REOP - Session 10 Network design & Lagrangian relaxation

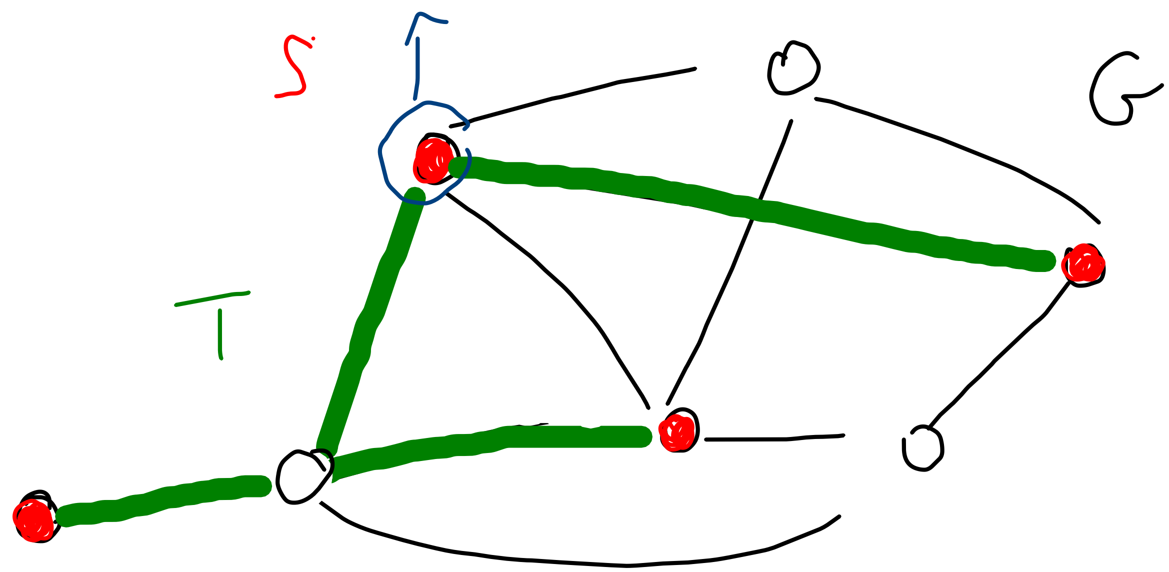
I/ Network design

A) (Graphical) Steiner tree

Problem description:

Input: an undirected graph $G = (V, E)$, edge weights $w : E \to \mathbb{R}_+$
a set $S \subseteq V$ of terminals

Task: find a tree T of minimum weight among those that cover S
(eg. every vertex of S is incident to an edge of $E(T)$)



Two special cases:

1) when $S = V$, it is a Minimum Spanning Tree pb

2) when $|S| = 2$, $S = \{o, d\}$, it is a Shortest Path pb

$\longrightarrow$ both are polynomial problems

Thm : The Steiner tree problem is NP-hard

B) Exact algorithms

- Dynamic programming (Dreyfus - Wagner) :
  complexity $O(3^{|T|}n + 2^{|T|}n^2 + mn + n^2 \log n)$
  where $n = |V|$
  $m = |E|$ and $|T|$ is the size of the resulting tree
  since $|T| \geq |S|$, it is exponential in $|S|$

