CSE 521: Algorithms

Linear Programming

Slides by Paul Beame, Anna Karlin, probably nameless others, … and occasionally L. Ruzzo

Linear Programming

- The process of minimizing a linear objective function subject to a finite number of linear equality and inequality constraints.
- Like “dynamic programming”, the word “programming” is historical and predates computer programming.
- Example applications:
  - airline crew scheduling
  - manufacturing and production planning
  - telecommunications network design
- “Few problems studied in computer science have greater application in the real world.”

Applications

“Delta Air Lines flies over 2,500 domestic flight legs every day, using about 450 aircraft from 10 different fleets. The fleet assignment problem is to match aircraft to flight legs so that seats are filled with paying passengers. Recent advances in mathematical programming algorithms and computer hardware make it possible to solve optimization problems of this scope for the first time. Delta is the first airline to solve to completion one of the largest and most difficult problems in this industry. Use of the Coldstart model is expected to save Delta Air Lines $300 million over the next three years.”

References – many, e.g.:

Ch 7 of text by Dasgupta, Papadimitriou, Vazirani

http://www.cse.ucsd.edu/users/dasgupta/mcgrawhill/chap7.pdf

“Understanding and Using Linear Programming” by Matousek & Gartner

“Linear Programming”, by Howard Karloff

Simplex section available through Google books preview

“Linear Algebra and Its Applications”, by G Strang, ch 8

“Linear Programming”, by Vasek Chvatal

“Intro to Linear Optimization”, by Bertsimas & Tsitsiklis

An Example: The Diet Problem

- A student is trying to decide on lowest cost diet that provides sufficient amount of protein, with two choices:
  - steak: 2 units of protein/pound, \$3/pound
  - peanut butter: 1 unit of protein/pound, \$2/pound
- In proper diet, need 4 units protein/day.

Let x = # pounds peanut butter/day in the diet.

Let y = # pounds steak/day in the diet.

Goal: minimize $2x + 3y$ (total cost)
subject to constraints:

$x + 2y \geq 4$

$x \geq 0, \quad y \geq 0$

This is an LP- formulation of our problem

An Example: The Diet Problem

Goal: minimize $2x + 3y$ (total cost)
subject to constraints:

$x + 2y \geq 4$

$x \geq 0, \quad y \geq 0$

- This is an optimization problem.
- Any solution meeting the nutritional demands is called a *feasible solution*
- A feasible solution of minimum cost is called the *optimal solution*.

Linear Program - Definition

A linear program is a problem with n variables $x_1,\ldots,x_n$, that has:

1. A linear objective function, which must be minimized/maximized. Looks like:

   $$\min\ (\max)\ c_1x_1+c_2x_2+\ldots+c_nx_n$$

2. A set of m linear constraints. A constraint looks like:

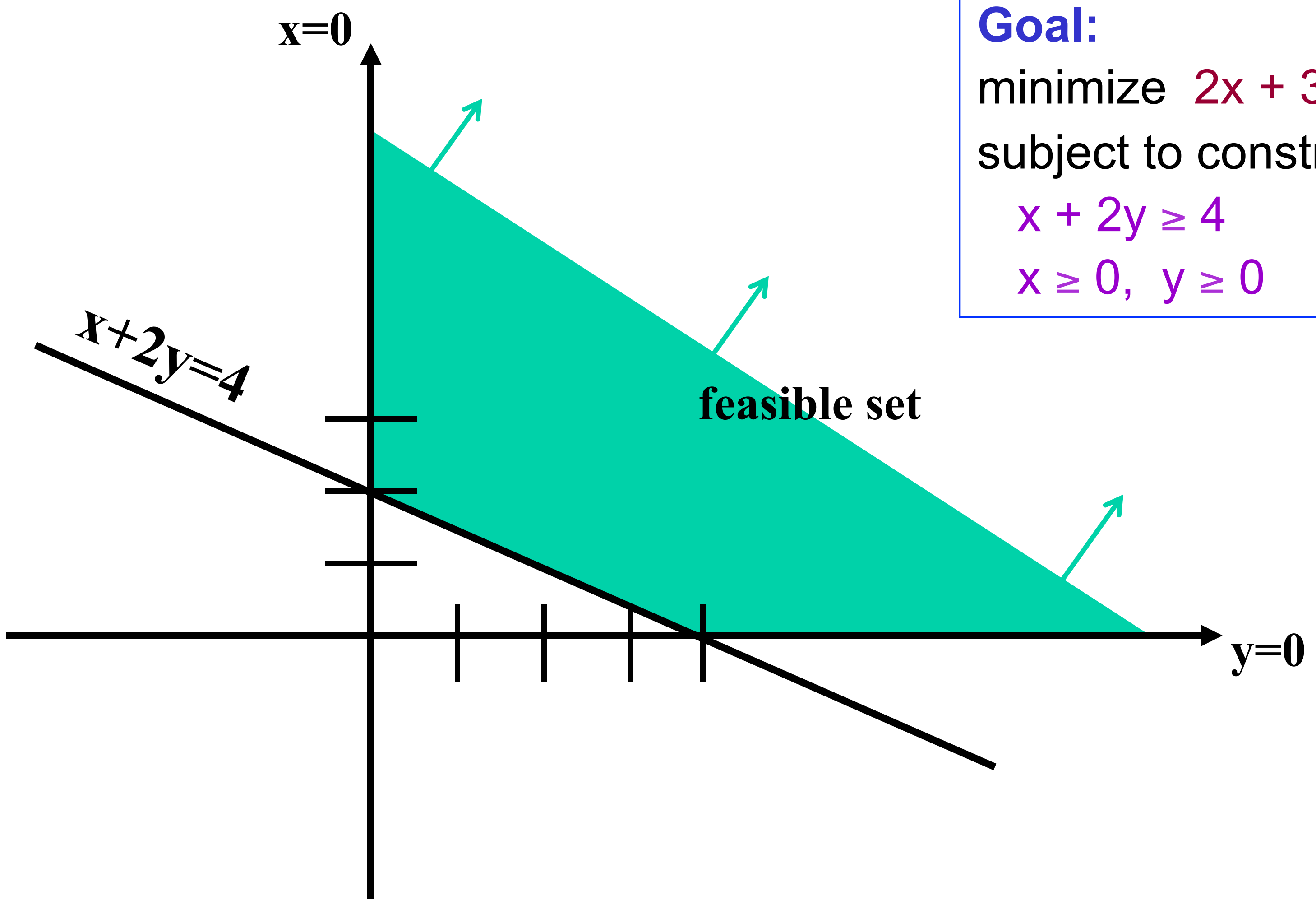
   $$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \le b_i \text{ (or } \ge \text{ or } = \text{)}$$

Note: the values of the coefficients c_i, $a_{i,j}$ are given in the problem input.

Feasible Set

- 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 linear cost function. Defines a family of parallel hyperplanes (lines in 2D, planes in 3D, etc.). Want to find one of minimum cost → must occur at a vertex (corner) of the feasible set.

Visually…
x = peanut butter, y = steak



Goal:

minimize $2x + 3y$ (cost)

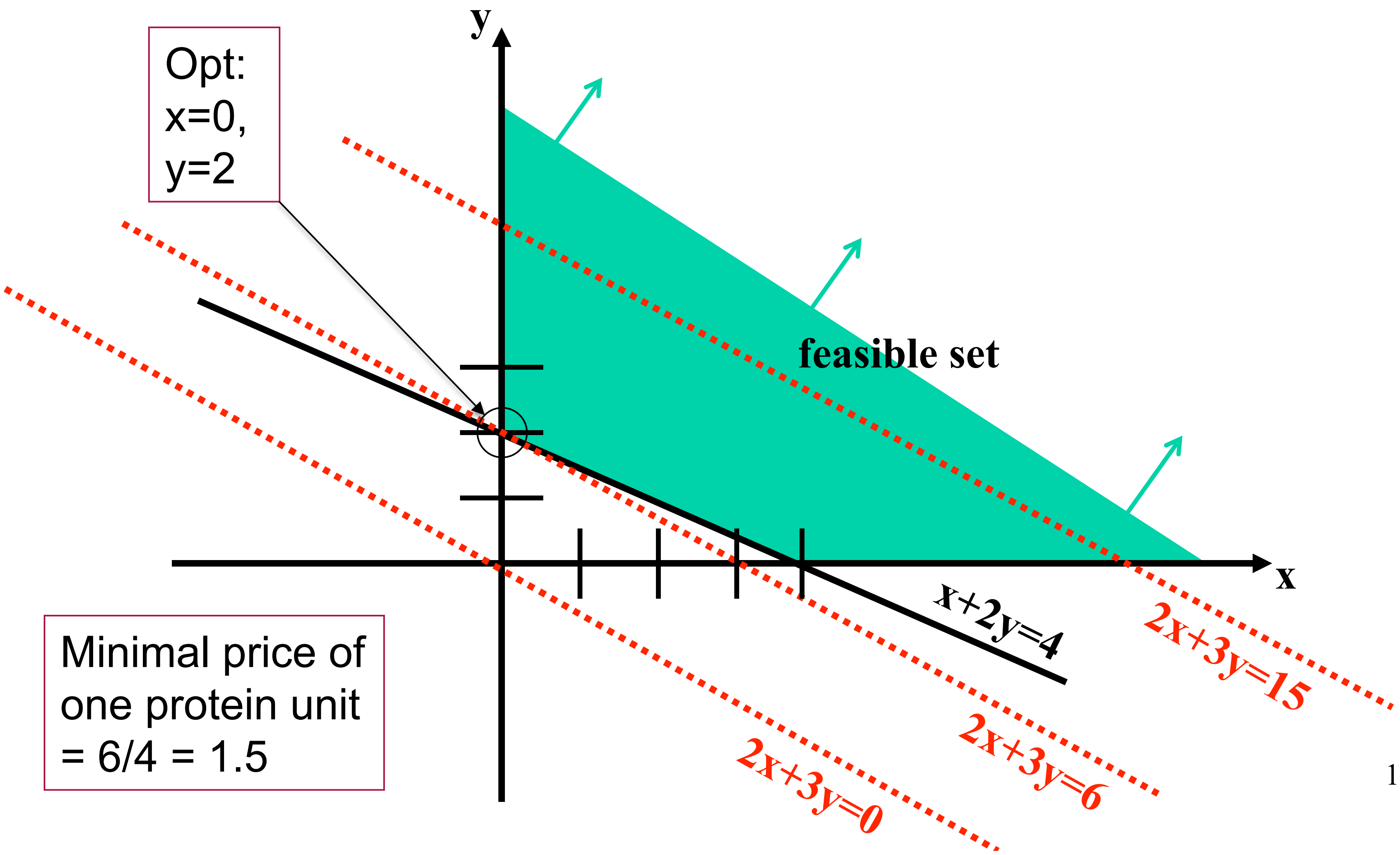
subject to constraints:

$x + 2y \geq 4$

$x \geq 0, \quad y \geq 0$

Optimal vector occurs at some corner of the feasible set

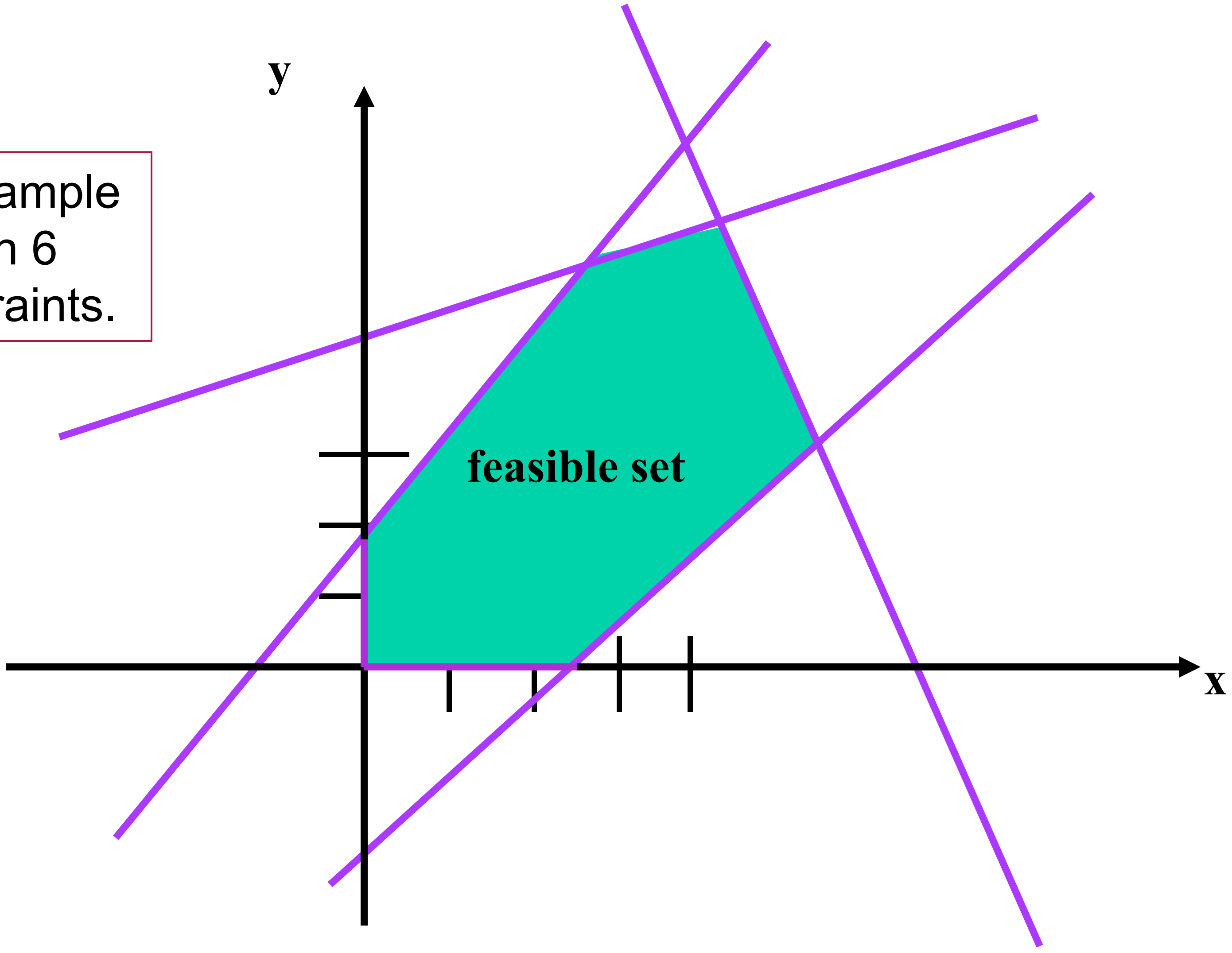
Opt:
x=0,
y=2



Minimal price of one protein unit = 6/4 = 1.5

Optimal vector occurs at some corner of the feasible set

An example with 6 constraints.



Standard Form of a Linear Program.

Minimize $b_1y_1 + b_2y_2 + \ldots + b_my_m$

subject to $\Sigma_{1 \le i \le m}\, a_{ij}y_i \ge c_j \qquad j = 1..n$

$y_i \ge 0 \qquad i = 1..m$

or

Maximize $c_1x_1 + c_2x_2 + \ldots + c_nx_n$

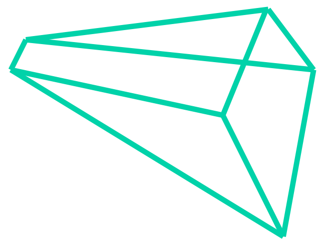
subject to $\Sigma_{1 \le j \le n}\, a_{ij}x_j \le b_i \qquad i = 1..m$

$x_j \ge 0 \qquad j = 1..n$

The Feasible Set

- Intersection of a set of half-spaces is called a *polyhedron*.
- A bounded, nonempty polyhedron is a *polytope*.

There are 3 cases:



- feasible set is empty.
- cost function is unbounded on feasible set.
- cost has a minimum (or max) on feasible set.

(First two cases uncommon for real problems in economics and engineering.)

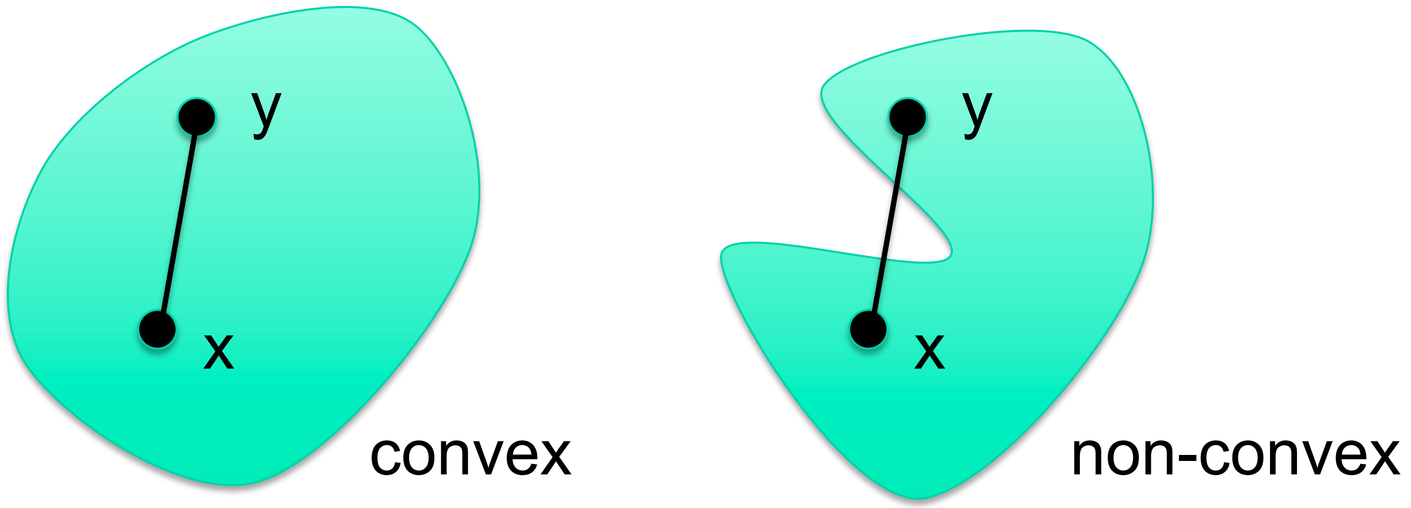
GEOMETRIC INTERLUDE: CONVEXITY

Convexity

A set of points is *convex* if for all pairs x, y in the set, every point on the line segment

$\{ t\,x+(1-t)\,y \mid 0 < t < 1 \}$

connecting them is also in the set



Convexity and half-spaces

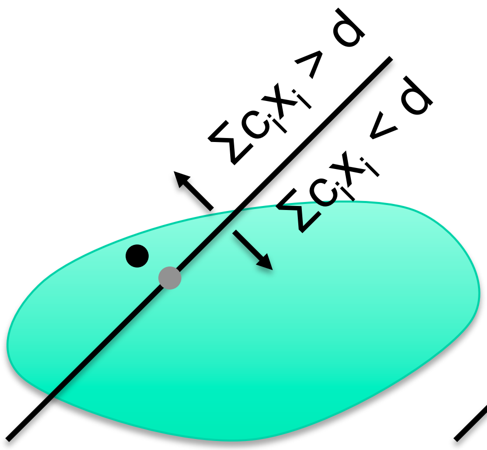
- An inequality

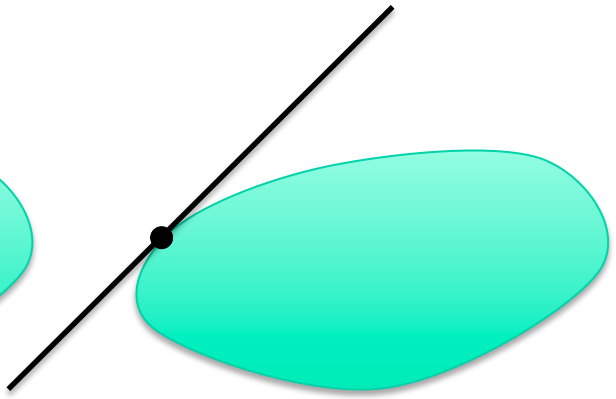
  $a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \leq b$

  defines a *half-space*.
- Half-spaces are convex
- Intersections of convex sets are convex
- So, the feasible region for a linear program is always a convex polyhedron
- Sometimes, edges/faces/etc are distracting; convexity may be all you need

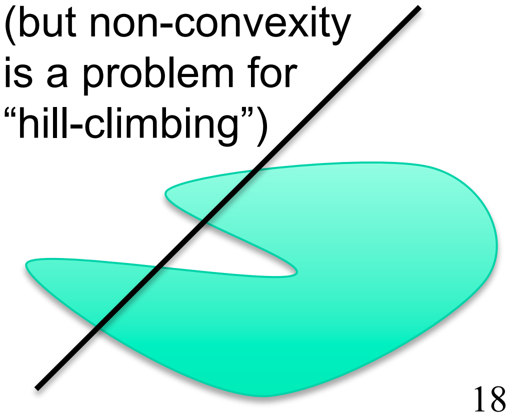
Max/Min is always at a “corner”

