CSC373: Algorithm Design, Analysis and Complexity Fall 2017

DENIS PANKRATOV

NOVEMBER 1, 2017

Linear Function

$f: \mathbb{R}^n \to \mathbb{R}$ is linear if it can be written as $f(x) = a^T x$ for some $a \in \mathbb{R}^n$

Example: $f(x_1, x_2) = 3x_1 - 5x_2 = \binom{3}{-5}^T \binom{x_1}{x_2}$

Linear equation: $f(x) = c$ where f is linear and $c \in \mathbb{R}$

Example: $3x_1 - 5x_2 = 4$

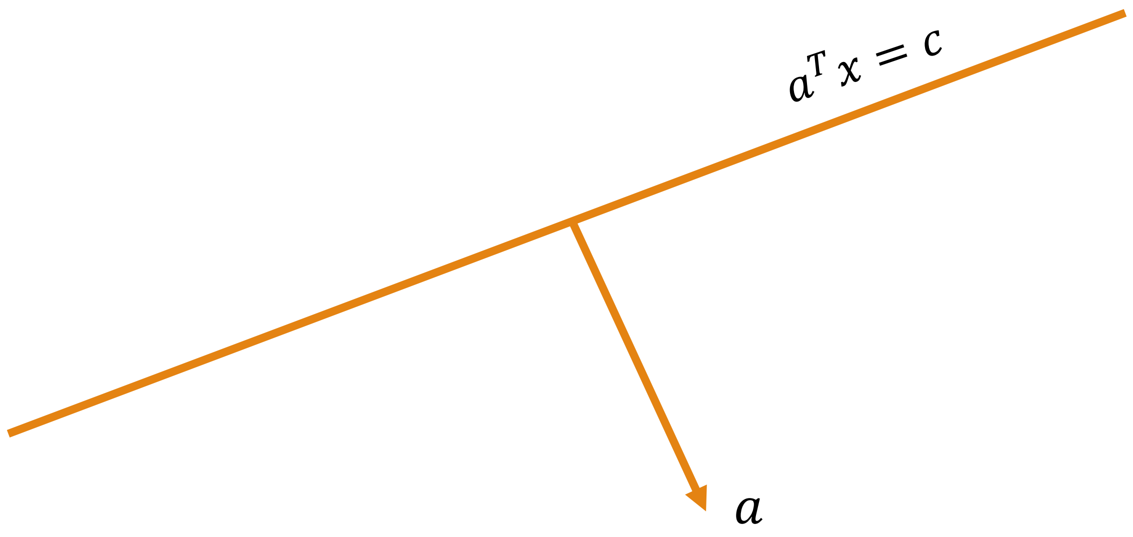
Geometrically: a line

Example: $x_2 = \frac{3}{5}x_1 - \frac{4}{5}$

Linear Function

$a^T x = c$ is geometrically a line in 2D, a plane in 3D, and a hyperplane in nD

a is a normal vector, i.e., the hyperplane is perpendicular to a

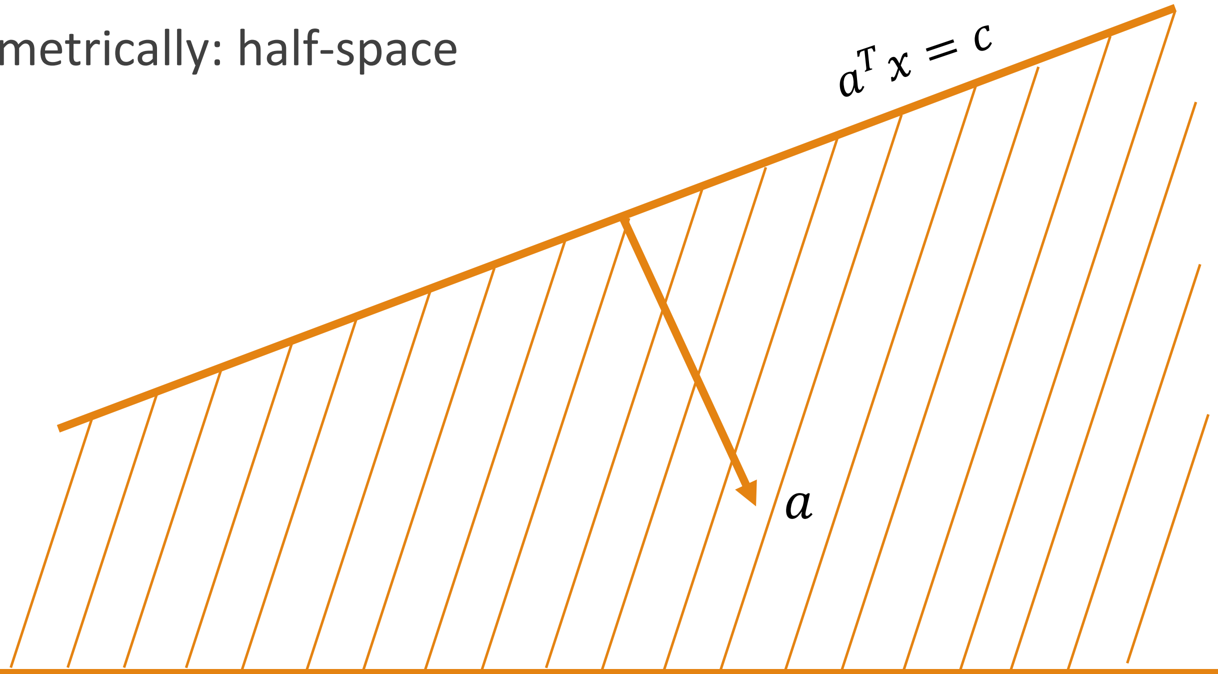


Linear Inequality

$f(x) \geq c$ where f is linear and $c \in \mathbb{R}$ or $f(x) \leq c$

Example: $3x_1 - 5x_2 \geq 4$

Geometrically: half-space



Linear Programming (LP)

Could be min

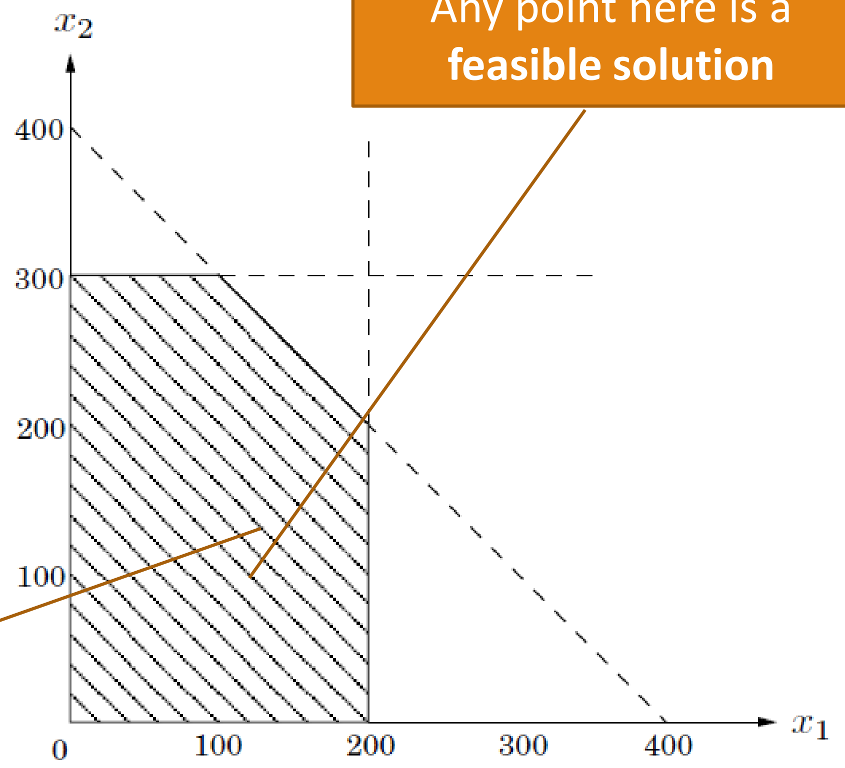
Linear objective!

Linear constraints: inequalities/equalities

$$
\begin{aligned}
\textbf{Objective function} \quad & \max x_1 + 6x_2 \\
\textbf{Constraints} \quad & x_1 \leq 200 \\
& x_2 \leq 300 \\
& x_1 + x_2 \leq 400 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Linear Programming, Geometrically

Objective function	$\max x_1 + 6x_2$
Constraints	$x_1 \leq 200$
	$x_2 \leq 300$
	$x_1 + x_2 \leq 400$
	$x_1, x_2 \geq 0$

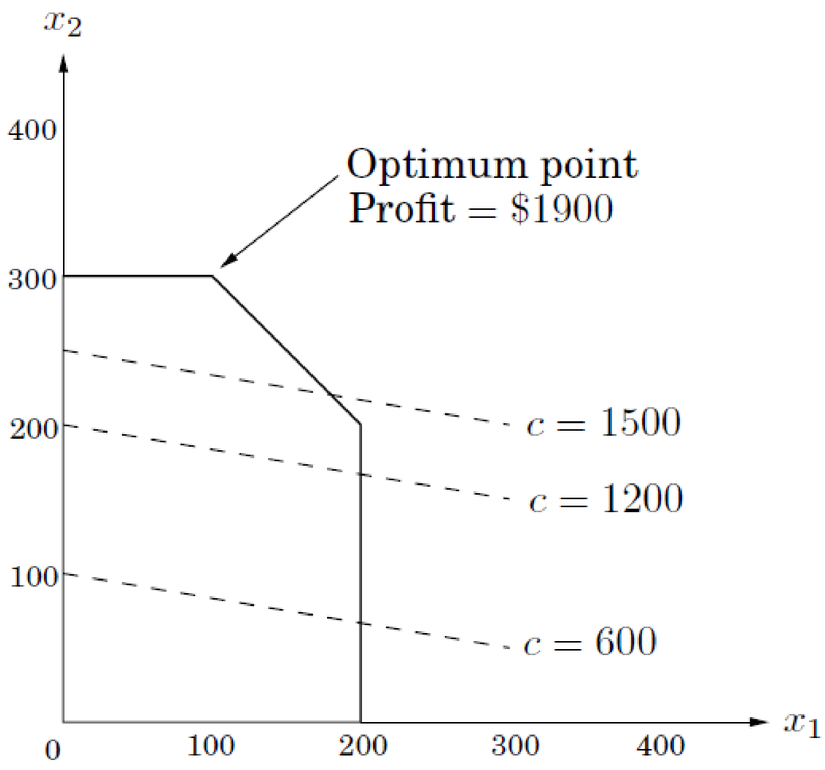


Any point here is a **feasible solution**

Feasible region – polytope, aka intersection of half-spaces!