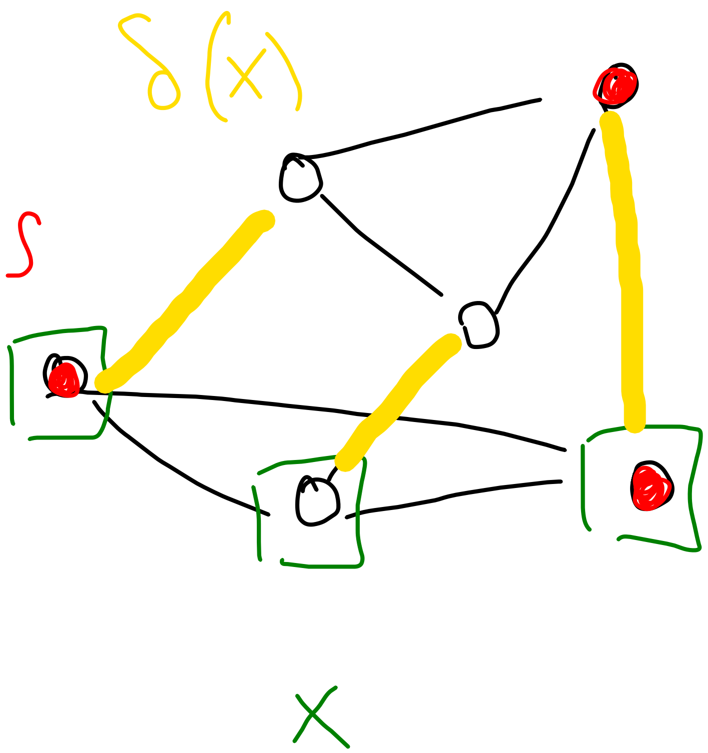
- ILP formulation : similar to the Traveling Salesperson (TSP)
  We fix $r \in S$ as the root
  Decision variable : $x_e = \begin{cases} 1 & \text{is we select edge } e \text{ in } E(T) \\ 0 & \text{otherwise} \end{cases}$
  Objective function : $\min \sum_{e \in E} w(e) x_e$
  Constraint : $\forall X \subseteq V$ such that $\begin{cases} r \notin X \\ X \cap S \neq \emptyset \end{cases}$, we impose ...

Constraint : $\forall X \subseteq V$ such that $\left| \begin{array}{l} r \notin X \\ X \cap S \neq \emptyset \end{array} \right.$, we impose ...

... that there is ≥ 1 edge coming into X

$$\iff \sum_{e \in \delta(X)} x_e \geq 1 \quad \forall X \subseteq V \setminus \{r\}, \quad X \cap S \neq \emptyset \qquad (*)$$

↳ set of edges incident to X | exponential nb of constraints :(



Proof: If x represents a Steiner tree, then it satisfies $(*)$.

Conversely, suppose $x \in \{0,1\}^E$ satisfies $(*)$.

Let $T = \{e \in E : x_e = 1\}$

- T is connected. Suppose $\exists s \in S$ such that s & r are not connected by the tree T. Let's call $C(s)$ & $C(r)$ their respective connected components in T. Taking $X = C(s)$ yields a contradiction: if $r \notin X$, then there is an edge incident to X in T by $(*)$ which contradicts the definition of $C(s)$

- T covers S : $X = \{s\}$, $s \in S$
- ~~T has no cycles~~ we can remove cycles safely

⚠ Exponentially many constraints: we can't use the relaxation of (ILP) directly in a Branch & Bound

⇒ When we solve the relaxation (LP) ($x_e \in$ ~~{0,1}~~ $\in [0,1]$)
- start with a small subset of constraints
- solve
- check if a nonincluded constraint is violated
  ↳ separation pb: efficiently solvable for TSP, MST, Steiner tree (related to min cut)
- include it & loop

Branch & Bound becomes Branch & Cut with this constraint generation technique

C) Heuristic

Local search: similar to facility location

1) sites ↓ LS
2) clients

High-level search: find the vertices $V(T)$ of the Steiner tree

$$V(T) = \underbrace{S}_{\text{terminals}} \cup \underbrace{(V(T) \setminus S)}_{\text{Steiner points}}$$

Low-level search: complete the solution with the edges $E(T)$, which boils down to solving an MST on $G[V(T)]$

II/ Lagrangian relaxation (outside of the program)

