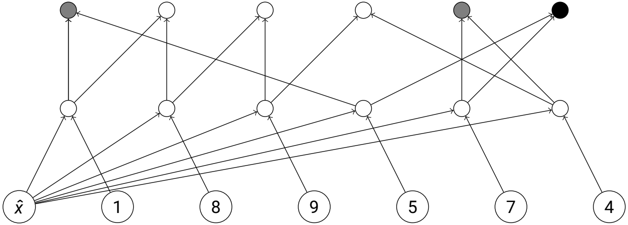


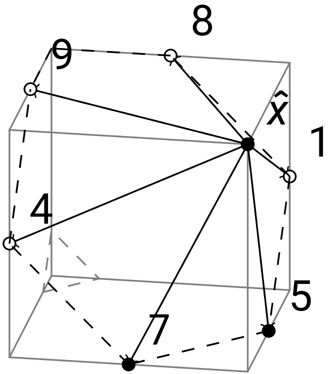
Small Example



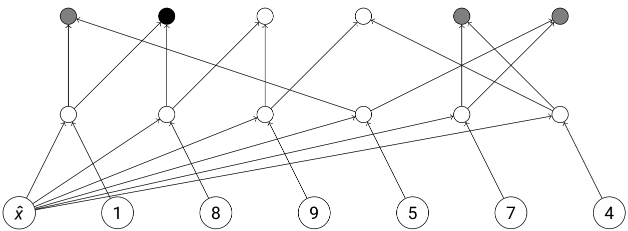


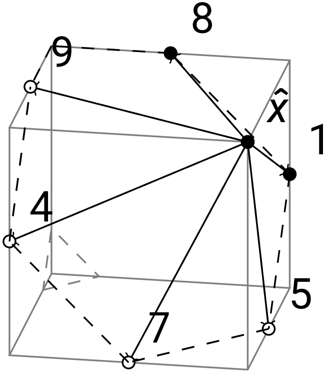
Small Example



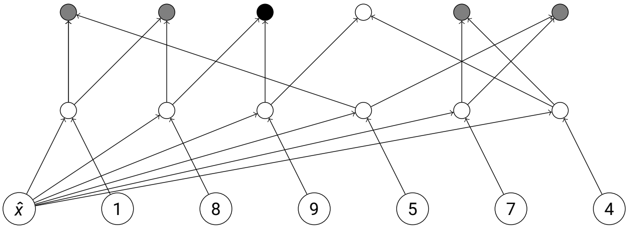


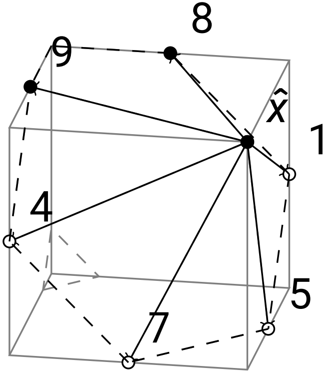
Small Example



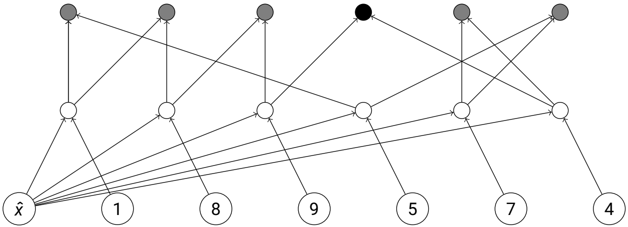


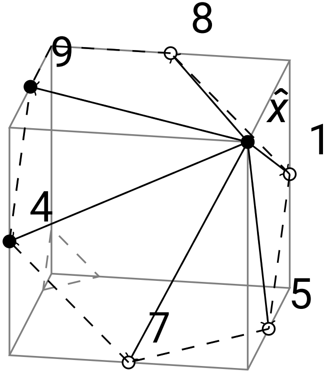
Small Example



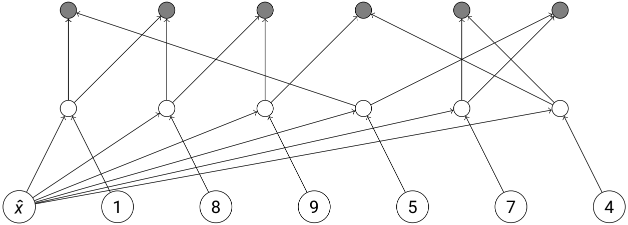


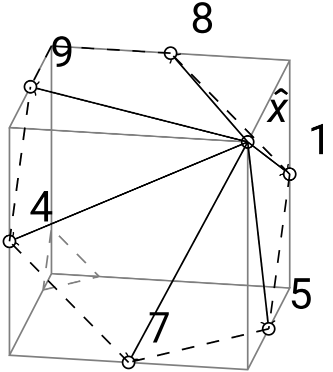
Small Example





Small Example





Symmetry Handling in the Presence of Custom Constraints

Christopher Hojny

Aussois COW, January 6, 2025

supported by NWO grant OCENW.M.21.299

Motivation

Symmetry handling is important

- many optimization problems contain symmetries
- disabling symmetry handling makes, e.g., SCIP 8 by 16% slower on MIPLIB 2017

Motivation

Symmetry handling is important

- many optimization problems contain symmetries
- disabling symmetry handling makes, e.g., SCIP 8 by 16% slower on MIPLIB 2017

But

- some problems contain lazy constraints
- solvers cannot detect symmetries automatically

Traveling Salesperson Problem

(for undirected weighted graph $G = (V, E, w)$)

$$\min \sum_{e \in E} w_e x_e$$

$$\sum_{e \in \delta(v)} x_e = 2 \qquad e \in E,$$

$$\sum_{e \in \delta(S)} x_e \geq 2 \qquad \emptyset \subsetneq S \subsetneq V,$$

$$x \in \{0, 1\}^E$$

Main Question

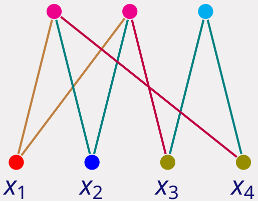
Question

Can we inform a solver about symmetries of lazy or custom constraints to benefit from powerful build-in symmetry handling methods?

Symmetry Detection

- solvers detect symmetries by building auxiliary graph
- for each inequality, define a graph whose automorphisms correspond to symmetries
- combine these graphs for all inequalities

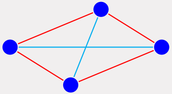
$$\begin{array}{lrcrcrcrcl}
\min & x_1 & - & x_2 & + & 2x_3 & + & 2x_4 & & \\
 & & & & & x_3 & + & x_4 & \leq & 1 \\
 & -x_1 & + & x_2 & + & 3x_3 & & & \leq & 4 \\
 & -x_1 & + & x_2 & & & + & 3x_4 & \leq & 4
\end{array}$$



Symmetry Detection for Lazy Constraints

- same idea works for lazy constraint
- define auxiliary graph for **entire family** of lazy constraints

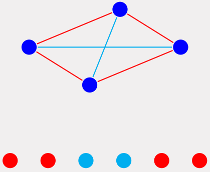
Example: TSP



Symmetry Detection for Lazy Constraints

Example: TSP

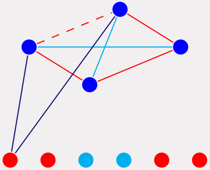
- same idea works for lazy constraint
- define auxiliary graph for **entire family** of lazy constraints



Symmetry Detection for Lazy Constraints

Example: TSP

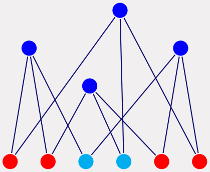
- same idea works for lazy constraint
- define auxiliary graph for **entire family** of lazy constraints



Symmetry Detection for Lazy Constraints

Example: TSP

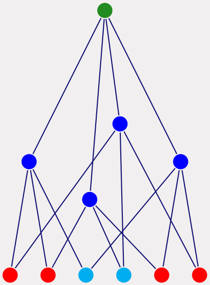
- same idea works for lazy constraint
- define auxiliary graph for **entire family** of lazy constraints



Symmetry Detection for Lazy Constraints

- same idea works for lazy constraint
- define auxiliary graph for **entire family** of lazy constraints

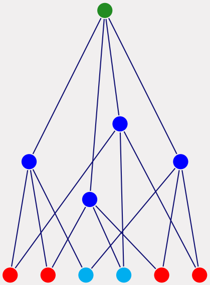
Example: TSP



Symmetry Detection for Lazy Constraints

- same idea works for lazy constraint
- define auxiliary graph for **entire family** of lazy constraints
- We have implemented this idea in SCIP 9:
  - symmetry information can be added via callback
  - SCIP extends its internal symmetry detection graph by user information

Example: TSP



Usage of Callback

```c
static
SCIP_DECL_CONSGETPERMSYMGRAPH(consGetPermsymGraphTSP)
{
   SCIP_CONSDATA* consdata;
   int* idx;
   int vidx, nnodes, v;

   consdata = SCIPconsGetData(cons);
   nnodes = consdata->nnodes;
   SCIP_CALL( SCIPallocBufferArray(scip, &idx, nnodes + 1) );

   for( v = 0; v < nnodes; ++v ){
      SCIP_CALL( SCIPaddSymgraphOpnode(scip, graph, 0, &idx[v]) );
   }
   SCIP_CALL( SCIPaddSymgraphConsnode(scip, graph, cons, 0.0, 0.0, &idx[nnodes]) );

   for( v = 0; v < consdata->nedges; ++v ){
      vidx = SCIPgetSymgraphVarnodeidx(scip, graph, consdata->vars[v]);
      SCIP_CALL( SCIPaddSymgraphEdge(scip, graph, idx[consdata->first[v]], vid, FALSE, 0.0) );
      SCIP_CALL( SCIPaddSymgraphEdge(scip, graph, idx[consdata->second[v]], vid, FALSE, 0.0) );
   }

   for( v = 0; v < nnodes; ++v ){
      SCIP_CALL( SCIPaddSymgraphEdge(scip, graph, idx[v], idx[nnodes], FALSE, 0.0) );
   }
   *success = TRUE;
   SCIPfreeBufferArray(scip, &idx);

   return SCIP_OKAY;
}
```

Summary

- symmetry detection in presence of lazy or custom constraints is possible
- framework also allows to detect reflection symmetries
- check the preprint for more information on
  - theory behind symmetry detection graphs
  - rules for building these graphs
  - specialized graph for MINLP

preprint

MOTIVATION: Find efficient separation algorithms for rank-1 Chvátal-Gomory cuts derived from Knapsack sets

Given $\bar{\mathbf{x}} \in P := \{\mathbf{x} \in \mathbb{R}^n_+ \mid \underbrace{\mathbf{a}^T\mathbf{x} \le b}_{0<u_0<1},\ \underbrace{x_i \le 1, i = 1, \dots, n}_{0 \le u_i < 1}\}$ with $b, a_i \in \mathbb{Z}_+$

Rank-1 Chvátal-Gomory cut: $$z(\mathbf{u}) = \sum_{i \in I} \lfloor u_0 a_i + u_i \rfloor \bar{x}_i - \lfloor u_0 b + \sum_{i \in I} u_i \rfloor > 0 \qquad (1)$$

Lemma 1 (*Selecting best multipliers u_i given u_0*)

Given $\bar{\mathbf{u}} \in \mathbb{R}^{n+1}_+$ with $z(\bar{\mathbf{u}}) > 0$, define $J(\bar{\mathbf{u}}) = \{i \in I \mid \lfloor \bar{u}_0 a_i + \bar{u}_i \rfloor = \lfloor \bar{u}_0 a_i \rfloor + 1\}$. Then, we have that $z(\mathbf{u}) \ge z(\bar{\mathbf{u}})$ for any $\mathbf{u} \in \mathbb{R}^{n+1}_+$ such that $u_0 = \bar{u}_0$ and

$$u_i = \begin{cases} 1 - (u_0 a_i - \lfloor u_0 a_i \rfloor) & \text{if } i \in J(\bar{\mathbf{u}}) \\ 0 & \text{if } i \in I \setminus J(\bar{\mathbf{u}}). \end{cases} \qquad (2)$$

Theorem 2 (*Discretising multipliers u_0*)

Given $U_0 := \left\{ \frac{p}{a_i} \mid i = 1, \dots, n,\ p = 1, \dots, a_i - 1 \right\}$, to get $\mathbf{u} \in \mathbb{R}^{n+1}_+$ maximizing $z(\mathbf{u})$, for each $u_0 \in U_0$ we look for $J(u_0) \subset I$ such that

$$z(u_0) = \sum_{i \in I} \lfloor u_0 a_i \rfloor \bar{x}_i + \sum_{j \in J(u_0)} \bar{x}_j - \lfloor u_0 b + \sum_{j \in J(u_0)} (1 - (u_0 a_j - \lfloor u_0 a_j \rfloor)) \rfloor > 0.$$

To find $J(u_0)$ we need to solve a sequence of n Knapsack Problems.

- The exact separation of a rank-1 CG-cut has complexity $O(bn^2 KP)$. If we use dynamic programming for KP, we get $O(b^2 n^3)$.
- For a fractional KP heuristic, the complexity is $O(bn^2)$.

Results on the Generalized Assignment Problem (GAP), SCIP heuristic lifted-cover [Letchford2019], GUROBI cover cuts, and our CG-cut. Gap closed: $\left(1 - \frac{LP_0 - LP_{CG}}{Opt - LP_{CG}}\right) \cdot 100$, runtime in seconds.

	SCIP		**Gurobi**		**Exact KP**		**Fractional KP**	
instance	gap cl.	time	gap cl.	time	gap cl.	time	gap cl.	time
d05100	15.59%	0.2	11.13%	0	56.69%	0	55.52%	0
d05200	13.39%	0.3	13.59%	0	54.20%	1	54.03%	0
d10100	22.59%	0.4	5.93%	0	63.69%	1	63.41%	0
d10200	15.90%	0.5	11.72%	0	51.30%	1	50.46%	0
d20100	26.39%	0.9	2.72%	0	66.79%	2	66.60%	1
d20200	30.93%	1.5	2.00%	0	70.56%	3	69.86%	1
d20400	29.12%	2.6	5.89%	1	70.27%	4	69.90%	2
d201600	26.77%	10.8	39.39%	2	74.61%	8	73.58%	5

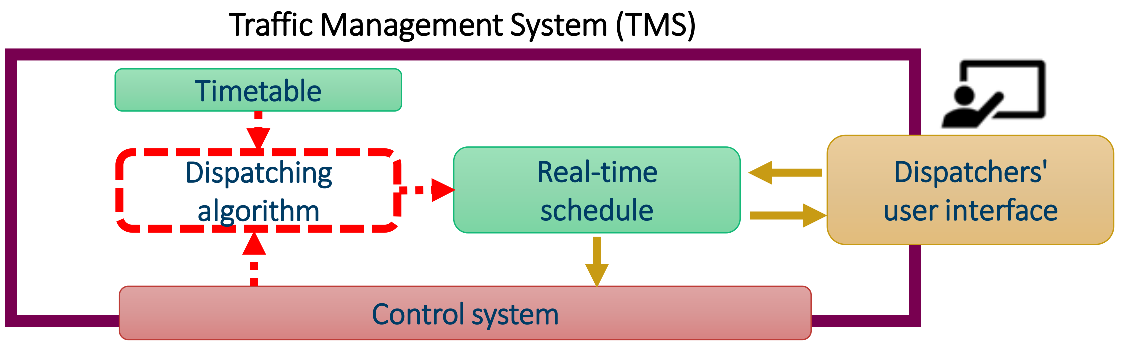
Preprint: Giacomo Maggiorano, Stefano Gualandi, Pasquale Avella, and Michele Mele.
Rank-1 Chvátal-Gomory cuts from Knapsack sets: A computational study, 2024.

The DISPLIB 2025 computational competition

Giorgio Sartor, Oddvar Kloster,
Bjørnar Luteberget and Carlo Mannino

Teknologi for et bedre samfunn

Motivation



Traffic Management System (TMS)

Timetable

Dispatching algorithm

Real-time schedule

Dispatchers' user interface

Control system

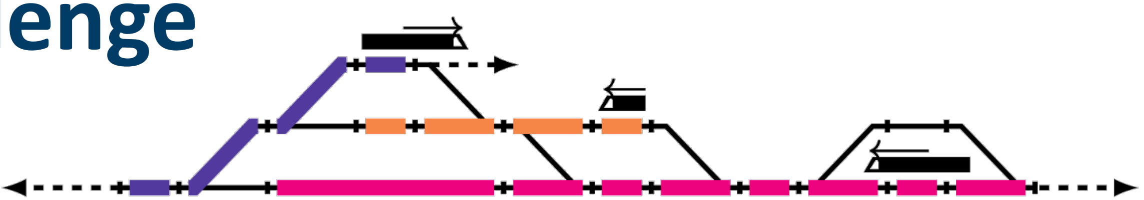
- «train dispatching» in Google Scholar returns 17600 results from 2020…
- …and probably each one of them uses a different set of instances! **WHY??** 😭 (crying of frustration)
- Many countries still consider the sharing of railways data as a violation of national security
  - But they publish a public timetable (not in machine-readable format) 🤦
  - And (in Europe) they are required to publish a network statement
- Lack of a standard format
- Existing formats (e.g., RailML) are way too detailed and complex for non-experts
- Other research communities have gained a lot from standardized, comprehensive benchmark libraries
  - Vehicle Routing (TSP, CVRP, …)
  - SATLIB, MIPLIB, MaxSAT Evalutations, …

The DISPLIB 2025 Competition: spirit and rules

- The competition challenges the research community to find innovative and effective algorithms for solving a diverse set of real-life train dispatching instances
- The instances come from different countries and have different characteristics: some have many routing options and few trains while others have few routing options and many trains.
  - (thanks to SINTEF Digital, Siemens Mobility, data.sbb.ch for confirmed sets of instances so far…)
  - (three new data sources under way, pending data release, **more are welcome!**)

- General rules:
  - The usage of commercial MIP solver is allowed
  - The usage of ML pre-training is allowed, and the learning phase does not count against the time limit
  - The time limit to solve each instance is 10 minutes, maximum 8 CPUs and 32GB of RAM. Teams using GPUs are limited to 1 GPU unit and 24 GB of GPU memory.
  - The source code does not need to be submitted, but the winners may be required to show additional proof of compliance to the rules above

The DISPLIB 2025 Competition: a train dispatching challenge



- DISPLIB: a new train dispatching benchmark library
  - Wide range of real-life instances from all over the world
  - Simple but powerful problem definition

- The DISPLIB 2025 Competition
  - Schedule and route trains from a wide range of real-world use cases
  - No deep knowledge of railways needed to start
  - Winners will be invited to a special session at ODS 2025
  - … and get an expedited review process in the Journal of Rail Transport Planning & Management (JRTPM)
  - FINAL SUBMISSION: End of April 2025

Get started now!!

displib.github.io

SINTEF

Technology for a better society

Jannik Trappe

Volker Kaibel

Cyclic Transversal Polytopes and Parity-Based Facets of Well-Known Polytopes

Aussois

OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$\mathcal{B}(1)$ $\mathcal{B}(2)$ $\mathcal{B}(3)$ $\mathcal{B}(4)$

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$\mathcal{B}(1) \subseteq \mathbb{F}_2^d$ $\mathcal{B}(2) \subseteq \mathbb{F}_2^d$ $\mathcal{B}(3) \subseteq \mathbb{F}_2^d$ $\mathcal{B}(4) \subseteq \mathbb{F}_2^d$

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$$\begin{array}{cccc}
\mathcal{B}(1) & \mathcal{B}(2) & \mathcal{B}(3) & \mathcal{B}(4) \\
\begin{matrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} &
\begin{matrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{matrix} &
\begin{matrix} 0 & 1 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{matrix} &
\begin{matrix} 1 \\ 0 \\ 0 \\ 0 \end{matrix}
\end{array}$$

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$$\mathcal{B}(1) \qquad \mathcal{B}(2) \qquad \mathcal{B}(3) \qquad \mathcal{B}(4)$$

$$\begin{array}{ccc} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array} \qquad \begin{array}{cc} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{array} \qquad \begin{array}{cc} 0 & 1 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{array} \qquad \begin{array}{c} 1 \\ 0 \\ 0 \\ 0 \end{array}$$

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$$\overset{\mathcal{B}(1)}{\begin{matrix}0 & 0 & 1\\ 0 & 1 & 1\\ 0 & 1 & 0\\ 0 & 0 & 1\end{matrix}} + \overset{\mathcal{B}(2)}{\begin{matrix}1 & 1\\ 0 & 1\\ 0 & 1\\ 0 & 1\end{matrix}} + \overset{\mathcal{B}(3)}{\begin{matrix}0 & 1\\ 0 & 0\\ 0 & 1\\ 0 & 0\end{matrix}} + \overset{\mathcal{B}(4)}{\begin{matrix}1\\ 0\\ 0\\ 0\end{matrix}} = \begin{matrix}0\\ 0\\ 0\\ 0\end{matrix}$$

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$$\overset{\mathcal{B}(1)}{\begin{matrix}0&0&1\\0&1&1\\0&1&0\\0&0&1\end{matrix}} + \overset{\mathcal{B}(2)}{\begin{matrix}1&1\\0&1\\0&1\\0&1\end{matrix}} + \overset{\mathcal{B}(3)}{\begin{matrix}0&1\\0&0\\0&1\\0&0\end{matrix}} + \overset{\mathcal{B}(4)}{\begin{matrix}1\\0\\0\\0\end{matrix}} = \begin{matrix}0\\0\\0\\0\end{matrix}$$

Special case: parity polytope

$$\begin{matrix}0&1\end{matrix} \qquad \begin{matrix}0&1\end{matrix} \qquad \begin{matrix}0&1\end{matrix} \qquad \begin{matrix}0&1\end{matrix}$$

Cyclic Transversal Polytopes (Frede, Merkert, Kaibel [2023])

$$\begin{array}{c} \mathcal{B}(1) \\ \begin{matrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} \end{array} + \begin{array}{c} \mathcal{B}(2) \\ \begin{matrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{matrix} \end{array} + \begin{array}{c} \mathcal{B}(3) \\ \begin{matrix} 0 & 1 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{matrix} \end{array} + \begin{array}{c} \mathcal{B}(4) \\ \begin{matrix} 1 \\ 0 \\ 0 \\ 0 \end{matrix} \end{array} = \begin{matrix} 0 \\ 0 \\ 0 \\ 0 \end{matrix}$$

Special case: parity polytope

$$0 \quad 1 \qquad\qquad 0 \quad 1 \qquad\qquad 0 \quad 1 \qquad\qquad 0 \quad 1$$

Jeroslow’s odd set inequalities $\rightsquigarrow$ Lifted odd set inequalities

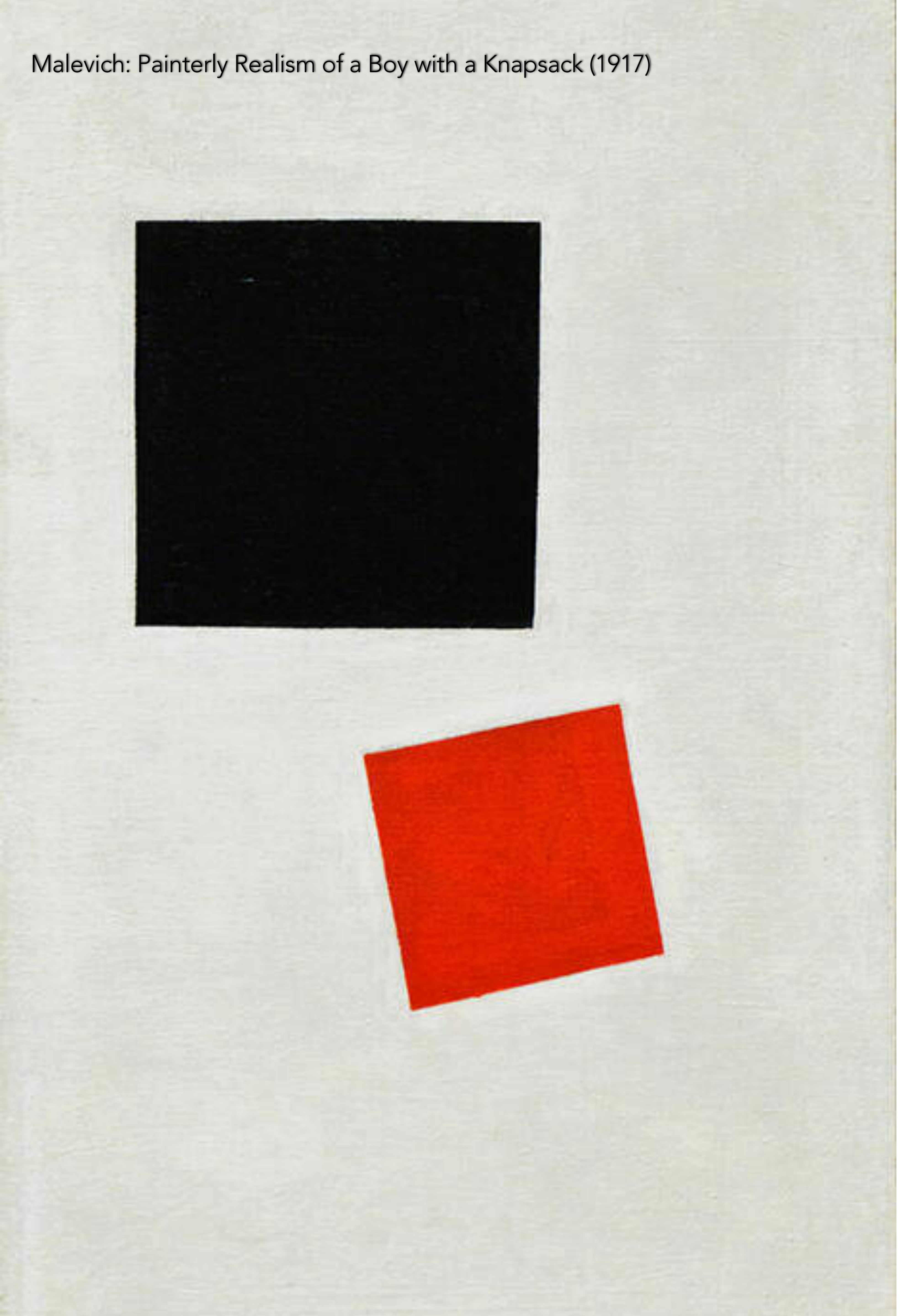
Cyclic Transversal Polytopes

Cyclic Transversal Polytopes

- Matching polytopes
- Stable set polytopes
- Cut polytopes
- ...

Lifted Odd Set Inequalities

- Edmond’s blossom inequalities
- Odd hole inequalities
- Cycle inequalities
- ...


Malevich: Painterly Realism of a Boy with a Knapsack (1917)

Generalized assignment and knapsack problems in the random order model

Max Klimm

joint work with Martin Knaack

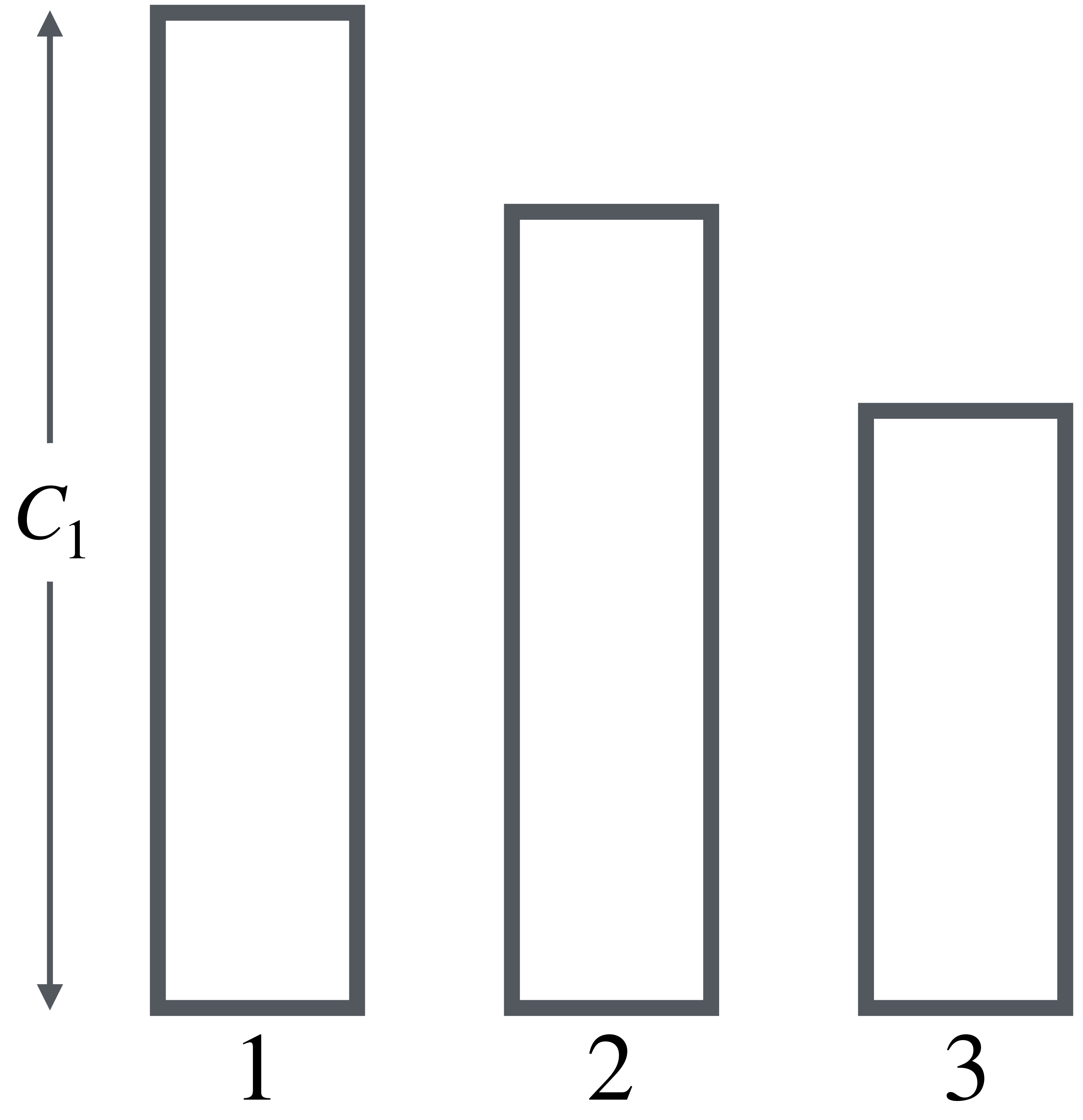
Berlin Mathematics Research Center MATH+
Technische Universität Berlin
Discrete Optimization

