Linear Programming

Linear Programming, LP Duality Theorem, LP-Relaxation

Linear Program (LP)

- A set of linear equations/inequalities.
- Maximize or minimize a given linear objective function.

$$
\begin{aligned}
\text{maximize} \quad & c_1x_1 + c_2x_2 + \cdots c_nx_n \\
\text{subject to} \quad & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1 \\
& a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2 \\
& \qquad\qquad\qquad \vdots \\
& a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \le b_m \\
& x_1, x_2, \dots, x_n \ge 0
\end{aligned}
$$

Example

- Suppose a factory can produce two kinds of products: oil and sugar.
- Profit for 1 tons of sugar: 1
- Profit for 1 tons of oil: 6
- Limited resources, can produce at most
  - 200 tons of sugar
  - 300 tons of oil
  - Overall weight is at most 400 tons
- Problem: maximize the profit

$$
\begin{aligned}
\text{maximize} \quad & x_1 + 6x_2 \\
\text{subject to} \quad & x_1 \leq 200 \\
& x_2 \leq 300 \\
& x_1 + x_2 \leq 400 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Feasible Region

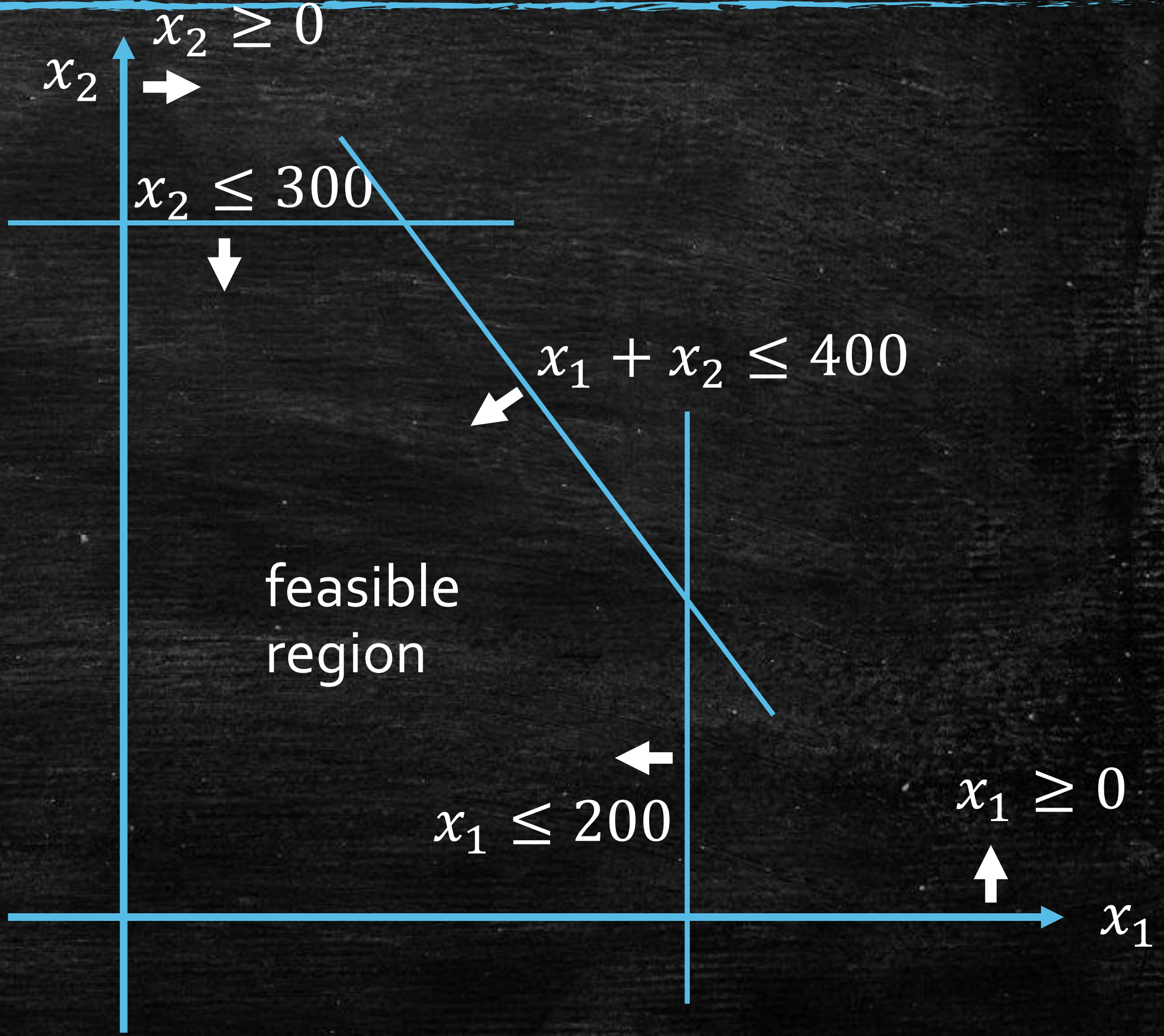
maximize $x_1 + 6x_2$

subject to $x_1 \leq 200$

$x_2 \leq 300$

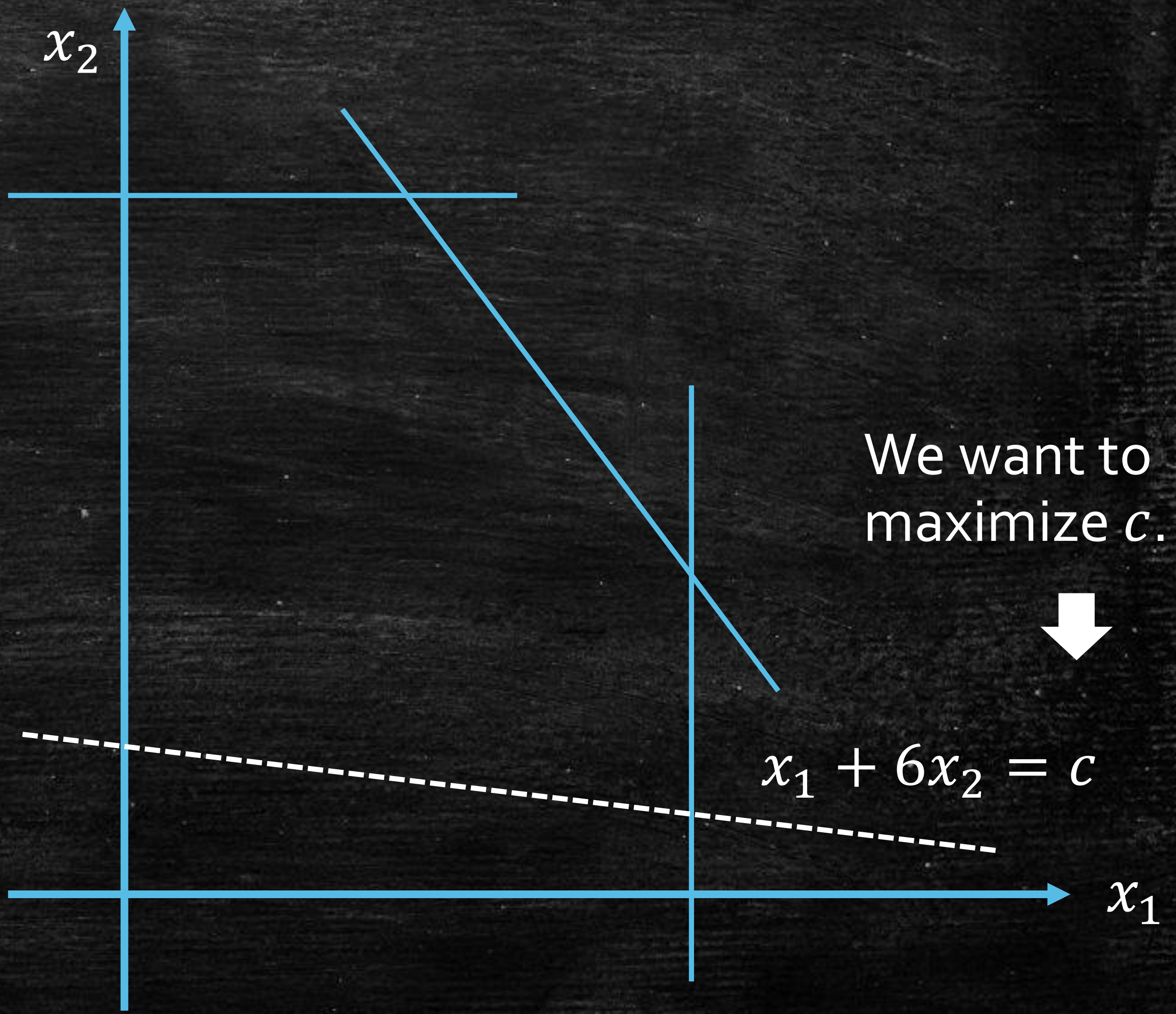
$x_1 + x_2 \leq 400$

$x_1, x_2 \geq 0$



Maximizing the Objective

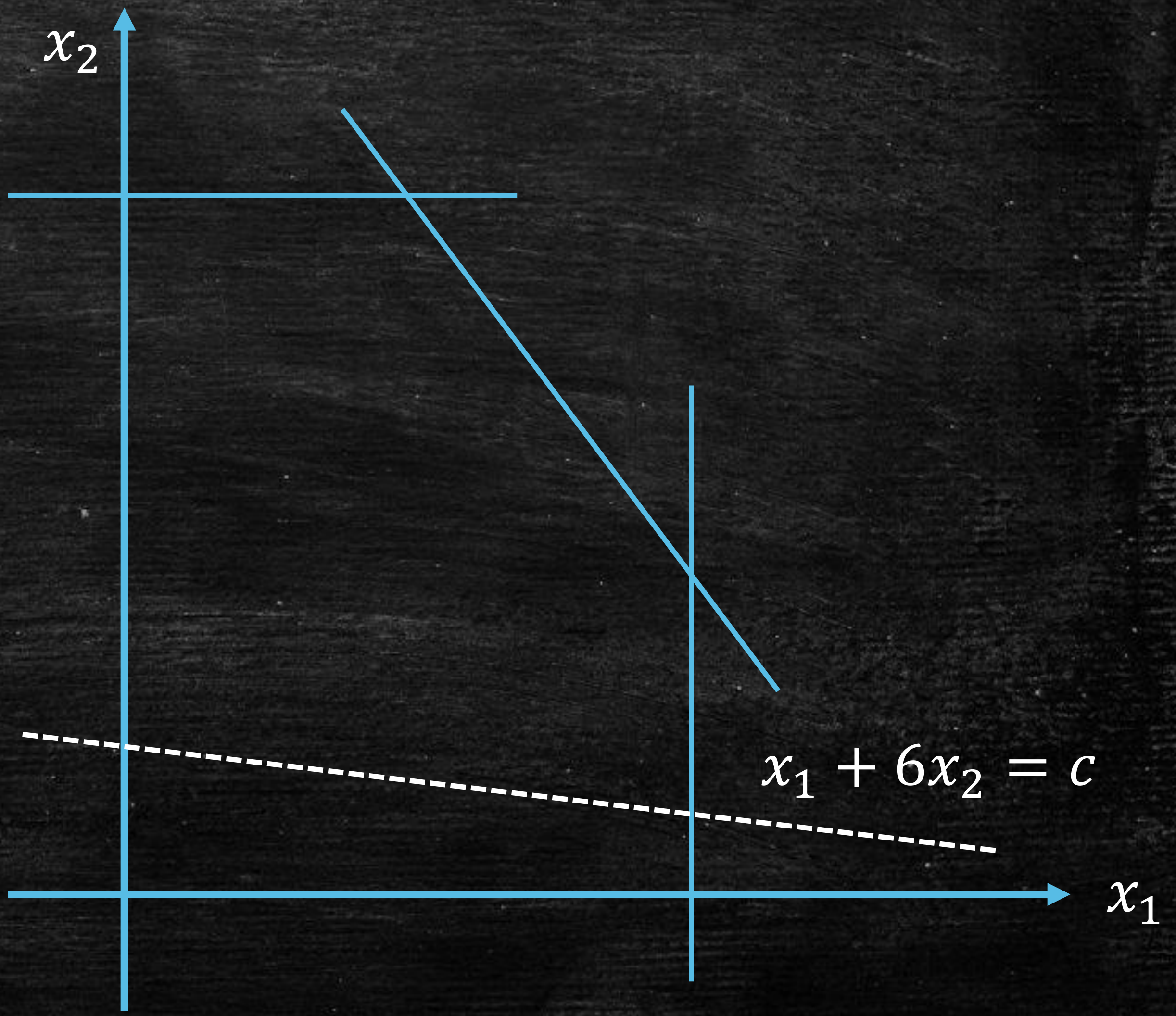
$$\begin{aligned} \text{maximize} \quad & x_1 + 6x_2 \\ \text{subject to} \quad & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$



We want to maximize c.

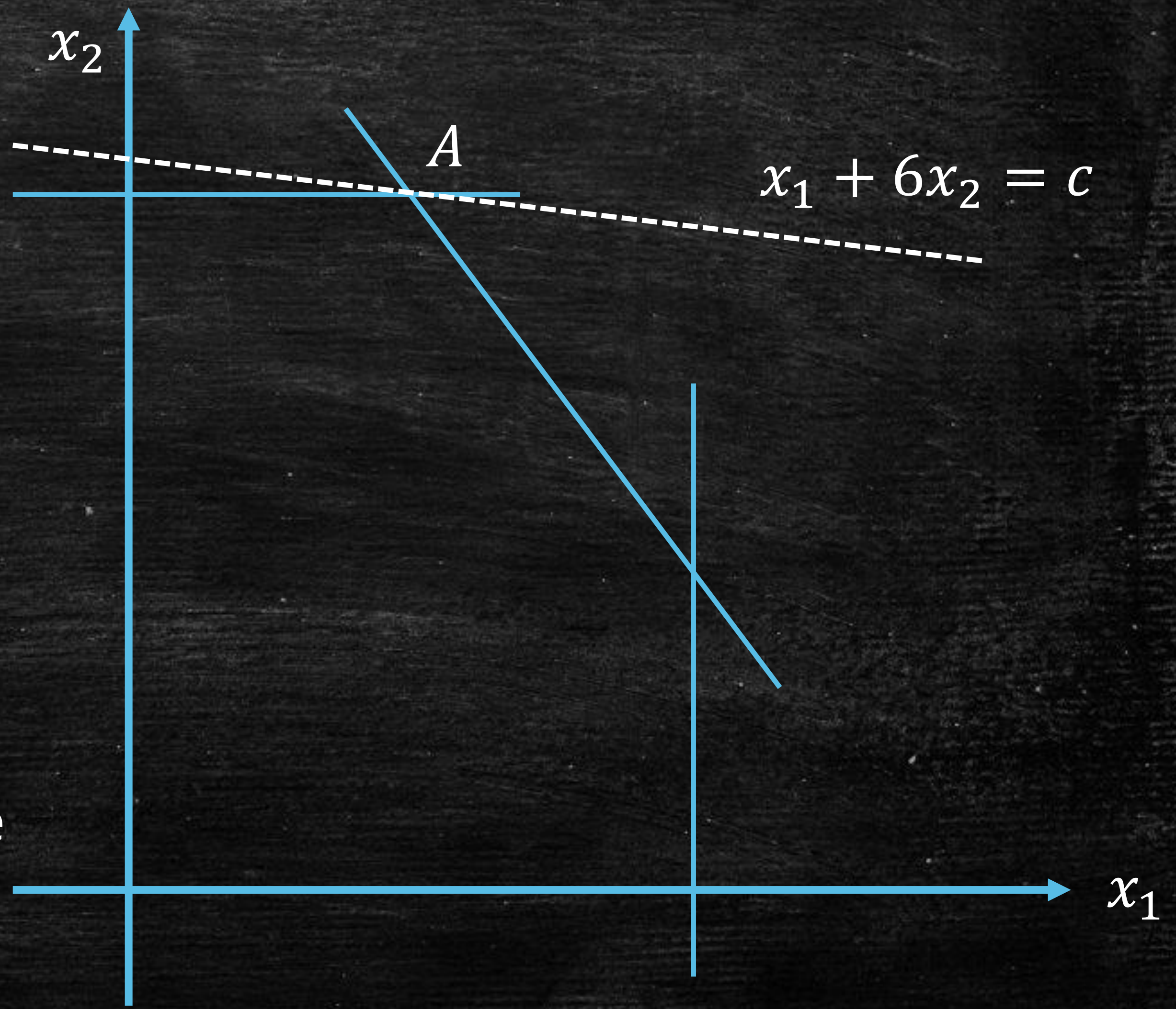
Maximizing the Objective

$$\begin{aligned} \text{maximize} \quad & x_1 + 6x_2 \\ \text{subject to} \quad & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$



Maximizing the Objective

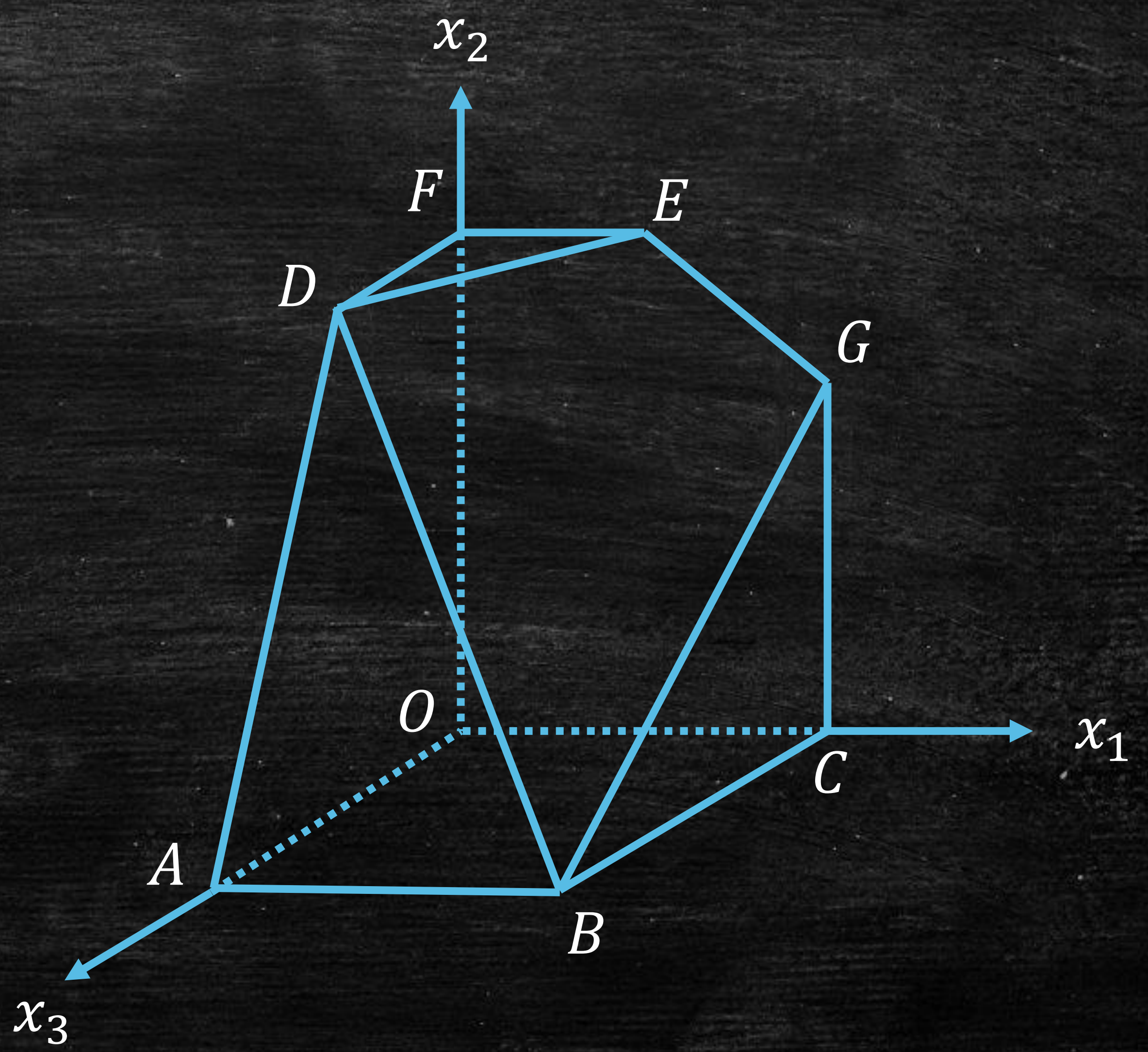
$$
\begin{aligned}
\text{maximize}\quad & x_1 + 6x_2 \\
\text{subject to}\quad & x_1 \le 200 \\
& x_2 \le 300 \\
& x_1 + x_2 \le 400 \\
& x_1, x_2 \ge 0
\end{aligned}
$$



Optimum is obtained at vertex A, where $(x_1, x_2) = (100, 300)$ and $c = 1900$.