A) Motivating example: Multi-Agent Pathfinding

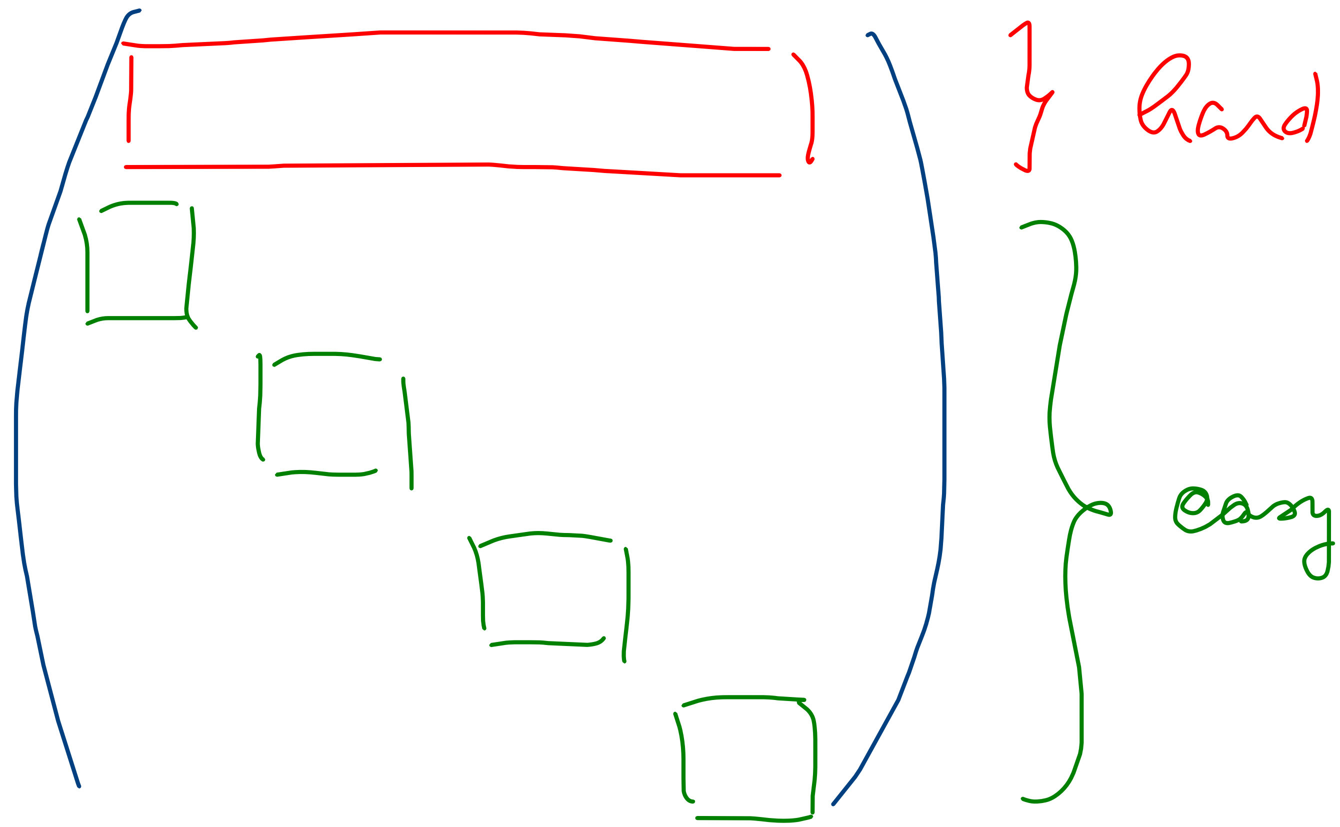
The MAPF problem consists in finding paths for a set A of agents on a graph G such that

Easy • The path P_a of agent a leads from its origin o_a to its destination d_a (in a given window)

Hard (•) The paths of two agents $a_1 \neq a_2$ cannot visit the same vertex v at the same time t

If we forget the hard constraint, we can plan the path of every train independently $\longrightarrow$ decomposition (efficient)
The hard constraint introduces dependencies & makes the pb NP-hard