Linear Programming, Geometrically

Objective function	$\max x_1 + 6x_2$
Constraints	$x_1 \leq 200$
	$x_2 \leq 300$
	$x_1 + x_2 \leq 400$
	$x_1, x_2 \geq 0$

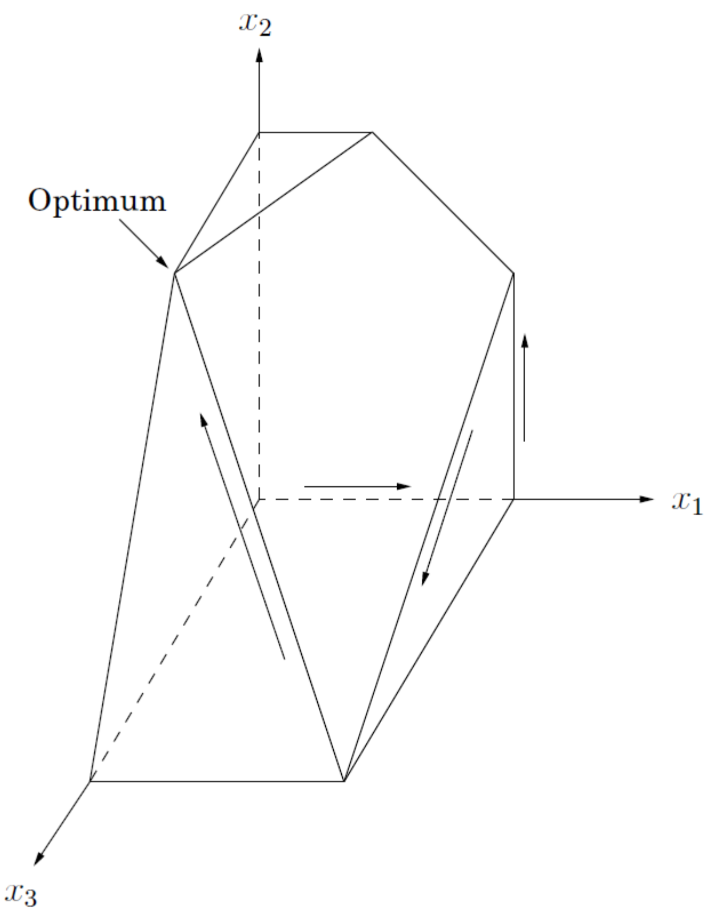


Linear Programming, Geometrically

Constraints

Feasible region

Example in 3 dimensions



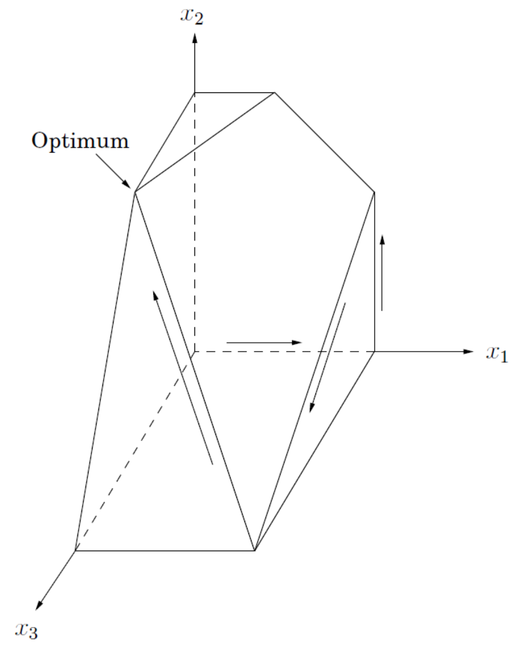
Linear Programming, Geometrically

How to visualize feasible regions in more than 3 dimensions?

Trick for n dimensional spaces for any $n \geq 4$

Step 1: imagine something in 3 dimensions

Step 2: in your mind, say n as loudly as possible



LP, General Formulation

Input: $c, a_1, a_2, \dots, a_m \in \mathbb{R}^n, b \in \mathbb{R}^m$

Goal:

$$
\begin{aligned}
\text{Maximize } & c^T x \\
\text{Subject to } & a_1^T x \le b_1 \\
& a_2^T x \le b_2 \\
& \vdots \\
& a_m^T x \le b_m \\
& x \ge 0
\end{aligned}
$$

n variables

m constraints

n more constraints

In general, we allow equality constraints, and inequalities in the other direction, and minimization objective. We will later see how to incorporate such changes

LP, Standard Matrix Form

Input: $c \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$

Goal:

Maximize $c^T x$

Subject to $Ax \leq b$

$x \geq 0$

n variables

m constraints

n more constraints

$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$, row i is a_i

Does LP Always Have an Optimal Solution?

NO! LP could fail to have a solution for two reasons:

(1) Linear program is *infeasible*, i.e., feasibility region is empty:

$$\{x \mid Ax \leq b\} = \emptyset$$

Example: constraints include $x_1 \leq 1$ and $x_1 \geq 2$.

(2) Linear program is *unbounded*, i.e., not constrained enough.

Example: maximize $x_1 + x_2$ subject to $x_1, x_2 \geq 0$.

When LP has an optimal solution, it also has one at a vertex of the feasible polytope!

You Have Seen LPs Before

Maximum flow

Input: directed graph $G = (V, E)$, $c: E \to \mathbb{R}_{\geq 0}$ edge capacities, s – start vertex, t – terminal

Output: valid flow f of maximum value

For each edge (u, v) introduce a variable f_{uv}

Flow value is $\sum_{(s,v)\in E} f_{sv}$

Flow is valid if it satisfies:

- Capacity constraints: for every edge (u, v) we have $0 \leq f_{uv} \leq c(u, v)$
- Flow conservation constraints: for every vertex v we have $\sum_{(u,v)\in E} f_{uv} = \sum_{(v,w)\in E} f_{v,w}$

Max Flow as LP

Flow value is $\sum_{(s,v)\in E} f_{sv}$

Linear objective!

Flow is valid if it satisfies:

- Capacity constraints: for every edge (u, v) we have $0 \le f_{uv} \le c(u, v)$
- Flow conservation constraints: for every vertex v we have $\sum_{(u,v)\in E} f_{uv} = \sum_{(v,w)\in E} f_{v,w}$

Linear constraints: inequalities/equalities

maximize $\displaystyle\sum_{(s,v)\in E} f_{sv}$

$0 \le f_{uv} \le c(u, v)$ for all $(u, v) \in E$

$\displaystyle\sum_{(u,v)\in E} f_{uv} = \sum_{(v,w)\in E} f_{v,w}$ for all $v \in V - \{s, t\}$

Single-source Shortest Path as LP

Input: directed graph $G = (V, E)$, $w: E \to \mathbb{R}_{\geq 0}$, s – start vertex, t – terminal vertex

Output: weight of a shortest-weight path from s to t

Variables: for each vertex v we have variable d_v

Exercise: prove formally that this works!

Why max?

$$
\begin{aligned}
&\text{maximize} \quad d_t \\
&\text{subject to} \\
&\qquad d_v \leq d_u + w(u, v) \quad \text{for each edge } (u, v) \in E\,, \\
&\qquad d_s = 0\,.
\end{aligned}
$$

If objective was min., then we could set all variables d_v to 0.

Yet Another LP Problem

For max flow and single-source shortest path specialized algorithms outperform LP-based algorithms

LP would not be so useful if we could always create specialized algorithms for all problems

It seems we can’t always do that, e.g.

Multicommodity-flow problem

Input: directed graph G=(V,E) $c: E \rightarrow \mathbb{R}_{\geq 0}$ edge capacities

k commodities $K_1, K_2, \dots, K_k$, where $K_i = (s_i, t_i, d_i)$ and s_i is the start vertex of commodity i, t_i is the terminal vertex, d_i is the demand.

Output: valid multicommodity flow $(f_1, f_2, \dots, f_k)$, where f_i has value d_i and all the f_i jointly satisfy the constraints

Multicommodity-flow Problem

Input: directed graph G=(V,E) $c: E \rightarrow \mathbb{R}_{\geq 0}$ edge capacities

k commodities $K_1, K_2, \ldots, K_k$, where $K_i = (s_i, t_i, d_i)$ and s_i is the start vertex of

commodity i, t_i is the terminal vertex, d_i is the demand.

Output: valid multicommodity flow $(f_1, f_2, \ldots, f_k)$, where f_i has value d_i and all the f_i jointly satisfy the constraints

$$
\begin{array}{rcll}
\sum_{i=1}^{k} f_{iuv} & \leq & c(u,v) & \text{for each } u, v \in V, \\
\sum_{v \in V} f_{iuv} - \sum_{v \in V} f_{ivu} & = & 0 & \text{for each } i = 1, 2, \ldots, k \text{ and} \\
 & & & \text{for each } u \in V - \{s_i, t_i\}, \\
\sum_{v \in V} f_{i,s_i,v} - \sum_{v \in V} f_{i,v,s_i} & = & d_i & \text{for each } i = 1, 2, \ldots, k, \\
f_{iuv} & \geq & 0 & \text{for each } u, v \in V \text{ and} \\
 & & & \text{for each } i = 1, 2, \ldots, k.
\end{array}
$$

The only known polynomial time algorithm for this problem is based on solving this LP! No specialized algorithms known.

