Linear Programming

Linear Programming – Lecture Content

- LP Definition
- LP Motivation & application
- Solving LP
  - Convex space
  - Complexity
  - Simplex method
- IP
  - LP approximation using rounding technique
- LP Duality

LP Definition

Linear Programming Introduction

A Linear Programming model seeks to maximize or minimize a linear function, subject to a set of linear constraints.

The linear model consists of the following components:

- A set of decision variables.
- An objective function.
- A set of constraints.

Linear Programming

The name is historical, it should really be called Linear Optimization.

The problem consists of three parts:

A linear function to be maximized

$$\text{maximize } f(\mathbf{x}) = c_1x_1 + c_2x_2 + \ldots + c_nx_n$$

Problem constraints

$$\begin{aligned} \text{subject to} \quad & a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \leq b_1 \\ & a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n \leq b_2 \\ & \ldots \\ & a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \leq b_m \end{aligned}$$

Non-negative variables

$$\text{e.g. } x_1, x_2 \geq 0$$

Linear Programming

The problem is usually expressed in matrix form and then it becomes:

$$\begin{array}{rl} \text{Maximize} & \mathbf{c}^\top \mathbf{x} \\ \text{subject to} & \mathtt{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq 0 \end{array}$$

where $\mathtt{A}$ is a $m \times n$ matrix.

The objective function, $\mathbf{c}^\top \mathbf{x}$, and constraints, $\mathtt{A}\mathbf{x} \leq \mathbf{b}$, are all linear functions of $\mathbf{x}$.

Matrix Form Transformation

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

- Important for some mathematical analysis and software solvers usage
- For a constraint of "equal or greater then" $a_1x_1+a_2x_2+....+a_nx_n>=b_i$
  - Multiply be -1 to flip sign $-a_1x_1-a_2x_2-....-a_nx_n \leq -b_i$
- For a constraint of equality $a_1x_1+a_2x_2+....+a_nx_n=b_i$
  - write as two inequalities $a_1x_1+a_2x_2+....+a_nx_n \leq b_i$ & $a_1x_1+a_2x_2+....+a_nx_n \geq b_i$
- For minimum problem
  - Solve as maximum for Objective function –z (argmin(z) = argmax(-z))
- For a negative $x_i \leq 0$
  - Define new variable $x_i'=- x_i$, assign and solve
- For an unrestricted sign x_i (urs)
  - Define new variables $x_i' \geq 0$ and $x_i'' \geq 0$ as $x_i = x_i' - x_i''$, assign and solve.

LP
Motivation & Application

Linear Programming Motivation

- The Importance of Linear Programming
- Many real world problems lend themselves to linear programming modeling.
- Many real world problems can be approximated by linear models.
- There are well-known successful applications in:
    - Manufacturing
    - Marketing
    - Finance (investment)
    - Advertising
    - Agriculture

The Galaxy Industries Production Problem – A Prototype Example

Galaxy manufactures two toy doll models:

- Space Ray
- Zapper.

Resources are limited to

- 1000 pounds of special plastic.
- 40 hours of production time per week.

The Galaxy Industries Production Problem – A Prototype Example

Marketing requirement

- Total production cannot exceed 700 pieces.
- Number of pieces of Space Rays cannot exceed number of pieces of Zappers by more than 350.

Technological requirement

- Space Rays requires 2 pounds of plastic and 3 minutes of labor per piece.
- Zappers requires 1 pound of plastic and 4 minutes of labor per piece.

Profit

- Space Ray - $8 profit per piece
- produce Zappers - $5 profit per piece

Management is seeking a production schedule that will increase the company's profit.

The Galaxy Linear Programming Model

Decisions variables:

- X_1 = # Space Rays
- X_2 = # Zappers

Objective Function:

- Weekly profit, to be maximized

The Galaxy Linear Programming Model (2)

Max $8X_1 + 5X_2$ (Weekly profit)

subject to

$2X_1 + 1X_2 \leq 1000$ (Plastic)

$3X_1 + 4X_2 \leq 2400$ (Production Time)

$X_1 + X_2 \leq 700$ (Total production)

$X_1 - X_2 \leq 350$ (Mix)

$X_j \geq 0, \; j = 1,2$ (Nonnegativity)

Linear Programming – another Example

- Locating a new machine to an existing layout of 3 existing machines
- The 3 known machines are located at following (x, y) coordinates
  - (3,0), (0,-3), (-2,1)
- Goal is to locate the new machine at position (x_o, y_o) where sum of distances from other machines is minimal using street distance
- Street distance between point (x_d, y_d) to (x_s, y_s) is

$$\mathbf{d=|x_d-x_s|+|y_d-y_s|}$$

- **So,**

$$\textbf{Minimize } Z = |x_o-3|+|y_o|+|x_o|+|y_o+3|+|x_o+2|+|y_o-1|$$

Linear Programming – another Example (2)

Solution: absolute value cannot be included in a linear programming. Recall $|x| = \max\{x, -x\}$

Minimize $z = (P_{1x}+P_{1y}) + (P_{2x}+P_{2y}) + (P_{3x}+P_{3y})$

Subject to

$P_{1x} \geq -(x_o-3)$

$P_{1x} \geq x_o-3$

$P_{1y} \geq -(y_o)$

$P_{1y} \geq y_o$

$P_{2x} \geq -(x_o+2)$

$P_{2x} \geq x_o+2$

$P_{2y} \geq -(y_o-1)$

$P_{2y} \geq y_o-1$

$P_{3x} \geq -(x_o)$

$P_{3x} \geq x_o$

$P_{3y} \geq -(y_o+3)$

$P_{3y} \geq y_o+3$

$P_{ix}, P_{iy} \geq 0$ for $i=1,2,3$

- **Minimize** $|x_o-3|+|y_o|+|x_o|+|y_o+3|+|x_o+2|+|y_o-1|$
- x_o, y_o, P_{ix}, P_{iy} are our new variables
- P_{ix} represents horizontal distance to the i^{th} machine
- P_{iy} represents vertical distance to the i^{th} machine

Linear Programming – another Example (3)

- P_{ix} represents horizontal distance to the i^{th} machine
- P_{iy} represents vertical distance to the i^{th} machine
- The Objective function reflects the desire to minimize total distance between new machine to all others
- The constraints relate the P variables in terms of x_o, y_o
- The constraints for each P_{ix}(or P_{iy}) variable allow each P_{ix} (or P_{iy}) to equal the maximum of $x_o - x_i$ and $-(x_o - x_i)$
- Since Program is minimization and smallest any of variables can be is max $\{x_o - x_i, -(x_o - x_i)\}$ each P_{ix} naturally equal its least possible value
  - This value will be the absolute value of $x_o - x_i$