Constraint matrix



B) Def. of Lagrangian relaxation

We consider the general ILP

$$z_I = \min_x c^T x \quad \text{subject to} \quad \left| \begin{array}{l} x \in \mathbb{Z}^n \\ A_{easy}\, x \leq b_{easy} \\ A_{hard}\, x \leq b_{hard} \end{array} \right.$$

We are going to penalize the hard constraint in the objective instead of enforcing it: let $\lambda \in \mathbb{R}_+^{d_{hard}}$

$$z_{LR}(\lambda) = \min_x \underbrace{c^T x + \lambda^T (A_{hard}\, x - b_{hard})}_{\tilde{c}(\lambda)^T x \,+\, \text{constant}} \quad \text{s.t.} \quad \left| \begin{array}{l} x \in \mathbb{Z}^n \\ A_{easy}\, x \leq b_{easy} \end{array} \right.$$

$z_{LR}(\lambda)$ can be computed efficiently for any value of λ

Prop: $z_{LR}(\lambda) \leq z_I \quad \forall \lambda \geq 0$ it is a relaxation and we want it to be tight

We want to chose λ so that $z_{LR}(\lambda)$ is as high as possible

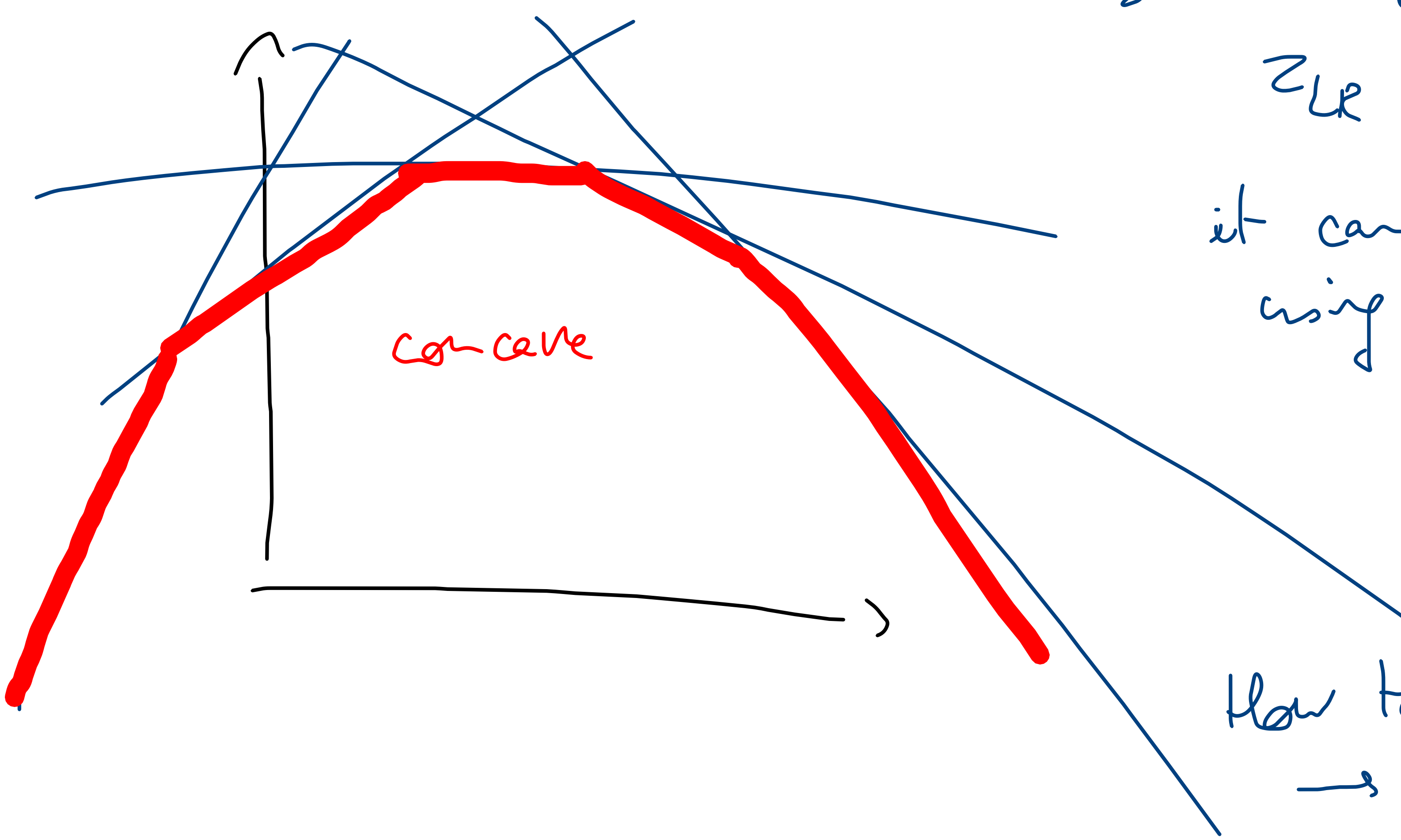
$$z_{LD} = \max_{\lambda \geq 0} z_{LR}(\lambda)$$

