CSC373

Linear Programming

Illustration Courtesy:
Kevin Wayne & Denis Pankratov

Recap

- Network flow
  - Ford-Fulkerson algorithm
    - Ways to make the running time polynomial
  - Correctness using max-flow, min-cut
  - Applications:
    - Edge-disjoint paths
    - Multiple sources/sinks
    - Circulation
    - Circulation with lower bounds
    - Survey design
    - Image segmentation
    - Profit maximization

Brewery Example

- A brewery can invest its inventory of corn, hops and malt into producing some amount of ale and some amount of beer
  - Per unit resource requirement and profit of the two items are as given below

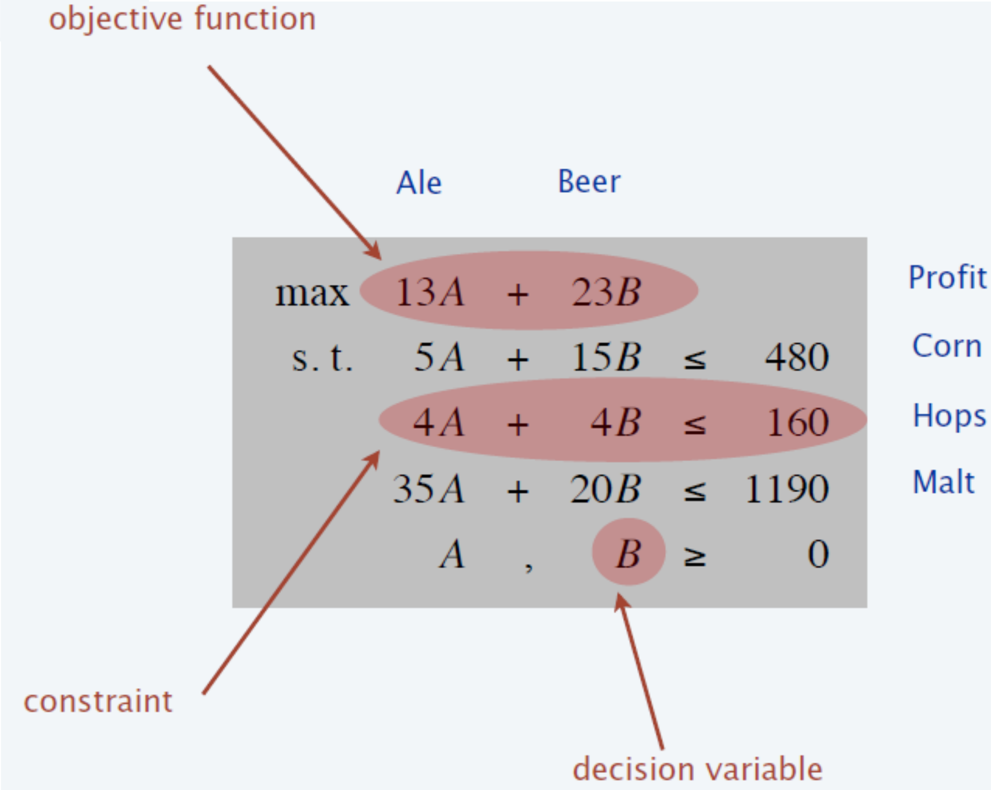
Beverage	Corn (pounds)	Hops (ounces)	Malt (pounds)	Profit ($)
Ale (barrel)	5	4	35	13
Beer (barrel)	15	4	20	23
constraint	480	160	1190	

Example Courtesy: Kevin Wayne

Brewery Example

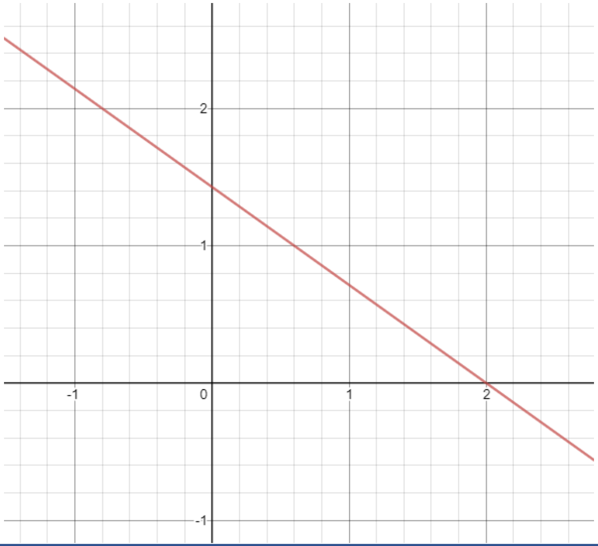
Beverage	Corn (pounds)	Hops (ounces)	Malt (pounds)	Profit ($)
Ale (barrel)	5	4	35	13
Beer (barrel)	15	4	20	23
constraint	480	160	1190	

- Suppose it produces A units of ale and B units of beer
- Then we want to solve this program:



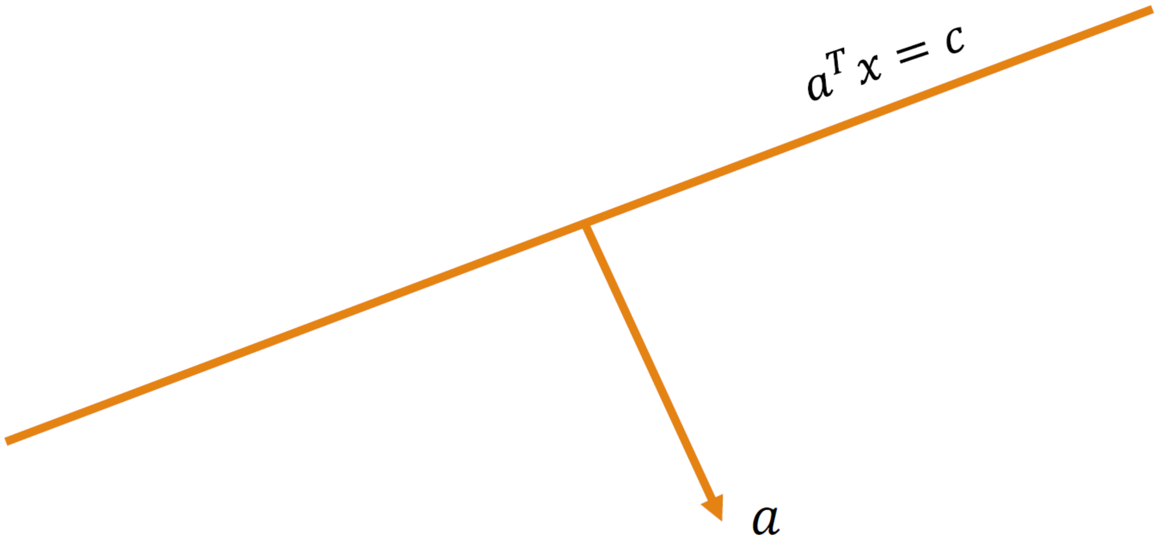
Linear Function

- $f: \mathbb{R}^n \to \mathbb{R}$ is a linear function if $f(x) = a^T x$ 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 objective: f
- Linear constraints:
  - $g(x) = c$, where $g: \mathbb{R}^n \to \mathbb{R}$ is a linear function and $c \in \mathbb{R}$
  - Line in the plane (or a hyperplane in $\mathbb{R}^n$)
  - Example: $5x_1 + 7x_2 = 10$



Linear Function

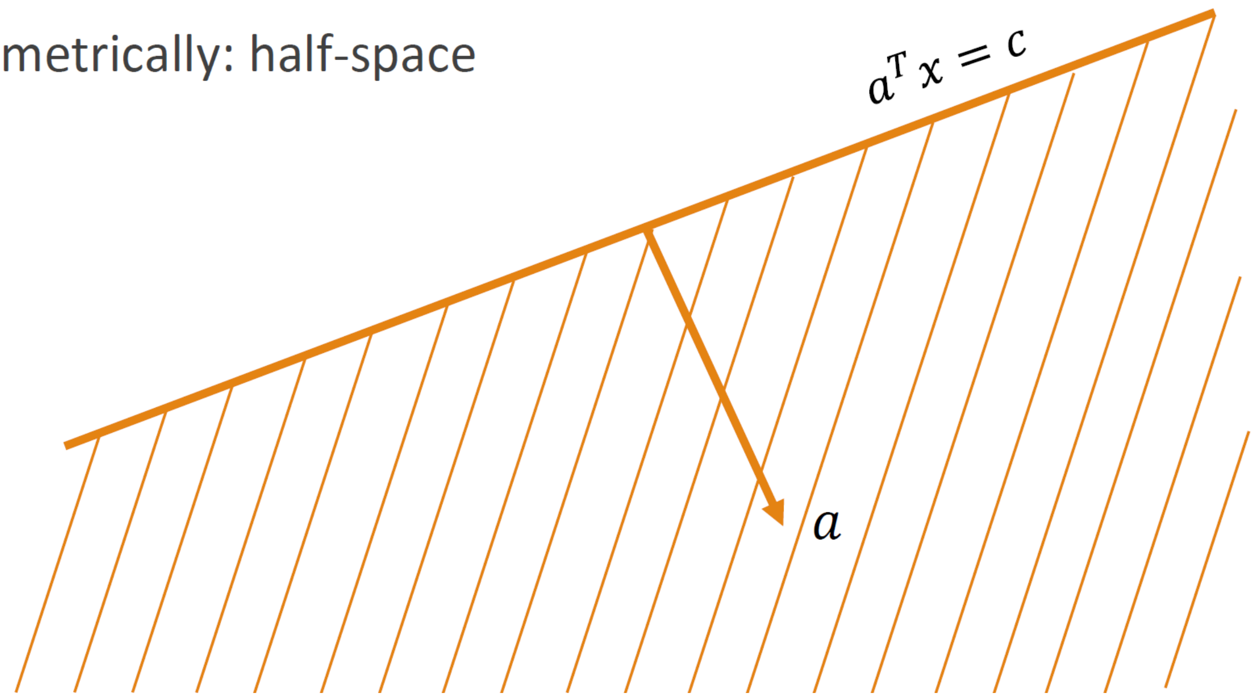
- Geometrically, a is the normal vector of the line(or hyperplane) represented by $a^T x = c$



Linear Inequality

- $a^T x \leq c$ represents a "half-space"

Geometrically: half-space



Linear Programming

- Maximize/minimize a linear function subject to linear equality/inequality constraints

Could be min

Linear objective!