Another Example with Three variables

$$
\begin{array}{ll}
\text{maximize} & x_1 + 6x_2 + 13x_3 \\
\text{subject to} & x_1 \leq 200 \\
& x_2 \leq 300 \\
& x_1 + x_2 + x_3 \leq 400 \\
& x_2 + 3x_3 \leq 600 \\
& x_1, x_2, x_3 \geq 0
\end{array}
$$



Another Example with Three variables

maximize $x_1 + 6x_2 + 13x_3$

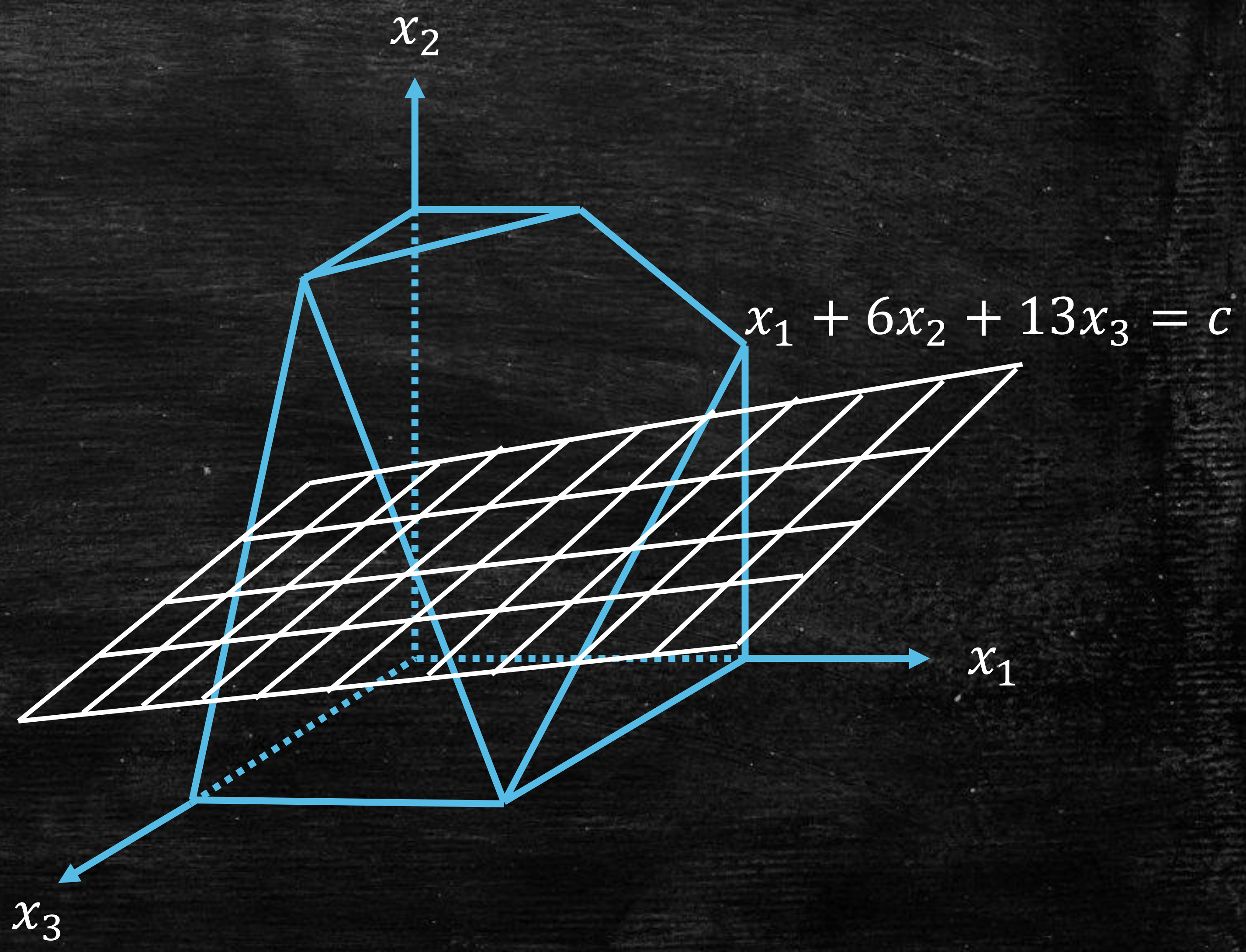
subject to $x_1 \le 200$

$x_2 \le 300$

$x_1 + x_2 + x_3 \le 400$

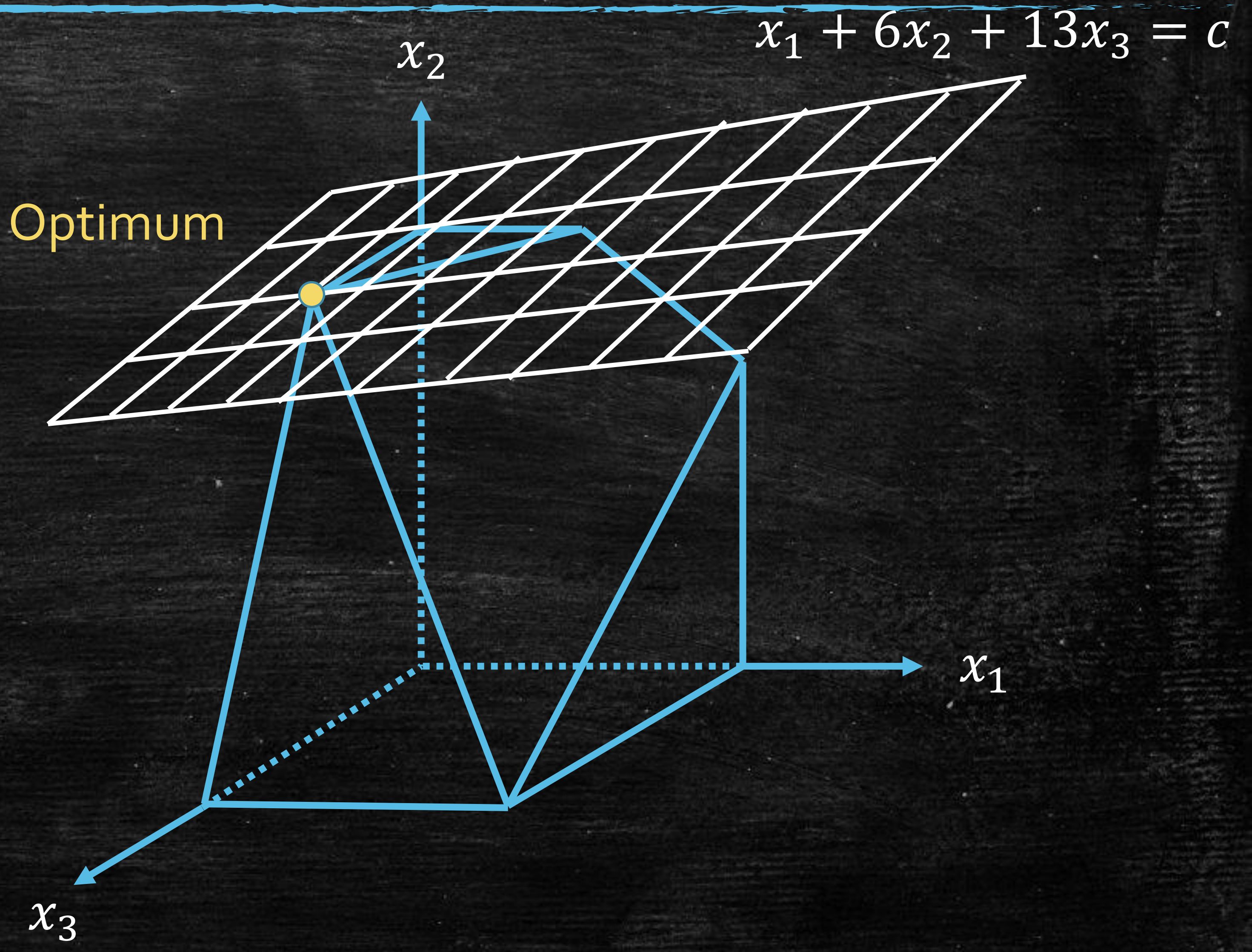
$x_2 + 3x_3 \le 600$

$x_1, x_2, x_3 \ge 0$



Another Example with Three variables

$$\begin{aligned} \text{maximize} \quad & x_1 + 6x_2 + 13x_3 \\ \text{subject to} \quad & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 + x_3 \leq 400 \\ & x_2 + 3x_3 \leq 600 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$



Important Observations

1. There always exists an optimum $\boldsymbol{x} = (x_1, \ldots, x_n)$ at a vertex of the polytope.
   - Linear objective $\Longrightarrow c = c_1x_1 + \cdots + c_nx_n$ is a hyperplane.
   - Optimum is obtained only when the whole feasible region is below the hyperplane and the hyperplane "barely" intersect the region by a point.
2. The feasible region is always convex.
   - Linear Constraints $\Longrightarrow$ feasible region is bounded by hyperplanes.
3. A local maximum is also a global maximum.
   - By the convexity of the feasible region…

Simplex Method

- Choose an arbitrary starting vertex.
- Iteratively move to an adjacent vertex along an edge if such movement increase the objective.
- Terminate when we reach a local maximum.

Starting Point

maximize $x_1 + 6x_2$

subject to $x_1 \leq 200$ 1

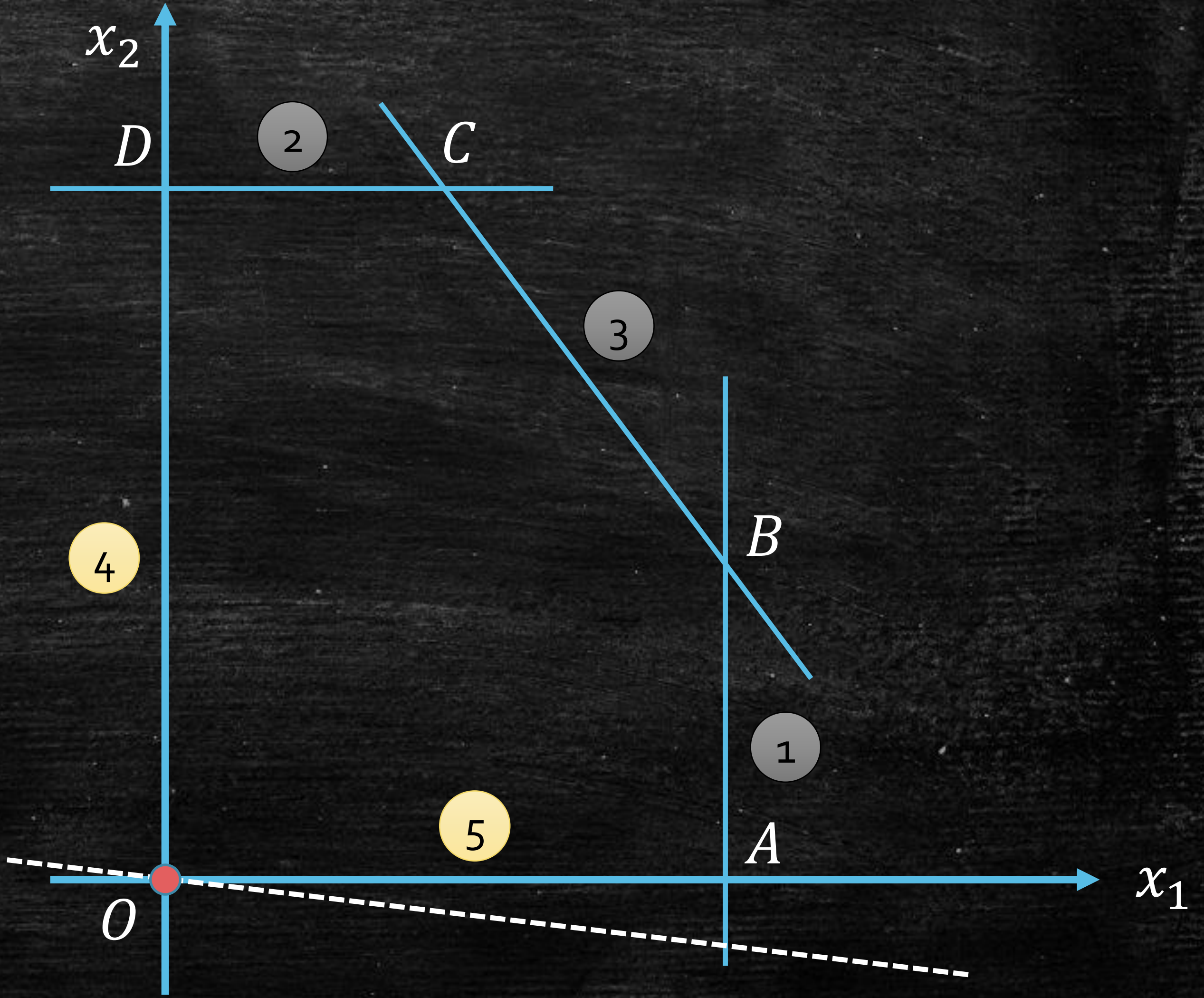
$x_2 \leq 300$ 2

$x_1 + x_2 \leq 400$ 3

$x_1 \geq 0$ 4

$x_2 \geq 0$ 5

Starting from vertex O.



Moving

maximize $x_1 + 6x_2$

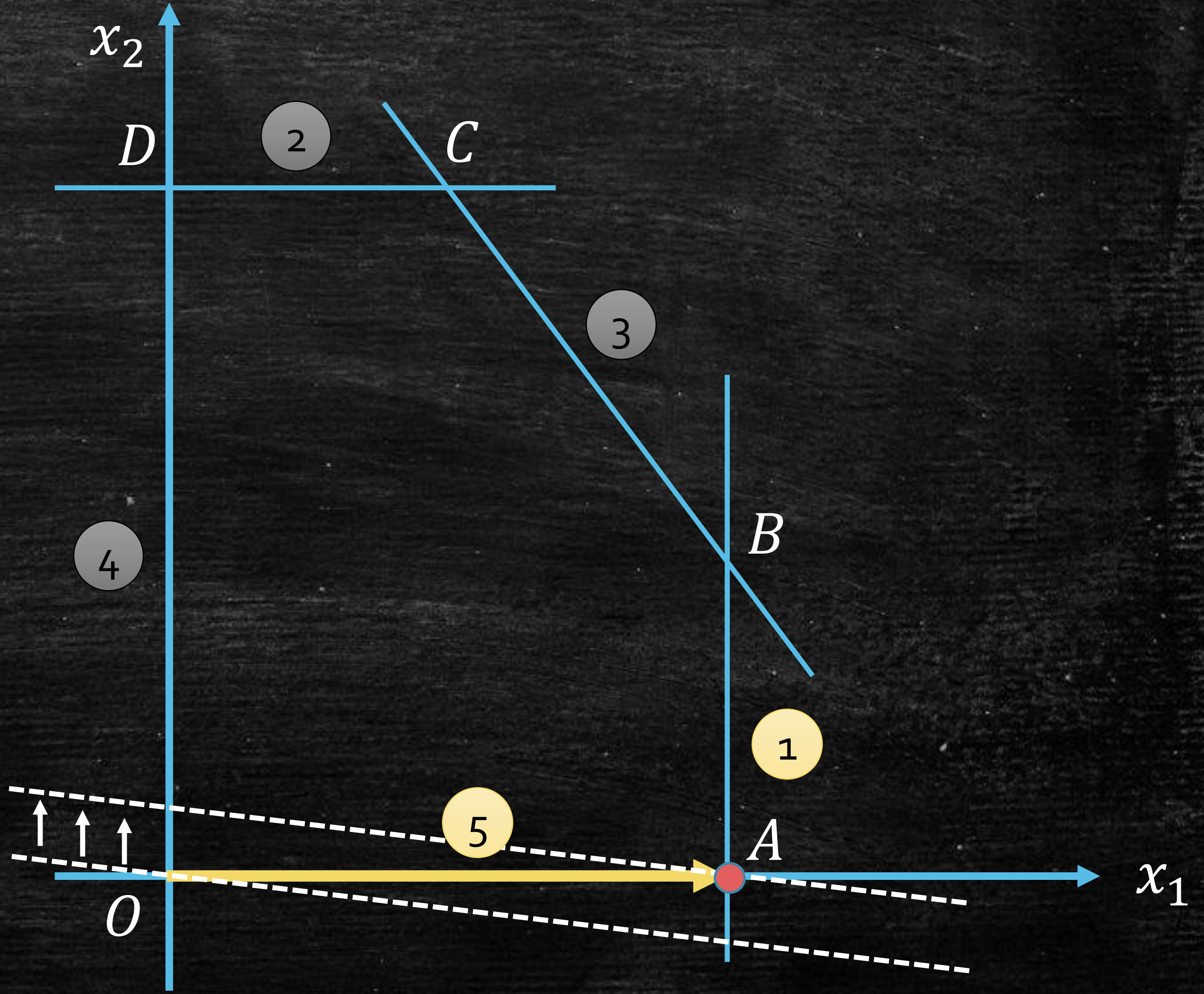
subject to $x_1 \leq 200$ 1

$x_2 \leq 300$ 2

$x_1 + x_2 \leq 400$ 3

$x_1 \geq 0$ 4

$x_2 \geq 0$ 5



Moving from O to A increases the objective.

Rewrite LP: View A as Origin

maximize $x_1 + 6x_2$

subject to $x_1 \leq 200$ 1

$\rightarrow y_1 \geq 0$ 1

$x_2 \leq 300$ 2

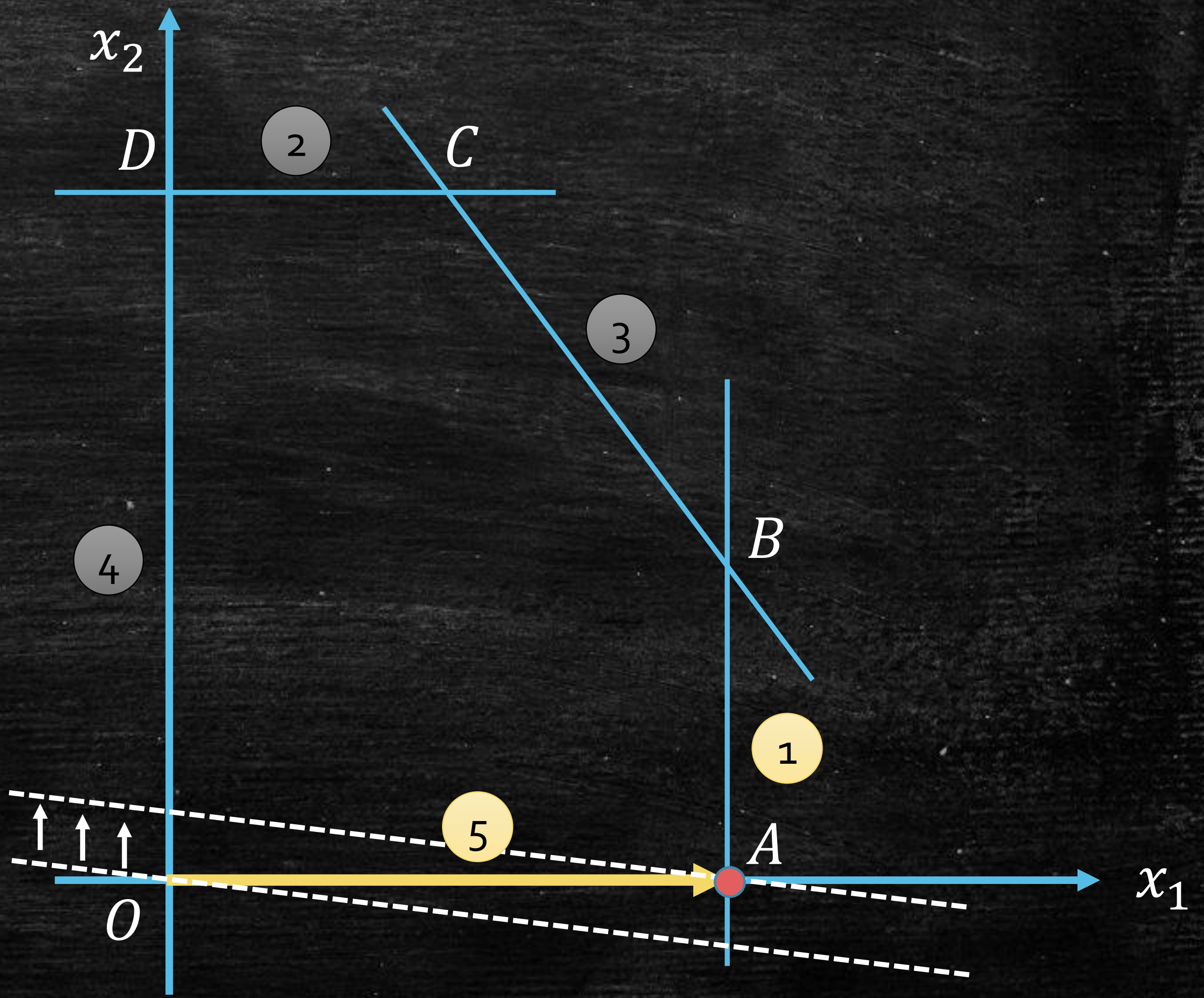
$x_1 + x_2 \leq 400$ 3

$x_1 \geq 0$ 4

$x_2 \geq 0$ 5

$\rightarrow y_2 \geq 0$ 5

Moving from O to A increases the objective.



