Gems of TCS

Integer Linear Programming

Sasha Golovnev

October 4, 2021

Avoiding Scurvy

- Orange costs \$1,
  grapefruit costs \$1;
  we have budget of \$2/day
- Orange weighs 100gm,
  grapefruit weighs 200gm,
  we can carry 300gm
- Orange has 100gm of vitamin C,
  grapefruit has 150gm of vitamin C,
  maximize daily vitamin C intake.

Avoiding Scurvy. Plot

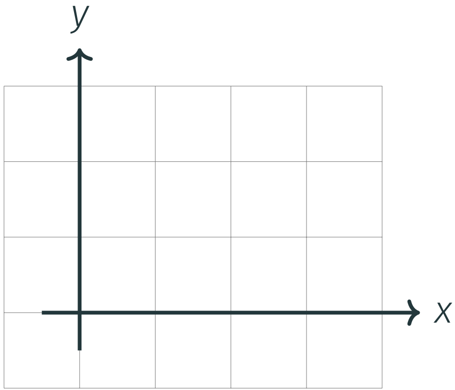
$$\max 2x + 3y$$

$$\begin{aligned} x + y &\leq 2 \\ x + 2y &\leq 3 \\ x &\geq 0 \\ y &\geq 0 \end{aligned}$$

Avoiding Scurvy. Plot

$$\max 2x + 3y$$

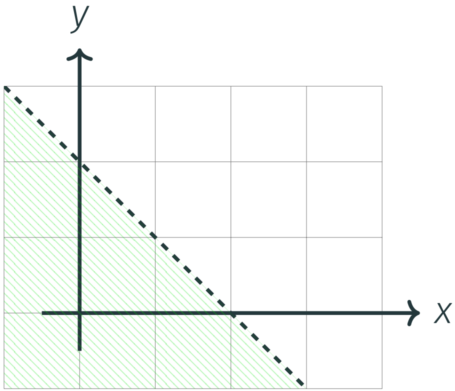
$$x + y \le 2$$
$$x + 2y \le 3$$
$$x \ge 0$$
$$y \ge 0$$



Avoiding Scurvy. Plot

$$\max 2x + 3y$$

$$x + y \leq 2$$
$$x + 2y \leq 3$$
$$x \geq 0$$
$$y \geq 0$$



Avoiding Scurvy. Plot

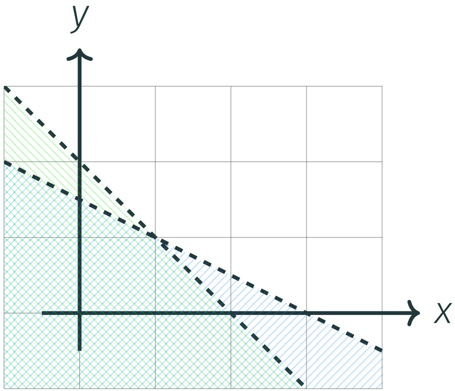
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 3$$

$$x \geq 0$$

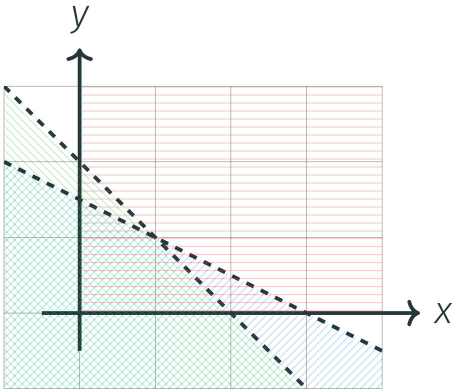
$$y \geq 0$$



Avoiding Scurvy. Plot

$$\max 2x + 3y$$

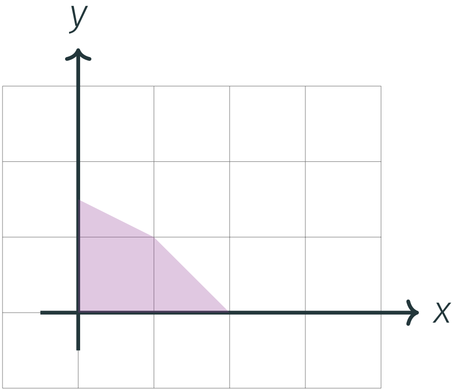
$$x + y \leq 2$$
$$x + 2y \leq 3$$
$$x \geq 0$$
$$y \geq 0$$