- Linear extrema are not in the interior of a convex set
  - E.g.: maximize $c_1x_1+c_2x_2+\ldots+c_nx_n$
  - If max were in the interior, there’s always a better interior point just off the hyperplane $c^Tx = d$
- On a polyhedron, max may = line, face, …, but includes vertices thereof, so always a “corner,” (though maybe not uniquely a corner)



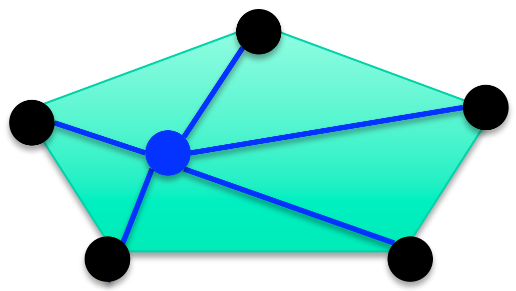


(but non-convexity is a problem for “hill-climbing”)



Convex combinations

- Defn: a *convex combination* of points/ vectors $p_1, p_2, \ldots, p_n$ is a point $\alpha_1 p_1 + \alpha_2 p_2 + \ldots + \alpha_2 p_n$ where $\alpha_i > 0$ and $\sum_i \alpha_i = 1$
- Fact: the set of all convex combinations of $p_1, p_2, \ldots, p_n$ sweep out their convex hull



Another Fact

A linear function is monotonic along any line

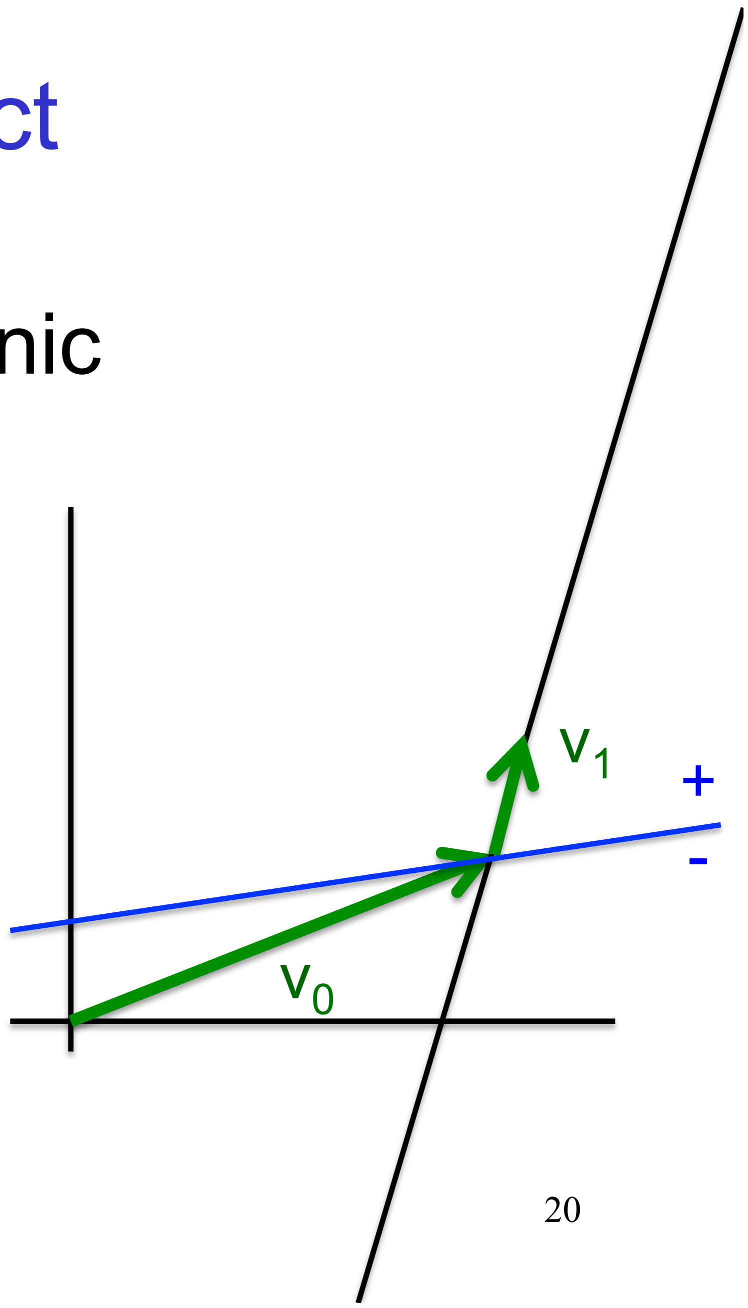
A line: $v_0 + t\,v_1,\ t \in \mathbb{R}$

A linear function: $c^T x$

$c^T(v_0 + t\,v_1) =$

$= (c^T v_0) + t\,(c^T v_1)$

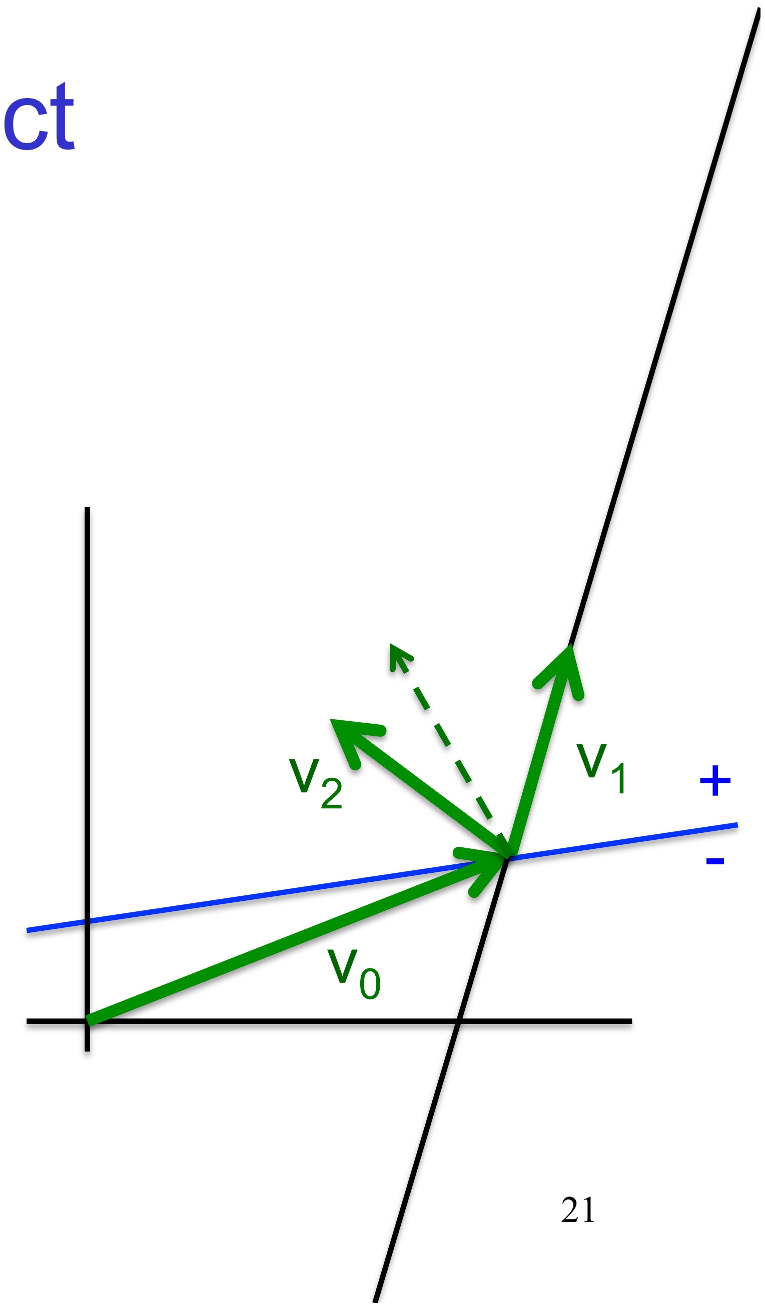
$= d_0 + d_1\,t$ for some constants d_0, d_1



Another Fact

If a linear function is increasing along both v_1 and v_2, then it is increasing along any convex combination $\alpha v_1 + (1-\alpha) v_2$, $0 < \alpha < 1$

Proof: similar



Solving LP

There are several algorithms that solve any linear program optimally.

- The Simplex method (to be discussed) – fast in practice, tho not polynomial-time in worst case
- The Ellipsoid method – polynomial, but impractical
- Interior point methods – polynomial, competes w/ simplex

They can be implemented in various ways.

There are many existing software packages for LP.

It is convenient to use LP as a “black box” for solving various optimization problems.

THE SIMPLEX METHOD

Towards the Simplex Method

The Toy Factory Problem:

A toy factory produces dolls and cars.

Danny, a new employee, can produce 2 cars and/or 3 dolls a day. However, the packaging machine can only pack 4 items a day. The profit from each doll is $10 and from each car is $15. What should Danny be asked to do?

Step 1: Describe the problem as an LP problem.

Let x_1,x_2 denote the number of cars and dolls produced by Danny.

The Toy Factory Problem

Let x_1, x_2 denote the number of cars and dolls produced by Danny.

Objective:

$\text{Max } z = 15x_1 + 10x_2$

s.t.