Rewrite LP: View A as Origin

$\cancel{\text{maximize } x_1 + 6x_2}$

maximize $200 - y_1 + 6y_2$?

subject to $x_1 \leq 200$ 1

$\rightarrow y_1 \geq 0$ 1

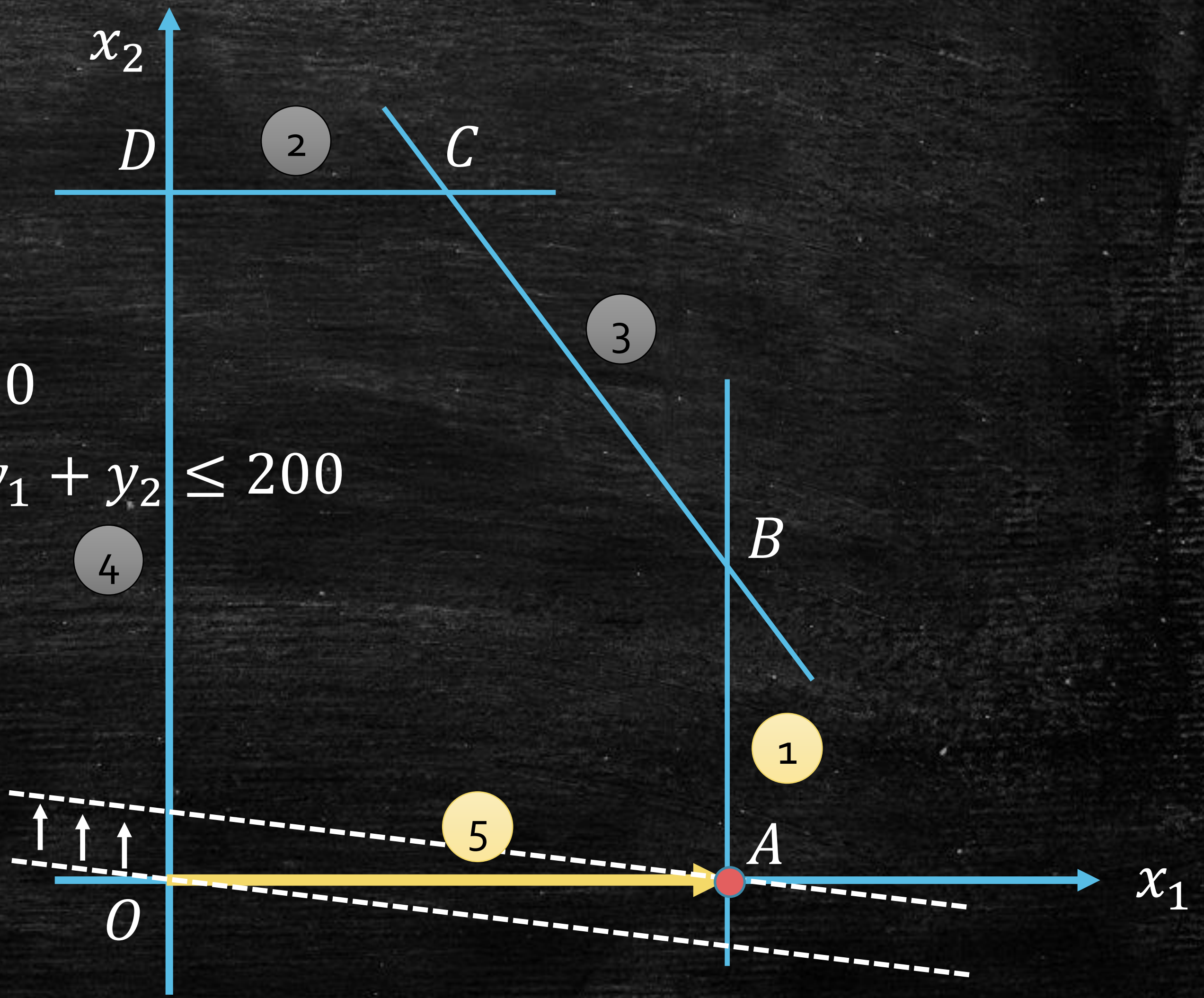
2 $x_2 \leq 300 \rightarrow y_2 \leq 300$

3 $x_1 + x_2 \leq 400 \rightarrow -y_1 + y_2 \leq 200$

4 $x_1 \geq 0 \rightarrow y_1 \leq 200$

$x_2 \geq 0$ 5

$\rightarrow y_2 \geq 0$ 5



Moving from O to A increases the objective.

Rewrite LP: View A as Origin

maximize $200 - y_1 + 6y_2$

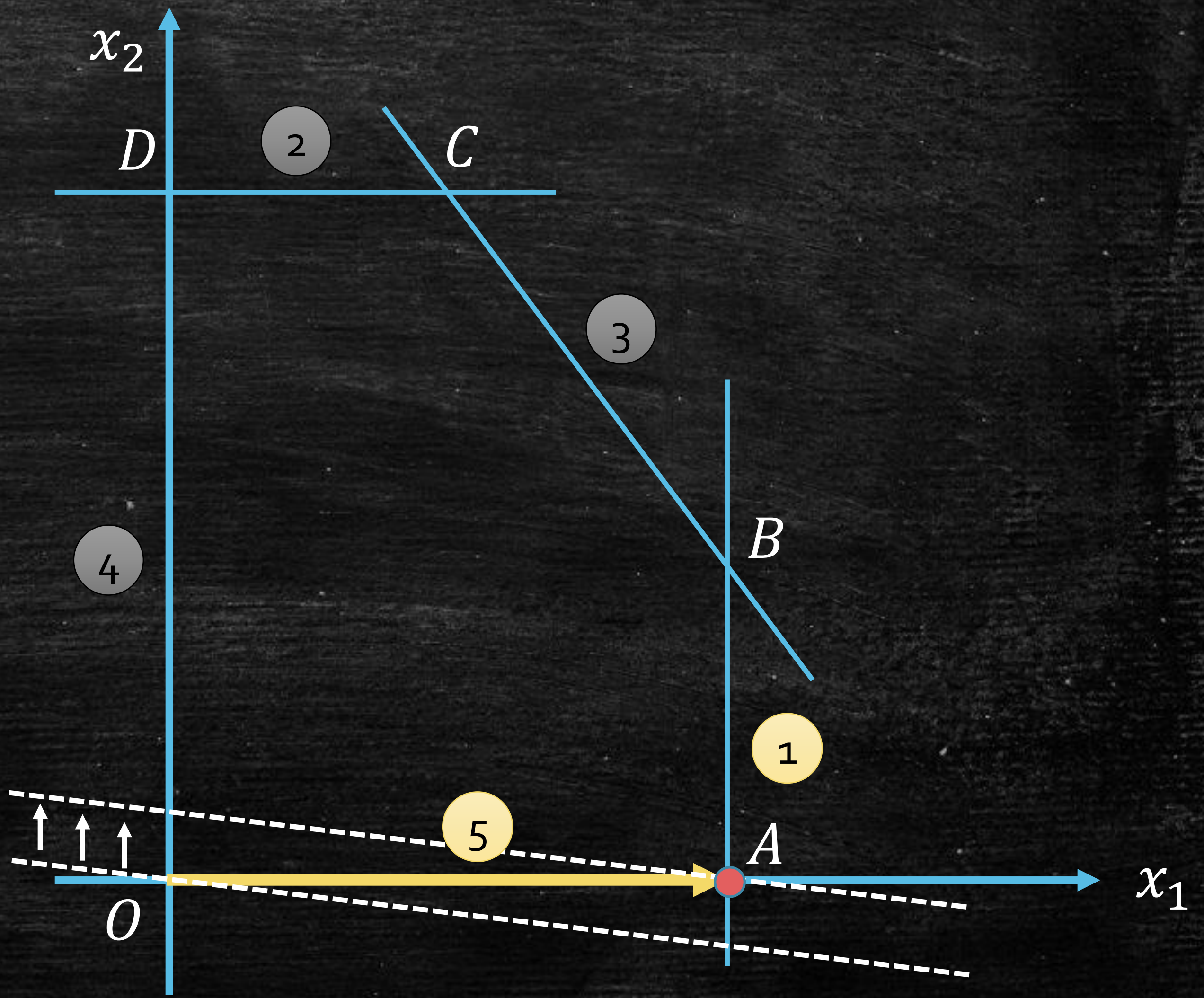
subject to $y_1 \geq 0$ ①

$y_2 \leq 300$ ②

$-y_1 + y_2 \leq 200$ ③

$y_1 \leq 200$ ④

$y_2 \geq 0$ ⑤



Moving from O to A increases the objective.

Moving

maximize $200 - y_1 + 6y_2$

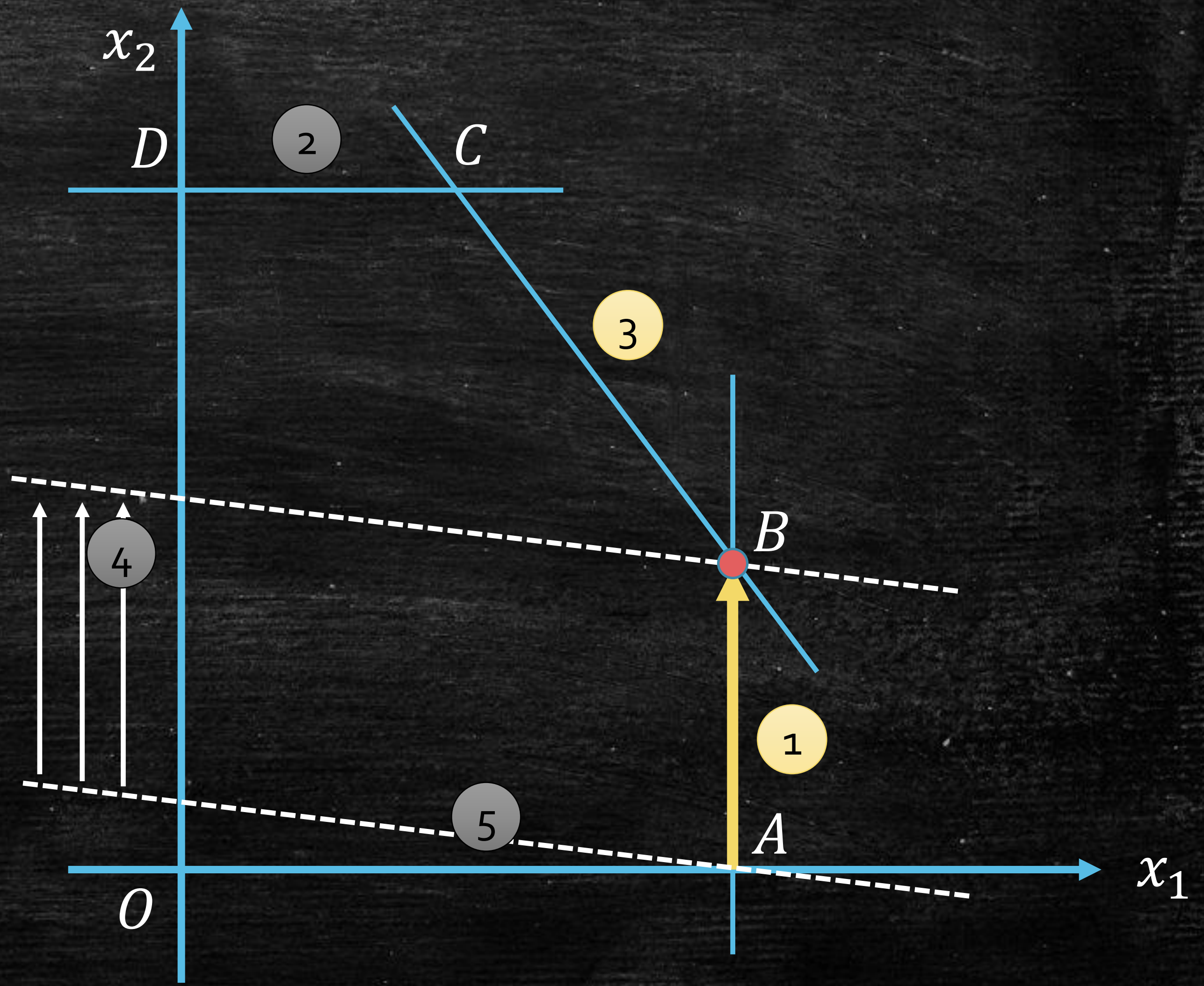
subject to $y_1 \geq 0$ 1

$y_2 \leq 300$ 2

$-y_1 + y_2 \leq 200$ 3

$y_1 \leq 200$ 4

$y_2 \geq 0$ 5



Moving from A to B increases the objective.

Rewrite LPa: View B as Origin

maximize $200 - y_1 + 6y_2$

subject to $y_1 \geq 0$ 1

$\rightarrow z_1 \geq 0$ 1

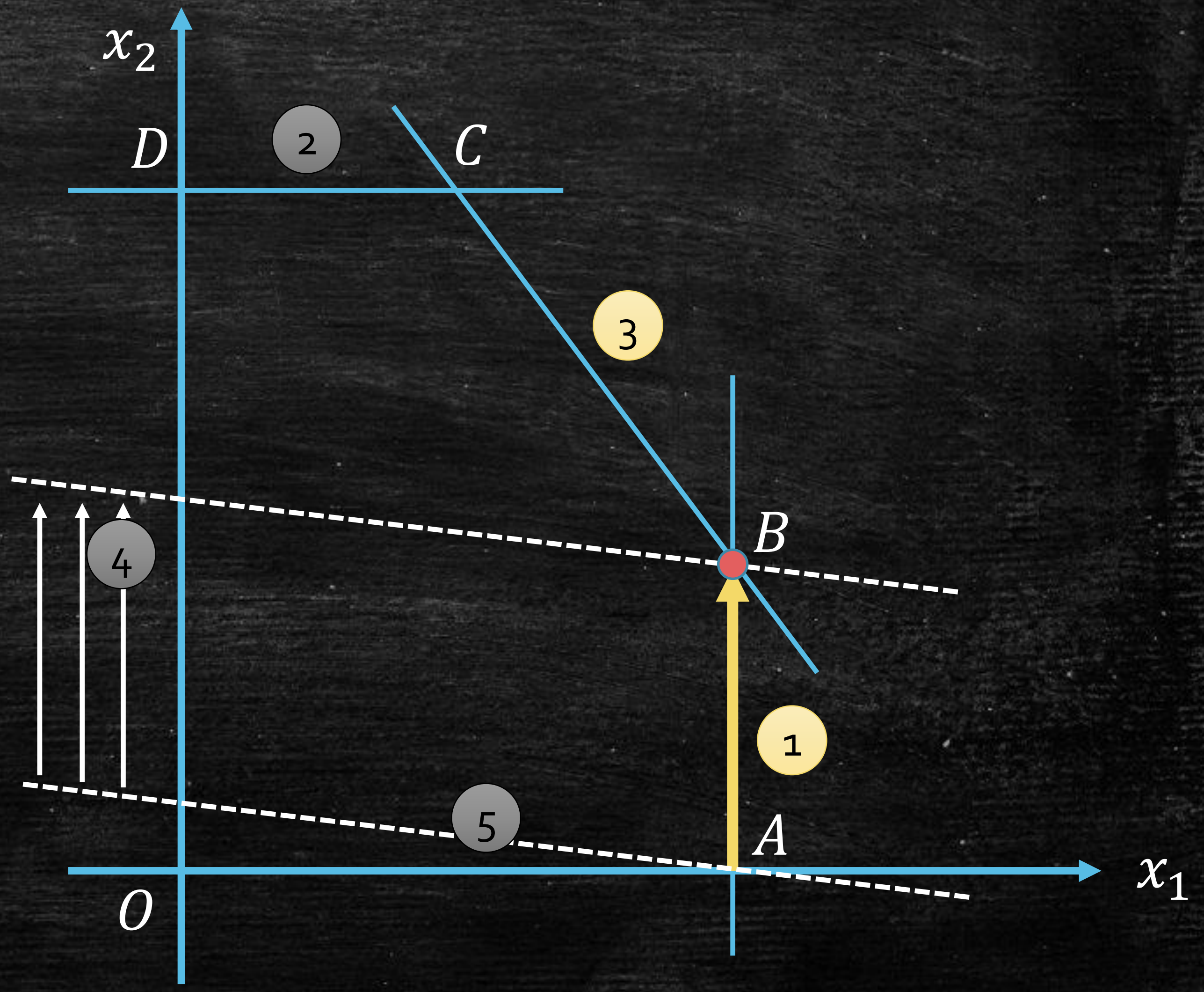
$y_2 \leq 300$ 2

$-y_1 + y_2 \leq 200$ 3

$\rightarrow z_2 \geq 0$ 3

$y_1 \leq 200$ 4

$y_2 \geq 0$ 5



Moving from A to B increases the objective.

Rewrite LP: View B as Origin

maximize $1400 + 5z_1 - 6z_2$

subject to $y_1 \geq 0$ 1

$\rightarrow z_1 \geq 0$ 1

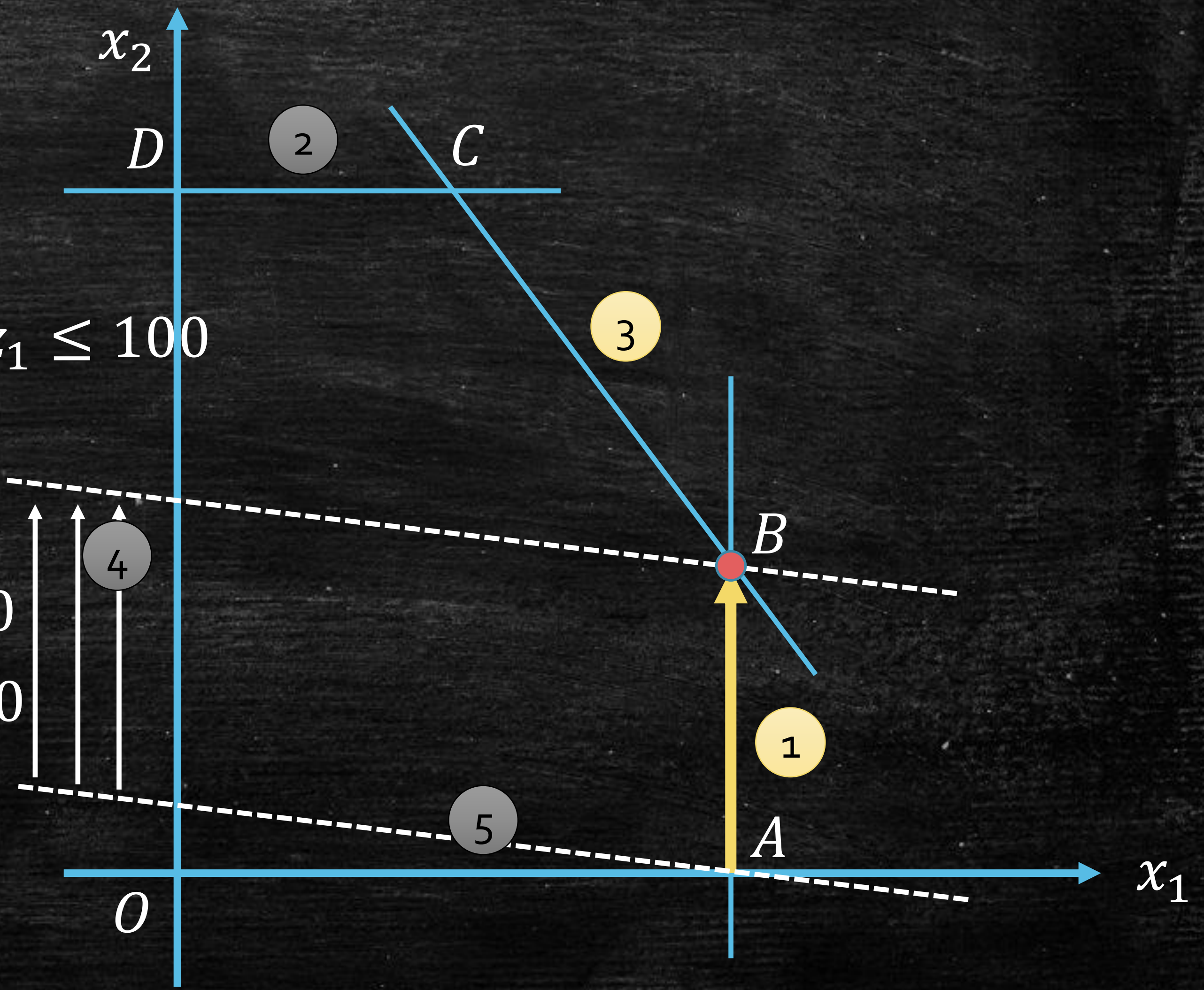
2 $y_2 \leq 300 \rightarrow -z_2 + z_1 \leq 100$

$-y_1 + y_2 \leq 200$ 3

$\rightarrow z_2 \geq 0$ 3

4 $y_1 \leq 200 \rightarrow z_1 \leq 200$

5 $y_2 \geq 0 \rightarrow -z_1 + z_2 \leq 200$



Moving from A to B
increases the objective.

Rewrite LP: View B as Origin

maximize $1400 + 5z_1 - 6z_2$

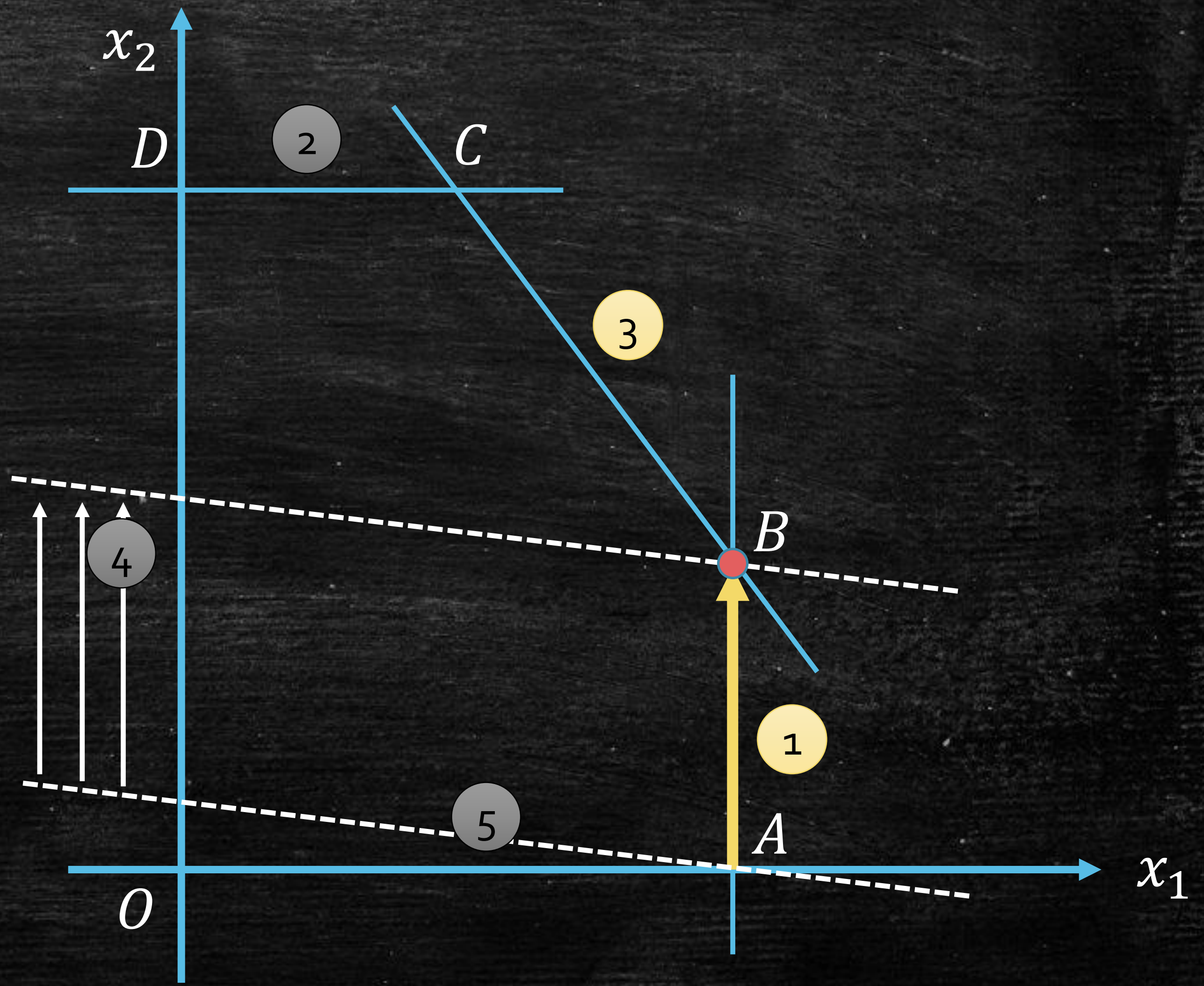
subject to $z_1 \geq 0$ 1

$-z_2 + z_1 \leq 100$ 2

$z_2 \geq 0$ 3

$z_1 \leq 200$ 4

$-z_1 + z_2 \leq 200$ 5



Moving from A to B increases the objective.