Objective function $\max x_1 + 6x_2$

Constraints

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

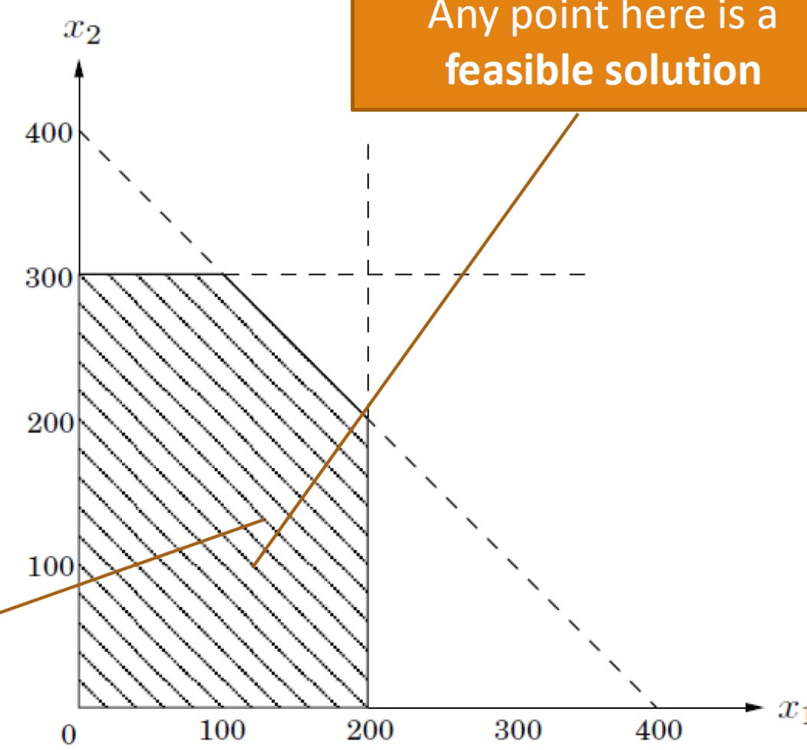
Linear constraints: inequalities/equalities

Geometrically...

Objective function $\max x_1 + 6x_2$

Constraints

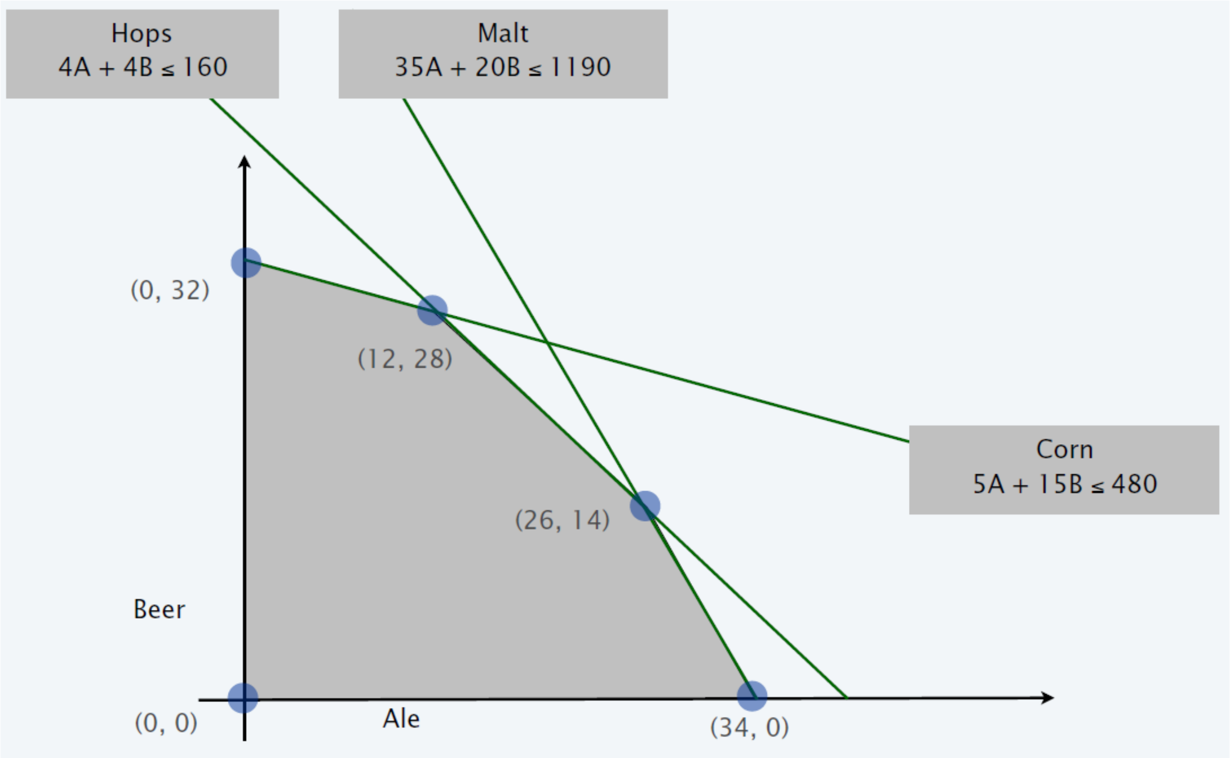
$$x_1 \leq 200$$
$$x_2 \leq 300$$
$$x_1 + x_2 \leq 400$$
$$x_1, x_2 \geq 0$$