General assignment problem

General assignment problem

m bins

- capacity C_i



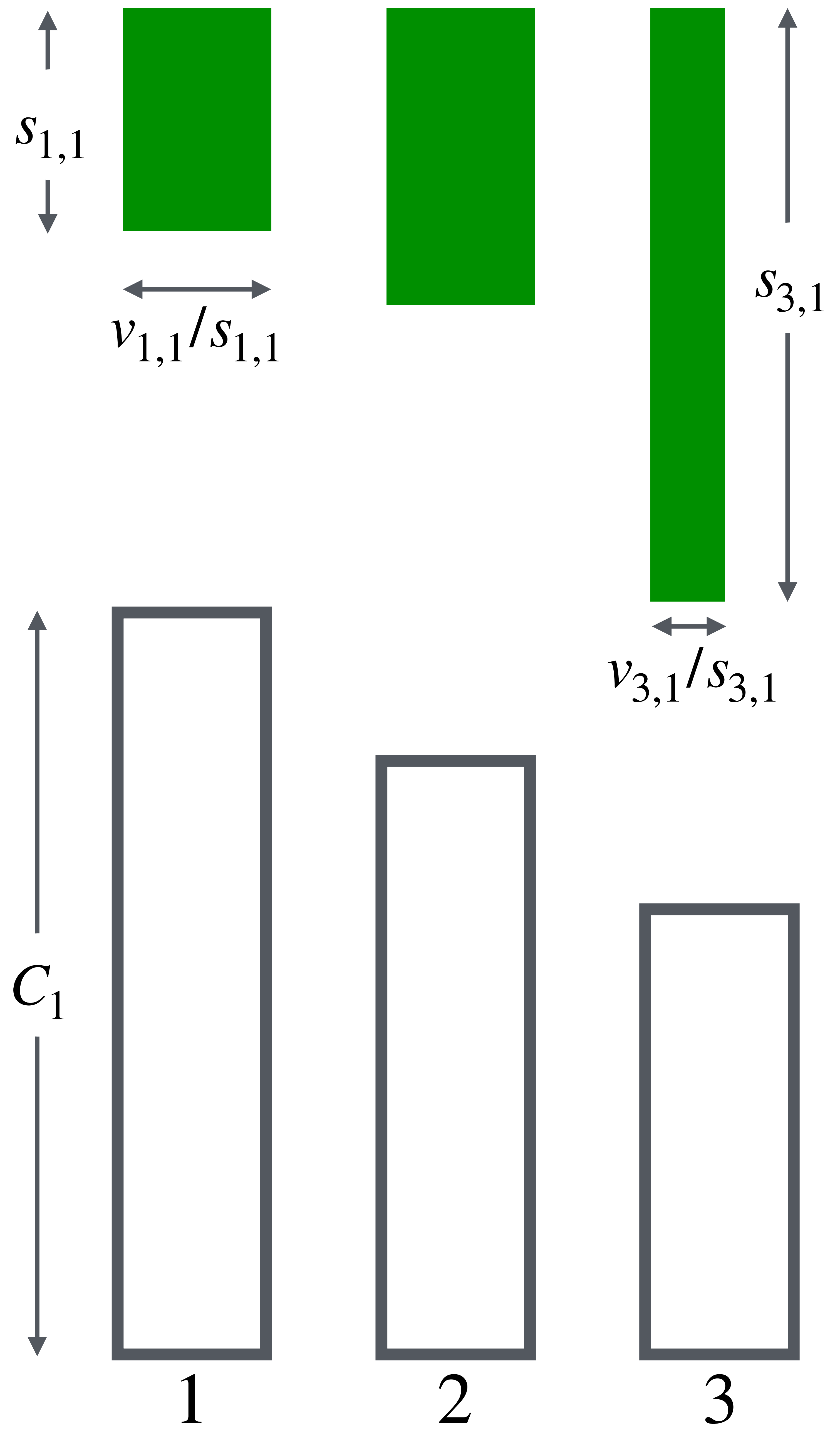
General assignment problem

m bins

- capacity C_i

n items

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i



General assignment problem

m bins

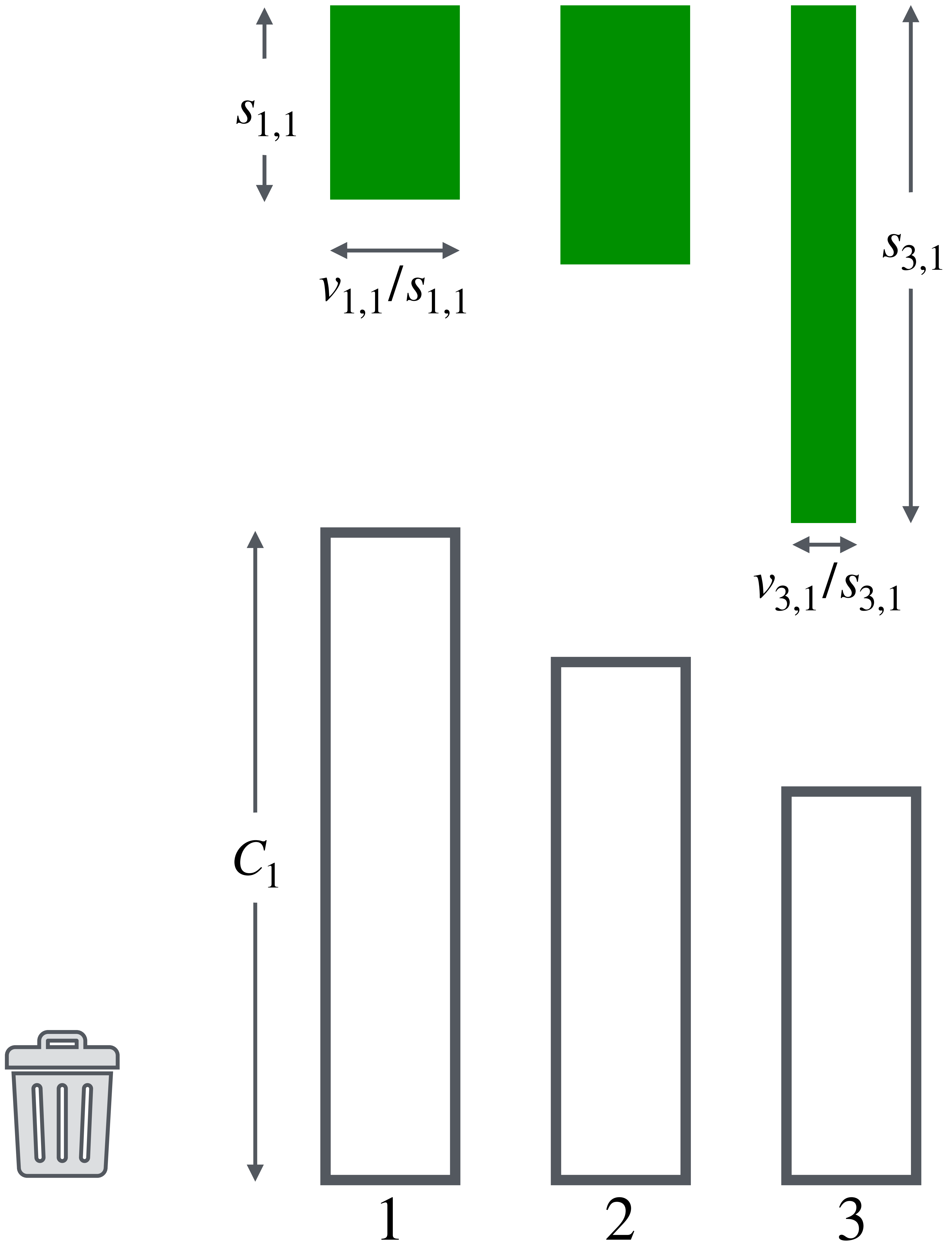
- capacity C_i

n items

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable



General assignment problem

m bins

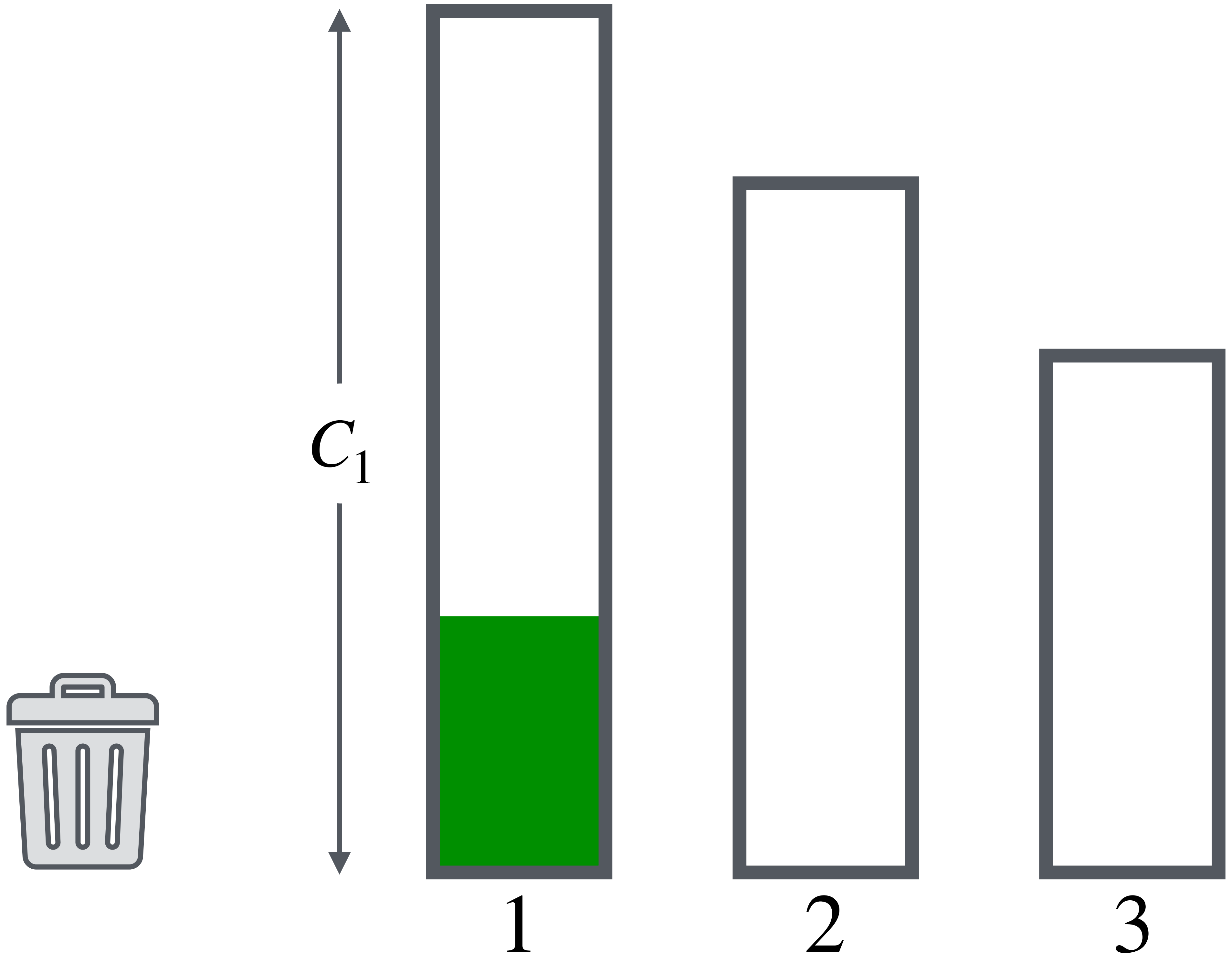
- capacity C_i

n items

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable



General assignment problem

m bins

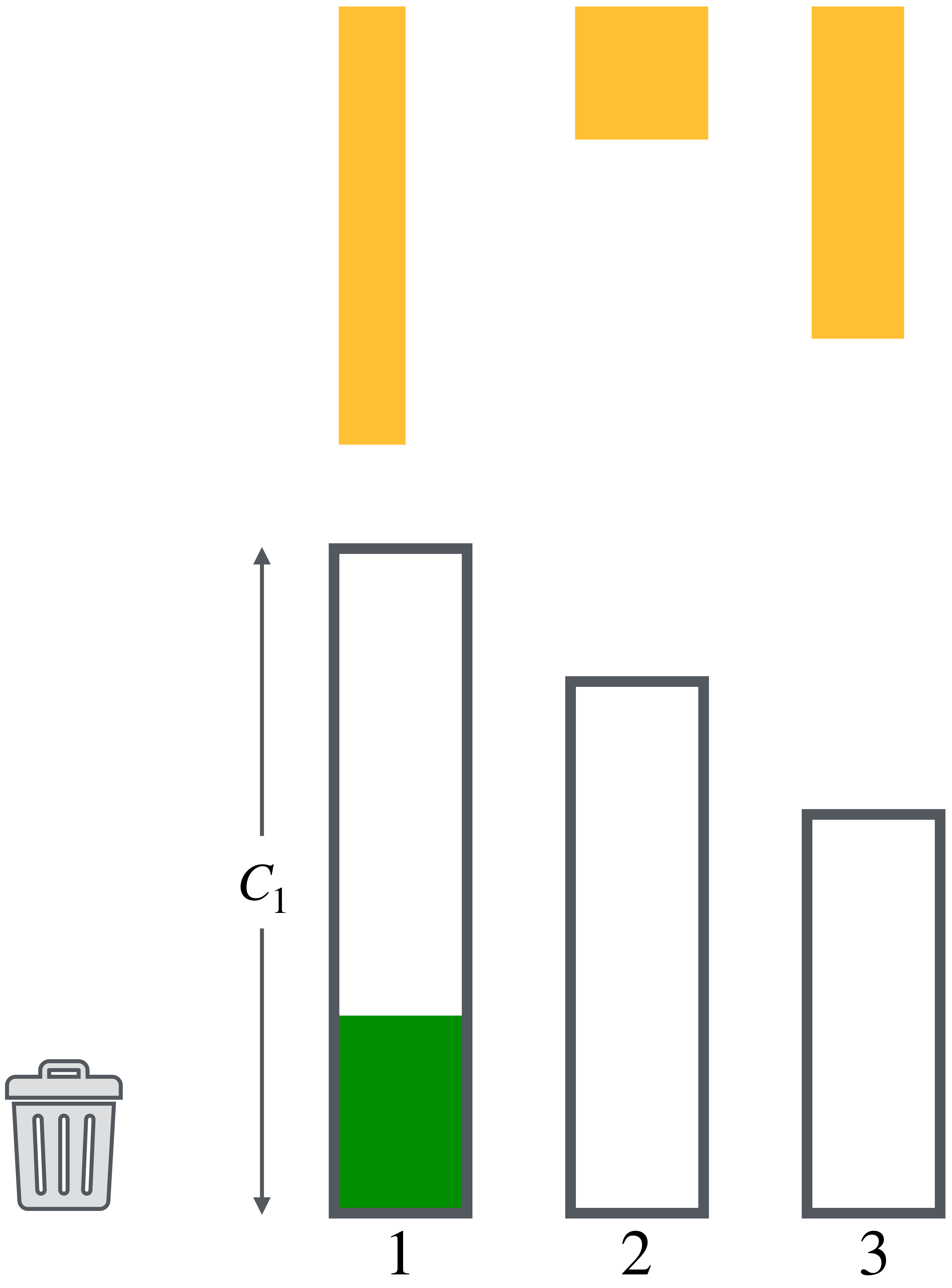
- capacity C_i

n items

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable



General assignment problem

m bins

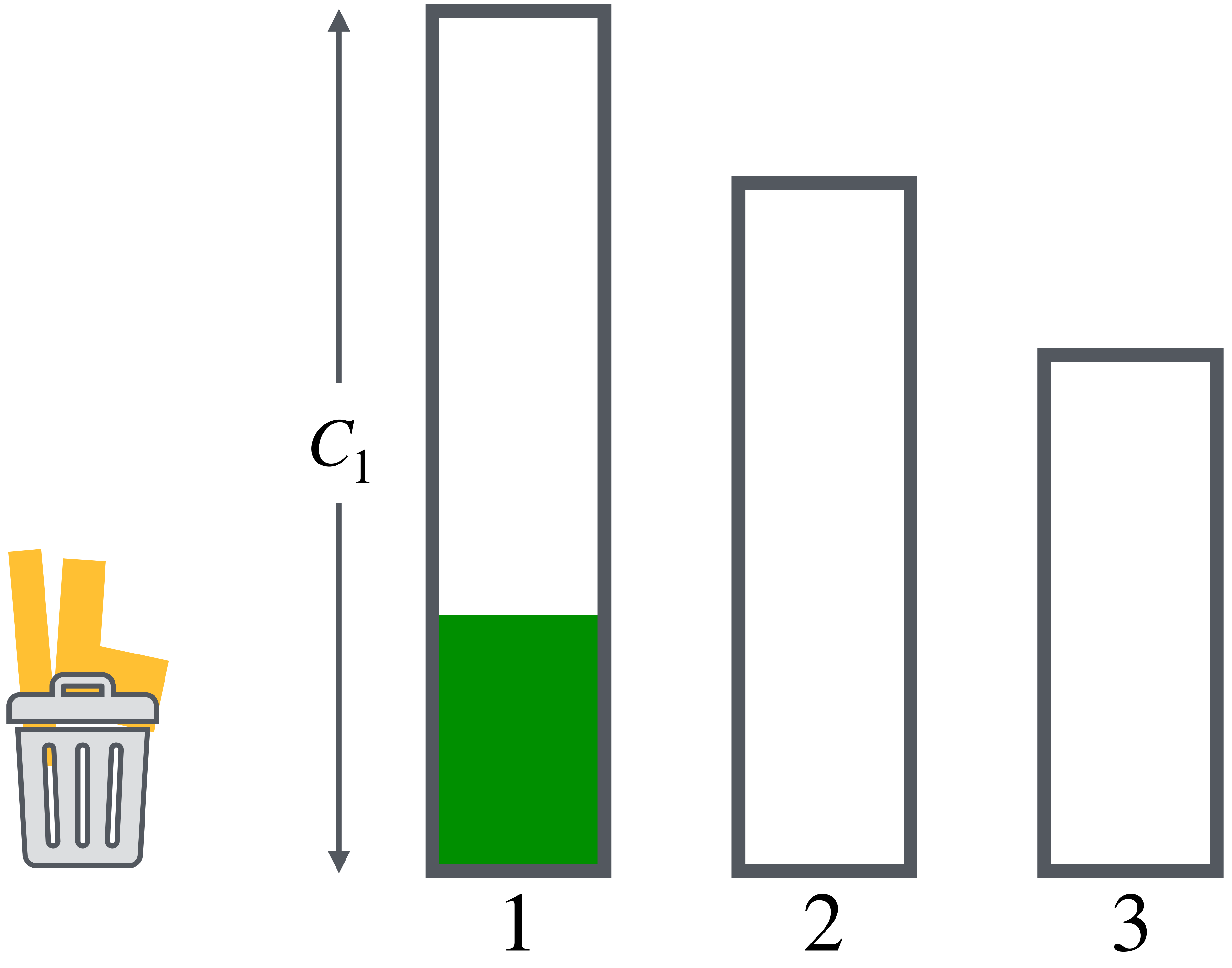
- capacity C_i

n items

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable



General assignment problem

m **bins**

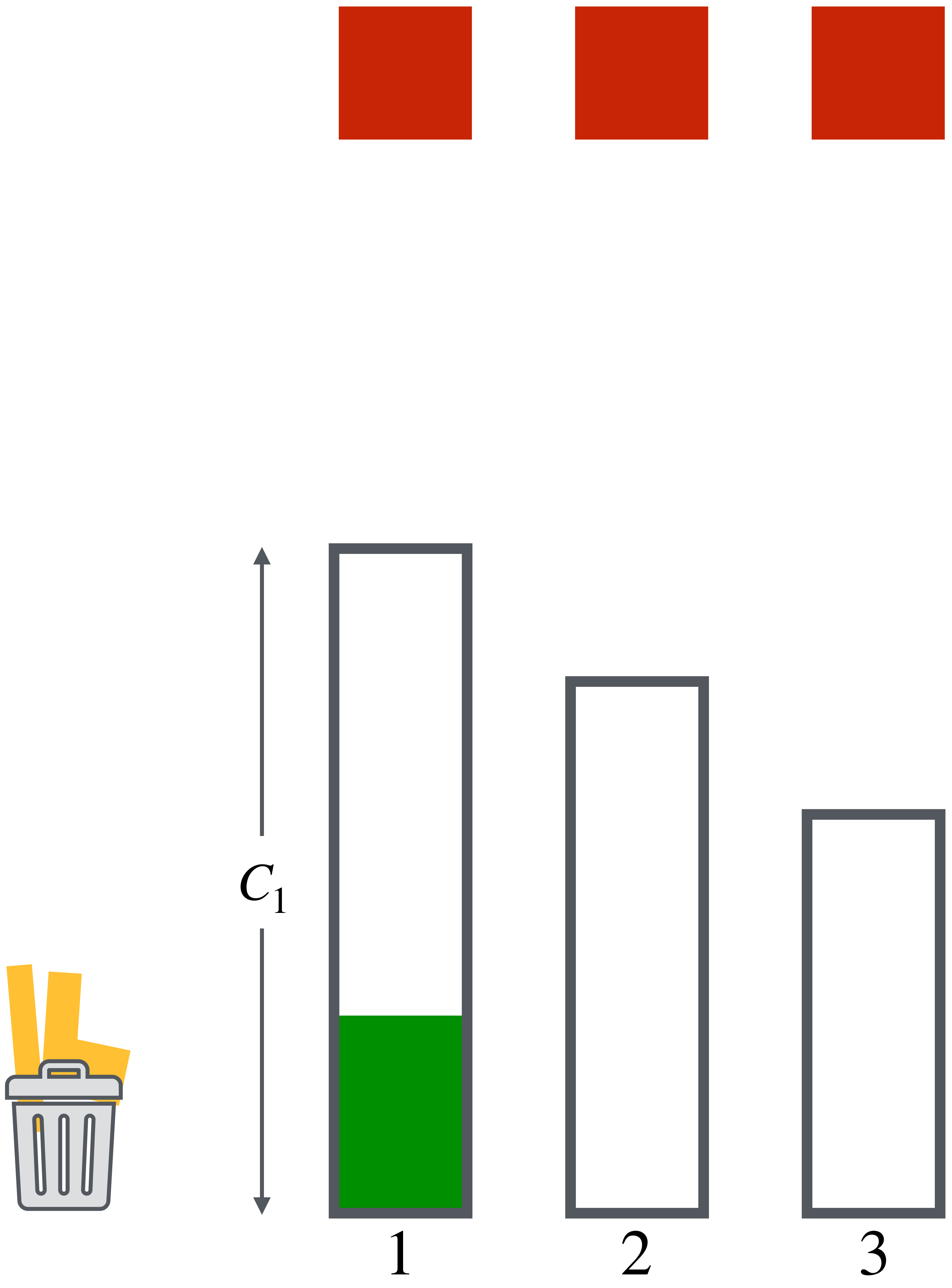
- capacity C_i

n **items**

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable



General assignment problem

m bins

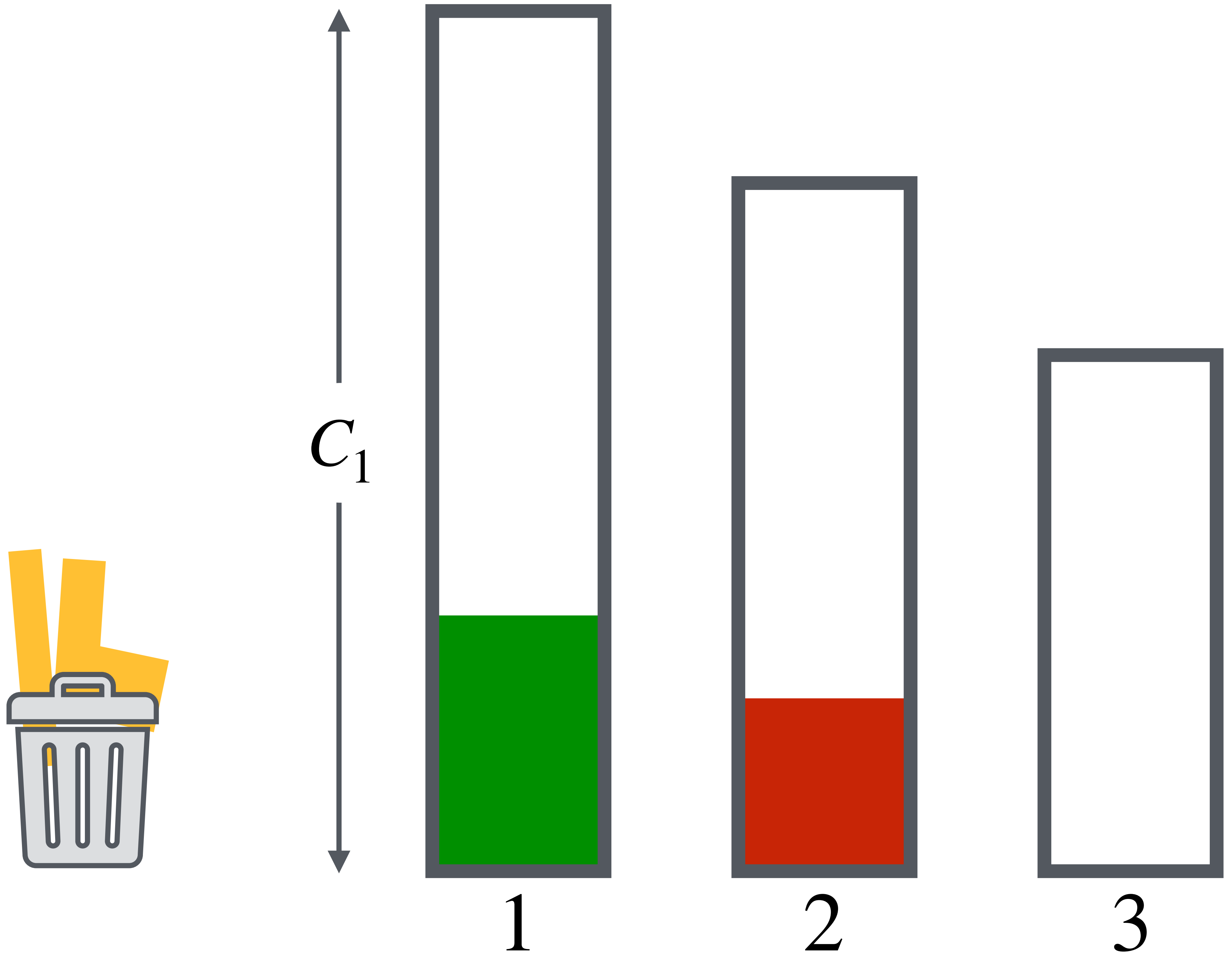
- capacity C_i

n items

- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable



General assignment problem

m bins

- capacity C_i

n items

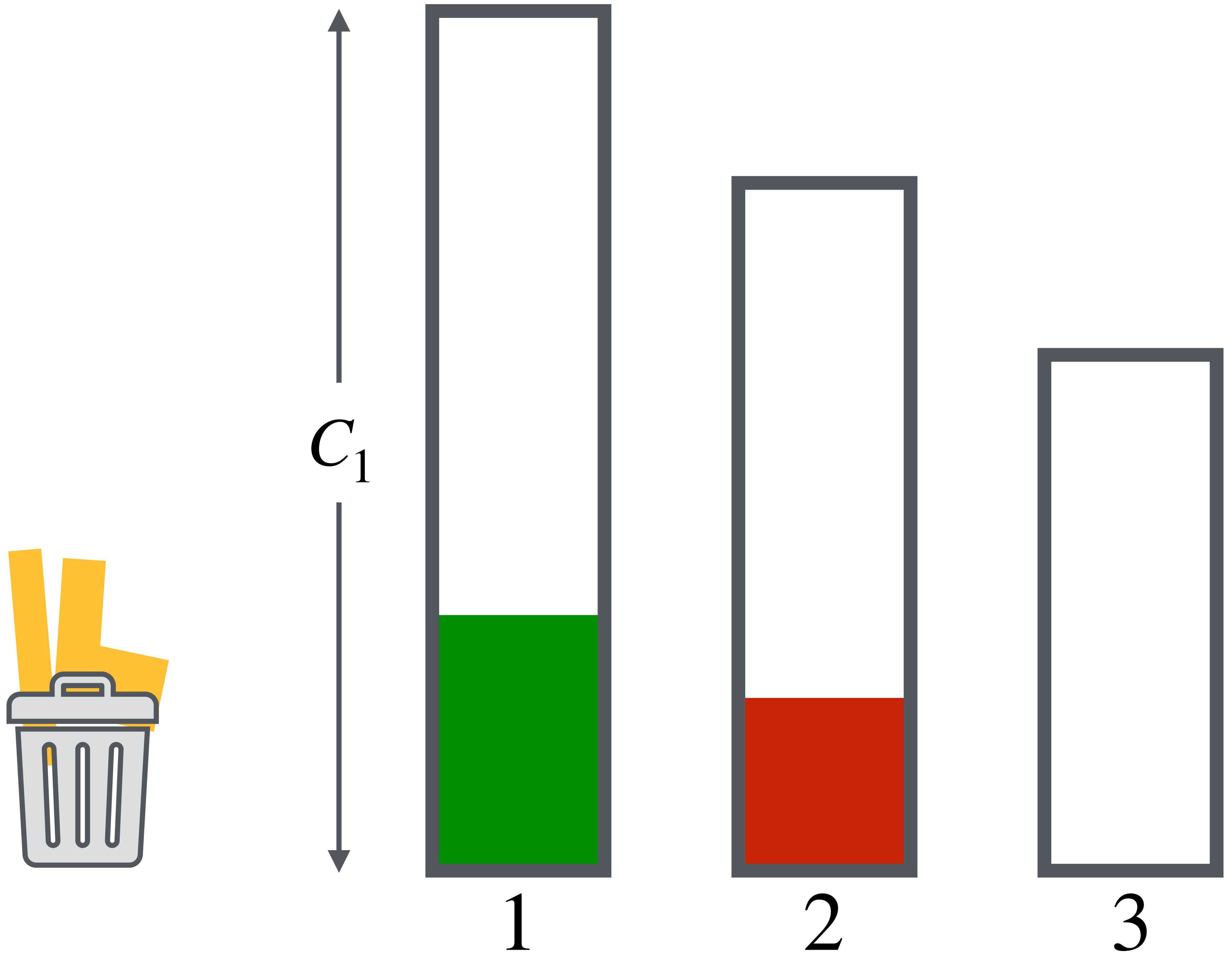
- value $v_{i,j}$ when packed in bin i
- size $s_{i,j}$ when packed in bin i

items arrive in random order

- packing decision immediate and irrevocable

algorithm knows only n

- goal is to maximize expected value of items packed
- competitive ratio: $\dfrac{\text{expected value of algorithm}}{\text{optimal value}}$



Related work and our results

Related work and our results

known competitive ratios

Related work and our results

known competitive ratios

secretary problem
($m = 1$, $C = 1$, $s_j = 1$)

Related work and our results

known competitive ratios

secretary problem
$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

Related work and our results

known competitive ratios

secretary problem

$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

knapsack problem

$(m = 1)$

Related work and our results

known competitive ratios

secretary problem $(m = 1, C = 1, s_j = 1)$	tight $1/e$ [Dynkin 1963; Lindley1961]
knapsack problem $(m = 1)$	$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007] $1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018] $1/6.65$ [Albers, Khan, Ladewig 2021]

Related work and our results

known competitive ratios

secretary problem

$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

knapsack problem

$(m = 1)$

$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007]

$1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018]

$1/6.65$ [Albers, Khan, Ladewig 2021]

general assignment problem

Related work and our results

known competitive ratios

secretary problem $(m = 1, C = 1, s_j = 1)$	tight $1/e$ [Dynkin 1963; Lindley1961]
knapsack problem $(m = 1)$	$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007] $1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018] $1/6.65$ [Albers, Khan, Ladewig 2021]
general assignment problem	$1/8.10$ [Kesselheim, Radke, Tönnis, Vöcking 2018] $1/6.99$ [Naori, Raz 2019; Albers, Khan, Ladewig 2021]

Related work and our results

known competitive ratios

our results

secretary problem

$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

knapsack problem

$(m = 1)$

$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007]

$1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018]

$1/6.65$ [Albers, Khan, Ladewig 2021]

general assignment problem

$1/8.10$ [Kesselheim, Radke, Tönnis, Vöcking 2018]

$1/6.99$ [Naori, Raz 2019; Albers, Khan, Ladewig 2021]

$$\left.\right\} \quad \frac{1-\ln(2)}{2} = \frac{1}{6.52}$$

Related work and our results

known competitive ratios

our results

secretary problem

$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

knapsack problem

$(m = 1)$

$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007]

$1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018]

$1/6.65$ [Albers, Khan, Ladewig 2021]

general assignment problem

$1/8.10$ [Kesselheim, Radke, Tönnis, Vöcking 2018]

$1/6.99$ [Naori, Raz 2019; Albers, Khan, Ladewig 2021]

$$\frac{1 - \ln(2)}{2} = \frac{1}{6.52}$$

fractional knapsack problem

Related work and our results

known competitive ratios

our results

secretary problem
$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

knapsack problem
$(m = 1)$

$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007]
$1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018]
$1/6.65$ [Albers, Khan, Ladewig 2021]

general assignment problem

$1/8.10$ [Kesselheim, Radke, Tönnis, Vöcking 2018]
$1/6.99$ [Naori, Raz 2019; Albers, Khan, Ladewig 2021]

$$\frac{1-\ln(2)}{2} = \frac{1}{6.52}$$

fractional knapsack problem

$1/9.37$ [Karrenbauer, Kovalevskaya 2020]
$1/4.39$ [Gilberti, Karrenbauer 2021]

Related work and our results

known competitive ratios

our results

secretary problem
$(m = 1, C = 1, s_j = 1)$

tight $1/e$ [Dynkin 1963; Lindley1961]

knapsack problem
$(m = 1)$

$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007]
$1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018]
$1/6.65$ [Albers, Khan, Ladewig 2021]

general assignment problem

$1/8.10$ [Kesselheim, Radke, Tönnis, Vöcking 2018]
$1/6.99$ [Naori, Raz 2019; Albers, Khan, Ladewig 2021]

$$\left.\right\} \frac{1 - \ln(2)}{2} = \frac{1}{6.52}$$

fractional knapsack problem

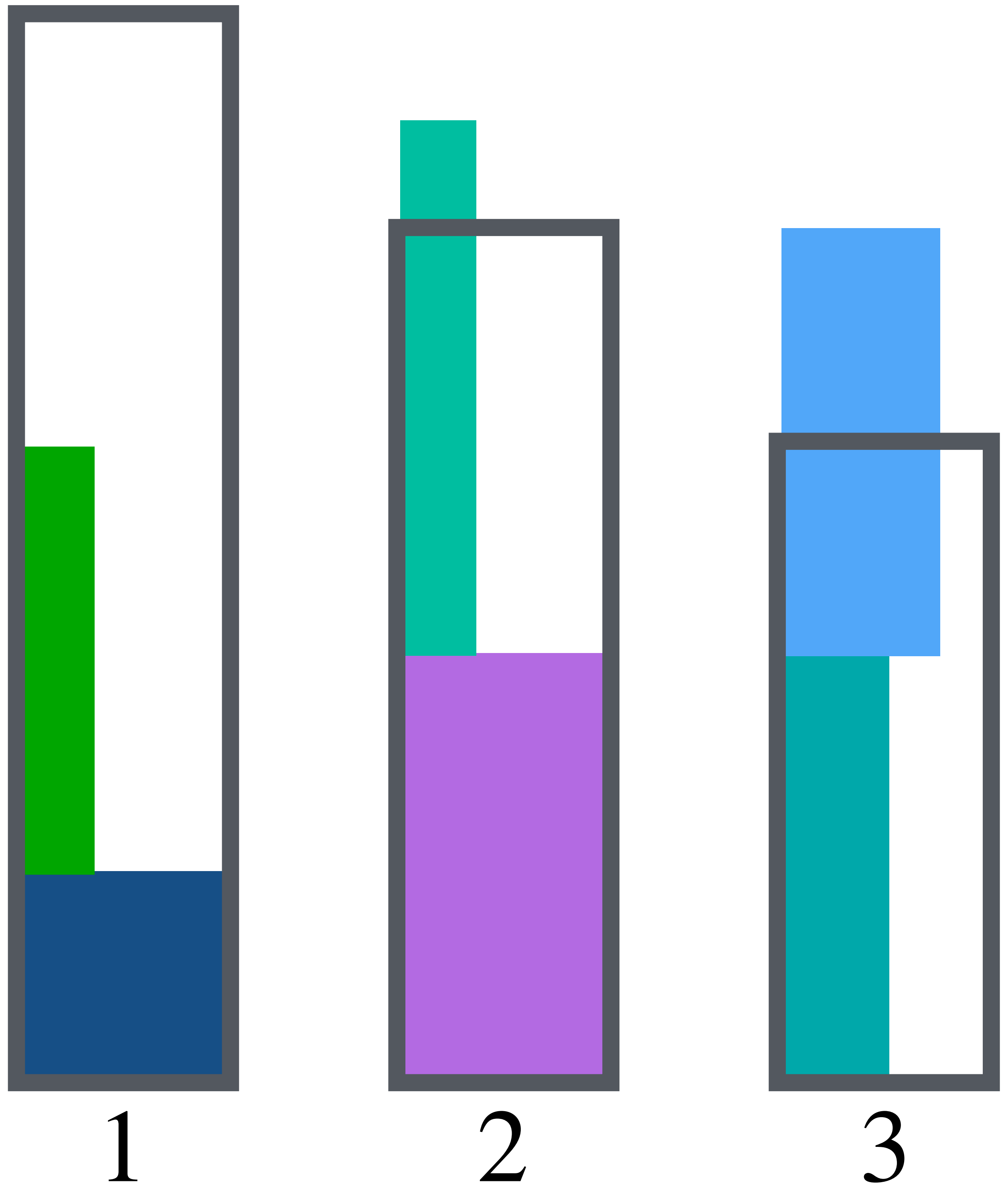
$1/9.37$ [Karrenbauer, Kovalevskaya 2020]
$1/4.39$ [Gilberti, Karrenbauer 2021]

tight $1/e$

Related work and our results

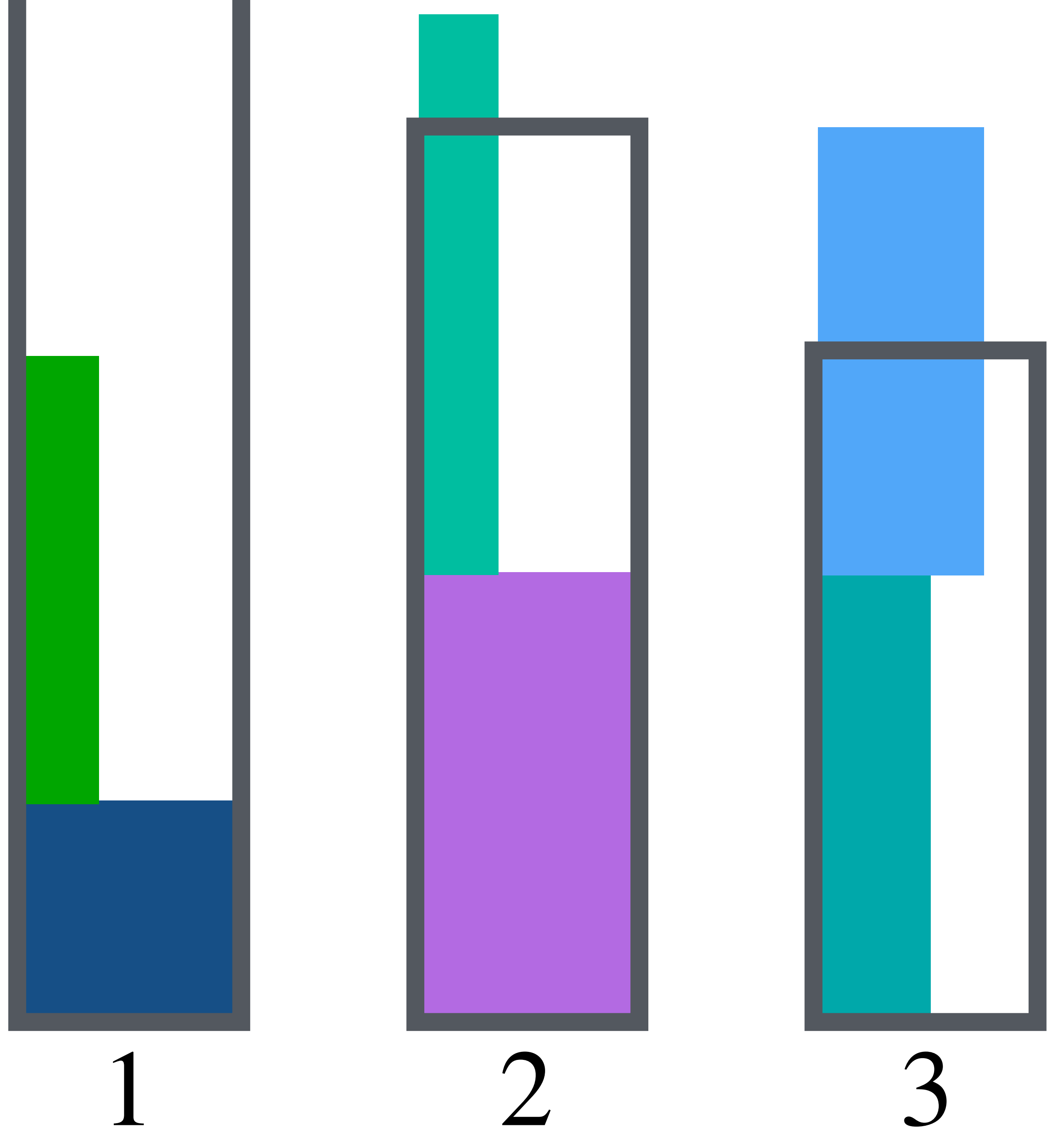
known competitive ratios		our results
secretary problem $(m = 1,\ C = 1,\ s_j = 1)$	tight $1/e$ [Dynkin 1963; Lindley1961]	
knapsack problem $(m = 1)$	$1/27.1$ [Babaioff, Immorlica, Kempe, Kleinberg 2007] $1/8.06$ [Kesselheim, Radke, Tönnis, Vöcking 2018] $1/6.65$ [Albers, Khan, Ladewig 2021]	$\boxed{\frac{1-\ln(2)}{2} = \frac{1}{6.52}}$
general assignment problem	$1/8.10$ [Kesselheim, Radke, Tönnis, Vöcking 2018] $1/6.99$ [Naori, Raz 2019; Albers, Khan, Ladewig 2021]	
fractional knapsack problem	$1/9.37$ [Karrenbauer, Kovalevskaya 2020] $1/4.39$ [Gilberti, Karrenbauer 2021]	tight $1/e$