Moving

maximize $1400 + 5z_1 - 6z_2$

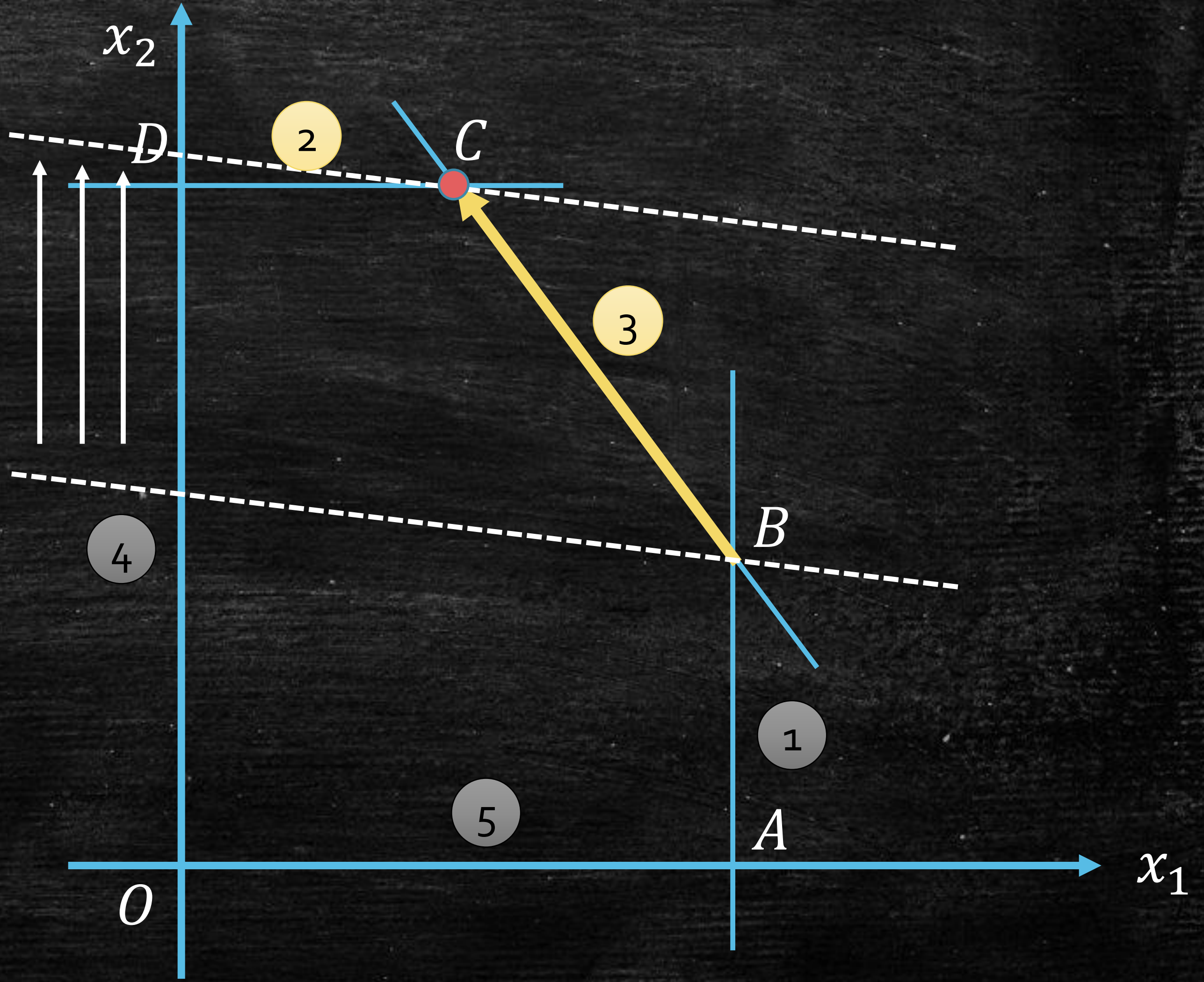
subject to $z_1 \geq 0$ 1

$-z_2 + z_1 \leq 100$ 2

$z_2 \geq 0$ 3

$z_1 \leq 200$ 4

$-z_1 + z_2 \leq 200$ 5



Moving from B to C increases the objective.

Rewrite LP: View C as Origin

maximize $1900 - 5t_1 - t_2$

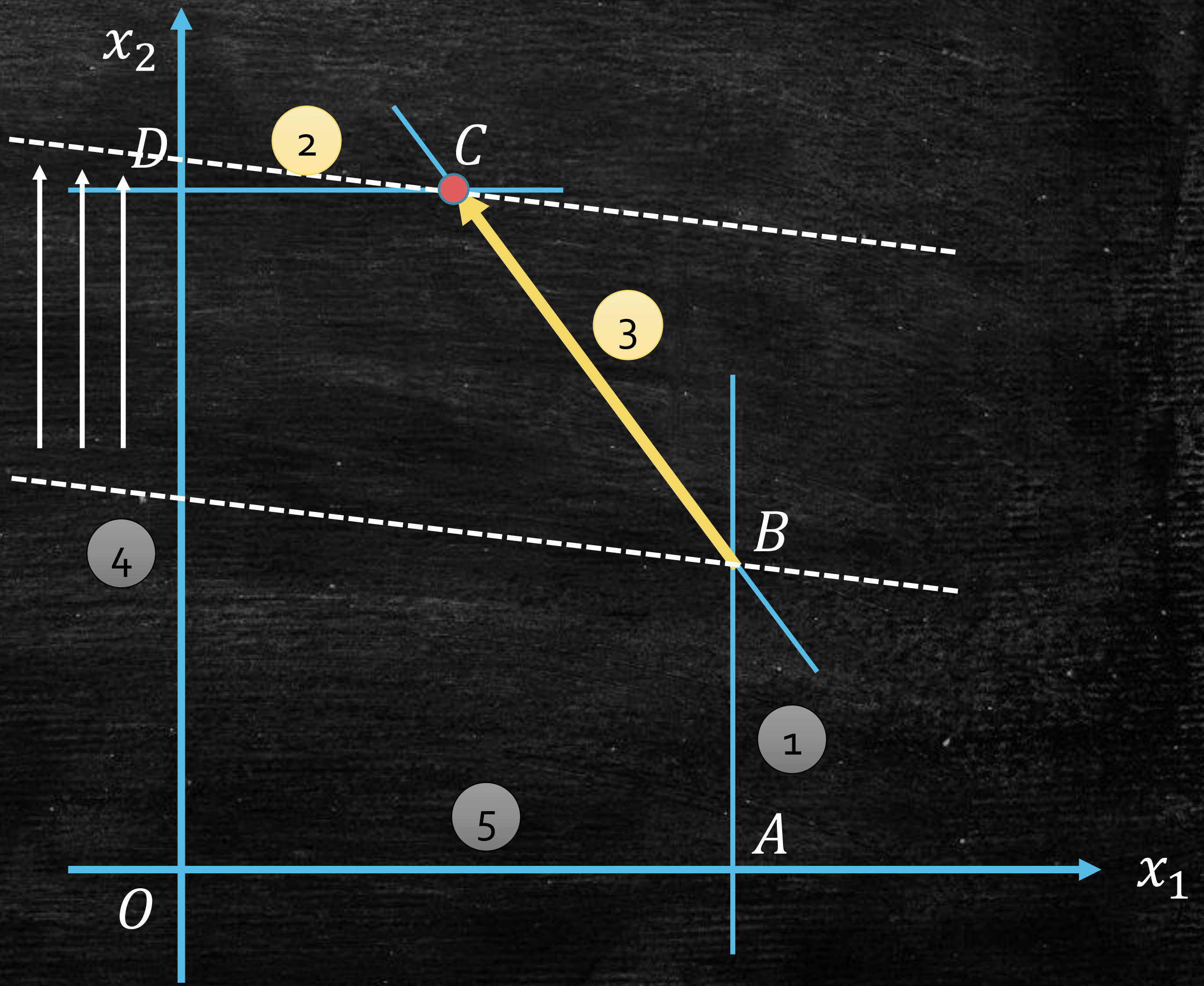
subject to $t_1 - t_2 \le 100$ 1

$t_1 \ge 0$ 2

$t_2 \ge 0$ 3

$-t_1 + t_2 \le 100$ 4

$t_1 \le 300$ 5



Moving from B to C increases the objective.

Output

maximize $1900 - 5t_1 - t_2$

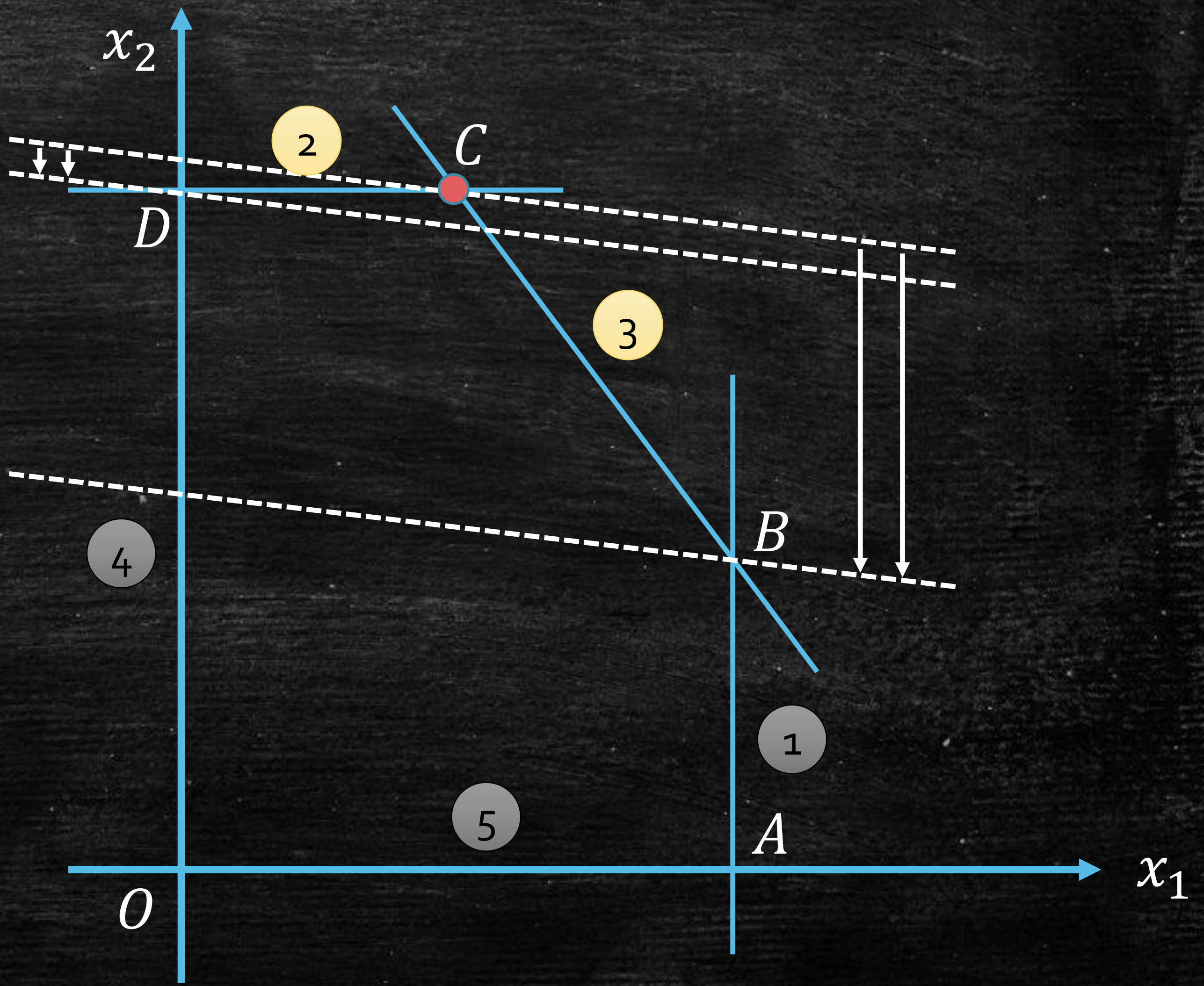
subject to $t_1 - t_2 \leq 100$ (1)

$t_1 \geq 0$ (2)

$t_2 \geq 0$ (3)

$-t_1 + t_2 \leq 100$ (4)

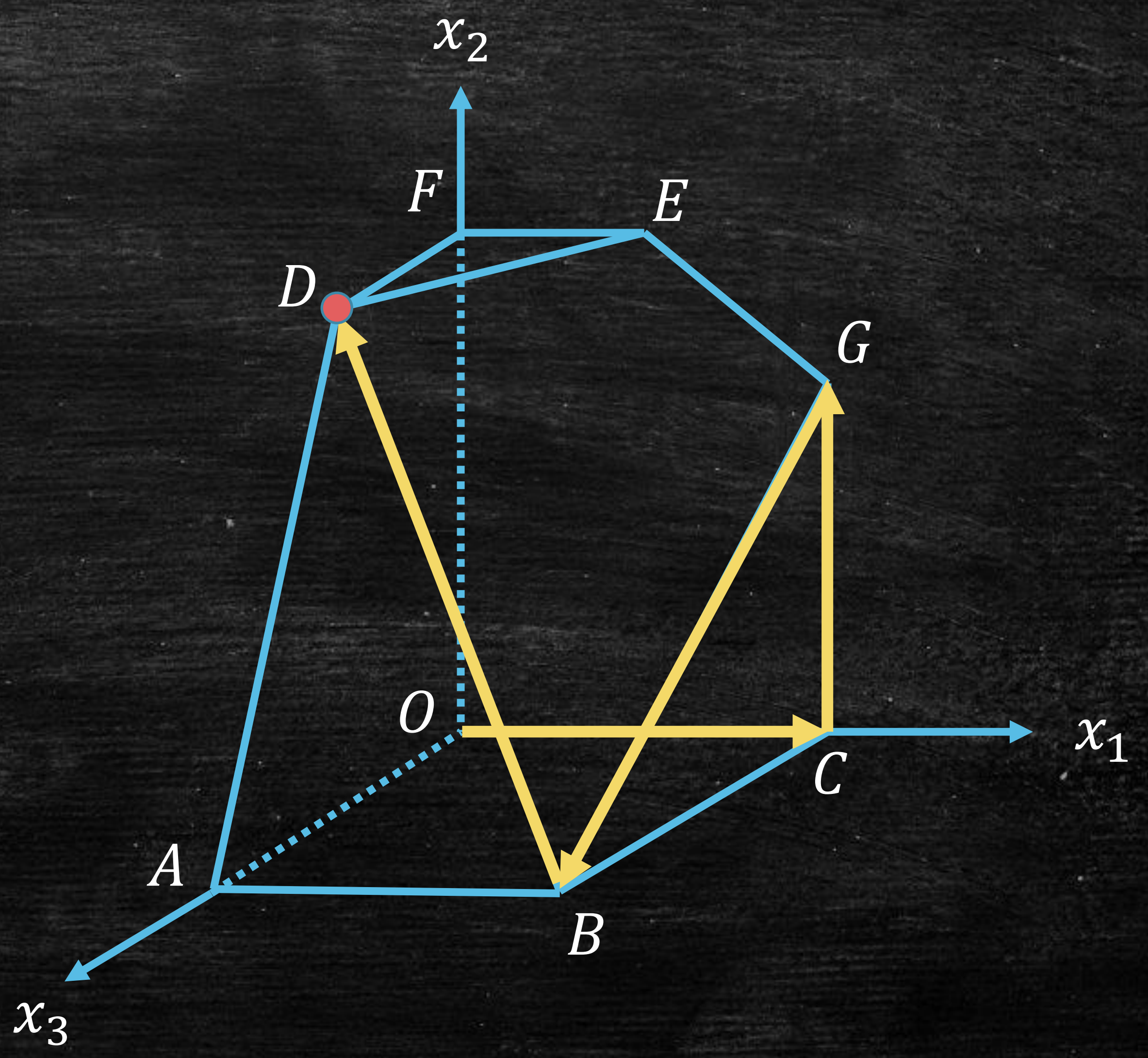
$t_1 \leq 300$ (5)



C is a local maximum: Moving to either D or B decreases the objective.

Simplex Method

$$
\begin{aligned}
\text{maximize} \quad & x_1 + 6x_2 + 13x_3 \\
\text{subject to} \quad & x_1 \leq 200 \\
& x_2 \leq 300 \\
& x_1 + x_2 + x_3 \leq 400 \\
& x_2 + 3x_3 \leq 600 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

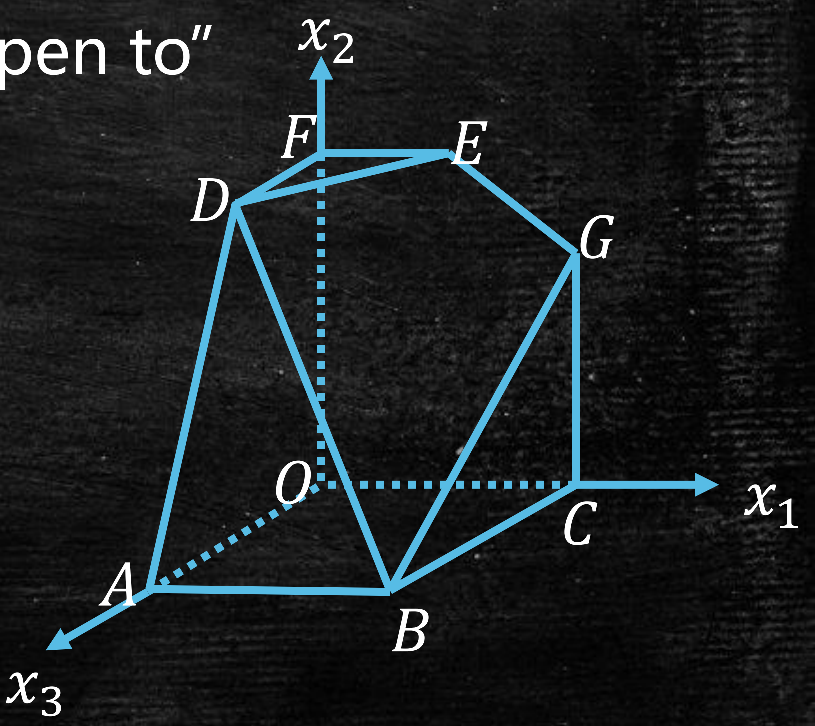


Some Details in Simplex Method

- What exactly is a vertex?
  - A point at the intersection of n linearly independent hyperplanes.
  - n hyperplanes intersect at exactly one point in $\mathbb{R}^n$
- What exactly is an edge?
  - The intersection of $n - 1$ linearly independent hyperplanes.
  - $n - 1$ hyperplanes intersect at a line in $\mathbb{R}^n$
- How do we "move from one vertex to another adjacent vertex along an edge"?
  - Relax one of the n constraint and impose another.
  - The new vertex can be computed by solving a system of n linear equations.

Missing Details not Covered in This Lecture…

- How to find a starting vertex?
- How to find a neighbor that guarantees increment to objective?
- Degenerated vertex: $n + 1$ hyperplanes "happen to" intersect at a single point.
  - E.g., Vertex B and D
- Unbounded feasible region…
- And many more…



How to program?

- Choice 1:
  - Consider all the missing details.
  - Boost it by some heuristic pivoting rule.
- Choice 2:
  - Use open-source LP solver!
  - E.g., GLPK.
- Choice 3:
  - Pay some money to buy faster LP solver.
  - E.g., Gurobi.
  - Actually, you do not need to pay (free for education).

Time Complexity for Simplex Method

- There are exponentially many vertices: $\binom{m}{n}$ for m constraints and n variables.
- Worst-case running time: exponential
  - Many attempts have failed.
  - e.g., choose neighbors with highest objective value, choose neighbors randomly, etc.
- [Teng & Spielman] Smoothed analysis
  - Average case polynomial time if add random Gaussian noise to the constraints.
- Runs fast in practice, and most commonly used.

