APPROXIMATION ALGORITHMS

LINEAR PROGRAMMING & DETERMINISTIC ROUNDING

RASMUS PAGH
UNIVERSITY OF COPENHAGEN

TODAY

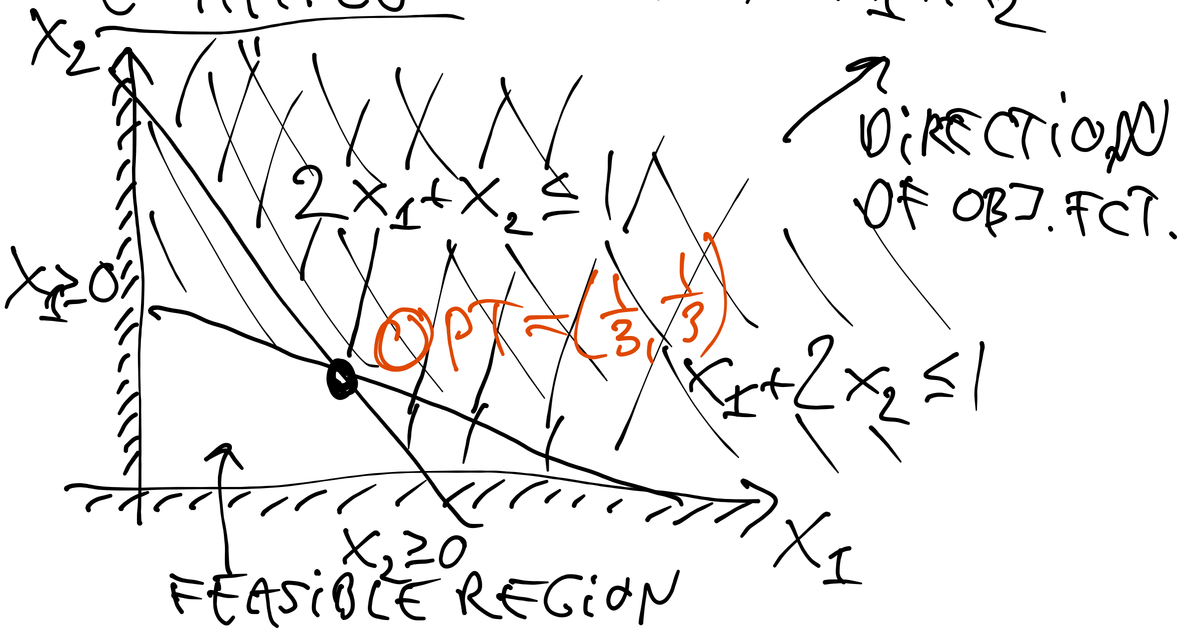
- QUICK OVERVIEW/REMINDER FOR LINEAR PROGR.
- CASE STUDY: VERTEX COVER, SET COVER
- WORKING WITH LPs IN PYTHON
- CASE STUDY: PRIZE-COLLECTING STEINER TREE PROBLEM

LINEAR PROGRAM (CANONICAL FORM)

- DECISION VARIABLES $x_1, \ldots, x_n \in \mathbb{R}$, $x_i \geq 0 \ \forall i$
- LINEAR CONSTRAINTS OF THE FORM $\sum_{i=1}^{n} a_i x_i \leq b_j$
- A LINEAR OBJECTIVE FUNCTION $\max \sum_{i=1}^{n} c_i x_i$

EXAMPLE

MAX $x_1 + x_2$



IN HIGHER DIMENSIONS:

- FEASIBLE REGION IS INTERSECT. OF HALFSPACES
- OPTIMAL SOLUTION CAN BE:
  - UNIQUE IN A "CORNER"
  - ALONG A "FACE" OF POLYTOPE
  - NOT EXIST: UNBOUNDED OR INFEASIBLE

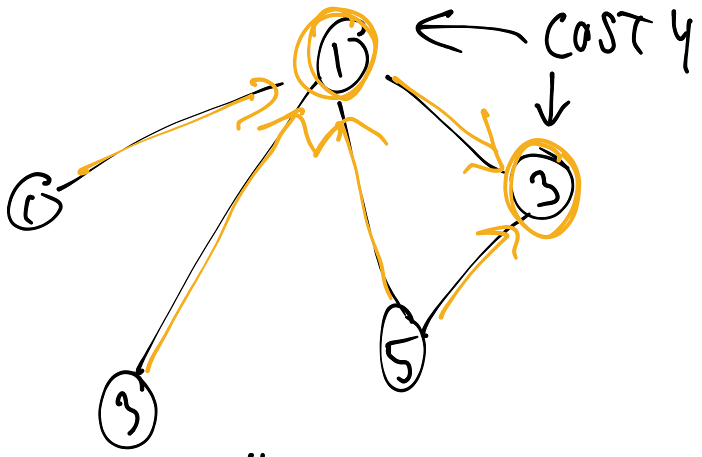
PAUSE TO THINK

HOW CAN WE TRANSLATE A GENERAL LINEAR PROGRAM WITH $\geq$, $=$ CONSTRAINTS; MAYBE MINIMIZING INSTEAD OF MAXIMIZING; MAYBE ALLOWING VALUES OF x_i LESS THAN 0; INTO AN EQUIVALENT LP IN CANONICAL FORM?