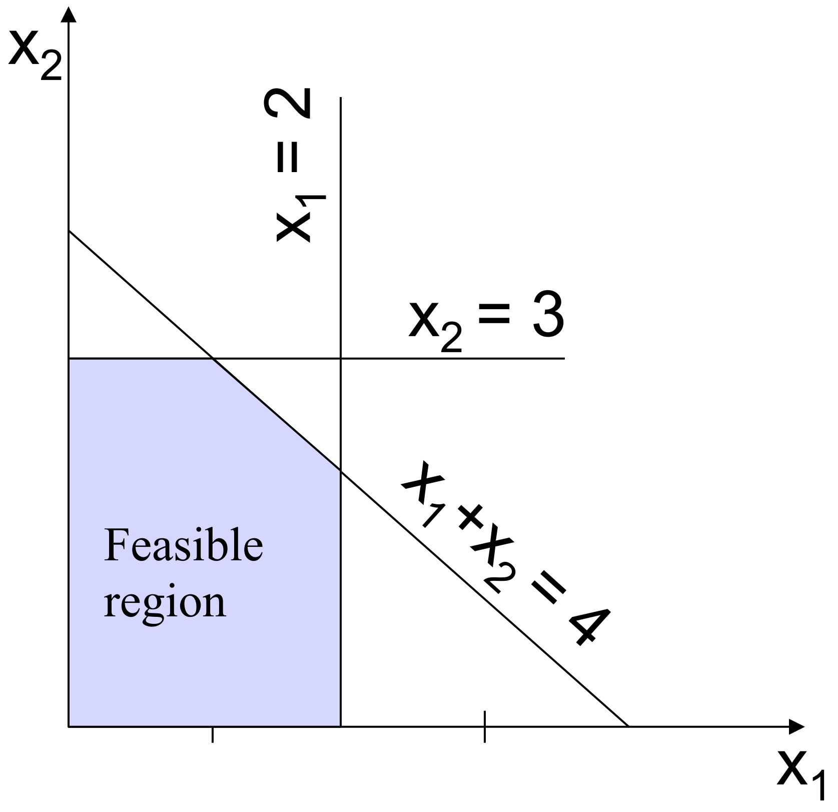
$x_1 \leq 2$

$x_2 \leq 3$

$x_1 + x_2 \leq 4$

$x_1 \geq 0$

$x_2 \geq 0$



The Toy Factory Problem

Let x_1, x_2 denote the number of cars and dolls produced by Danny.

Objective:

$$\text{Max } z = 15x_1 + 10x_2$$

s.t.

$$x_1 \le 2$$
$$x_2 \le 3$$
$$x_1 + x_2 \le 4$$
$$x_1 \ge 0$$
$$x_2 \ge 0$$

Equivalently

$$\text{Max } z = c^T x$$

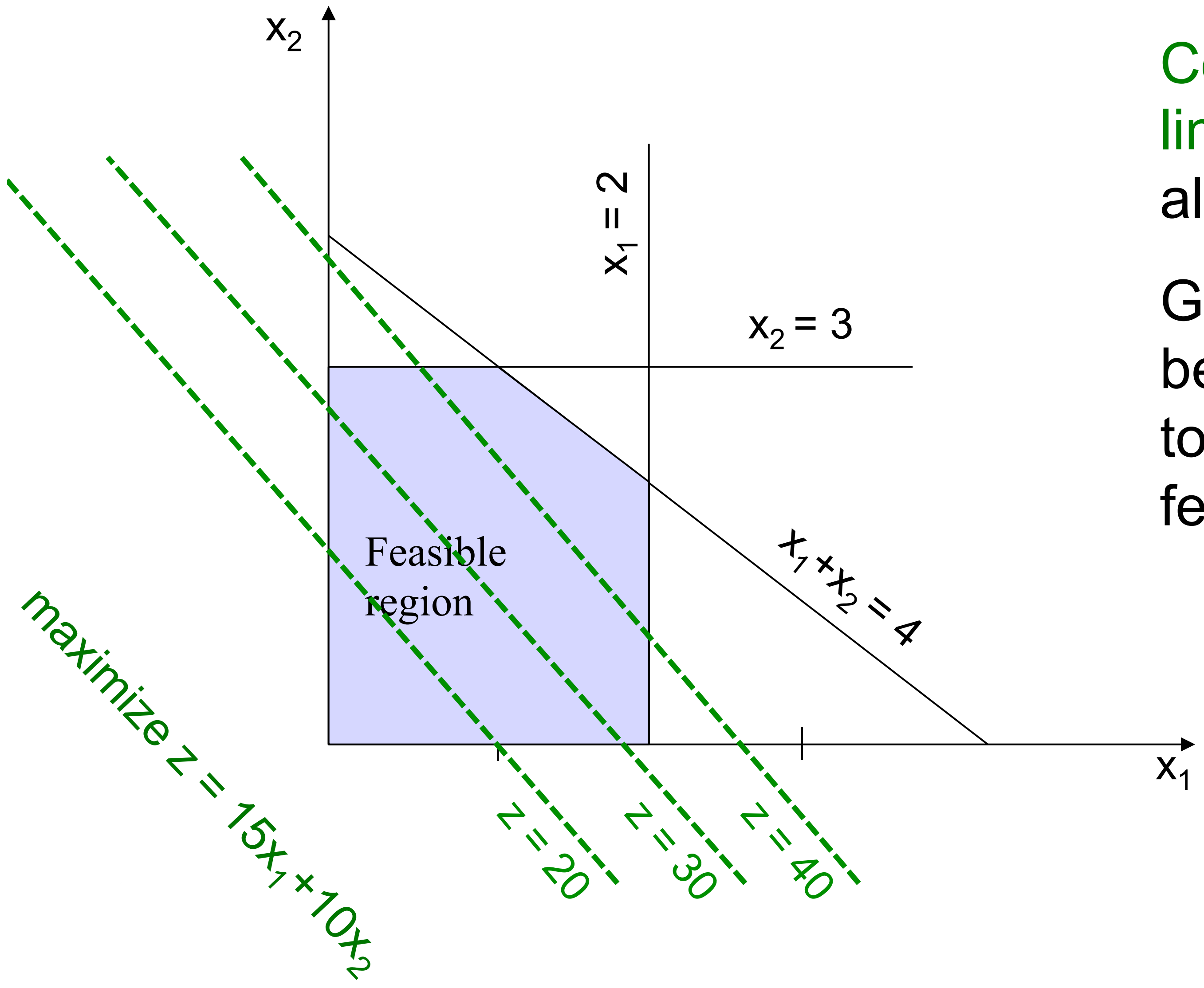
s.t.

$$Ax \le b,$$
$$x \ge 0,$$

where

$$c^T = (15,10), \quad A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}$$
$$b^T = (2,3,4)$$

The Toy Factory Problem

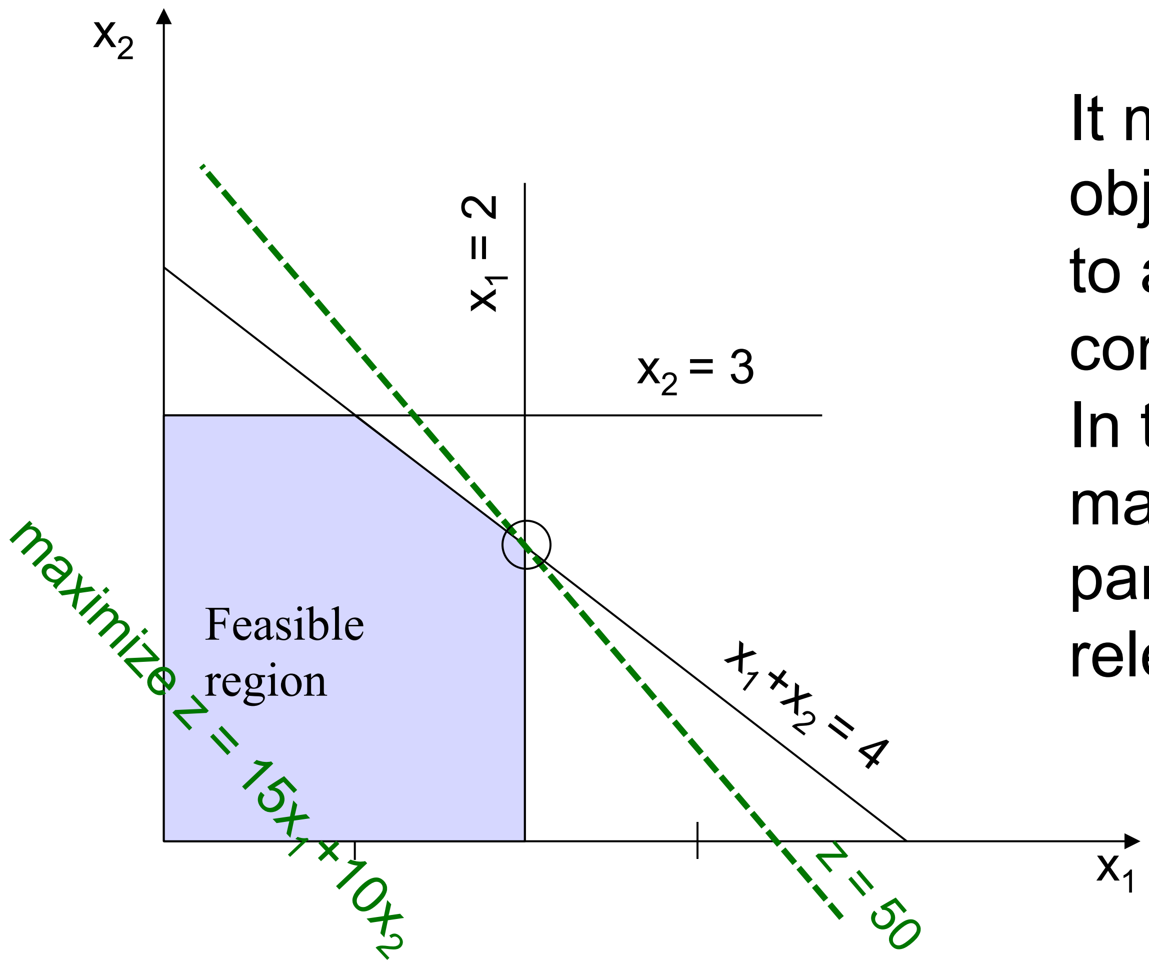


Constant profit lines – They are always parallel.

Goal: Find the best one touching the feasible region.

Important Observations:

1. Optimum solution to the LP is always at a vertex!

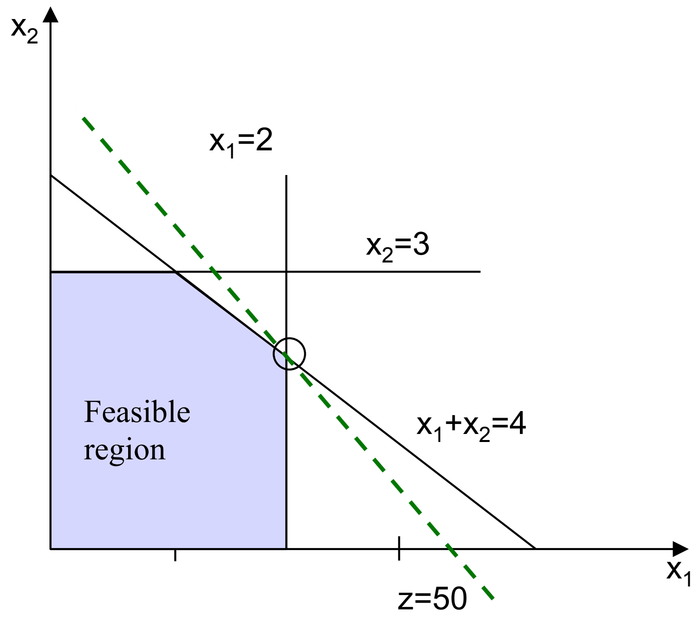


It might be that the objective line is parallel to a constraint (e.g., consider $z=15x_1+15x_2$). In this case there are many optima, but in particular, one is at a relevant vertex.