Basic idea of algorithm



Basic idea of algorithm

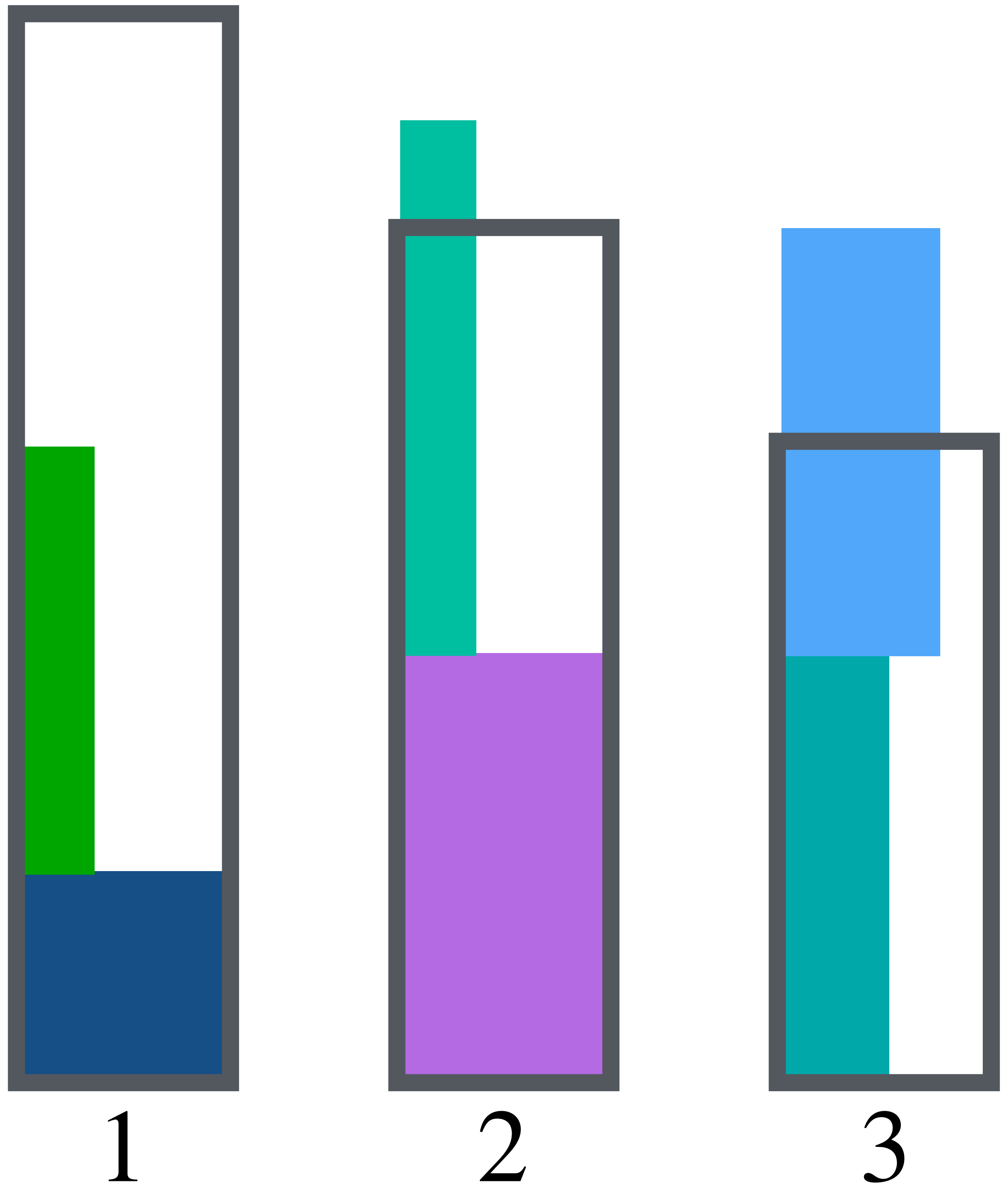
first compute infeasible solution
where 1 item per bin may overlap



Basic idea of algorithm

first compute infeasible solution
where 1 item per bin may overlap

discard first $n/2$ items



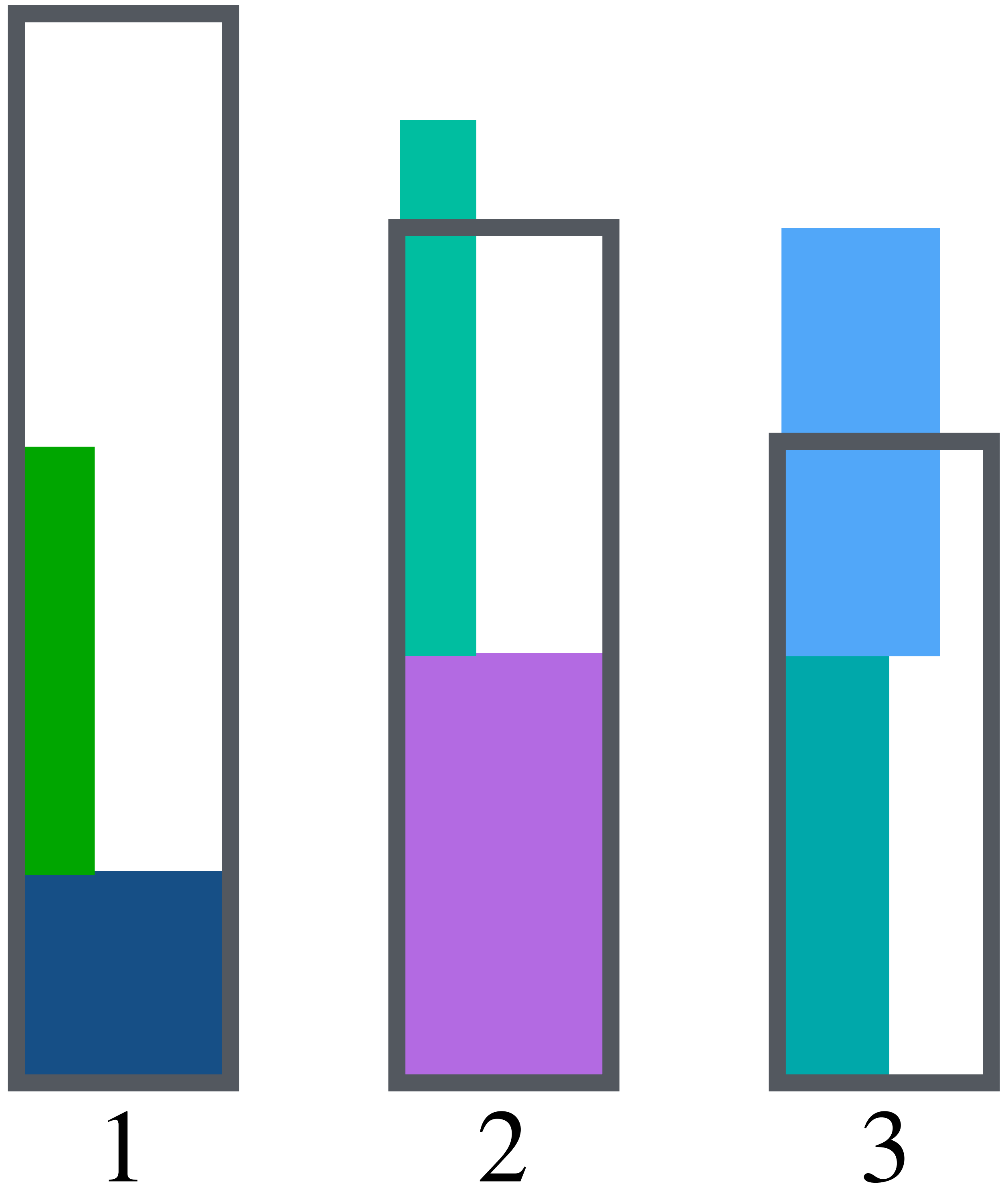
Basic idea of algorithm

first compute infeasible solution
where 1 item per bin may overlap

discard first $n/2$ items

for further items

- solve LP relaxation with all items seen so far
- assign item to bins with probability from the LP solution



Basic idea of algorithm

first compute infeasible solution where 1 item per bin may overlap

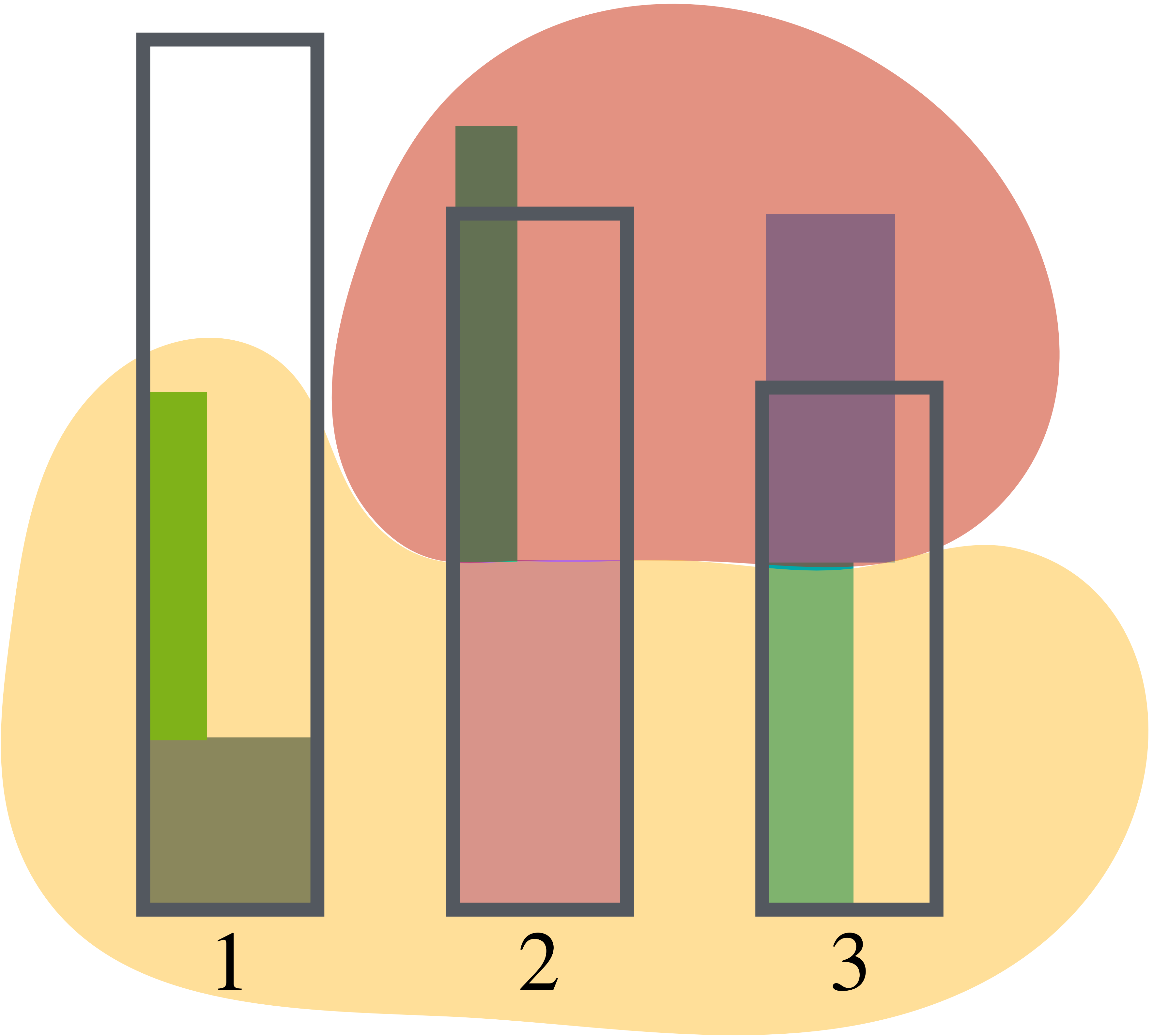
discard first $n/2$ items

for further items

- solve LP relaxation with all items seen so far
- assign item to bins with probability from the LP solution

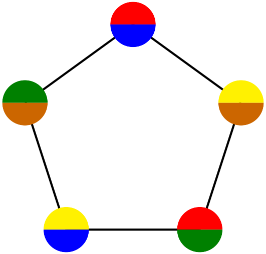
to obtain feasible solution choose with probability 1/2:

- solution without overlapping items 🟥
- overlapping items 🟨



Fractional Chromatic Numbers from Exact Decision Diagrams

Timo Brand (TU Munich) and Stephan Held (U Bonn)



Aussois, January 6, 2025

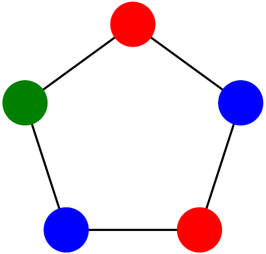
Fractional Chromatic Number

Chromatic number via stable set cover

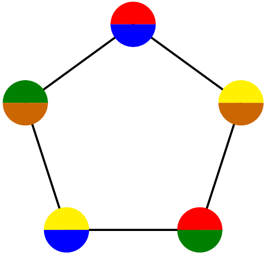
$$\begin{array}{lll} \min & \sum_{S\in\mathcal{S}} x_S & \\ s.t. & \sum_{S\in\mathcal{S}:v\in S} x_S \geq 1 & \forall v\in V \\ & x_S \in \{0,1\} & \forall S\in\mathcal{S} \end{array}$$

LP relaxation: fractional chromatic number

$$\begin{array}{lll} \min & \sum_{S\in\mathcal{S}} x_S & \\ s.t. & \sum_{S\in\mathcal{S}:v\in S} x_S \geq 1 & \forall v\in V \\ & x_S \in [0,1] & \forall S\in\mathcal{S} \end{array}$$

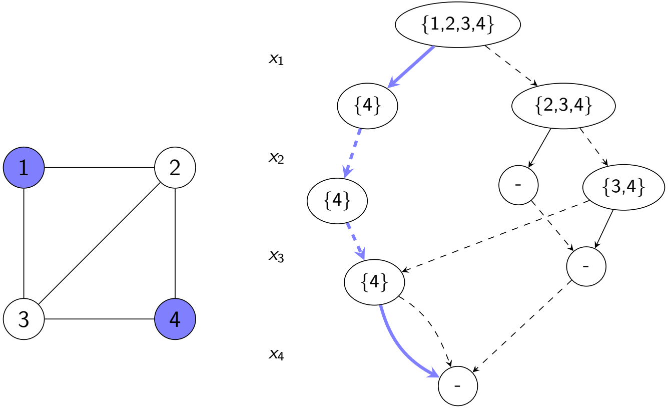


$\geq$



B-&-P: [Mehrotra & Trick '96, Gualandi & Malucelli '10, Malaguti, Monaci & Toth '10, Cook, H. & Sewell '12]
$\mathcal{S}$:= set of stable sets

Graph Coloring with Decision Diagrams (van Hoeve '22)

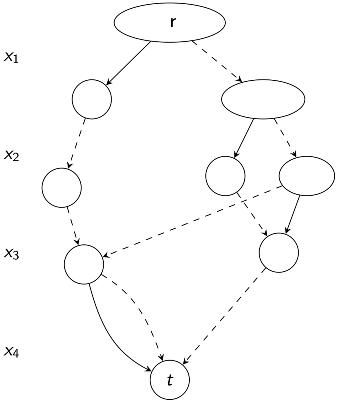


$|V| + 1$ layers
(one level of arcs per vertex)

solid arc $\widehat{=}$ vertex $i \in S$
dashed arc $\widehat{=}$ vertex $i \notin S$
stable sets $\widehat{=}$ maximal paths

Exact decision diagrams represent stable sets exactly.
Relaxed decision diagrams may contain unstable sets (van Hoeve's focus).

Flow ILP on Decision Diagrams (van Hoeve '22)



Flow ILP for graph coloring:

$$\begin{aligned}
\min \quad & \sum_{a\in\delta^+(r)} y_a \\
\text{s.t.} \quad & \sum_{a=(u,v):L(u)=j,\ell(a)=1} y_a \geq 1 && \forall j \in V \\
& \sum_{a\in\delta^-(u)} y_a - \sum_{a\in\delta^+(u)} y_a = 0 && \forall u \in N \setminus \{r,t\} \\
& y_a \in \{0,\dots,n\} && \forall a \in A
\end{aligned}$$

Covering of solid arc sets in each level
with an integral r-t-flow.

Van Hoeve reported lower bounds similar to set cover LP for relaxed decision diagrams.
Q: Which one is better?

Fractional Chromatic Numbers from Exact Decision Diagrams

Theorem (Brand & H.' 24)

In an exact decision diagram, the linear relaxation of the flow ILP determines the fractional chromatic number χ_f.

Consequences

- alternative method to compute χ_f.
- relaxed decision diagrams provide lower bounds for χ_f and χ.
  (set cover LP requires pricing to optimality)
- Using exact decision diagrams, we could solve a previously open DIMACS instance:

$$\chi(r1000.1c) = 98.$$

(Solving ILP with exact-SCIP [Eifler, Gleixner '22] in 3h).

Paper: arXiv:2411.03003, code & data archive: `https://doi.org/10.60507/FK2/ZE9C3L`

The Power of Proportional Fairness
for Non-Clairvoyant Polytope Scheduling

Sven Jäger[1]   Alexander Lindermayr[2]   **Nicole Megow**[2]

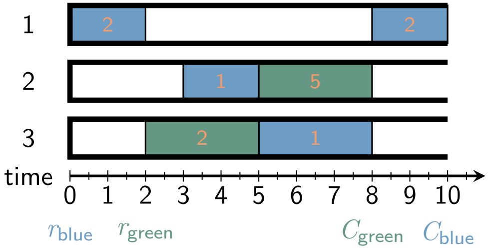
[1]University of Kaiserslautern-Landau (RPTU), Germany
[2]University of Bremen, Germany

Combinatorial Optimization Workshop, Aussois 01/2025

(Online) Unrelated Machine Scheduling

$R \mid r_j, \mathrm{pmtn} \mid \sum w_j C_j$

- n jobs, m unrelated machines
- processing requirements p_j
- speeds $s_{ij} \geq 0$
- find schedule $x_{ij}(t) \in \{0, 1\}$
- preemption and migration
- minimize $\sum_j w_j C_j$

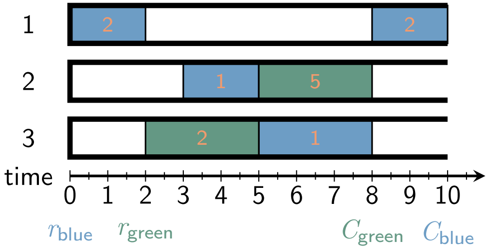


$$p_{\text{blue}} = 2 \cdot 2 + 2 \cdot 1 + 3 \cdot 1 + 2 \cdot 2 = 13$$

(Online) Unrelated Machine Scheduling

$R \mid r_j, \mathrm{pmtn} \mid \sum w_j C_j$

- n jobs, m unrelated machines
- processing requirements p_j
- speeds $s_{ij} \geq 0$
- find schedule $x_{ij}(t) \in \{0, 1\}$
- preemption and migration
- minimize $\sum_j w_j C_j$



$$p_{\text{blue}} = 2 \cdot 2 + 2 \cdot 1 + 3 \cdot 1 + 2 \cdot 2 = 13$$

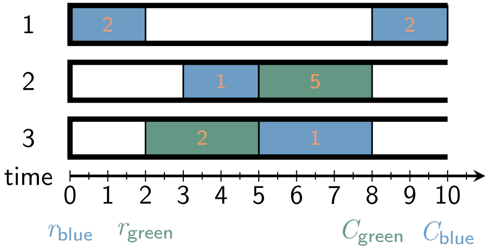
$Q \mid r_j, \mathrm{pmtn} \mid \sum w_j C_j$

- uniform speeds $s_i = s_{ij}\ \forall j$ on each machine i

(Online) Unrelated Machine Scheduling

$R \mid r_j, \mathrm{pmtn} \mid \sum w_j C_j$

- n jobs, m unrelated machines
- processing requirements p_j
- speeds $s_{ij} \geq 0$
- find schedule $x_{ij}(t) \in \{0, 1\}$
- preemption and migration
- minimize $\sum_j w_j C_j$



$Q \mid r_j, \mathrm{pmtn} \mid \sum w_j C_j$

- uniform speeds $s_i = s_{ij}$ $\forall j$ on each machine i

$$p_{\text{blue}} = 2 \cdot 2 + 2 \cdot 1 + 3 \cdot 1 + 2 \cdot 2 = 13$$

Online job arrival (onl-r_j)
job j unknown before r_j

Non-clairvoyance (nclv)
p_j unknown

(Online) Polytope Scheduling Problem (PSP)

Polytope Scheduling [Im, Kulkarni, and Munagala JACM'18]

- n jobs
- release dates r_j, weights w_j
- processing requirements p_j
- "rate" polytope $\mathcal{P} = \{y \in \mathbb{R}^n_{\geq 0} \mid By \leq 1\}$ for some $B \in \mathbb{Q}^{D \times n}$
- at any time t, choose rates $y(t) \in \mathcal{P}$
- $C_j := \arg\min_t \left(\int_{t'=0}^{t} y_j(t') \, \mathrm{d}t' \right) \geq p_j$
- minimize $\sum_j w_j C_j$

(Online) Polytope Scheduling Problem (PSP)

Polytope Scheduling [Im, Kulkarni, and Munagala JACM'18]

- n jobs
- release dates r_j, weights w_j
- processing requirements p_j
- "rate" polytope $\mathcal{P} = \{y \in \mathbb{R}^n_{\geq 0} \mid By \leq 1\}$ for some $B \in \mathbb{Q}^{D \times n}$
- at any time t, choose rates $y(t) \in \mathcal{P}$
- $C_j := \arg\min_t \left(\int_{t'=0}^{t} y_j(t') \, \mathrm{d}t' \right) \geq p_j$
- minimize $\sum_j w_j C_j$

Unrelated Machine Scheduling is a PSP with the polytope (before projection)

$$\left\{ (y, x) \in \mathbb{Q}_{\geq 0}^{n \times (m \times n)} \;\middle|\; y_j = \sum_{i=1}^{m} s_{ij} x_{ij} \, \forall j, \; \sum_{j=1}^{n} x_{ij} \leq 1 \, \forall i, \; \sum_{i=1}^{m} x_{ij} \leq 1 \, \forall j \right\} .$$

Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \leq \rho \cdot \mathrm{OPT}(I)$ for all I.

Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

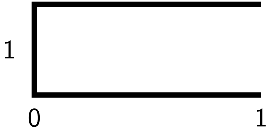
Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \leq \rho \cdot \mathrm{OPT}(I)$ for all I.

Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

Adversarial strategy: ensure that no job finishes until time 1; then complete all.



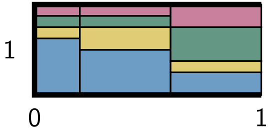
Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \leq \rho \cdot \mathrm{OPT}(I)$ for all I.

Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

Adversarial strategy: ensure that no job finishes until time 1; then complete all.



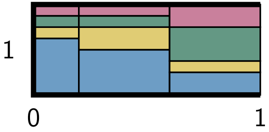
Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \leq \rho \cdot \mathrm{OPT}(I)$ for all I.

Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

Adversarial strategy: ensure that no job finishes until time 1; then complete all.



$\mathrm{ALG}(I) \geq n$

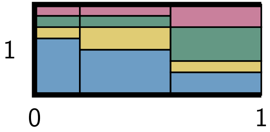
Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \leq \rho \cdot \mathrm{OPT}(I)$ for all I.

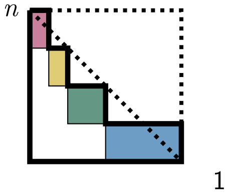
Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

Adversarial strategy: ensure that no job finishes until time 1; then complete all.



$\mathrm{ALG}(I) \geq n$



$\mathrm{OPT}(I) \leq \frac{1}{2}n + \frac{1}{2}$

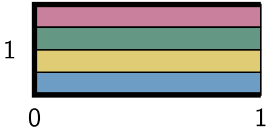
Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \le \rho \cdot \mathrm{OPT}(I)$ for all I.

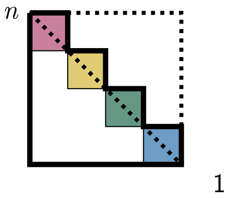
Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

Adversarial strategy: ensure that no job finishes until time 1; then complete all.



$\mathrm{ALG}(I) \ge n$



$\mathrm{OPT}(I) = \frac{1}{2}n + \frac{1}{2}$

Ratio approaches 2 if all jobs receive the same rate ⟶ Round-Robin [MPT94]

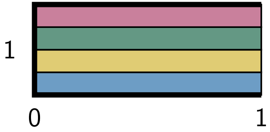
Non-Clairvoyant Scheduling

We say an online algorithm is ρ-competitive if $\mathrm{ALG}(I) \leq \rho \cdot \mathrm{OPT}(I)$ for all I.

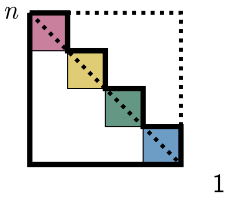
Theorem (Motwani, Phillips, Torng '94)

There is no better-than-2-competitive non-clairvoyant algorithm for minimizing the total completion time on a single machine.

Adversarial strategy: ensure that no job finishes until time 1; then complete all.



$\mathrm{ALG}(I) \geq n$



$\mathrm{OPT}(I) = \frac{1}{2}n + \frac{1}{2}$

Ratio approaches 2 if all jobs receive the same rate $\longrightarrow$ Round-Robin [MPT94]

From the jobs perspective, we seek fair rates.

Proportional Fairness

Fair allocation and market equilibria: Fisher markets [Eisenberg and Gale '59], one-sided matching markets [Jain and Vazirani '10] [Garg, Tröbst, Vazirani '22]

Proportional Fairness (PF): [Nash 1950, Eisenberg & Gale 1959, Kaneko & Nakamura 1979]

$$\mathrm{PF}(J) \quad \operatorname*{arg\,max}_{y \in \mathcal{P}} \left(\prod_{j \in J} y_j^{w_j} \right)^{1/\sum w_j} = \operatorname*{arg\,max}_{y \in \mathcal{P}} \sum_{j \in J} w_j \log y_j \ .$$

At any time t, schedule PF$(J(t))$ on set of available jobs $J(t)$. [Im, Kulkarni, Munagala 2018]

Proportional Fairness

Fair allocation and market equilibria: Fisher markets [Eisenberg and Gale '59], one-sided matching markets [Jain and Vazirani '10] [Garg, Tröbst, Vazirani '22]

Proportional Fairness (PF): [Nash 1950, Eisenberg & Gale 1959, Kaneko & Nakamura 1979]

$$\mathrm{PF}(J) \quad \operatorname*{arg\,max}_{y \in \mathcal{P}} \left(\prod_{j \in J} y_j^{w_j} \right)^{1/\sum w_j} = \operatorname*{arg\,max}_{y \in \mathcal{P}} \sum_{j \in J} w_j \log y_j \ .$$

At any time t, schedule PF$(J(t))$ on set of available jobs $J(t)$. [Im, Kulkarni, Munagala 2018]

Round-Robin is the special case of PF for $1 \mid \mathrm{pmtn} \mid \sum C_j$

Proportional Fairness

Fair allocation and market equilibria: Fisher markets [Eisenberg and Gale '59], one-sided matching markets [Jain and Vazirani '10] [Garg, Tröbst, Vazirani '22]

Proportional Fairness (PF): [Nash 1950, Eisenberg & Gale 1959, Kaneko & Nakamura 1979]

$$\mathsf{PF}(J) \quad \operatorname*{arg\,max}_{y \in \mathcal{P}} \left(\prod_{j \in J} y_j^{w_j} \right)^{1/\sum w_j} = \operatorname*{arg\,max}_{y \in \mathcal{P}} \sum_{j \in J} w_j \log y_j \ .$$

At any time t, schedule PF($J(t)$) on set of available jobs $J(t)$. [Im, Kulkarni, Munagala 2018]

Round-Robin is the special case of PF for $1 \mid \mathrm{pmtn} \mid \sum C_j$

Important subclass of PSP: PF-monotone PSP (short $\textsc{MonPSP}$)

$$y = \mathsf{PF}(J) \text{ and } y' = \mathsf{PF}(J') \text{ with } J' \subseteq J \implies y_j \leq y'_j \ \ \forall j \in J' \ .$$

Our Results

We improve the analysis of PF for PSP via PF-monotonicity and α-superadditivity:

Theorem 1

PF is 4-competitive for MonPSP.

$Q \mid r_j, \mathrm{pmtn} \mid \sum w_j C_j$ and $R \mid r_j, \mathrm{pmtn}, s_{ij} \in \{0, 1\} \mid \sum w_j C_j$ are MonPSP.

Theorem 2

PF has a competitive ratio of at most

- $2\alpha + 1$ for α-superadditive PSP with non-uniform release dates, and
- 2α for α-superadditive PSP with uniform release dates.

- $R \mid \mathrm{pmtn} \mid \sum w_j C_j$ is 1.81-superadditive.
- $Q \mid \mathrm{pmtn} \mid \sum C_j$ is 1-superadditive.
- $R \mid \mathrm{pmtn}, s_{ij} \in \{0, 1\} \mid \sum C_j$ is 1-superadditive.
- $P \mid \mathrm{pmtn} \mid \sum w_j C_j$ is 1-superadditive.

Our Results

We improve the analysis of PF for PSP via PF-monotonicity and α-superadditivity:

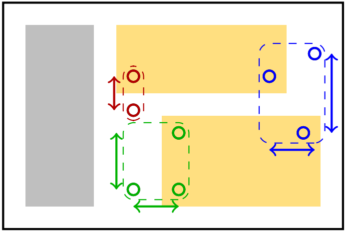
	old bounds (poly-time)		**our bounds**
Problem	onl-r_j & nclv	onl-r_j	onl-r_j & nclv
PSP	128 [IKM18]	128	**27**
MonPSP	25.74 [IKM18]	25.74	**4**
$R \mid r_j, \text{pmtn} \mid \sum w_j C_j$	32 [IKMP14]	5.78 [CPS+96]	**4.62**
$R \mid \text{pmtn} \mid \sum w_j C_j$	32 [IKMP14]	-	**3.62**
$R \mid r_j, \text{pmtn}, s_{ij} \in \{0, 1\} \mid \sum w_j C_j$	25.74	5.78	**4**
$R \mid \text{pmtn}, s_{ij} \in \{0, 1\} \mid \sum C_j$	25.74	-	**2**
$R \mid r_j, \text{pmtn}, s_{ij} = s_i \mid \sum w_j C_j$	25.74	5.78	**4**
$R \mid \text{pmtn}, s_{ij} = s_i \mid \sum C_j$	25.74	-	**2**

Integrating routing congestion into analytic placement

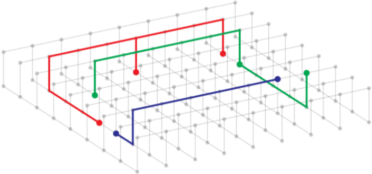
Martin Drees

Research Institute for Discrete Mathematics, Bonn

Placement and routing



- Place cells overlap-free
- Minimize netlength
- Global: Avoid high density

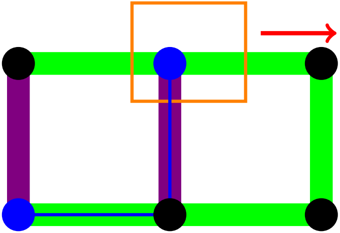


- Connect pins with disjoint Steiner trees
- Global: Avoid high congestion

Flat analytic placement

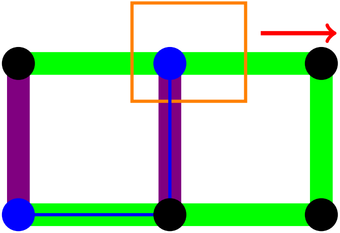
- Minimize *wirelength* $+ \lambda \cdot$ *density_penalty*
- Use variant of gradient descent

Considering routing congestion



- Extend objective function:
  $\textit{wirelength} + \lambda_1 \cdot \textit{density\_penalty} + \lambda_2 \cdot \textit{congestion\_penalty}$
- Given: Grid graph with congestion costs on edges
- Goal: Efficiently compute congestion costs for nets

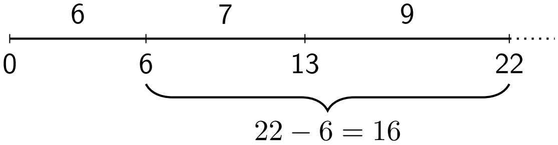
Considering routing congestion



- Extend objective function:
  $\textit{wirelength} + \lambda_1 \cdot \textit{density\_penalty} + \lambda_2 \cdot \textit{congestion\_penalty}$
- Given: Grid graph with congestion costs on edges
- Goal: Efficiently compute congestion costs for nets
- Simplifications:
  - Two-terminal nets (introduce Steiner vertices for larger nets)
  - Pins are on vertices of grid graph (interpolate)
  - Only L-shaped paths (subdivide)

Evaluating congestion costs efficiently

- Same congestion costs for many paths $\implies$ preprocessing
- For every row and column, compute consecutive sums
- L-shaped paths can be efficiently computed using these



Sparse Sub-gaussian Random Projections for Semidefinite Programming Relaxations

Lars Schewe (joint work with Monse Guedes-Ayala, Pierre-Louis Poirion, Akiko Takeda)

Aussois 2025

Our problem

SDP-relaxations

- Powerful,
- but often very large problems

The approach

Random projections

- In general: Projections to small spaces approximately preserve distances.
- Can be exploited for various optimization algorithms

Our case

- Projecting the matrix variable of an arbitrary SDP

$$\begin{aligned} \min \quad & \langle C, X \rangle \\ \text{s.t.} \quad & \langle A_i, X \rangle = b_i \quad i \in \{1, \dots, m\}, \\ & X \succeq 0 \end{aligned} \qquad \begin{aligned} \min \quad & \langle PCP^\top, Y \rangle \\ \text{s.t.} \quad & \langle PA_iP^\top, Y \rangle = b_i \quad i \in \{1, \dots, m\}, \\ & Y \succeq 0 \end{aligned}$$

Results

- Bounds on the projection error
- Able to reconstruct feasible solutions
- Works reasonably well for problems with few constraints

Paper

Sparse Sub-gaussian Random Projections for Semidefinite Programming Relaxations
Monse Guedes-Ayala, Pierre-Louis Poirion, Lars Schewe, Akiko Takeda

https://arxiv.org/abs/2406.14249

Bonus: Solving real-world optimization problems in electricty transmission networks

Cannot present my projects (yet)

- …but I am happy to talk about it.

