南京大學
NANJING UNIVERSITY

Georg-August-Universität
Göttingen

Computational Methods for Optimization

Linear Programming, Integer Programming, and Dynamic Programming

Wenzhong Li

lwz@nju.edu.cn

- Multivariable optimization problems with constraints are difficult to solve.
- General methods do not exist.
- Linear programming
  - Can be used to solve a simple type of multivariable constrained optimization problem,
    - Both objective function and constraint functions are linear
  - Software packages are widely available.

Linear Programming (LP)

- Standard form
- Maximize

$$y = f(x_1, x_2, \cdots, x_n) = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

- Subject to:

$$a_{11} x_1 + \cdots + a_{1n} x_n \leq b_1$$

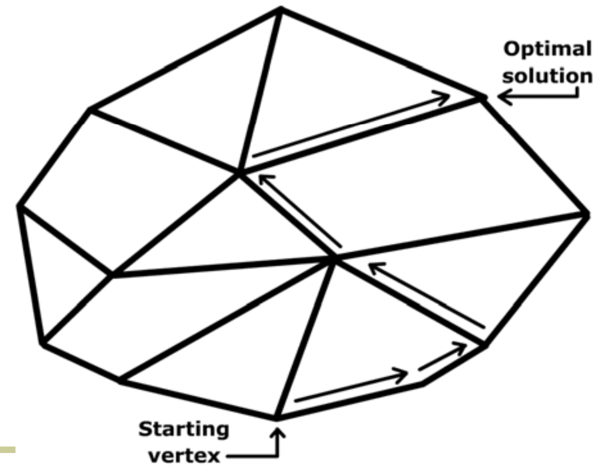
$$\vdots$$

$$a_{m1} x_1 + \cdots + a_{mn} x_n \leq b_m$$

$$x_1 \geq 0, \cdots, x_n \geq 0$$

Simplex Method

- Feasible region: the set of $(x_1, \cdots, x_n)$ that satisfies the constraints
  - (1) The feasible region is a convex polytope
  - (2) The optimal solution must occur at corner points of the feasible region.
- A simple idea: we can check all corner points to find the minimum value (by using computer)
- However, for a problem with n variables and m constraints, the number of corner points is $\binom{n+m}{m} = \frac{(n+m)!}{n!m!}$
  - For example, if n=50, m=100, $\binom{150}{100} = \frac{150!}{100!50!} \approx 2 * 10^{40}$
- The simplex method walks along edges of the polytope to extreme points with lower and lower objective values, until the minimum value is reached.
- Software tools are available
  - Matlab, Mathematica, Maple, LINDO



Example: Crops Planting Problem

- A family farm plants corn, wheat and oats.

  Area: 625 acres

  Water: 1000 acre-ft available

  Labor: up to 300 hours per week

Requirements per acre	Corn	Wheat	Oats
Irrigation (acre-ft)	3.0	1.0	1.5
Labor (hrs/week)	0.8	0.2	0.3
Yield ($)	400	200	250

- Find the amount of each crop to be planted for maximum profit.

- Variables:

$$x_1, x_2, x_3 = acres\ of\ corn, wheat, oats$$
$$w = irrigation\ required$$
$$l = labor\ required$$
$$t = total\ acres$$
$$y = total\ profit$$

- Assumptions

$$w = 3.0x_1 + 1.0x_2 + 1.5x_3$$
$$l = 0.8x_1 + 0.2x_2 + 0.3x_3$$
$$t = x_1 + x_2 + x_3$$
$$y = 400x_1 + 200_{x2} + 250_{x3}$$
$$w \leq 1000, \quad l \leq 300, \quad t \leq 625$$
$$x_1 \geq 0; x_2 \geq 0; x_3 \geq 0$$

- Objective:

$$Maximize\ y$$

- Apply the linear programming method.

- Formulate the linear programming model in the standard form

$$
\begin{aligned}
&Maximize: \;\; y = 400x_1 + 200x_2 + 250x_3 \\
&Subject\ to: \\
&3.0x_1 + 1.0x_2 + 1.5x_3 \leq 1000 \\
&0.8x_1 + 0.2x_2 + 0.3x_3 \leq 300 \\
&x_1 + x_2 + x_3 \leq 625 \\
&x_1 \geq 0, x_2 \geq 0, x_3 \geq 0
\end{aligned}
$$

- We use computer program to solve the model,

$$x_1 = 187.5, x_2 = 437.5, x_3 = 0$$

- Answer:
- The optimal solution is to plant 187.5 acres of corn, 437.5 acres of wheat, and no oats. This should yield $162,500 profit.