Solving LP

Linear Programming – 2D example

$$\max_{x_1, x_2} f(x_1, x_2)$$

Cost function:

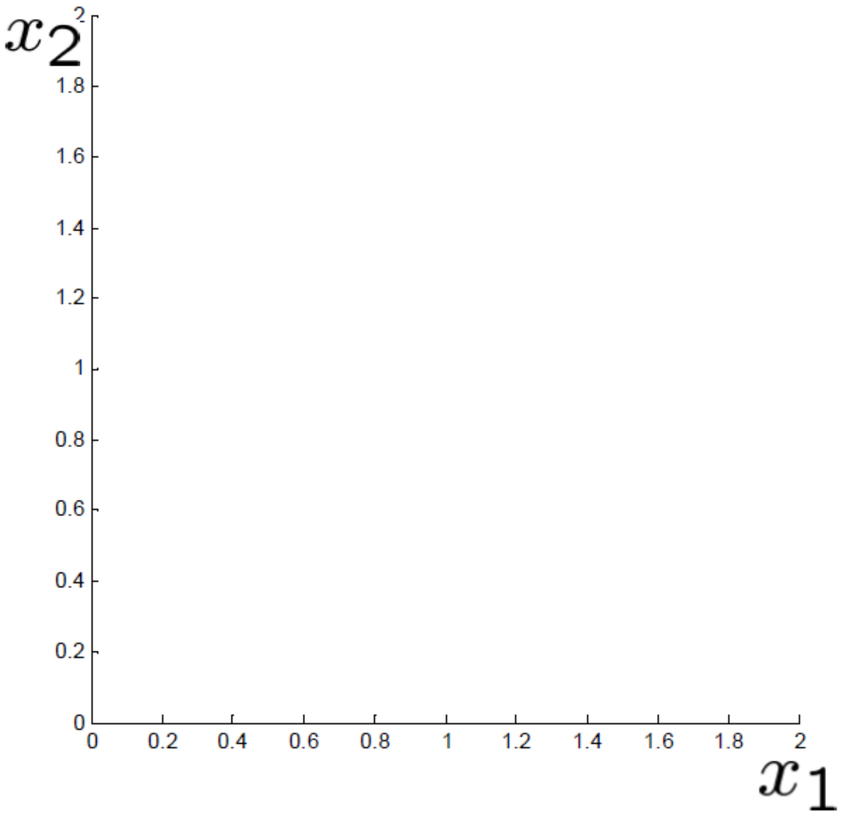
$$f(x_1, x_2) = 0x_1 + 0.5x_2$$

Inequality constraints:

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 &\leq 1.5 \\ x_1 + \frac{8}{3}x_2 &\leq 4 \\ \\ x_1, x_2 &\geq 0 \end{aligned}$$

Example – feasible region from inequalities

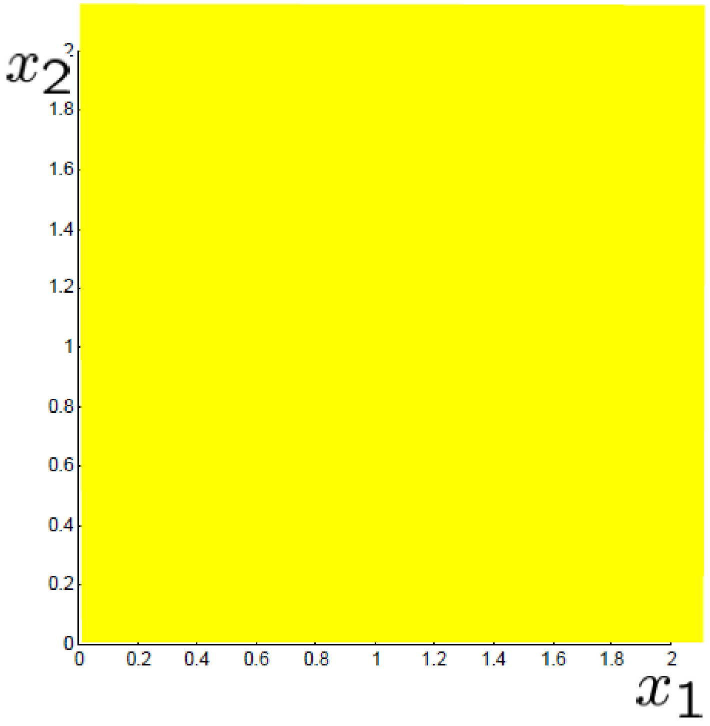
$$
\begin{aligned}
x_1 + x_2 &\leq 2 \\
x_1 &\leq 1.5 \\
x_1 + \frac{8}{3}x_2 &\leq 4 \\
x_1, x_2 &\geq 0
\end{aligned}
$$



Feasible Region

Inequality constraints:

$$x_1, x_2 \geq 0$$

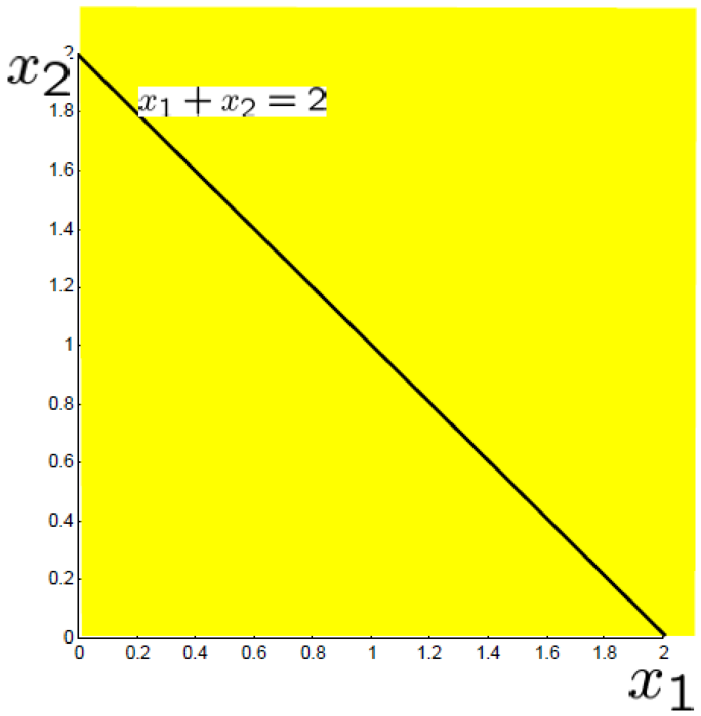


Feasible Region

Inequality constraints:

$$x_1 + x_2 \leq 2$$

$$x_1, x_2 \geq 0$$

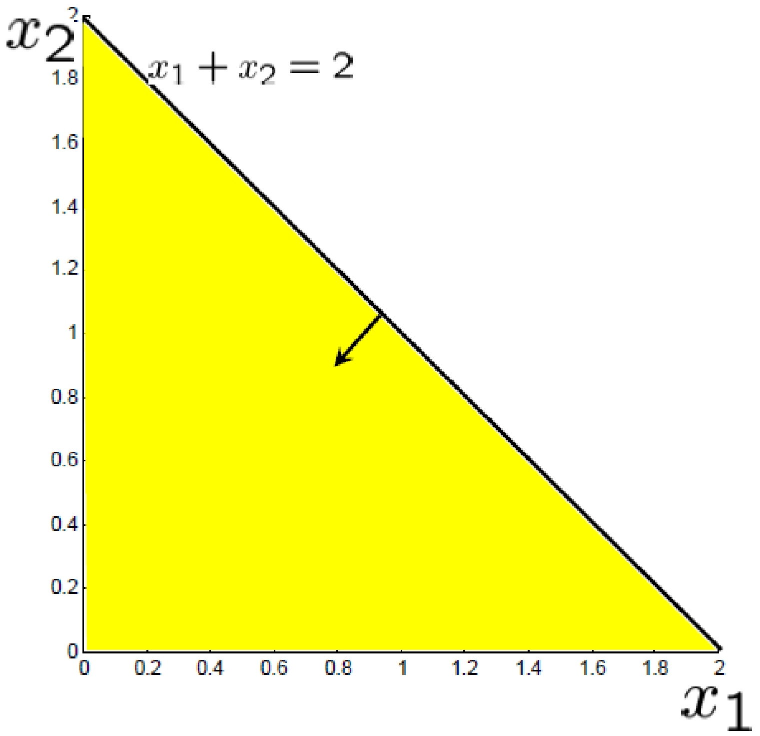


Feasible Region

Inequality constraints:

$$x_1 + x_2 \leq 2$$

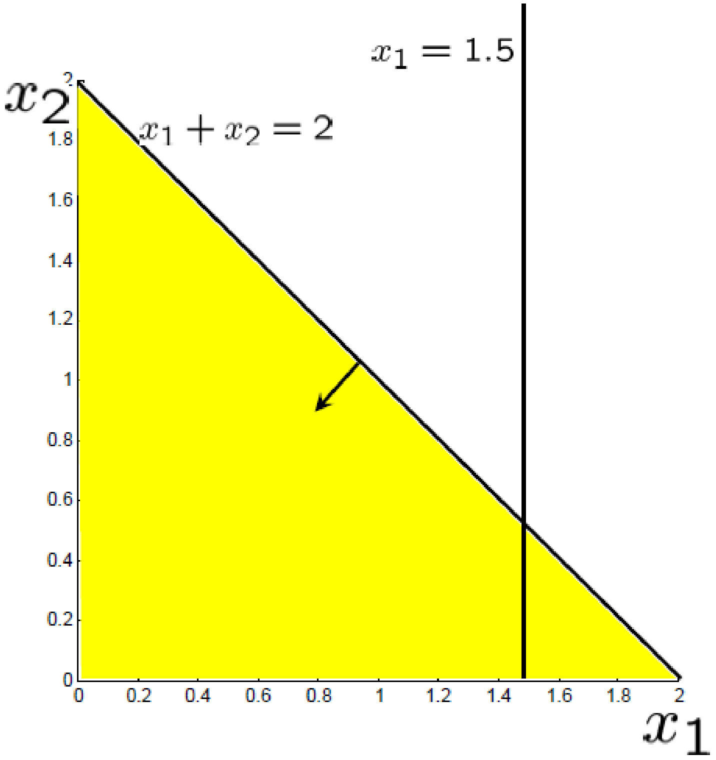
$$x_1, x_2 \geq 0$$



Feasible Region

Inequality constraints:

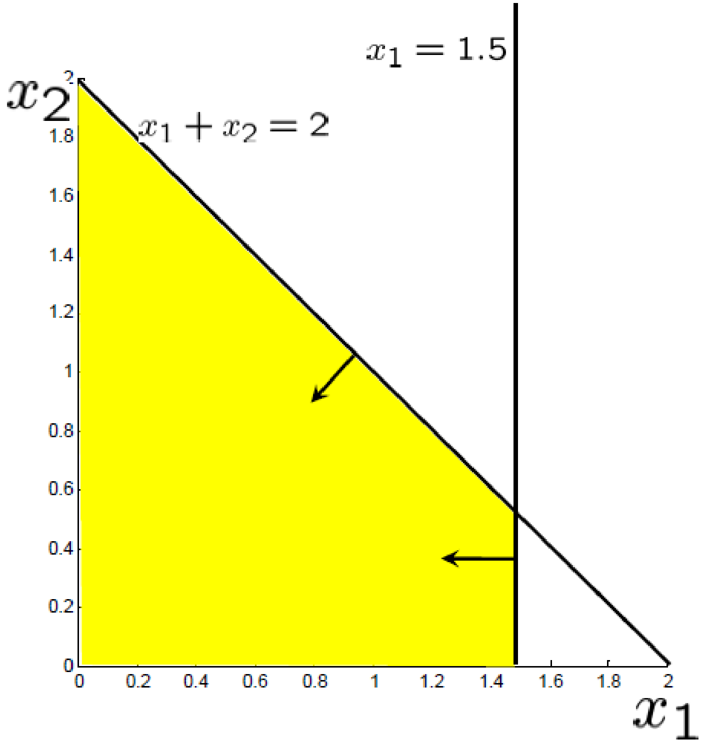
$$
\begin{aligned}
x_1 + x_2 &\leq 2 \\
x_1 &\leq 1.5 \\
x_1, x_2 &\geq 0
\end{aligned}
$$



Feasible Region

Inequality constraints:

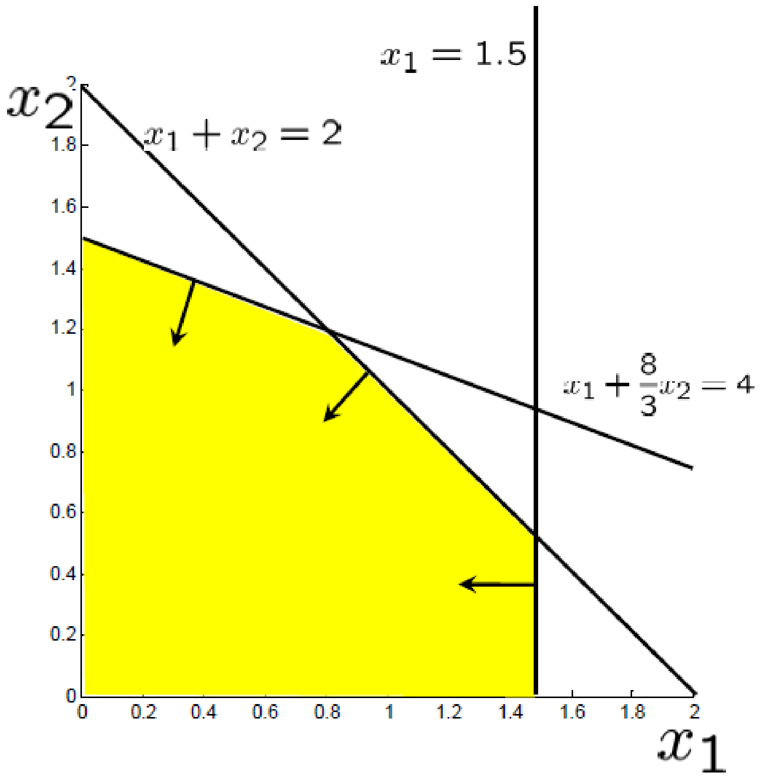
$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 &\leq 1.5 \\ \\ x_1, x_2 &\geq 0 \end{aligned}$$



Feasible Region

Inequality constraints:

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 &\leq 1.5 \\ x_1 + \frac{8}{3}x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$



LP example

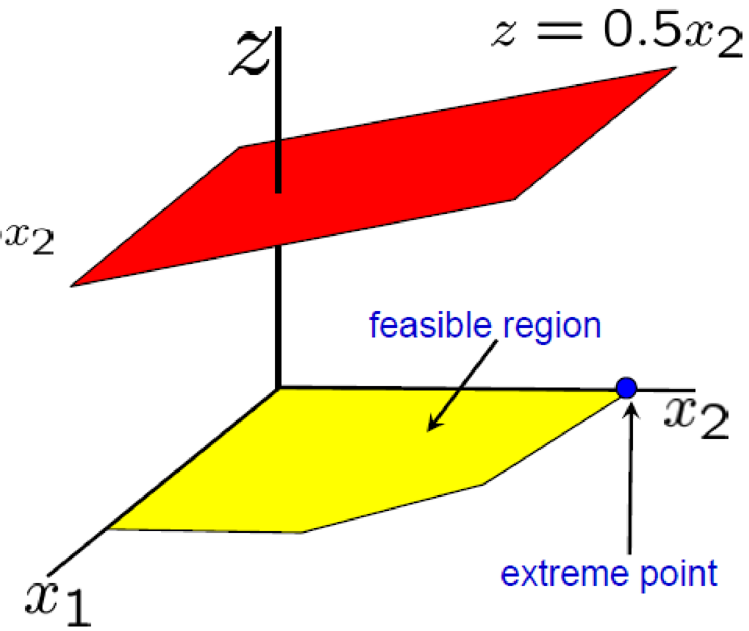
$$\max_{x_1, x_2} f(x_1, x_2)$$

Cost function:

$$f(x_1, x_2) = 0x_1 + 0.5x_2$$

Inequality constraints:

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 &\leq 1.5 \\ x_1 + \frac{8}{3}x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$



LP example

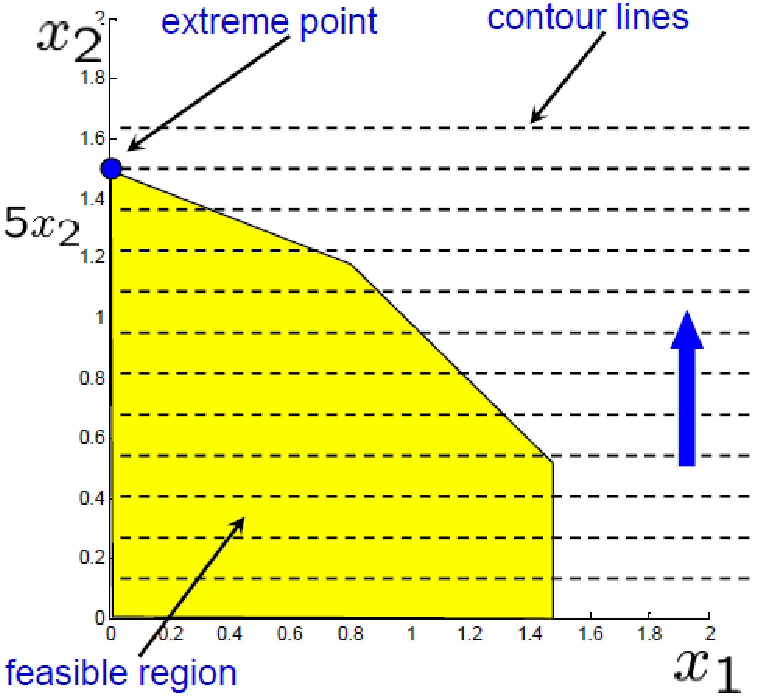
$$\max_{x_1, x_2} f(x_1, x_2)$$

Cost function:

$$f(x_1, x_2) = 0x_1 + 0.5x_2$$

Inequality constraints:

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 &\leq 1.5 \\ x_1 + \frac{8}{3}x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$