Avoiding Scurvy. Plot

$$\max 2x + 3y$$

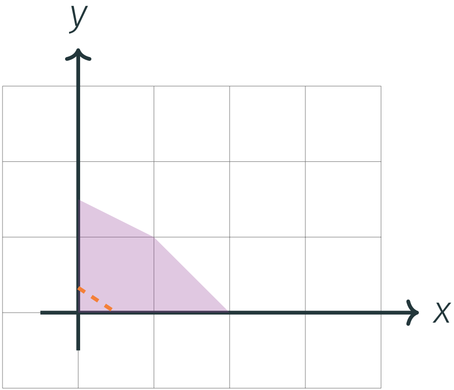
$$x + y \leq 2$$
$$x + 2y \leq 3$$
$$x \geq 0$$
$$y \geq 0$$



Avoiding Scurvy. Plot

$\max 2x + 3y$

$$x + y \leq 2$$
$$x + 2y \leq 3$$
$$x \geq 0$$
$$y \geq 0$$



Avoiding Scurvy. Plot

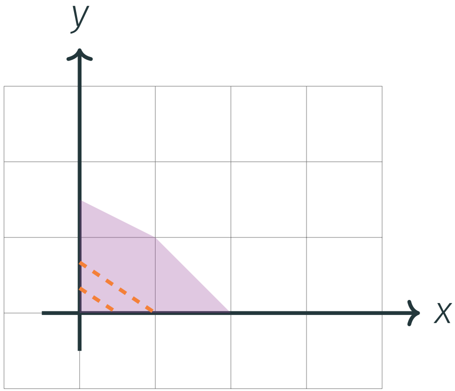
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 3$$

$$x \geq 0$$

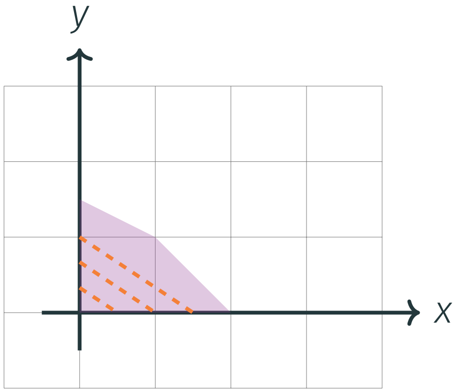
$$y \geq 0$$



Avoiding Scurvy. Plot

$$\max 2x + 3y$$

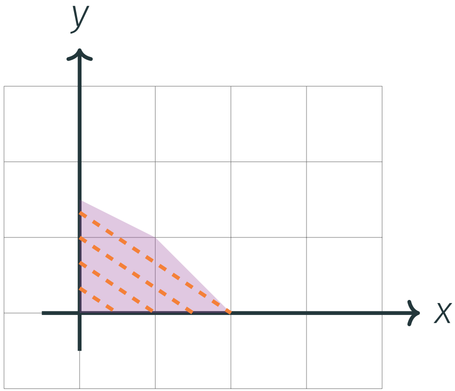
$$x + y \leq 2$$
$$x + 2y \leq 3$$
$$x \geq 0$$
$$y \geq 0$$



Avoiding Scurvy. Plot

$$\max 2x + 3y$$

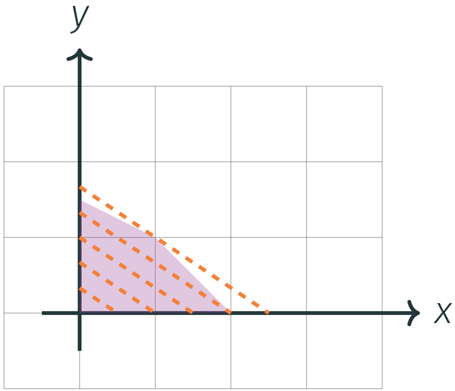
$$x + y \le 2$$
$$x + 2y \le 3$$
$$x \ge 0$$
$$y \ge 0$$