Small Sot for Dantzig: Creator for Simplex

“During my first year at Berkeley, I arrived late one day to one of Neyman’s classes,” Dantzig recalled years later. “On the blackboard were two problems, which I assumed had been assigned for homework. I copied them down.

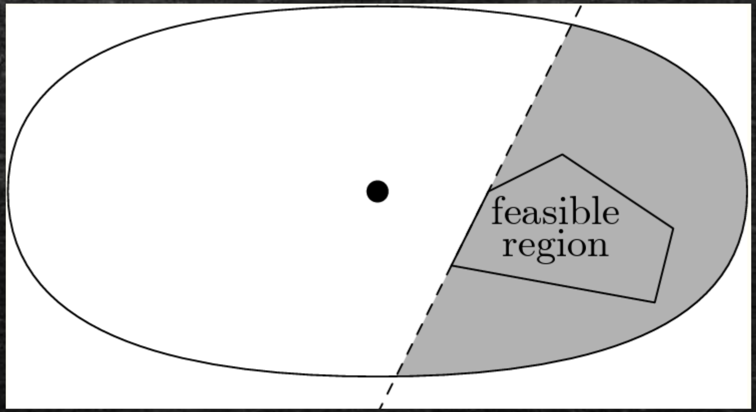
“A few days later,” he said, “I apologized to Neyman for taking so long to do the homework -- the problems seemed to be a little harder to do than the usual. He told me to throw [the homework] on his desk.”

Early one morning about six weeks later, Dantzig found Neyman banging excitedly on the front door of his apartment. What Dantzig had copied off the blackboard was not homework but examples of two famous unsolved problems in statistics.

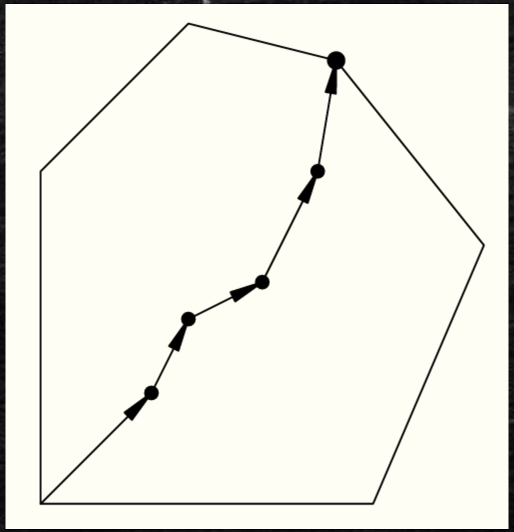
Dantzig had solved one, and Neyman wanted to send out one of his papers for immediate publication.

Polynomial Time Algorithms for LP

- Ellipsoid Method



- Interior Point Method



Standard Form LP

- Maximization as objective with "≤" constraints and non-negative variables.

$$
\begin{array}{ll}
\text{maximize} & c_1x_1 + c_2x_2 + \cdots c_nx_n \\
\text{subject to} & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leq b_1 \\
& a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leq b_2 \\
& \vdots \\
& a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \leq b_m \\
& x_1, x_2, \dots, x_n \geq 0
\end{array}
$$

$$
\begin{array}{ll}
\text{maximize} & \mathbf{c}^\top \mathbf{x} \\
\text{subject to} & A\mathbf{x} \leq \mathbf{b} \\
& \mathbf{x} \geq \mathbf{0}
\end{array}
$$

Other Forms Reduce to Standard Form

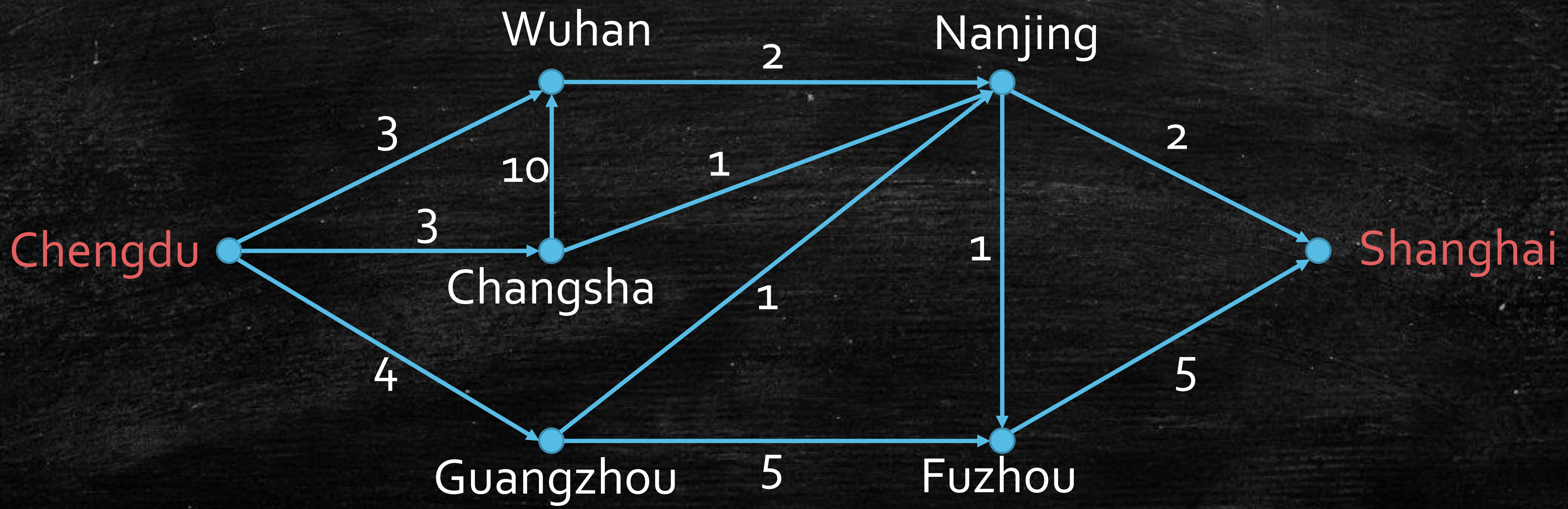
- Minimization to Maximization
  - $\min c_1x_1 + \cdots + c_nx_n \quad \Leftrightarrow \quad \max\, - c_1x_1 - \cdots - c_nx_n$
- $\geq$-inequalities
  - $a_1x_1 + \cdots + a_nx_n \geq b \quad \Leftrightarrow \quad -a_1x_1 - \cdots - a_nx_n \leq -b$
- Inequality $\Leftrightarrow$ Equality
  - $a_1x_1 + \cdots + a_nx_n = b \quad \Leftrightarrow \quad \begin{cases} a_1x_1 + \cdots + a_nx_n \leq b \\ a_1x_1 + \cdots + a_nx_n \geq b \end{cases}$
  - $a_1x_1 + \cdots + a_nx_n \leq b \quad \Leftrightarrow \quad a_1x_1 + \cdots + a_nx_n + s = b$
- Variable with unrestricted signs
  - Introduce two variables x^+ and x^- with standard constraints $x^+, x^- \geq 0$
  - Replace x with $x^+ - x^-$

Take-Home Message

- A linear program can be solved in a polynomial time.
- Whenever a problem can be formulated by a linear program, it is polynomial-time solvable.

Maximum Flow Problem

- **Input:**
  - Railway system: a directed graph $G(V, E)$, s and t.
  - Edges Capacity: $w(e)$ for each $e \in E$. (Maximum number of passengers a day.)
- **Output:**
  - The maximum number of passengers we can send from s to t a day.



Formulation as Linear Program

- The maximum flow problem can be formulated by a linear program.

$$
\begin{aligned}
&\text{maximize} && \sum_{u:(s,u)\in E} f_{su} && \\
&\text{subject to} && 0 \le f_{uv} \le c_{uv} && \forall (u,v) \in E \\
& && \sum_{v:(v,u)\in E} f_{vu} = \sum_{w:(v,w)\in E} f_{uw} && \forall u \in V \setminus \{s,t\}
\end{aligned}
$$

- Ford-Fulkerson Method implements the simplex method.

Part II:
LP Duality

Motivation

- We have seen that the optimal solution for the LP below is $(x_1,\ x_2) = (100, 300)$, with value 1900.
  - Geometric argument, argument based on simplex method
- Let's try to prove it by some simple observations from the LP itself!

$$
\begin{aligned}
\text{maximize} \quad & x_1 + 6x_2 \\
\text{subject to} \quad & x_1 \leq 200 \\
& x_2 \leq 300 \\
& x_1 + x_2 \leq 400 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Motivation

$$
\begin{aligned}
\text{maximize} \quad & x_1 + 6x_2 & \\
\text{subject to} \quad & x_1 \le 200 & \text{(i)} \\
& x_2 \le 300 & \text{(ii)} \\
& x_1 + x_2 \le 400 & \text{(iii)} \\
& x_1, x_2 \ge 0 &
\end{aligned}
$$

- Let's try adding (i) to 6 times (ii): $x_1 + 6x_2 \le 200 + 6 \times 300 = 2000$
- We know that any solution (x_1, x_2) cannot yield objective value greater than 2000.
- Can we combine the inequality in a better way to show that the objective value is at most 1900?

Motivation

$$\begin{aligned} \text{maximize} \quad & x_1 + 6x_2 & \\ \text{subject to} \quad & x_1 \le 200 & \text{(i)} \\ & x_2 \le 300 & \text{(ii)} \\ & x_1 + x_2 \le 400 & \text{(iii)} \\ & x_1, x_2 \ge 0 & \end{aligned}$$

- Can we combine the inequality in a better way to show that the objective value is at most 1900?
- Yes, we can:
  - Multiple (ii) by 5 and add to (iii): $x_1 + 6x_2 \le 300 \times 5 + 400 = 1900.$
- This proves that $(x_1, x_2) = (100, 300)$ with objective value 1900 is optimal!

Let's try this one...

- Suppose we multiple (i) by y_1, (ii) by y_2, (iii) by y_3, and (iv) by y_4.
- We have $(y_1 + y_3)x_1 + (y_2 + y_3 + y_4)x_2 + (y_3 + 3y_4)x_3 \leq 200y_1 + 300y_2 + 400y_3 + 600y_4$.
- We need $y_1, y_2, y_3, y_4 \geq 0$ to keep the inequality.
- To find an upper bound to the objective $x_1 + 6x_2 + 13x_3$, we need to make sure $x_1 + 6x_2 + 13x_3 \leq (y_1 + y_3)x_1 + (y_2 + y_3 + y_4)x_2 + (y_3 + 3y_4)x_3$ holds for every (x_1, x_2, x_3).
- Since $x_1, x_2, x_3 \geq 0$, we must have:
  - $y_1 + y_3 \geq 1$
  - $y_2 + y_3 + y_4 \geq 6$
  - $y_3 + 3y_4 \geq 13$

$$
\begin{aligned}
\text{maximize}\quad & x_1 + 6x_2 + 13x_3 \\
\text{subject to}\quad & x_1 \leq 200 && \text{(i)} \\
& x_2 \leq 300 && \text{(ii)} \\
& x_1 + x_2 + x_3 \leq 400 && \text{(iii)} \\
& x_2 + 3x_3 \leq 600 && \text{(iv)} \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

Let's try this one…

- $(y_1 + y_3)x_1 + (y_2 + y_3 + y_4)x_2 + (y_3 + 3y_4)x_3 \leq 200y_1 + 300y_2 + 400y_3 + 600y_4$.
- Since $x_1, x_2, x_3 \geq 0$, we must have:
  - $y_1 + y_3 \geq 1$
  - $y_2 + y_3 + y_4 \geq 6$
  - $y_3 + 3y_4 \geq 13$
- Now, we want to find the tightest possible upper-bound to $x_1 + 6x_2 + 13x_3$.
- This means we want to minimize $200y_1 + 300y_2 + 400y_3 + 600y_4$.

$$
\begin{aligned}
\text{maximize} \quad & x_1 + 6x_2 + 13x_3 \\
\text{subject to} \quad & x_1 \leq 200 && \text{(i)} \\
& x_2 \leq 300 && \text{(ii)} \\
& x_1 + x_2 + x_3 \leq 400 && \text{(iii)} \\
& x_2 + 3x_3 \leq 600 && \text{(iv)} \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

Dual Program

- The problem of finding the tightest upper-bound can be formulated by another linear program!
- This linear program is called the dual program, and the original one is called the primal program.

$$
\begin{aligned}
\text{maximize}\quad & x_1 + 6x_2 + 13x_3 \\
\text{subject to}\quad & x_1 \le 200 \\
& x_2 \le 300 \\
& x_1 + x_2 + x_3 \le 400 \\
& x_2 + 3x_3 \le 600 \\
& x_1, x_2, x_3 \ge 0
\end{aligned}
$$

$$
\begin{aligned}
\text{minimize}\quad & 200y_1 + 300y_2 + 400y_3 + 600y_4 \\
\text{subject to}\quad & y_1 + y_3 \ge 1 \\
& y_2 + y_3 + y_4 \ge 6 \\
& y_3 + 3y_4 \ge 13 \\
& y_1, y_2, y_3, y_4 \ge 0
\end{aligned}
$$

Dual Program

- Factory Example:

$$\begin{aligned} \text{maximize} \quad & x_1 + 6x_2 \\ \text{subject to} \quad & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$

$$\begin{aligned} \text{minimize} \quad & 200y_1 + 300y_2 + 400y_3 \\ \text{subject to} \quad & y_1 + y_3 \geq 1 \\ & y_2 + y_3 \geq 6 \\ & y_1, y_2, y_3 \geq 0 \end{aligned}$$

- Dual program for standard form:

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^\top \mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

$$\begin{aligned} \text{minimize} \quad & \mathbf{b}^\top \mathbf{y} \\ \text{subject to} \quad & \mathbf{y}^\top A \geq \mathbf{c}^\top \\ & \mathbf{y} \geq \mathbf{0} \end{aligned}$$

Weak Duality Theorem

- By our motivation of dual program, we obtain the following theorem.
- Theorem [Weak Duality Theorem]. If $\hat{\mathbf{x}}$ is a feasible solution to (a) and $\hat{\mathbf{y}}$ is a feasible solution to (b), then $\mathbf{c}^\top \hat{\mathbf{x}} \le \mathbf{b}^\top \hat{\mathbf{y}}$.

$$\begin{array}{ll} \text{maximize} & \mathbf{c}^\top \mathbf{x} \\ \text{subject to} & A\mathbf{x} \le \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0} \end{array} \qquad \text{(a)}$$

$$\begin{array}{ll} \text{minimize} & \mathbf{b}^\top \mathbf{y} \\ \text{subject to} & \mathbf{y}^\top A \ge \mathbf{c}^\top \\ & \mathbf{y} \ge \mathbf{0} \end{array} \qquad \text{(b)}$$

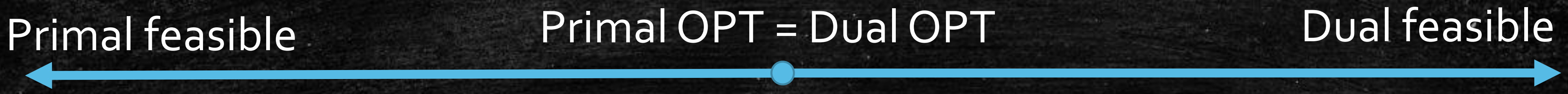
Strong Duality Theorem: This gap is always closed!

Strong Duality Theorem

- Theorem [Strong Duality Theorem]. Let $\mathbf{x}^*$ be the optimal solution to (a) and $\mathbf{y}^*$ be the optimal solution to (b), then $\mathbf{c}^\top\mathbf{x}^* = \mathbf{b}^\top\mathbf{y}^*$.

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^\top\mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} \le \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0} \end{aligned} \qquad \text{(a)}$$

$$\begin{aligned} \text{minimize} \quad & \mathbf{b}^\top\mathbf{y} \\ \text{subject to} \quad & \mathbf{y}^\top A \ge \mathbf{c}^\top \\ & \mathbf{y} \ge \mathbf{0} \end{aligned} \qquad \text{(b)}$$



Application of Strong Duality Theorem

- Max-Flow-Min-Cut Theorem
- Minimax Theorem
- Kőnig-Egerváry Theorem
- Design approximation algorithms:
  - Dual fitting
  - Primal-Dual Schema
- Economic interpretation: “resource allocation”-”resource valuation”

Understanding of Duality

- You are selling 3 resources. You need to make price (r_j).
- You know how many resources people needs
  - R_1: 5
  - R_2: 5
  - R_3: 3
- Revenue: $\max 5p_1 + 5p_2 + 3p_3$.
- Recourses are not be sold to people directly, but by some kinds of products, each product has a price.
  - Product 1: use 2 R_1 and 3 R_2 with price 10.
  - Product 2: use 1 R_1, 1 R_2, and 2 R_3 with price 5.

Formalize The Linear Program

- Objective: $\max 5p_1 + 5p_2 + 3p_3$.
- Product 1: use 2 R_1 and 3 R_2 with price 10.
- Product 2: use 1 R_1, 1 R_2, and 2 R_3 with price 5.
- We do not want the cost of a product is larger than its price.
- Constraints:
  - $2p_1 + 3p_2 \leq 10$
  - $p_1 + p_2 + p_3 \leq 5$
  - $p_1, p_2, p_3 \geq 0$

The Dual Problem

Primal

$$\begin{aligned} \text{maximize} \quad & 5p_1 + 5p_2 + 3p_3 \\ \text{subject to} \quad & 2p_1 + 3p_2 \leq 10 \\ & p_1 + p_2 + p_3 \leq 5 \\ & p_1, p_2, p_3 \geq 0 \end{aligned}$$

Dual

$$\begin{aligned} \text{minimize} \quad & 10y_1 + 5y_2 \\ \text{subject to} \quad & 2y_1 + y_2 \geq 5 \\ & 3y_1 + y_2 \geq 5 \\ & y_2 \geq 3 \\ & y_1, y_2, y_3 \geq 0 \end{aligned}$$

- Dual means the best way for people to buy products to get the required resources.

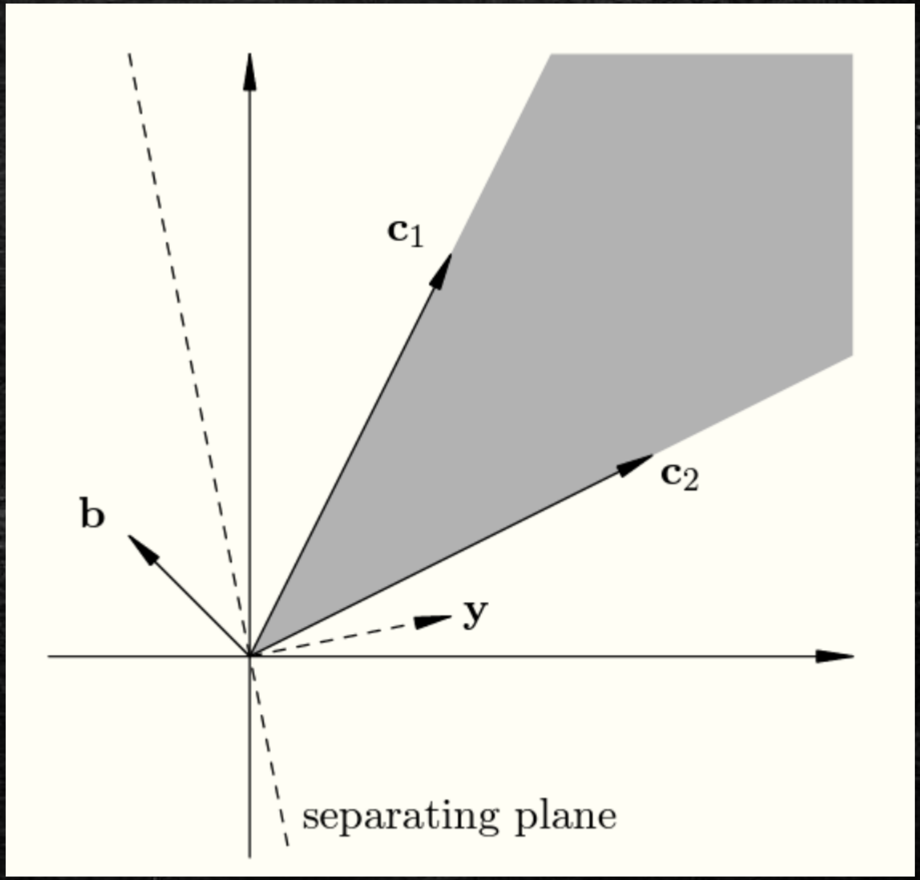
Understanding of The Dual Theorem

- Weak Duality Theorem
  - Because each product's price is larger than the cost.
  - People must pay more to get the required resources.
- Strong Duality Theorem
  - Assume there is a Factory for manufacture these products.
  - People's best cost is D (optimal dual objective)
  - The factory can afford at most D as cost.
  - We sell resources to the factory and get P (the primal solution).
  - The best we can get should be exactly D!

Proof of Strong Duality Theorem

- **Theorem [Farkas Lemma]**. Exactly one of the followings holds for matrix $A \in \mathbb{R}^{m\times n}$ and vector $\mathbf{b} \in \mathbb{R}^m$:
  1. There exists $\mathbf{x} \in \mathbb{R}^n$ with $\mathbf{x} \geq \mathbf{0}$ such that $A\mathbf{x} = \mathbf{b}$.
  2. There exists $\mathbf{y} \in \mathbb{R}^m$ such that $A^\top \mathbf{y} \geq \mathbf{0}$ and $\mathbf{b}^\top \mathbf{y} < 0$.
- $\{A\mathbf{x} \mid \mathbf{x} \geq \mathbf{0}\}$ is the grey area.
- 1 says that $\mathbf{b}$ is inside the grey area.
- 2 says that we can separate the grey area and $\mathbf{b}$ by a hyperplane (defined by the normal vector $\mathbf{y}$).
  - In this case $\mathbf{b}$ must be outside the grey area.