- Discussion: what if the variables are required to be integers?

$$x_1 = 187.5, x_2 = 437.5, x_3 = 0$$

- According to previous result, will it be possible to use

$$x_1 = [187.5] = 188, x_2 = [437.5] = 438, x_3 = 0$$

as the optimal solution?

Integer Programming (IP)

- Integer programming
  - A kind of linear programming problem with constraints of decision variables taking integer values
  - Both decision variables and objective function must be linear
- Solution: branch-and-bound method
  - (1) LP relaxation
  - (2) Branch into two subproblems
  - (3) Bound the solution, remove the lower branch
  - (4) repeat until an integer solution is achieved, or no solution is returned.
- Software packages are also available for IP

- Example: solve the following IP problem.

$$\max\ z = 5x_1 + 8x_2;$$
$$x_1 + x_2 \leq 6;$$
$$5x_1 + 9x_2 \leq 45;$$
$$x_1 \geq 0, x_2 \geq 0, x_1\ and\ x_2\ are\ integers;$$

$$LP0: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, (x_1 \in I,\ x_2 \in I); \\ Solution: x_1 = 2.25, x_2 = 3.75, z = 41.25 \end{array}$$

$$Branch$$
$$Bound: 0 \le z^* \le 41.25$$

$$LP1: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, x_2 \ge 4, (x_1 \in I,\ x_2 \in I); \\ Solution: x_1 = 1.8, x_2 = 4, z = 41 \end{array}$$

$$LP2: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, x_2 \le 3, (x_1 \in I,\ x_2 \in I); \\ Solution: x_1 = 3, x_2 = 3, z = 39 \end{array}$$

$$Branch$$
$$Bound: 39 \le z^* \le 41$$

$$LP3: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, x_2 \ge 4, x_1 \le 1(x_1 \in I,\ x_2 \in I); \\ Solution: x_1 = 1, x_2 = 4.44, z = 40.56 \end{array}$$

$$LP4: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, x_2 \ge 4, x_1 \ge 2, (x_1 \in I,\ x_2 \in I); \\ Solution: null \end{array}$$

$$Branch$$
$$Bound: 39 \le z^* \le 40.56$$

$$LP5: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, x_2 \ge 4, x_1 \le 1, x_2 \le 4, (x_1 \in I,\ x_2 \in I); \\ Solution: x_1 = 1, x_2 = 4, z = 37 \end{array}$$

$$LP6: \begin{array}{l} max\ z = 5x_1 + 8x_2; \\ x_1 + x_2 \le 6; \\ 5x_1 + 9x_2 \le 45; \\ x_1 \ge 0, x_2 \ge 0, x_2 \ge 4, x_1 \le 1, x_2 \ge 5(x_1 \in I,\ x_2 \in I); \\ Solution: x_1 = 0, x_2 = 5, z = 40 \end{array}$$

Reconsider the Crops Planting Problem

- A family farm plants corn, wheat and oats.
  Area: 625 acres
  Water: 1000 acre-ft available
  Labor: up to 300 hours per week

Requirements per acre	Corn	Wheat	Oats
Irrigation (acre-ft)	3.0	1.0	1.5
Labor (hrs/week)	0.8	0.2	0.3
Yield ($)	400	200	250

- The farm with 625 acres is divided into 6 sub areas:
  5 plots of 120 acres and 1 plot of 25 acres
- Each subarea only plant one crop
- Find the amount of each crop to be planted for maximum profit.

- Variables

  x_1, x_2,x_3 = number of 120-acre plots of corn/wheat/oats planted
  x_4,x_5,x_6 = number of 25-acre plots of corn/wheat/oats planted
  w = irrigation required
  l = labor required
  t = total acreage planted
  y= total profit

- Assumptions

  $w=120(3.0x_1+1.0x_2+1.5x_3)+25(3.0x_4+1.0x_5+1.5x_6)$
  $l=120(0.8x_1+0.2x_2+0.3x_3)+25(0.8x_4+0.2x_5+0.3x_6)$
  $t=120(x_1+x_2+x_3)+25(x_4+x_5+x_6)$
  $y=120(400x_1+200x_2+250x_3)+25(400x_4+200x_5+250x_6)$
  $w \le 1000$
  $t \le 625$
  $x_1+x_2+x_3 \le 5$
  $x_4+x_5+x_6 \le 1$
  $x_1,\ldots,x_6$ are nonnegative integers

- Objective: Maximize y

- Use integer programming method

- Maximize:
  $y=48000x_1+24000x_2+30000x_3+10000x_4+5000x_5+6250x_6$
- S.t.

$$
\begin{aligned}
375x_1 + 125x_2 + 187.5x_3 + 75x_4 + 25x_5 + 37.5x_6 &\leq 1000 \\
100x_1 + 25x_2 + 37.5x_3 + 20x_4 + 5x_5 + 7.5x_6 &\leq 300 \\
x_1 + x_2 + x_3 &\leq 5 \\
x_4 + x_5 + x_6 &\leq 1
\end{aligned}
$$

- Solve the problem use LINDO

$x_1=0,\ x_2=0,\ x_3=5,\ x_4=0,\ x_5=0,\ x_6=1$

$y=156250$

- Answer:
- The best plan is to plant oats in every plot, which results in total profit of $156250 for the season.