LP example – change of cost function

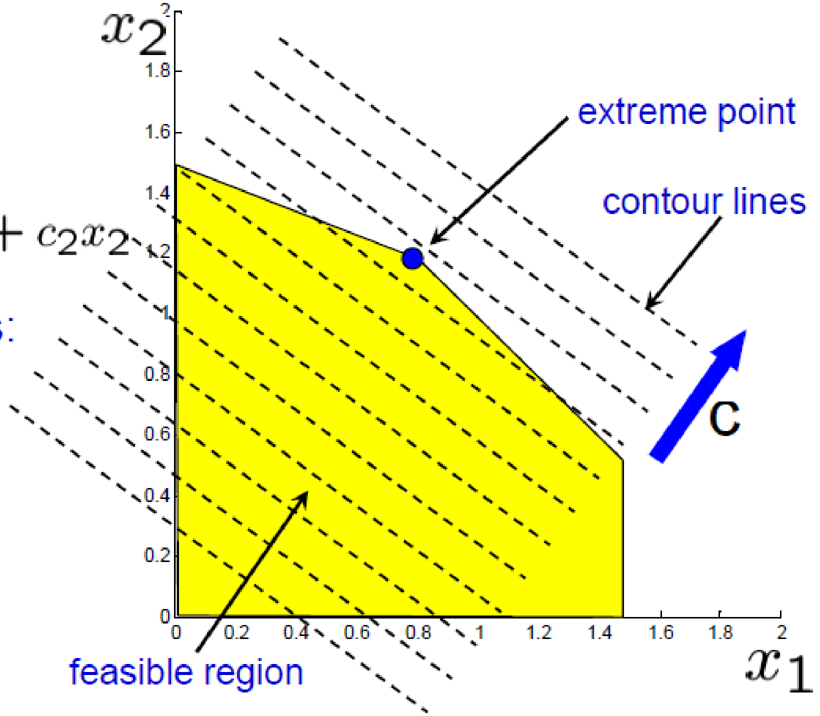
$$\max_{x_1, x_2} f(x_1, x_2)$$

Cost function:

$$\begin{aligned} f(x_1, x_2) &= \mathbf{c}^\top \mathbf{x} \\ &= c_1 x_1 + c_2 x_2 \end{aligned}$$

Inequality constraints:

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 &\leq 1.5 \\ x_1 + \frac{8}{3} x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$

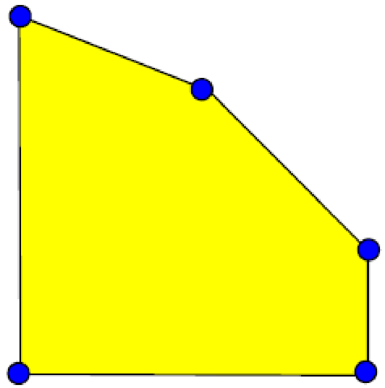


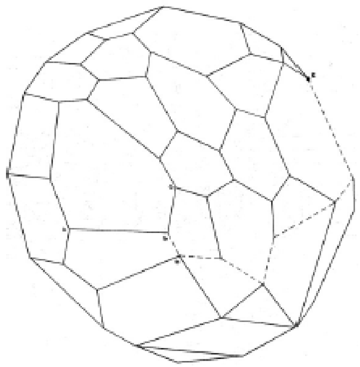
Linear Programming – optima at vertices

- The key point is that for any (linear) objective function the optima only occur at the corners (vertices) of the feasible polygonal region (never on the interior region).

- Similarly, in 3D the optima only occur at the vertices of a polyhedron (and in nD at the vertices of a polytope).

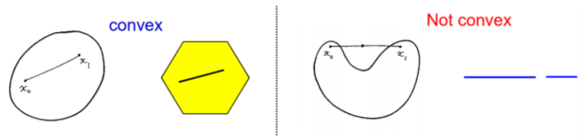
- However, the optimum is not necessarily unique: it is possible to have a set of optimal solutions covering an edge or face of a polyhedron.



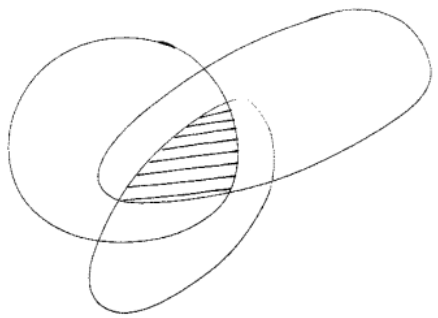


Convex Set

- A set $D \subseteq R^n$ is convex if the line joining n-dim vector points x and y lies inside D
  - Formally if for all $x,y \in D$ and for all $\lambda \in [0,1]$, the **line segment** $\lambda x+(1-\lambda)y \subseteq D$



- Convex set is closed under intersection (intersection of convex sets is convex)
  - The line segment between any two vectors in the intersection also belongs to each of the convex sets, and therefore lies in the intersection as well



Polyhedron are Convex

- Any equality of the form $a^T x = b$ is a **hyperplane**
- Any inequality of the form $a^T x \le b$ is a **halfspace**
- For the matrix form of $Ax \le b$, $x \ge 0$, the intersection of halfSpaces is a **polyhedron.** If a polyhedron is bounded it is called **polytope**
- halfspace is convex:

Given $a^T x \ge b$ and $a^T y \ge b$

Let $p = \lambda x + (1-\lambda) y$ and prove $\mathbf{a^T p \ge b}$ $(0 \le \lambda \le 1)$

1) $a^T x \ge b \rightarrow \lambda a^T x \ge \lambda b$

2) $a^T y \ge b \rightarrow (1-\lambda) a^T y \ge (1-\lambda) b$

So $1+2 \rightarrow \lambda a^T x + (1-\lambda) a^T y \ge b$

And we get $\lambda a^T x + (1-\lambda) a^T y = a^T(\lambda x + (1-\lambda) y) = a^T p \ge b$

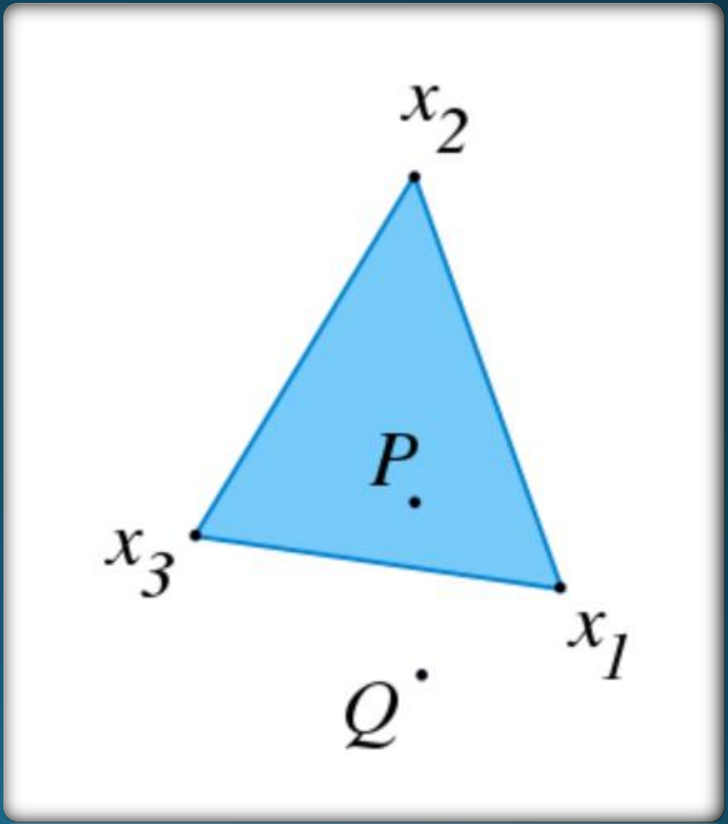
- Corollary: Polyhedron is convex
  - Intersection of convex sets (halfSpaces)