Avoiding Scurvy. Plot

$$\max 2x + 3y$$

$$x + y \leq 2$$
$$x + 2y \leq 3$$
$$x \geq 0$$
$$y \geq 0$$



Avoiding Scurvy II

$$\max 2x + 3y$$

$$\begin{aligned} x + y &\leq 2 \\ x + 2y &\leq 2.5 \\ x &\geq 0 \\ y &\geq 0 \end{aligned}$$

Avoiding Scurvy II

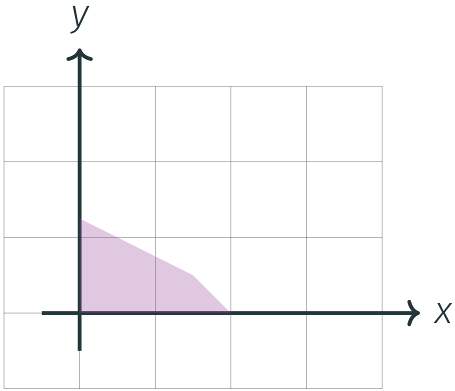
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 2.5$$

$$x \geq 0$$

$$y \geq 0$$



Avoiding Scurvy II

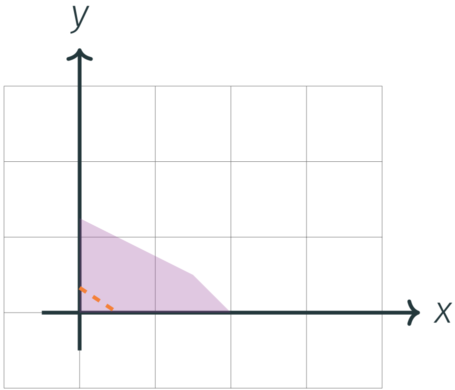
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 2.5$$

$$x \geq 0$$

$$y \geq 0$$



Avoiding Scurvy II

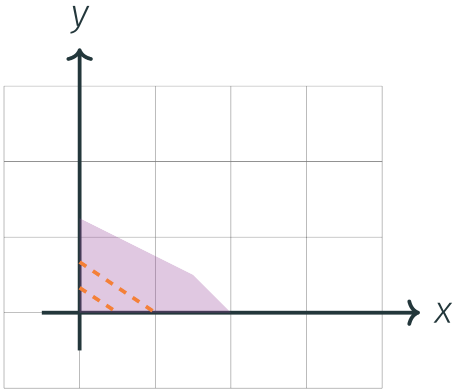
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 2.5$$

$$x \geq 0$$

$$y \geq 0$$



Avoiding Scurvy II

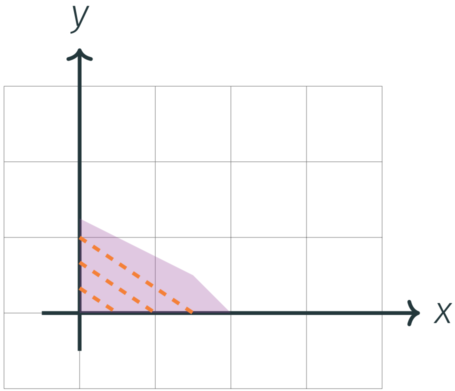
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 2.5$$

$$x \geq 0$$

$$y \geq 0$$



Avoiding Scurvy II

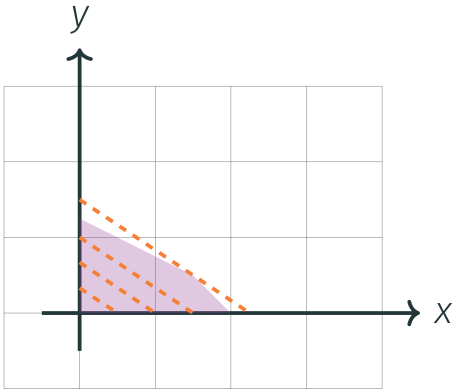
$$\max 2x + 3y$$

$$x + y \leq 2$$

$$x + 2y \leq 2.5$$

$$x \geq 0$$

$$y \geq 0$$



Linear programming

Input: A set of linear inequalities $Ax \leq b$.

Output: Real solution that optimizes the objective function.

Integer linear programming

Input: A set of linear inequalities $\mathbf{Ax} \leq \mathbf{b}$.