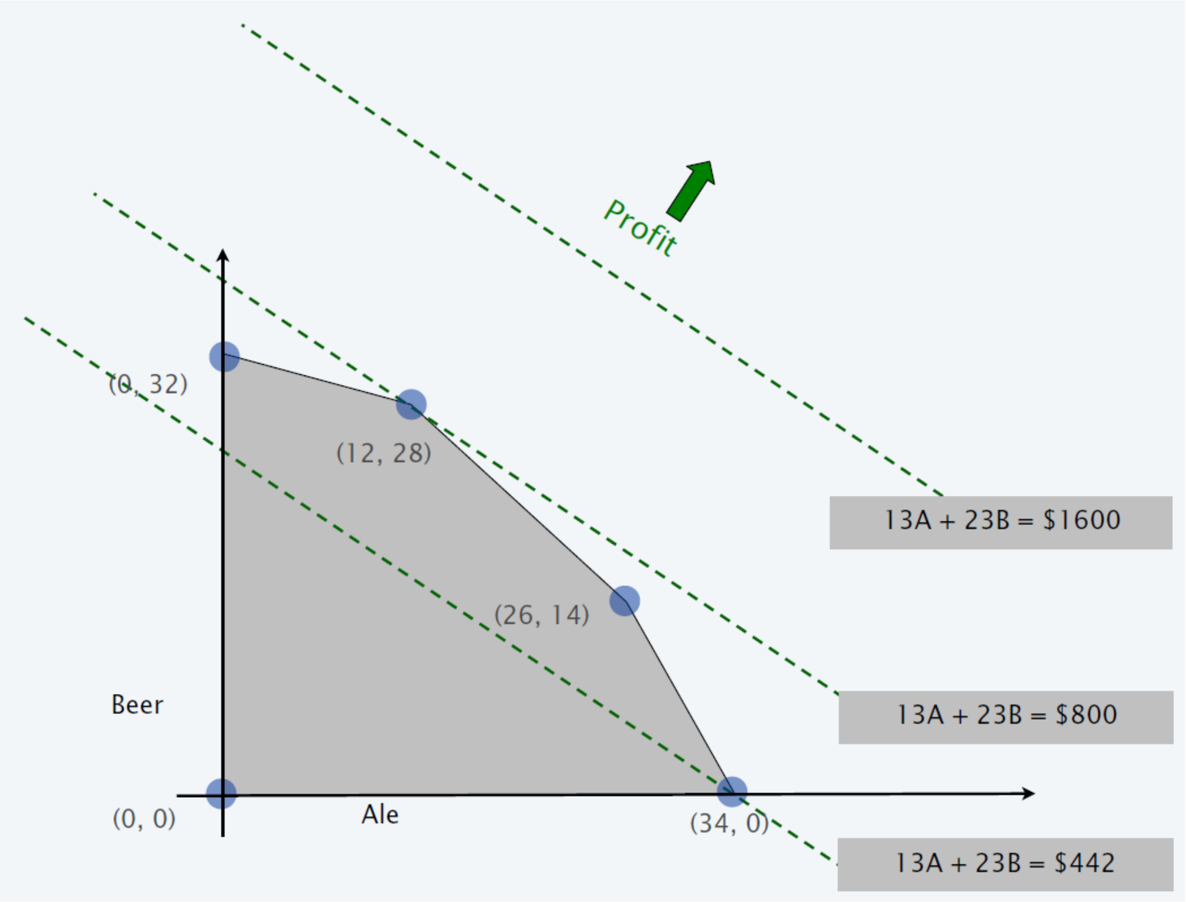
Any point here is a **feasible solution**