Ask me about electricity networks!

Something (surprising?) about the assignment problem

Marco Di Summa

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \text{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n \times d}$ and $P \in \mathbb{R}^{d \times n}$.

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \text{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n \times d}$ and $P \in \mathbb{R}^{d \times n}$.
- Each row c_i of $C \to$ linear function $c_i x$

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \text{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n \times d}$ and $P \in \mathbb{R}^{d \times n}$.
- Each row c_i of $C \rightarrow$ linear function $c_i x$
- Let $K_i = \{x \in \mathbb{R}^d : c_i x \geq c_h x \ \forall \, h\}$

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \operatorname{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n \times d}$ and $P \in \mathbb{R}^{d \times n}$.
- Each row c_i of C → linear function $c_i x$
- Let $K_i = \{x \in \mathbb{R}^d : c_i x \geq c_h x \ \forall\, h\}$. K_i is a cone in $\mathbb{R}^d$ (and these cones form the normal fan of a polytope).

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \text{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n \times d}$ and $P \in \mathbb{R}^{d \times n}$.
- Each row c_i of $C \rightarrow$ linear function $c_i x$
- Let $K_i = \{x \in \mathbb{R}^d : c_i x \geq c_h x \ \forall\, h\}$. K_i is a cone in $\mathbb{R}^d$ (and these cones form the normal fan of a polytope).
- Each column p_j of $P \rightarrow$ point $p_j \in \mathbb{R}^d$

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n \times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \operatorname{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n \times d}$ and $P \in \mathbb{R}^{d \times n}$.
- Each row c_i of $C \rightarrow$ linear function $c_i x$
- Let $K_i = \{x \in \mathbb{R}^d : c_i x \geq c_h x \ \forall \, h\}$. K_i is a cone in $\mathbb{R}^d$ (and these cones form the normal fan of a polytope).
- Each column p_j of $P \rightarrow$ point $p_j \in \mathbb{R}^d$
- Any feasible solution to the assignment problem can be seen as assigning each cone K_i to a different point p_j (i.e., translate K_i so that p_j is its apex)

Something (surprising?) about the assignment problem

Marco Di Summa

- Assignment problem: Given a matrix $A \in \mathbb{R}^{n\times n}$, select n elements —one per row and per column— that maximize the sum of the entries of A.
- Let $d = \text{rank}(A)$. Then $A = CP$ with $C \in \mathbb{R}^{n\times d}$ and $P \in \mathbb{R}^{d\times n}$.
- Each row c_i of C → linear function $c_i x$
- Let $K_i = \{x \in \mathbb{R}^d : c_i x \geq c_h x \ \forall\, h\}$. K_i is a cone in $\mathbb{R}^d$ (and these cones form the normal fan of a polytope).
- Each column p_j of P → point $p_j \in \mathbb{R}^d$
- Any feasible solution to the assignment problem can be seen as assigning each cone K_i to a different point p_j (i.e., translate K_i so that p_j is its apex)
- If (but not "only if") the solution to the assignment problem is also optimal, the interiors of the translated cones are pairwise disjoint.

Something (surprising?) about the assignment problem

Marco Di Summa

This can be done also when the cones do not form the normal fan of a polytope:

Something (surprising?) about the assignment problem

Marco Di Summa

This can be done also when the cones do not form the normal fan of a polytope:

Theorem (perhaps useless, but did you know —and can you believe— this?)

Given n *points in* $\mathbb{R}^d$ *and* n *cones with pairwise disjoint interiors, it is always possible to "assign" cones to points so that the interiors of the translated cones are pairwise disjoint (and this can be done by solving an assignment problem).*

Something (surprising?) about the assignment problem

Marco Di Summa

This can be done also when the cones do not form the normal fan of a polytope:

Theorem (perhaps useless, but did you know —and can you believe— this?)

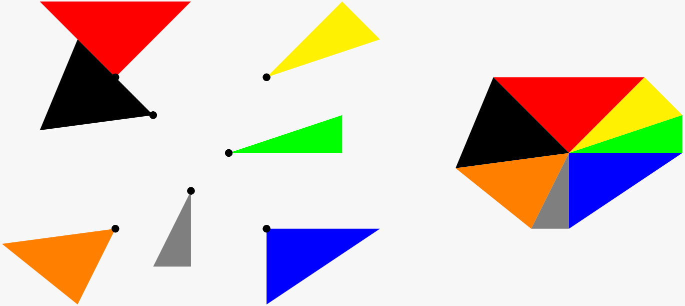
Given n points in $\mathbb{R}^d$ and n cones with pairwise disjoint interiors, it is always possible to "assign" cones to points so that the interiors of the translated cones are pairwise disjoint (and this can be done by solving an assignment problem).

Something (surprising?) about the assignment problem

This can be done also when the cones do not form the normal fan of a polytope:

Theorem (perhaps useless, but did you know —and can you believe— this?)

Given n points in $\mathbb{R}^d$ and n cones with pairwise disjoint interiors, it is always possible to "assign" cones to points so that the interiors of the translated cones are pairwise disjoint (and this can be done by solving an assignment problem).



Identifying when thresholds from the Paris Agreement are breached: the minmax average, a novel smoothing approach

Why we might have breached the 1.5°C limit already in July 2023

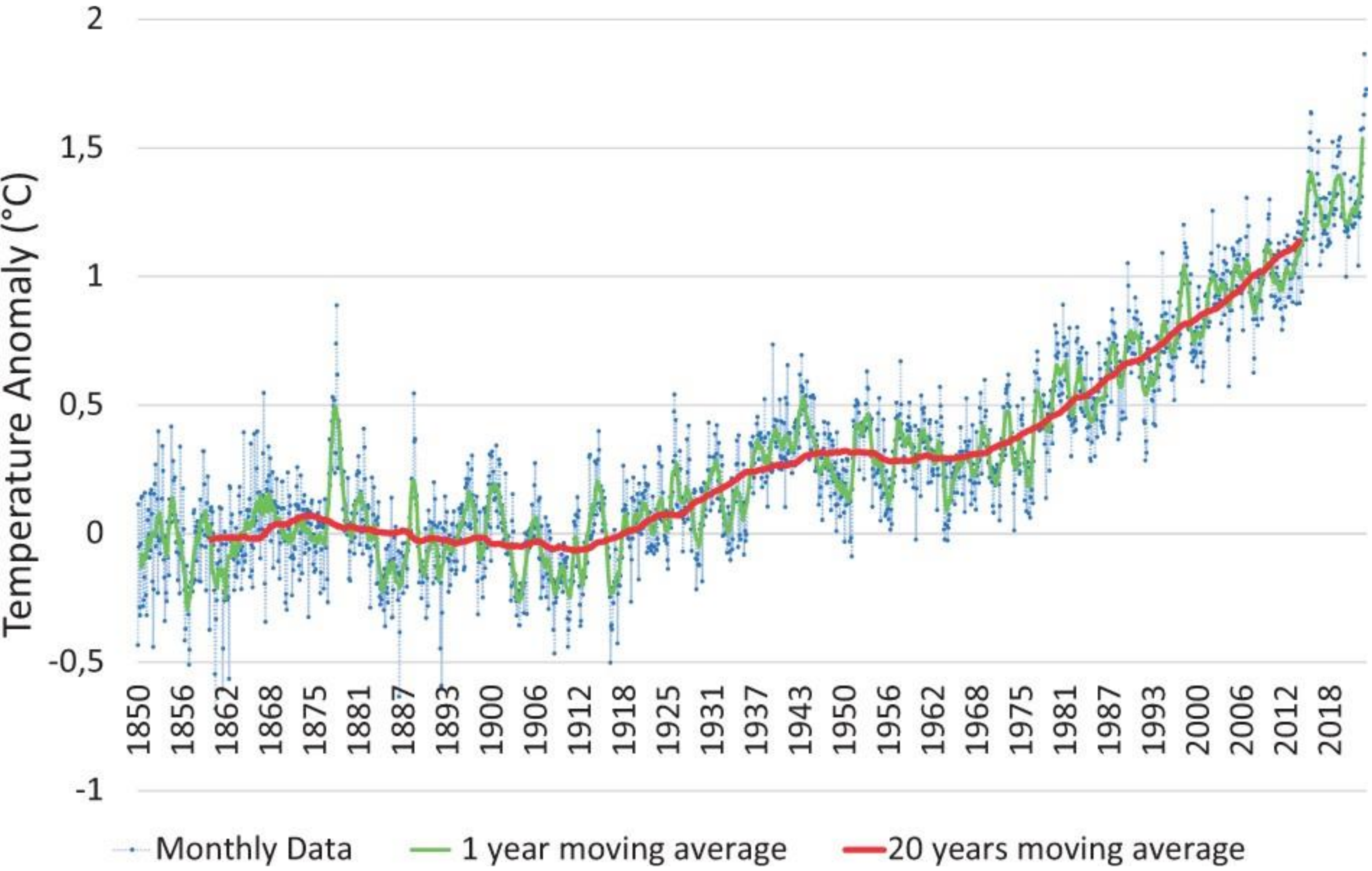
Aussois, January 2025

When will we reach 1.5°C ?

Approaching 1.5 °C: how will we know we’ve reached this crucial warming mark?

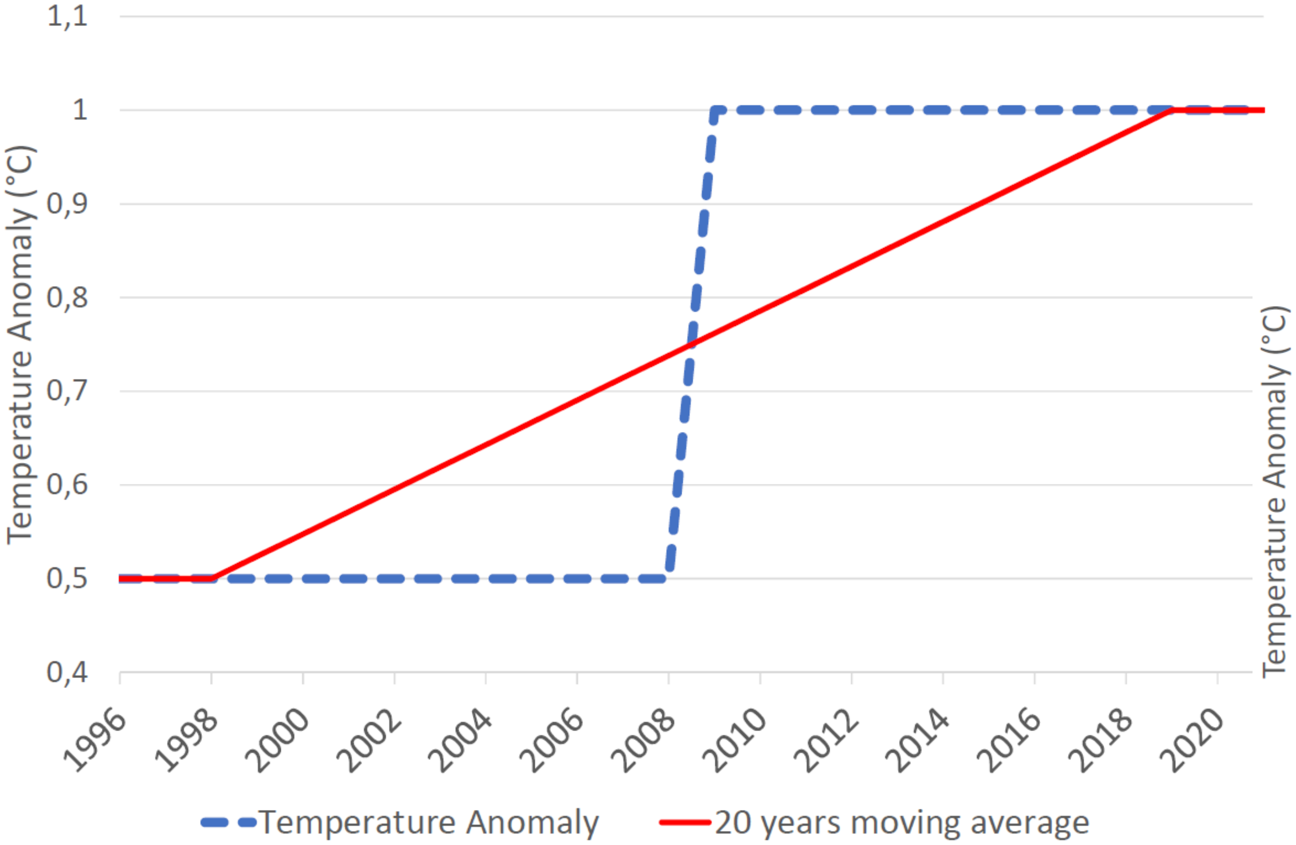
Assessing global mean temperature rise using the average warming over the p one or two decades will delay formal recognition of when Earth breaches the F agreement’s 1.5 °C guard rail. Here is what’s needed to avoid the wait.

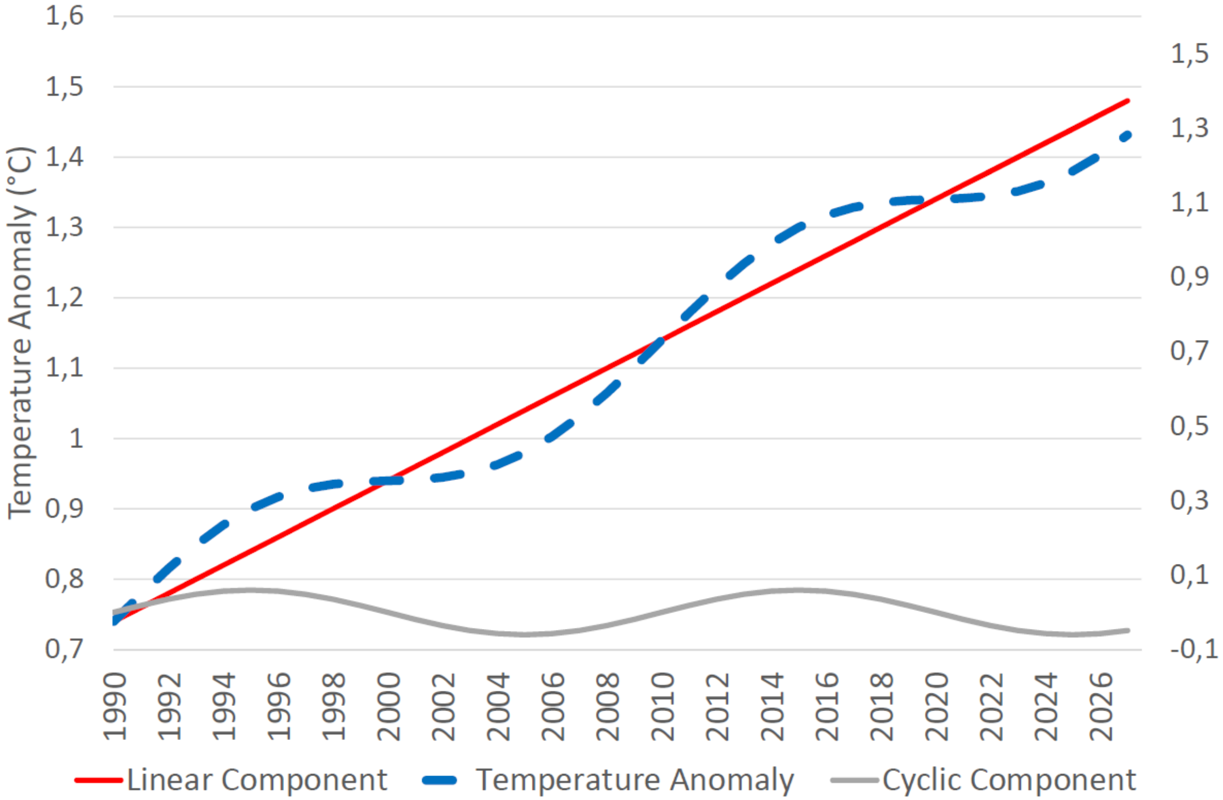
By Richard A. Betts, Stephen E. Belcher, Leon Hermanson, Albert Klein Tank, Jason A. Lowe,



Claim : a sound methodology should give given the natural answers when data is monotone

Purely data based methodologies: moving average, LOESS, …



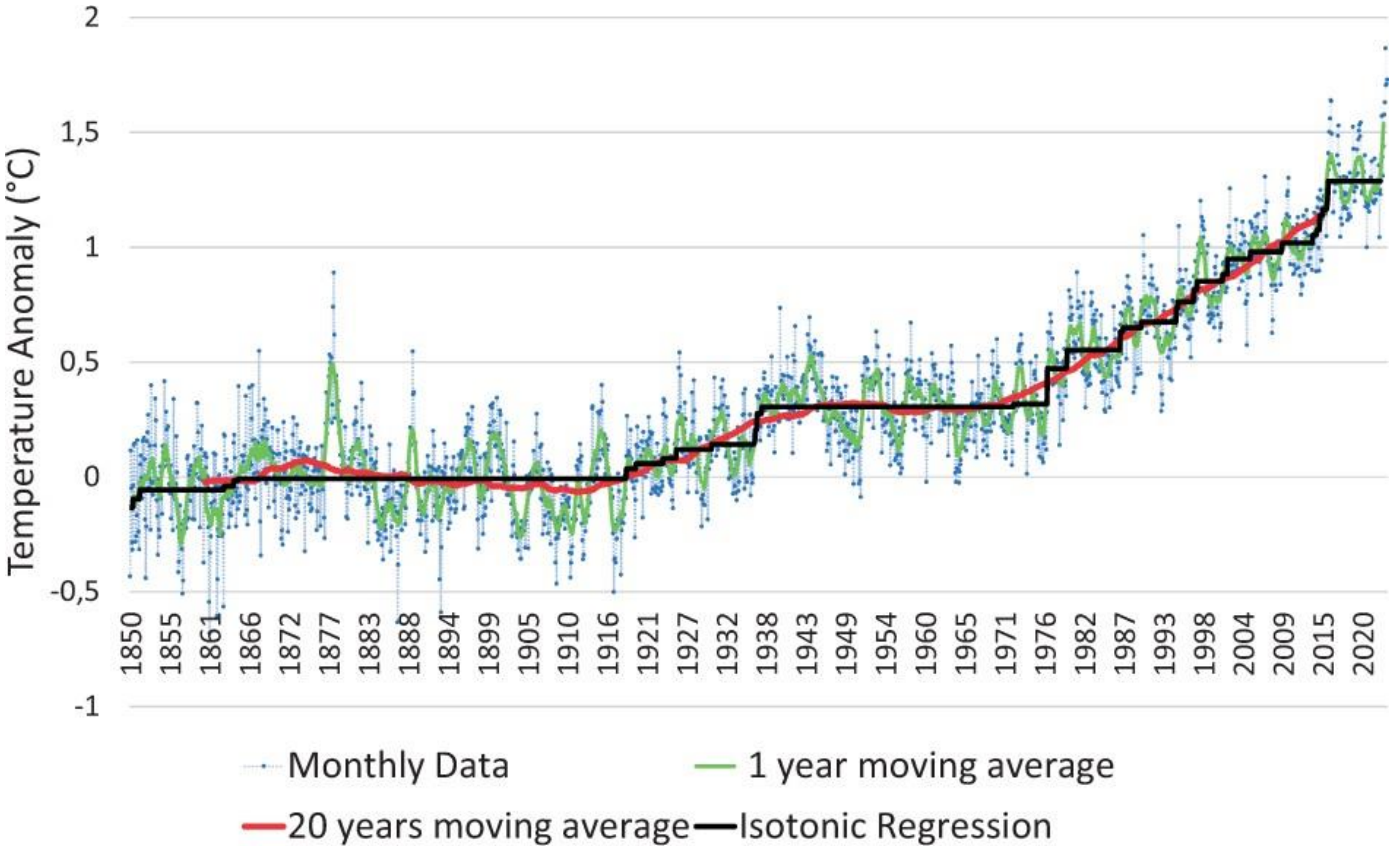


First attempt : Isotonic Regression

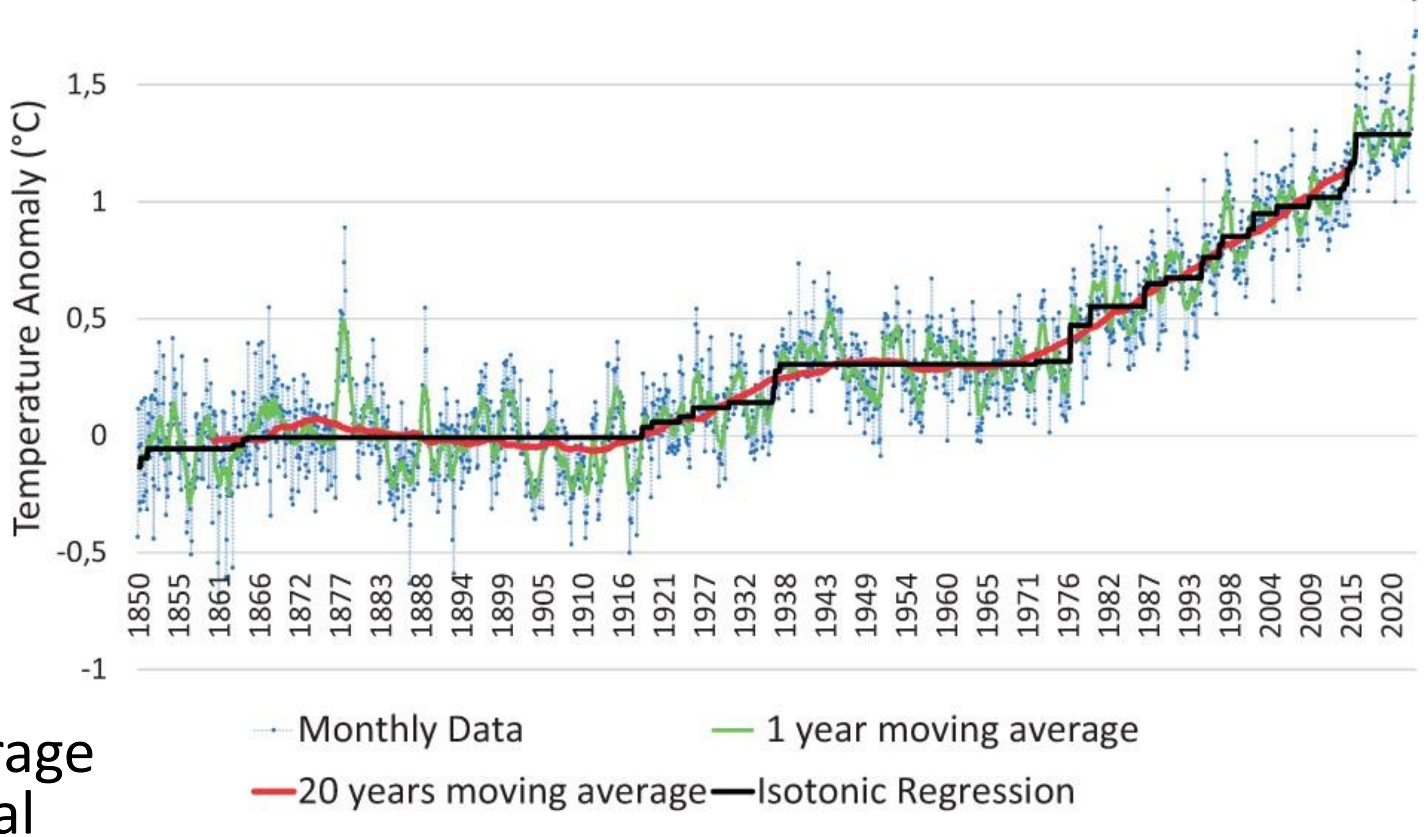
Our question has a clear answer only when the time series is *montonously increasing*. So why not
computing the closest time series with that property ?

$$\min_x \sum_{i=M}^{N} (x_i - T_i)^2$$

$$x_i \geq x_{i-1} \qquad \forall i > M$$



Properties of the isotonic regression



- Interval decomposition
    - Constant within
    - Strictly increasing across
- Within an interval
    - The interval value is the average of the data within the interval
- Within the interval, the data is decreasing in the following sense

$$\text{for } j \in [m, n-1], T_{m,j} \geq T_{j+1,n}$$

- The endpoint n of the interval [m,n] is the minimizer of $T_{m,j} \text{ for } j \geq m$

21 Best MJ, Chakravarti N. Active set algorithms for isotonic regression; a unifying framework. *Math Prog* 1990;47:425–39.

MinMax Average

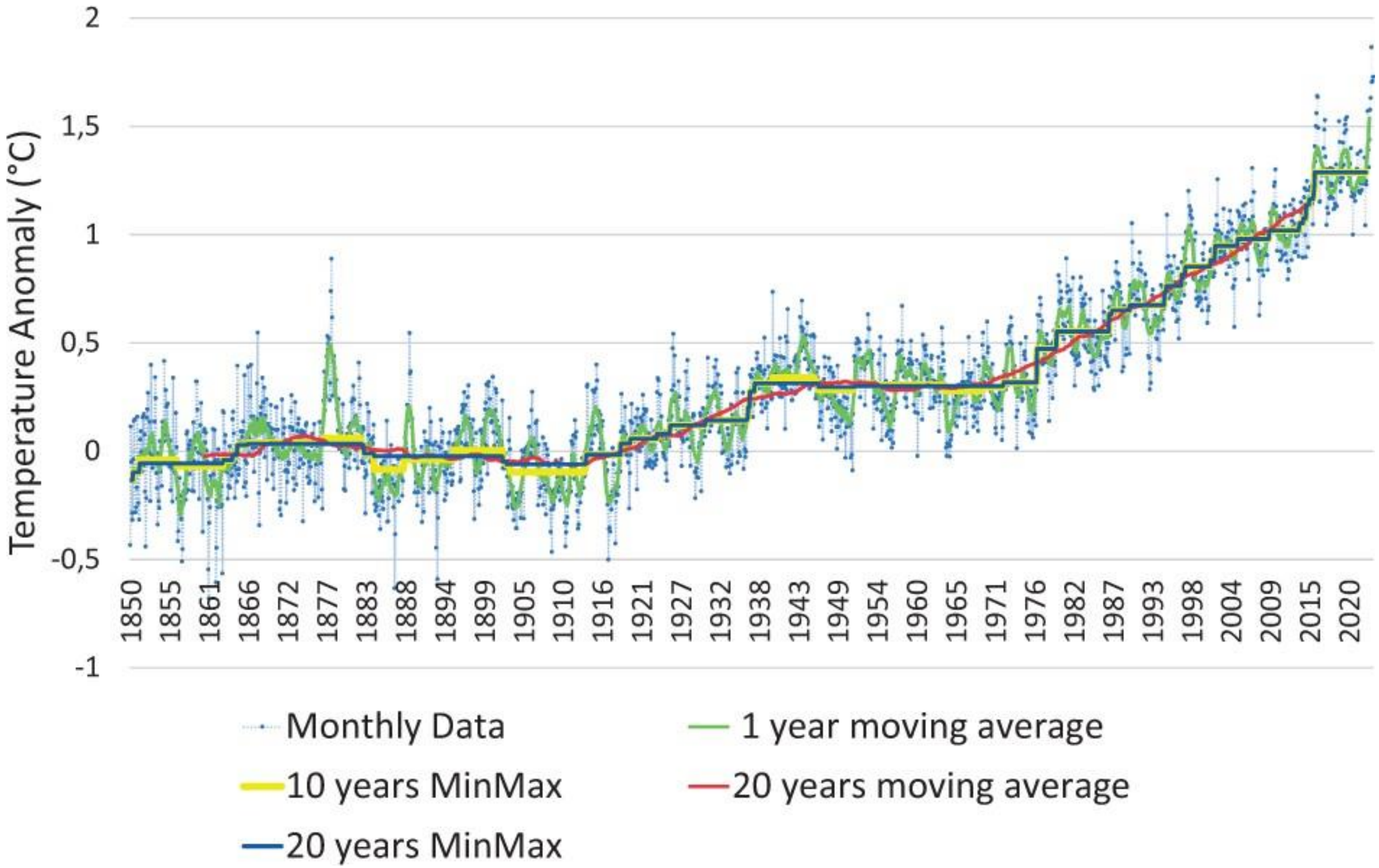
- Intuition (necessary conditions) : Reaching the threshold L « for good » in period i means
  - $T_i \geq L$
  - $T_{i,i+1} \geq L$
  - $T_{i,i+2} \geq L$
  - … up to period i+K-1

$$\underline{T}_i^K \triangleq \min_{p \in [i, i+K-1]} T_{i,p}$$

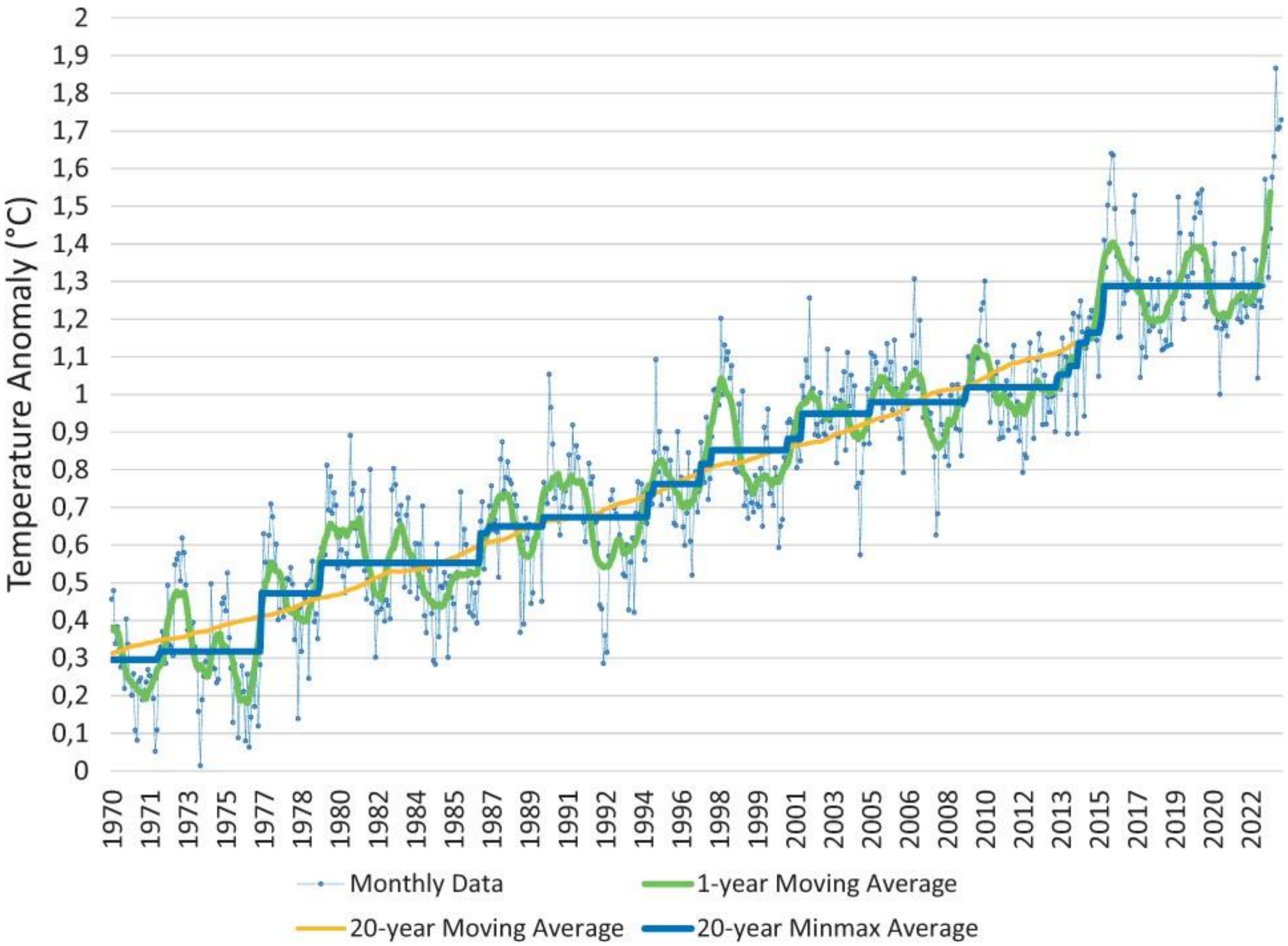
$$\overline{T}_i^K \triangleq \max_{p \in [i, i+K-1]} T_{i,p}$$

$$\tilde{T}_i^K = \begin{cases} T_i & \text{if } i = M, \\ \tilde{T}_{i-1}^K & \text{if } \underline{T}_i^K \leq \tilde{T}_{i-1}^K \leq \overline{T}_i^K, \\ \underline{T}_i^K & \text{if } \tilde{T}_{i-1}^K < \underline{T}_i^K, \\ \overline{T}_i^K & \text{if } \tilde{T}_{i-1}^K > \overline{T}_i^K. \end{cases}$$