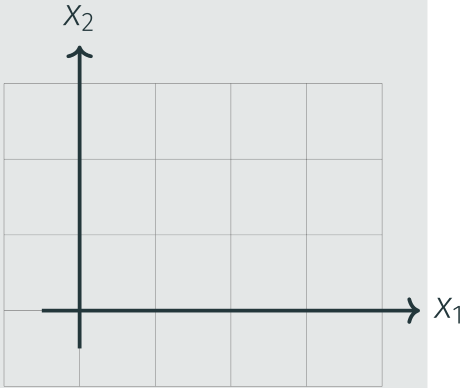
Output: Integer solution that optimizes the objective function.

Example

$$x_1 \geq 0.5$$
$$-x_1 + 8x_2 \geq 0$$
$$-x_1 - 8x_2 \geq -8$$

Example

$$x_1 \geq 0.5$$
$$-x_1 + 8x_2 \geq 0$$
$$-x_1 - 8x_2 \geq -8$$

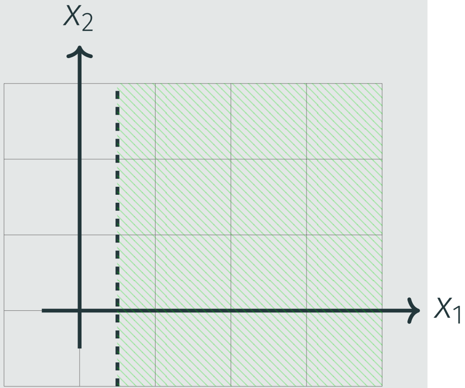


Example

$$x_1 \geq 0.5$$

$$-x_1 + 8x_2 \geq 0$$

$$-x_1 - 8x_2 \geq -8$$

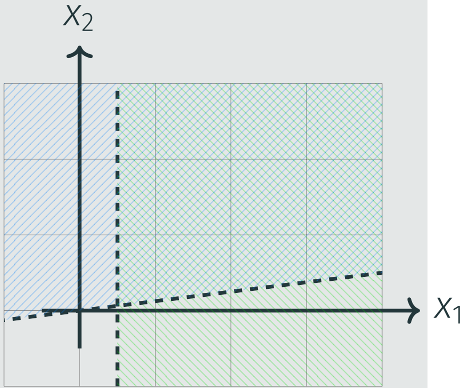


Example

$$x_1 \geq 0.5$$

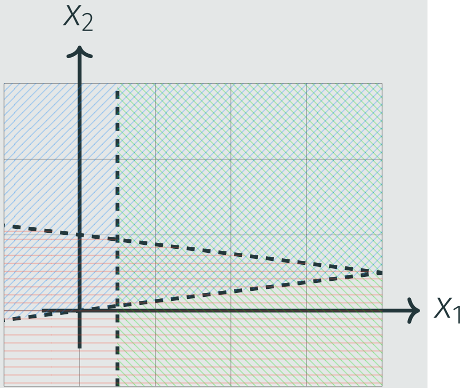
$$-x_1 + 8x_2 \geq 0$$

$$-x_1 - 8x_2 \geq -8$$



Example

$$x_1 \geq 0.5$$
$$-x_1 + 8x_2 \geq 0$$
$$-x_1 - 8x_2 \geq -8$$



LP

Find a real
solution of a system of
linear inequalities

LP

Find a real
solution of a system of
linear inequalities

Can be solved
efficiently (Lecture 9)

LP

Find a real solution of a system of linear inequalities

Can be solved efficiently (Lecture 9)

ILP

Find an integer solution of a system of linear inequalities

LP	ILP
Find a real solution of a system of linear inequalities	Find an integer solution of a system of linear inequalities
Can be solved efficiently (Lecture 9)	No polynomial algorithm known!