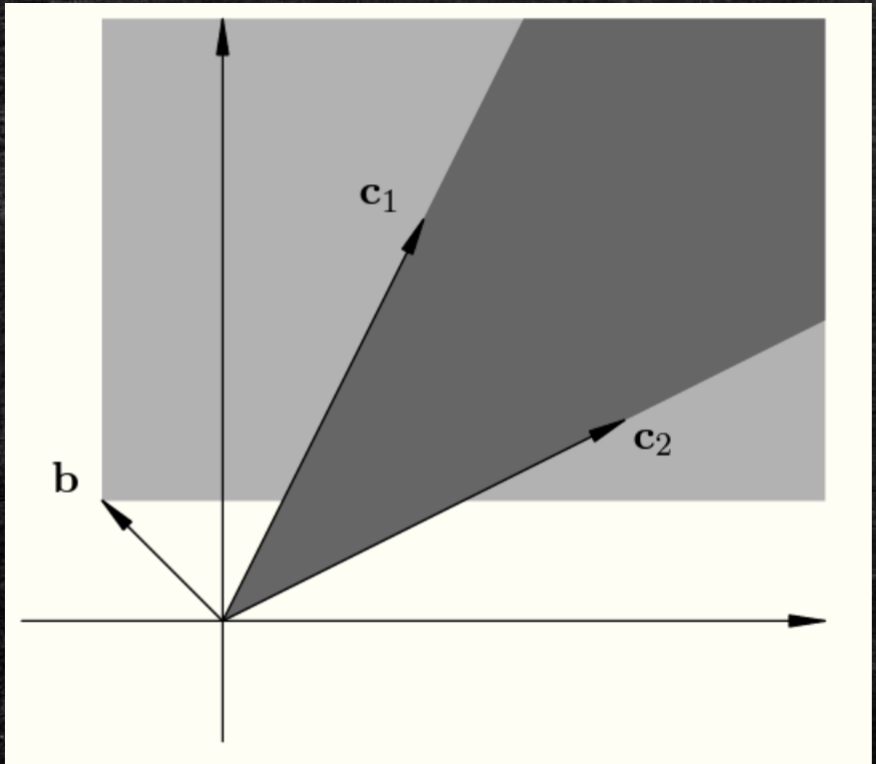
Illustration for $A = [\boldsymbol{c}_1 \quad \boldsymbol{c}_2]$



A Corollary to Farkas Lemma

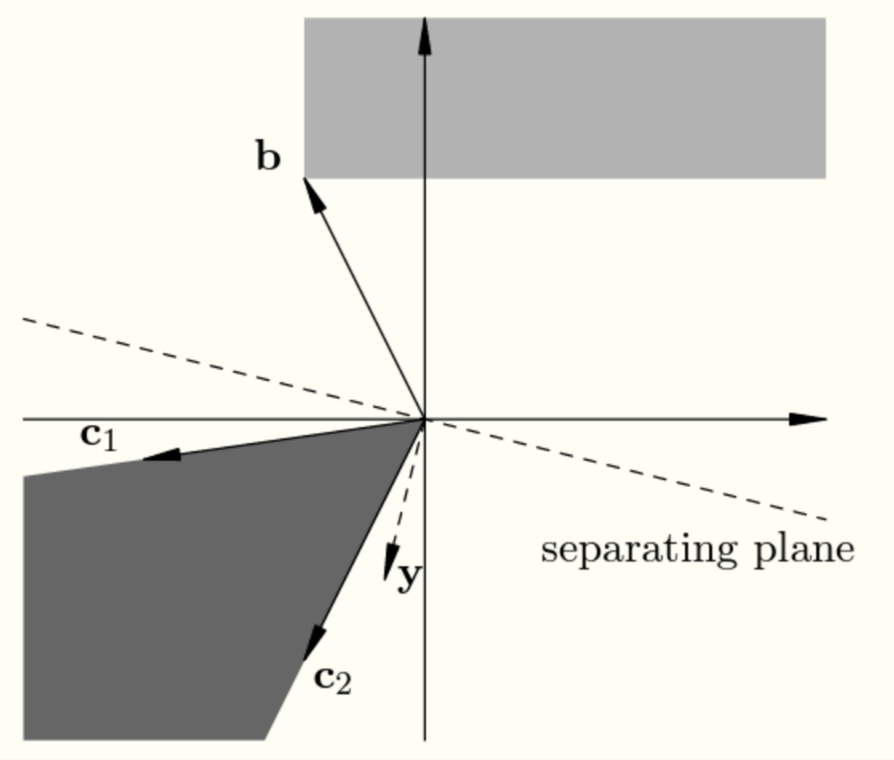
- **Corollary**. Exactly one of the followings holds for matrix $A \in \mathbb{R}^{m \times n}$ and vector $\mathbf{b} \in \mathbb{R}^m$:
  1. There exists $\mathbf{x} \in \mathbb{R}^n$ with $\mathbf{x} \geq \mathbf{0}$ such that $A\mathbf{x} \geq \mathbf{b}$.
  2. There exists $\mathbf{y} \in \mathbb{R}^m$ with $\mathbf{y} \leq \mathbf{0}$ such that $A^\top \mathbf{y} \geq \mathbf{0}$ and $\boldsymbol{b}^\top \mathbf{y} < 0$.

Case 1 of the Corollary



- $\{A\mathbf{x} \mid \mathbf{x} \geq \mathbf{0}\}$ is the dark grey area.
- $\{\mathbf{x} \mid \mathbf{x} \geq \mathbf{b}\}$ is the light grey area.
- 1 says that the two areas intersect.

Case 2 of the Corollary



- $\{A\mathbf{x} \mid \mathbf{x} \geq \mathbf{0}\}$ is the dark grey area.
- $\{\mathbf{x} \mid \mathbf{x} \geq \mathbf{b}\}$ is the light grey area.
- 2 describes that the two areas do not intersect.
- We can find a separating plane with normal vector $\mathbf{y}$.
  - Thus, $A^\top \mathbf{y} \geq 0$ and $\mathbf{b}^\top \mathbf{y} < 0$
- We must have $\mathbf{y} \leq \mathbf{0}$:
  - If this fails for one entry: $y_i > 0$
  - $\mathbf{z} = (\varepsilon, \dots, \varepsilon, z_i = 1, \varepsilon, \dots, \varepsilon)$ and $\boldsymbol{y}$ on same side
  - $\mathbf{z}$ is in the first quadrant, and it will eventually intersect the light grey area after extension.
  - The two areas are on the same side with $\mathbf{y}$.

Proof of the Corollary

- Define $A' \in \mathbb{R}^{m\times(n+m)}$ by $A' = [A \quad -I]$.
- Apply Farkas Lemma on A' and $\mathbf{b}$.
- Let P1 and P2 be 1 and 2 in Farkas Lemma; Q1 and Q2 be 1 and 2 in the corollary.
- We aim to show P1 $\Leftrightarrow$ P2 and Q1 $\Leftrightarrow$ Q2.

Proof of the Corollary

- Define $A' \in \mathbb{R}^{m\times(n+m)}$ by $A' = [A \quad -I]$.
- P1 $\Leftrightarrow \exists \mathbf{x}' \in \mathbb{R}^{n+m}$ s.t. $\mathbf{x}' \geq \mathbf{0}$ and $A'\mathbf{x}' = \mathbf{b}$.
- (by writing $\mathbf{x}' = \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix}$) $\Leftrightarrow$ $[A \quad -I] \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix} = \mathbf{b}$ (where $\mathbf{x} \geq \mathbf{0}$, $\bar{\mathbf{x}} \geq \mathbf{0}$)
- $\Leftrightarrow$ $A\mathbf{x} - \bar{\mathbf{x}} = \mathbf{b}$ $\Leftrightarrow A\mathbf{x} \geq \mathbf{b}$ (since $\bar{\mathbf{x}} \geq \mathbf{0}$)
- $\Leftrightarrow$ Q1

Proof of the Corollary

- Define $A' \in \mathbb{R}^{m\times(n+m)}$ by $A' = [A \quad -I]$.
- P2 $\Leftrightarrow \exists \mathbf{y} \in \mathbb{R}^m$ s.t. ${A'}^\top \mathbf{y} \geq \mathbf{0}$ and $\mathbf{b}^\top \mathbf{y} < 0$.
- $\Leftrightarrow \begin{bmatrix} A^\top \\ -I \end{bmatrix} \mathbf{y} \geq \mathbf{0}$ and $\mathbf{b}^\top \mathbf{y} < 0$
- $\Leftrightarrow A^\top \mathbf{y} \geq \mathbf{0}, \quad -\mathbf{y} \geq 0,$ and $\mathbf{b}^\top \mathbf{y} < 0$
- $\Leftrightarrow$ Q2

Now we are ready to prove strong duality theorem…

- Weak duality: $\mathbf{c}^\top \mathbf{x} \le \mathbf{b}^\top \mathbf{y}^*$ holds for any $\mathbf{x} \ge \mathbf{0}$.
- Suppose strong duality fails: $\mathbf{c}^\top \mathbf{x} < \mathbf{b}^\top \mathbf{y}^*$.
- There does not exist $\mathbf{x} \ge \mathbf{0}$ satisfying $A\mathbf{x} \le \mathbf{b}$ and $\mathbf{c}^\top \mathbf{x} \ge \mathbf{b}^\top \mathbf{y}^*$.
- We cannot have $\begin{bmatrix} -A \\ \mathbf{c}^\top \end{bmatrix} \mathbf{x} \ge \begin{bmatrix} -\mathbf{b} \\ \mathbf{b}^\top \mathbf{y}^* \end{bmatrix}$ and $\mathbf{x} \ge \mathbf{0}$.
- Q1 in corollary fails for matrix $\begin{bmatrix} -A \\ \mathbf{c}^\top \end{bmatrix}$ and vector $\begin{bmatrix} -\mathbf{b} \\ \mathbf{b}^\top \mathbf{y}^* \end{bmatrix}$.
- Thus, Q2 must be true.

Now we are ready to prove strong duality theorem…

- Q2 is true for matrix $\begin{bmatrix} -A \\ \mathbf{c}^\top \end{bmatrix}$ and vector $\begin{bmatrix} -\mathbf{b} \\ \mathbf{b}^\top \mathbf{y}^* \end{bmatrix}$.
- There exist $\mathbf{y} \in \mathbb{R}^m$ and $w \in \mathbb{R}$ such that

$$[-A^\top \quad \mathbf{c}] \begin{bmatrix} \mathbf{y} \\ w \end{bmatrix} \geq \mathbf{0}, \quad [-\mathbf{b}^\top \quad \mathbf{b}^\top \mathbf{y}^*] \begin{bmatrix} \mathbf{y} \\ w \end{bmatrix} < 0, \quad \text{and} \quad \begin{bmatrix} \mathbf{y} \\ w \end{bmatrix} \leq \mathbf{0}.$$

- After matrix multiplications,

$$\begin{cases} -A^\top \mathbf{y} + w\mathbf{c} \geq \mathbf{0} \\ -\mathbf{b}^\top \mathbf{y} + w\mathbf{b}^\top \mathbf{y}^* < 0 \\ \mathbf{y} \leq \mathbf{0} \\ w \leq 0 \end{cases}$$

Proof of Strong Duality Theorem

$$\begin{cases} -A^\top \mathbf{y} + w\boldsymbol{c} \geq \mathbf{0} \\ -\mathbf{b}^\top \mathbf{y} + w\mathbf{b}^\top \mathbf{y}^* < 0 \\ \mathbf{y} \leq \mathbf{0} \\ w \leq 0 \end{cases}$$

- Suppose $w < 0$. We divide w on both sides:

$$\begin{cases} -A^\top \left(\dfrac{\mathbf{y}}{w}\right) + \mathbf{c} \leq \mathbf{0} \\ -\mathbf{b}^\top \left(\dfrac{\mathbf{y}}{w}\right) + \mathbf{b}^\top \mathbf{y}^* > 0 \\ \left(\dfrac{\mathbf{y}}{w}\right) \geq \mathbf{0} \end{cases}$$

- $\left(\dfrac{\mathbf{y}}{w}\right)$ is a better solution than $\mathbf{y}^*$ in the dual LP, contradiction!

Proof of Strong Duality Theorem

$$\begin{cases} -A^\top \mathbf{y} + w\mathbf{c} \geq \mathbf{0} \\ -\boldsymbol{b}^\top \mathbf{y} + w\mathbf{b}^\top \mathbf{y}^* < 0 \\ \mathbf{y} \leq \mathbf{0} \\ w \leq 0 \end{cases}$$

- Let's then do the case $w = 0$.
- We have $-A^\top \mathbf{y} \geq \mathbf{0}$, $-\mathbf{b}^\top \mathbf{y} < 0$, and $\mathbf{y} \leq \mathbf{0}$.
- Q2 in Corollary holds for $-A$ and $-\mathbf{b}$.
- So Q1 must be false: $\nexists \mathbf{x} \geq 0 \colon (-A)\mathbf{x} \geq -\mathbf{b}$.
- The feasible region for the primal LP is empty!

Part III:
LP-Relaxation

Integer Program

- If we require each variable in a linear program is an integer, we obtain an integer program (IP), or integer linear program (ILP).
- Many problem can be formulated as IP.
- Standard form:

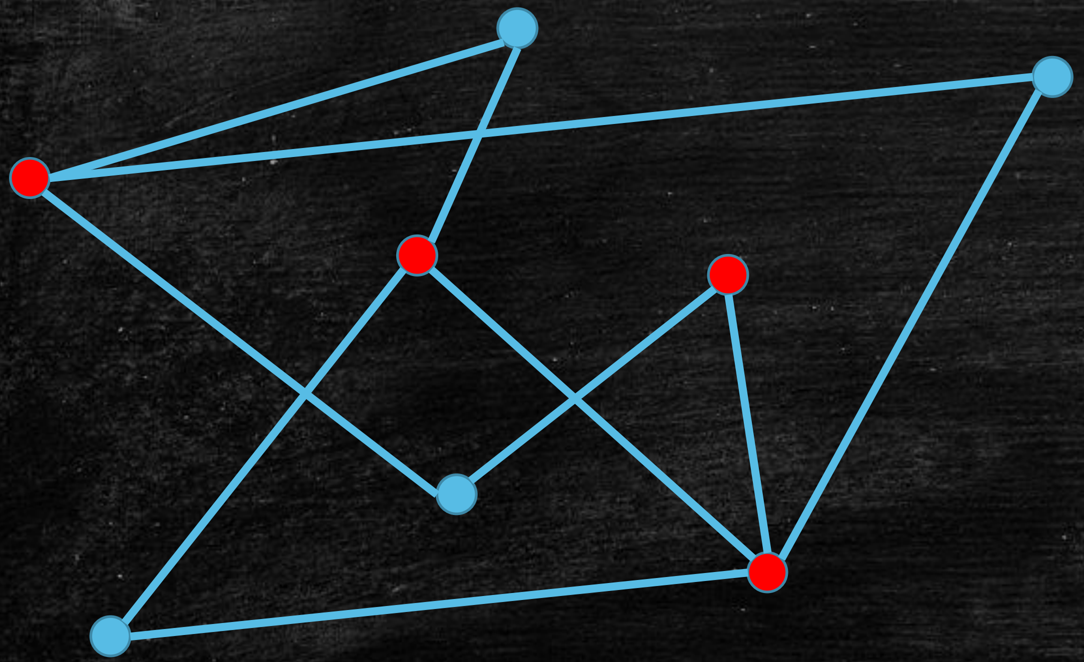
$$
\begin{aligned}
\text{maximize} \quad & \mathbf{c}^\top \mathbf{x} \\
\text{subject to} \quad & A\mathbf{x} \leq \mathbf{b} \\
& \mathbf{x} \geq \mathbf{0} \\
& \mathbf{x} \in \mathbb{Z}^n
\end{aligned}
$$

LP-Relaxation

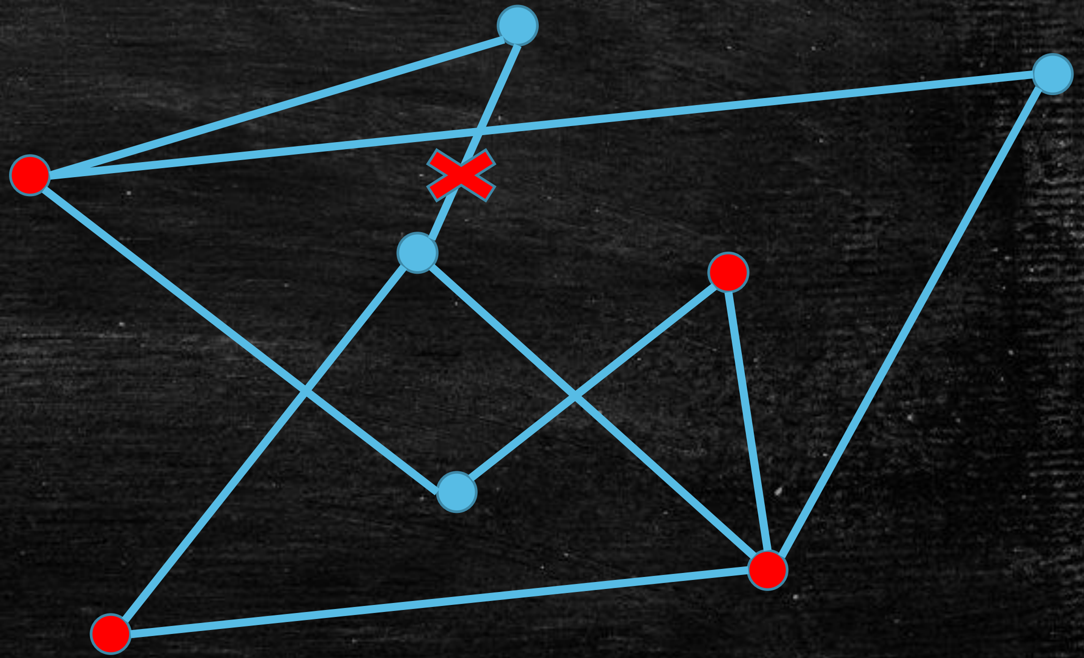
- Integer Programming is NP-complete, even for the zero-one special case $\forall i$: $x_i \in \{0, 1\}$.
- We can use the fact that LP is polynomial-time solvable to design approximation algorithm.
- Relax $x_i \in \{0,1\}$ to $0 \leq x_i \leq 1$.
- Then "round" the fractional solution to integral one:
  - E.g., $x_i = 0.7$ is rounded to $x_i = 1$, $x_i = 0.2$ is rounded to $x_i = 0$.
- and show that the rounded solution is feasible and achieves good approximation guarantee.

LP-Relaxation Example: Vertex Cover

- Given an undirected graph $G = (V, E)$, a subset of vertices $S \subseteq V$ is a vertex cover if S contains at least one endpoint of every vertex.



a vertex cover



not a vertex cover

LP-Relaxation Example: Vertex Cover

Problem [(Minimum) Vertex Cover]. Given an undirected graph, find a vertex cover with minimum number of vertices.

- Formulation by integer program:
  - $x_u = 1$ represents $u \in V$ is selected in the cover; $x_u = 0$ otherwise.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{v \in V} x_v & \\
\text{subject to} \quad & x_u + x_v \geq 1 & \forall (u, v) \in E \\
& x_v \in \{0, 1\} & \forall v \in V
\end{aligned}
$$

LP-Relaxation Example: Vertex Cover

Problem [(Minimum) Vertex Cover]. Given an undirected graph, find a vertex cover with minimum number of vertices.

- Relax it to a linear program below:

$$
\begin{array}{lll}
\text{minimize} & \displaystyle\sum_{v \in V} x_v & \\
\text{subject to} & x_u + x_v \geq 1 & \forall (u, v) \in E \\
 & 0 \leq x_v \leq 1 & \forall v \in V
\end{array}
$$

LP-Relaxation Example: Vertex Cover

- OPT(IP) – optimal objective value $\sum_{v \in V} x_v$ for IP
  - This is the objective we want for vertex cover
- OPT(LP) – optimal objective value $\sum_{v \in V} x_v$ for LP
- OPT(IP) $\geq$ OPT(LP): because LP has a larger feasible region.

$$\begin{array}{lll} \text{minimize} & \displaystyle\sum_{v \in V} x_v & \\ \text{subject to} & x_u + x_v \geq 1 & \forall (u,v) \in E \\ & x_v \in \{0,1\} & \forall v \in V \end{array}$$

Integer Program (IP)

$$\begin{array}{lll} \text{minimize} & \displaystyle\sum_{v \in V} x_v & \\ \text{subject to} & x_u + x_v \geq 1 & \forall (u,v) \in E \\ & 0 \leq x_v \leq 1 & \forall v \in V \end{array}$$

Linear Program (LP)

LP-Relaxation Example: Vertex Cover

An approximation algorithm for vertex cover:

- Formulate the problem as an integer program and obtain its LP-relaxation.
- Solve the linear program and obtain its optimal solution $\{x_v^*\}_{v \in V}$.
- Return $S = \{v \mid x_v^* \geq \frac{1}{2}\}$

Correctness

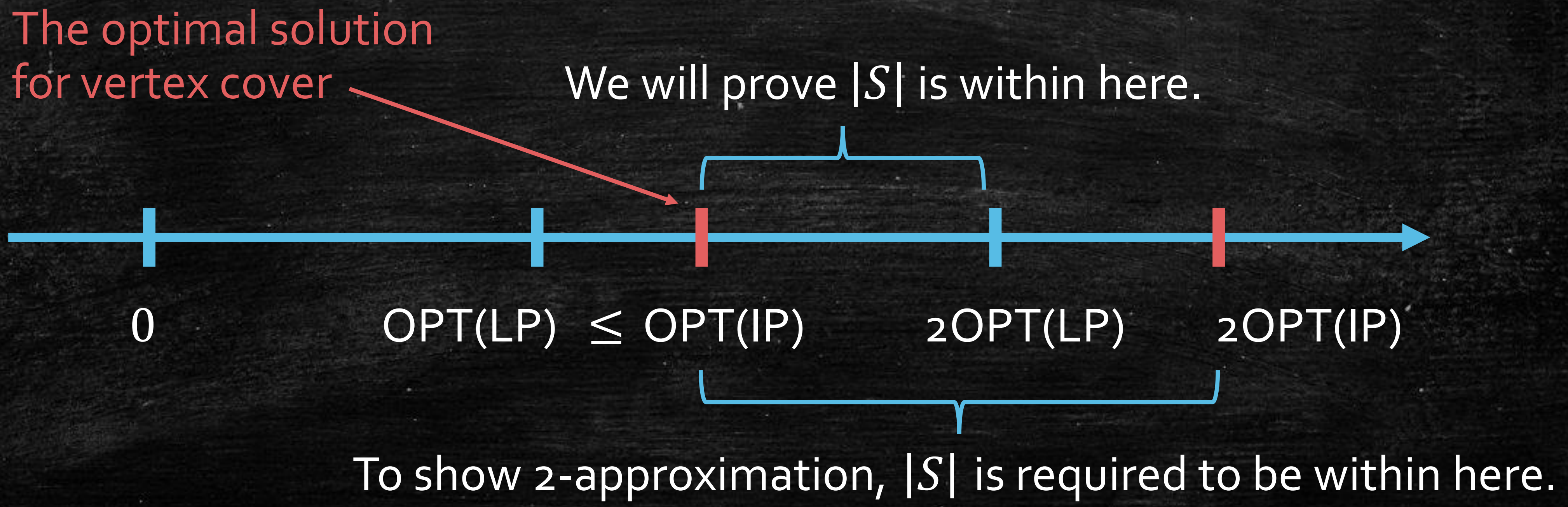
S returned by the algorithm is vertex cover.