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

Back to Brewery Example



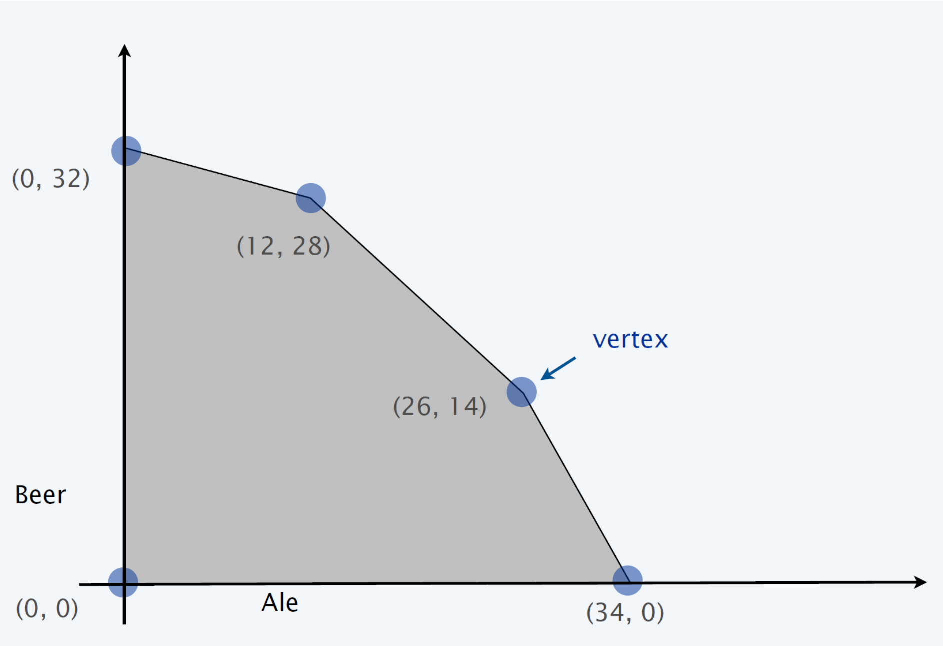
Back to Brewery Example



Optimal Vertex

OUT OF SYLLABUS

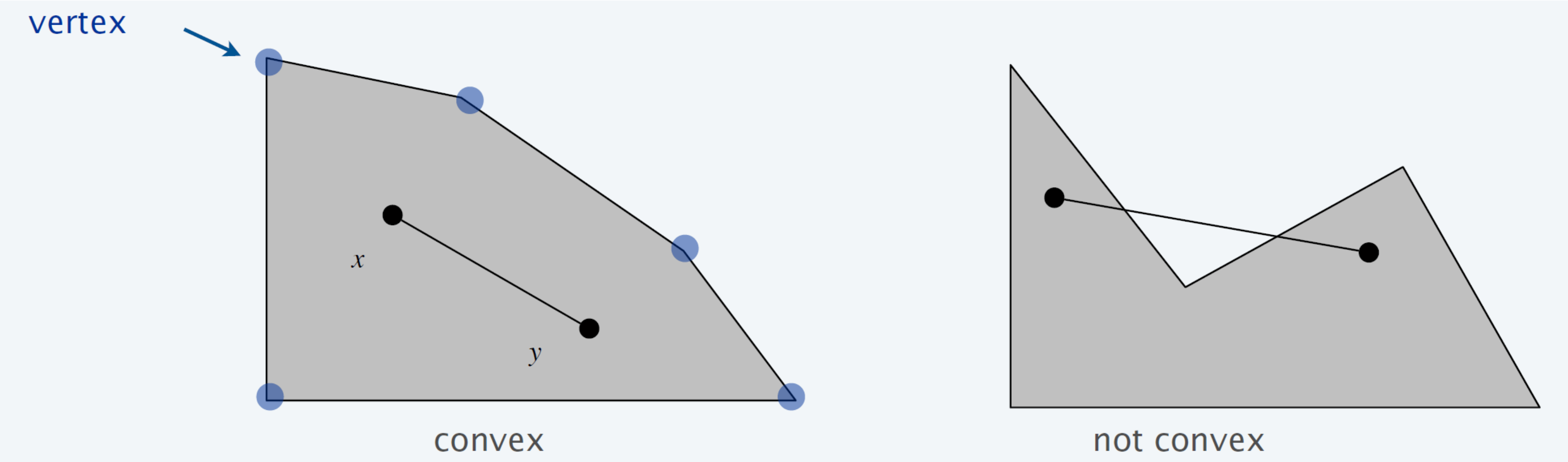
- Claim: Regardless of the objective function, there must be a vertex that is an optimal solution



Optimal Vertex

OUT OF SYLLABUS

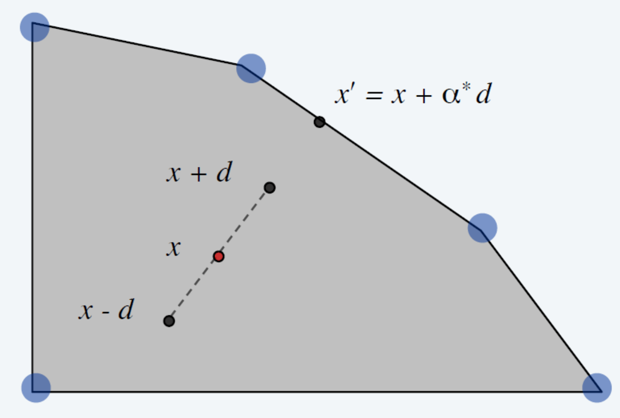
- Convex set: S is convex if
$$x, y \in S, \lambda \in [0,1] \Rightarrow \lambda x + (1-\lambda) y \in S$$
- Vertex: A point which cannot be written as a strict convex combination of any two points in the set
- Observation: Feasible region of an LP is a convex set



Optimal Vertex

OUT OF SYLLABUS

- Intuitive proof of the claim:
  - Start at some point x in the feasible region
  - If x is not a vertex:
    - Find a direction d such that points within a positive distance of ϵ from x in both d and $-d$ directions are within the feasible region
    - Objective must *not decrease* in at least one of the two directions
    - Follow that direction until you reach a new point x for which at least one more constraint is "tight"
  - Repeat until we are at a vertex