Algorithm for ILP

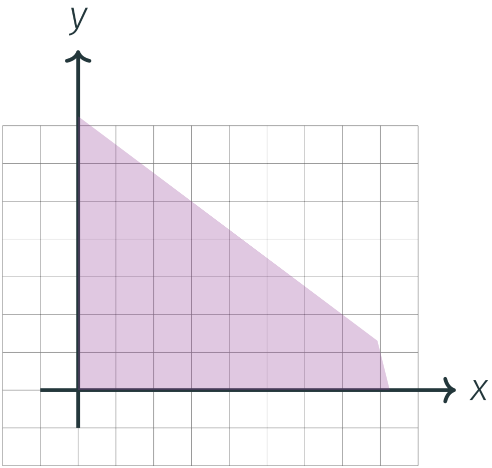
$$\max 2x + y$$

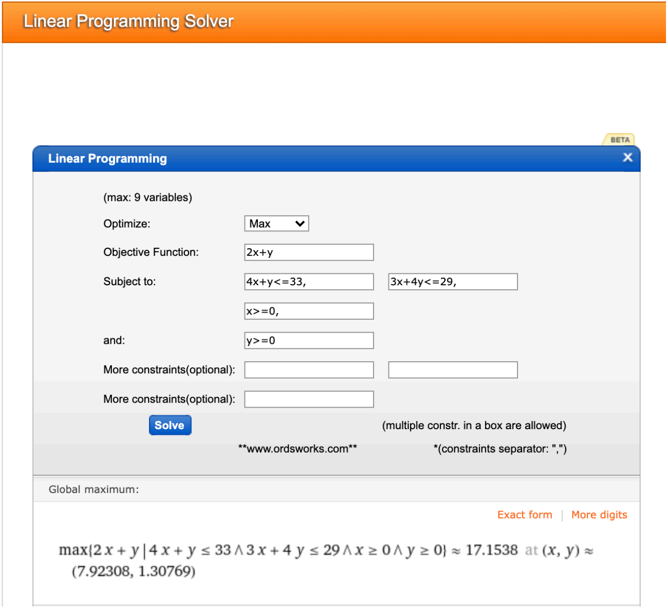
$$\begin{aligned} 4x + y &\leq 33 \\ 3x + 4y &\leq 29 \\ x &\geq 0 \\ y &\geq 0 \\ x, y &\in \mathbb{Z} \end{aligned}$$

Algorithm for ILP

$$\max 2x + y$$

$$4x + y \leq 33$$
$$3x + 4y \leq 29$$
$$x \geq 0$$
$$y \geq 0$$
$$x, y \in \mathbb{Z}$$

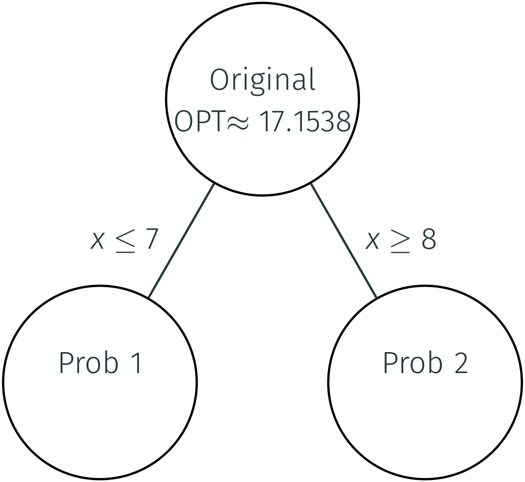




Linear Programming Solver
BETA
Linear Programming
(max: 9 variables)
Optimize:
Max
Objective Function:
2x+y
Subject to:
4x+y<=33,
3x+4y<=29,
x>=0,
and:
y>=0
More constraints(optional):
More constraints(optional):
Solve
(multiple constr. in a box are allowed)
www.ordsworks.com
*(constraints separator: ",")
Global maximum:
Exact form | More digits
max{2x + y | 4x + y ≤ 33 ∧ 3x + 4y ≤ 29 ∧ x ≥ 0 ∧ y ≥ 0} ≈ 17.1538 at (x, y) ≈ (7.92308, 1.30769)