LP Optimum on vertices

- Linear function optimum over convex set is on the convex vertices
  - a **convex combination** is a linear combination of points (which can be vectors, scalars) where all coefficients are non-negative and sum to 1.

    $p = \Sigma\lambda_i x_i, \quad \Sigma\lambda_i = 1, \quad 0 \le \lambda_i \le 1$
  - As a particular example, every convex combination of two points lies on the line segment between the points.
  - **The convex polyhedron is identical to the set of all its vertices convex combinations.**
  - **Thus, $z(p) = c^T p$**

    $= c^T \Sigma\lambda_i x_i = \Sigma\lambda_i c^T x_i = \Sigma\lambda_i z(x_i) \le \Sigma\lambda_i z(x_{Max}) = z(x_{Max})\Sigma\lambda_i = z(x_{Max})$
- In addition, for LP Local optimum over adjacent vertices is also global



Feasible Set - Summary

Each linear inequality divides n-dimensional space into Two halfSpaces, one where The inequality is satisfied, and one where it's not.

Feasible Set : solutions to a family of linear inequalities.

The feasible set is the intersection of the halfSpaces where all inequalities are satisfied.

An intersection of halfSpaces is called a convex polyhedron. So the feasible set is a convex polyhedron.

A bounded and nonempty polyhedron is called a convex polytope.

Feasible Set

- Convex set
- segment line
- Convex combination
- hyperplane
- Halfspace
  - convexity
- Polyhedron\Polytope
  - Convexity
- LP Feasible Set
  - A convex polyhedron

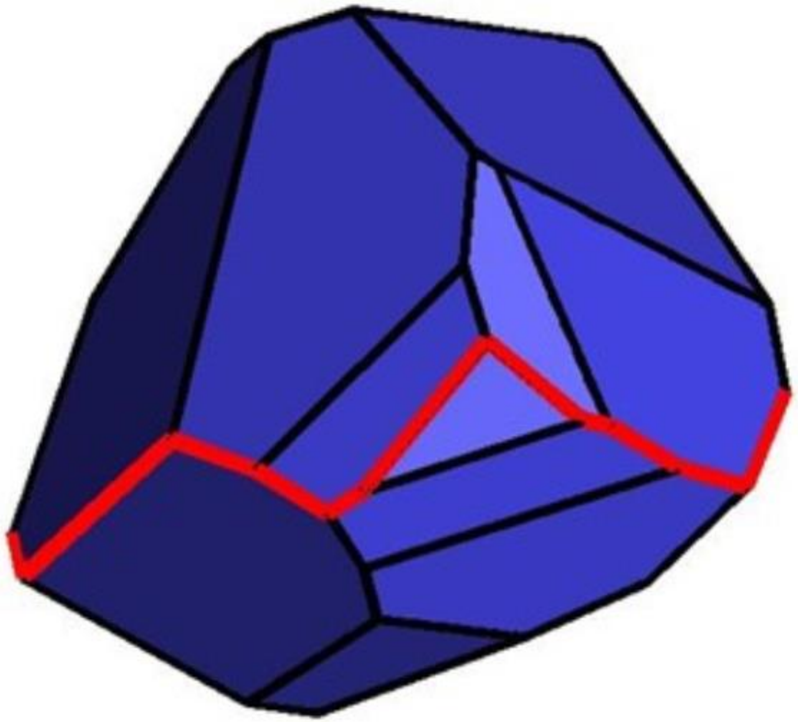
Optimal objective value

- with respect to feasible set
  - feasible set is empty
    - problem is not feasible
  - Feasible set is unbounded
    - if in direction of increase (for maximum problem) then cost function is unbounded on feasible set.
    - if in direction of decrease (for maximum problem) then cost has maximum on feasible set.
  - Feasible set is bounded and nonempty (a polytope)
    - cost has a maximum (or minimum) on feasible set.
    - This case is most common for real problems in economics and engineering

Solving Linear Programming – Complexity Outline

- Linear Programming is in P
- Interior-point Method (Poly time)
  - Karmarkar 84
- Ellipsoid Method (Poly time)
  - Khaciyan 79
- Simplex Method
  - Dantzig, 47
  - Worst-Case Exponential
    - The Klee Minty Construction, 72
  - Good in practice and wildly used
    - Average-Case: polynomial
    - Smoothed Complexity: Polynomial

The Simplex Method



Simplex Method – General Idea

- Two Phase Method
  - 1st Phase: Find Initial Vertex.
    - Converting LP to standard form using slack, access and artificial variables.
    - Similar Ideas of those shown in the begging when transforming into matrix form
    - Done in Polynomial Time
  - 2nd Phase:
    - Given Initial Vertex run greedily to next vertex where increase slope is the largest.
    - Stop when there is no option to increase.
    - Surely terminates since always increase function, hence no vertex repeats twice

Smoothed Analysis of Algorithms:

worst case

$$\max_x T(x)$$

average case

$$\text{avg}_r\, T(r)$$

smoothed complexity

$$\max_x \text{avg}_r\, T(x+\varepsilon r)$$

Smoothed Analysis of Algorithms

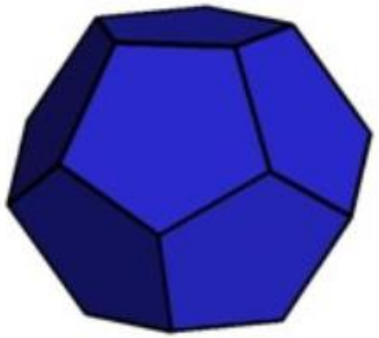
- Interpolate between Worst case and Average Case.
- Consider neighborhood of *every* input instance
- If low, have to be unlucky to find bad input instance