MinMax Average

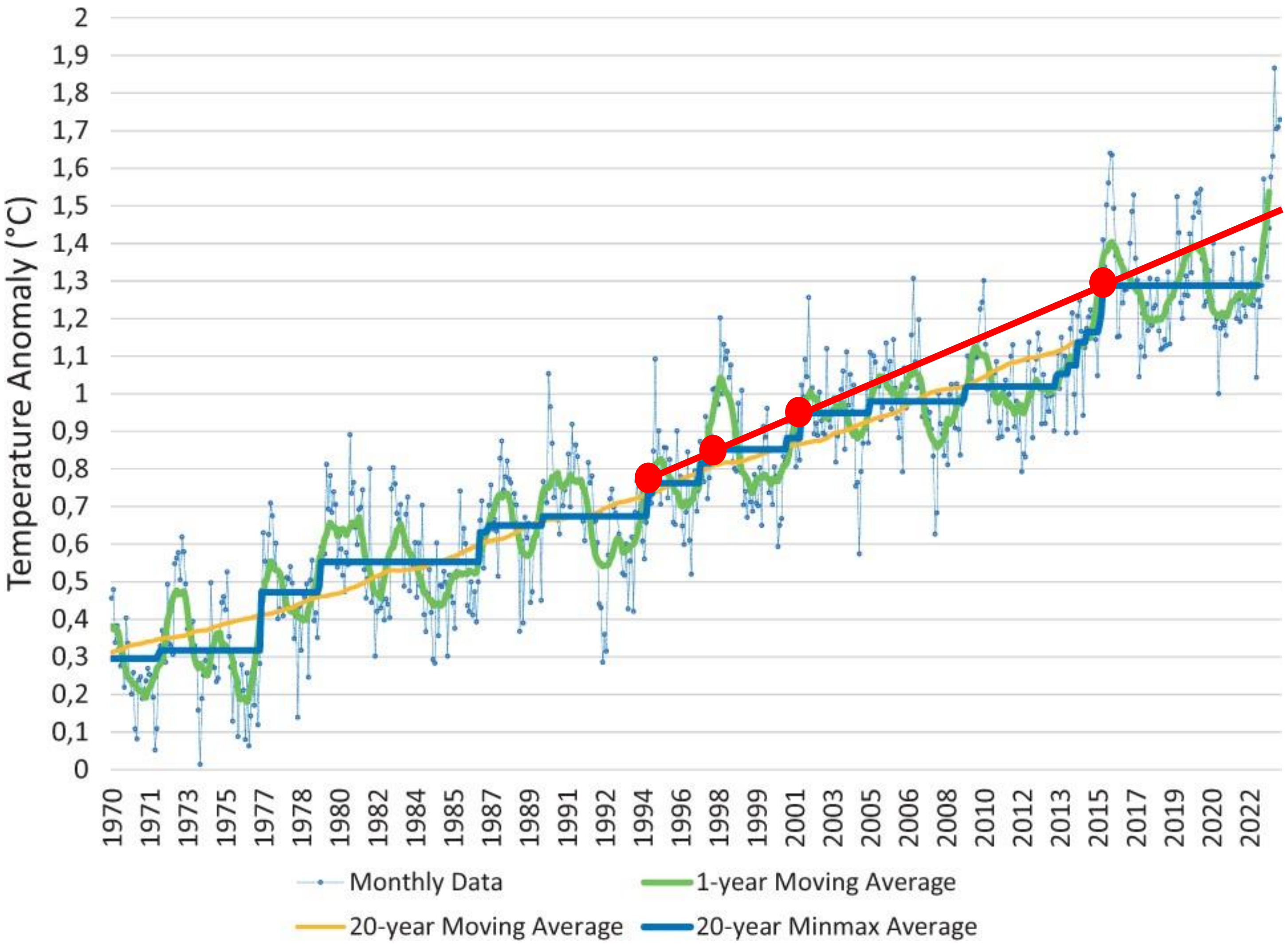


The recent period : 1970 - 2023



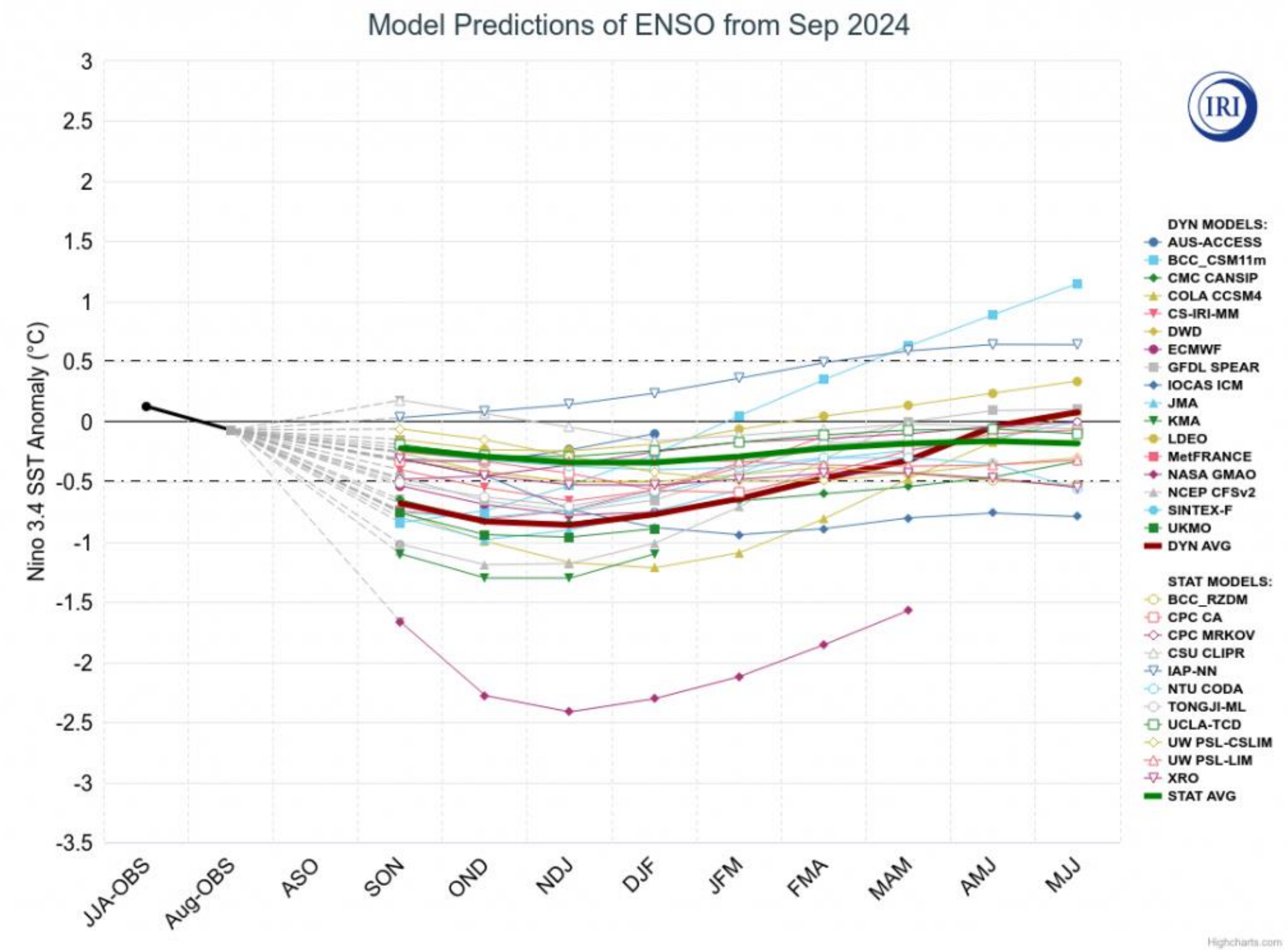
- Each constant interval spans 1 or 2 ups and down (nearly by construction)
- Big El Ninõ in ‘97-’98 and in ‘15-’16.
- « Hiatus » of 2001-2013
- « Hiatus » of 2016-2023

The recent period : 1970 - 2023

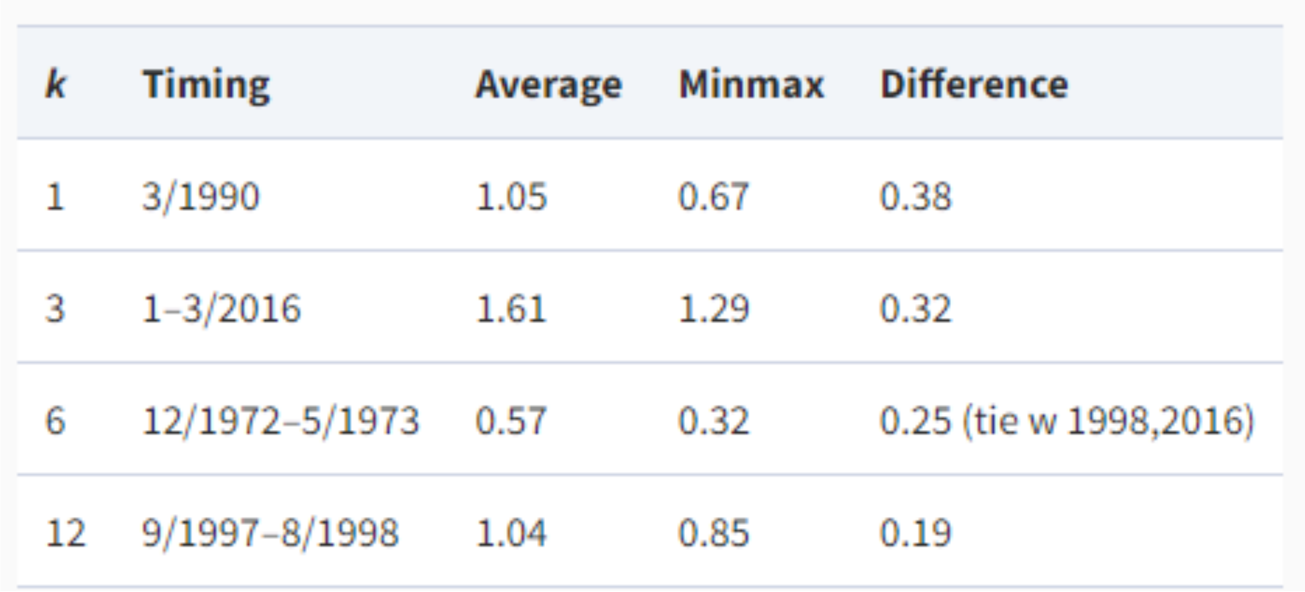


- Each constant interval spans 1 or 2 ups and down (nearly by construction)
- Big El Ninõ in ‘97-’98 and in ‘15-’16.
- « Hiatus » of 2001-2013
- « Hiatus » of 2016-2023
- +0.25°C/decade since 1994

El Ninõ projections : very hard



Maximum difference between k-months moving and minmax averages



k	Timing	Average	Minmax	Difference	'23-'24	Implied lower bound
1	3/1990	1.05	0.67	0.38	1.88	1.50
3	1–3/2016	1.61	1.29	0.32	1.79	1.47
6	12/1972–5/1973	0.57	0.32	0.25 (tie w 1998,2016)	1.75	1.50
12	9/1997–8/1998	1.04	0.85	0.19	1.69	1.50

Average July 2023 – September 2024 : 1.68°C

For the July 2023 minmax average to be below 1.5°C, all these records would need to be broken

Conclusion

- Many ways to smooth out up-and-downs, but some make more sense (less assumptions, closer to meaning of “threshold” in English, closer to data)
- We’ll know for sure if we have passed 1.5°C in 2023 only after the end of the next La Ninã, so probably in 2026 or 2027.
  - We might pass the threshold in some datasets and not others, but most datasets differ by just a constant
- At the current rate of temp increase (+0.25°C/decade), we’ll breach the hard +2°C threshold of the Paris agreement in 2043 already

On b-closures of polyhedra

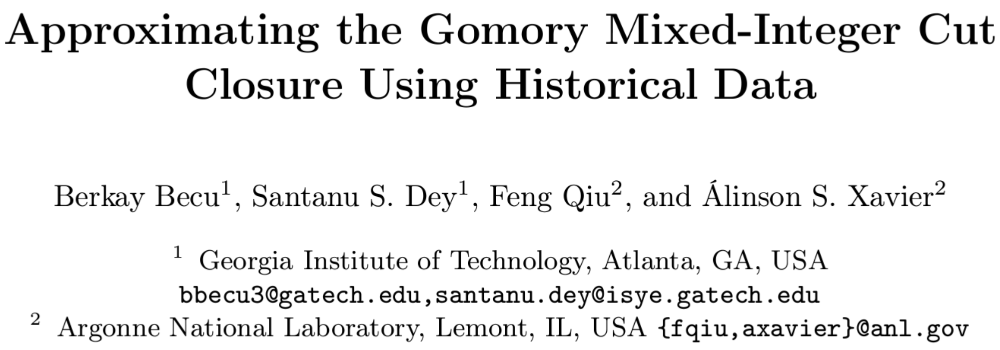
Diego Moran Ramirez
Rensselaer Polytechnic Institute

For $b \in \mathbb{Z}^n$ and **fixed** matrix A let:

$$P(b) = \{x \in \mathbb{R}^n \ : \ Ax \leq b\}.$$

This talk: "finiteness" of cutting planes closures and convex hulls for the associated infinite family of polyhedra.

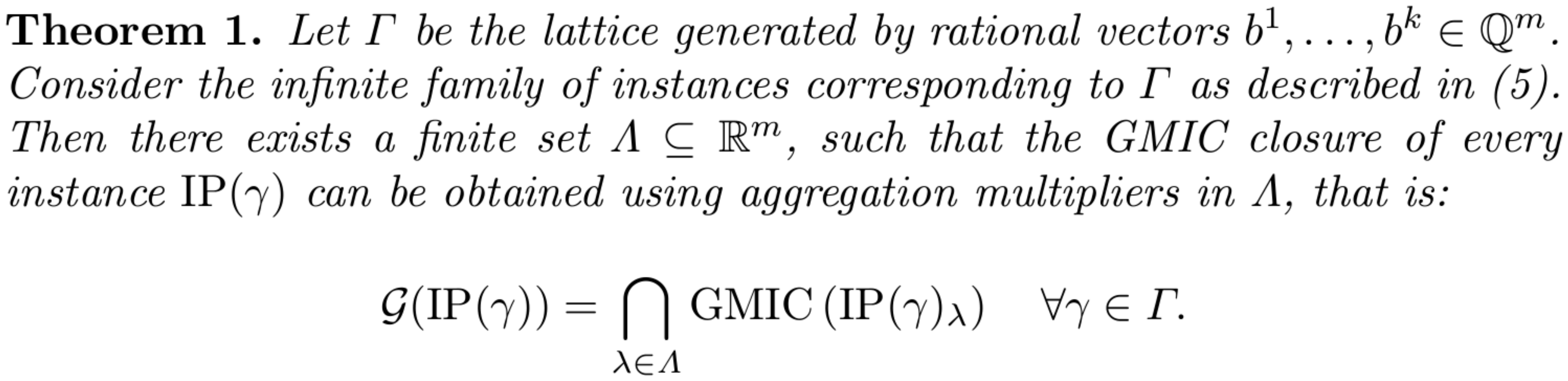
EXAMPLE CLOSURE



Approximating the Gomory Mixed-Integer Cut Closure Using Historical Data

Berkay Becu[1], Santanu S. Dey[1], Feng Qiu[2], and Álinson S. Xavier[2]

[1] Georgia Institute of Technology, Atlanta, GA, USA
`bbecu3@gatech.edu,santanu.dey@isye.gatech.edu`
[2] Argonne National Laboratory, Lemont, IL, USA `{fqiu,axavier}@anl.gov`



Theorem 1. *Let Γ be the lattice generated by rational vectors $b^1, \ldots, b^k \in \mathbb{Q}^m$. Consider the infinite family of instances corresponding to Γ as described in (5). Then there exists a finite set $\Lambda \subseteq \mathbb{R}^m$, such that the GMIC closure of every instance* $\mathrm{IP}(\gamma)$ *can be obtained using aggregation multipliers in Λ, that is:*

$$\mathcal{G}(\mathrm{IP}(\gamma)) = \bigcap_{\lambda \in \Lambda} \mathrm{GMIC}\left(\mathrm{IP}(\gamma)_\lambda\right) \quad \forall \gamma \in \Gamma.$$

There exists a finite set of aggregations that define the Gomory Mixed-Integer closure (GMIC) for **any** polyhedron in the infinite family.

EXAMPLE HULL

THE b-HULL OF AN INTEGER PROGRAM*

L.A. WOLSEY

CORE, Université Catholique de Louvain, Louvain-la-Neuve, Belgium

There exist functions $f^1, \ldots, f^T$ such that

$$P_I(b) = \left\{ x \in \mathbb{R}^n : Ax \leq b, \sum_{i=1}^{n} f^t(A^i) x_i \geq f^t(b) \right\}.$$

The integer hull of **any** polyhedron in the infinite family is defined by "finitely" many additional inequalities.

CORE RESULT: STRUCTURE OF POLYHEDRA IN THE FAMILY

There exist $b^1, \dots, b^T \in \mathbb{Z}^n$ s.t. for any $b \in \mathbb{Z}^n$:

$$P(b) = P(b^t) + z_b$$

for some $t = 1, \dots, T$ and $z_b \in \mathbb{Z}^n$.

Implications: we generalize Becu et al.'s and Wolsey's results for **any** reasonable closure and convex hull.

"FINITENESS" OF T-BRANCH CLOSURES (EX: T=1,SPLITS)

$$\mathbf{SC}(P(b)) = \bigcap_{S \in \mathcal{S}} \operatorname{conv}(P(b) \setminus S) = \bigcap_{S \in \mathcal{S}} \operatorname{conv}(P(b^t) \setminus (S - z_b)) + z_b$$

There is a finite list of split sets such that the split closure of **any** P(b) is defined by these splits translated by z_b.

Conclusion: finite up to translation.

FINITENESS OF K-LATTICE CLOSURES (L IS A MIXED-INTEGER LATTICE)

$$\mathbf{LC}(P(b)) = \bigcap_{L \in \mathcal{L}} \operatorname{conv}(P(b) \cap L) = \bigcap_{L \in \mathcal{L}} \operatorname{conv}(P(b^t) \cap L) + z_b$$

There is a finite list of lattices such that the lattice closure of **any** P(b) is defined by these lattices.

Conclusion: truly finitely defined.

THE INTEGER HULL IS “FINITELY” DEFINED

$$P_I(b) = P_I(b^t) + z_b$$

The integer hull of **any** polyhedron in the infinite family is defined by “finitely” many additional inequalities

Conclusion: finite up to r.h.s. translation.

Analyzing Election Data for Polarization: A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting
Note: a ballot with n candidates is a sorted list of a subset of candidates
Embed each ballot in a given metric space
Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head
Borda Pessimistic
Borda Average

3 OPT MODELS: Discrete k-Median Where Each Centroid is Embedding of a Cast Ballot
Discrete k-Median Where Each Centroid is Embedding of any Legal Ballot
Continuous k-Median

Analyzing Election Data for Polarization: A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting
Note: a ballot with n candidates is a sorted list of a subset of candidates
Embed each ballot in a given metric space
Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head – embed in "n choose 2" dimensions, where each dimension corresponds to a pair of candidates i < j and it is
+1 if i is preferred to j; -1 if j is preferred to i; 0 if neither i nor j listed

Borda Pessimistic – embed in n dimensions, where each dimension corresponds to a candidate i and if it is
n-j means that it is the jth preferred candidate; 0 if not on ballot

Borda Average – embed in n dimensions, where each dimension corresponds to a candidate i and if it is
n-j means that it is the jth preferred candidate; z if not on ballot
where z is set so that each embedded point sums to (n-1)+(n-2)+…+1

EXAMPLE: n=4 (candidates {1,2,3,4}) Ballot: 3>1 H2H=$(\underset{12}{1}, \underset{13}{-1}, \underset{14}{1}, \underset{23}{-1}, \underset{24}{0}, \underset{34}{1})$

Analyzing Election Data for Polarization: A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting

Note: a ballot with n candidates is a sorted list of a subset of candidates

Embed each ballot in a given metric space

Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head – embed in "n choose 2" dimensions, where each dimension corresponds to a pair of candidates i < j and it is

+1 if i is preferred to j; -1 if j is preferred to i; 0 if neither i nor j listed

Borda Pessimistic – embed in n dimensions, where each dimension corresponds to a candidate i and if it is

n-j means that it is the jth preferred candidate; 0 if not on ballot

Borda Average – embed in n dimensions, where each dimension corresponds to a candidate i and if it is

n-j means that it is the jth preferred candidate; z if not on ballot

where z is set so that each embedded point sums to (n-1)+(n-2)+…+1

EXAMPLE: n=4 (candidates {1,2,3,4}) Ballot: 3>1, BP = (2, 0, 3, 0)

Analyzing Election Data for Polarization:
A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting

Note: a ballot with n candidates is a sorted list of a subset of candidates

Embed each ballot in a given metric space

Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head – embed in “n choose 2” dimensions, where each dimension corresponds to a pair of candidates i < j and it is +1 if i is preferred to j; -1 if j is preferred to i; 0 if neither i nor j listed

Borda Pessimistic – embed in n dimensions, where each dimension corresponds to a candidate i and if it is n-j means that it is the jth preferred candidate; 0 if not on ballot

Borda Average – embed in n dimensions, where each dimension corresponds to a candidate i and if it is n-j means that it is the jth preferred candidate; z if not on ballot where z is set so that each embedded point sums to (n-1)+(n-2)+…+1

EXAMPLE: n=4 (candidates {1,2,3,4}) Ballot: 3>1; BA = (2, .5, 3, .5)

Analyzing Election Data for Polarization: A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting

Note: a ballot with n candidates is a sorted list of a subset of candidates

Embed each ballot in a given metric space

Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head

Borda Pessimistic

Borda Average

3 OPT MODELS: Discrete k-Median Where Each Centroid is Embedding of a Cast Ballot

minimize

$$\sum_{i \in C} \sum_{j \in D} w_i d(i,j) x_{ij}$$

subject to the constraints

$$\sum_{i \in C} y_i = k,$$

$$x_{ij} \leq y_i, \qquad \text{for each } i \in C,\ j \in D,$$

$$x_{ij} \in \{0,1\}, \quad y_i \in \{0,1\}, \qquad \text{for each } i \in C,\ j \in D.$$

C = D = embeddings of cast ballots d(i,j) = L1 distance between embeddings of i & j w(i) = # of ballots cast for i

Analyzing Election Data for Polarization:
A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting
Note: a ballot with n candidates is a sorted list of a subset of candidates
Embed each ballot in a given metric space
Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head
Borda Pessimistic
Borda Average

3 OPT MODELS: Discrete k-Median Where Each Centroid is Embedding of any Legal Ballot

PROBLEM!!
when n=15
there are too
many legal
ballots!

minimize

$$\sum_{i \in C} \sum_{j \in D} w_i d(i,j) x_{ij}$$

subject to the constraints

$$\sum_{i \in C} y_i = k,$$

$$x_{ij} \leq y_i, \qquad \text{for each } i \in C,\ j \in D,$$

$$x_{ij} \in \{0,1\}, \quad y_i \in \{0,1\}, \qquad \text{for each } i \in C,\ j \in D.$$

C = embeddings of cast ballots D = embeddings of legal ballots d(i,j) = L1 distance between embeddings of i & j w(i) = # ballots for

Analyzing Election Data for Polarization:
A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting

Note: a ballot with n candidates is a sorted list of a subset of candidates

Embed each ballot in a given metric space

Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head

Borda Pessimistic

Borda Average

3 OPT MODELS: Continuous k-Median w.r.t. L1 metric

Analyzing Election Data for Polarization:
A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting

Note: a ballot with n candidates is a sorted list of a subset of candidates

Embed each ballot in a given metric space

Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head

Borda Pessimistic

Borda Average

3 OPT MODELS: Continuous k-Median w.r.t. L1 metric

THIS IS A DISCRETE OPTIMIZATION PROBLEM!

IF A SET OF POINTS IS ASSIGNED TO SAME CLUSTER

(BY ASSIGNMENT VARIABLES)

OPTIMAL CENTROID IS:

FOR EACH DIMENSION

MEDIAN VALUE IN THAT DIMENSION

(I.E., A COUNTING PROBLEM)

Analyzing Election Data for Polarization:
A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting

Note: a ballot with n candidates is a sorted list of a subset of candidates

Embed each ballot in a given metric space

Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head

Borda Pessimistic

Borda Average

3 OPT MODELS: Discrete k-Median Where Each Centroid is Embedding of any Legal Ballot

CAN HANDLE LIKE CONTINUOUS k-MEDIAN!

“CONTINUOUS CENTROID” BUT CONSTRAINED TO BE LEGAL BALLOT

Analyzing Election Data for Polarization: A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting
Note: a ballot with n candidates is a sorted list of a subset of candidates
Embed each ballot in a given metric space
Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head
Borda Pessimistic
Borda Average

3 OPT MODELS: Discrete k-Median Where Each Centroid is Embedding of any Legal Ballot
CAN HANDLE LIKE CONTINUOUS k-MEDIAN!
“CONTINUOUS CENTROID” BUT CONSTRAINED TO BE LEGAL BALLOT

QUESTION: PREVIOUS USE OF THIS APPROACH?

Analyzing Election Data for Polarization: A Question About Formulations

Moon Duchin, David Shmoys, Kris Tapp

FRAMEWORK: Given full set of cast ballots in an election using rank choice voting
Note: a ballot with n candidates is a sorted list of a subset of candidates
Embed each ballot in a given metric space
Consider resulting clusters with respect to given optimization model

3 EMBEDDINGS: Head-to-Head
Borda Pessimistic
Borda Average

3 OPT MODELS: Discrete k-Median Where Each Centroid is Embedding of any Legal Ballot
CAN HANDLE LIKE CONTINUOUS k-MEDIAN!
“CONTINUOUS CENTROID” BUT CONSTRAINED TO BE LEGAL BALLOT

QUESTION: PREVIOUS USE OF THIS APPROACH?

ANALYSIS OF >1000 SCOTTISH ELECTIONS IN PROGRESS!

On fractional tree-independence-number-fragility

Andrea Munaro (University of Parma)

January 6th, 2025

Contains joint works with:

- E. Galby (Chalmers University of Technology) and S. Yang (Queen's University Belfast)
- C. Dallard (University of Fribourg), M. Milanič (University of Primorska) and S. Yang (Queen's University Belfast)

Claim: **Fractional tree-α-fragility** allows to **unify** and **extend** a large number of PTASes on both sparse and dense graph classes

Planar graphs and unit disk graphs

The following problems admit a PTAS:

1. Find max independent set in planar graph (Baker 1983)

$\rightsquigarrow$ **Layering technique**

2. Find max number of pairwise non-intersecting disks in a collection of unit disks in $\mathbb{R}^2$

(Hochbaum, Maass 1985)

$\rightsquigarrow$ **Shifting technique**

Planar graphs and unit disk graphs

The following problems admit a PTAS:

1. Find max independent set in planar graph (Baker 1983)

$\rightsquigarrow$ **Layering technique**

2. Find max number of pairwise non-intersecting disks in a collection of unit disks in $\mathbb{R}^2$ (Hochbaum, Maass 1985)

$\rightsquigarrow$ **Shifting technique**

Common theme: solve small subproblems via dynamic programming

Planar graphs and unit disk graphs

The following problems admit a PTAS:

1. Find max independent set in planar graph (Baker 1983)

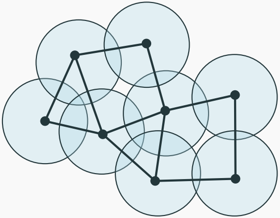
$\rightsquigarrow$ **Layering technique**

2. Find max number of pairwise non-intersecting disks in a collection of unit disks in $\mathbb{R}^2$ (Hochbaum, Maass 1985)

$\rightsquigarrow$ **Shifting technique**

Common theme: solve small subproblems via dynamic programming

Intersection graph $\rightsquigarrow$ jump from geometric to graph-theoretic world



Motivating questions

Is there any underlying graph-theoretic reason for the existence of PTASes for Independent Set on these seemingly unrelated graph classes?

Is there a general notion under which PTASes using Baker's technique can be obtained? (Grigoriev, Bodlaender 2007)

Baker's technique

Theorem (Vertex Decomposition Theorem, Baker 1983)

Given a planar graph G and k $\in \mathbb{N}$*,* $V(G)$ *can be partitioned into k (possibly empty) sets* $X_1, \ldots, X_k$ *in such a way that, for every* $i \in \{1, \ldots, k\}$*,* $\mathrm{tw}(G - X_i) = O(k)$*.*

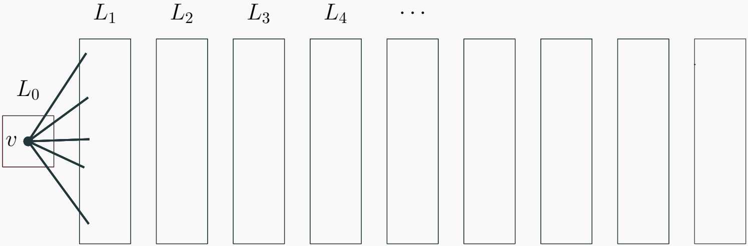
Moreover, such a partition together with tree decompositions of width $O(k)$ *of the respective subgraphs can be computed in polynomial time.*

Baker's technique

Theorem (Vertex Decomposition Theorem, Baker 1983)

Given a planar graph G and $k \in \mathbb{N}$, $V(G)$ *can be partitioned into k (possibly empty) sets* $X_1, \dots, X_k$ *in such a way that, for every* $i \in \{1, \dots, k\}$, $\mathrm{tw}(G - X_i) = O(k)$.

Moreover, such a partition together with tree decompositions of width $O(k)$ *of the respective subgraphs can be computed in polynomial time.*

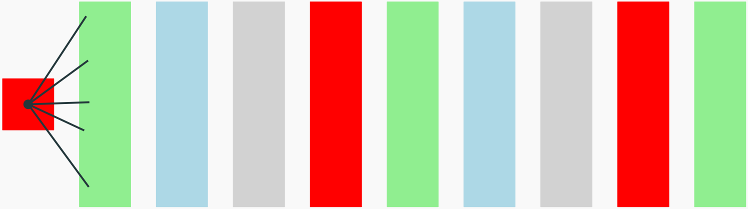


Baker's technique

Theorem (Vertex Decomposition Theorem, Baker 1983)

Given a planar graph G and $k \in \mathbb{N}$, $V(G)$ *can be partitioned into k (possibly empty) sets* $X_1, \ldots, X_k$ *in such a way that, for every* $i \in \{1, \ldots, k\}$, $\mathrm{tw}(G - X_i) = O(k)$.

Moreover, such a partition together with tree decompositions of width $O(k)$ *of the respective subgraphs can be computed in polynomial time.*

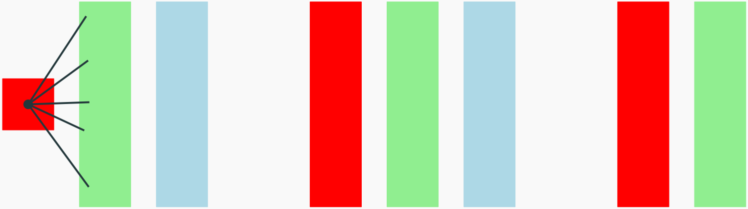


Baker's technique

Theorem (Vertex Decomposition Theorem, Baker 1983)

Given a planar graph G and $k \in \mathbb{N}$*,* $V(G)$ *can be partitioned into k (possibly empty) sets* $X_1, \ldots, X_k$ *in such a way that, for every* $i \in \{1, \ldots, k\}$*,* $\mathrm{tw}(G - X_i) = O(k)$*.*

Moreover, such a partition together with tree decompositions of width $O(k)$ *of the respective subgraphs can be computed in polynomial time.*



Beyond planar graphs: Proper minor-closed classes

VDTs exist for:

- graphs of bounded genus (Eppstein 2000)
- apex-minor-free graphs (Eppstein 2000)
- H-minor-free graphs (DeVos et al 2004; Demaine, Hajiaghayi, Kawarabayashi 2005)

$\rightsquigarrow$ Bidimensionality theory: link between PTASes and subexponential FPT algorithms

(Demaine, Hajiaghayi, 2005)

!!! VDTs for intersection graphs of geometric objects are something too strong to ask for

Beyond proper minor-closed classes: Efficient fractional tw-fragility

First relaxation of a VDT: Approximate partition of vertex set.

Beyond proper minor-closed classes: Efficient fractional tw-fragility

First relaxation of a VDT: Approximate partition of vertex set.

Definition (Dvořák 2016)

A graph class $\mathcal{G}$ is **efficiently fractionally tw-fragile** if $\exists\, f\colon \mathbb{N} \to \mathbb{N}$ and an algorithm that, $\forall\, r \in \mathbb{N}$ and $G \in \mathcal{G}$, returns in time poly$(|V(G)|)$ a collection of subsets $X_1, X_2, \ldots, X_m \subseteq V(G)$ such that each vertex of G belongs to at most m/r of the subsets and moreover, for every $i \in \{1, \ldots, m\}$, the algorithm also returns a tree decomposition of $G - X_i$ of width at most $f(r)$.

Beyond proper minor-closed classes: Efficient fractional tw-fragility

First relaxation of a VDT: Approximate partition of vertex set.

Definition (Dvořák 2016)

A graph class $\mathcal{G}$ is **efficiently fractionally tw-fragile** if $\exists\, f\colon \mathbb{N} \to \mathbb{N}$ and an algorithm that, $\forall\, r \in \mathbb{N}$ and $G \in \mathcal{G}$, returns in time poly$(|V(G)|)$ a collection of subsets $X_1, X_2, \ldots, X_m \subseteq V(G)$ such that each vertex of G belongs to at most m/r of the subsets and moreover, for every $i \in \{1, \ldots, m\}$, the algorithm also returns a tree decomposition of $G - X_i$ of width at most $f(r)$.

PTAS frameworks of maximization problems on efficiently fractionally tw-fragile classes

(Dvořák, Lahiri 2021; Dvořák 2022)

Beyond proper minor-closed classes: Efficient fractional tw-fragility

First relaxation of a VDT: Approximate partition of vertex set.

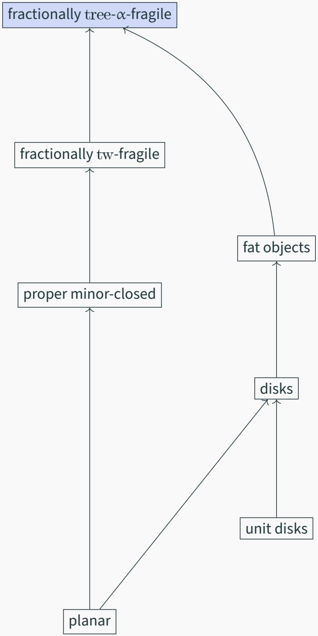
Definition (Dvořák 2016)

A graph class $\mathcal{G}$ is **efficiently fractionally tw-fragile** if $\exists\, f\colon \mathbb{N} \to \mathbb{N}$ and an algorithm that, $\forall\, r \in \mathbb{N}$ and $G \in \mathcal{G}$, returns in time poly$(|V(G)|)$ a collection of subsets $X_1, X_2, \ldots, X_m \subseteq V(G)$ such that each vertex of G belongs to at most m/r of the subsets and moreover, for every $i \in \{1, \ldots, m\}$, the algorithm also returns a tree decomposition of $G - X_i$ of width at most $f(r)$.

PTAS frameworks of maximization problems on efficiently fractionally tw-fragile classes

(Dvořák, Lahiri 2021; Dvořák 2022)

!!! Unit disk graphs are not fractionally tw-fragile (no sublinear separators)


fractionally tree-α-fragile
fractionally tw-fragile
fat objects
proper minor-closed
disks
unit disks
planar

Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$, for fixed d, is efficiently fractionally $\mathrm{tree}\text{-}\alpha$-fragile.

Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$, for fixed d, is efficiently fractionally $\mathrm{tree}\text{-}\alpha$-fragile.

A collection of objects is *c*-fat if it satisfies a sort of **“low-density property”**.

Slight generalization of (Chan 2003), implicitly used by (Har-Peled, Quanrud 2017).

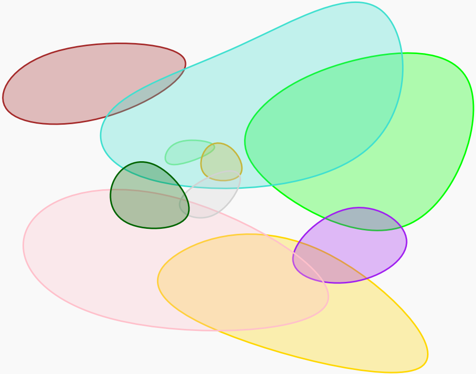
Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$*, for fixed d, is efficiently fractionally* $\mathrm{tree}\text{-}\alpha$*-fragile.*

A collection of objects is *c*-fat if it satisfies a sort of **“low-density property”**.

Slight generalization of (Chan 2003), implicitly used by (Har-Peled, Quanrud 2017).



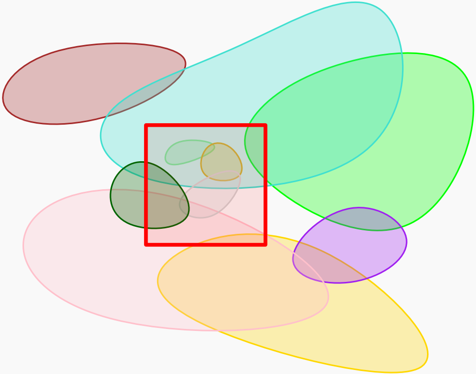
Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$*, for fixed d, is efficiently fractionally* $\mathrm{tree}\text{-}\alpha$*-fragile.*

A collection of objects is *c*-fat if it satisfies a sort of **“low-density property”**.