Branching on x

Branching on x



Branching

$$\max 2x + y$$

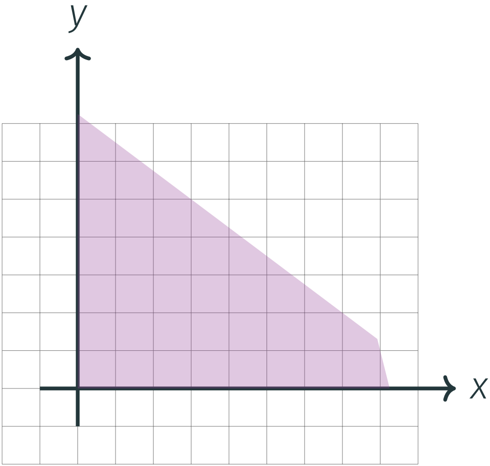
$$4x + y \leq 33$$

$$3x + 4y \leq 29$$

$$x \geq 0$$

$$y \geq 0$$

$$x, y \in \mathbb{Z}$$



Branching

$$\max 2x + y$$

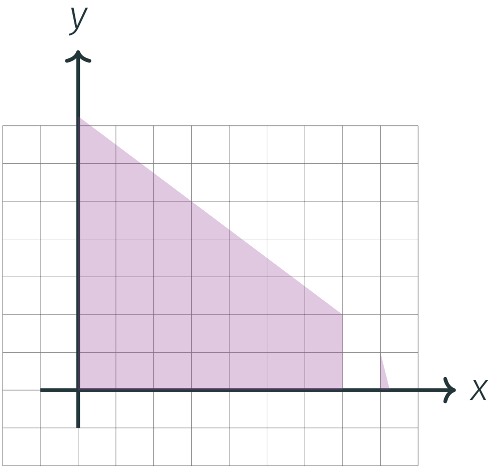
$$4x + y \leq 33$$

$$3x + 4y \leq 29$$

$$x \geq 0$$

$$y \geq 0$$

$$x, y \in \mathbb{Z}$$



Linear Programming Solver

BETA

Linear Programming

(max: 9 variables)

Optimize:	Max	
Objective Function:	2x+y	
Subject to:	4x+y<=33,	3x+4y<=29,
	x>=0,	
and:	y>=0,	
More constraints(optional):	x<=7	
More constraints(optional):		

Solve (multiple constr. in a box are allowed)

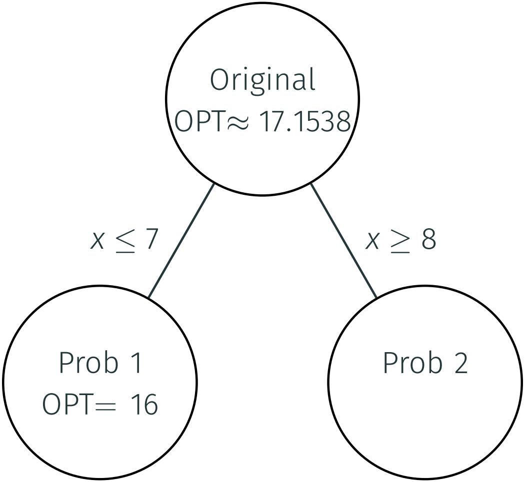
www.ordsworks.com *(constraints separator: ",")

Global maximum:

$$\max\{2x + y \mid 4x + y \leq 33 \wedge 3x + 4y \leq 29 \wedge x \geq 0 \wedge y \geq 0 \wedge x \leq 7\} = 16$$

at $(x, y) = (7, 2)$

Branching on x



Linear Programming Solver

BETA

Linear Programming

(max: 9 variables)

Optimize:	Max	
Objective Function:	2x+y	
Subject to:	4x+y<=33,	3x+4y<=29,
	x>=0,	
and:	y>=0,	
More constraints(optional):	x>=8	
More constraints(optional):		

Solve (multiple constr. in a box are allowed)

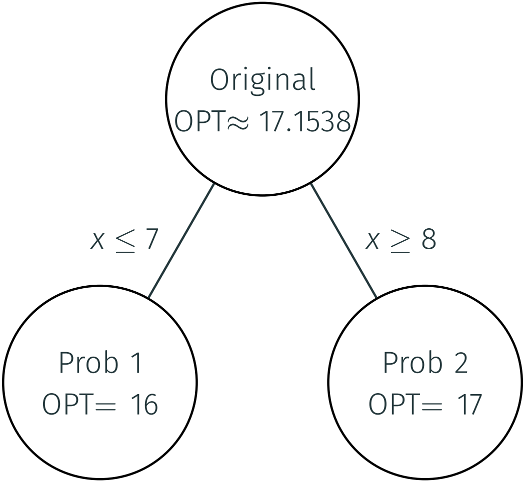
www.ordsworks.com *(constraints separator: ",")

Global maximum:

$$\max\{2x + y \mid 4x + y \leq 33 \land 3x + 4y \leq 29 \land x \geq 0 \land y \geq 0 \land x \geq 8\} = 17$$

at $(x, y) = (8, 1)$

Branching on x



Heuristic Algorithms for ILP

Applications

Applications

- Scheduling
- Planning
- Networks
- ...