Dynamic Programming (DP)

- DP is introduced to solve a special case of IP problem, which is called (0-1) integer programming problem.
- Example:
- ***0-1 knapsack problem***
  - A thief robbing a store finds n items; the ith item is worth v_i dollars and weighs w_i pounds, where v_i and w_i are integers. He wants to take as valuable a load as possible, but he can carry at most W pounds in his knapsack for some integer W.
  - Which items should he take?

- 0-1 knapsack problem formulation

$$x_i \in \{0,1\} \ indicates \ that \ the \ ith \ item \ is \ taken$$
$$v_i : value \ of \ item \ i$$
$$w_i : weight \ of \ item \ i$$
$$W : weight \ limit$$
$$Max \ C = \sum_{i=1}^{n} x_i v_i$$
$$s.t. \sum_{i=1}^{n} x_i w_i \leq W, (v_i, w_i, W \in I)$$

- Why DP?
  - Natural Recursion may be expensive.
  - For example, the Fibonacci: F(n)=F(n-1)+F(n-2)
    - Recursive implementation memoryless :

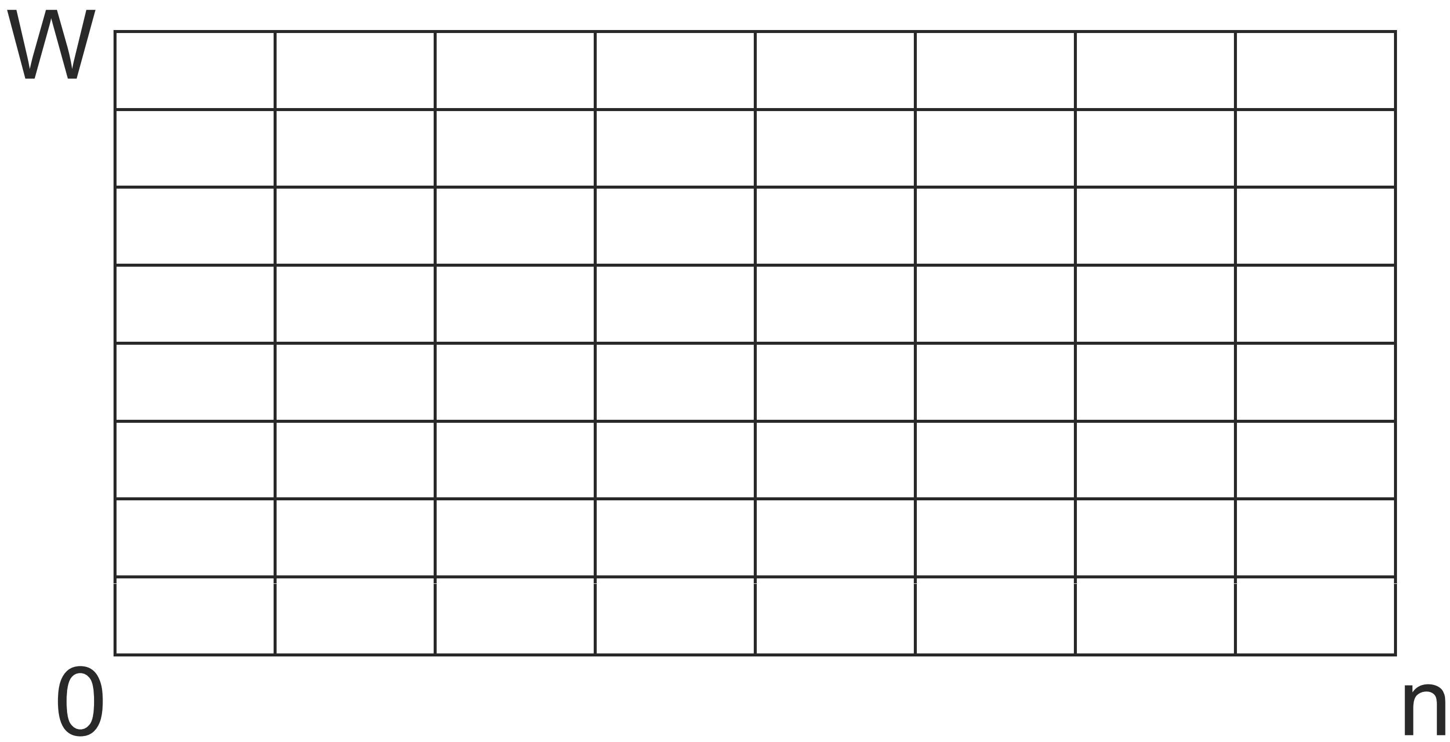
      time= $O((\frac{1+\sqrt{5}}{2})^n)$
    - DP implementation: memorial