E.G.: MINIMIZE $x_1 + x_2 + x_3$

UNDER
$$x_1 + 2x_2 = 1$$
$$x_1 - x_3 \geq 0$$
$$2x_1 + x_2 + x_3 \leq 5$$
$$x_1, x_2, x_3 \in \mathbb{R}$$

MINIMUM WEIGHTED VERTEX COVER



"COVERS" ALL EDGES

INPUT: GRAPH $G=(V,E)$

WEIGHT FUNCTION $w: V \to \mathbb{R}$

OBJECTIVE: CHOOSE $S \subseteq V$ SUCH THAT FOR $\{u,v\} \in E$: $u \in S \vee v \in S$, MINIMIZING $\sum_{v \in S} w(v)$

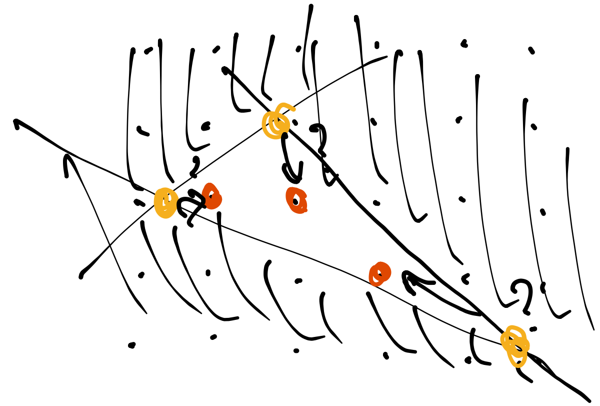
EQUIVALENT FORMULATION AS INTEGER LINEAR PROGRAM:

MINIMIZE $\sum_{v \in V} w(v) \cdot x_v$

SUBJECT TO $x_u + x_v \geq 1$ FOR ALL $\{u,v\} \in E$

$x_v \in \{0,1\}$ INDICATES WHETHER $v \in S$

INTEGER SOLUTIONS SATISFYING LINEAR CONSTR



ROUNDING AN LP SOLUTION

MINIMIZE $\sum_{v \in V} w(v) \cdot x_v$

SUBJECT TO $x_u + x_v \geq 1$ FOR ALL $\{u,v\} \in E$

$\cancel{x_v \in \{0,1\}}$ INDICATES WHETHER $v \in S$

$x_v \geq 0 \qquad x_v \leq 1$

OPTIMAL SOLUTION TO "LP RELAXATION"

$x^* \in [0,1]^n$

$\downarrow$

$\hat{x} \in \{0,1\}^n$

$$\hat{x}_v = \begin{cases} 1 & \text{IF } x_v^* \geq 1/2 \\ 0 & \text{OTHERWISE} \end{cases}$$

ANALYSIS

$$\sum_{v \in V} w(v) \cdot \hat{x}_v \leq \sum_{v \in V} w(v) \cdot 2x_v^*$$

$$= 2 \sum_{v \in V} w(v) x_v^*$$

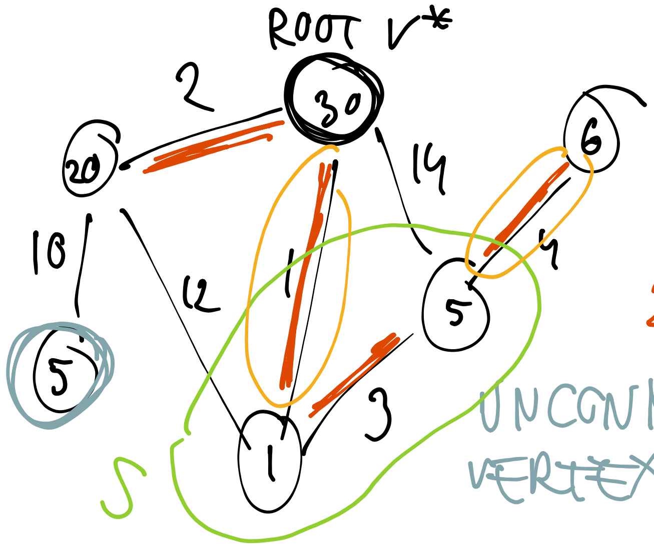
$$\leq 2 \cdot OPT$$

$\leftarrow$ SATISFIES ALL LP CONSTRAINTS AND INTEGRALITY

CAN BE GENERALIZED TO SET COVER:

CHOOSE COLLECTION OF SETS WHOSE UNION COVERS ALL ELEMENTS IN GROUND SET

PRIZE-COLLECTING STEINER TREE PROBLEM



EDGE COST: $2+1+3+4=10$

UNCONNECTED VERTEX COST: 5

GOAL: MINIMIZE TOTAL COST FOR TREE T WITH ROOT v^*

INPUT. $G=(V,E)$
FOR EACH $e\in E$: COST c_e
FOR EACH $v\in V$: COST π_v

OUTPUT: $x_e\in\{0,1\}$ IS e IN T?
$y_v\in\{0,1\}$ IS v IN T

OBJECTIVE: MINIMIZE $\sum_{e\in E} c_e x_e + \sum_{v\in V} \pi_i (1-y_i)$

PAUSE AND THINK: IF x AND y ARE NOT VALID THEN A CONSTRAINT OF THE FORM (*) IS VIOLATED.

CONSTRAINTS TO ENSURE x,y ARE VALID

CONSIDER $\emptyset \subsetneq S \subsetneq V$ AND $v\in S$, $v^*\notin S$

IF $y_v=1$ THEN THERE IS SOME PATH FROM v^* TO v

$\Rightarrow$ $x_e=1$ FOR SOME EDGE CROSSING CUT $(V\setminus S, S)$ ← SET OF SUCH EDGES $\delta(S)$

LINEAR CONSTRAINT: (*) $\sum_{e\in\delta(S)} x_e \geq y_v, \quad \forall S \subsetneq V\setminus\{v^*\}$

LP RELAXATION

ISSUE!
EXPONENTIAL NUMBER OF CONSTRAINTS

MINIMIZE $\sum_{e \in E} c_e x_e + \sum_{v \in V} \pi(v)(1-y_v)$

SUBJECT TO $\sum_{e \in \delta(S)} x_e \geq y_v, \quad \forall S \subseteq V \setminus \{v^*\}, S \neq \emptyset, v \in S$

$y_{v^*} = 1, \quad x_e \in [0,1], \quad y_v \in [0,1]$

MAX-FLOW MIN CUT:
CAN DETERMINE IN POLYNOMIAL TIME IF A CONSTRAINT IS VIOLATED
$\rightarrow$ CAN FIND S

DECISIONS

- CHOOSE VERTICES TO CONNECT $\leftarrow$ TODAY
- CHOOSE EDGES IN TREE $\leftarrow$ STEINER TREE, LATER

ROUNDING:

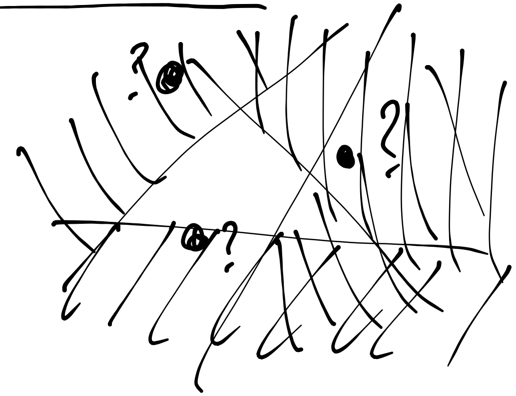
OPT. LP SOLUTION

$$\hat{y}_v = \begin{cases} 1 & \text{if } y_v^* \geq 2/3 \\ 0 & \text{OTHERWISE} \end{cases}$$

SUGGESTS A 3-APPROXIMATION

OBSERVE THAT $\sum_{v \in V} \pi(v)(1-\hat{y}_v) \leq 3 \sum_{v \in V} \pi(v)(1-y_v^*)$

SOLVING LPS WITH A SEPARATION ORACLE



- START WITH NO CONSTRAINTS
- FIND POTENTIAL SOLUTION, ASK ORACLE IF IT VIOLATES A CONSTRAINT
- FIND A POTENTIAL SOLUTION INCLUDING THE NEW CONSTRAINT
- REPEAT

ELLIPSOID METHOD

LPS WITH A SEPARATION ORACLE RUNNING IN POLYNOMIAL TIME CAN BE SOLVED IN POLYNOMIAL TIME.

NOT PRACTICAL.
SOMETIMES USED IN PRACTICE:
ADD CONSTRAINTS "LAZILY"