Linear Programming is Everywhere

Used heavily in

- Microeconomics
- Manufacturing
- VLSI (very large scale integration) design
- Logistics/transportation
- Portfolio optimization
- Bioengineering (flux balance analysis)
- Company management more broadly: often want to maximize profits or minimize costs, use linear models for simplicity
- Operations research
- Design of approximation algorithms
- Proving theorems, as a proof technique
- …

Complexity of LP

Input: $c \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$

$$\begin{aligned} \text{Maximize } & c^T x \\ \text{Subject to } & Ax \leq b \\ & x \geq 0 \end{aligned}$$

Is the above easy to solve in polynomial time? Is it NP-hard? Is it easy in practice?

Complexity of LP

Fascinating and counter-intuitive story

1947 – Dantzig invents simplex algorithm. Simplex runs incredibly fast in practice (linear or near-linear time)

1973 – Klee and Minty give an example on which simplex runs in exponential time

1979 – Khachian invents ellipsoid method – the first polynomial time algorithm for LP. It does not give an exact solution, but for any $\epsilon > 0$ it gives an ϵ-approximation in poly time. Khachian's algorithm is not very fast in practice.

1984 – Karmarkar invents interior point methods – new poly time algorithm for LP. Various versions of interior point methods are sometimes used in practice.

2004 – Spielman and Teng introduce "smoothed analysis" to explain great empirical performance of simplex

Complexity of LP

Example when worst-case analysis fails miserably

Led to development of new great algorithms and ideas

Bottom line: linear programming is easy in theory and practice!

LP Solutions

Input: $c \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$

Goal:

$$\begin{aligned} \text{Maximize } & c^T x \\ \text{Subject to } & Ax \leq b \\ & x \geq 0 \end{aligned}$$

Note: optimal solution x might still be rational, even if c, b, A are integral

LP Fractional Solutions Example

$$
\begin{array}{rcrcrcl}
 & & \max & & z & & \\
-3x_1 & + & 2x_2 & + & z & \leq & 0 \\
x_1 & - & x_2 & + & z & \leq & 0 \\
x_1 & + & x_2 & & & = & 1 \\
 & & & & x_1, x_2 & \geq & 0
\end{array}
$$

Optimal solution: $z = \frac{1}{7}, x_1 = \frac{3}{7}, x_2 = \frac{4}{7}$

Integer Programming, IP

If we restrict solution to be integral, then we obtain an instance of an Integer Program

Input: $c \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$

Goal:

$$\text{Maximize } c^T x$$

$$\text{Subject to } Ax \leq b$$

$$\boldsymbol{x} \in \mathbb{Z}^{\boldsymbol{n}}$$

Does this make the problem harder or easier?

Integer Programming

The problem is intuitively harder than LP: feasible region for LP is a nice single continuous object, while feasible region for IP is a potentially huge collection of discrete points. Discrete objects tend to be harder to handle than continuous ones.

How hard is it?

NP-hard!

Consider **0/1 feasibility problem (special case of IP feasibility problem)**

Input: $b \in \mathbb{Z}^m, A \in \mathbb{Z}^{m \times n}$

Question: does there exist $x \in \{0,1\}^n$ such that $Ax \leq b$?

0/1 feasibility is hard IMPLIES IP feasibility is hard IMPLIES IP is hard

0/1 Feasibility Problem is NP-complete

Input: $b \in \mathbb{Z}^m, A \in \mathbb{Z}^{m \times n}$

Question: does there exist $x \in \{0,1\}^n$ such that $Ax \leq b$?

Step 1: IP feasibility is in NP. Given a solution x simply multiply it by A and compare with b. Matrix multiplication is in P, so it gives a polynomial time verifier.

Step 2: we will show how to reduce 3SAT to IP Feasibility Problem in polynomial time.

Step 2: 3SAT $\leq_p$ 0/1 Feasibility Problem

Given 3CNF formula φ, need to construct b, A such that

- Construction runs in polynomial time
- φ is satisfiable if and only if there exist $x \in \{0,1\}^n$ such that $Ax \leq b$

Suppose φ is defined on n variables $x_1, \ldots, x_n$. It has the form

$$C_1 \vee C_2 \vee \cdots \vee C_m,$$

where C_i is a clause consisting of 3 literals.

Step 2: 3SAT $\leq_p$ 0/1 Feasibility Problem

Suppose φ is defined on n variables $x_1, \dots, x_n$. It has the form

$$C_1 \vee C_2 \vee \cdots \vee C_m,$$

where C_i is a clause consisting of 3 literals.

Convert each clause into an inequality as follows:

- Positive literal x_i turns into x_i
- Negative literal $\neg x_i$ turns into $(1 - x_i)$
- Connective $\vee$ turns into $+$
- The inequality is ≥ 1

Example: $C_1 = x_1 \vee \neg x_{17} \vee x_{32}$ turns into $x_1 + (1 - x_{17}) + x_{32} \geq 1$

Step 2: 3SAT $\leq_p$ 0/1 Feasibility Problem

Example: $C = x_1 \vee \neg x_{17} \vee x_{32}$ turns into $x_1 + (1 - x_{17}) + x_{32} \geq 1$

Perform this conversion for each clause. You end up with a system of m inequalities over n variables. Can be expressed as $Ax \leq b$.

(1) conversion clearly takes polynomial time

(2) φ is satisfiable if and only if there exist $x \in \{0,1\}^n$ such that $Ax \leq b$, because

1 corresponds to T, 0 corresponds to F

each inequality is satisfied if and only if the corresponding clause is satisfied

Conclusion

0/1 Feasibility Problem is NP-complete → IP Feasibility Problem is NP-hard → Integer Programming is NP-hard

BUT

Linear Programming is easy (in P)

Adding the restriction that solution is integral tremendously increases the complexity

Side notes

IP feasibility (when variables can be any integers, not necessarily 0/1) is, in fact, NP-complete. We have shown it is NP-hard, so to show it is NP-complete, we need to show it is in NP. This is nontrivial, but follows from the known linear algebraic techniques (essentially, Cramer's rule).

IP feasibility reduces to 0/1 feasibility in polytime (exercise!)

Integer programming is self-reducible: if decision problem is in P then the search problem is also in P. (exercise! hint: easily follows from the previous point)

Back to Linear Programming

2 popular forms of LP

Standard form:

Maximize $c^T x$
Subject to $Ax \leq b$
$x \geq 0$

Introduce new vars s →

Slack form:

$$z = c^T x$$
$$s = b - Ax$$
$$s, x \geq 0$$

What if Your LP is not in Standard Form?

Could happen for several reasons:

Standard form:
Maximize $c^T x$
Subject to $Ax \leq b$
$x \geq 0$

(1) your problem is minimization instead of maximization

(2) your constraints contain equalities

(3) your constraints contain inequalities $\geq$ instead of $\leq$

(4) your variable x_i does not have a corresponding constraint $x_i \geq 0$

Transformations that “Preserve Solutions”

Transform your LP formulation L into another LP formulation L’ such that a solution to L’ can be efficiently turned into a solution to L

(1) To turn minimization problem into maximization, multiply objective by -1

(2) Replace each equality constraint $a^T x = b$ by two inequalities: $a^T x \leq b$ and $a^T x \geq b$

(3) Multiply an inequality of the form $a^T x \geq b$ by -1 to obtain $-a^T x \leq -b$

(4) For each unconstrained variable x_i introduce two new variables x_i^+ and x_i^-

Replace each occurrence of x_i with $x_i^+ - x_i^-$

Introduce two inequalities $x_i^+ \geq 0$ and $x_i^- \geq 0$

LP Transformations: Example

$$\begin{array}{lrcrcl}
\text{minimize} & -2x_1 & + & 3x_2 & & \\
\text{subject to} & & & & & \\
& x_1 & + & x_2 & = & 7 \\
& x_1 & - & 2x_2 & \leq & 4 \\
& x_1 & & & \geq & 0\ ,
\end{array}$$

$$\begin{array}{lrcrcl}
\text{maximize} & 2x_1 & - & 3x_2 & & \\
\text{subject to} & & & & & \\
& x_1 & + & x_2 & = & 7 \\
& x_1 & - & 2x_2 & \leq & 4 \\
& x_1 & & & \geq & 0\ .
\end{array}$$

$$\begin{array}{lrcrcrcl}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & = & 7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0\ .
\end{array}$$

LP Transformations: Example Cont'd

$$\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & = & 7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0\ .
\end{array}$$

$$\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2' & + & 3x_2'' & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2' & - & x_2'' & \leq & 7 \\
& x_1 & + & x_2' & - & x_2'' & \geq & 7 \\
& x_1 & - & 2x_2' & + & 2x_2'' & \leq & 4 \\
& & & x_1, x_2', x_2'' & & & \geq & 0\ .
\end{array}$$

$$\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2 & + & 3x_3 & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2 & - & x_3 & \leq & 7 \\
& -x_1 & - & x_2 & + & x_3 & \leq & -7 \\
& x_1 & - & 2x_2 & + & 2x_3 & \leq & 4 \\
& & & x_1, x_2, x_3 & & & \geq & 0\ .
\end{array}$$

Standard Form!

How to Make Sure LP Solution is Optimal

$$\begin{aligned} \max \ x_1 + 6x_2 \\ x_1 \leq 200 \\ x_2 \leq 300 \\ x_1 + x_2 \leq 400 \\ x_1, x_2 \geq 0 \end{aligned}$$

Suppose I say that $(x_1, x_2) = (100{,}300)$ is optimal with objective value 1900

How can you check this?

