

# Numerical Optimization

## Lecture 8: Integer Programming: Lagrangian Relaxation

王浩

信息科学与技术学院

**Email: [wanghao1@shanghaitech.edu.cn](mailto:wanghao1@shanghaitech.edu.cn)**

# Contents

## Modeling:

- ◆ Uncapacitated fixed-charge location problem
- ◆ Capacitated fixed-charge location problem

## Solution Methods:

- ◆ Heuristics
- ◆ Lagrange Relaxation
- ◆ Subproblem Solution
- ◆ Lower Bound and Upper Bound
- ◆ Termination

# Facility Location (FL)

- ◆ Facilities: warehouses, retailers, or other physical facilities
- ◆ Determine the number and locations of facilities
- ◆ Extended to many other public-sectors: bus stations, fire courses, telecommunications hubs, satellite orbits, bank account, and other items...
- ◆ Mathematical (subproblems) formulations are common to see in many other MILP
- ◆ Versions of FL: uncapacitated fixed-charge location, capacitated, multi-echelon, multi-product

# Uncapacitated fixed-charge location problem (UFLP)

- ◆ **Problem Statement:** choose facility locations in order to minimize the total cost of building the facilities and transporting goods from facilities to customers
  - two echelons: facility locations (warehouses/distribution centers (DC)) to serve fixed locations (customers)
  - each potential DC location has a fixed cost to open, known
  - transportation cost per unit of product from a DC to a customer, known
  - single product
  - DCs have no capacity restrictions
- ◆ **Objective:** to minimize the fixed cost and transportation costs
- ◆ **Decision Variables:** decide which DC serves each customers
- ◆ **Constraints:** every customer must be served by some open DC

# Formulation

## ◆ Sets:

- $I$  = set of customers
- $J$  = set of potential facility locations

## ◆ Parameters:

- $h_i$  = annual demand of customer  $i \in I$
- $c_{ij}$  = cost to transport one unit of demand from facility  $j \in J$  to customer  $i \in I$
- $f_j$  = fixed (annual) cost to open a facility at site  $j \in J$

## ◆ Decision Variables:

- $x_j$  = 1 if facility  $j$  is opened, 0 otherwise
- $y_{ij}$  = the fraction of customer  $i$ 's demand that is served by facility  $j$

# UFLP

minimize  $\sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} c_{ij} h_i y_{ij} \longrightarrow \text{Optimal value } z^*$

subject to  $\sum_{j \in J} y_{ij} = 1 \quad \forall i \in I$

assignment constraint

$$y_{ij} \leq x_j \quad \forall i \in I, \forall j \in J$$

linking constraint

$$x_j \in \{0, 1\} \quad \forall j \in J$$

$$y_{ij} \geq 0 \quad \forall i \in I, \forall j \in J$$

alternative of linking constraint:  $\sum_{i \in I} y_{ij} \leq |I| x_j, \quad \forall j \in J$

# Solution Methods: heuristics

## ◆ Greedy-add

- Starting with all facilities closed and open the single facility that can serve all customers with the smallest objective
- At each iteration, open the facility that gives the largest decrease in objective
- When open one facility, assign the nearest open facility to each customer
- Stop when no facility can be opened that will decrease the objective

## ◆ Greedy drop

- Starting with all facilities open and close single facility gives the largest decrease in objective

# Solution Methods: heuristics

- ◆ Improvement heuristics: “swap” or “exchange” algorithm
  - Starting with a feasible solution and attempt to improve it
  - At each iteration, find a pair  $(j, k)$  of facilities with  $j$  open and  $k$  closed such that if  $j$  closed and  $k$  opened, the objective would decrease
  - If such a pair can be found, the swap is made and the procedure continues
  - If not, attempt to open a closed facility or close a open facility to decrease the objective
- ◆ All heuristics are proved to perform well in practice, meaning they return good solutions and execute quickly



# Solution Methods: Lagrangian Relaxation

- ◆ Lagrangian relaxation is a standard technique for integer programming
- ◆ Basic idea is to remove a set of constraints to create a problem that's easier to solve than the original
- ◆ To yield a **lower bound** on the optimal value
- ◆ Any feasible solutions provides an **upper bound** on the optimal value

Which constraint to relax?