LP, Standard Formulation

- **Input:** $c, a_1, a_2, \dots, a_m \in \mathbb{R}^n, b \in \mathbb{R}^m$
  - There are n variables and m constraints
- **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

LP, Standard Matrix Form

- Input: $c, a_1, a_2, \ldots, a_m \in \mathbb{R}^n, b \in \mathbb{R}^m$
  - There are n variables and m constraints
- Goal:

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

n variables

m constraints

n more constraints

LP Tricks I

- What if the LP is not in standard form?
  - Constraints that use $\geq$
    - $a^T x \geq b \;\; \Leftrightarrow \;\; -a^T x \leq -b$
  - Constraints that use equality
    - $a^T x = b \;\; \Leftrightarrow \;\; a^T x \leq b, \;\; a^T x \geq b$
  - Objective function is a minimization
    - Minimize $c^T x$ $\Leftrightarrow$ Maximize $-c^T x$
  - Variable is unconstrained
    - x with no constraint $\Leftrightarrow$ Replace x by two variables x' and x'', replace every occurrence of x with $x' - x''$, and add constraints $x' \geq 0, \;\; x'' \geq 0$

LP Transformation Example

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

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

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

LP Tricks II

- Constraint: $|x| \leq 3$
  - Replace with constraints $x \leq 3$ and $-x \leq 3$
  - What if the constraint is $|x| \geq 3$?
- Objective: minimize $3|x| + y$
  - Add a variable t
  - Add the constraints $t \geq x$ and $t \geq -x$ (so $t \geq |x|$)
  - Change the objective to minimize $3t + y$
  - What if the objective is to *maximize* $3|x| + y$?
- Objective: minimize $\max(3x + y, x + 2y)$
  - Hint: minimizing $3|x| + y$ in the earlier bullet was equivalent to minimizing $\max(3x + y, -3x + y)$
- …

Optimal Solution

- Does an LP always have an optimal solution?

- No! The LP can "fail" for two reasons:
  1. It is *infeasible*, i.e., $\{x \mid Ax \leq b\} = \emptyset$
     - E.g., the set of constraints is $\{x_1 \leq 1, -x_1 \leq -2\}$
  2. It is *unbounded*, i.e., the objective function can be made arbitrarily large (for maximization) or small (for minimization)
     - E.g., "maximize x_1 subject to $x_1 \geq 0$"

- But if the LP has an optimal solution, we know that there must be a vertex which is optimal

Simplex Algorithm

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

- Simple algorithm
  - Easy to specify geometrically, but quite tricky to implement given just the LP in the standard form

- Worst-case running time
  - #vertices of feasible region can be exponential
  - Excellent performance in practice on many classes of LPs

Running Time for LPs

Year	Algorithm	Running Time
1947	Dantzig's Simplex	Exponential
1979	Khachiyan's Ellipsoid	$O(n^6 L)$
1984	Karmarkar's projective method	$O(n^{3.5} L)$
1989	Vaidya's method	$O\left((n+m)^{1.5} nL\right)$
2019	Cohen, Lee, Song, Zhang	$\tilde{O}(n^{2+1/6}\, L)$
2020	Jiang, Song, Weinstein, Zhang	$\tilde{O}(n^{2+1/18}\, L)$

n = #variables
m = #constraints
L = #bits of input

Duality

Certificate of Optimality

- Suppose you design a state-of-the-art LP solver that can solve very large problem instances

- You want to convince someone that you have this new technology without showing them the code
  - Idea: They can give you very large LPs and you can quickly return the optimal solutions
  - Question: But how would they know that your solutions are optimal, if they don't have the technology to solve those LPs?

Certificate of Optimality

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

- Suppose I tell you that $(x_1, x_2) = (100{,}300)$ is optimal with objective value 1900
- How can you check this?
  - Note: Can easily substitute (x_1, x_2), and verify that it is feasible, and its objective value is indeed 1900

Certificate of Optimality

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

- Claim: $(x_1, x_2) = (100{,}300)$ is optimal with objective value 1900

- Any solution that satisfies these inequalities also satisfies their positive combinations
  - E.g. 2*first_constraint + 5*second_constraint + 3*third_constraint
  - Try to take combinations which give you $x_1 + 6x_2$ on LHS

Certificate of Optimality

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

- Claim: $(x_1, x_2) = (100{,}300)$ is optimal with objective value 1900

- first_constraint + 6*second_constraint
  - $x_1 + 6x_2 \leq 200 + 6 * 300 = 2000$
  - This shows that no feasible solution can beat 2000

Certificate of Optimality

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

- Claim: $(x_1, x_2) = (100{,}300)$ is optimal with objective value 1900

- 5*second_constraint + third_constraint
  - $5x_2 + (x_1 + x_2) \le 5 * 300 + 400 = 1900$
  - This shows that no feasible solution can beat 1900
    - No need to proceed further
    - We already know one solution that achieves 1900, so it must be optimal!

Is There a General Algorithm?

- Introduce variables y_1, y_2, y_3 by which we will be multiplying the three constraints
  - Note: These need not be integers. They can be reals.