Vertex Covers

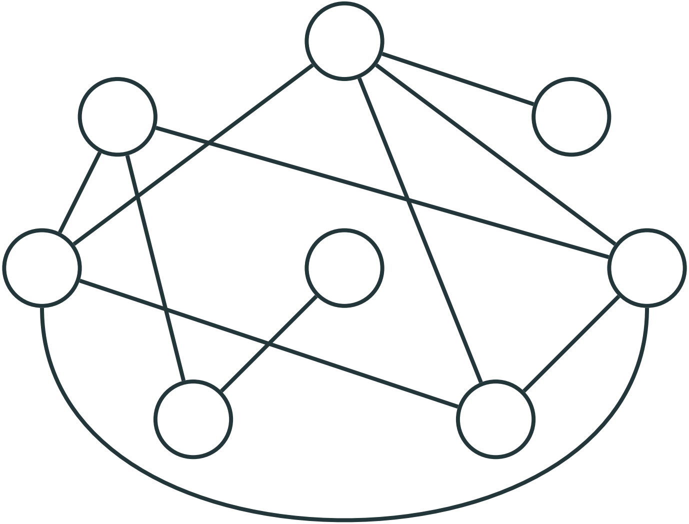
- A Vertex Cover of a graph G is a set of vertices C such that every edge of G is connected to some vertex in C.

Vertex Covers

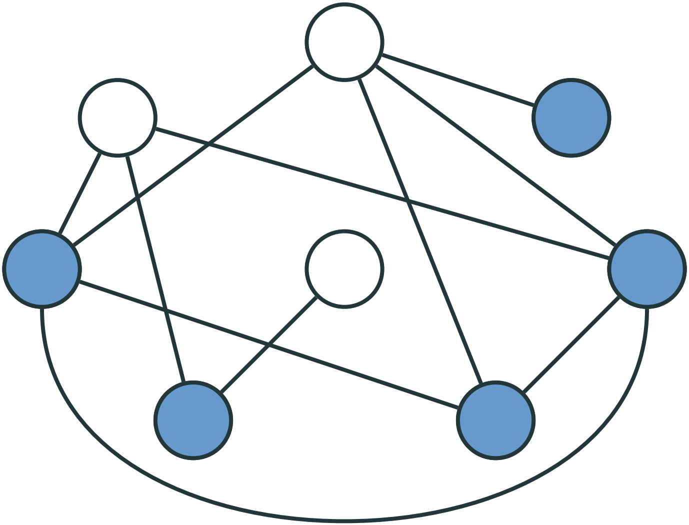
- A Vertex Cover of a graph G is a set of vertices C such that every edge of G is connected to some vertex in C.

- A Minimum Vertex Cover is a vertex cover of the smallest size.

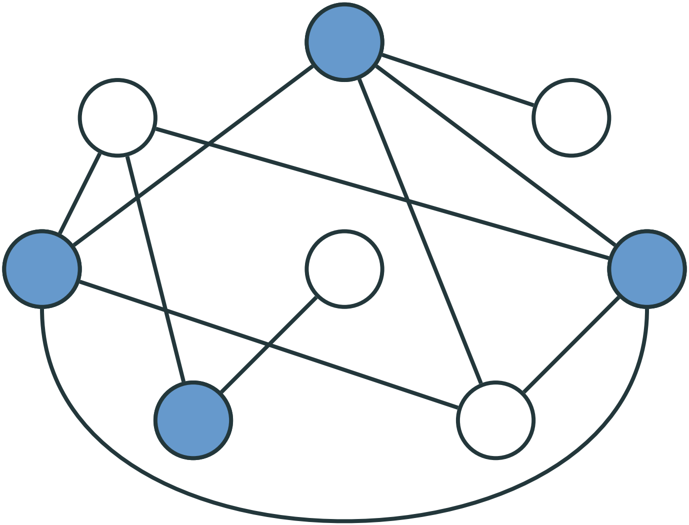
Vertex Covers: Examples



Vertex Covers: Examples



Vertex Covers: Examples



Vertex Cover as ILP

- Introduce binary variable for every vertex: $x_1, \ldots, x_n$:
    - $x_i = 1$ iff x_i belongs to Vertex Cover