Important Observations:

2. If a feasible vertex has an objective function value that is better than or equal to all its adjacent feasible vertices, then it is optimal. I.e., a local opt is a global opt. (WHY??)

3. There are a finite number of feasible vertices.



The Simplex method: Hill-climbing on polytope edges:

Travel from one feasible vertex to a neighboring one until a local maximum.

The Simplex Method

Phase 1 (start-up): Find any feasible vertex. In standard LPs the origin can serve as the start vertex.

Phase 2 (iterate): Repeatedly move to a better adjacent feasible vertex until none can be found. The final vertex is the optimum point.

Example: The Toy Factory Problem

Phase 1: start at (0,0)

Objective value = Z(0,0)=0

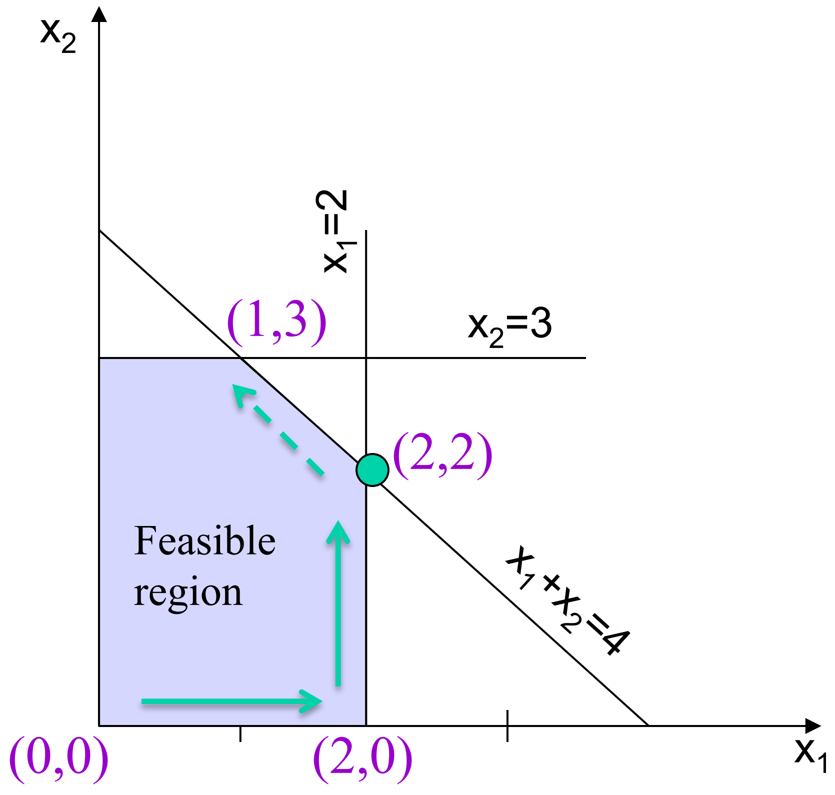
Iteration 1: Move to (2,0).

Z(2,0)=30. An Improvement

Iteration 2: Move to (2,2)

Z(2,2)=50. An Improvement

Iteration 3: Consider moving to (1,3), Z(1,3)=45 < 50. Conclude that (2,2) is optimum!



Finding Feasible Vertices Algebraically

The simplex method is easy to follow graphically. But how is it implemented in practice?

Notes:

- At a vertex a *subset* of the inequalities are equations.
- It is possible to find the intersection of linear equations, i.e., simultaneous solution to a set of eqns
- We will add *slack variables* – to determine which inequalities are *active* (i.e., tight) and which are *not active*

Adding Slack Variables

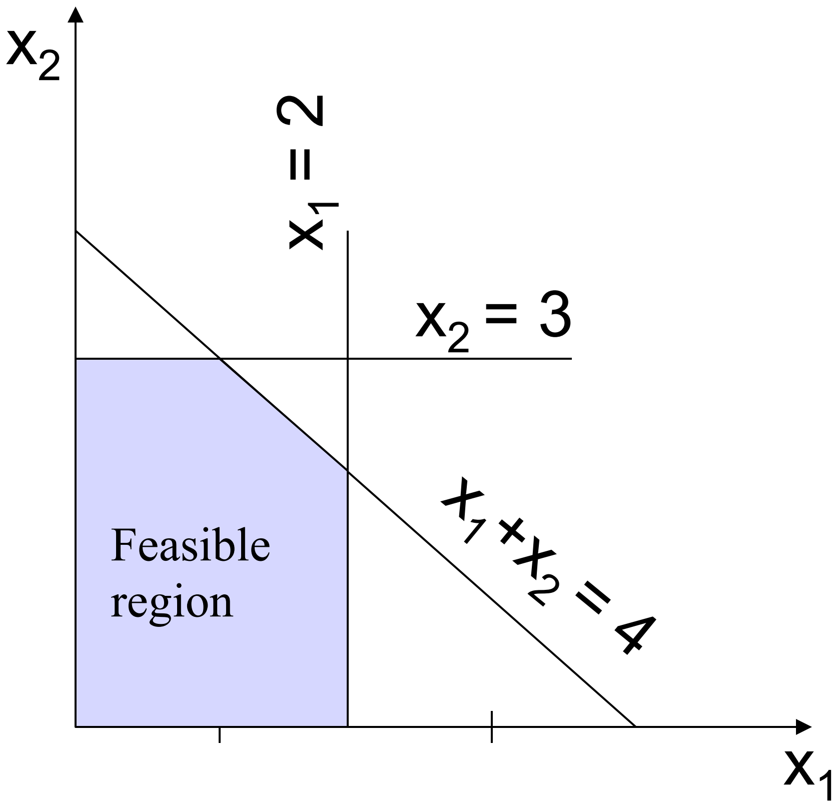
Let s_1, s_2, s_3 be the slack variables

Objective:

Max $z = 15x_1 + 10x_2$

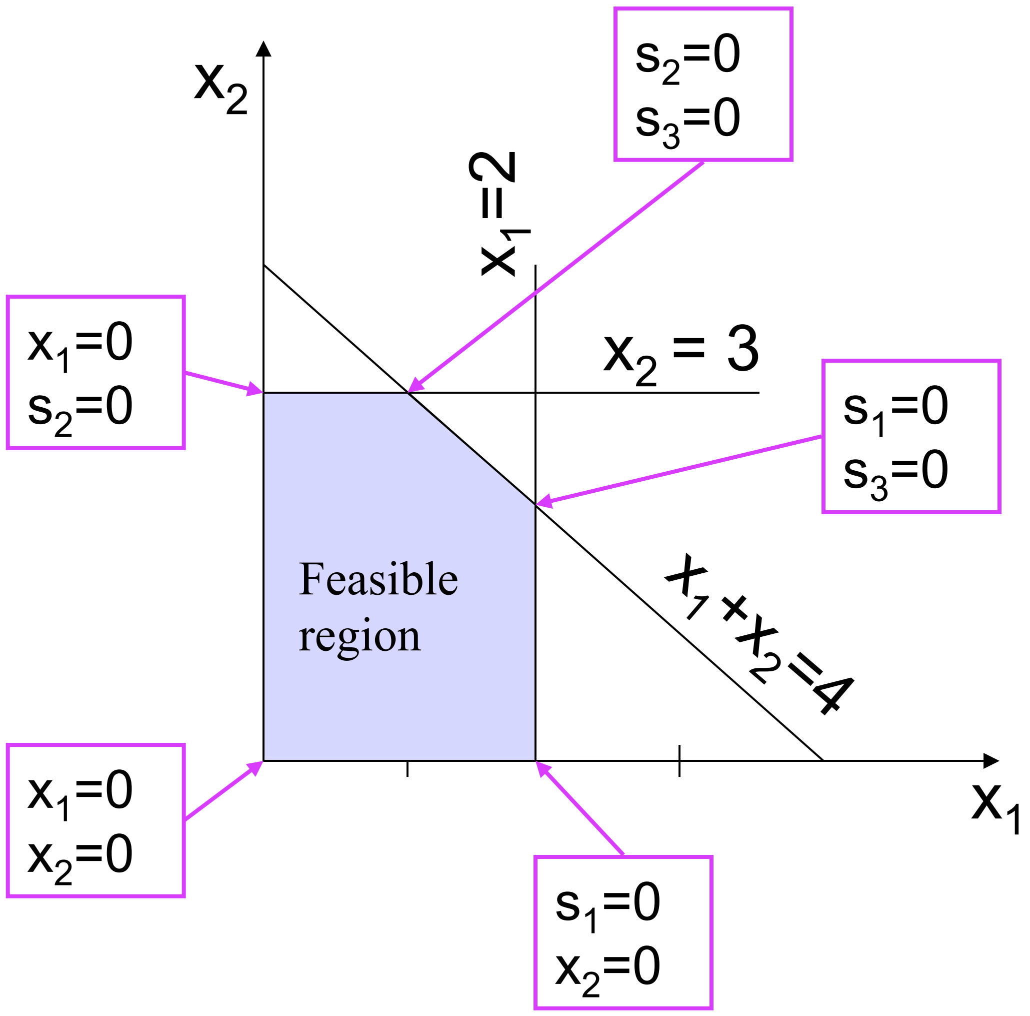
s.t

$$x_1 + s_1 = 2$$
$$x_2 + s_2 = 3$$
$$x_1 + x_2 + s_3 = 4$$
$$x_1, x_2, s_1, s_2, s_3 \geq 0$$



A vertex: Some variables (slack or original) are zero.

Adding Slack Variables



$x_1 + s_1 = 2$

$x_2 + s_2 = 3$

$x_1+x_2 +s_3 = 4$

$x_1, x_2, s_1, s_2, s_3 \geq 0$

Moving between vertices: Decide which two variables are set to zero.

The Simplex Method - Definitions

Nonbasic variable: a variable currently set to zero by the simplex method.

Basic variable: a variable that is not currently set to zero by the simplex method.

