CSE 521: Design & Analysis of Algorithms I

Linear Programming

From slides by Paul Beame

Linear Programming

- The process of minimizing a linear objective function subject to a finite number of linear equality and inequality constraints.
- 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."

Linear Programming

- Suggested Readings:
  - Chapter 7 of text by Dasgupta, Papadimitriou, Vazirani (link on web page).
  - "Linear Programming", by Howard Karloff
    - First 34 pages on Simplex Algorithm available through Google books preview
  - "Linear Programming", by Vasek Chvatal
  - "Understanding and Using Linear Programming", by Jiri Matousek and Bernd Gartner

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:

$x + 2y \geq 4$

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

This is an LP- formulation of our problem

An Example: The Diet Problem

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

$x + 2y \geq 4$
$x \geq 0, \; y \geq 0$

- 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*.

5

Linear Program - Definition

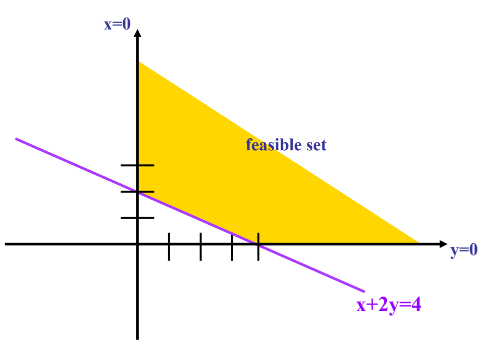
A linear program is a problem with **n** variables $x_1, \ldots, x_n$, that has:

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:
   $a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \leq b_i$ (or $\geq$ or $=$)

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

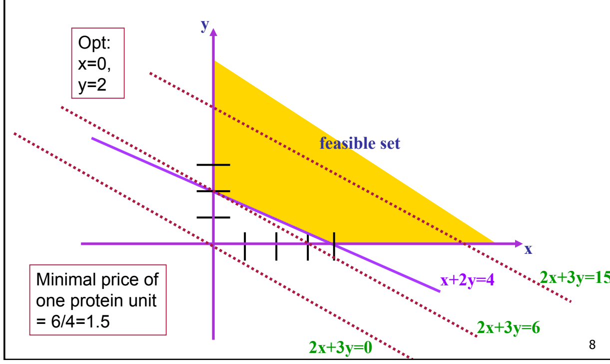
6

Visually…
x= peanut butter, y = steak



7

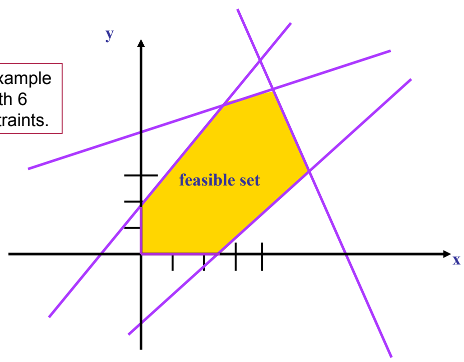
Optimal vector occurs at some corner of the feasible set!



8

Optimal vector occurs at some corner of the feasible set

An Example with 6 constraints.



Standard Form of a Linear Program.

Maximize $c_1x_1 + c_2x_2 + \ldots + c_nx_n$

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

$x_j \geq 0$ $j=1..n$

or

Minimize $b_1y_1 + b_2y_2 + \ldots + b_my_m$

subject to $\Sigma_{1\leq i\leq m}\, a_{ij}y_i \geq c_j$ $j=1...n$

$y_i \geq 0$ $i=1..m$

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.
  - Convex: for any 2 points in feasible set, the line segment joining them is in feasible set.
- The linear cost functions, defines a family of parallel hyperplanes (lines in 2D, planes in 3D, etc.). Want to find one of minimum cost → must occur at corner of feasible set.
  - Corner= can't be expressed as convex combination of 2 or more points in feasible set.

The Feasible Set

- Intersection of a set of half-spaces, called a polyhedron.
- If it's bounded and nonempty, it's a polytope.

There are 3 cases:

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

First two cases very uncommon for real problems in economics, science and engineering.

Solving LPs

- There are several algorithms that solve any linear program optimally.
  - The Simplex method (class of methods, usually very good but worst-case exponential for known methods)
  - The Ellipsoid method (polynomial-time)
  - More
- These algorithms 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.

LP formulation: another example

Bob's bakery sells bagel and muffins.

To bake a dozen bagels Bob needs **5** cups of flour, **2** eggs, and **1** cup of sugar.

To bake a dozen muffins Bob needs **4** cups of flour, **4** eggs and **2** cups of sugar.

Bob can sell bagels at **$10**/dozen and muffins at **$12**/dozen.

Bob has **50** cups of flour, **30** eggs and **20** cups of sugar.