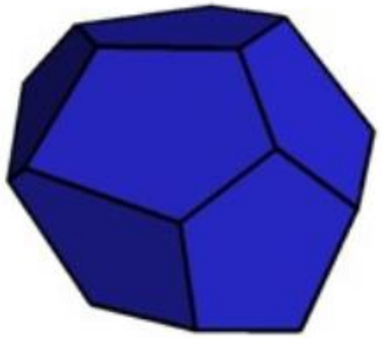
Smoothed Analysis of Simplex Method

$$\begin{array}{ll} \max & z^T x \\ \text{s.t.} & A\,x \le y \end{array}$$

$$\begin{array}{ll} \max & z^T x \\ \text{s.t.} & (A + \sigma G)\,x \le y \end{array}$$

G is Gaussian





Smoothed Analysis of Simplex Method

- Worst-Case: exponential
- Average-Case: polynomial
- Smoothed Complexity: polynomial

$$\begin{aligned} \max \quad & z^T x \\ \text{s.t.} \quad & \boldsymbol{a}_i^T x \leq \pm 1, \\ & \|\boldsymbol{a}_i\| \leq 1 \end{aligned} \quad \rightsquigarrow \quad \begin{aligned} \max \quad & z^T x \\ \text{s.t.} \quad & (\boldsymbol{a}_i + \sigma \boldsymbol{g}_i)^T x \leq \pm 1 \end{aligned}$$

Integer Programming

Integer Programming (IP)

- An LP problem with an additional constraint that variables will only get an integral value, maybe from some range.
- BIP – binary integer programming: variables should be assigned only 0 or 1.
- Can model many problems.
- NP-hard to solve!

An integer linear program in canonical form is expressed as:

$$\begin{aligned} &\text{maximize} && \mathbf{c}^\mathrm{T}\mathbf{x} \\ &\text{subject to} && A\mathbf{x} \le \mathbf{b}, \\ & && \mathbf{x} \ge \mathbf{0}, \\ &\text{and} && \mathbf{x} \in \mathbb{Z}^n, \end{aligned}$$

and an IP in standard form is expressed as

$$\begin{aligned} &\text{maximize} && \mathbf{c}^\mathrm{T}\mathbf{x} \\ &\text{subject to} && A\mathbf{x} + \mathbf{s} = \mathbf{b}, \\ & && \mathbf{s} \ge \mathbf{0}, \\ & && \mathbf{x} \ge \mathbf{0}, \\ &\text{and} && \mathbf{x} \in \mathbb{Z}^n, \end{aligned}$$

IP Formulation

Prove BIP is NPC

- BIP Search Problem:

Find x_n that Minimize $\Sigma c_n x_n$

$$
\begin{aligned}
\text{Subject to:} \quad & Ax \geq b \\
& x \geq 0 \\
& x_n \in \{0, 1\}
\end{aligned}
$$

- BIP Decision Problem:

Is there x_n such that $\Sigma c_n x_n \leq K$

$$
\begin{aligned}
\text{Subject to:} \quad & Ax \geq b \\
& x \geq 0 \\
& x_n \in \{0, 1\}
\end{aligned}
$$

Prove BIP is NPC

- BIP ∈ NP since given a certificate a verifier would check in poly time that all constraints are satisfied and the sum is at most K.
- Reduction of Vertex Cover Decision problem to BIP Decision Problem would prove it is NPH (next slide), and therefore NPC

- VC Search Problem:

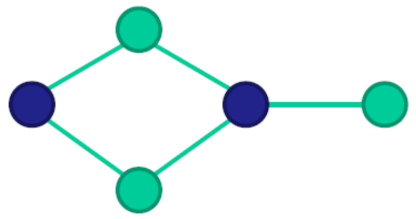
Given a graph G(V,E) and a number K find the smallest subset of nodes V'⊆V such that each edge in the graph touches at least one node of the subset V'.

VC $\leq_p$ BIP

- VC decision problem: Given a graph G(V,E) and a number K is there a subset of at most K nodes $V' \subseteq V$ such that each edge in the graph touches at least one node of the subset V'.

- So, for each vertex $v \in V$ we define x_v - if v in the cover then x_v equals 1, otherwise 0.

  We get the following BIP:



$$x_v = \begin{cases} 1 & v \in VC \\ 0 & v \notin VC \end{cases}$$

Does $\langle G,k \rangle$ Satisfy $\Sigma x_v \leq K$

Subject to: $x_v + x_u \geq 1$ for every edge $(u,v) \in V$

$x_v \in \{0,1\}$ for every $v \in V$

Solving IP using LP relaxation + rounding

- IP is NPC (BIP is polynomial-time reducible to IP).
- How to solve?
- One approach is relaxation of IP to LP. Can be done immediately by converting the integer constraint to real positive constraint
- OPT(LP) is at least not worse then OPT(IP) due to constraint relaxation (increasing feasible set).
- The solution of the LP will probably not be included in the original IP non-convex feasible set.
  - Only linear constraints are kept satisfied
  - Rounding should be done wisely so linear constraint stay satisfied
- Approximation analysis purpose to show additive or constant factor approximation boundary

Example:
LP relaxation + Rounding + Approximation

Formulation of vertex cover

- Input: undirected graph $G = (V, E)$
- Goal: Find $S \subseteq V$ such that

  1) $\forall \{i, j\} \in E$, $\{i, j\} \cap S \neq \phi$

  2) $|S|$ is minimized

Formulation of IP version of vertex cover:

$$\text{Let} \qquad x_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{if } i \notin S \end{cases}$$

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in V} x_i \\ \text{subject to} \quad & x_i \in \{0, 1\}, \ \forall i \in V \\ & x_i + x_j \geq 1, \ \forall \{i, j\} \in E. \end{aligned}$$

Example: LP relaxation + Rounding + Approximation

Formulation of LP Relaxation version of vertex cover:

$$\begin{aligned} \text{minimize} \quad & \sum_{i \in V} x_i \\ \text{subject to} \quad & 0 \le x_i \le 1, \forall i \in V \\ & x_i + x_j \ge 1, \forall \{i,j\} \in E. \end{aligned}$$

This is LP and solvable in polynomial time.