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

- After multiplying and adding constraints, we get:
  $(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \leq 200y_1 + 300y_2 + 400y_3$

Is There a General Algorithm?

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

- We have:
  $(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \leq 200y_1 + 300y_2 + 400y_3$

- What do we want?
  - $y_1, y_2, y_3 \geq 0$ because otherwise direction of inequality flips
  - LHS to look like objective $x_1 + 6x_2$
    - In fact, it is sufficient for LHS to be an upper bound on objective
    - So, we want $y_1 + y_3 \geq 1$ and $y_2 + y_3 \geq 6$

Is There a General Algorithm?

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

- We have:
  $(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \leq 200y_1 + 300y_2 + 400y_3$

- What do we want?
  - $y_1, y_2, y_3 \geq 0$
  - $y_1 + y_3 \geq 1, \; y_2 + y_3 \geq 6$
  - Subject to these, we want to minimize the upper bound $200y_1 + 300y_2 + 400y_3$

Is There a General Algorithm?

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

- We have:

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

- What do we want?
  - This is just another LP!
  - Called the dual
  - Original LP is called the primal

$$\min\ 200y_1 + 300y_2 + 400y_3$$

$$y_1 + y_3 \geq 1$$

$$y_2 + y_3 \geq 6$$

$$y_1, y_2, y_3 \geq 0$$

Is There a General Algorithm?

PRIMAL

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

DUAL

$$\min \; 200y_1 + 300y_2 + 400y_3$$
$$y_1 + y_3 \geq 1$$
$$y_2 + y_3 \geq 6$$
$$y_1, y_2, y_3 \geq 0$$

- The problem of verifying optimality is another LP
  - For any (y_1, y_2, y_3) that you can find, the objective value of the dual is an upper bound on the objective value of the primal
  - If you found a specific (y_1, y_2, y_3) for which this dual objective becomes equal to the primal objective for the (x_1, x_2) given to you, then you would know that the given (x_1, x_2) is optimal for primal (and your (y_1, y_2, y_3) is optimal for dual)

Is There a General Algorithm?

PRIMAL

$$\max \; x_1 + 6x_2$$
$$x_1 \le 200$$
$$x_2 \le 300$$
$$x_1 + x_2 \le 400$$
$$x_1, x_2 \ge 0$$

DUAL

$$\min \; 200y_1 + 300y_2 + 400y_3$$
$$y_1 + y_3 \ge 1$$
$$y_2 + y_3 \ge 6$$
$$y_1, y_2, y_3 \ge 0$$

- The problem of verifying optimality is another LP
  - Issue 1: But…but…if I can't solve large LPs, how will I solve the dual to verify if optimality of (x_1, x_2) given to me?
    - You don't. Ask the other party to give you both (x_1, x_2) and the corresponding (y_1, y_2, y_3) for proof of optimality
  - Issue 2: What if there are no (y_1, y_2, y_3) for which dual objective matches primal objective under optimal solution (x_1, x_2)?
    - As we will see, this can't happen!

Is There a General Algorithm?

Primal LP

$$\max\ \mathbf{c}^T\mathbf{x}$$
$$\mathbf{A}\mathbf{x} \le \mathbf{b}$$
$$\mathbf{x} \ge 0$$

Dual LP

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

- General version, in our standard form for LPs

Is There a General Algorithm?

Primal LP	**Dual LP**
$\max\ \mathbf{c}^T\mathbf{x}$	$\min\ \mathbf{y}^T\mathbf{b}$
$\mathbf{A}\mathbf{x} \leq \mathbf{b}$	$\mathbf{y}^T\mathbf{A} \geq \mathbf{c}^T$
$\mathbf{x} \geq 0$	$\mathbf{y} \geq 0$

- $c^T x$ for any feasible x $\leq$ $y^T b$ for any feasible y
- $\max_{\text{primal feasible } x} c^T x \leq \min_{\text{dual feasible } y} y^T b$
- If there is (x^*, y^*) with $c^T x^* = (y^*)^T b$, then both must be optimal
- In fact, for optimal (x^*, y^*), we claim that this must happen!
  - Does this remind you of something? Max-flow, min-cut…

Weak Duality

Primal LP

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

Dual LP

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

- From here on, assume primal LP is feasible and bounded
- Weak duality theorem:
  - For any primal feasible x and dual feasible y, $c^T x \leq y^T b$
- Proof:

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

Strong Duality

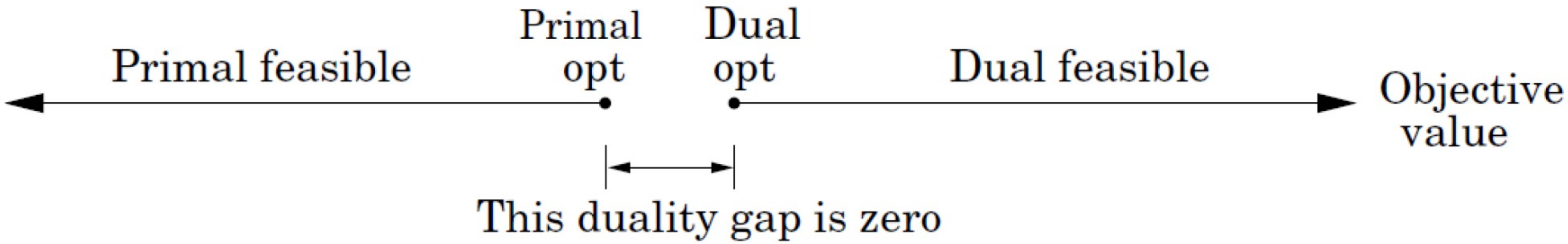
Primal LP

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

Dual LP

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

- Strong duality theorem:
  - For any primal optimal x^* and dual optimal y^*, $c^T x^* = (y^*)^T b$