How many bagels and muffins should Bob bake in order to maximize his revenue?

LP formulation: Bob's bakery

	Bagels	Muffins	Avail.
Flour	5	4	50
Eggs	2	4	30
Sugar	1	2	20
Revenue	10	12	

$$A = \begin{pmatrix} 5 & 4 \\ 2 & 4 \\ 1 & 2 \end{pmatrix}$$

$$c^T = \begin{pmatrix} 10 & 12 \end{pmatrix} \quad b = \begin{pmatrix} 50 \\ 30 \\ 20 \end{pmatrix}$$

Maximize $10x_1+12x_2$

s.t. $5x_1+4x_2 \le 50$

$2x_1+4x_2 \le 30$

$x_1+2x_2 \le 20$

$x_1 \ge 0,\ x_2 \ge 0$

Maximize $\mathbf{c}\cdot\mathbf{x}$

s.t. $A\mathbf{x} \le \mathbf{b}$

$\mathbf{x} \ge 0$.

Idea of the Simplex Method

The Toy Factory Problem (TFP):

A toy factory produces dolls and cars.

Danny, a new employee, is hired. He can produce 2 cars and 3 dolls a day. However, the packaging machine can only pack 4 items a day. The company's 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.

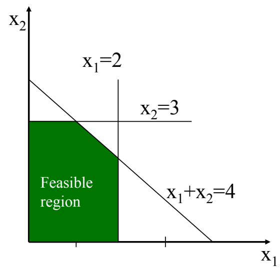
The Toy Factory Problem

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

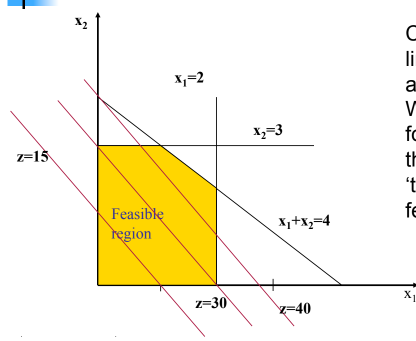
Objective:

Max $z = 15x_1 + 10x_2$

s.t
$x_1 \leq 2$
$x_2 \leq 3$
$x_1 + x_2 \leq 4$
$x_1 \geq 0$
$x_2 \geq 0$



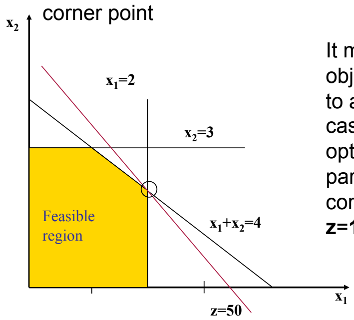
The Toy Factory Problem



Constant profit lines – They are always parallel. We are looking for the best one that still 'touches' the feasible region.

Important Observations:

1. An optimum solution to the LP is always at a corner point

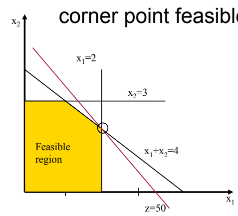


It might be that the objective line is parallel to a constraint. In this case there are many optimum points, in particular at the relevant corner points (consider $z = 15x_1 + 15x_2$).

Important Observations:

2. If a corner point feasible solution has an objective function value that is better than or equal to all its adjacent corner point feasible solutions then it is optimal.

3. There is a finite number of corner point feasible solutions.



The Simplex method: Travel along the corner points till a local maximum.

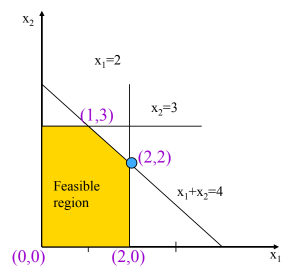
The Simplex Method

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

Phase 2 (iterate): Repeatedly move to a better adjacent corner point feasible solution until no further better adjacent corner point feasible solution can be found. The final corner point defines 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!



A Central Result of LP Theory: Duality Theorem

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

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

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

Primal: **Maximize** c^Tx **subject to** $Ax \le b$, $x \ge 0$
Dual: **Minimize** b^Ty **subject to** $A^Ty \ge c$, $y \ge 0$

- 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 $x+2y \le 4$,
$2x + 5y \le 1$,
$x - 3y \le 2$,
$x \ge 0$, $y \ge 0$

Dual:
minimize $4p + q + 2r$ s.t
$p+2q + r \ge 2$,
$2p+5q - 3r \ge 3$,
$p,q,r \ge 0$

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

25

Simple Example

- The druggist wants to maximize the price p, 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: $p \leq 3/2$ → objective value = $4p = 6$
- Not coincidence that it's equal the minimal cost in original problem.