Rounding procedure:

$$x_i^* = \begin{cases} 1 & \text{if } x_i \ge \frac{1}{2} \\ 0 & \text{if } x_i < \frac{1}{2} \end{cases}$$

Example: LP relaxation + Rounding + Approximation

Claim *LP relaxation using rounding technique returns a feasible solution.*

Proof. Since $\forall(i,j)\ x_i + x_j \geq 1$, hence there exists at least one vertex which is greater than $\frac{1}{2}$; i.e., $x_i \geq \frac{1}{2} \Rightarrow x_i^* = 1$. Which guarantees the feasible solution existence.

Claim *Solving LP relaxation using rounding technique is 2-approximation for Vertex Cover.*

$$\boxed{x_i^* = \begin{cases} 1 & \text{if } x_i \geq \frac{1}{2} \\ 0 & \text{if } x_i < \frac{1}{2} \end{cases}}$$

Proof.

$$\sum_{i \in V} x_i^* \leq \sum_{i \in V} 2x_i = 2\sum_{i \in V} x_i = 2 \cdot \mathrm{OPT}_{LP} \leq 2 \cdot \mathrm{OPT}_{IP}$$

$$\Longrightarrow \text{Round} \leq 2 \cdot \mathrm{OPT}_{IP}$$

LP Duality

LP Duality

Primal and Dual Algebra

Primal

$$\begin{aligned} & Max \quad \sum_j c_j X_j \\ & s.t. \quad \sum_j a_{ij} X_j \le b_i \quad i = 1,\ldots,m \\ & \qquad X_j \ge 0 \quad j = 1,\ldots,n \end{aligned}$$

$$\begin{aligned} & Max \quad C'X \\ & s.t. \quad AX \le b \\ & \qquad X \ge 0 \end{aligned}$$

Dual

$$\begin{aligned} & \text{Min} \quad \sum_i b_i Y_i \\ & \text{s.t.} \quad \sum_i a_{ij} Y_i \ge c_j \quad j = 1,\ldots,n \\ & \qquad Y_i \ge 0 \quad i = 1,\ldots,m \end{aligned}$$

$$\begin{aligned} & Min \quad b'Y \\ & s.t. \quad A'Y \ge C \\ & \qquad Y \ge 0 \end{aligned}$$

Duality in LP

- Whenever we solve an LP problem, we solve two problems:
  - Primal resource allocation problem
  - Dual resource valuation problem
- Primal problem has n variables and m constraints
- Dual problem have m variables and n constraints
- The right-hand sides of the dual constraints come from the objective function coefficients in the primal problem
- Coefficient of the objective function in the dual problem come from the right-hand side of the original problem.
- Maximum in primal turns into minimum in dual

Example

Primal

$$
\begin{aligned}
&Max\ \ 40x_1 + 30x_2 && (profits)\\
&s.t. \quad x_1 + \ \ x_2 \le 120 && (land)\\
&\qquad\ \ 4x_1 + 2x_2 \le 320 && (labor)\\
&\qquad\ \ \ x_1, \quad x_2 \ge \ \ 0
\end{aligned}
$$

Dual

$$
\begin{aligned}
&\qquad (land) \quad (labor)\\
&Min\ \ 120y_1 + 320y_2\\
&s.t. \quad y_1 + \ 4y_2 \ge 40 && (x_1)\\
&\qquad\ \ y_1 + \ 2y_2 \ge 30 && (x_2)\\
&\qquad\ \ y_1, \quad y_2 \ge \ 0
\end{aligned}
$$

Relations between Primal and Dual

The dual of the dual problem is again the primal problem.

Weak duality : Let x and y be any feasible solution to the PLP and DLP respectively. Then $c^Tx \leq y^Tb$.

Strong duality : if PLP is feasible and has a finite optimum then DLP is feasible and has a finite optimum.

Furthermore, if x* and y* are optimal solutions for PLP and DLP then $c^Tx^* = y^{*T}b$

Theorem (The Weak Duality Theorem).

Let $P = \max(c^\top x \mid Ax \leq b, x \geq 0, x \in \mathbb{R}^n)$,
and let D *be its dual LP*, $\min(b^\top y \mid A^\top y \geq c, y \geq 0, y \in \mathbb{R}^m)$.

If x *is a feasible solution for* P
and y *is a feasible solution for* D, *then* $c^\top x \leq b^\top y$.

Proof.

$$
\begin{aligned}
c^\top x &= x^\top c \\
&\leq x^\top (A^\top y) && \text{(Since } y \text{ feasible for } D \text{ and } x \geq 0\text{)} \\
&= (Ax)^\top y \\
&\leq b^\top y && \text{(Since } x \text{ is feasible for } P \text{ and } y \geq 0\text{)} \quad \square
\end{aligned}
$$

Four Possible Primal Dual Problems

Dual Primal	Finite optimum	Unbounded	Infeasible
Finite optimum	1	x	x
Unbounded	x	x	2
Infeasible	x	3	4

Corollary of Weak Duality

Primal unbounded

↓

max(P) = ∞

↓

D infeasible because
min(D) < ∞ if it was feasible

Four Possible Primal Dual Problems

Primal \ Dual	Finite optimum	Unbounded	Infeasible
Finite optimum	1	x	x
Unbounded	x	x	2
Infeasible	x	3	4

Corollary of Weak Duality

Dual unbounded

↓

min(D) = -∞

↓

P infeasible because
max(P) > -∞ if it was feasible

Four Possible Primal Dual Problems

Primal \ Dual	Finite optimum	Unbounded	Infeasible
Finite optimum	1	x	x
Unbounded	x	x	2
Infeasible	x	3	4

Specific Case Example:

It's possible for both P,D to be infeasible. For example:

A=0, b<0, c>0

Four Possible Primal Dual Problems

Primal \ Dual	Finite optimum	Unbounded	Infeasible
Finite optimum	1	x	x
Unbounded	x	x	2
Infeasible	x	3	4

Remaining possibilities
Due to **strong duality** (if one is feasible, then also the other one)

Four Possible Primal Dual Problems

Primal \ Dual	Finite optimum	Unbounded	Infeasible
Finite optimum	1	x	x
Unbounded	x	x	2
Infeasible	x	3	4

Questions

THANKS!

Questions:

- What is the time complexity of the simplex method according to:
  1. Worst-Case analysis?
  2. Smoothed complexity analysis?
- IP and LP – which one belong to P / NPC?
- What characteristics of LP causes its optimum to be on the vertices?