

# Network design & Lagrangian relaxation

Guillaume Dalle (ENPC, CERMICS)

REOP - Class 10 (11/01/2021)

## Contents

<b>1 Network design</b>	<b>1</b>
1.1 Graphical Steiner tree . . . . .	1
1.2 Exact algorithms . . . . .	1
1.3 Approximation algorithms & heuristics . . . . .	1
1.4 Variants . . . . .	2
<b>2 Lagrangian relaxation</b>	<b>2</b>
2.1 Motivating example: MAPF . . . . .	2
2.2 Definition . . . . .	2
2.3 Quality of the bound . . . . .	3

## 1 Network design

### 1.1 Graphical Steiner tree

Problem description

- Input: an undirected graph  $G = (V, E)$  with edge weights  $w : E \rightarrow \mathbb{R}_+$ , a set of terminals  $S \subset V$
- Task: find a tree  $T$  of minimum weight covering  $S$

We already know two polynomial special cases:

- $|S| = 2$ : shortest path problem
- $S = V$ : minimum spanning tree (MST)

Unfortunately, the general version is NP-hard

### 1.2 Exact algorithms

- Dynamic programming (Dreyfus-Wagner): complexity  $O(3^{|T|}n + 2^{|T|}n^2 + mn + n^2 \log n)$
- ILP: formulation is similar to TSP, with exponential number of constraints  $\implies$  branch & cut approach.  
Let  $r \in S$  be any fixed terminal (root):

$$\min_{x \in \{0,1\}^E} \sum_{e \in E} w(e)x_e \quad \text{s.t.} \quad \sum_{e \in \delta(X)} x_e \geq 1 \quad \forall X \subseteq V, r \notin X, X \cap S \neq \emptyset$$

### 1.3 Approximation algorithms & heuristics

- 2-approximation algorithm (Kou-Markowsky-Berman):
  - Let  $\bar{G}$  be the metric closure of  $G$ , and  $\bar{w}$  the (shortest path) edge weights
  - Find an MST  $\bar{T}$  in  $\bar{G}[S]$  with respect to  $\bar{w}$

- Decompress it to obtain a Steiner tree  $T$  in  $G$
- Local search: same idea as for facility location.
  - High-level search on the vertices  $V(T)$  of the Steiner tree (terminals  $\cup$  Steiner points)
  - If the vertices  $V(T)$  are fixed, choosing the edges  $E(T)$  amounts to finding an MST

## 1.4 Variants

- Euclidean Steiner tree: NP-hard
- Manhattan Steiner tree: NP-hard

## 2 Lagrangian relaxation

Sometimes the bound provided by linear relaxation of a MILP is not tight enough.

In a few cases, one can change the formulation and handle an exponential number of constraints with constraint generation (see TSP). We now present another method which is useful to decompose structured problems.

### 2.1 Motivating example: MAPF

The Multi-Agent Path Finding problem consists in finding paths for a set  $A$  of agents on a graph  $G$  such that:

1. The path  $P_a$  of agent  $a$  leads it from its origin  $o_a$  to its destination  $d_a$
2. The paths of two agents  $a_1$  and  $a_2$  cannot visit the same vertex at the same time

Without the conflict constraint, this decomposes into a set of  $|A|$  independent shortest path problems, which can be quickly solved with any standard algorithm from the class.

Unfortunately, the additional constraint makes MAPF NP-hard, which is why we need an additional trick to exploit the problem's natural decomposition.

### 2.2 Definition

We consider the general ILP

$$z_I = \min_x c^\top x \quad \text{s.t.} \quad \begin{cases} A_{\text{easy}}x \leq b_{\text{easy}} \\ A_{\text{hard}}x \leq b_{\text{hard}} \\ x \in \mathbb{Z}^n \end{cases}$$

Our premise is that this problem would be easy to solve if we had only the easy constraints. We are going to use this insight by solving a new problem where the the hard constraints are penalized instead of being strictly enforced:

$$z_{LR}(\lambda) = \min_x c^\top x + \lambda^\top (A_{\text{hard}}x - b_{\text{hard}}) \quad \text{s.t.} \quad \begin{cases} A_{\text{easy}}x \leq b_{\text{easy}} \\ x \in \mathbb{Z}^n \end{cases}$$

We can compute  $z_{LR}(\lambda)$  for any value of  $\lambda$ , but we are interested in the best possible bound, so it makes sense to search for

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

Since  $\lambda \mapsto z_{LR}(\lambda)$  is a minimum of linear functions of  $\lambda$ , it is concave and we can maximize it using simple methods such as projected supergradient ascent (the analogue of subgradient descent).

### 2.3 Quality of the bound

How does this Lagrangian dual bound  $z_{LD}$  compare to the linear relaxation  $z_{lin}$ ? To answer that, let us define

$$X_{\text{easy}} = \{x \in \mathbb{R}^n \mid A_{\text{easy}}x = b_{\text{easy}}\} \quad \text{and} \quad X_{\text{hard}} = \{x \in \mathbb{R}^n \mid A_{\text{hard}}x = b_{\text{hard}}\}$$

Then Geoffrion's theorem tells us

$$z_{LD} = \min_x c^\top x \quad \text{s.t.} \quad \begin{cases} x \in \text{conv}(X_{\text{easy}} \cap \mathbb{Z}^n) \\ x \in X_{\text{hard}} \end{cases}$$

whereas the linear relaxation yields a weaker lower bound

$$z_{lin} = \min_x c^\top x \quad \text{s.t.} \quad \begin{cases} x \in X_{\text{easy}} \\ x \in X_{\text{hard}} \end{cases}$$