How to Make Sure LP Solution is Optimal

$$\begin{aligned} \max \ x_1 + 6x_2 & \\ x_1 &\leq 200 \\ x_2 &\leq 300 \\ x_1 + x_2 &\leq 400 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Suppose I say that $(x_1, x_2) = (100,300)$ is optimal with objective value 1900

Take the first constraint and add to it 6 times the second constraint to get

$$x_1 + 6x_2 \leq 2000$$

This shows that **ANY SOLUTION AT ALL** can achieve value at most 2000

How to Make Sure LP Solution is Optimal

$$
\begin{aligned}
\max\ \ x_1 + 6x_2 & \\
x_1 &\le 200 \\
x_2 &\le 300 \\
x_1 + x_2 &\le 400 \\
x_1, x_2 &\ge 0
\end{aligned}
$$

Suppose I say that $(x_1, x_2) = (100{,}300)$ is optimal with objective value 1900

Can we add some other combination of constraints to get this bound even closer to 1900?
Try 5 times the second constraint plus the third constraint to get

$$5x_2 + (x_1 + x_2) = x_1 + 6x_2 \le 5 \times 300 + 400 = 1900$$

This shows that **ANY SOLUTION AT ALL** can achieve value at most 1900
Therefore, the above solution that achieves 1900 is optimal!

Is There an Algorithm to Verify if Solution is Optimal?

Introduce variables y_1, y_2, y_3 to denote multipliers of the constraints

Multiplier	Inequality
y_1	$x_1 \leq 200$
y_2	$x_2 \leq 300$
y_3	$x_1 + x_2 \leq 400$

What do we want from these multipliers?

(1) $y_i \geq 0$ otherwise if the multiplier is negative multiplying by it flips the inequality

After multiplication and addition we get the inequality:

$$(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \leq 200y_1 + 300y_2 + 400y_3.$$

(2) want the LHS to look like the objective $x_1 + 6x_2$, but in fact it is also enough to bound the objective, i.e., we want $x_1 + 6x_2 \leq (y_1 + y_3)x_1 + (y_2 + y_3)x_2$

Is There an Algorithm to Verify if Solution is Optimal?

$$\max \; x_1 + 6x_2$$
$$x_1 \leq 200$$
$$x_2 \leq 300$$
$$x_1 + x_2 \leq 400$$
$$x_1, x_2 \geq 0$$

Multiplier	Inequality
y_1	$x_1 \leq 200$
y_2	$x_2 \leq 300$
y_3	$x_1 + x_2 \leq 400$

$$(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \leq 200y_1 + 300y_2 + 400y_3.$$

What do we want from these multipliers?

(1) $y_i \geq 0$

(2)

$$x_1 + 6x_2 \leq 200y_1 + 300y_2 + 400y_3 \quad \text{if} \quad \left\{ \begin{array}{c} y_1, y_2, y_3 \geq 0 \\ y_1 + y_3 \geq 1 \\ y_2 + y_3 \geq 6 \end{array} \right\}.$$

(3) minimize the bound $200y_1 + 300y_2 + 400y_3$

Is There an Algorithm to Verify if Solution is Optimal?

$$
\begin{aligned}
\max\ x_1 + 6x_2 \\
x_1 \le 200 \\
x_2 \le 300 \\
x_1 + x_2 \le 400 \\
x_1, x_2 \ge 0
\end{aligned}
$$

Multiplier	Inequality
y_1	$x_1 \le 200$
y_2	$x_2 \le 300$
y_3	$x_1 + x_2 \le 400$

What do we want from these multipliers?

$$
\begin{aligned}
\min\ 200y_1 + 300y_2 + 400y_3 \\
y_1 + y_3 \ge 1 \\
y_2 + y_3 \ge 6 \\
y_1, y_2, y_3 \ge 0
\end{aligned}
$$

That's another LP – called the DUAL! Original LP is called the PRIMAL.

Is There an Algorithm to Verify if Solution is Optimal?

PRIMAL

$$
\begin{aligned}
\max\ x_1 + 6x_2 \\
x_1 \le 200 \\
x_2 \le 300 \\
x_1 + x_2 \le 400 \\
x_1, x_2 \ge 0
\end{aligned}
$$

DUAL

$$
\begin{aligned}
\min\ 200y_1 + 300y_2 + 400y_3 \\
y_1 + y_3 \ge 1 \\
y_2 + y_3 \ge 6 \\
y_1, y_2, y_3 \ge 0
\end{aligned}
$$

The problem of certifying optimality of an LP is LP itself

If the dual LP has solution y_1, y_2, y_3 that gives the same value as the solution x_1, x_2, x_3 to the primal, then you know that your primal solution was in fact OPTIMAL

Another View of Optimality Certificate

Suppose you find a new super fast LP solver and build a company around this knowledge

You provide a service to customers by solving their huge LPs that they can’t solve themselves

You want your algorithm to remain a secret, but the customers demand to know that your LP solver is producing optimal solutions

How do they check your solutions? What do you need to do in order to convince customers?

Formulate the dual, solve it – it is an LP after all and you have a super fast LP solver

Send your customers the dual solution alongside with the primal solution

Customers can check if using the multipliers for the dual solutions gives the same bound as the primal solution (it requires just adding together linear inequalities, which can be done even by a resource-bounded customers). Moreover, you haven’t revealed anything about the algorithm!

Does this always work?

Does the Dual Optimum Always Coincide with the Primal Optimum?

General version of the dual. Note: **use standard form!**

Primal LP:

$$\max\ \mathbf{c}^T\mathbf{x}$$
$$\mathbf{A}\mathbf{x} \leq \mathbf{b}$$
$$\mathbf{x} \geq 0$$

Dual LP:

$$\min\ \mathbf{y}^T\mathbf{b}$$
$$\mathbf{y}^T\mathbf{A} \geq \mathbf{c}^T$$
$$\mathbf{y} \geq 0$$

Let me stress it again: to write down dual LP, first write down primal LP in standard form, then use the above formula. Otherwise it will get confusing!

Weak Duality Theorem: If x is a feasible solution to the primal and y is a feasible solution to the dual, then the value of solution $x \ \leq$ the value of solution y.

Weak Duality

Primal LP:

$$\max\ \mathbf{c}^T\mathbf{x}$$
$$\mathbf{A}\mathbf{x} \leq \mathbf{b}$$
$$\mathbf{x} \geq 0$$

Dual LP:

$$\min\ \mathbf{y}^T\mathbf{b}$$
$$\mathbf{y}^T\mathbf{A} \geq \mathbf{c}^T$$
$$\mathbf{y} \geq 0$$

Weak Duality Theorem: If x is a feasible solution to the primal and y is a feasible solution to the dual, then the value of solution $x \ \leq$ the value of solution y.

Proof: the value of solution x is $c^T x$ and the value of solution y is $y^T b$. We have

$$c^T x \leq (y^T A)x = y^T(Ax) \leq y^T b,$$

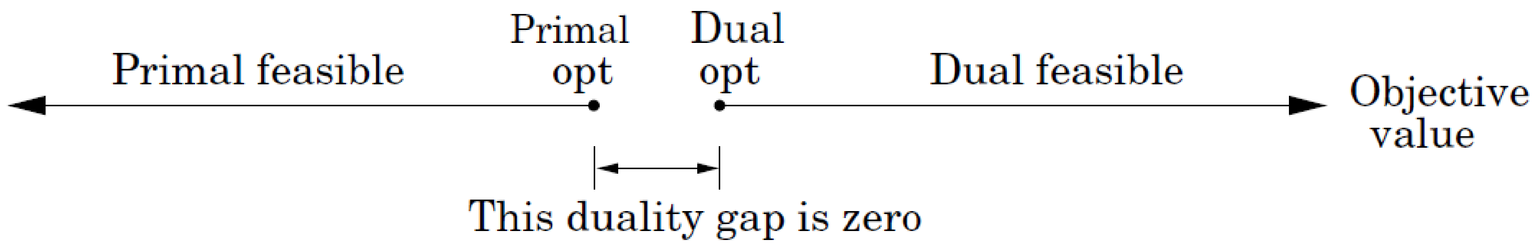
the first inequality is from the definition of the dual, the second inequality is from the definition of the primal.

Does the Dual Optimum Always Coincide with the Primal Optimum?

Weak duality shows that the primal optimum is always bounded by the dual optimum

Strong duality shows that the optimums actually coincide!

Strong duality theorem: if the primal LP has a bounded optimum, then so does the dual LP, and the two optimal values coincide.



One of the most important theorems in the theory of linear programming!

Strong Duality

Strong duality theorem: if the primal LP has a bounded optimum, then so does the dual LP, and the two optimal values coincide.