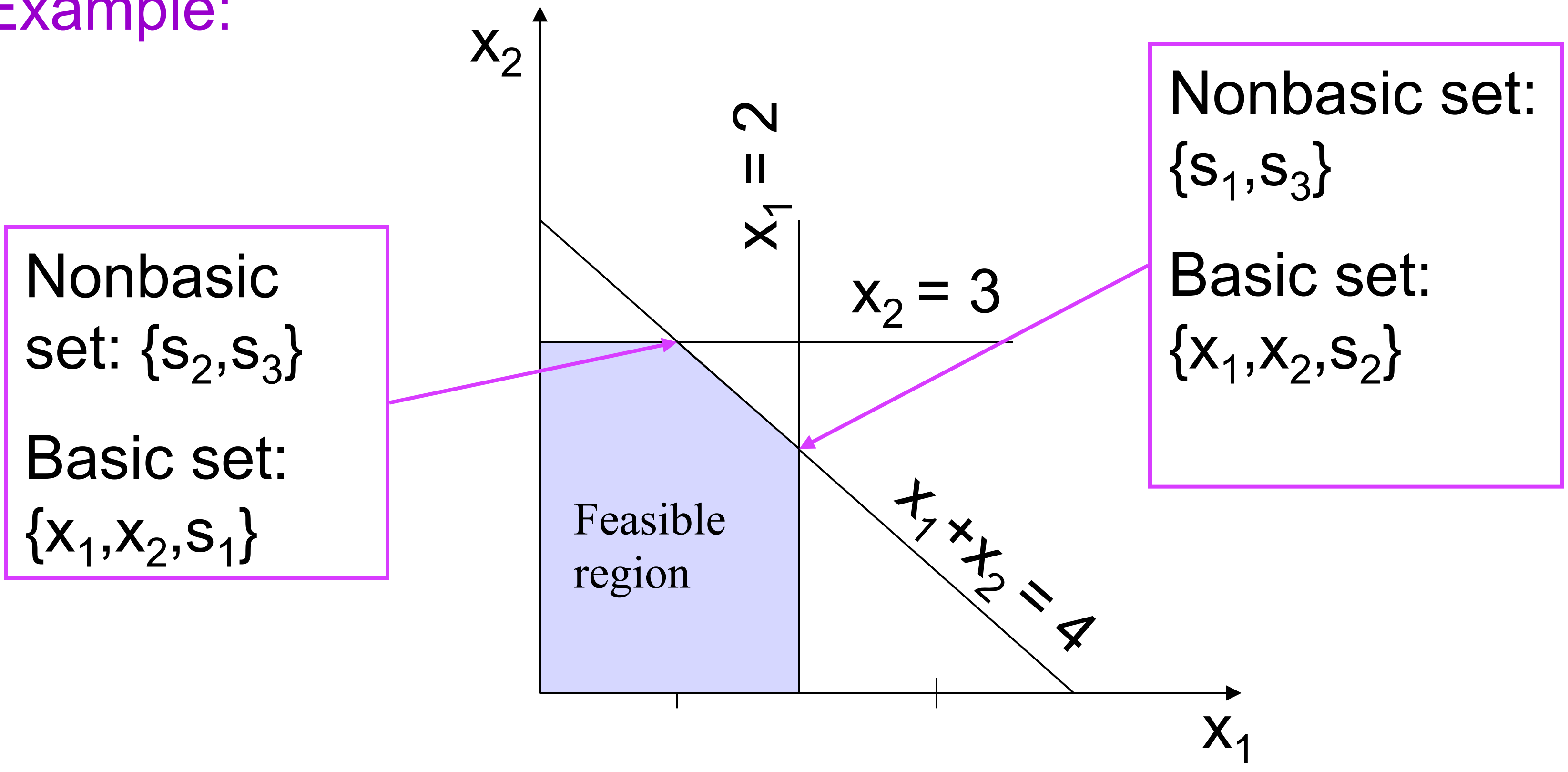
A basis: As simplex proceeds, the variables are always assigned to the basic set or the nonbasic set. The current assignment of the variables is called the basis.

Nonbasic, variables set to zero, corresponding constraint is active.

The Simplex Method

At two adjacent vertices, the nonbasic sets and the basic sets are identical except for one member.

Example:



x_2
$x_1 = 2$
$x_2 = 3$
$x_1 + x_2 = 4$
Feasible region
x_1
Nonbasic set: $\{s_2,s_3\}$
Basic set: $\{x_1,x_2,s_1\}$
Nonbasic set: $\{s_1,s_3\}$
Basic set: $\{x_1,x_2,s_2\}$

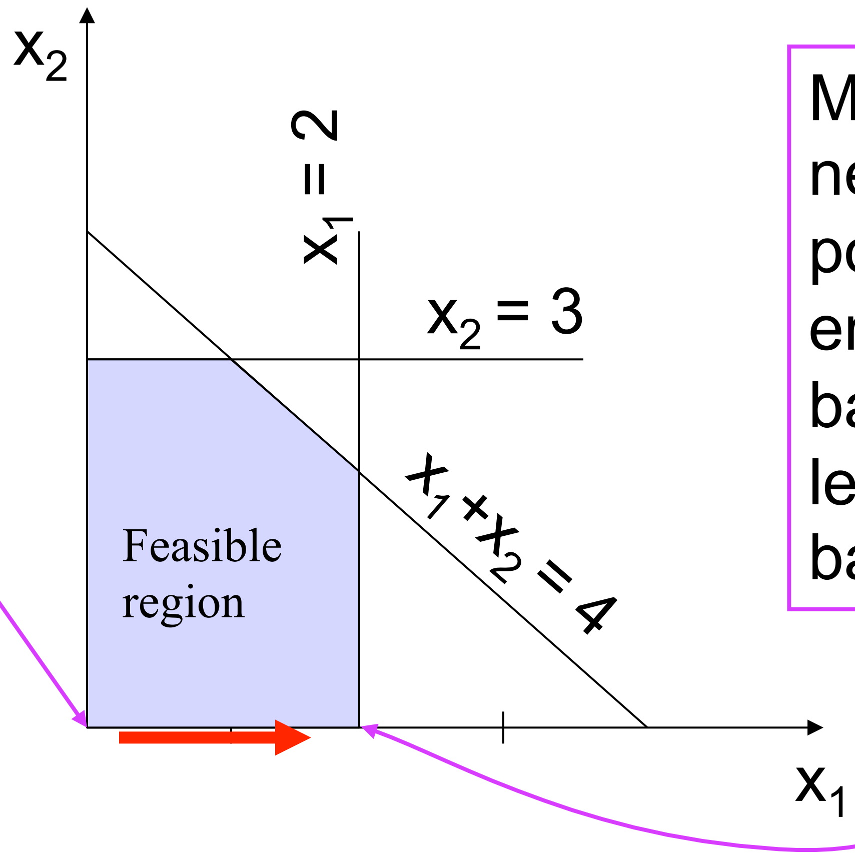
The Simplex Method

The process is therefore based on swapping a pair of variables between the basic and the nonbasic sets.

It is done in a way that improves the objective function.

Example:

Current cornerpoint



Moving to a new corner point: x_1 enters the basic set, s_1 leaves the basic set

The Simplex Method – more details

Phase 1 (start-up): Find initial feasible vertex.

Phase 2 (iterate):

1. Can the current objective value be improved by swapping a basic variable? If not - stop.
2. Select entering basic variable, e.g. via greedy heuristic: choose the nonbasic variable that gives the fastest rate of increase in the objective function value.
3. Select the leaving basic variable by applying the minimum ratio (tightest constraint) test
4. Update equations to reflect new basic feasible solution.

The Simplex Method – example

Objective:

$$\text{Max } z = 15x_1 + 10x_2$$

s.t.

$$x_1 + s_1 = 2$$
$$x_2 + s_2 = 3$$
$$x_1 + x_2 + s_3 = 4$$
$$x_1, x_2, s_1, s_2, s_3 \geq 0$$

Phase 1 (start-up): Initial feasible solution:

$x_1 = 0$, $x_2 = 0$,
$s_1 = 2$, $s_2 = 3$, $s_3 = 4$

Nonbasic set = $\{x_1, x_2\}$
Basic set = $\{s_1, s_2, s_3\}$

Phase 2 (iterate):

1. Are we optimal? NO, z's value can increase by increasing either x_1 or x_2.
2. Select entering basic variable: increasing x_1 improves the objective value faster (15 > 10) (greedy heuristic)

The Simplex Method – example

$$
\begin{array}{l}
s_1 = 2 - x_1 \\
s_2 = 3 - x_2 \\
s_3 = 4 - x_1 + x_2 \\
\hline
z = 15x_1 + 10x_2
\end{array}
$$

and $x_1, x_2, s_1, s_2, s_3 \geq 0$

Phase 1 (start-up): Initial feasible solution:

$x_1 = 0, x_2 = 0,$
$s_1 = 2, s_2 = 3, s_3 = 4$

Nonbasic set = $\{x_1, x_2\}$
Basic set = $\{s_1, s_2, s_3\}$

The “Simplex tableau”: a convenient way to visualize the state of the algorithm (esp. for by-hand examples):

- Constraints above the line
- Objective below the line
- Basic vars & z in LHS; constants and non-basic in RHS
- Constants trivially = basic var values when nonbasic=0

The Simplex Method – example

$$\begin{array}{l} s_1 = 2 - x_1 \\ s_2 = 3 \qquad - x_2 \\ s_3 = 4 - x_1 + x_2 \\ \hline z = 15x_1 + 10x_2 \end{array}$$

and $x_1, x_2, s_1, s_2, s_3 \geq 0$

Phase 1 (start-up): Initial feasible solution:

$x_1 = 0, x_2 = 0,$
$s_1 = 2, s_2 = 3, s_3 = 4$

Nonbasic set = $\{x_1, x_2\}$
Basic set = $\{s_1, s_2, s_3\}$

Phase 2 (iterate):

1. Are we optimal? NO, z's value can increase by increasing either x_1 or x_2.
2. Select entering basic variable: increasing x_1 improves the objective value faster (15 > 10) (greedy heuristic)

The Simplex Method – example

$$\begin{aligned} s_1 &= 2 - x_1 \\ s_2 &= 3 - x_2 \\ s_3 &= 4 - x_1 + x_2 \\ \hline z &= 15x_1 + 10x_2 \end{aligned}$$

and $x_1, x_2, s_1, s_2, s_3 \geq 0$

Phase 1 (start-up): Initial feasible solution:

$x_1 = 0, x_2 = 0,$
$s_1 = 2, s_2 = 3, s_3 = 4$

Nonbasic set = $\{x_1, x_2\}$
Basic set = $\{s_1, s_2, s_3\}$

Phase 2:

3. Select leaving variable:

eqn 1: could raise x_1 to 2 without violating $s_1 \geq 0$ $\leftarrow$ TIGHTEST
eqn 2: no constraint
eqn 3: with $x_2 = s_3 = 0$, could raise x_1 to 4