Applications of Linear Programming

Network Flow via LP

- Problem
  - Input: directed graph $G = (V, E)$, edge capacities $c: E \to \mathbb{R}_{\geq 0}$
  - Output: Value $v(f^*)$ of a maximum flow f^*

- Flow f is valid if:
  - Capacity constraints: $\forall (u, v) \in E: 0 \leq f(u, v) \leq c(u, v)$
  - Flow conservation: $\forall u: \sum_{(u,v)\in E} f(u, v) = \sum_{(v,u)\in E} f(v, u)$

- Maximize $v(f) = \sum_{(s,v)\in E} f(s, v)$

Linear constraints

Linear objective!

Network Flow via LP

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

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

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

Exercise: Write the dual of this LP.
What is the dual trying to find?

Shortest Path via LP

- Problem
  - Input: directed graph $G = (V, E)$, edge weights $w: E \to \mathbb{R}_{\geq 0}$, source vertex s, target vertex t
  - Output: weight of the shortest-weight path from s to t

- Variables: for each vertex v, we have variable d_v

Why max?

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

Exercise: prove formally that this works!

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

But...but...

- For these problems, we have different combinatorial algorithms that are much faster and run in strongly polynomial time

- Why would we use LP?

- For some problems, we don't have faster algorithms than solving them via LP

Multicommodity-Flow

- Problem:
  - Input: directed graph $G = (V, E)$, edge capacities $c: E \to \mathbb{R}_{\geq 0}$, k commodities (s_i, t_i, d_i), where s_i is source of commodity i, t_i is sink, and d_i is demand.
  - Output: valid multicommodity flow $(f_1, f_2, \dots, f_k)$, where f_i has value d_i and all f_i jointly satisfy the constraints

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

The only known polynomial time algorithm for this problem is based on solving LP!

Integer Linear Programming

- Variable values are restricted to be integers
- Example:
  - Input: $c \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$
  - Goal:

$$\begin{aligned} \text{Maximize} \quad & c^T x \\ \text{Subject to} \quad & Ax \leq b \\ & \boldsymbol{x} \in \{\mathbf{0}, \mathbf{1}\}^{\boldsymbol{n}} \end{aligned}$$

- Does this make the problem easier or harder?
  - Harder. We'll later prove that this is "NP-complete".

LPs are everywhere…

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

Exercise: Formulating LPs

- A canning company operates two canning plants (A and B).
- Three suppliers of fresh fruits:
  - S1: 200 tonnes at $11/tonne
  - S2: 310 tonnes at $10/tonne
  - S3: 420 tonnes at $9/tonne
- Shipping costs in $/tonne:

From:	To: Plant A	Plant B
S1	3	3.5
S2	2	2.5
S3	6	4

- Plant capacities and labour costs:

	Plant A	Plant B
Capacity	460 tonnes	560 tonnes
Labour cost	$26/tonne	$21/tonne

- Selling price: $50/tonne, no limit
- Objective: Find which plant should get how much supply from each grower to maximize profit

Exercise: Formulating LPs

- Similarly to the brewery example from earlier:
  - A brewery can invest its inventory of corn, hops and malt into producing three types of beer
  - Per unit resource requirement and profit are as given below
  - The brewery cannot produce positive amounts of *both* A and B
  - Goal: maximize profit

Beverage	Corn (kg)	Hops (kg)	Malt (kg)	Profit ($)
A	5	4	35	13
B	15	4	20	23
C	10	7	25	15
Limit	500	300	1000	

Exercise: Formulating LPs

- Similarly to the brewery example from the beginning:
  - A brewery can invest its inventory of corn, hops and malt into producing three types of beer
  - Per unit resource requirement and profit are as given below
  - The brewery can only produce C in integral quantities up to 100
  - Goal: maximize profit

Beverage	Corn (kg)	Hops (kg)	Malt (kg)	Profit ($)
A	5	4	35	13
B	15	4	20	23
C	10	7	25	15
Limit	500	300	1000	

Exercise: Formulating LPs

- Similarly to the brewery example from the beginning:
  - A brewery can invest its inventory of corn, hops and malt into producing three types of beer
  - Per unit resource requirement and profit are as given below
  - Goal: maximize profit, but if there are multiple profit-maximizing solutions, then…
    - Break ties to choose those with the largest quantity of A
    - Break any further ties to choose those with the largest quantity of B

Beverage	Corn (kg)	Hops (kg)	Malt (kg)	Profit ($)
A	5	4	35	13
B	15	4	20	23
C	10	7	25	15
Limit	500	300	1000	


NOW I KNOW
LINEAR PROGRAMMING