To prove the theorem, we will use the following technical tool: **Farkas lemma**

Farkas lemma (one of many-many versions): Exactly one of the following holds:

(1) exists x such that $Ax \leq b$

(2) exists y such that $y^T A = 0, y \geq 0, y^T b \ < 0$

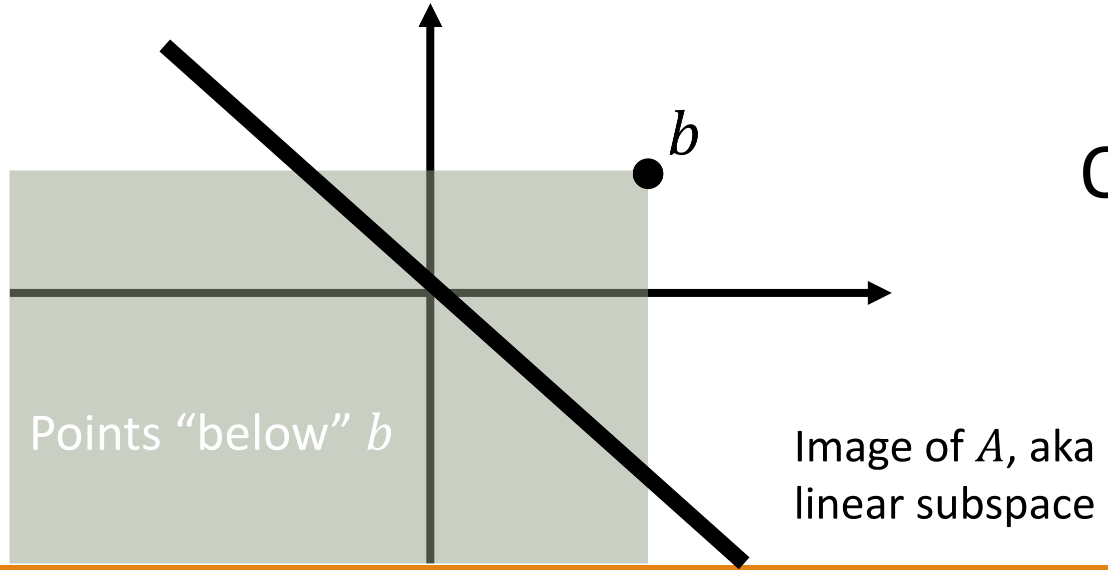
Farkas Lemma – Geometric Intuition

Farkas lemma (one of many-many versions): Exactly one of the following holds:

(1) exists x such that $Ax \leq b$

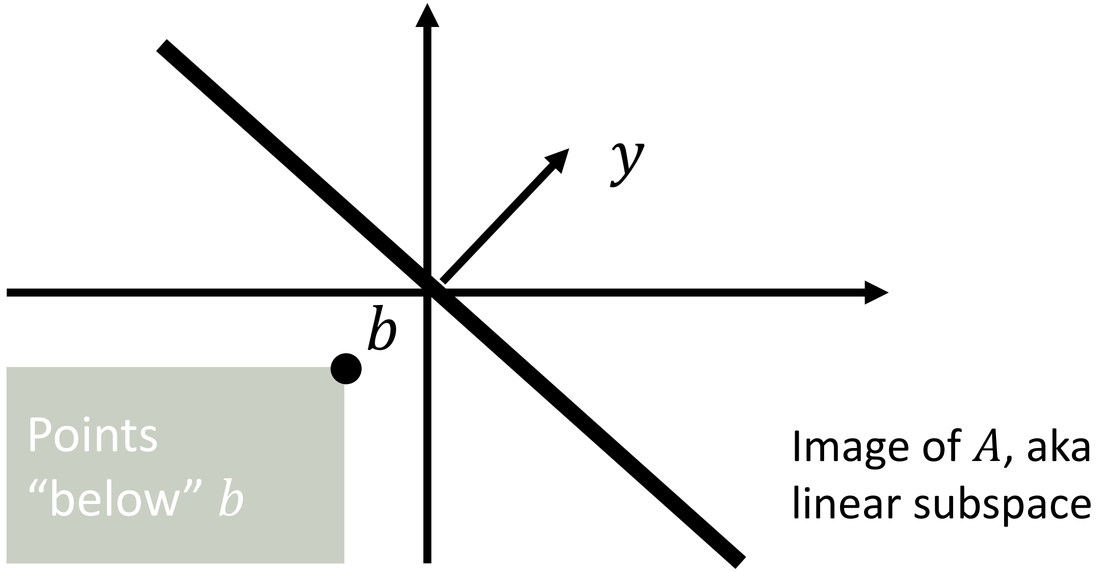
(2) exists y such that $y^T A = 0, y \geq 0, y^T b \ < 0$

(1) Image of A contains a point “below” b



OR

(2) The region “below” point b doesn’t intersect image of A
this is witnessed by a normal vector to the image of A



Strong Duality Proof

Primal LP:

$$\begin{aligned} &\max\ \mathbf{c}^T\mathbf{x} \\ &\mathbf{A}\mathbf{x} \le \mathbf{b} \\ &\mathbf{x} \ge 0 \end{aligned}$$

Dual LP:

$$\begin{aligned} &\min\ \mathbf{y}^T\mathbf{b} \\ &\mathbf{y}^T\mathbf{A} \ge \mathbf{c}^T \\ &\mathbf{y} \ge 0 \end{aligned}$$

Strong duality theorem (special case): Assume both primal and dual have finite optimal values. The two optimal values coincide.

Proof: Let x^* be an optimal primal solution, let $z^* = c^T x^*$ be the optimal value. By weak duality there is no y such that $y^T A \ge c^T$ and $y^T b \le z^*$, i.e., there is no y such that

$$\begin{pmatrix} -A^T \\ b^T \end{pmatrix} y \le \begin{pmatrix} c \\ z^* \end{pmatrix}$$

Strong Duality Proof

There is no y such that $\begin{pmatrix} -A^T \\ b^T \end{pmatrix} y \leq \begin{pmatrix} c \\ z^* \end{pmatrix}$

By Farkas lemma, there is x and λ such that

$$\begin{pmatrix} x^T & \lambda \end{pmatrix} \begin{pmatrix} -A^T \\ b^T \end{pmatrix} = 0, x \geq 0, \lambda \geq 0, -x^T c + \lambda z^* < 0$$

Case 1: $\lambda > 0$, then rescale (x, λ) by λ to get $(x/\lambda, 1)$. By the above we get $Ax = b$, $x \geq 0$ and $c^T x > z^*$, which contradicts the optimality of z^*.

Case 2: $\lambda = 0$, then we get $Ax = 0$ and $c^T x > 0$, so we can add x to our optimal solution x^* without contradicting inequalities and improving the value of the objective. Moreover, we can do it infinitely many times, contradicting the fact that our primal has a finite optimal value.

Simplex

Has excellent empirical performance

Has terrible worst-case performance

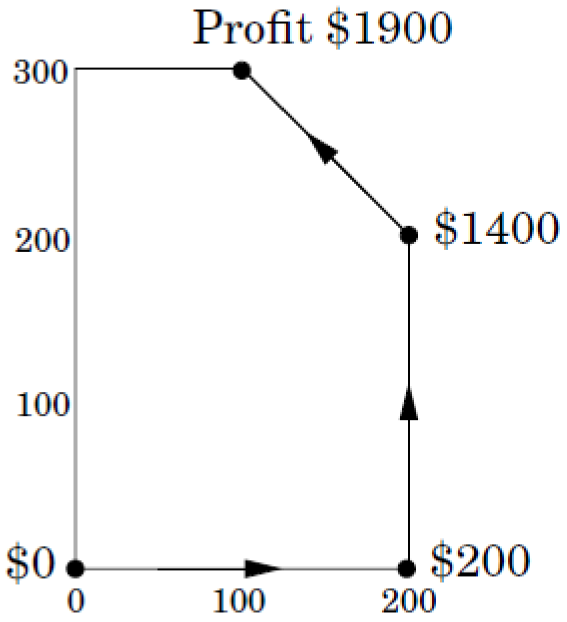
Easy to specify geometrically

```
let v be any vertex of the feasible region
while there is a neighbor v' of v with better objective value:
    set v = v'
```

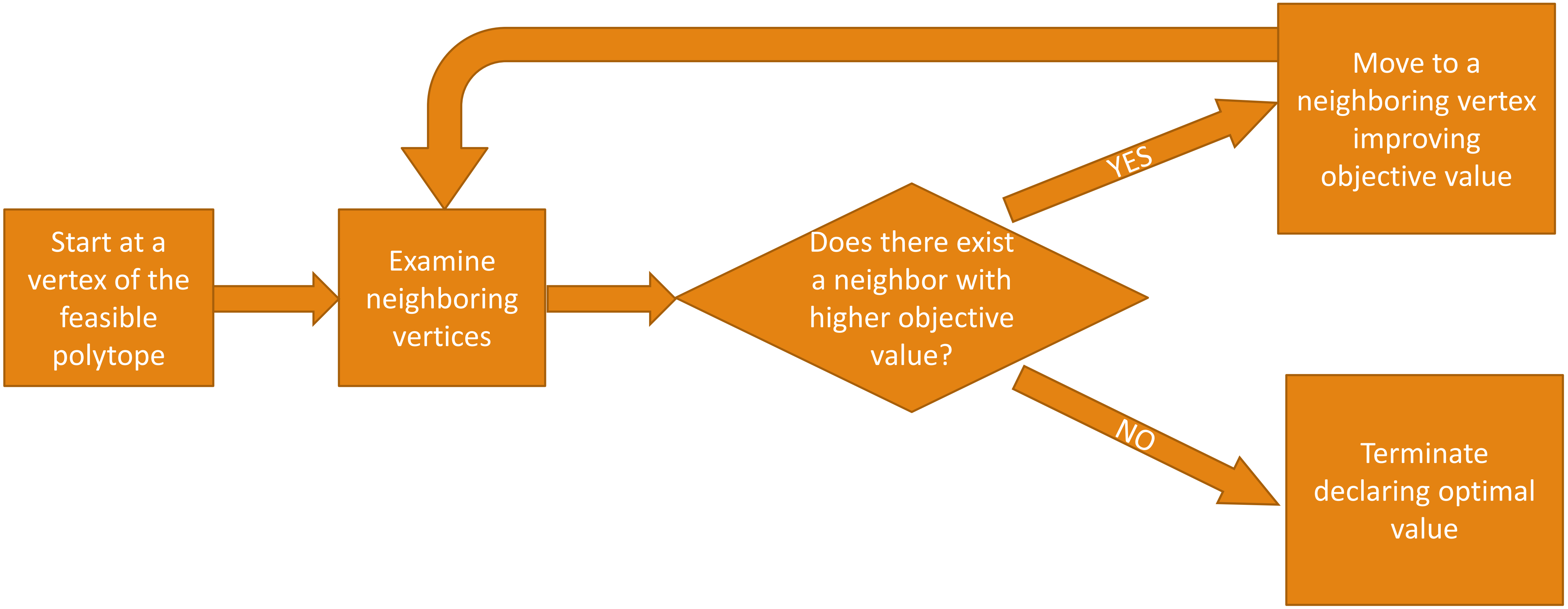
Simplex Geometric View

```
let v be any vertex of the feasible region
while there is a neighbor v' of v with better objective value:
    set v = v'
```

$$\max \; x_1 + 6x_2$$
$$x_1 \leq 200$$
$$x_2 \leq 300$$
$$x_1 + x_2 \leq 400$$
$$x_1, x_2 \geq 0$$



Simplex


Start at a vertex of the feasible polytope
Examine neighboring vertices
Does there exist a neighbor with higher objective value?
YES
Move to a neighboring vertex improving objective value
NO
Terminate declaring optimal value

Simplex: How to Actually Implement it?