The Simplex Method – example

$$\begin{aligned} s_1 &= 2 - x_1 \\ s_2 &= 3 - x_2 \\ s_3 &= 4 - x_1 + x_2 \\ \hline z &= 15x_1 + 10x_2 \end{aligned}$$

and $x_1, x_2, s_1, s_2, s_3 \geq 0$

$$\begin{aligned} x_1 &= 2 - s_1 \\ s_2 &= 3 - x_2 \\ s_3 &= 2 + s_1 + x_2 \\ \hline z &= 30 - 15s_1 + 10x_2 \end{aligned}$$

and $x_1, x_2, s_1, s_2, s_3 \geq 0$

Phase 2:

Rewrite eqns, substituting $2-s_1$ for x_1

3. Select leaving variable:

eqn 1: could raise x_1 to 2 without violating $s_1 \geq 0$ ← TIGHTEST

eqn 2: no constraint

eqn 3: with $x_2 = s_3 = 0$, could raise x_1 to 4

The Simplex Method – example

$$\begin{aligned} x_1 &= 2 - s_1 \\ s_2 &= 3 - x_2 \\ s_3 &= 2 + s_1 + x_2 \\ \hline z &= 30 - 15s_1 + 10x_2 \end{aligned}$$

and $x_1, x_2, s_1, s_2, s_3 \geq 0$

The new vertex:

$z=30$

Nonbasic set = $\{s_1, x_2\}$ (rhs)

Basic set = $\{x_1 , s_2, s_3\}$ (lhs)

Phase 2:

End of iteration 1.

The Simplex Method – example

Phase 2 (iteration 2):

1. Are we optimal? NO, z's value can increase by increasing the values of x_2.
2. Select entering basic variable: the only candidate is x_2.
3. Select the leaving basic variables: The minimum ratio test. For x_1 the ratio is infinite, for s_2 the ratio is 3/1=3, for s_3 the ratio is 2/1=2. s_3 has the smallest ratio.
4. Update the equations to reflect the new basic feasible solution: $x_1=2$, $x_2=2$, $s_1=0$, $s_2=1$, $s_3=0$. $z=50$. Nonbasic set = $\{s_1, s_3\}$, Basic set = $\{x_1, s_2, x_3\}$,

End of iteration 2.

The Simplex Method – example

Phase 2 (iteration 3):

1. Are we optimal? YES, z's value can not increase by increasing the value of either s_1 or s_3.

End of example.

Remarks:

The Simplex tableau gives a quick way to select the variable to enter and the variable to leave the basis.

In case of a tie, both directions are ok, there is no sure-fire heuristic to determine which will terminate first.

Simplex Algorithm: An Example in 3D

Maximize $5x + 4y + 3z$

subject to

$$\begin{aligned} 2x + 3y + z &\le 5 \\ 4x + y + 2z &\le 11 \\ 3x + 4y + 2z &\le 8 \\ x, y, z &\ge 0 \end{aligned}$$

Simplex Algorithm: A 3D Example

Maximize $5x + 4y + 3z$

subject to $2x + 3y + z \le 5$

$4x + y + 2z \le 11$

$3x + 4y + 2z \le 8$

$x,y,z \ge 0.$

Step 0: convert inequalities into equalities by introducing slack variables a,b,c.

Define: $a = 5-2x-3y-z \Rightarrow a \ge 0$

$b = 11-4x-y-2z \Rightarrow b \ge 0$

$c = 8-3x-4y-2z \Rightarrow c \ge 0$

$F = 5x+4y+3z$, objective function

Example of Simplex Method, continued.

Step 1: Find initial feasible solution:

$x=0, y=0, z=0 \quad \Rightarrow \quad a=5,\ b=11,\ c=8 \ \Rightarrow F=0.$

Step 2: Find feasible solution with higher value of F

For example, can increase x to get $F=5x$.

How much can we increase x?

$a = 5-2x-3y-z \geq 0 \quad \Rightarrow \quad x \leq 5/2$ **most stringent**

$b = 11-4x-y-2z \geq 0 \quad \Rightarrow \quad x \leq 11/4$

$c = 8-3x-4y-2z \geq 0 \quad \Rightarrow \quad x \leq 8/3$

$\Rightarrow$ increase x to 5/2 $\Rightarrow F= 25/2,\ a=0,\ b=1,\ c=1/2$

Example of Simplex Method, continued.

Want to keep doing this, need to get back into state where x,b,c on l.h.s. of equations.

$a = 5-2x-3y-z \Rightarrow x = 5/2 - 3/2\, y - 1/2\, z - 1/2\, a$ (*)

Substituting (*) into other equations:

$b = 11-4x-y-2z \geq 0 \Rightarrow b = 1 + 5y + 2a$

$c = 8-3x-4y-2z \geq 0 \Rightarrow c = 1/2 + 1/2\, y - 1/2\, z + 3/2\, a$

$F = 5x+4y+3z \Rightarrow F = 25/2 - 7/2\, y + 1/2\, z - 5/2\, a$

In order to increase F again, should increase z

Simplex Example, continued.

How much can we increase **z**?

$x = 5/2 - 3/2\, y - 1/2\, z - 1/2\, a \quad \Rightarrow z \le 5$

$b = 1 + 5y + 2a \quad \Rightarrow$ no restriction

$c = 1/2 + 1/2\, y - 1/2\, z + 3/2\, a \quad \Rightarrow z \le 1$ **most stringent (^)**

Setting **z** = **1** yields

$x=2,\ y=0,\ z=1,\ a=0,\ b = 1,\ c = 0.$

$F = 25/2 - 7/2\, y + 1/2\, z - 5/2\, a \quad \Rightarrow F = 13.$

Again, construct system of equations.

From (^) $\quad z = 1 + y + 3a - 2c.$

Simplex Example, continued.

Substituting back into other equations:

$z = 1 + y + 3a - 2c.$

$x = 5/2 - 3/2\, y - 1/2\, z - 1/2\, a \Rightarrow x = 2 - 2y - 2a + c$

$b = 1 + 5y + 2a \Rightarrow b = 1 + 5y + 2a$

$F = 25/2 - 7/2\, y + 1/2\, z - 5/2\, a \Rightarrow F = 13 - 3y - a - c$

And we’re done.

Simplex – Loose Ends

- Finding an initial feasible point
- Unboundedness
- Infeasibility
- Degeneracy, cycling, and pivot choice

Initial Feasible Point

- Problem: Maximize $\mathbf{c}^T\mathbf{x}$ subject to $A\mathbf{x} \leq \mathbf{b}$, $\mathbf{x} \geq \mathbf{0}$
- Equational form:
  - add slack variables so $A\mathbf{x} = \mathbf{b}$ (new A = old A|I, x longer)
- Initial point:
  - Make $\mathbf{b} \geq \mathbf{0}$ (multiply rows by -1 as needed) Solve auxiliary LP:
    - maximize $-\mathbf{1}^T\mathbf{y}$ s.t. $A\mathbf{x} + \mathbf{y} = \mathbf{b}$, $\mathbf{x},\mathbf{y} \geq \mathbf{0}$
    - $\mathbf{x} = \mathbf{0}$, $\mathbf{y} = \mathbf{b}$ is feasible start for aux LP
    - Solution is $\mathbf{y} = \mathbf{0}$ iff original is feasible
    - And the resulting $\mathbf{x}$ is feasible start for original LP

Degeneracy, Cycling, Pivot Rules

- Some (of many) *pivot rules*:
  - Largest coefficient (Danzig)
  - Largest increase
  - Bland’s rule (entering/exiting vars w/ min index)
  - Random edge
  - Steepest edge – seems to be best in practice
- In n dimensions, n hyperplanes can define a point. If more intersect at a vertex, the LP is *degenerate*, and most of the above rules may *stall* there, i.e., “move” to same vertex (with different basis); some may *cycle* there: infinite loop

DUALITY

A Central Result of LP Theory: Duality Theorem

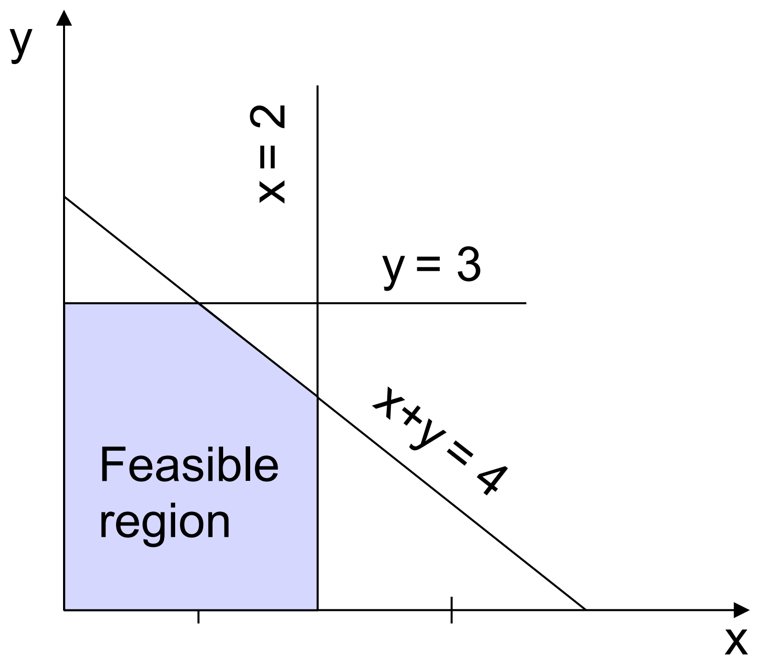
- Every linear program has a dual
- If the original is a maximization, the dual is a minimization and vice versa
- Solution of one leads to solution of other

Primal: Maximize $\mathbf{c}^T\mathbf{x}$ subject to $A\mathbf{x} \le \mathbf{b}$, $\mathbf{x} \ge 0$

Dual: Minimize $\mathbf{b}^T\mathbf{y}$ subject to $A^T\mathbf{y} \ge \mathbf{c}$, $\mathbf{y} \ge 0$

If one has optimal solution so does the other, and their values are the same.

Toy Factory



Maximize $z = 15x+10y$

s.t. $x, y \geq 0$ & $x \leq 2$ [1]

$y \leq 3$ [2]

$x+y \leq 4$ [3]

So:

$z = 15x+10y \leq 15(x+y) \leq 15*4 = 60$ (15*[3])

$z = 15x+10y \leq 14(x+y)+x \leq 14*4+1*2 = 58$ (14*[3]+1*[1])

$z = 15x+10y \leq 10(x+y)+5x \leq 10*4+5*2 = 50$ (10*[3]+5*[1])

… etc …

In general–positive linear combination of primal constraints that bound its objective coefficients also bound its value. The dual LP finds exactly the *optimal* such combination.