Slight generalization of (Chan 2003), implicitly used by (Har-Peled, Quanrud 2017).



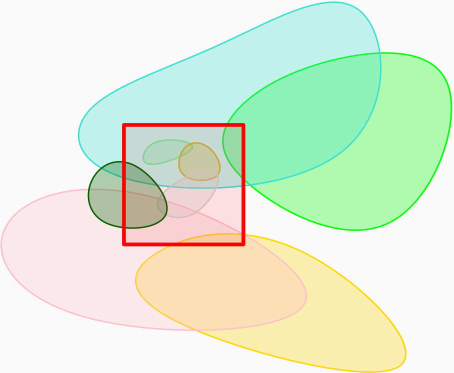
Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$*, for fixed d, is efficiently fractionally* $\mathrm{tree}\text{-}\alpha$*-fragile.*

A collection of objects is *c*-fat if it satisfies a sort of **“low-density property”**.

Slight generalization of (Chan 2003), implicitly used by (Har-Peled, Quanrud 2017).



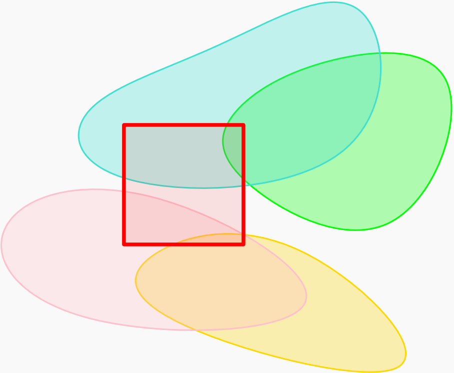
Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$*, for fixed d, is efficiently fractionally* $\mathrm{tree}\text{-}\alpha$*-fragile.*

A collection of objects is *c*-fat if it satisfies a sort of **“low-density property”**.

Slight generalization of (Chan 2003), implicitly used by (Har-Peled, Quanrud 2017).



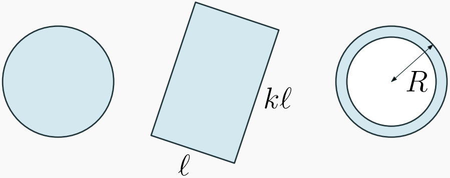
Second relaxation of a VDT: Replace tw with the more powerful $\mathrm{tree}\text{-}\alpha$.

Theorem

The class of intersection graphs of c-fat collections of objects in $\mathbb{R}^d$, for fixed d, is efficiently fractionally $\mathrm{tree}\text{-}\alpha$-fragile.

A collection of objects is *c*-fat if it satisfies a sort of **“low-density property”**.

Slight generalization of (Chan 2003), implicitly used by (Har-Peled, Quanrud 2017).



PTAS frameworks

Let ψ be a fixed CMSO_2 formula expressing an *h*-near-monotone property.

(c, h, ψ)-MAX WEIGHT INDUCED SUBGRAPH

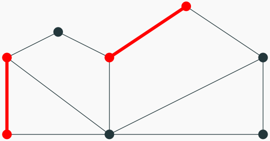
Input: A graph G equipped with a weight function $w\colon V(G) \to \mathbb{Q}_+$.
Task: Find a set $F \subseteq V(G)$ such that:

1. $G[F] \models \psi$,
2. $\omega(G[F]) \leqslant c$,
3. F is of maximum weight subject to the conditions above,

or conclude that no such set exists.

- MAX WEIGHT INDEPENDENT SET
- MAX WEIGHT INDUCED MATCHING
- MAX WEIGHT INDUCED FOREST
- MAX WEIGHT INDUCED PLANAR SUBGRAPH
- ...

Given a finite family $\mathcal{H}$ of connected non-null subgraphs of G, a **distance-d $\mathcal{H}$-packing** in G is a subfamily of subgraphs from $\mathcal{H}$ which are at pairwise distance at least d.



MAX WEIGHT DISTANCE-d PACKING

Input: Graph G, finite family $\mathcal{H} = \{H_j\}_{j \in J}$ of connected non-null subgraphs of G with $|V(H_j)| \leqslant h$ for each $j \in J$, weight function $w\colon J \to \mathbb{Q}_+$.

Task: Find a distance-d $\mathcal{H}$-packing in G of maximum weight.

Theorem

The following problems admit a PTAS on every efficiently fractionally tree-α-*fragile class:*

1. (c, h, ψ)-MAX WEIGHT INDUCED SUBGRAPH;
2. MAX WEIGHT DISTANCE-2 PACKING.

Theorem

MAX WEIGHT DISTANCE-p PACKING, *for even* $p \in \mathbb{N}$, *admits a PTAS on:*

3. *every class of intersection graphs of c-fat collections of objects in* $\mathbb{R}^d$, *for fixed d;*
4. *every class of bounded layered tree-independence number (provided that tree decomposition and layering are computable in poly-time).*

Theorem

The following problems admit a PTAS on every efficiently fractionally tree-α*-fragile class:*

1. (c, h, ψ)-MAX WEIGHT INDUCED SUBGRAPH;
2. MAX WEIGHT DISTANCE-2 PACKING.

Theorem

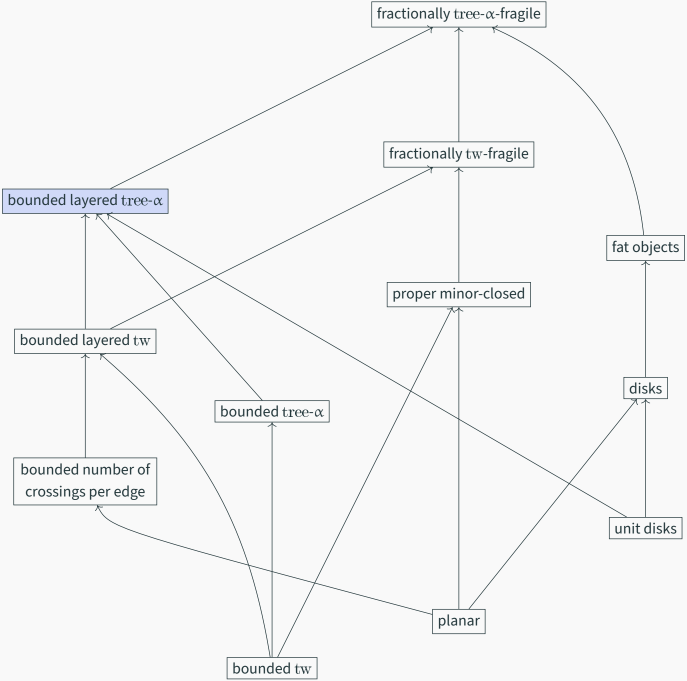
MAX WEIGHT DISTANCE-p PACKING, *for even* $p \in \mathbb{N}$*, admits a PTAS on:*

3. *every class of intersection graphs of c-fat collections of objects in* $\mathbb{R}^d$*, for fixed d;*
4. *every class of bounded layered tree-independence number (provided that tree decomposition and layering are computable in poly-time).*

Allow to generalize and extend several PTASes for:

- intersection graphs of fat objects (Chan 2003; Erlebach, Jansen, Seidel 2005)
- efficiently fractionally tw-fragile classes (Dvořák 2022; Dvořák, Lahiri 2021)
- intersection graphs of low-density objects (Har-Peled, Quanrud 2017)

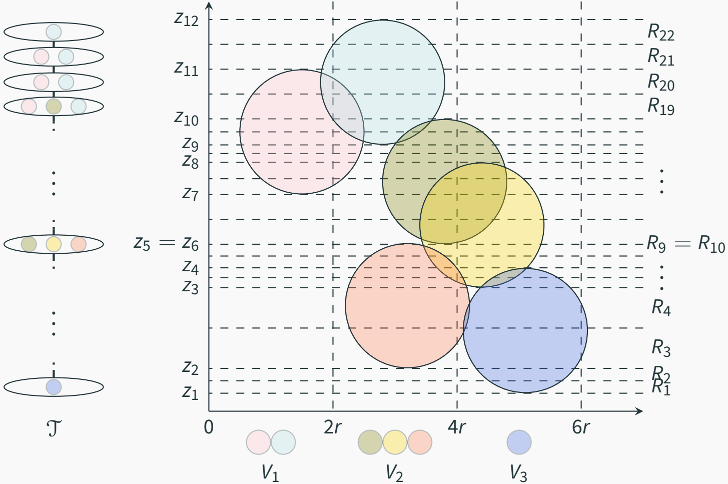
Complement PTASes for unweighted minimization problems on intersection graphs of fat objects (Dvořák, Lokshtanov, Panolan, Saurabh, Xue, Zehavi 2023)


fractionally tree-α-fragile
fractionally tw-fragile
bounded layered tree-α
fat objects
proper minor-closed
bounded layered tw
disks
bounded tree-α
bounded number of crossings per edge
unit disks
planar
bounded tw

Definition

The **layered independence number** of a tree decomposition $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ of a graph G is the minimum integer ℓ such that, for some layering $(V_0, V_1, \ldots)$ of G, and for each bag X_t and layer V_i, we have $\alpha(G[X_t \cap V_i]) \leqslant \ell$.

The **layered tree-independence number** of a graph G is the minimum layered independence number of a tree decomposition of G.



Bounded layered tree-independence number

Layered $\mathrm{tree}\text{-}\alpha$ generalizes layered tw (Dujmović, Morin, Wood 2017)

Classes with bounded layered $\mathrm{tree}\text{-}\alpha$:

- Graphs embeddable on a surface of b. genus with b. number of crossings per edge; $\rightsquigarrow$ b. layered tw (Dujmović, Eppstein, Wood 2017)
- (g, k)-string graphs $\rightsquigarrow$ b. layered tw (Dujmović, Joret, Morin, Norin, Wood 2018)
- Intersection graphs of k-similarly-sized c-fat families of objects in $\mathbb{R}^2$
- Unit width rectangle graphs
- g-map graphs
- Hyperbolic uniform disk graphs
- Spherical uniform disk graphs

Subexponential-time exact algorithms: Square-root phenomenon

Theorem (de Berg, Bodlaender, Kisfaludi-Bak, Marx, van der Zanden 2020)

There exist ETH-tight $2^{O(\sqrt{n})}$-time algorithms for the unweighted version of many problems on intersection graphs of similarly-sized fat objects in $\mathbb{R}^d$.

Key property: $\exists$ balanced separators that can be covered with $O(\sqrt{n})$ cliques.

However, very little is known about the weighted case.

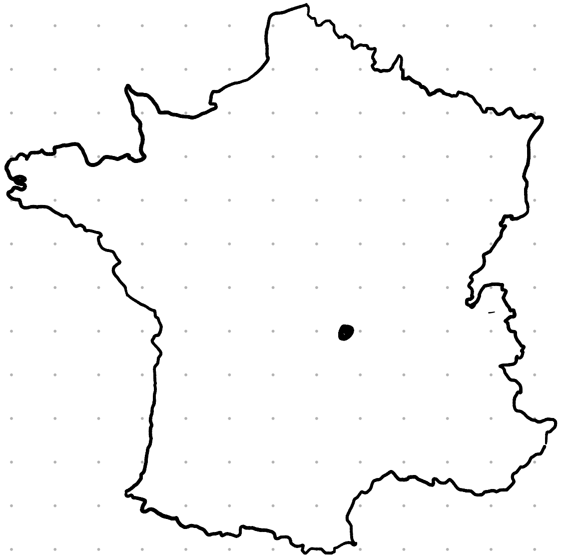
Key observation: Graph classes with bounded layered $\mathrm{tree}\text{-}\alpha$ have $O(\sqrt{n})$ $\mathrm{tree}\text{-}\alpha$.

Theorem

Let $\ell, d \in \mathbb{N}$ be fixed constants, with d even. Let G be a n-vertex graph for which we can compute, in time $\mathrm{poly}(n)$*, a tree decomposition and a layering witnessing layered tree-independence number at most ℓ. Then* MAX WEIGHT DISTANCE-d PACKING *can be solved in $2^{O(\sqrt{n}\log n)}$ time.*

Thank you!

MIP Workshop – Now in Europe!



Save The Date

July 1-3, Clermont-Ferrand

invited speakers + poster session

cheap registration + student housing

theory / computation / application

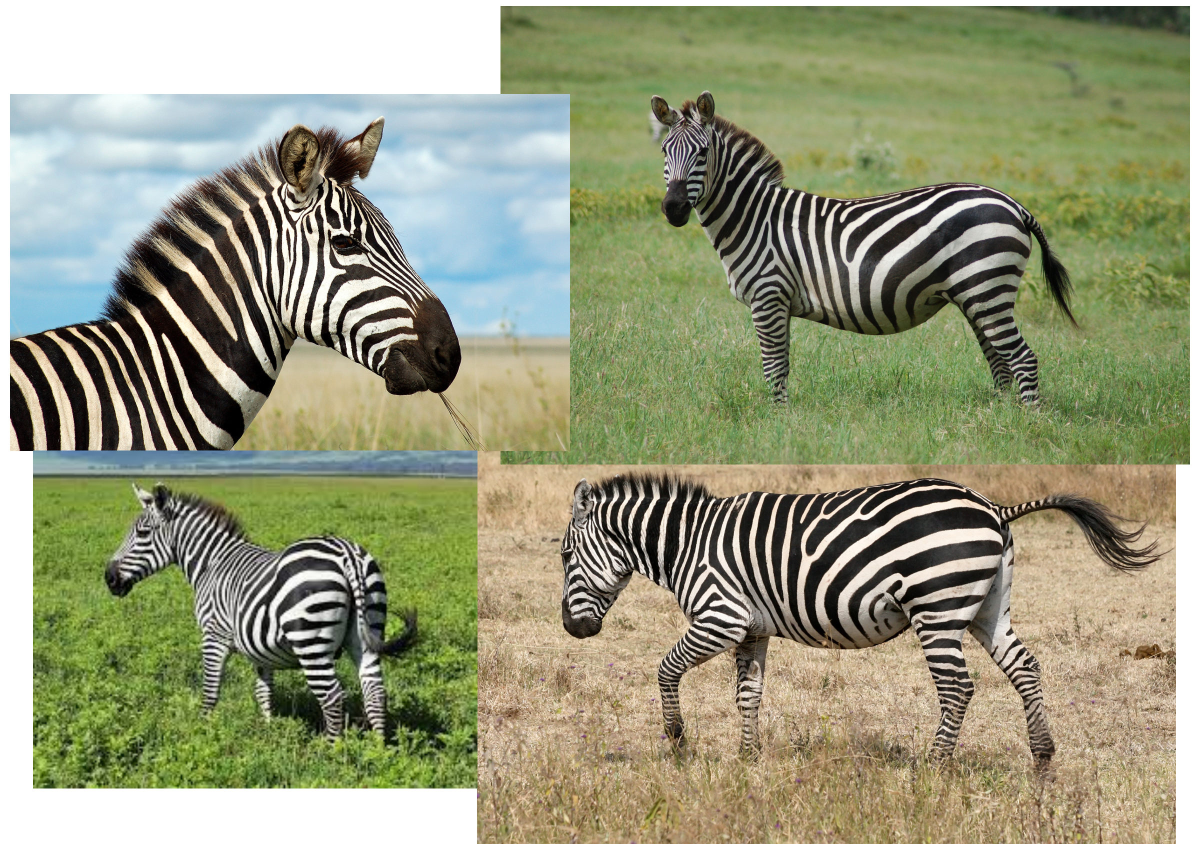
mixedinteger.org/EUROMIP/2025

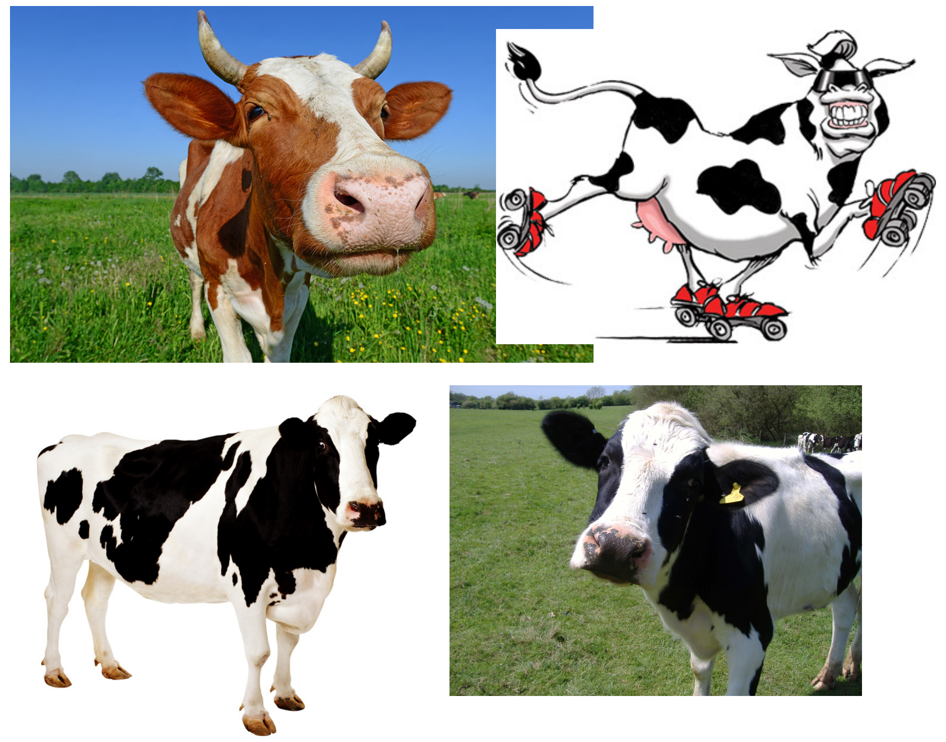
A $2+\epsilon$-Approximation Algorithm for Metric k-Median

Ola Svensson

EPFL

Joint work with Vincent Cohen-Addad, Fabrizio Grandoni, Euiwoong Lee, Chris Schwiegelshohn





k-Median

- Given a set of *n* points X and a distance metric $dist$
- Find a set of *k* centers $C \subseteq X$
- So that the distance of each point to its nearest center is minimized:

k-Median

- Given a set of *n* points X and a distance metric $dist$
- Find a set of *k* centers $C \subseteq X$
- So that the distance of each point to its nearest center is minimized:

$$\sum_{x \in X} \min_{c \in C} \operatorname{dist}(x, c)$$

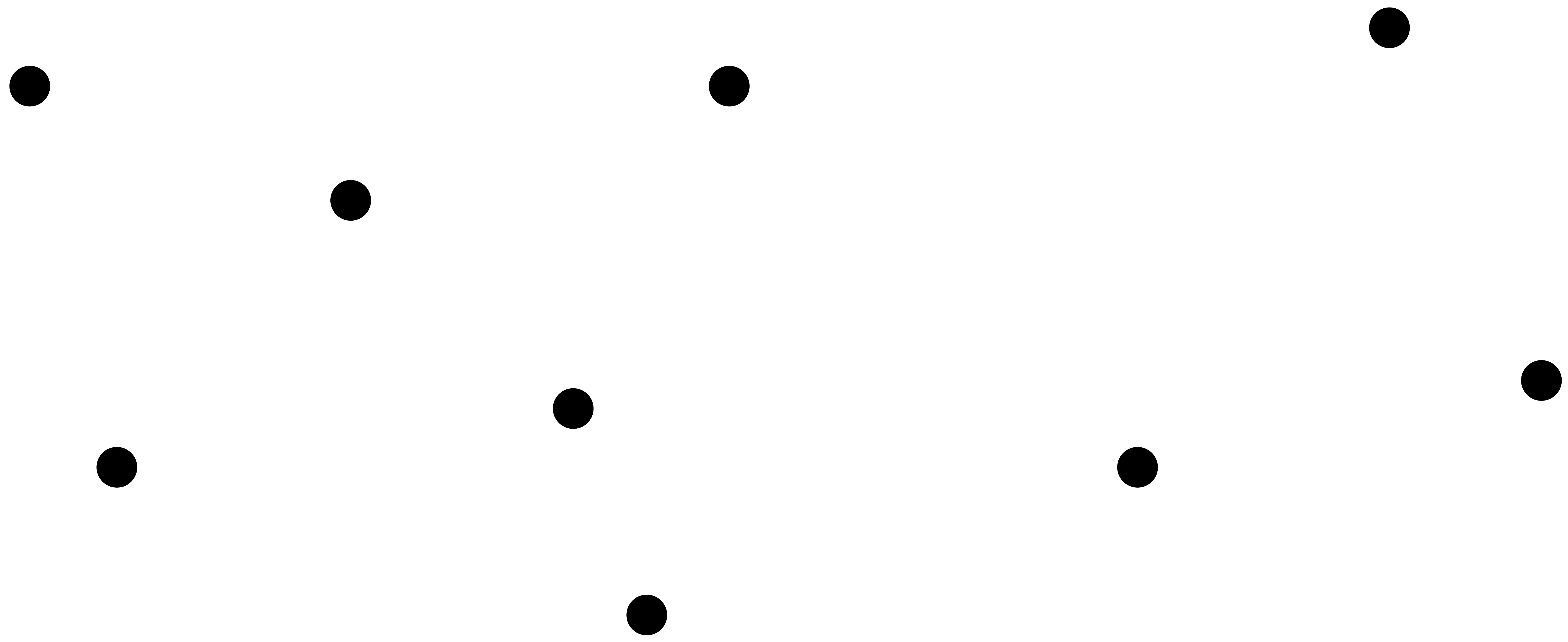
k-Median

- Given a set of *n* points X and a distance metric $dist$
- Find a set of *k* centers $C \subseteq X$
- So that the distance of each point to its nearest center is minimized:

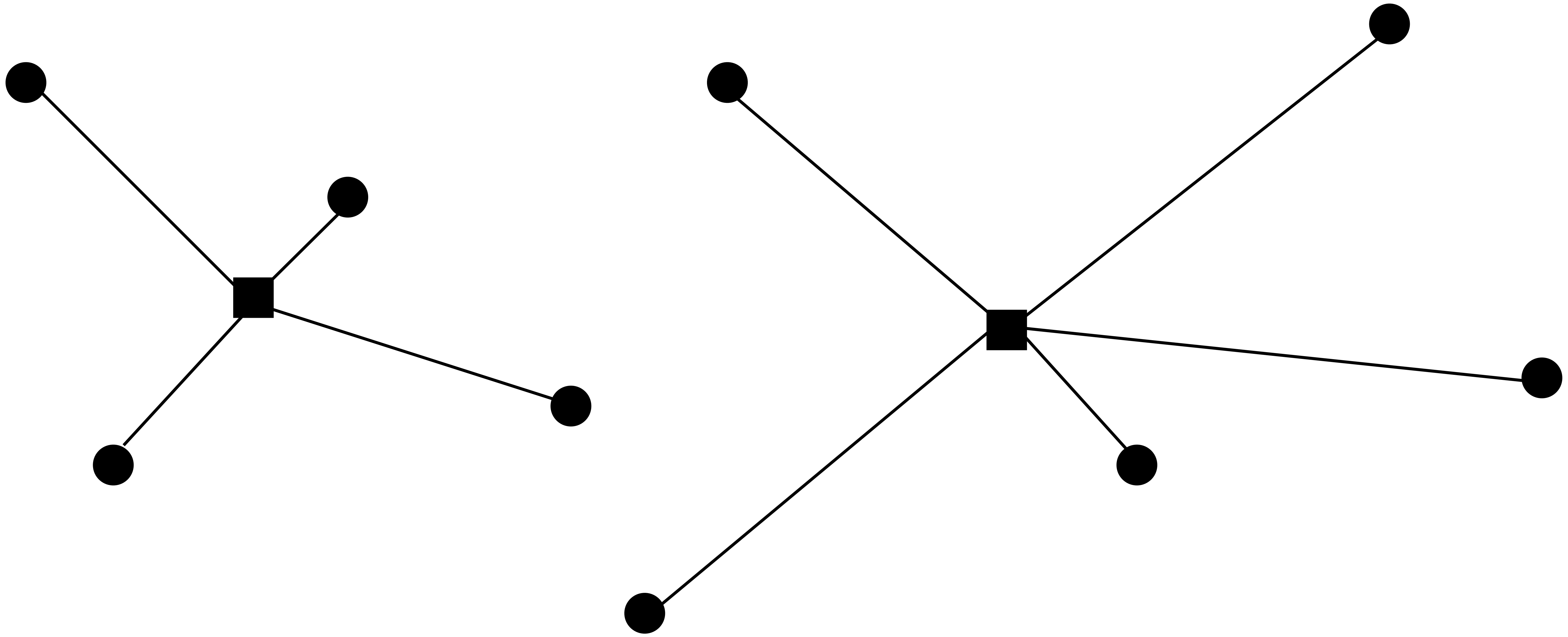
$$\sum_{x \in X} \min_{c \in C} \mathrm{dist}(x, c)$$

closest center to *x*

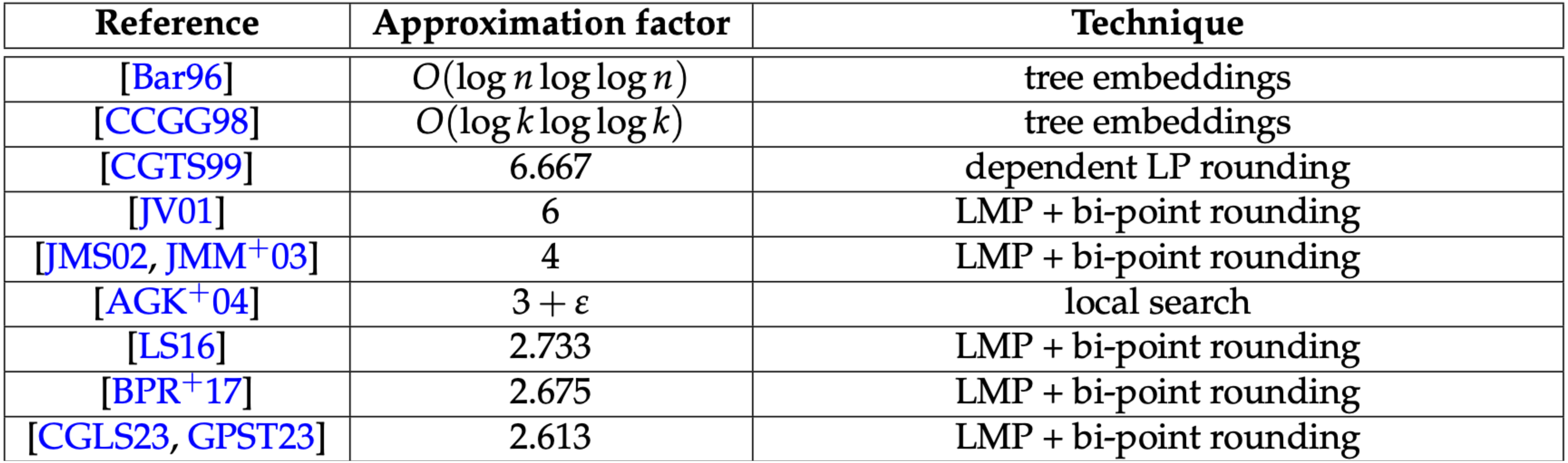
Example on the plane



Example on the plane

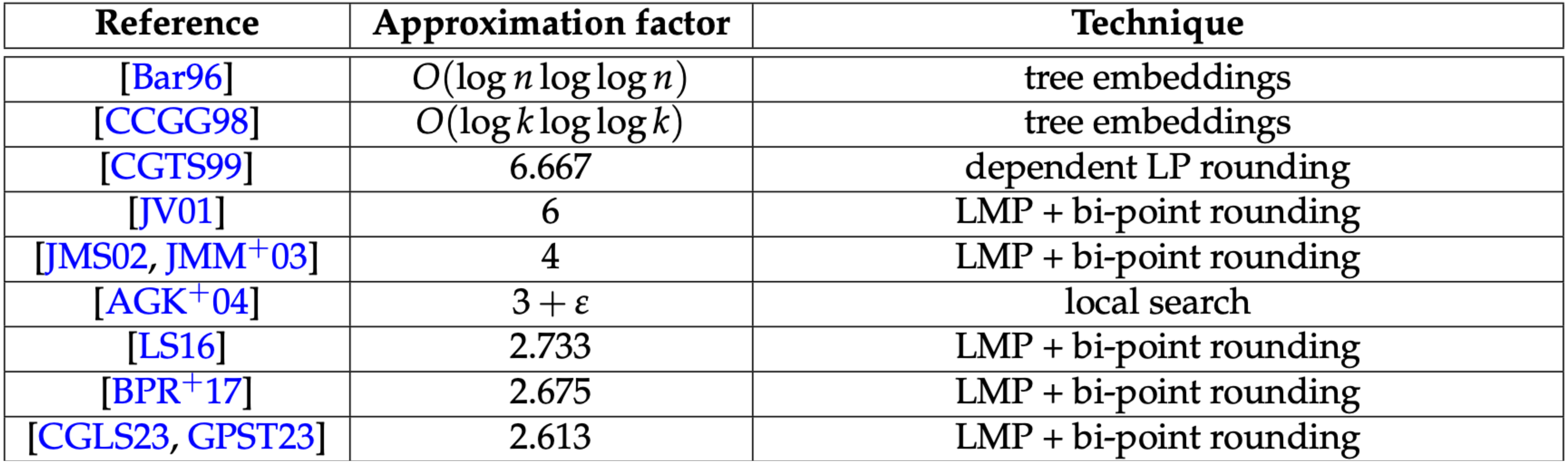


A classical question in combinatorial optimization



Reference	**Approximation factor**	**Technique**
[Bar96]	$O(\log n \log \log n)$	tree embeddings
[CCGG98]	$O(\log k \log \log k)$	tree embeddings
[CGTS99]	6.667	dependent LP rounding
[JV01]	6	LMP + bi-point rounding
[JMS02, JMM$^+$03]	4	LMP + bi-point rounding
[AGK$^+$04]	$3 + \varepsilon$	local search
[LS16]	2.733	LMP + bi-point rounding
[BPR$^+$17]	2.675	LMP + bi-point rounding
[CGLS23, GPST23]	2.613	LMP + bi-point rounding

A classical question in combinatorial optimization



Reference	Approximation factor	Technique
[Bar96]	$O(\log n \log\log n)$	tree embeddings
[CCGG98]	$O(\log k \log\log k)$	tree embeddings
[CGTS99]	6.667	dependent LP rounding
[JV01]	6	LMP + bi-point rounding
[JMS02, JMM+03]	4	LMP + bi-point rounding
[AGK+04]	$3+\varepsilon$	local search
[LS16]	2.733	LMP + bi-point rounding
[BPR+17]	2.675	LMP + bi-point rounding
[CGLS23, GPST23]	2.613	LMP + bi-point rounding

JMS02,JMM+03: A 2-approximation algorithm that opens k centers in expectation!

Theorem 1.1. *For every $\varepsilon > 0$, there is a randomized polynomial-time algorithm for k-median that returns a solution with cost at most $(2 + \varepsilon)$opt with high probability.*

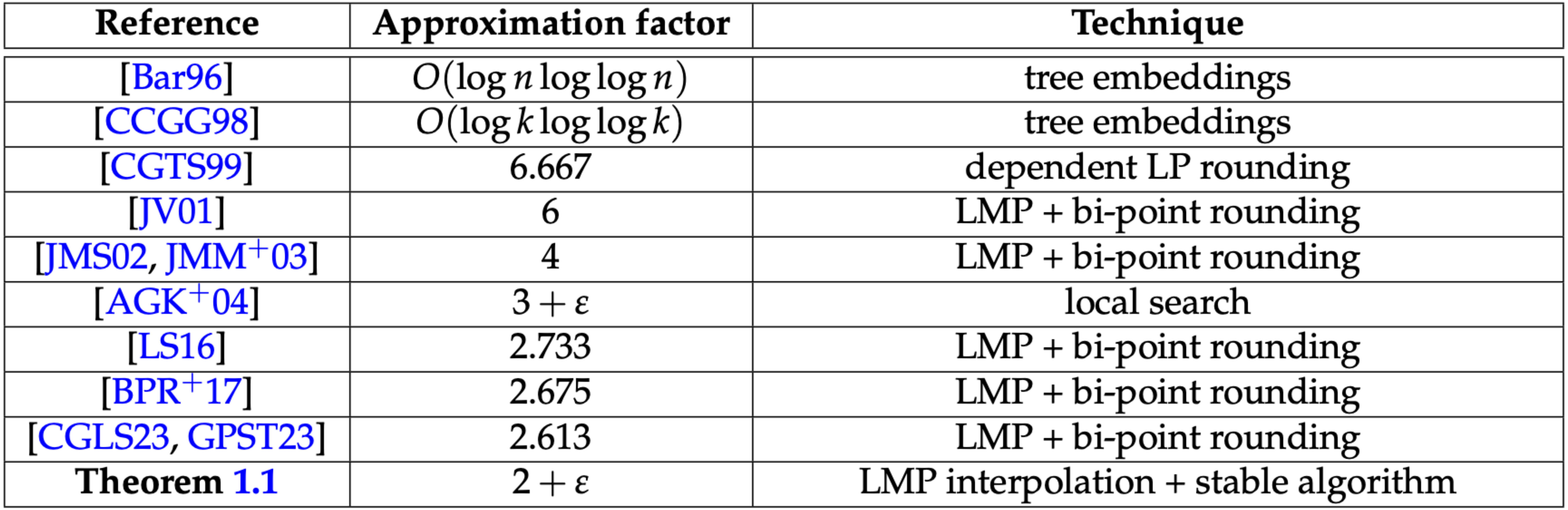
Theorem 1.1. *For every $\varepsilon > 0$, there is a randomized polynomial-time algorithm for k-median that returns a solution with cost at most $(2 + \varepsilon)$opt with high probability.*

Theorem 1.2. *For every $\varepsilon \in (0, 1/6)$, there is a polynomial-time algorithm for k-median that returns a solution containing at most $k + O(\log n/\varepsilon^2)$ many centers and of cost at most $(2 + \varepsilon)$opt.*

+

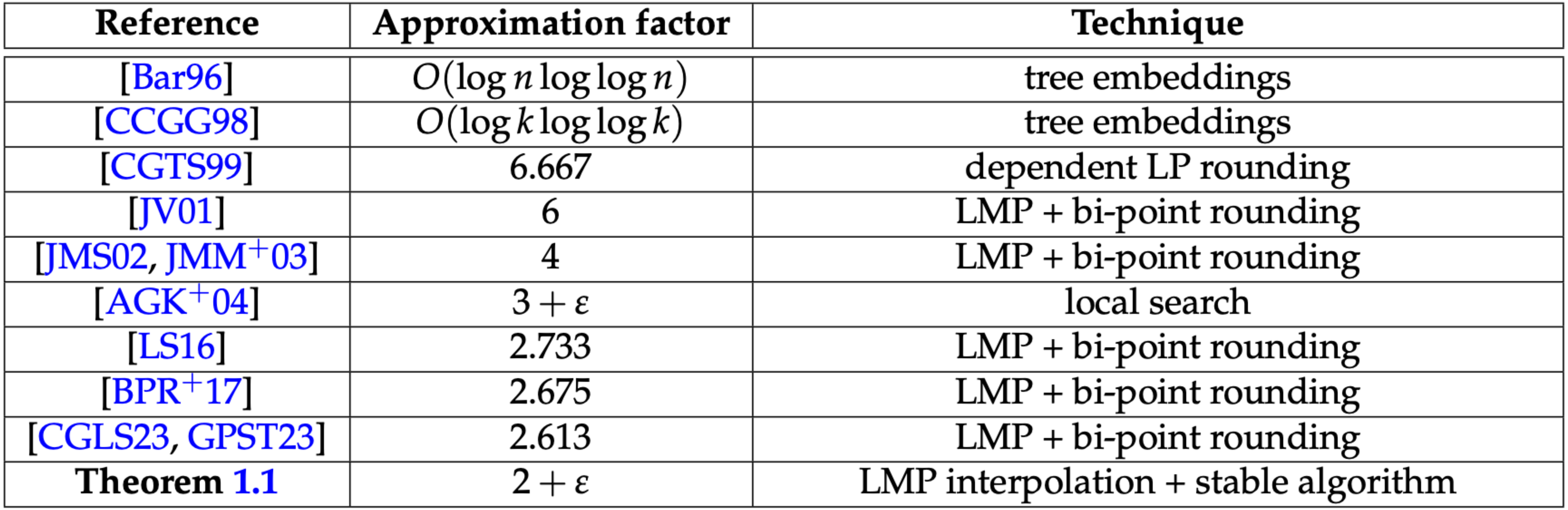
Theorem 1.3. *For any $\varepsilon, \zeta > 0$, there exists a randomized polynomial-time algorithm that, given a $(\zeta / \log n)$-stable k-median instance, returns a solution of cost at most $(2 + O(\varepsilon))$opt with high probability.*

Closing the gap



Reference	**Approximation factor**	**Technique**
[Bar96]	$O(\log n \log\log n)$	tree embeddings
[CCGG98]	$O(\log k \log\log k)$	tree embeddings
[CGTS99]	6.667	dependent LP rounding
[JV01]	6	LMP + bi-point rounding
[JMS02, JMM$^+$03]	4	LMP + bi-point rounding
[AGK$^+$04]	$3+\varepsilon$	local search
[LS16]	2.733	LMP + bi-point rounding
[BPR$^+$17]	2.675	LMP + bi-point rounding
[CGLS23, GPST23]	2.613	LMP + bi-point rounding
Theorem 1.1	$2+\varepsilon$	LMP interpolation + stable algorithm

Closing the gap



Reference	Approximation factor	Technique
[Bar96]	$O(\log n \log\log n)$	tree embeddings
[CCGG98]	$O(\log k \log\log k)$	tree embeddings
[CGTS99]	6.667	dependent LP rounding
[JV01]	6	LMP + bi-point rounding
[JMS02, JMM^{+}03]	4	LMP + bi-point rounding
[AGK^{+}04]	$3+\varepsilon$	local search
[LS16]	2.733	LMP + bi-point rounding
[BPR^{+}17]	2.675	LMP + bi-point rounding
[CGLS23, GPST23]	2.613	LMP + bi-point rounding
Theorem 1.1	$2+\varepsilon$	LMP interpolation + stable algorithm

NP-hard to do better than 1+2/e

Integrality gap of standard LP is at least 2

We are looking for postdocs

Soft deadline January 20

EPFL, the Swiss Federal Institute of Technology in Lausanne, is one of the most dynamic university campuses in Europe and ranks among the top 20 universities worldwide. The EPFL employs more than 6,500 people supporting the three main missions of the institutions: education, research and innovation. The EPFL campus offers an exceptional working environment at the heart of a community of more than 17,000 people, including over 12,500 students and 4,000 researchers from more than 120 different countries.