Recall two forms of LP:

Standard form:

Maximize $c^T x$
Subject to $Ax \leq b$
$x \geq 0$

Slack form:

$\mathrm{z} = c^T x$
$s = b - Ax$
$s, x \geq 0$

All steps of Simplex can be conveniently performed in Slack form!

"A mathematical representation of surplus resources." In real life problems, it's unlikely that all resources will be used completely, so there usually are unused resources.

Slack variables represent the unused resources between the LHS and RHS of each constraint.

Slack Form

$$
\begin{array}{lrcrcrcr}
\text{maximize} & 2x_1 & - & 3x_2 & + & 3x_3 & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2 & - & x_3 & \leq & 7 \\
& -x_1 & - & x_2 & + & x_3 & \leq & -7 \\
& x_1 & - & 2x_2 & + & 2x_3 & \leq & 4 \\
& & & & x_1, x_2, x_3 & & \geq & 0 \ .
\end{array}
$$

$$
\begin{array}{ll}
\text{maximize} & \overbrace{2x_1 - 3x_2 + 3x_3}^{\text{Nonbasic Variables}} \\
\text{subject to} & \\
\text{Basic Variables} & \left\{
\begin{array}{rcrcrcrcr}
x_4 & = & 7 & - & x_1 & - & x_2 & + & x_3 \\
x_5 & = & -7 & + & x_1 & + & x_2 & - & x_3 \\
x_6 & = & 4 & - & x_1 & + & 2x_2 & - & 2x_3
\end{array}\right. \\
& x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \ .
\end{array}
$$

Slack Form: Convenient Notation

maximize $\quad 2x_1 - 3x_2 + 3x_3$

subject to

$$
\begin{aligned}
x_4 &= 7 - x_1 - x_2 + x_3 \\
x_5 &= -7 + x_1 + x_2 - x_3 \\
x_6 &= 4 - x_1 + 2x_2 - 2x_3 \\
x_1, x_2, x_3, x_4, x_5, x_6 &\geq 0 .
\end{aligned}
$$

$\Longleftrightarrow$

$$
\begin{aligned}
z &= 2x_1 - 3x_2 + 3x_3 \\
x_4 &= 7 - x_1 - x_2 + x_3 \\
x_5 &= -7 + x_1 + x_2 - x_3 \\
x_6 &= 4 - x_1 + 2x_2 - 2x_3 .
\end{aligned}
$$

Simplex: Starting at a Vertex

Standard form:

Maximize $c^T x$
Subject to $Ax \leq b$
$x \geq 0$

Slack form:

$\mathrm{z} = c^T x$
$s = b - Ax$
$s, x \geq 0$

Observe that if $b \geq 0$ then $x = 0$ (the all-0 vector) is feasible

Thus, if $b \geq 0$ we can start simplex at $x = 0$

We will assume that $b \geq 0$ for now. We will talk about how to drop this assumption later

In slack form it means: set nonbasic variables to 0

Simplex Example

$$
\begin{array}{llllllll}
\text{maximize} & 3x_1 & + & x_2 & + & 2x_3 & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2 & + & 3x_3 & \leq & 30 \\
& 2x_1 & + & 2x_2 & + & 5x_3 & \leq & 24 \\
& 4x_1 & + & x_2 & + & 2x_3 & \leq & 36 \\
& & x_1, x_2, x_3 & & & & \geq & 0 .
\end{array}
$$

$$
\begin{array}{rcrcrcrcr}
z & = & & & 3x_1 & + & x_2 & + & 2x_3 \\
x_4 & = & 30 & - & x_1 & - & x_2 & - & 3x_3 \\
x_5 & = & 24 & - & 2x_1 & - & 2x_2 & - & 5x_3 \\
x_6 & = & 36 & - & 4x_1 & - & x_2 & - & 2x_3
\end{array}
$$

To increase the value of z:

(1) Find a nonbasic variable with a positive coefficient, e.g., x_1 (called **entering variable**)

(2) See how much you can increase this nonbasic variable without violating constraints

Simplex Example

Try to increase!

$$
\begin{aligned}
z &= && 3x_1 &&+ x_2 &&+ 2x_3 \\
x_4 &= 30 &&- x_1 &&- x_2 &&- 3x_3 \\
x_5 &= 24 &&- 2x_1 &&- 2x_2 &&- 5x_3 \\
x_6 &= 36 &&- 4x_1 &&- x_2 &&- 2x_3
\end{aligned}
$$

Obstacles!

$x_1 \leq 30$

$x_1 \leq 24/2 = 12$

$x_1 \leq 36/4 = 9$

Otherwise basic variable becomes negative

Tightest obstacle!

Simplex Example

$$
\begin{aligned}
z &= & 3x_1 &+ x_2 + 2x_3 \\
x_4 &= 30 &- x_1 &- x_2 - 3x_3 \\
x_5 &= 24 &- 2x_1 &- 2x_2 - 5x_3 \\
x_6 &= 36 &- 4x_1 &- x_2 - 2x_3
\end{aligned}
$$

$x_1 \leq 36/4 = 9$

Tightest obstacle!

Solve tightest obstacle for the nonbasic variable

$$x_1 = 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4}$$

Substitute x_1 in all other questions (called **pivot**)

This turns x_1 into a basic variable and x_6 into a non-basic variable

x_6 is called **leaving variable**

Simplex Example

$$
\begin{aligned}
z &= & & 3x_1 &+& x_2 &+& 2x_3 \\
x_4 &= 30 &-& x_1 &-& x_2 &-& 3x_3 \\
x_5 &= 24 &-& 2x_1 &-& 2x_2 &-& 5x_3 \\
x_6 &= 36 &-& 4x_1 &-& x_2 &-& 2x_3
\end{aligned}
\quad\Longrightarrow\quad
\begin{aligned}
z &= 27 + \frac{x_2}{4} + \frac{x_3}{2} - \frac{3x_6}{4} \\
x_1 &= 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4} \\
x_4 &= 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4} \\
x_5 &= 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2} .
\end{aligned}
$$

Note: after this step **basic feasible solution**, i.e., substituting 0 for nonbasic variables improves the value of z from 0 to 27.

What next? Rinse and repeat!

(1) Find a nonbasic variable with a positive coefficient in the objective (**entering variable**)
(2) Find the tightest obstacle (**leaving variable**)
(3) Solve for the entering variable using the tightest obstacle and update the LP (**pivot**)

Simplex Example

Entering variable
Try to increase!

$$
\begin{aligned}
z &= 27 + \frac{x_2}{4} + \frac{x_3}{2} - \frac{3x_6}{4} \\
x_1 &= 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4} \\
x_4 &= 21 - \frac{3x_2}{4} - \frac{5x_3}{2} + \frac{x_6}{4} \\
x_5 &= 6 - \frac{3x_2}{2} - 4x_3 + \frac{x_6}{2} .
\end{aligned}
$$

Leaving variable
Tightest obstacle!

Pivot!

$$
\begin{aligned}
z &= \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - \frac{11x_6}{16} \\
x_1 &= \frac{33}{4} - \frac{x_2}{16} + \frac{x_5}{8} - \frac{5x_6}{16} \\
x_3 &= \frac{3}{2} - \frac{3x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8} \\
x_4 &= \frac{69}{4} + \frac{3x_2}{16} + \frac{5x_5}{8} - \frac{x_6}{16} .
\end{aligned}
$$

Simplex Example

Entering variable
Try to increase!

$$\begin{aligned}
z &= \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - \frac{11x_6}{16} \\
x_1 &= \frac{33}{4} - \frac{x_2}{16} + \frac{x_5}{8} - \frac{5x_6}{16} \\
x_3 &= \frac{3}{2} - \frac{3x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8} \\
x_4 &= \frac{69}{4} + \frac{3x_2}{16} + \frac{5x_5}{8} - \frac{x_6}{16} .
\end{aligned}$$

Pivot!

$$\begin{aligned}
z &= 28 - \frac{x_3}{6} - \frac{x_5}{6} - \frac{2x_6}{3} \\
x_1 &= 8 + \frac{x_3}{6} + \frac{x_5}{6} - \frac{x_6}{3} \\
x_2 &= 4 - \frac{8x_3}{3} - \frac{2x_5}{3} + \frac{x_6}{3} \\
x_4 &= 18 - \frac{x_3}{2} + \frac{x_5}{2} .
\end{aligned}$$

Leaving variable
Tightest obstacle!