(Dual constraints insure that every feasible point in the dual is a combination that bounds primal objective coefficients; dual LP minimizes over them all.)

Primal: Maximize $c^T x$ subject to $Ax \leq b,\ x \geq 0$

Dual: Minimize $b^T y$ subject to $A^T y \geq c,\ y \geq 0$

- In the primal, **c** is cost function and **b** was in the constraint. In the dual, reversed.
- Inequality sign is changed and minimization turns to maximization.

Primal:

maximize $2x + 3y$

s.t

$$x + 2y \leq 4,$$
$$2x + 5y \leq 1,$$
$$x - 3y \leq 2,$$
$$x \geq 0,\ y \geq 0$$

Dual:

minimize $4p + q + 2r$ s.t

$$p + 2q + r \geq 2,$$
$$2p + 5q - 3r \geq 3,$$
$$p,q,r \geq 0$$

In $p*1^{st}+q*2^{nd}+r*3^{rd}$, coef of x is ≥ 2, coef of y is ≥ 3, etc. etc.

Proof of Weak Duality

- Suppose that
  - $\mathbf{x}$ satisfies $A\mathbf{x} \le \mathbf{b}$, $\mathbf{x} \ge 0$
  - $\mathbf{y}$ satisfies $A^T\mathbf{y} \ge \mathbf{c}$, $\mathbf{y} \ge 0$
- Then
  - $\mathbf{c}^T\mathbf{x} \le (A^T\mathbf{y})^T \mathbf{x}$ since $\mathbf{x} \ge 0$ and $A^T\mathbf{y} \ge \mathbf{c}$
    $= \mathbf{y}^T A \mathbf{x}$ by definition
    $\le \mathbf{y}^T\mathbf{b}$ since $\mathbf{y} \ge 0$ and $A\mathbf{x} \le \mathbf{b}$
    $= \mathbf{b}^T\mathbf{y}$ by definition
- This says that any feasible solution to the primal (maximization problem) has an objective function value at most that of any feasible solution of the dual (minimization) problem.
- *Strong duality:* says the optima of the two are equal

Simple Example

- Diet problem: minimize $2x + 3y$

  subject to $x + 2y \geq 4$,

  $x \geq 0,\ y \geq 0$
- Dual problem: maximize $4p$

  subject to $p \leq 2$,

  $2p \leq 3$,

  $p \geq 0$
- Dual: the problem faced by a druggist who sells synthetic protein, trying to compete with peanut butter and steak

Simple Example

- The druggist wants to maximize the price p of one unit of protein, subject to constraints:
  - synthetic protein must not cost more than protein available in foods.
  - price must be non-negative or he won't sell any
  - revenue to druggist will be 4p
- Solution: $p \leq 3/2 \rightarrow$ objective value = $4p = 6$
- Not coincidence that it's equal the minimal cost in original problem.

What's going on?

- Notice: feasible sets completely different for primal and dual, but nonetheless an important relation between them.
- Duality theorem says that in the competition between the grocer and the druggist the result is always a tie.
- Optimal solution to primal tells purchaser what to do.
- Optimal solution to dual fixes the natural prices at which economy should run.
- The food x and synthetic prices y are optimal when
  - grocer sells zero of any food that is priced above its synthetic equivalent.
  - druggist charges 0 for any synthetic that is oversupplied in the diet.

Duality Theorem

Druggist’s max revenue = Purchasers min cost

Practical Use of Duality:

- Sometimes simplex algorithm (or other algorithms) will run faster on the dual than on the primal.
- Can be used to bound how far you are from optimal solution.
- Important implications for economists.

Example: Max Flow

Variables: $\mathbf{f_{uv}}$ - the flow on edge $e=(u,v)$.

Max $\Sigma_u \mathbf{f_{su}}$	Dual variables
s.t.	
$\mathbf{f_{uv} \le c_{uv}}, \quad \forall (u,v) \in E$	$\mathbf{h_{uv}}$
$\Sigma_u \mathbf{f_{uv}} - \Sigma_w \mathbf{f_{vw}} = 0, \quad \forall v \in V-\{s,t\}$	$\mathbf{g_v}$
$\mathbf{f_{uv} \ge 0}, \quad \forall (u,v) \in E$	

Dual to: Max Flow

Variables: g_v, h_{uv}

Min $\Sigma_{uv}\, c_{uv}\, h_{uv}$

s.t.

$$h_{sv} + g_v \geq 1, \quad \forall v \in V\text{-}s$$

$$h_{uv} + g_v - g_u \geq 0, \quad \forall u,v \in V\text{-}\{s,t\}$$

$$h_{ut} - g_u \geq 0, \quad \forall u \in V\text{-}t$$

$$h_{uv} \geq 0, \quad \forall (u,v) \in E$$

WLOG at minimum
$h_{uv} = \max(g_u - g_{v,} 0)$
for $u,v \neq s,t$
$h_{ut} = \max(g_u, 0)$
$h_{sv} = \max(1 - g_v, 0)$

Dual Solution: Given **st**-cut **(S,T)** with $S = s \cup A$
Set $g_v = 1$ for $v \in A$ and $g_v = 0$ otherwise
Set $h_{uv} = 1$ for $u \in A$ and v not in A
Set $h_{uv} = 0$ otherwise
Value is exactly the value of the cut

INTEGER PROGRAMMING

Integer Programming (IP)

- An LP problem with an additional requirement that variables will only get an integral value, maybe from some range.
- 01P – binary integer programming: variables should be assigned only 0 or 1.
- Can model many problems.
- NP-hard to solve!

01P Example: Vertex Cover

Variables: for each $v \in V$, x_v – is v in the cover?

Minimize: $\Sigma_v x_v$

Subject to: $x_v \in \{0,1\}$

$x_i + x_j \geq 1 \quad \forall \{i,j\} \in E$

LP RELAXATION AND APPROXIMATION ALGORITHMS

LP-based approximations

- We don’t know any polynomial-time algorithm for any NP-complete problem
- We know how to solve LP in polynomial time
- We will see that LP can be used to get approximate solutions to some NP-complete problems.

Weighted Vertex Cover

Input: Graph $G=(V,E)$ with non-negative weights w_v on the vertices.

Goal: Find a minimum-cost set of vertices S, such that all the edges are covered. An edge is covered iff at least one of its endpoints is in S.

Recall: Vertex Cover is NP-complete.

The best known approximation factor is

$2 - 1/\mathrm{sqrt}(\log|\mathbf{V}|)$.

Weighted Vertex Cover

Variables: for each $v \in V$, x_v – is v in the cover?

Min $\Sigma_{v \in V} w_v x_v$

s.t.

$x_v + x_u \geq 1, \quad \forall (u,v) \in E$

$x_v \in \{0,1\} \quad \forall v \in V$

The LP Relaxation

This is **not** a linear program: the constraints of type $x_v \in \{0,1\}$ are not linear. We got an LP with integrality constraints on variables – an **integer linear programs (IP)** that is NP-hard to solve.

However, if we replace the constraints $x_v \in \{0,1\}$ by $x_v \geq 0$ and $x_v \leq 1$, we will get a linear program.

The resulting LP is called a **Linear Relaxation** of IP, since we relax the integrality constraints.

LP Relaxation of Weighted Vertex Cover

Min $\Sigma_{v \in V} w_v x_v$

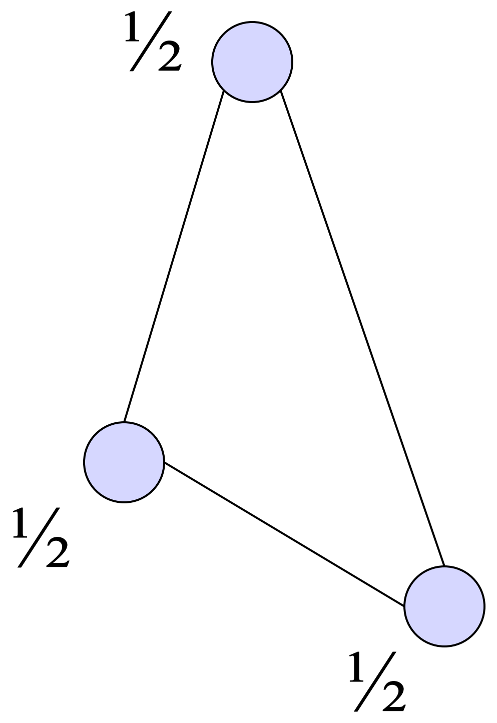
s.t.

$x_v + x_u \geq 1, \ \forall (u,v) \in E$

$x_v \geq 0, \ \forall v \in V$

$x_v \leq 1, \ \forall v \in V$

LP Relaxation of Weighted Vertex Cover - example



Consider the case of a 3-cycle in which all weights are 1.

An optimal VC has cost 2 (any two vertices)

An optimal relaxation has cost 3/2 (for all three vertices x_v=1/2)

The LP and the IP are different problems. Can we still learn something about Integral VC?

Why LP Relaxation Is Useful ?

The optimal value of LP-solution provides a bound on the optimal value of the original optimization problem. OPT_{LP} is always better than OPT_{IP} (why?)

Therefore, if we find an integral solution within a factor r of OPT_{LP}, it is also an r-approximation of the original problem.

It can be done by ‘wise’ rounding.

Approximation of Vertex Cover Using LP-Rounding

1. Solve the LP-Relaxation.

2. Let S be the set of all the vertices v with $x(v) \geq 1/2$. Output S as the solution.

Analysis: The solution is feasible: for each edge $e=(u,v)$, either $x(v) \geq 1/2$ or $x(u) \geq 1/2$

The value of the solution is: $\Sigma_{v \in s} w(v) = \Sigma_{\{v|x(v) \geq 1/2\}} w(v) \leq 2\Sigma_{v \in V} w(v)x(v) = 2OPT_{LP}$

Since $OPT_{LP} \leq OPT_{VC}$, the cost of the solution is $\leq 2OPT_{VC}$.

Linear Programming - Summary

- Of great practical importance to solve linear programs:
  - they model important practical problems
    - production, approximating the solution of inconsistent equations, manufacturing, network design, flow control, resource allocation.
  - solving an LP is often an important component of solving or approximating the solution to an **integer linear programming problem.**
- Can be solved in poly-time, but the simplex algorithm works very well in practice.
- One problem where you really do not want to roll your own code.