      time=O(n), with extra space O(n) or O(1)
- Basic Idea of DP
  - Computing each subproblem **only once**
    - Find a reverse topological order for the subproblem graph
    - Scheduling the subproblems according to the reverse topological order
    - Record the subproblem solutions for later use

- Step 1: divide into subproblems (DP equation)
  - Define c[i,w] to be the value of the solution for items 1,2,…,i, and maximum weight w.

$$c[i,w] = \begin{cases} 0 & \text{if } i = 0 \text{ or } w = 0 , \\ c[i-1,w] & \text{if } w_i > w , \\ \max(v_i + c[i-1, w-w_i], c[i-1,w]) & \text{if } i > 0 \text{ and } w \geq w_i . \end{cases}$$

- Step 2: Scheduling the subproblems according to the reverse topological order



- Step 3: non-recursive implementation

$\text{DYNAMIC-0-1-KNAPSACK}(v, w, n, W)$

for $w \leftarrow 0$ **to** W
  do $c[0, w] \leftarrow 0$
for $i \leftarrow 1$ **to** n
  do $c[i, 0] \leftarrow 0$
    for $w \leftarrow 1$ **to** W
      do if $w_i \leq w$
        then if $v_i + c[i-1, w-w_i] > c[i-1, w]$
          then $c[i, w] \leftarrow v_i + c[i-1, w-w_i]$
          else $c[i, w] \leftarrow c[i-1, w]$
        else $c[i, w] \leftarrow c[i-1, w]$

Reconsider the Crops Planting Problem

- A family farm plants corn, wheat and oats.
  Area: 625 acres
  Water: 1000 acre-ft available
  Labor: up to 300 hours per week

Requirements per acre	Corn	Wheat	~~Oats~~
Irrigation (acre-ft)	3.0	1.0	~~1.5~~
Labor (hrs/week)	0.8	0.2	~~0.3~~
Yield ($)	400	200	~~250~~

- The farm with 625 acres is divided into 6 sub areas
  5 plots of 120 acres and 1 plot of 25 acres
- Each subarea only plant one crop
- Find the amount of each crop to be planted for maximum profit.

- Can we formulate it as a 0-1 integer programming problem?
- Can we solve the problem use dynamic programming?

Example: Multicasting in Delay Tolerant Networks (Mobihoc 2009)

- Background: DTN
- Data source S send data items $d_1, ..., d_n$ with size $s_1, ..., s_n$ to destination sets $\mathbb{D}_1, ..., \mathbb{D}_n$
- S select ralays among nodes $R_1, ..., R_m$ with buffer size $B_1, ..., B_m$
- The objective is to minimized the number of relays used.

- The relay selection problem is formulated as a knapsack problem

$$\min \left| \{j | \sum_{i=1}^{n} x_{ij} > 0\} \right|$$

$$s.t. \sum_{i=1}^{n} x_{ij} s_i \leq B_j, \text{for } j = 1, ..., m$$

$$\frac{1}{|\mathbb{D}_i|} \sum_{k \in \mathbb{D}_i} \prod_{j=1}^{m} (1 - x_{ij} p_{jk}) \leq (1 - p), \text{for } i = 1, ..., n$$

Buffer limit

Delivery ratio guarantees

- where $x_{ij} \in \{0, 1\}$ indicates that data item d_i is placed on relay R_j
- p_{jk} is the success ration for S send data to k via R_j

Homework

- 1. Use Matlab/Maple/Mathematica to solve the crops plant problem.
- 2. Write a program to solve the 0-1 knapsack problem.
- 3. Paper reading:

(1) Wei Gao, Qinghua Li, Bo Zhao and Guohong Cao, Multicasting in Delay Tolerant Networks: A Social Network Perspective, Mobihoc 2009

(2) Juan Alonso and Kevin Fall, A Linear Programming Formulation of Flows over Time with piecewise constant capacity and transit times, tech report. 2003