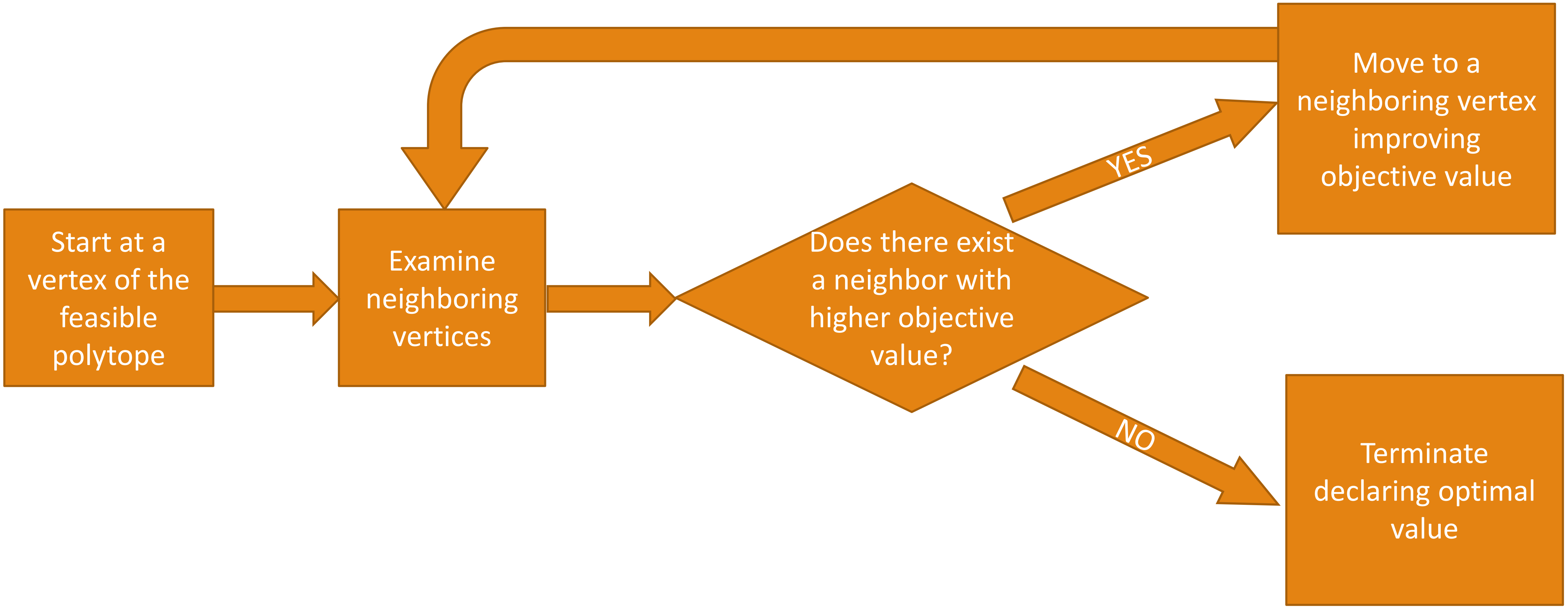
Simplex Example

$$
\begin{aligned}
z &= 28 - \frac{x_3}{6} - \frac{x_5}{6} - \frac{2x_6}{3} \\
x_1 &= 8 + \frac{x_3}{6} + \frac{x_5}{6} - \frac{x_6}{3} \\
x_2 &= 4 - \frac{8x_3}{3} - \frac{2x_5}{3} + \frac{x_6}{3} \\
x_4 &= 18 - \frac{x_3}{2} + \frac{x_5}{2} .
\end{aligned}
$$

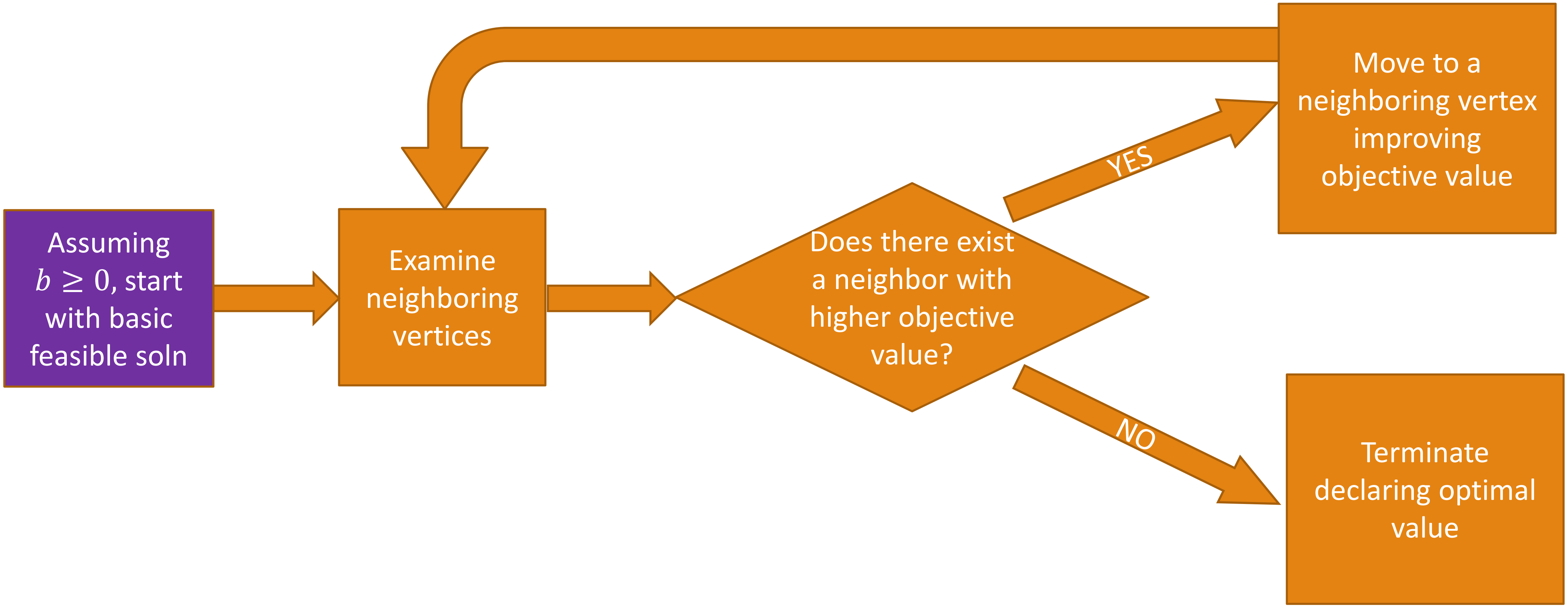
No leaving variable! What next?

We are done!