- Proof. Consider an arbitrary edge $(u, v) \in E$.
- We have $x_u^* + x_v^* \geq 1$ by feasibility, which implies we have either $x_u^* \geq \frac{1}{2}$ or $x_v^* \geq \frac{1}{2}$, or both.
- By our algorithm, we have either $u \in S$ or $v \in S$, or both.

The algorithm is a 2-approximation.

The algorithm is a 2-approximation algorithm: $|S| \leq 2 \cdot \mathrm{OPT(IP)}$.

- Proof. Since we have $\mathrm{OPT(IP)} \geq \mathrm{OPT(LP)}$, it suffices to prove $|S| \leq 2 \cdot \mathrm{OPT(LP)}$.



The algorithm is a 2-approximation.

The algorithm is a 2-approximation algorithm: $|S| \le 2 \cdot \mathrm{OPT(IP)}$.

- Proof. Since we have $\mathrm{OPT(IP)} \ge \mathrm{OPT(LP)}$, it suffices to prove $|S| \le 2 \cdot \mathrm{OPT(LP)}$.
- $\mathrm{OPT(LP)} = \sum_{v \in V} x_v^* = \sum_{v : x_v^* < \frac{1}{2}} x_v^* + \sum_{v : x_v^* \ge \frac{1}{2}} x_v^*$
- $\ge \sum_{v : x_v^* < \frac{1}{2}} 0 + \sum_{v : x_v^* \ge \frac{1}{2}} \frac{1}{2} = \frac{1}{2} \cdot |S|$
- which implies $|S| \le 2 \cdot \mathrm{OPT(LP)}$.

Literature

- Vertex Cover cannot be approximated up to a factor smaller than 2 if the unique games conjecture is true.

Primal Dual Analysis

Dual Problem of Vertex Cover

- What is the Dual Problem of Vertex Cover?

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq 1 \qquad \forall (u,v) \in E$$

$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$

Linear Program (LP)

Dual Problem of Vertex Cover

Vertex Cover

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq 1 \qquad \forall (u,v) \in E$$

$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$

?

$$\text{maximize} \quad \sum_{e \in (u,v)} y_e$$

$$\text{subject to} \quad \sum_{v \in N(u)} y_{(u,v)} \leq 1 \qquad \forall u \in V$$

$$0 \leq y_{(u,v)} \qquad \forall (u,v) \in E$$

Dual Problem of Vertex Cover

Vertex Cover

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq 1 \qquad \forall (u, v) \in E$$

$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$

Maximum Matching

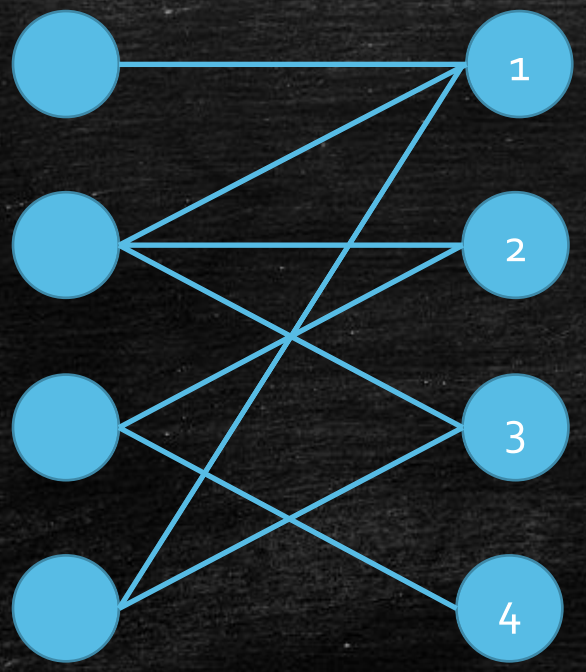
$$\text{maximize} \quad \sum_{e \in (u,v)} y_e$$

$$\text{subject to} \quad \sum\nolimits_{v \in N(u)} y_{(u,v)} \leq 1 \qquad \forall u \in V$$

$$0 \leq y_{(u,v)} \qquad \forall (u, v) \in E$$

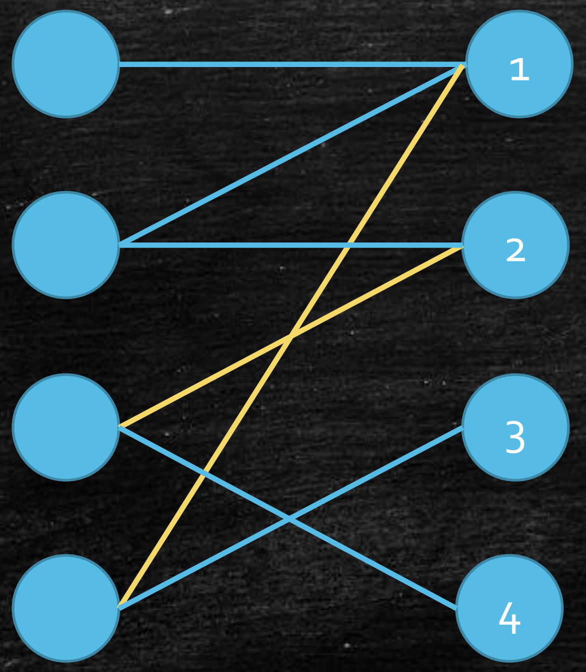
Greedy Algorithm for Matching

- Greedy: Fix an arbitrary order of B and match them to any one of its unmatched neighbor.



Greedy Algorithm for Matching

- Greedy: Fix an arbitrary order of B and match them to any one of its unmatched neighbor.



Prove it is 2-approximate by its Dual.

- Recall that it is 2-approximate.
- Primal Dual Analysis
  - Share the gain of matching to vertices $\left(y_{(u,v)} = 1\right)$.
  - Each matched edge (u, v): Gain of 1.
  - Total Gain = $\sum_e y_e = ALG$.
  - Gain Sharing
    - u get $0.5 \rightarrow x_u = 0.5$
    - v get $0.5 \rightarrow x_v = 0.5$
  - We have $\sum_u x_u = \sum_e y_e = ALG$.

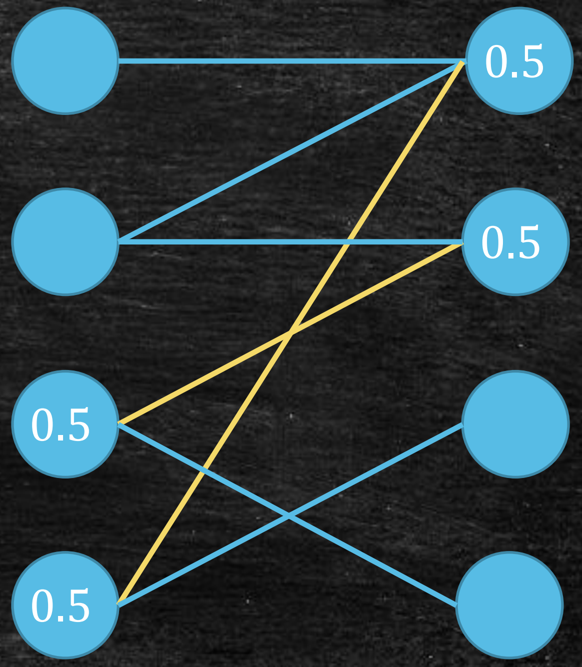
Is The Constructed Dual Solution Feasible?

Vertex Cover

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq 1 \qquad \forall (u, v) \in E$$

$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$



- If x is feasible,
- $\sum_v x_v \geq OPT(Dual) \geq OPT(primal) \geq OPT(Matching\ IP)$.
- $ALG = \sum_v x_v \geq OPT(Matching\ IP)$.

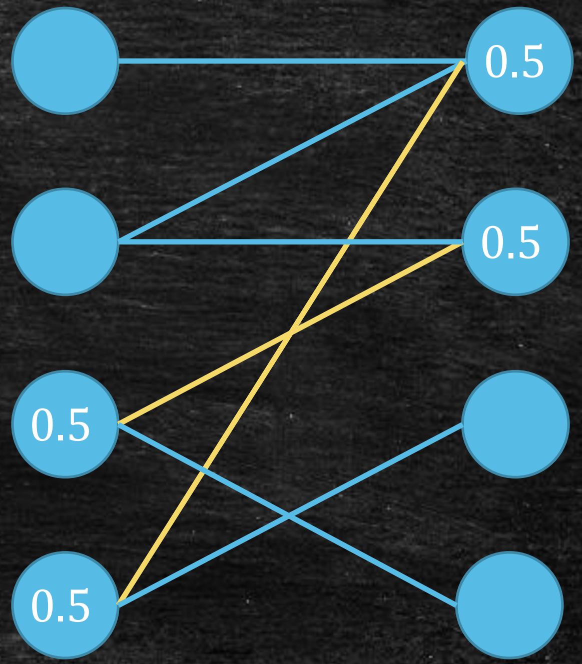
What do we have?

Vertex Cover

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq \mathbf{0.5} \qquad \forall (u,v) \in E$$

$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$



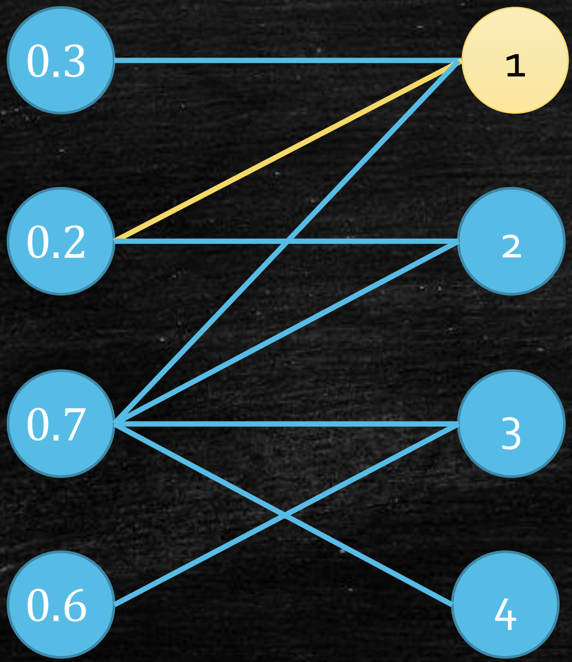
- If x is feasible,
- $2 \cdot \sum_v x_v \geq OPT(Dual) \geq OPT(primal) \geq OPT(Matching\ IP)$.
- $2 \cdot ALG = \sum_v x_v \geq OPT(Matching\ IP)$.

Ranking Algorithm

- A famous Algorithm for online matching.
- Proposed by Karp, Vazirani, and Vazirani in 1990.
- It is $1-\frac{1}{e}$ -competitive.
- $E(ALG) \geq \left(1-\frac{1}{e}\right) OPT.$
- The analysis in 1990's paper is extremely complex.
- Devanur et al. make it super simple in 2013.

Ranking Algorithm

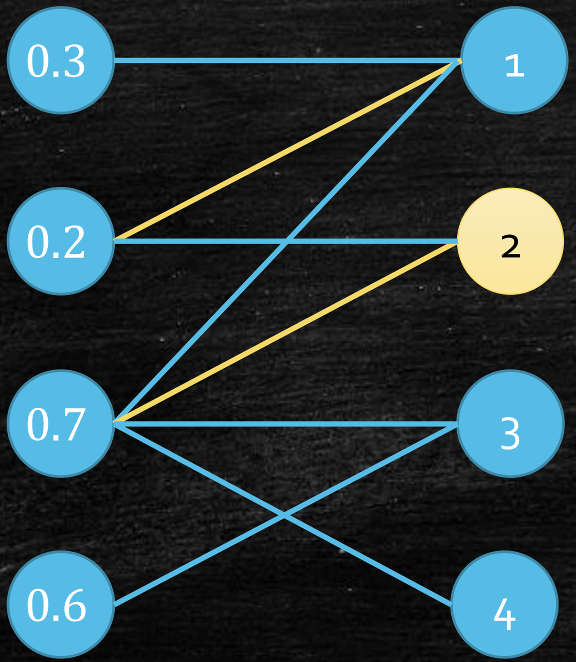
- Greedy: Fix an arbitrary order of B and match them to any one of its unmatched neighbor.
- Ranking: random a rank in $[0,1)$ for all vertices in A.



Choose the smallest rank!

Ranking Algorithm

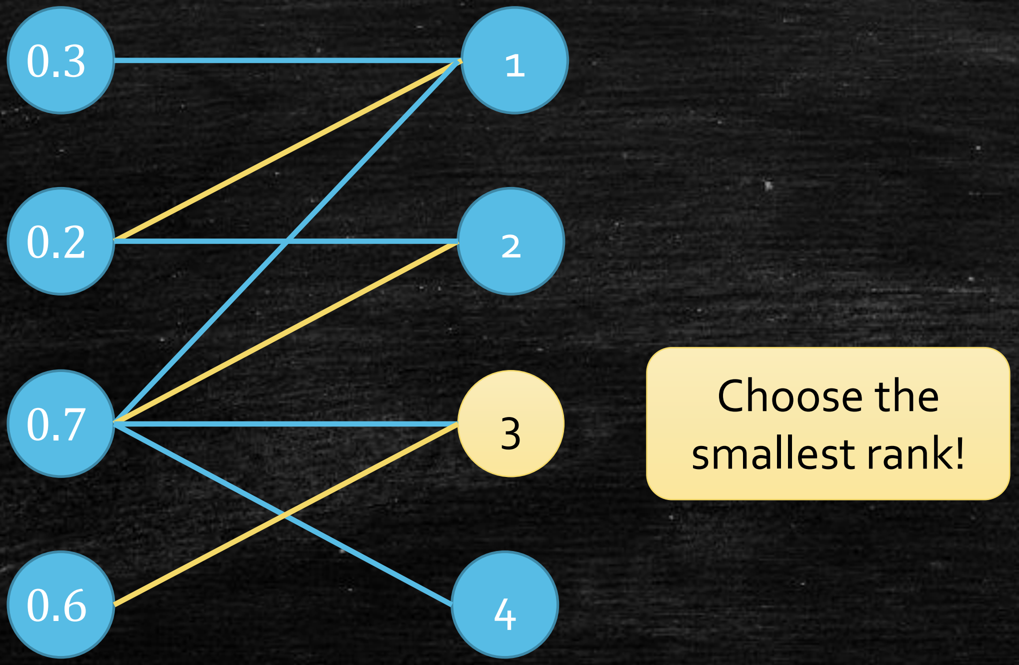
- Greedy: Fix an arbitrary order of B and match them to any one of its unmatched neighbor.
- Ranking: random a rank in $[0,1)$ for all vertices in A.



Choose the smallest rank!

Ranking Algorithm

- Greedy: Fix an arbitrary order of B and match them to any one of its unmatched neighbor.
- Ranking: random a rank in $[0,1)$ for all vertices in A.



Analysis of Ranking

- It is a randomized Algorithm, so we cares the expected Matching size Ranking: $E(ALG)$.
- Let do the same gain sharing thing.
- When Ranking Match one edge ($y_{(u,v)} = 1$)
  - ~~u get $0.5 \rightarrow x_u = 0.5$~~
  - ~~v get $0.5 \rightarrow x_v = 0.5$~~
  - Fix a gain sharing function $g(r) = e^{r-1}$.
  - v get $g(r_v) \rightarrow x_v = g(r_v)$.
  - u get $1 - g(r_v) \rightarrow x_u = 1 - g(r_v)$

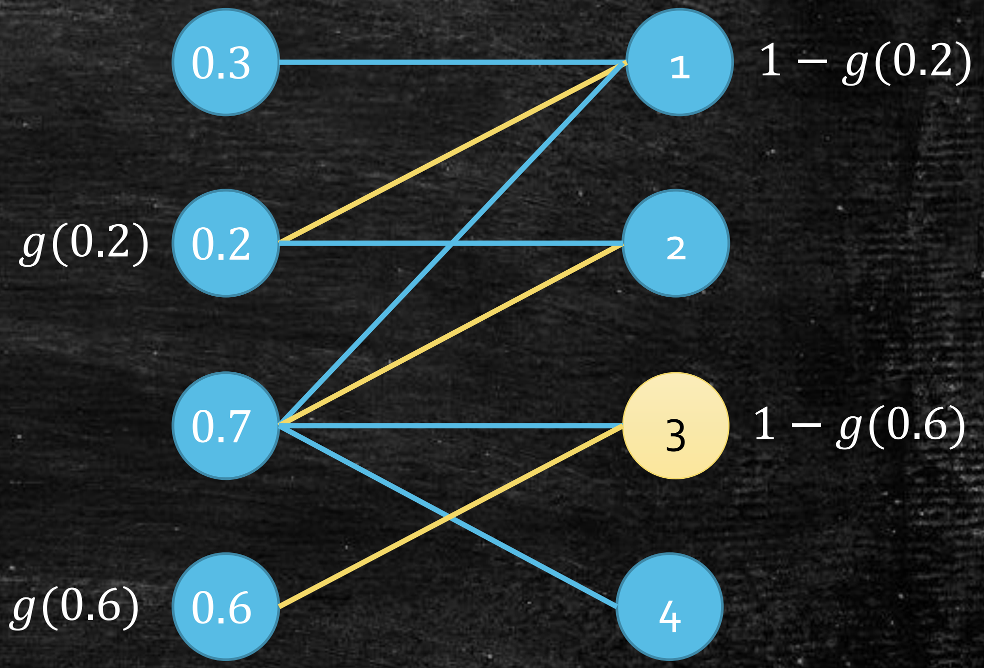
Primal Dual Analysis

Vertex Cover

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq 1 \qquad \forall (u,v) \in E$$

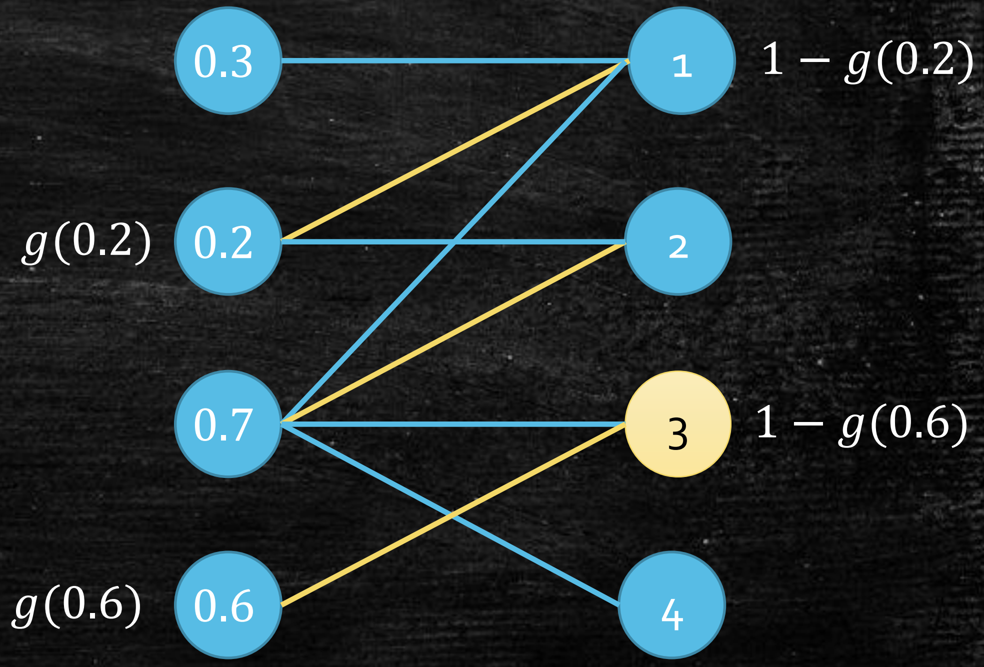
$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$



- We aim to prove:
- $E(y_u + y_v) \geq 1 - \frac{1}{e}$ for each $(u, v) \in E$.
- If it is true
  - $\frac{e}{e-1} E(y_v)$ is a feasible solution
  - $e/(e-1)E(ALG) = \sum_{v \in V} \frac{e}{e-1} E(y_v) \geq OPT(Matching\ IP)$

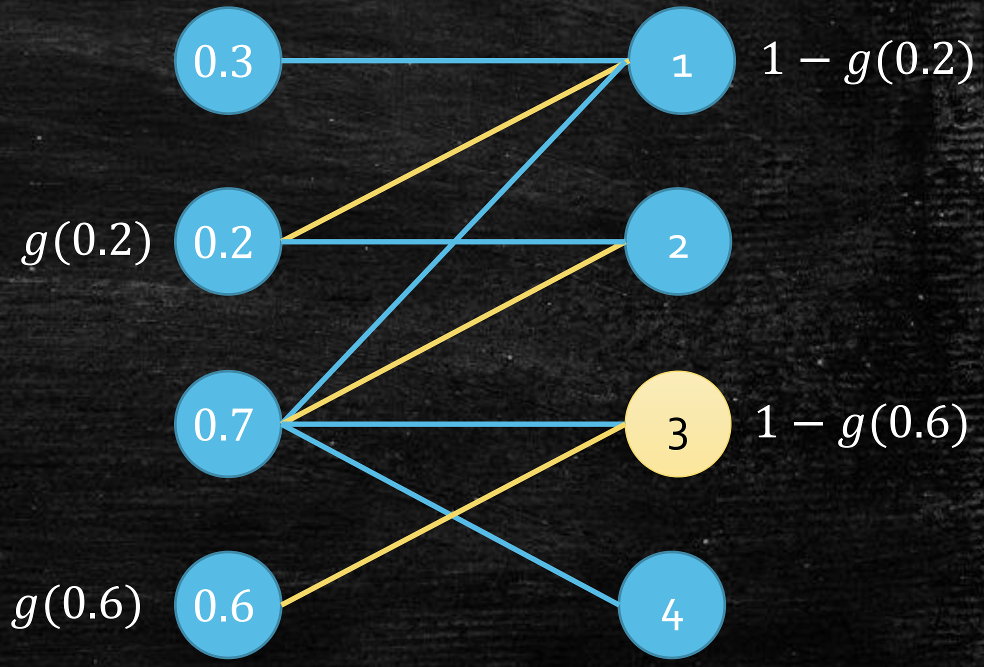
Gain Lower Bound

- Fix a pair (u, v).
- Fix all vertices' rank except v: r^{-v}
- We prove for any (u, v) any fixed rank,
- $E_{r_v}[y_u + y_v \mid r^{-v}] \geq 1 - \frac{1}{e}$
- If it is true
- $E[y_u + y_v] \geq 1 - \frac{1}{e}$



Gain Lower Bound

- Fix a pair (u, v).
- Fix all vertices' rank except v.
- Let us consider what happens to u if v does not exists in the graph.
  - Case 1: u match another vertex z.
  - Case 2: u match nothing.