Vertex Cover as ILP

- Introduce binary variable for every vertex: $x_1, \ldots, x_n$:
  - $x_i = 1$ iff x_i belongs to Vertex Cover
- $\forall i \in \{1, \ldots, n\},\ 0 \leq x_i \leq 1, x_i \in \mathbb{Z}$

Vertex Cover as ILP

- Introduce binary variable for every vertex: $x_1, \ldots, x_n$:
  - $x_i = 1$ iff x_i belongs to Vertex Cover
- $\forall i \in \{1, \ldots, n\},\ 0 \leq x_i \leq 1, x_i \in \mathbb{Z}$
- $\min \sum_i x_i$

Vertex Cover as ILP

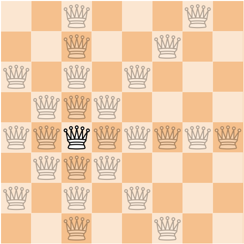
- Introduce binary variable for every vertex: $x_1, \ldots, x_n$:
  - $x_i = 1$ iff x_i belongs to Vertex Cover
- $\forall i \in \{1, \ldots, n\},\ 0 \leq x_i \leq 1, x_i \in \mathbb{Z}$
- $\min \sum_i x_i$
- For every edge (u, v) in th graph: $x_u + x_v \geq 1$

Implementation

```python
import networkx as nx
from mip import *

g = nx.Graph()
g.add_edges_from([(1, 2), (1, 3), (1, 5), (1, 6), (2, 5), (2, 0),
                  (3, 4), (3, 5), (3, 6), (5, 6), (7, 0)])

m = Model()
n = g.number_of_nodes()
x = [m.add_var(var_type=BINARY) for i in range(n)]
for u, v in g.edges():
    m += x[u]+x[v] >= 1
m.objective = minimize(xsum(x[i] for i in range(n)))
m.optimize()

selected = [i for i in range(n) if x[i].x >= 0.99]
print("selected items: {}".format(selected))
```

N Queens

Is it possible to place n queens on an $n \times n$ board such that no two of them attack each other?



N Queens as ILP

- n^2 0/1-variables: for $0 \le i, j < n$, $x_{ij} = 1$ iff queen is placed into cell (i, j)

N Queens as ILP

- n^2 0/1-variables: for $0 \leq i, j < n$, $x_{ij} = 1$ iff queen is placed into cell (i, j)
- For $0 \leq i < n$, ith row contains $= 1$ queen:

$$\sum_{j=1}^{n} x_{ij} = 1\,.$$

N Queens as ILP

- n^2 0/1-variables: for $0 \leq i, j < n$, $x_{ij} = 1$ iff queen is placed into cell (i, j)
- For $0 \leq i < n$, ith row contains $= 1$ queen:

$$\sum_{j=1}^{n} x_{ij} = 1.$$

- For $0 \leq j < n$, jth column contains $= 1$ queen:

$$\sum_{j=1}^{n} x_{ij} = 1.$$

N Queens as ILP

- n^2 0/1-variables: for $0 \le i, j < n$, $x_{ij} = 1$ iff queen is placed into cell (i, j)
- For $0 \le i < n$, ith row contains $= 1$ queen:
$$\sum_{j=1}^{n} x_{ij} = 1 .$$
- For $0 \le j < n$, jth column contains $= 1$ queen:
$$\sum_{j=1}^{n} x_{ij} = 1 .$$
- Each diagonal contains ≤ 1 queen:
$$\sum_{i=1}^{n} \sum_{j=1:\ i-j=k}^{n} x_{ij} \le 1 ; \quad \sum_{i=1}^{n} \sum_{j=1:\ i+j=k}^{n} x_{ij} \le 1$$