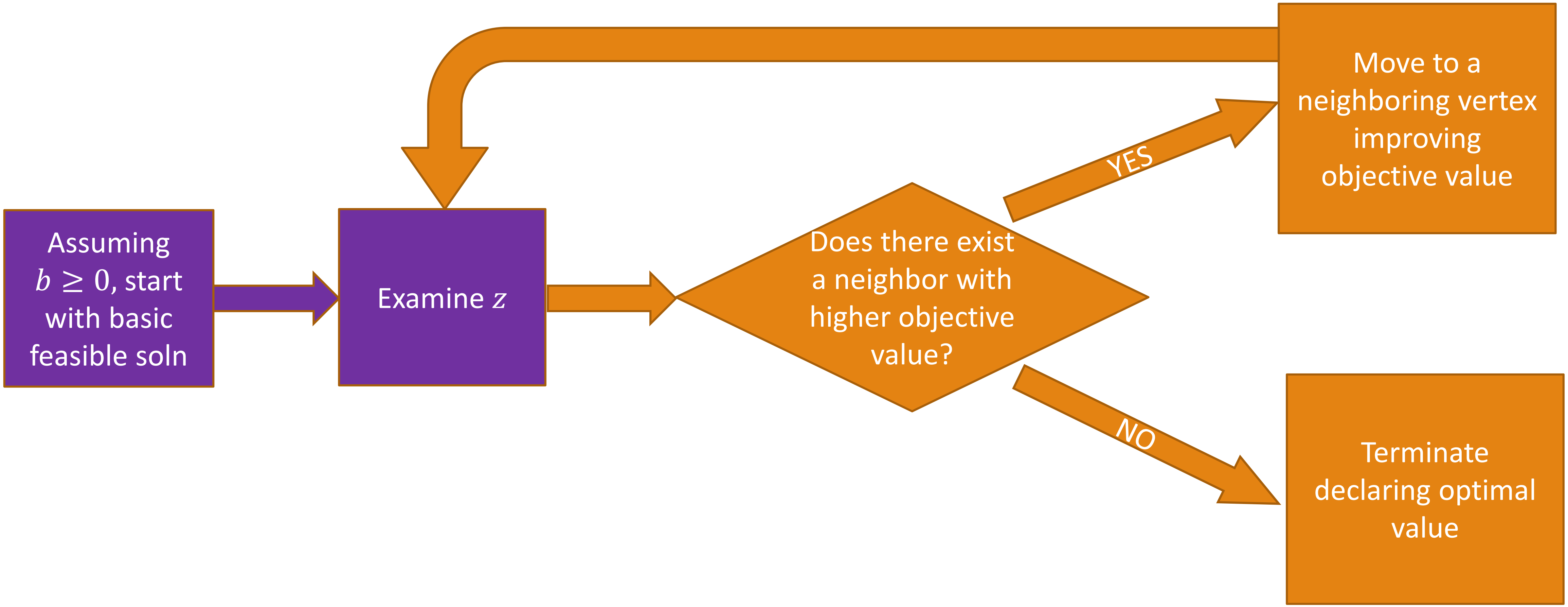
Compare with Geometric View



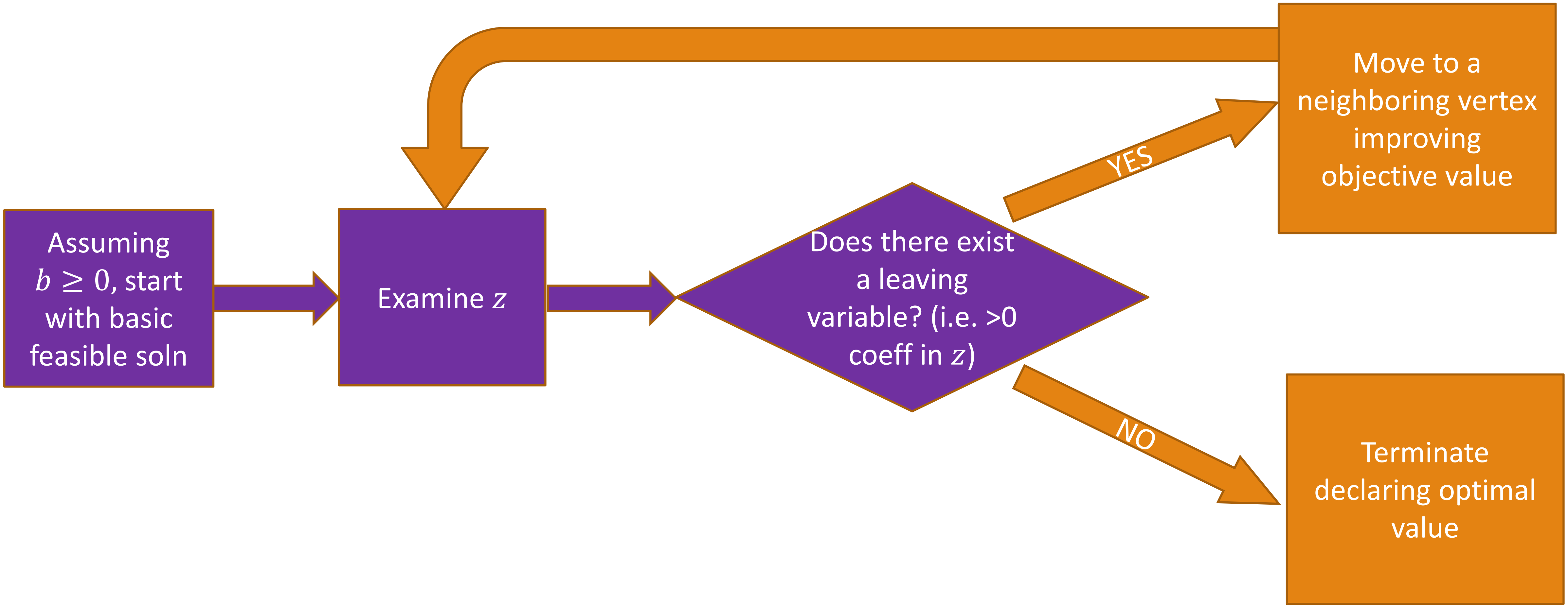
Compare with Geometric View



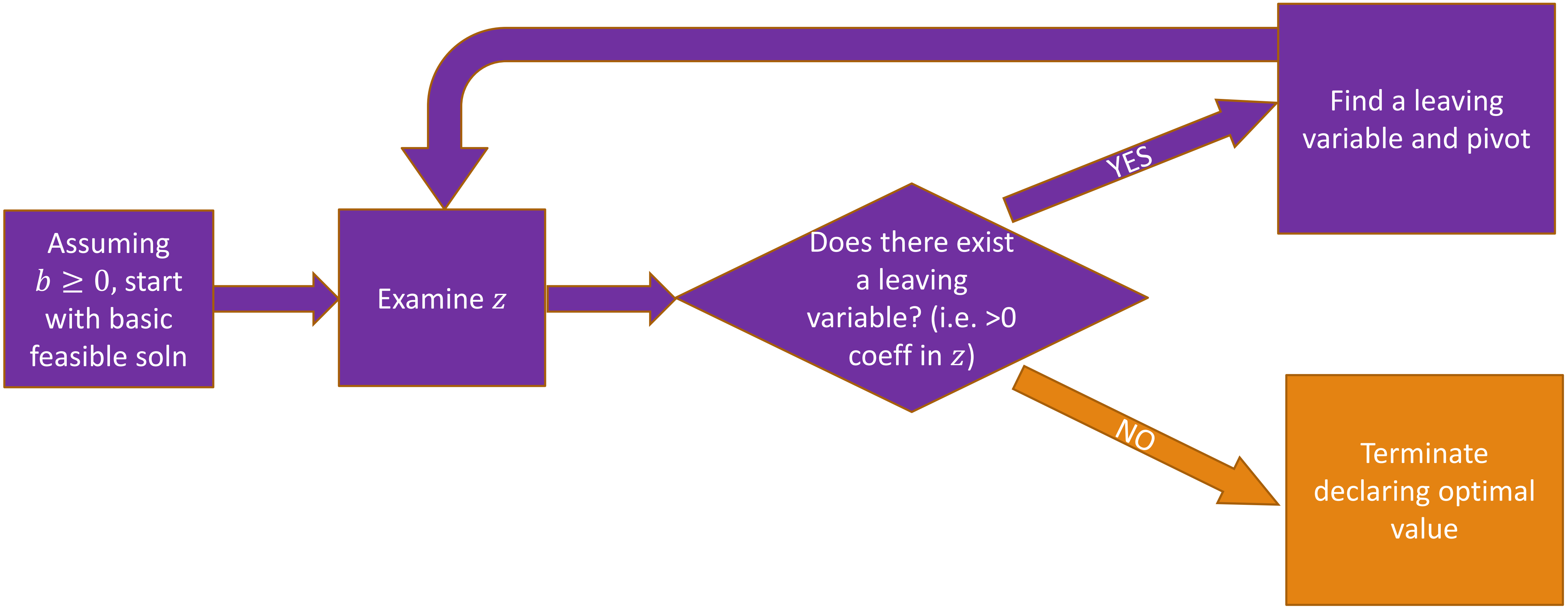
Compare with Geometric View



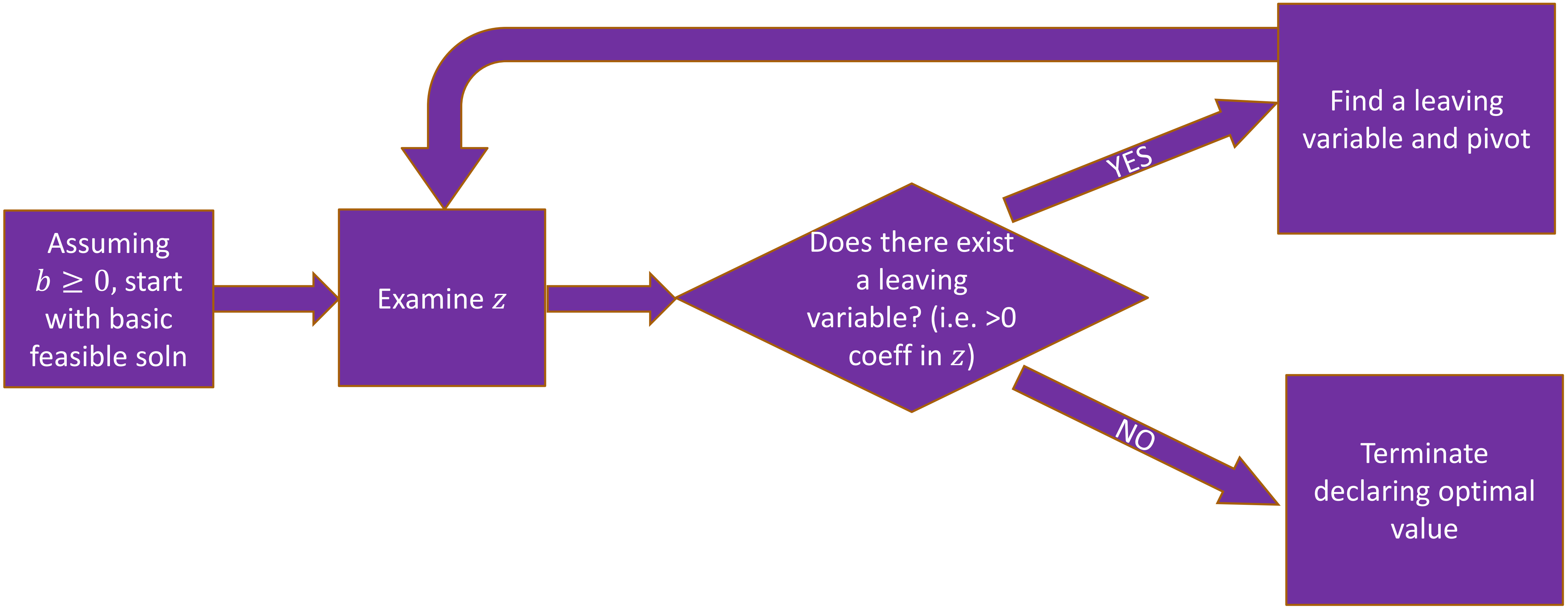
Compare with Geometric View



Compare with Geometric View



Compare with Geometric View



Simplex Outstanding Issues

What if entering variable is unconstrained, i.e., has no corresponding leaving variable?

Means that you can increase z as much as possible since entering variable has positive coefficient, declare that the LP is unbounded.

It is possible that pivoting leaves the value of objective unchanged. This is known as **degeneracy – can lead to cycling (infinite loop)**. One way to solve it is to perturb b by a small random amount in each coordinate. Another way is to break ties in choosing entering and leaving variables carefully, e.g., by smallest index (known as **Bland's rule**).

Simplex Outstanding Issues

What if initial basic solution is not feasible, i.e., it is not true that $b \geq 0$?

Then we create a new LP as follows:

- Create m new *artificial variables* $z_1, \ldots, z_m \geq 0$, where m is the number of equations.
- Add z_i to the left-hand side of the ith equation.
- Let the objective, to be *minimized*, be $z_1 + z_2 + \cdots + z_m$.

For this new LP, it's easy to come up with a starting vertex, namely, the one with $z_i = b_i$ for all i and all other variables zero. Therefore we can solve it by simplex, to obtain the optimum solution.

If optimum value is 0, then we can extract initial feasible solution for our LP.

Otherwise, LP is infeasible!

Simplex Outstanding Issue

Pseudocode? Proof of correctness? Analysis of runtime?

See textbook for details!

The End!