C) How to compute the Lagrangian dual $z_{LD} = \max_{\lambda \geq 0} z_{LR}(\lambda)$

$$z_{LR}(\lambda) = \min_x \underbrace{c^T x + \lambda^T (A_{hard} x - b_{hard})}_{f_x(\lambda)} \text{ s.t. } \left| \begin{array}{l} x \in \mathbb{Z}^n \\ A_{easy} x = b_{easy} \end{array} \right.$$

$z_{LR}(\lambda)$ is a pointwise minimum of a set of linear functions.

$\{ f_x : x \in \mathbb{Z}^n, A_{easy} x = b_{easy} \}$



z_{LR} is concave

it can be efficiently maximized using ~~subgradient descent~~ supergradient ascent

↳ the function isn't necessarily differentiable

How to compute a supergradient?

→ see the notes

D) How good is this bound?

$z_{LD} = \max_{\lambda \geq 0} z_{LR}(\lambda)$ compared with z_{lin} linear relax

Which one is better?

$X_{hard} = \{x \in \mathbb{R}^n : A_{hard}\, x = b_{hard}\}$

$X_{easy} = \{x \in \mathbb{R}^n : A_{easy}\, x = b_{easy}\}$

<u>Thm</u> (Geoffrion):

$$z_{LD} = \min_x c^T x \quad \text{s.t.} \quad \left| \begin{array}{l} x \in \operatorname{conv}(X_{easy} \cap \mathbb{Z}^n) \\ x \in X_{hard} \end{array} \right.$$

$\vee$/

while $$z_{lin} = \min_x c^T x \quad \text{s.t} \quad \left| \begin{array}{l} x \in X_{easy} \\ x \in X_{hard} \end{array} \right.$$