Postdoctoral position in the Theory Group

Mission
Your mission is to perform research within the theory of computation group at EPFL. Specific areas of research include combinatorial optimization, approximation algorithms, online algorithms, theoretical foundations of big data analysis (sublinear algorithms, streaming, etc.), (quantum) computational complexity (proof complexity, communication complexity, etc.), and quantum cryptography.

Brilliant and vibrant theory group that covers complexity, quantum, algorithms, game theory, theory of ML ... and includes faculty E. Abbe, A. Chiesa, Andres Christi, F. Eisenbrand, M. Göös, M. Kapralov, O. Svensson, and last but not least T. Vidick.

Set Covering and the Replication Conjecture

Gerard Cornuejols*[1]

[1]Carnegie Mellon University – United States

Abstract

Analogous to perfection in antiblocking theory is the notion of ”packing property” in blocking theory. A key insight on perfect graphs is the famous replication lemma proved by Laci Lovasz in 1972. In 1993, Michele Conforti and I proposed an analogous replication conjecture when the packing property holds. This conjecture is still open. This talk covers some recent developments related to the replication conjecture.

---

*Speaker

Benchmarking challenges for quantum optimization: the intractable decathlon

Many authors; presented by Giacomo Nannicini

University of Southern California

January 6–10, 2025

Using quantum computers for optimization

State of quantum optimization research

Continuous optimization:

- Very active.
- Rigorous complexity analysis.
- Requires fault tolerance.

Discrete optimization:

- Few rigorous complexity analyses.
- Plenty of heuristics.
- Many algorithms are designed for noisy devices and have been numerically tested already.

What discrete optimization problems should we use to benchmark the performance of quantum optimization algorithms?


THE INTRACTABLE DECAHHLON
10 PROBLEMS. NO MERCY.
2. LOW-AUTOCORLATION
3. BINARY SPACHCTION
3. PORTFOLIY OPTRIZTION
0. NOLICYCLE DOSIGN
MARKETSHARE
MARKETSHARE.
LOW-AUTOCORLATION
BINARY TORMAITIEN
SENQENTLES
NEHICLY DESIGN
STABLE SET
BTEINE ROUTTI
BINARY SPACKLING
PORTTOOLY DOSIGN.
STABLE SET
VEHICLY ROSIGNG
STABLE SET
SPORTS ORESGNN

The intractable decathlon

No.	Name	Description
1	Marketshare	Multi-dimensional subset-sum
2	LABS	Low autocorrelation binary sequences
3	Birkhoff	Birkhoff decomposition
4	Steiner	Steiner tree packing in graphs (VLSI Design/Wire Routing)
5	Sports	Sports Tournament Scheduling (STS)
6	Portfolio	Multi-period Portfolio optimization with transaction costs
7	Stable-Set	Unweighted Maximum Independent Set (MIS)
8	Network	Communications Network design problem
9	Routing	Capacitated vehicle routing problem (CVRP)
10	Topology	Graph topology design (Node-Degree-Diameter problem)

These problems have varying characteristics. All of them are extremely difficult for exact classical algorithms already at system sizes $\approx 10^2$ to 10^5.

"Quantum optimization benchmarking challenges" repo

We provide a repository with instances, guidelines, pointers to state-of-the-art algorithms, baseline results, updated results (e.g., best solutions, gap), ensuring:

- Comparability of used methods;
- Reproducibility of the respective solutions;
- Trackability of algorithmic and hardware improvements.

The benchmark is model-independent: we do not prescribe the model used to solve the problem.

Repository: `https://git.zib.de/bzfkocht/qbench/`. Out soon!

These problems cannot be solved with current technology.
We need your help to push the boundary of what optimization algorithms can do!

Quotient sparsification for submodular functions

Kent Quanrud*

Abstract

Graph sparsification has been an important topic with many structural and algorithmic consequences. Recently hypergraph sparsification has come to the fore and has seen exciting progress. In this paper we take a fresh perspective and show that they can be both be derived as corollaries of a general theorem on sparsifying matroids and monotone submodular functions.

Faster single-source shortest paths with negative real weights via proper hop distance*

Yufan Huang Peter Jin Kent Quanrud

December 10, 2024

Abstract

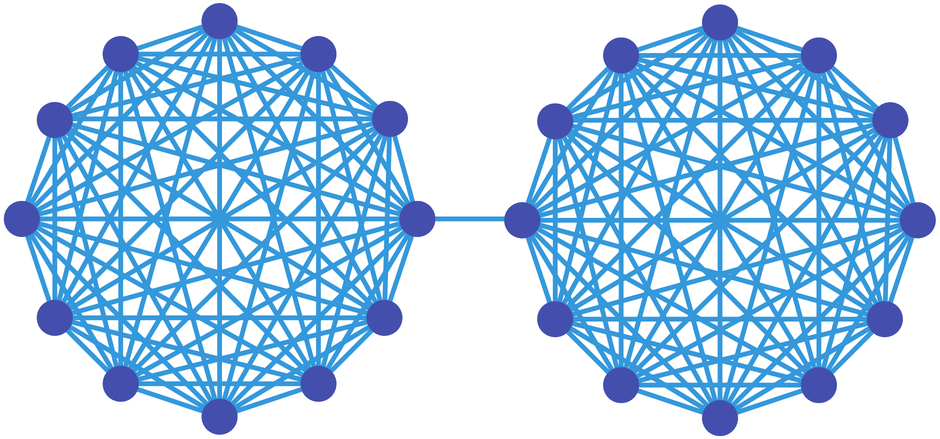
The textbook algorithm for single-source shortest paths with real-valued edge weights runs in $O(mn)$ time on a graph with m edges and n vertices. A recent breakthrough algorithm by Fineman [Fin24] takes $\tilde{O}\left(mn^{8/9}\right)$ randomized time. We present an $\tilde{O}\left(mn^{4/5}\right)$ randomized time algorithm building on ideas from [Fin24].

Graph cut sparsification

Input: (undirected) $G=(V,E)$

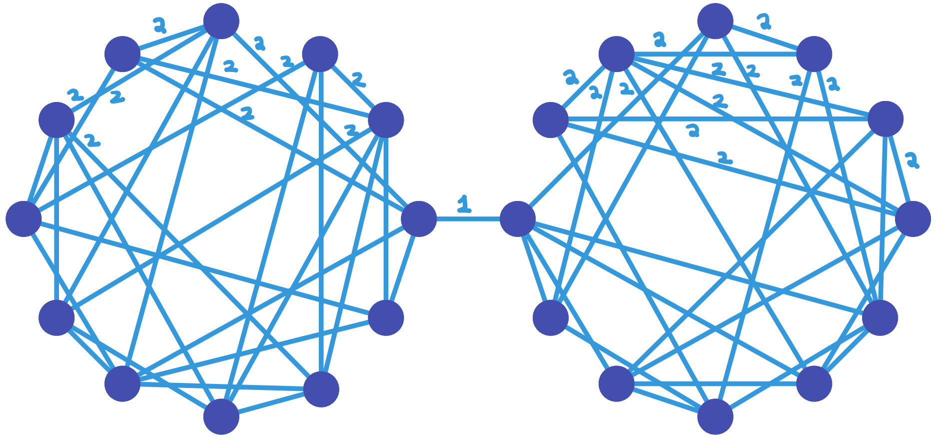
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all cuts have similar weight as in G

Benczúr, Karger (2002):

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all cuts: $(1-\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e) \le \sum_{e\in\partial(S)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e)$

Also: spectral [ST04, SS11], $|\hat{E}|=O(n/\varepsilon^2)$ [BSS12], [FHHP11], ~

Hypergraph (2-)cut sparsification

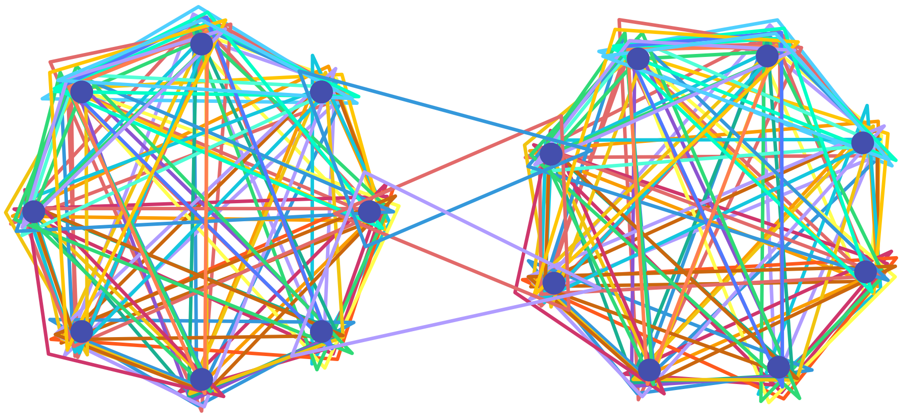
[KK15, CX18, SY19, BST19, CKN20, KKTY21, L22, JLS22, JLLS23]

$p =$ total size $= \sum_{e\in E} |e|$

Input: Hypergraph $G=(V,E)$

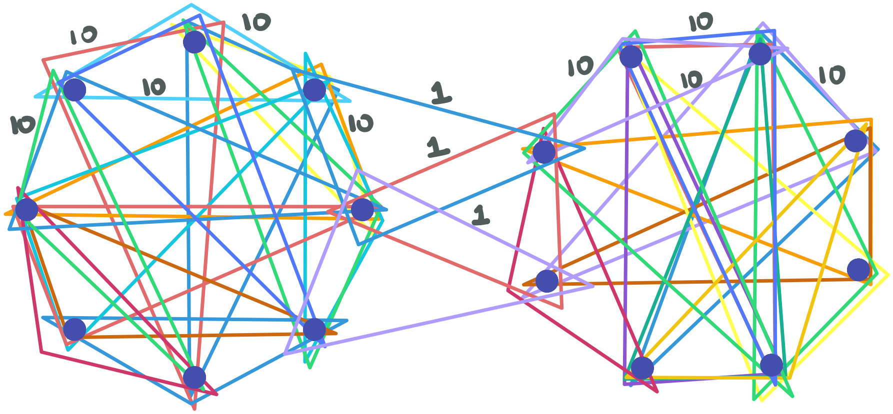
$n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



s.t. (a) $|\tilde{E}|$ small

(b) all 2-cuts have similar weight as in G

Chen, Khanna, Nagda 2020:

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all cuts: $(1-\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e) \le \sum_{e\in\partial(S)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S)}\tilde{w}(e)$

- Also: "spectral extensions, sums of symm. submodular fun,

(by others)

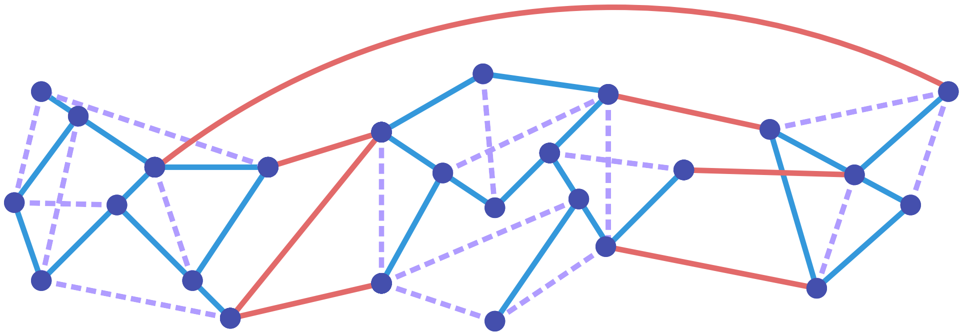
Quotients

Q is a quotient if $\overline{Q} = \mathcal{N} \setminus Q$ is closed

i.e., if $Q = \mathcal{N} \setminus \text{span}(S)$ for some $S \subseteq \mathcal{N}$.

e.g. for graphic matroid:

$\text{span}(S) = \{$all edges connected by $S\}$



$Q = E \setminus \text{span}(S)$

= edges cut by conn. comp. of S

$\mathcal{M} = (\overset{\text{"groundset"}}{\mathcal{N}}, \underset{\text{"independent sets"}}{\mathcal{I}})$

1. $\emptyset \in \mathcal{I}$
2. $S \subseteq T, T \in \mathcal{I} \Rightarrow S \in \mathcal{I}$
3. $S, T \in \mathcal{I}, |S| < |T| \Rightarrow$ $e \in T \setminus S$ s.t. $S+e \in \mathcal{I}$

maxim**al** = maxim**um**

$\text{rank}(S) = \max\{|I| : I \subseteq S, I \in \mathcal{I}\}$

"rank of $\mathcal{M}$" = $\text{rank}(\mathcal{N})$

Properties of f = rank:

- monotone: $f(S) \leq f(T)$ for $S \subseteq T$
- submodular: if $S \subseteq T$, and $e \in \mathcal{N}$,

  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$ and $e \in \mathcal{N}$,

  $f(e|T) = 0$ or $f(e|T) \geq 1$

$\text{span}(S) = \{e \in \mathcal{N} : f(S+e) = f(S)\}$ (including S)

S is "closed" if $S = \text{span}(S)$

Graphic Matroid

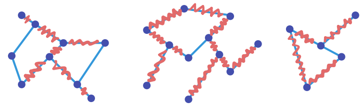
(i.e., forests)

fix $G = (V, E)$ (undirected)

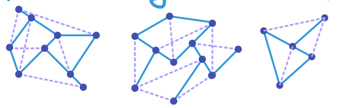
$\mathcal{N} = E$

$\mathcal{I} = \{F \subseteq E : F$ is a forest$\}$

$\text{rank}(S) = n - (\#\text{CC of } S)$



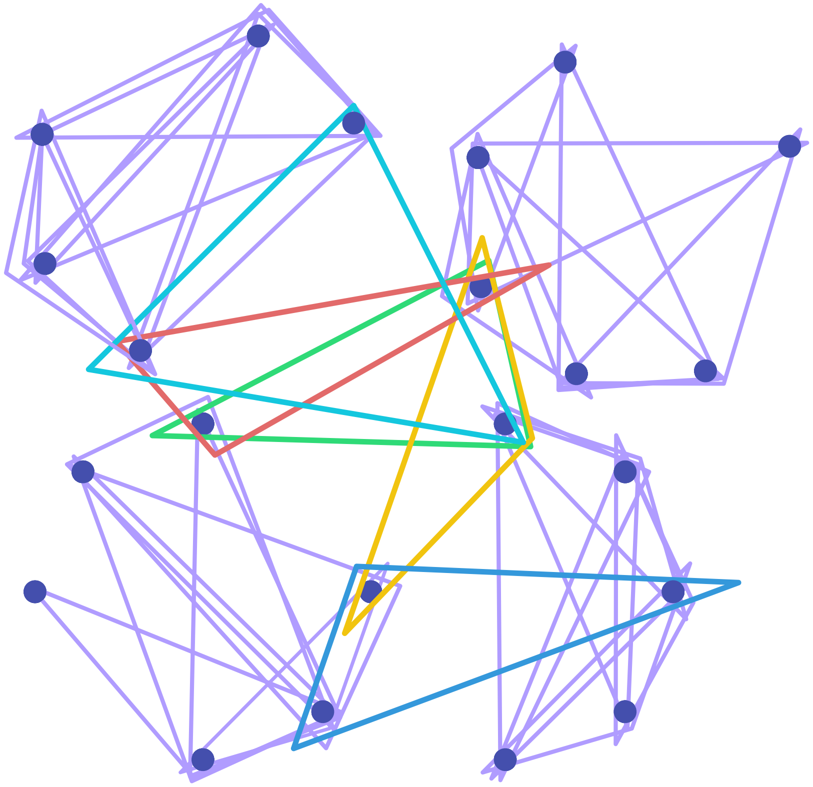
$\text{span}(S) = \{$edges connected by $S\}$



$f(S) = n - \left(\begin{array}{c} \text{\# connected} \\ \text{components of } S \end{array} \right)$

Hypergraphic polymatroid function

Quotient $Q = E \setminus \text{span}(S) = \left\{ \begin{array}{c} \text{edges cut by} \\ \text{conn. comp. of } S \end{array} \right\}$



quotients = k-cuts (for varying k)

k-cuts include 2-cuts

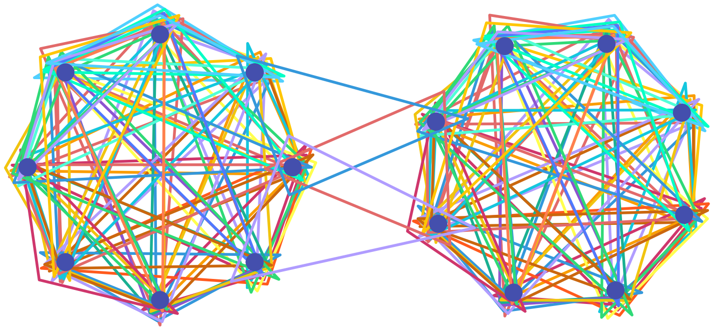
unlike graphs,

k-cut ≠ (half of) sum of 2-cuts over conn. comp.

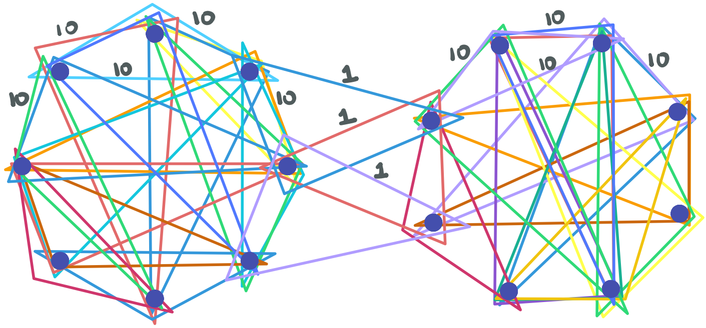
Submodular quotient sparsification

Input: $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (normalized monotone submodular)

weights $w: \mathcal{N} \to \mathbb{R}_{>0}$



Goal: $\tilde{w}: \mathcal{N} \to \mathbb{R}_{>0}$



s.t. (a) support($\tilde{w}$) small

(b) all quotients have similar weight as w/ w.

Let $f: 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be:

- monotone: $S \subseteq T \Rightarrow f(S) \leq f(T)$
- submodular: if $S \subseteq T$, $e \in \mathcal{N}$,
  $\underbrace{f(e|T)}_{f(T+e)-f(T)} \leq \underbrace{f(e|S)}_{f(S+e)-f(S)}$ "decreasing marginal returns"
- "normalized": for $T \subseteq \mathcal{N}$, $e \in \mathcal{N}$,
  $f(e|T) = 0$ or $f(e|T) \geq 1$
- $\text{span}_f(S) = \{ \overset{\text{(including } S)}{e \in \mathcal{N}} : f(e|S) = 0\}$
- S closed if $S = \text{span}_f(S)$
- Q quotient if $\bar{Q} = \mathcal{N} \setminus Q$ closed
  i.e., $Q = \mathcal{N} \setminus \text{span}(S)$ for some S
- "rank of f" $= f(\mathcal{N})$

Theorem

let $r = f(\mathcal{N})$.

- $|\text{support}(\tilde{w})| = O(r \log(rn)/\varepsilon^2)$
- $(1+\varepsilon)$-APX all quotients: $(1-\varepsilon)\, w(Q) \leq \tilde{w}(Q) \leq (1+\varepsilon)\, w(Q)$

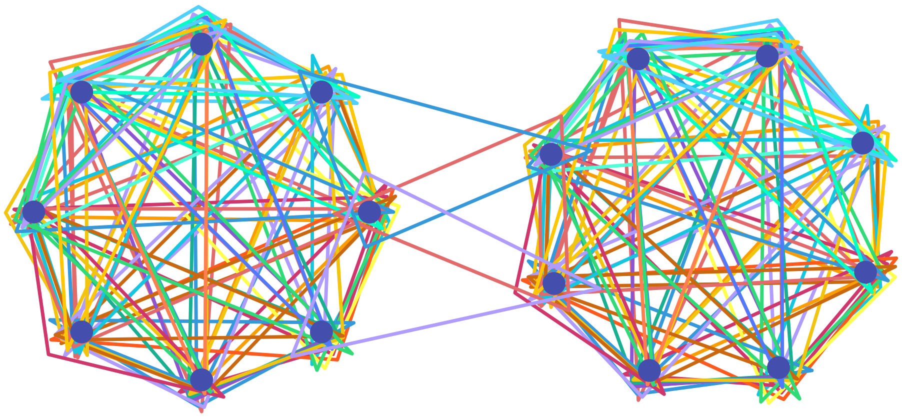
(w/ high prob., rand. poly time, w/ oracle access to f)

Hypergraph k-cut sparsification

p = total size = $\sum_{e\in E} |e|$

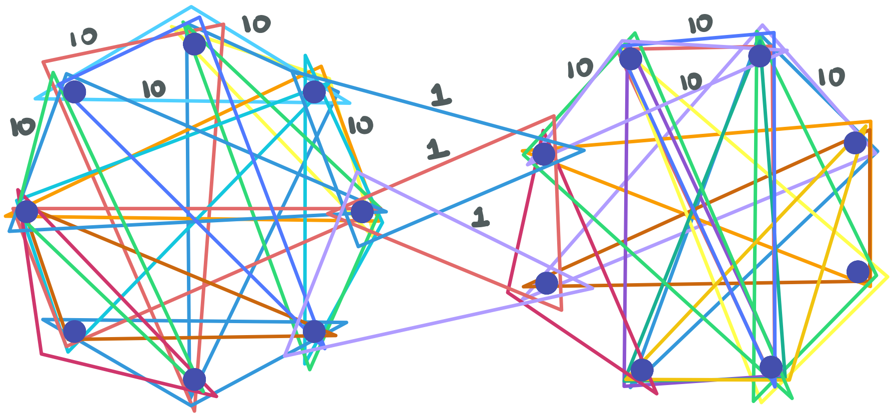
Input: Hypergraph $G=(V,E)$ $n=|V|$ $m=|E|$

$w(e)>0$ for $e\in E$



Goal: subgraph $\tilde{G}=(V,\tilde{E})$

$\tilde{w}(e)>0$ for $e\in\tilde{E}$



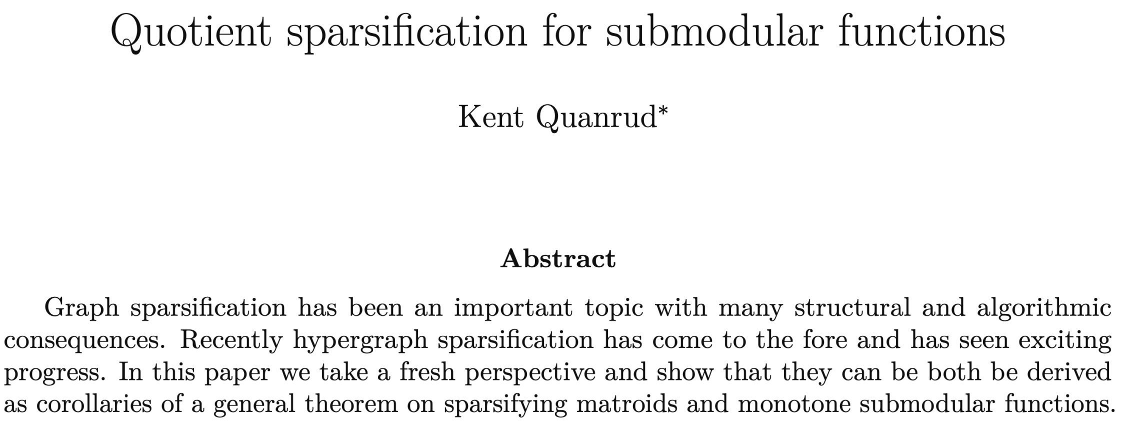
s.t. (a) $|\tilde{E}|$ small

(b) all k-cuts have similar weight as in G

Theorem

- $|\tilde{E}| = O(n\log(n)/\varepsilon^2)$
- $(1+\varepsilon)$-APX for all k-cuts
- $(1-\varepsilon)\sum_{e\in\partial(S_1,\dots,S_k)} \tilde{w}(e) \le \sum_{e\in\partial(S_1,\dots,S_k)} w(e) \le (1+\varepsilon)\sum_{e\in\partial(S_1,\dots,S_k)} \tilde{w}(e)$

(w/ high prob., in randomized $\tilde{O}(p)$ time)

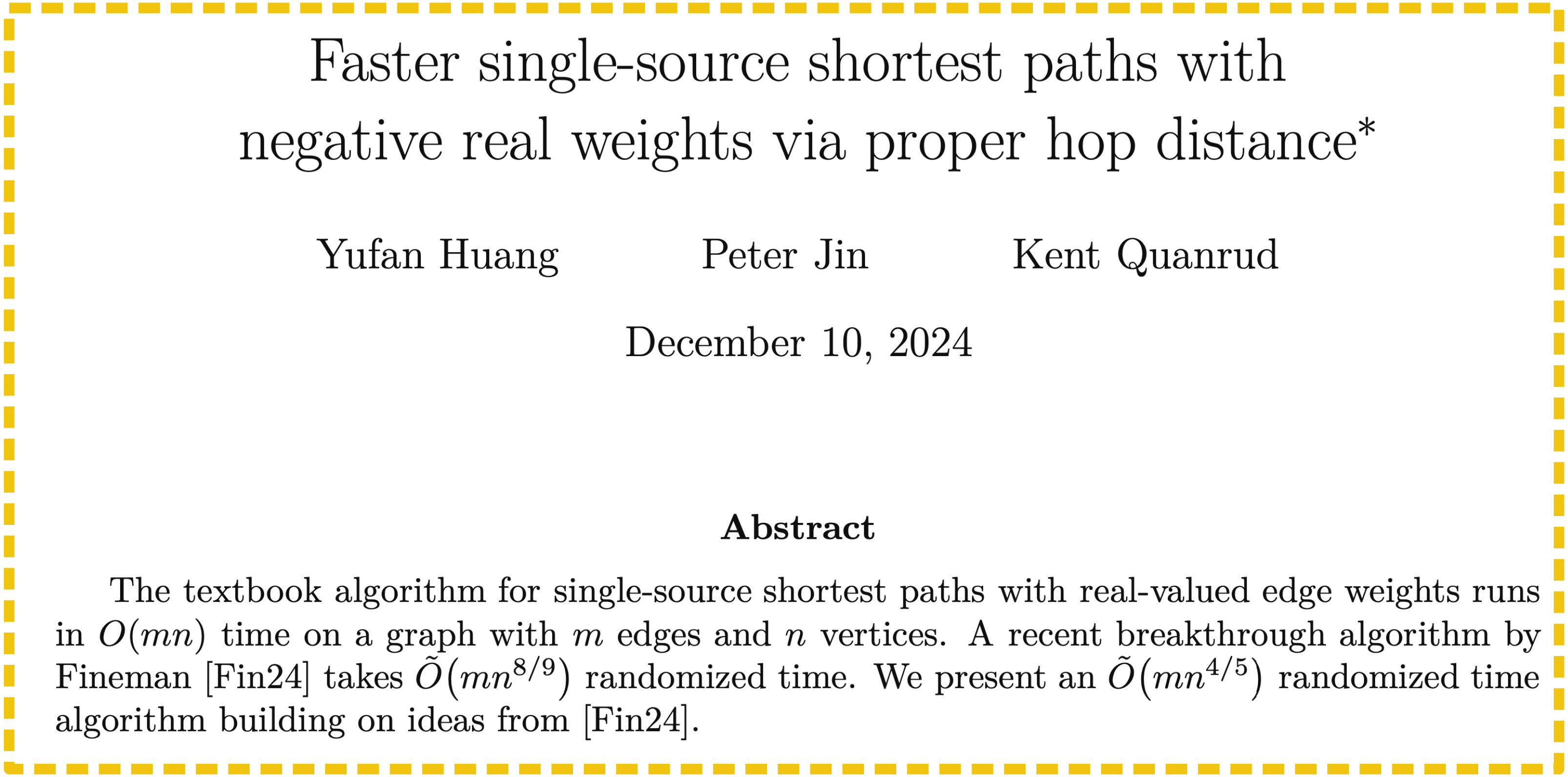


Quotient sparsification for submodular functions

Kent Quanrud*

Abstract

Graph sparsification has been an important topic with many structural and algorithmic consequences. Recently hypergraph sparsification has come to the fore and has seen exciting progress. In this paper we take a fresh perspective and show that they can be both be derived as corollaries of a general theorem on sparsifying matroids and monotone submodular functions.



Faster single-source shortest paths with negative real weights via proper hop distance*

Yufan Huang Peter Jin Kent Quanrud

December 10, 2024

Abstract

The textbook algorithm for single-source shortest paths with real-valued edge weights runs in $O(mn)$ time on a graph with m edges and n vertices. A recent breakthrough algorithm by Fineman [Fin24] takes $\tilde{O}(mn^{8/9})$ randomized time. We present an $\tilde{O}(mn^{4/5})$ randomized time algorithm building on ideas from [Fin24].

Recap:

Unweighted: BFS, $O(m+n)$

Weighted ≥ 0: Dijkstra's, $O(m+n\log n)$

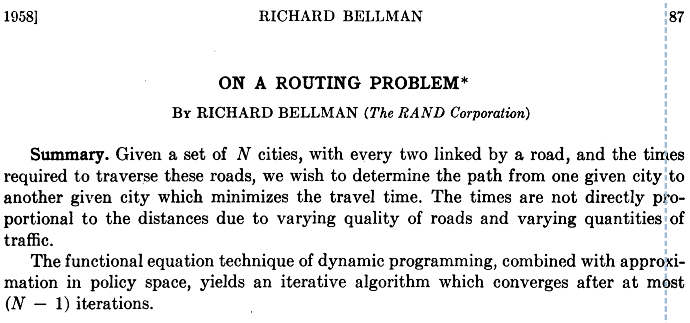
General weights: $O(mn)$

Shimbel 1955

Bellman 1958

Ford 1956

Moore 1959


1958] RICHARD BELLMAN 87

ON A ROUTING PROBLEM*

By RICHARD BELLMAN (*The RAND Corporation*)

Summary. Given a set of N cities, with every two linked by a road, and the times required to traverse these roads, we wish to determine the path from one given city to another given city which minimizes the travel time. The times are not directly proportional to the distances due to varying quality of roads and varying quantities of traffic.

The functional equation technique of dynamic programming, combined with approximation in policy space, yields an iterative algorithm which converges after at most $(N-1)$ iterations.

Integer weights $\leq \text{poly}(n)$

GT89

Gol95 $\tilde{O}(m\sqrt{n})$

CMSV17 $\tilde{O}(m^{10/7})$

BLN+20 $\tilde{O}(m+n^{1.5})$,

AMV20, $m^{4/3+o(1)}$

CKL20, $m^{1+o(1)}$ (!)

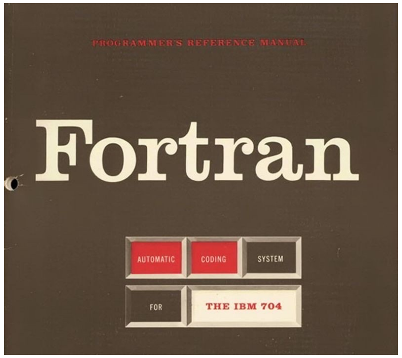
BNW22 $\tilde{O}(m)$ (!!)