26

Proof of Weak Duality

- Suppose that
  - $\mathbf{x}$ satisfies $A\mathbf{x} \leq \mathbf{b}$, $\mathbf{x} \geq 0$
  - $\mathbf{y}$ satisfies $A^T\mathbf{y} \geq \mathbf{c}$, $\mathbf{y} \geq 0$
- Then
  - $\mathbf{c}^T\mathbf{x} \leq (A^T\mathbf{y})^T\,\mathbf{x}$ since $\mathbf{x} \geq 0$ and $A^T\mathbf{y} \geq \mathbf{c}$
    $= \mathbf{y}^T A\,\mathbf{x}$ by definition
    $\leq \mathbf{y}^T\mathbf{b}$ since $\mathbf{y} \geq 0$ and $A\mathbf{x} \leq \mathbf{b}$
    $= \mathbf{b}^T\mathbf{y}$ 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 that the optima of the two are equal

27

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 diet x and vitamin prices y are optimal when
  - grocer sells zero of any food that is priced above its vitamin equivalent.
  - druggist charges 0 for any vitamin that is oversupplied in the diet.

28

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.
- Is used in algorithm design.
- Important implications for economists.

29

Example: Max Flow

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

Max $\Sigma_u f_{su}$

s.t.

$f_{uv} \leq c_{uv}, \quad \forall(u,v)\in E$

$\Sigma_u f_{uv} - \Sigma_w f_{vw} = 0, \quad \forall v\in V-\{s,t\}$

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

30

Ellipsoid Algorithm

- Running time is polynomial but depends on the # of bits **L** needed to represent numbers in **A, b,** and **c**
- **Idea:** Hunt lion in Sahara (under assumption there is at most one)
  - Fence Sahara in
  - Divide in 2 halves with another fence
  - Detect one half that has no lion.
  - Continue recursively on other side until fenced area so small that either find lion or can argue that no lion could fit in there.
- **In ellipsoid algorithm:**
  - Fenced area is ellipsoid
  - Solve feasibility problem: does there exist x s.t. $\mathbf{Ax \geq b}$?

31

Ellipsoid Algorithm

- Running time is polynomial but depends on the # of bits **L** needed to represent numbers in **A, b,** and **c**
  - Like capacity-scaling for network flow but a much bigger polynomial
  - Interior point methods running times also depend on **L**
- Method applies to large class of convex programs
  - Can be efficient for LPs with exponentially many constraints
- Open whether a strongly polynomial-time algorithm exists for LP
  - One where running time has # of operations polynomial in just **m** and **n**

32

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!

33

01P Example: Vertex Cover

Variables: for each $v \in V$, x_v – is v in the cover?
Minimize $\Sigma_v x_v$
Subject to: $x_v \in \{0,1\}$
$x_u + x_v \geq 1 \quad \forall (u,v) \in E$

34

01P Example: Weighted Set Cover

Input: a Collection $S_1, S_2, \ldots, S_n$ of subsets of $\{1,2,3,\ldots,m\}$ a cost p_i for set S_i.
Output: A collection of subsets whose union is $\{1,2,\ldots,m\}$.
Objective: Minimum total cost of selected subsets.

Variables: For each subset, x_i – is subset S_i selected for the cover?
Minimize $\sum_i p_i \cdot x_i$
Subject to: $x_i \in \{0,1\}^n$
$\sum_{j \in Si} x_i \geq 1 \quad \forall j=1..m$

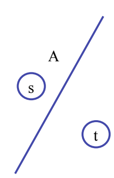
35

01P Example: Shortest Path

Given a directed graph $G(V,E)$, $s,t \in V$ and length p_e for edge e.

Variables: For each edge, x_e – is e in the path?

Minimize $\sum_e p_e x_e$
Subject to: $x_e \in \{0,1\} \quad \forall e \in E$
$\sum_{e \in A} x_e \geq 1 \quad \forall s \rightarrow t$ cut A



36

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.

37

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: Weighted Vertex Cover is NP-complete.

The best known approximation factor is 2- 1/sqrt(log|**V**|).

38

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.

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

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

39

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 program (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 the IP, since we relax the integrality constraints.

40

LP Relaxation of Weighted Vertex Cover

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

s.t.

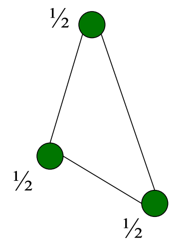
$x_v + x_u \geq 1, \ \forall (u,v)\in E$

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

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

41

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?

42

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.

43

Approximation of Weighted 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: $\Sigma_{v\in s}\, w_v = \Sigma_{\{v|\, x_v \geq 1/2\}}\, w_v$

$\leq 2\, \Sigma_{v\in V}\, w_v x_v = 2\, OPT_{LP}$

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

44

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.