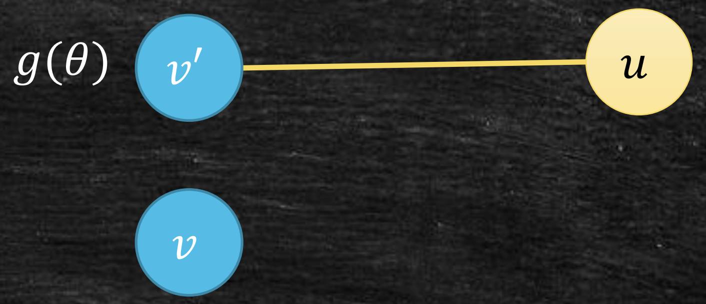


Case 1: u match nothing

- When we put back v.
- Whatever r_v, v will be matched.
- Question: will v always matched by u?
- $E_1[y_u + y_v] \geq \int_0^1 g(r)dr$

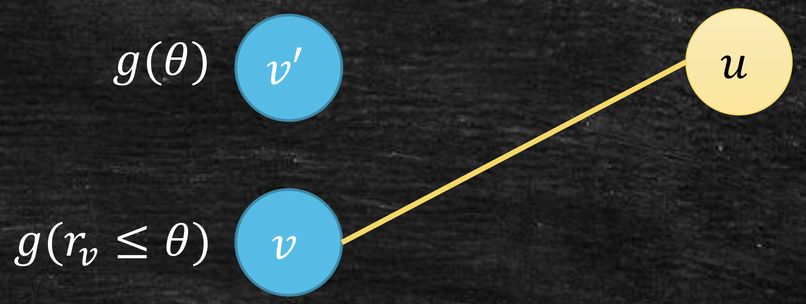
Case 2: u match a vertex

- Assume $r_{v'} = \theta$.
- What if we put back v with $r_v < \theta$?
  - Can we show v must be matched?
  - Can we show v must be matched by u?
  - $y_v \geq g(r_v)$
- What about $u's$ gain when we put back v?
  - Case 1: v do not change $u's$ choice $y_v = 1 - g(\theta)$.
  - Case 2: v changes $u's$ choice, what happens?



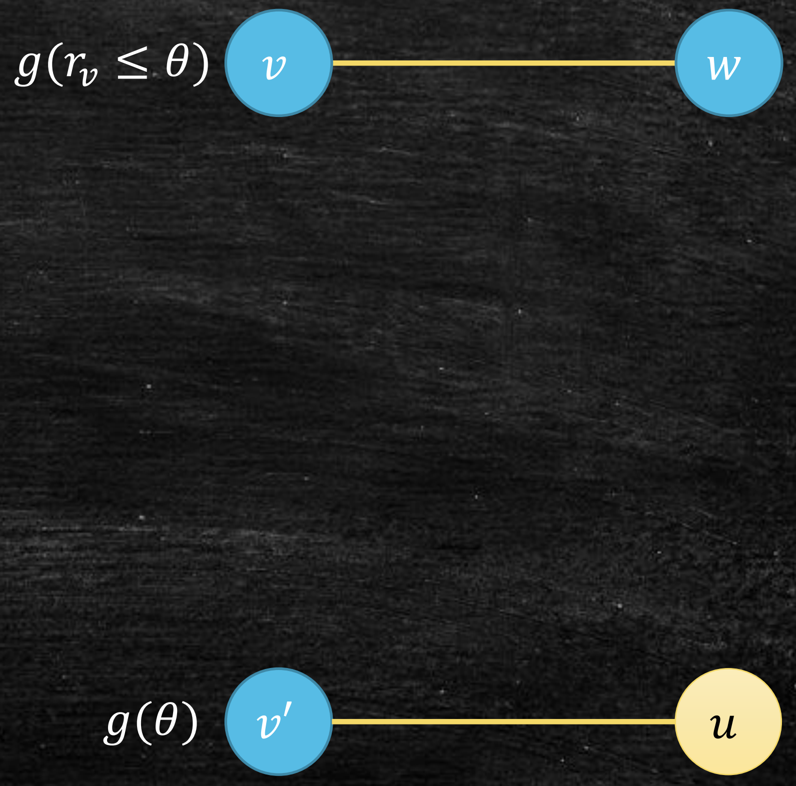
v changes $u's$ choice.

- Easy case:
  - u choose v $\rightarrow y_u > 1 - g(\theta)$



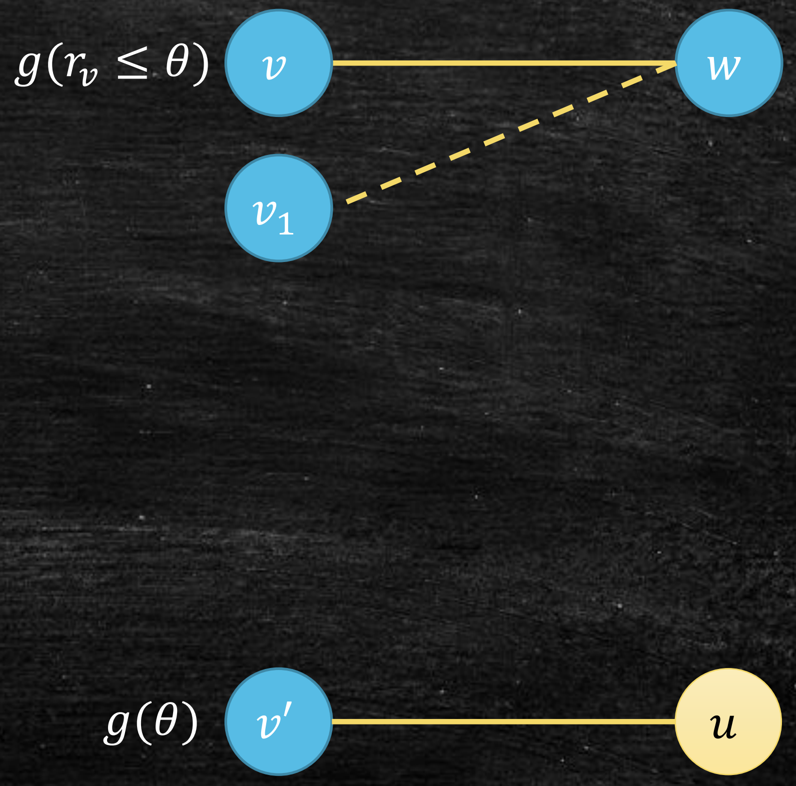
v changes $u's$ choice.

- Hard case:
  - w choose v



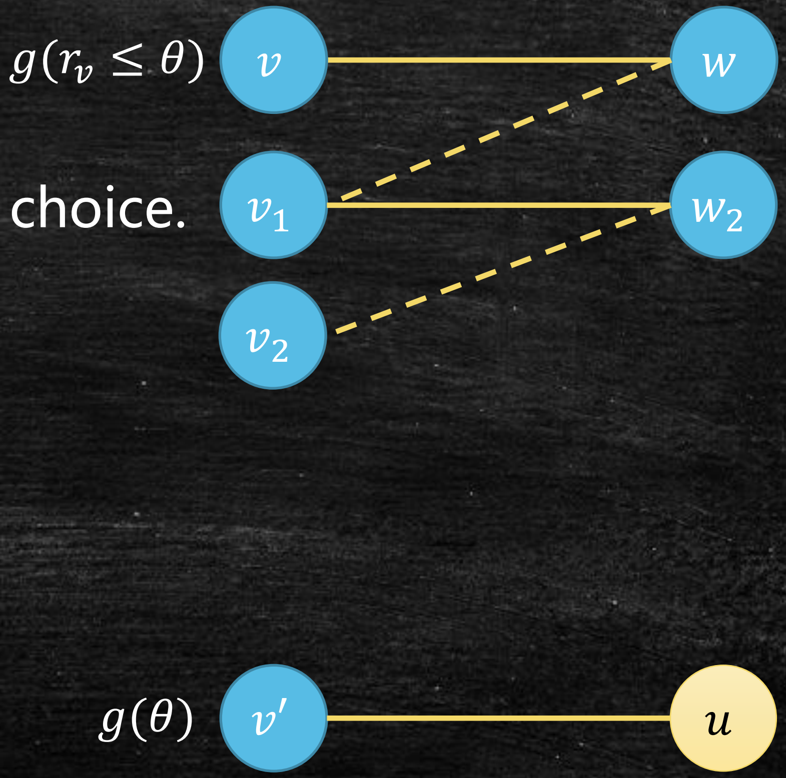
v changes $u's$ choice.

- Hard case:
  - w choose v
  - v_1 is $w's$ original choice.



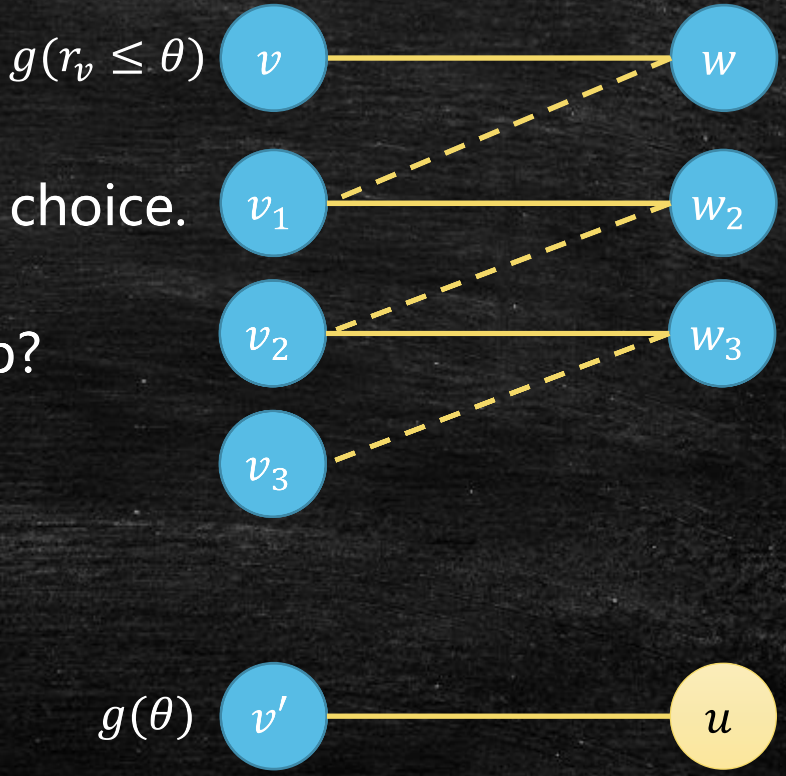
v changes $u's$ choice.

- Hard case:
  - w choose v
  - v_1 is $w's$ original choice.
  - w_2 choose v_1, do not want its original choice.



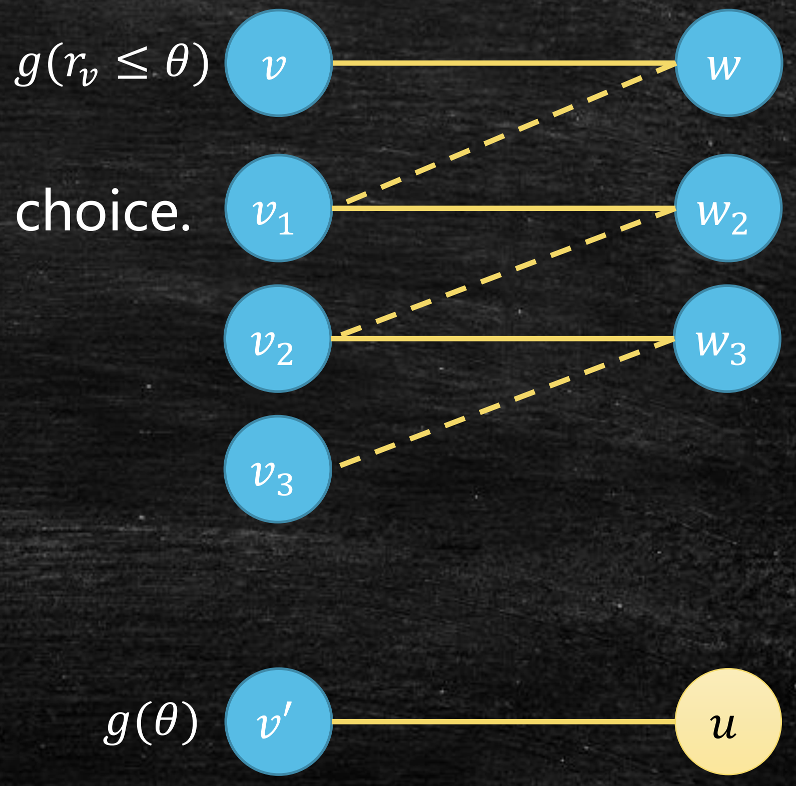
v changes $u's$ choice.

- Hard case:
  - w choose v
  - v_1 is $w's$ original choice.
  - w_2 choose v_1, do not want its original choice.
  - w_3 choose v_2, do not want v_3.
  - Question: when does the process stop?



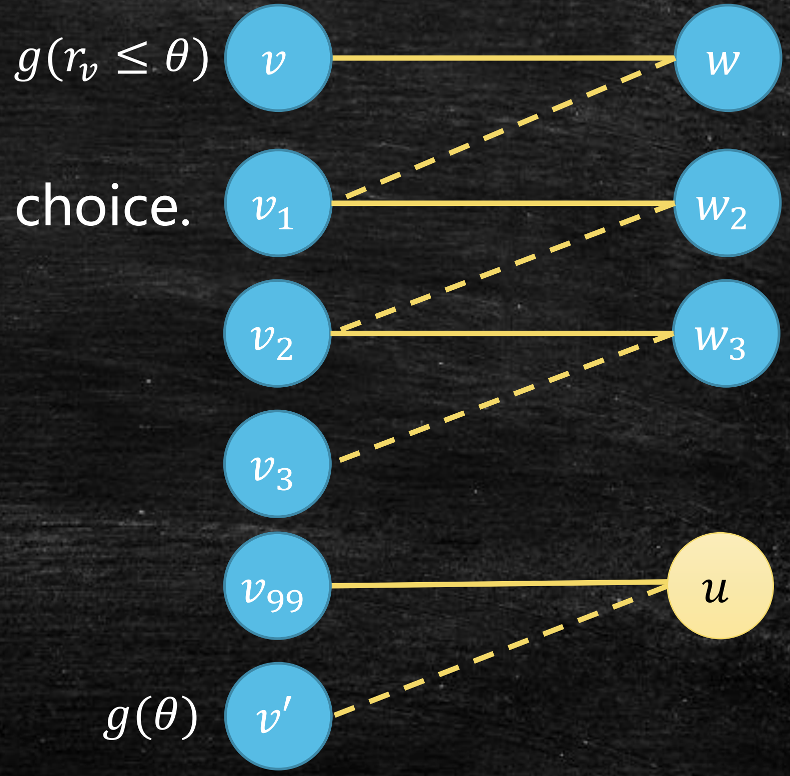
v changes $u's$ choice.

- Hard case:
  - w choose v
  - v_1 is $w's$ original choice.
  - w_2 choose v_1, do not want its original choice.
  - w_3 choose v_2, do not want v_3.
  - Why u changes his choice?



v changes $u's$ choice.

- Hard case:
  - w choose v
  - v_1 is $w's$ original choice.
  - w_2 choose v_1, do not want its original choice.
  - w_3 choose v_2, do not want v_3.
  - Why u changes his choice?
    - He prefer v_{99} to v'!
    - $r_{v_{99}} < r_{v'}$!
    - $1 - g\left(r_{v_{99}}\right) > 1 - g(r_{v'}) > 1 - g(\theta)$



Conclusion

- For a fixed (u, v), and a fixed rank for every v' other than v.
- Case 1: u match nothing:
  - $g(r) = e^{r-1}$
  - $E_{r_v}[y_u + y_v] \geq \int_0^1 g(r)dr = 1 - \frac{1}{e}$
- Case 2: u match $r_v = \theta$:
  - $y_v \geq g(r_v)$ if $y_v < \theta$
  - $y_u \geq 1 - g(\theta)$ for all $y_v \in [0,1)$
  - $E_{r_v}[y_u + y_v] \geq \int_0^\theta g(r)dr + 1 - g(\theta) = 1 - \frac{1}{e}$
- So, $E[y_u + y_v] \geq 1 - \frac{1}{\mathrm{e}}$

Primal Dual Analysis: Recall

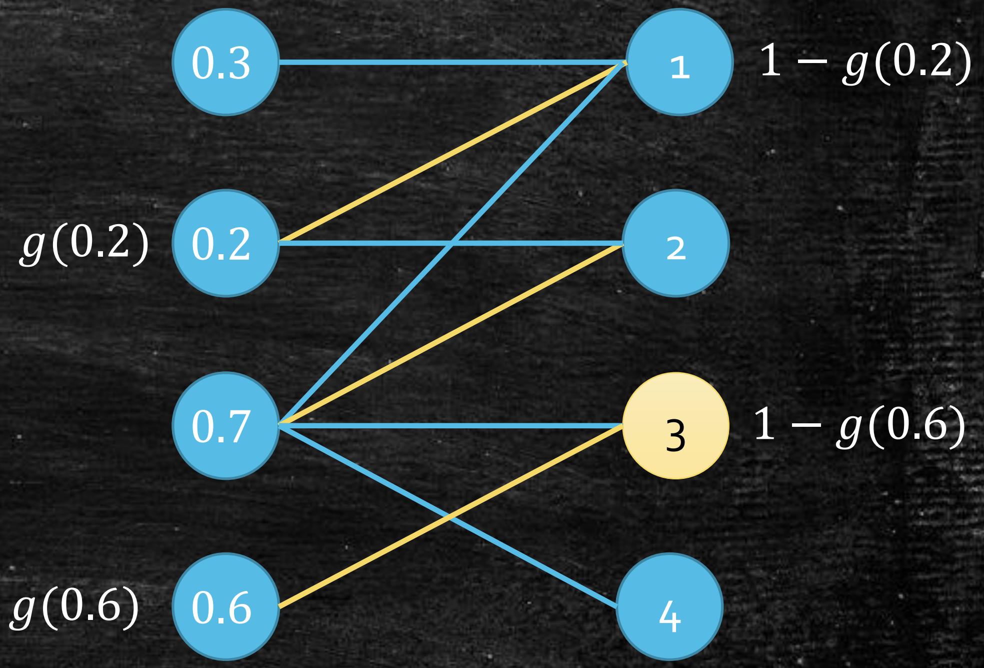
Vertex Cover

$$\text{minimize} \quad \sum_{v \in V} x_v$$

$$\text{subject to} \quad x_u + x_v \geq 1 \qquad \forall (u,v) \in E$$

$$0 \leq x_v \cancel{\leq 1} \qquad \forall v \in V$$

- We aim to prove:
- $E(y_u + y_v) \geq 1 - \frac{1}{e}$ for each $(u,v) \in E$.
- If it is true
  - $\frac{e}{e-1} E(y_v)$ is a feasible solution
  - $e/(e-1)E(ALG) = \sum_{v \in V} \frac{e}{e-1} E(y_v) \geq OPT(Matching\ IP)$



Analysis of Ranking

- $\frac{e}{(e-1)} E(ALG) \geq OPT$
- $E(ALG) \geq \left(1 - \frac{1}{e}\right) OPT$

Today’s Lecture

- Introduction to Linear Programming
- LP Duality Theorem
- LP-Relaxation – use LP to design approximation algorithms