BCF23

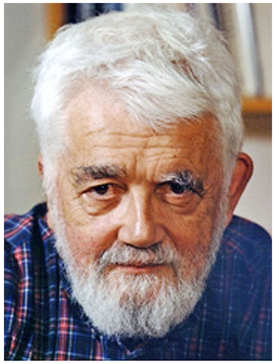
(hides $\log^{O(1)} W$)

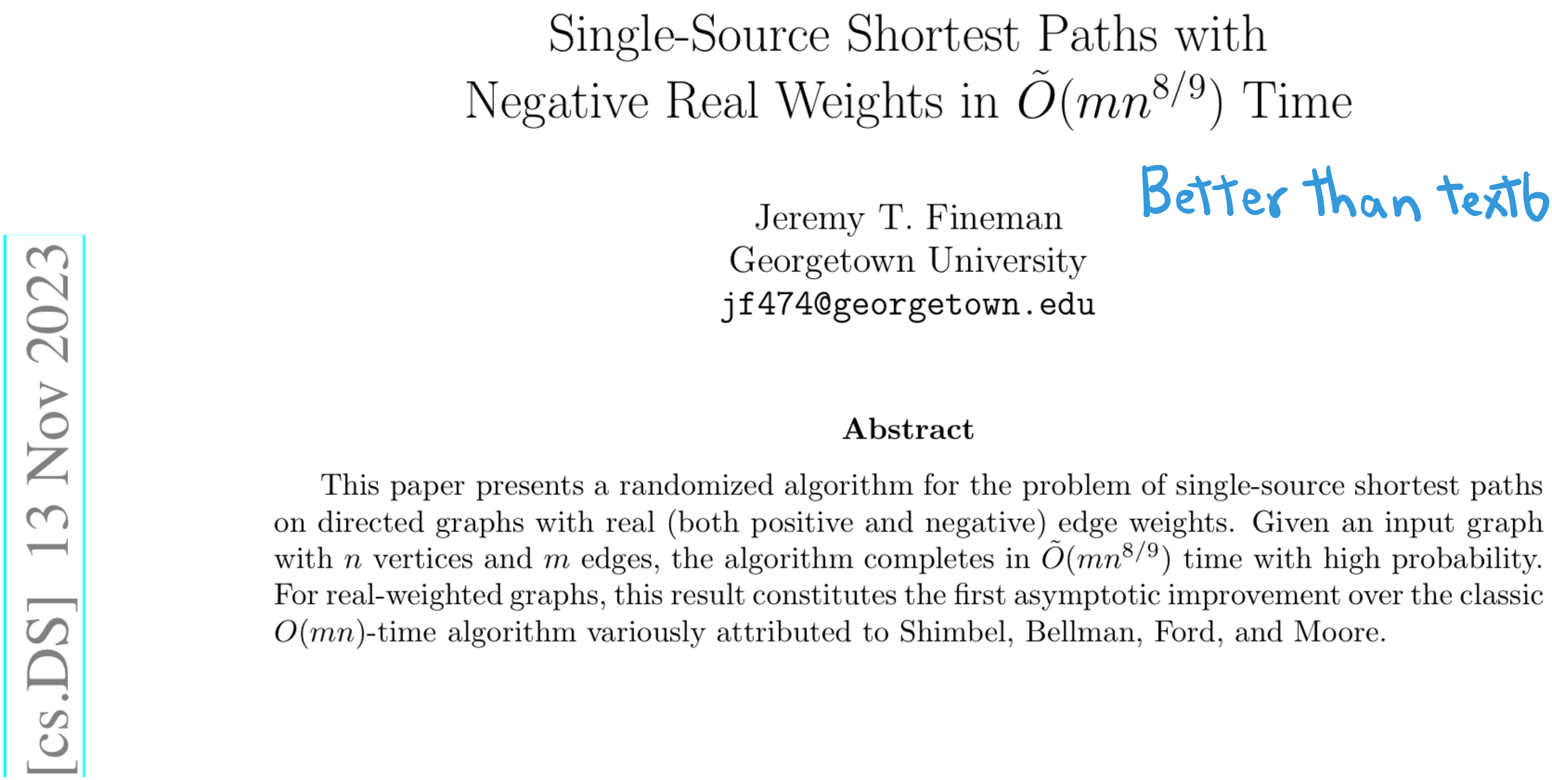

1955: IBM

1956: Illiac



1957: Fortran


1958: LISP



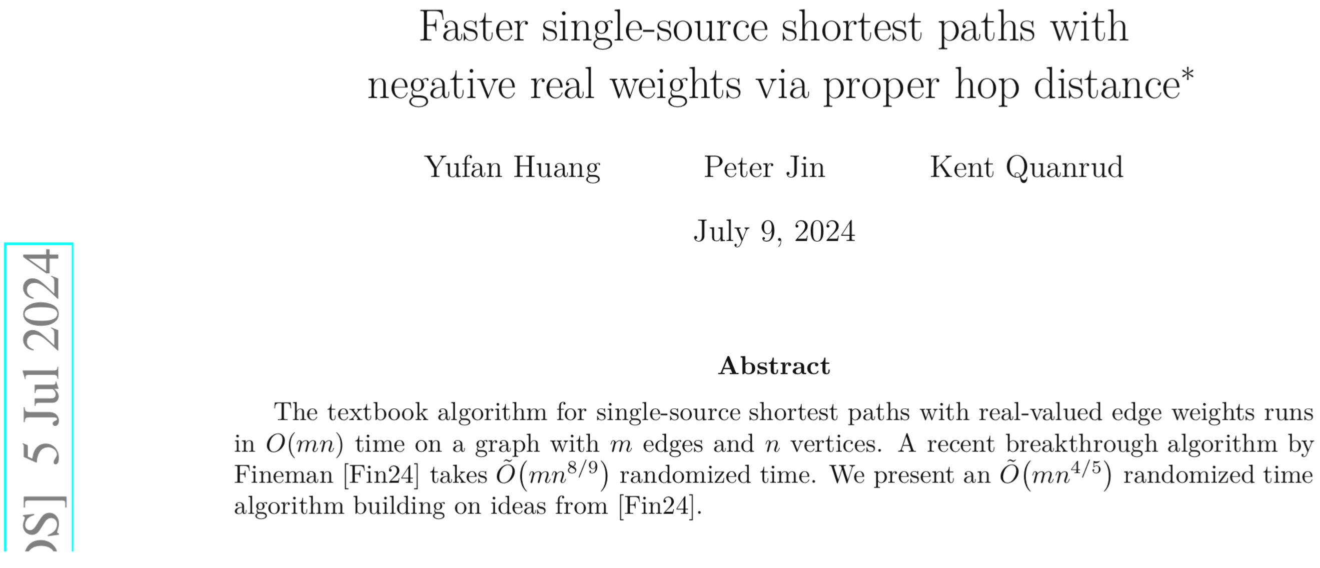
Single-Source Shortest Paths with Negative Real Weights in $\tilde{O}(mn^{8/9})$ Time

Better than textbook DP!

Jeremy T. Fineman
Georgetown University
jf474@georgetown.edu

Abstract

This paper presents a randomized algorithm for the problem of single-source shortest paths on directed graphs with real (both positive and negative) edge weights. Given an input graph with n vertices and m edges, the algorithm completes in $\tilde{O}(mn^{8/9})$ time with high probability. For real-weighted graphs, this result constitutes the first asymptotic improvement over the classic $O(mn)$-time algorithm variously attributed to Shimbel, Bellman, Ford, and Moore.



Faster single-source shortest paths with negative real weights via proper hop distance*

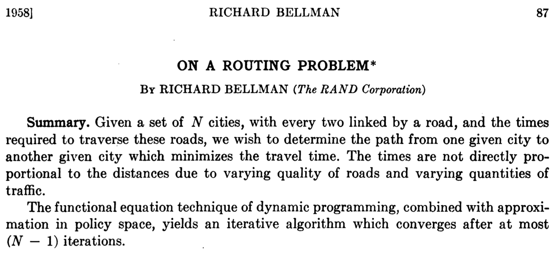
Yufan Huang Peter Jin Kent Quanrud

July 9, 2024

Abstract

The textbook algorithm for single-source shortest paths with real-valued edge weights runs in $O(mn)$ time on a graph with m edges and n vertices. A recent breakthrough algorithm by Fineman [Fin24] takes $\tilde{O}(mn^{8/9})$ randomized time. We present an $\tilde{O}(mn^{4/5})$ randomized time algorithm building on ideas from [Fin24].

topics



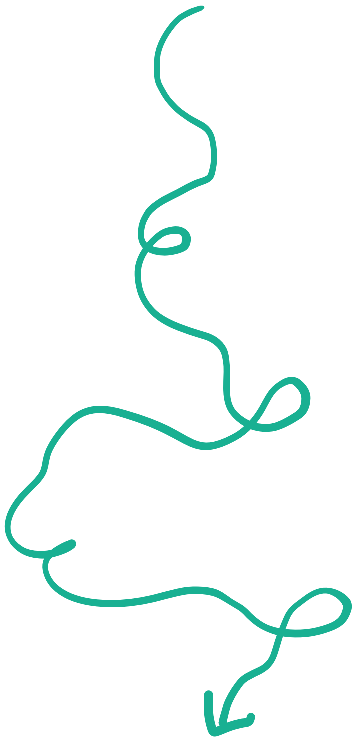
1958] RICHARD BELLMAN 87

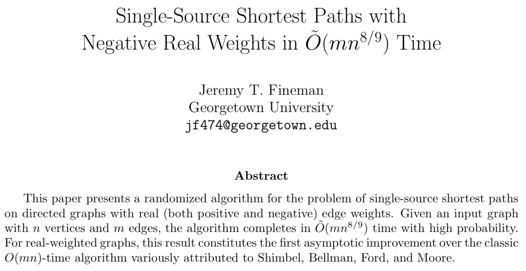
ON A ROUTING PROBLEM*

By RICHARD BELLMAN (*The RAND Corporation*)

Summary. Given a set of N cities, with every two linked by a road, and the times required to traverse these roads, we wish to determine the path from one given city to another given city which minimizes the travel time. The times are not directly proportional to the distances due to varying quality of roads and varying quantities of traffic.

The functional equation technique of dynamic programming, combined with approximation in policy space, yields an iterative algorithm which converges after at most $(N - 1)$ iterations.



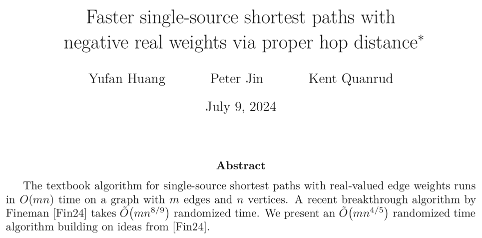


Single-Source Shortest Paths with Negative Real Weights in $\tilde{O}(mn^{8/9})$ Time

Jeremy T. Fineman
Georgetown University
`jf474@georgetown.edu`

Abstract

This paper presents a randomized algorithm for the problem of single-source shortest paths on directed graphs with real (both positive and negative) edge weights. Given an input graph with n vertices and m edges, the algorithm completes in $\tilde{O}(mn^{8/9})$ time with high probability. For real-weighted graphs, this result constitutes the first asymptotic improvement over the classic $O(mn)$-time algorithm variously attributed to Shimbel, Bellman, Ford, and Moore.



Faster single-source shortest paths with negative real weights via proper hop distance*

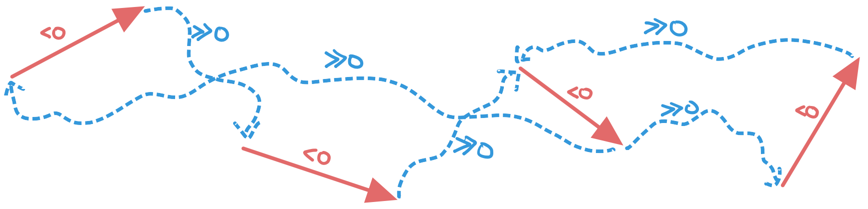
Yufan Huang Peter Jin Kent Quanrud

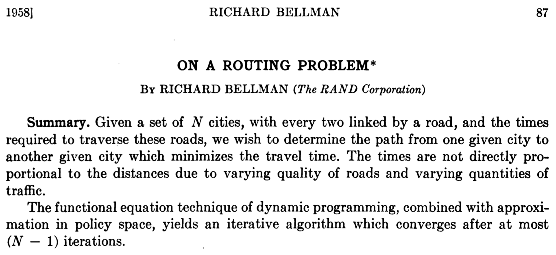
July 9, 2024

Abstract

The textbook algorithm for single-source shortest paths with real-valued edge weights runs in $O(mn)$ time on a graph with m edges and n vertices. A recent breakthrough algorithm by Fineman [Fin24] takes $\tilde{O}(mn^{8/9})$ randomized time. We present an $\tilde{O}(mn^{4/5})$ randomized time algorithm building on ideas from [Fin24].

topics




1958] RICHARD BELLMAN 87

ON A ROUTING PROBLEM*

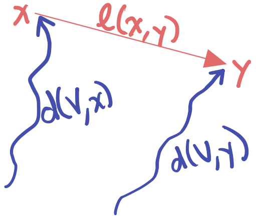
By RICHARD BELLMAN (*The RAND Corporation*)

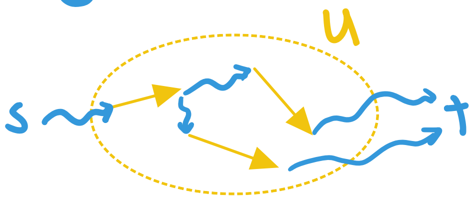
Summary. Given a set of N cities, with every two linked by a road, and the times required to traverse these roads, we wish to determine the path from one given city to another given city which minimizes the travel time. The times are not directly proportional to the distances due to varying quality of roads and varying quantities of traffic.

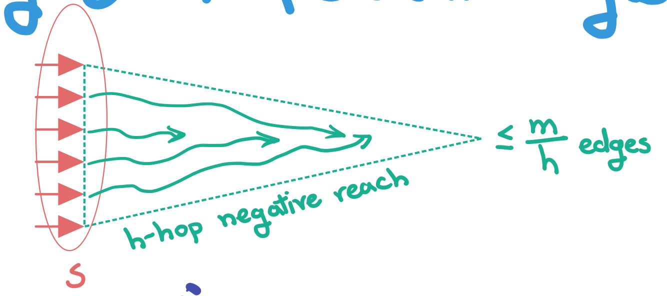
The functional equation technique of dynamic programming, combined with approximation in policy space, yields an iterative algorithm which converges after at most $(N - 1)$ iterations.

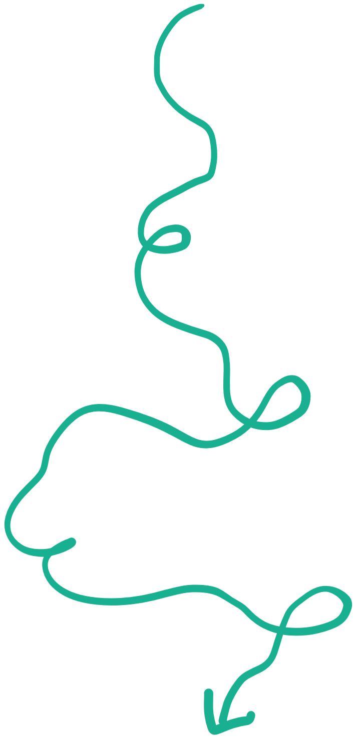
- vertex potentials

Johnson's APSP algo, "neutralizing" & "independent" edges



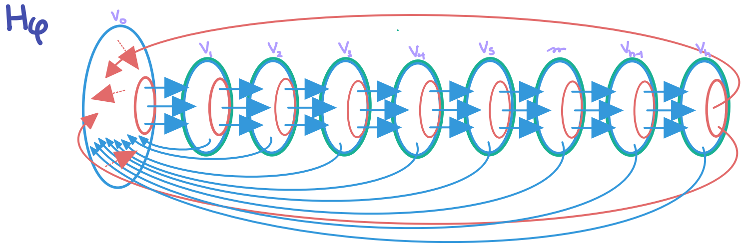




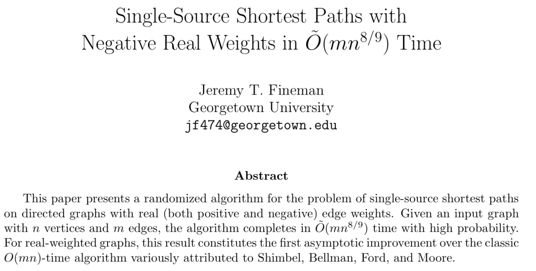


- hop distance, "proper hop distance"
  def, algo, key lemma, "negative sandwich"

- "Hop reduction"
  "remote edges",



k-hops in $G_s \rightarrow \frac{k}{h}$ hops in H_φ

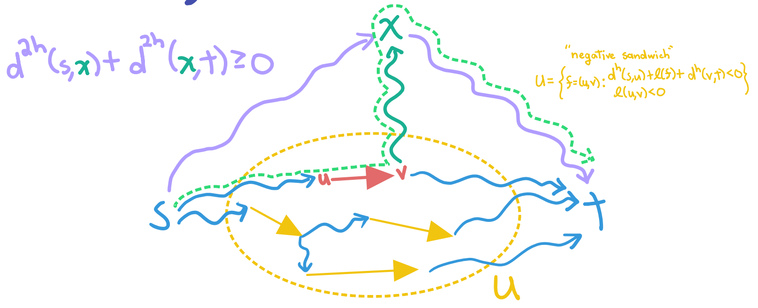

Single-Source Shortest Paths with
Negative Real Weights in $\tilde{O}(mn^{8/9})$ Time

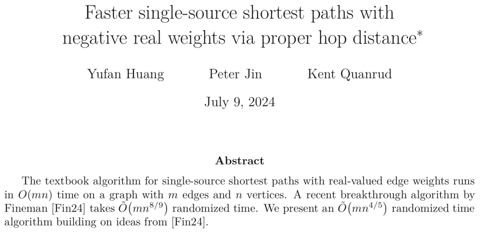
Jeremy T. Fineman
Georgetown University
jf474@georgetown.edu

Abstract

This paper presents a randomized algorithm for the problem of single-source shortest paths on directed graphs with real (both positive and negative) edge weights. Given an input graph with n vertices and m edges, the algorithm completes in $\tilde{O}(mn^{8/9})$ time with high probability. For real-weighted graphs, this result constitutes the first asymptotic improvement over the classic $O(mn)$-time algorithm variously attributed to Shimbel, Bellman, Ford, and Moore.

- "betweenness reduction",
  "remotization",




Faster single-source shortest paths with
negative real weights via proper hop distance*

Yufan Huang Peter Jin Kent Quanrud

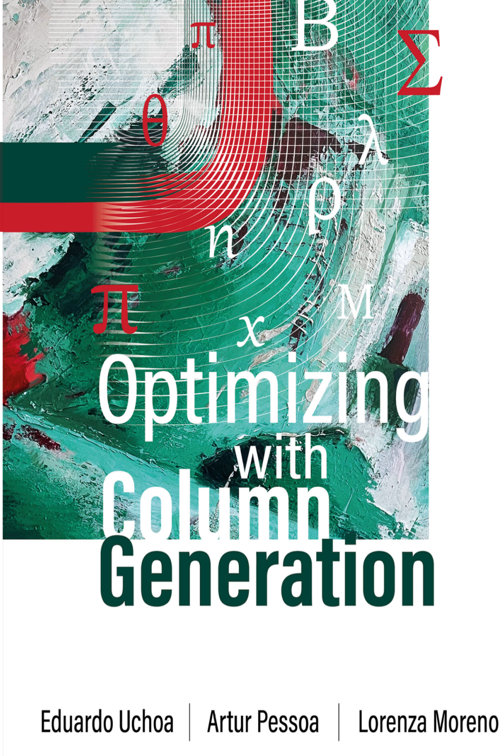
July 9, 2024

Abstract

The textbook algorithm for single-source shortest paths with real-valued edge weights runs in $O(mn)$ time on a graph with m edges and n vertices. A recent breakthrough algorithm by Fineman [Fin24] takes $\tilde{O}(mn^{8/9})$ randomized time. We present an $\tilde{O}(mn^{4/5})$ randomized time algorithm building on ideas from [Fin24].

oh my!

Aussois
January 2025



Optimizing
with
Column
Generation

Advanced Branch-Cut-and-Price Algorithms

Eduardo Uchoa
Universidade Federal Fluminense
INRIA International Chair (2022–2026)

Artur Pessoa
Universidade Federal Fluminense

Lorenza Moreno
Universidade Federal de Juiz de Fora

Column Generation (CG)

Method to solve Linear Programs (LPs) with a very large number of variables

Applied to important classes of Integer Programs (IPs), leading to Branch-and-Price (BP) and Branch-Cut-and-Price (BCP) algorithms:

- Vehicle routing
- Cutting and packing
- Airline planning
- Timetabling
- Crew scheduling
- Graph coloring
- Many others

The Paradox

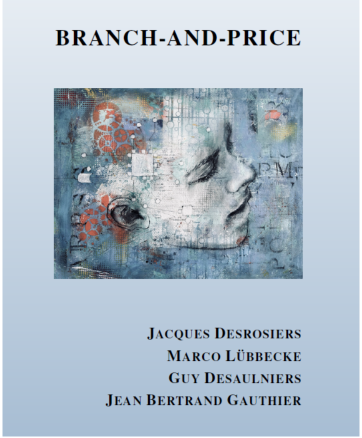
Column Generation is thriving:

- Hundreds of relevant papers published annually
- Modern advanced BCP algorithms much more powerful than BPs of 20 years ago
- Routinely applied in industry for million-dollar optimization problems

Yet, it remains a “well-kept secret”

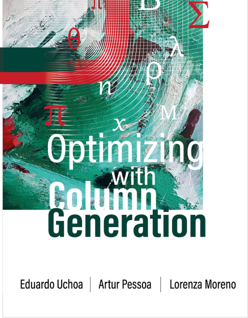
Key Challenges

- Educational Barriers:
    - No textbook (until very recently!)
    - Many key techniques are scattered in research articles
    - Non-standardized notation and terminology across literature
- Implementation Challenges:
    - Commercial solvers don't support Branch-and-Price
    - Open-source frameworks have limitations
    - May require custom coding for state-of-the-art performance
- Result: Technique is underutilized despite its potential



BRANCH-AND-PRICE

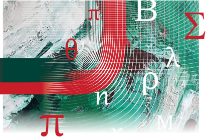
Jacques Desrosiers
Marco Lübbecke
Guy Desaulniers
Jean Bertrand Gauthier



Optimizing with Column Generation

Eduardo Uchoa | Artur Pessoa | Lorenza Moreno

“Optimizing with Column Generation”



Part I - Column Generation Basics:

- Five chapters covering CG *principles in-depth* (no contradiction!)
- Finished (300+ pages) and available for download at https://optimizingwithcolumngeneration.github.io/

“Optimizing with Column Generation”

Part II - Topics in Column Generation:

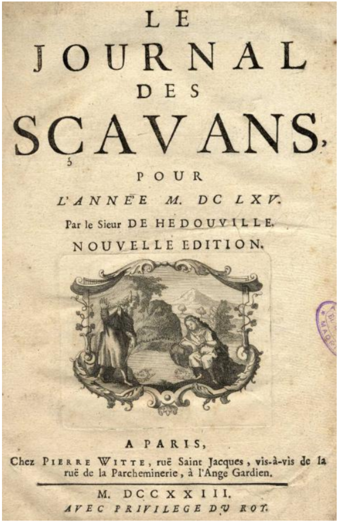
- Eight chapters covering the most advanced techniques in state-of-the-art BCP algorithms
- Expected to be finished by the end of 2025

OJMO: a Diamond Open Access journal in Mathematical Optimization

Michael Poss

The beginnings ...

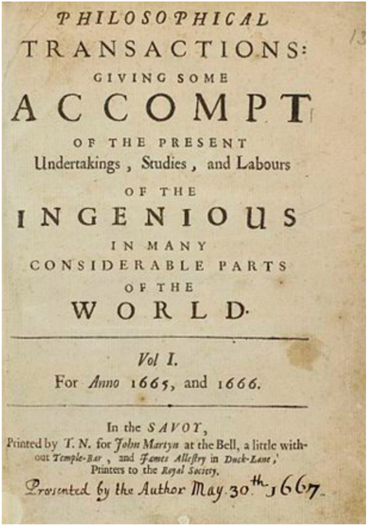
Back in the days, publishing was expensive!



LE
JOURNAL
DES
SCAVANS,
5
POUR
L'ANNEE M. DC LXV.
Par le Sieur DE HEDOUVILLE.
NOUVELLE EDITION.

A PARIS,
Chez PIERRE WITTE, ruë Saint Jacques, vis-à-vis de la ruë de la Parcheminerie, à l'Ange Gardien.
M. DCC XXIII.
AVEC PRIVILEGE DU ROY.

Paris, January 5, *1665*



PHILOSOPHICAL
TRANSACTIONS:
GIVING SOME
ACCOMPT
OF THE PRESENT
Undertakings, Studies, and Labours
OF THE
INGENIOUS
IN MANY
CONSIDERABLE PARTS
OF THE
WORLD.

Vol I.
For *Anno* 1665, and 1666.

In the *SAVOY*,
Printed by *T. N.* for *John Martyn* at the Bell, a little without *Temple-Bar*, and *James Allestry* in *Duck-Lane*,
Printers to the *Royal Society*.
Presented by the Author May.30th 1667

London, March 6, *1665*

The publishing oligopoly

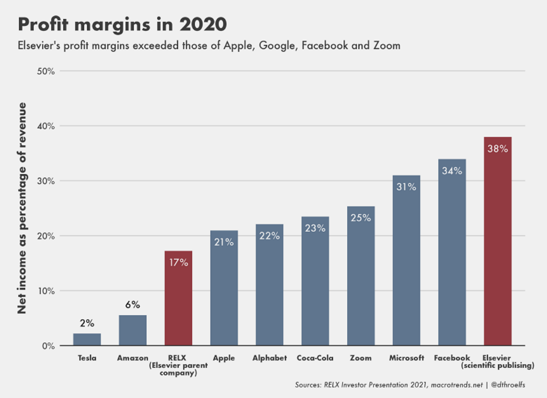
- Electronic publishing and LaTeX significantly reduced the costs
- Led to nearly **open** and **free** publications?

The publishing oligopoly

- Electronic publishing and LaTeX significantly reduced the costs
- Led to nearly **open** and **free** publications?
- Unfortunately, they led instead to ever-increasing profit margins:

The publishing oligopoly

- Electronic publishing and LaTeX significantly reduced the costs
- Led to nearly **open** and **free** publications?
- Unfortunately, they led instead to ever-increasing profit margins:



Profit margins in 2020

Elsevier's profit margins exceeded those of Apple, Google, Facebook and Zoom

Sources: RELX Investor Presentation 2021, macrotrends.net | @dthroelfs

The publishing oligopoly

A version of this story appeared in Science, Vol 386, Issue 6726.

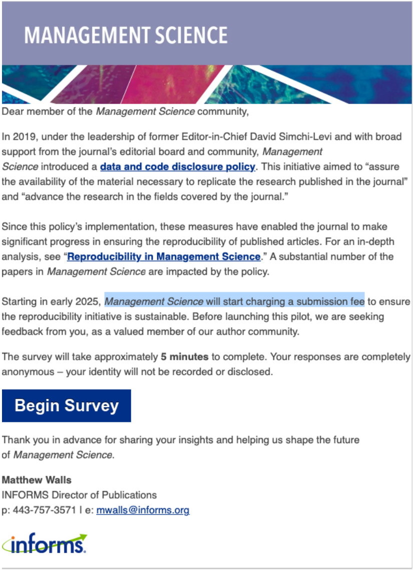
RELATED PODCAST

Making Latin American science visible, and advances in cooling tech

BY SARAH CRESPI, BRENT GROCHOLSKI, SOFIA MOUTINHO • PODCAST • **05 DEC 2024**

This story is part of a News series about global equity in science. **READ MORE** ›

In 2016, Marcus Oliveira, a biochemist at the Federal University of Rio de Janeiro, submitted a study on the metabolism of a tropical parasite to a mainstream open-access journal based in the United States. It was the ninth paper he had published in the journal, for which he had also volunteered time as a peer reviewer for dozens of articles. But this time he could not afford the $1200 article-processing charge (APC), as his grant funding was nearly depleted. He requested the fee waiver the journal says it offers to authors from lower income countries, but the negotiations were tense. “I felt morally assaulted,” he says. “At some point, the journal requested I send them a personal bank statement to prove I didn’t have the means.”



Dear member of the *Management Science* community,

In 2019, under the leadership of former Editor-in-Chief David Simchi-Levi and with broad support from the journal’s editorial board and community, *Management Science* introduced a **data and code disclosure policy**. This initiative aimed to “assure the availability of the material necessary to replicate the research published in the journal” and “advance the research in the fields covered by the journal.”

Since this policy’s implementation, these measures have enabled the journal to make significant progress in ensuring the reproducibility of published articles. For an in-depth analysis, see “**Reproducibility in Management Science**.” A substantial number of the papers in *Management Science* are impacted by the policy.

Starting in early 2025, *Management Science* will start charging a submission fee to ensure the reproducibility initiative is sustainable. Before launching this pilot, we are seeking feedback from you, as a valued member of our author community.

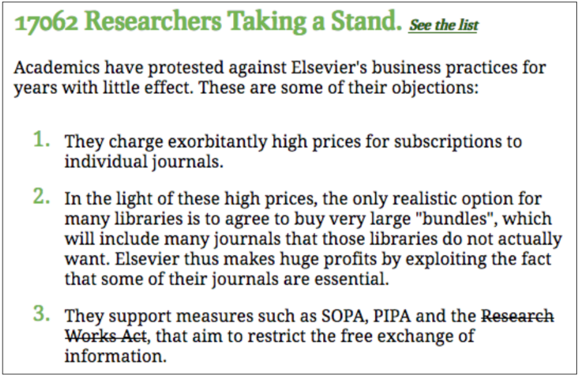
The survey will take approximately **5 minutes** to complete. Your responses are completely anonymous – your identity will not be recorded or disclosed.

Thank you in advance for sharing your insights and helping us shape the future of *Management Science*.

Matthew Walls
INFORMS Director of Publications
p: 443-757-3571 I e: mwalls@informs.org

The uprising

Sir Tim Gowers, Fields Medal 1998



17062 Researchers Taking a Stand. ***See the list***

Academics have protested against Elsevier's business practices for years with little effect. These are some of their objections:

1. They charge exorbitantly high prices for subscriptions to individual journals.
2. In the light of these high prices, the only realistic option for many libraries is to agree to buy very large "bundles", which will include many journals that those libraries do not actually want. Elsevier thus makes huge profits by exploiting the fact that some of their journals are essential.
3. They support measures such as SOPA, PIPA and the ~~Research Works Act~~, that aim to restrict the free exchange of information.

http://www.thecostofknowledge.com/

Taken from the presentation of Marie Farge

Diamond Open Access today

Excellent free journals exist today, for instance;

Machine learning, Artificial intelligence

- Journal of Artificial Intelligence Research (JAIR)
- Journal of Machine Learning Research (JMLR)

Theoretical Computer Science

- Advances in Combinatorics
- TheoretiCS
- Theory of Computing
- Innovations in Graph Theory (just started)

And many more: **https://freejournals.org/current-member-journals/**

Open Journal of Mathematical Optimization (OJMO)

Steering Committee

- Dimitris Bertsimas
- Martine Labbé
- Eva K. Lee
- Marc Teboulle

Area Editors

- **Continuous Optimization** - David Russell Luke
- **Discrete Optimization** - Sebastian Pokutta
- **Optimization under Uncertainty** - Guzin Bayraksan
- **Computational aspects and applications** - Jérôme Malick

As of today

- ranked Q2 at Scimago in Control and Optimization
- indexed in zbMATH, Scopus, dblp, MathSciNet
- 5 issues, 8-10 papers per issue
- >20 papers in the pipeline

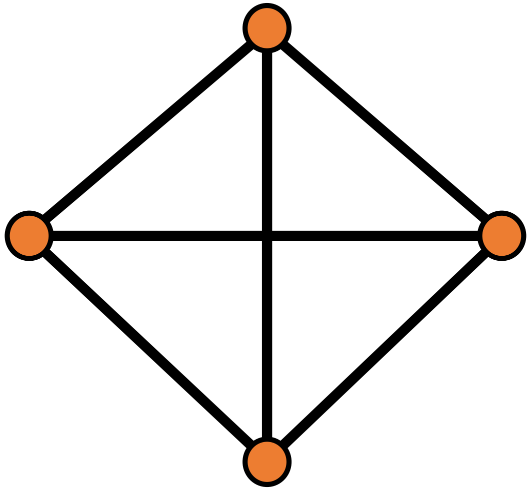
Visit https://ojmo.centre-mersenne.org/

Price of Anarchy for Graphic Matroid Congestion Games

Marc Uetz, University of Twente
(with Wouter Fokkema and Ruben Hoeksma)

Graphic Matroid Congestion Game

- Given graph $G = (V, E)$
- Players i select spanning tree T_i
- Affine cost function per edge e
- No. of players n_e on edge e,
  cost $c_e(n_e) = a_e\ n_e + b_e$

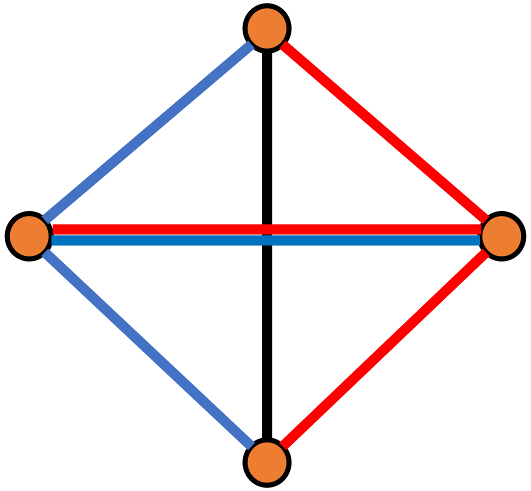


$c(n_e) = n_e$ for all edges e

Graphic Matroid Congestion Game

- Given graph $G = (V, E)$
- Players i select spanning tree T_i
- Affine cost function per edge e
- No. of players n_e on edge e,
  cost $c_e(n_e) = a_e\ n_e + b_e$

Total cost $\sum c(T_i)$ = 8

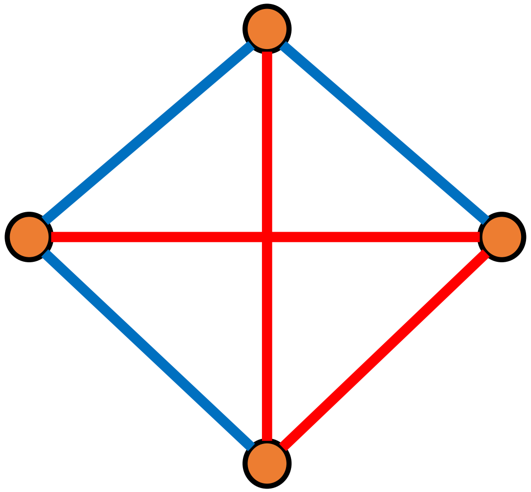


$c(T_1) = 4,$
$c(T_2) = 4$

Graphic Matroid Congestion Game

- Given graph $G = (V, E)$
- Players i select spanning tree T_i
- Affine cost function per edge e
- No. of players n_e on edge e,
  cost $c_e(n_e) = a_e\ n_e + b_e$

Total cost $\sum c(T_i)$ = 6



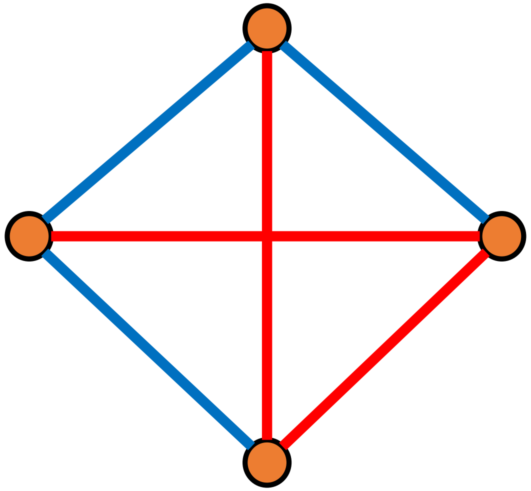
$c(\,T_1) = 3,$
$c(\,T_2) = 3$

Graphic Matroid Congestion Game

- Given graph $G = (V, E)$
- Players i select spanning tree T_i
- Affine cost function per edge e
- No. of players n_e on edge e,
  cost $c_e(n_e) = a_e\ n_e + b_e$

Total cost $\sum c(T_i)$ = 6

⇒ Price of Anarchy (PoA) $\geq$ 4/3



$c(T_1) = 3,$
$c(T_2) = 3$

Price of Anarchy Symmetric Congestion Games

PoA for arbitrary atomic congestion games and n players is at most $(5n-2)/(2n+1)$

[Christodolou & Koutsoupias STOC 2005]

Result: Tight Lower Bound Constructions

PoA for graphic matroid congestion games and n players is equal to $(5n-2)/(2n+1)^*$

[SAGT 2024]

(*) for $n = 2,3,4$ and $n \to \infty$