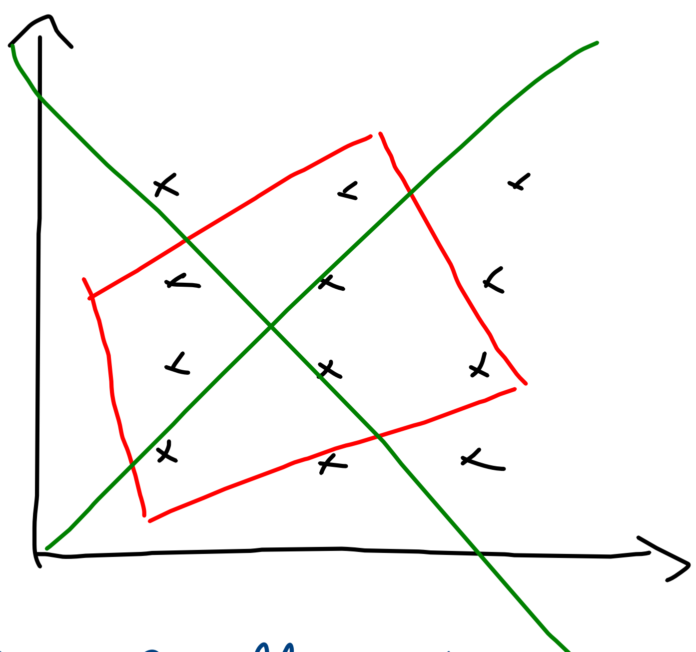
included

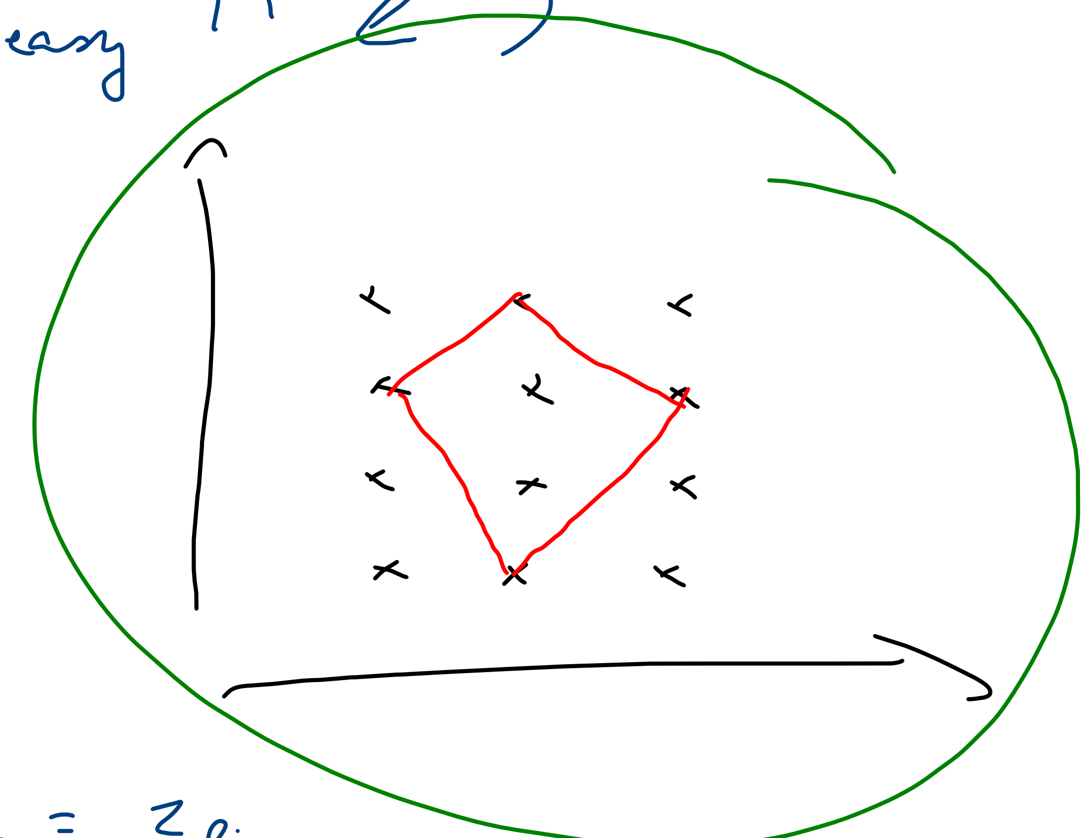
Since $X_{easy} \cap \mathbb{Z}^n \subset X_{easy}$ & X_{easy} convex,
$\operatorname{conv}(X_{easy} \cap \mathbb{Z}^n) \subset X_{easy}$

In the MAPF pb, $X_{easy} = \{ \text{independent shortest paths} \}$

The Shortest Path pb has a perfect formulation: the polyhedron has integer vertices

$$X_{easy} = \text{conv}(X_{easy} \cap \mathbb{Z}^n)$$





By Geoffrion's thm, $z_{LD} = z_{lin}$

III / Questions

wooclap.com/REOP2021GDFB for feedback

10h10 in amphi Cauchy