- How easy the relaxed problem is to solve
- How tight the resulted lower bound is
- How many constraints are being relaxed

# Lagrangian Relaxation (UFLP-LR<sub>λ</sub>)

- ◆ Relaxing the assignment constraint

$$\begin{aligned} \text{minimize} \quad & \sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} c_{ij} h_i y_{ij} + \sum_{i \in I} \lambda_i \left( 1 - \sum_{j \in J} y_{ij} \right) \\ & = \sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} (c_{ij} h_i - \lambda_i) y_{ij} + \sum_{i \in I} \lambda_i \end{aligned}$$

$$\begin{aligned} \text{subject to} \quad & \cancel{\sum_{j \in J} y_{ij} = 1} \quad \cancel{\forall i \in I} \\ & y_{ij} \leq x_j \quad \forall i \in I, \forall j \in J \\ & x_j \in \{0, 1\} \quad \forall j \in J \\ & y_{ij} \geq 0 \quad \forall i \in I, \forall j \in J \end{aligned}$$

Easy to solve!

# Subproblem solution

- ◆ Relaxing the assignment constraint

$$\sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} (c_{ij} h_i - \lambda_i) y_{ij} = \sum_{j \in J} \left[ f_j x_j + \sum_{i \in I} (c_{ij} h_i - \lambda_i) y_{ij} \right]$$

- ◆ If  $h_i c_{ij} - \lambda_i < 0$ , set  $y_{ij} = 1, \forall i \in I$ , the objective decreases

$$\beta_j := \sum_{i \in I} \min\{0, h_i c_{ij} - \lambda_i\}$$

- ◆ This will set  $x_j = 1$ , increase in the objective  $f_j$ ; so would you do this?
- ◆ Let check the overall change:

$$\beta_j + f_j$$

# Subproblem solution

- ◆ If there is a decrease,  $\beta_j + f_j < 0$ , set  $x_j = 1$ ; otherwise, don't do this!
- ◆ So the subproblem solution is given by:

$$x_j = \begin{cases} 1, & \text{if } \beta_j + f_j < 0 \\ 0, & \text{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} 1, & \text{if } x_j = 1 \quad (h_i c_{ij} - \lambda_i < 0) \\ 0, & \text{otherwise} \end{cases}$$

- ◆ We will use  $z_{LR}$  to denote the optimal objective value of the Lagrangian relaxed problem. Then

$$z_{LR}(\lambda) = \sum_{j \in J} \min\{0, \beta_j + f_j\} + \sum_{i \in I} \lambda_i$$

# Lower bound

- ◆ We have now solved the Lagrangian relaxation for given  $\lambda$
- ◆ It turns out that for any  $\lambda$ ,  $z_{LR}(\lambda)$  is always a lower bound on the optimal value

Theorem: For any  $\lambda \in \mathbb{R}^{|I|}$ ,  $z_{LR}(\lambda) \leq z^*$ .

**Proof.** Let  $(x, y)$  be a feasible solution for UFLP. Clearly it is feasible for the Lagrangian relaxed problem.

$$\sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} c_{ij} h_i y_{ij} + \sum_{i \in I} \lambda_i \left( 1 - \sum_{j \in J} y_{ij} \right) = 0$$

# Maximize the lower bound

- ◆ How do we choose the “best”  $\lambda$ ?

$$\text{maximize } z_{LR}(\lambda)$$

$$\max_{\lambda} \left\{ \begin{array}{ll} \text{minimize} & \sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} (c_{ij} h_i - \lambda_i) y_{ij} + \sum_{i \in I} \lambda_i \\ \text{subject to} & y_{ij} \leq x_j \quad \forall i \in I, \forall j \in J \\ & x_j \in \{0, 1\} \quad \forall j \in J \\ & y_{ij} \geq 0 \quad \forall i \in I, \forall j \in J \end{array} \right.$$

- ◆ Let  $\lambda^*$  be the optimal multiplier. Let  $z_{LR} = z_{LR}(\lambda^*)$

# Which is the better bound?

- ◆ Let  $z_{LP}$  be the LP relaxation of UFLP;  $z_{LR}$  and  $z_{LP}$  which one is better?

Theorem:  $z_{LP} \leq z_{LR}$ .

This is a general result for MILP!!!

Primal MILP

minimize

$$cx$$

subject to

$$Ax = b$$

$$Dx \leq e$$

$$x \geq 0 \text{ and integer}$$

minimize

$$cx + \lambda(Ax - b)$$

subject to

$$Dx \leq e$$

$$x \geq 0 \text{ and integer}$$

Lagrangian Relaxation

$$\begin{aligned}
z_{\text{LR}} &= \max_{\lambda} \left\{ \min_x cx + \lambda(Ax - b) \mid Dx \leq e, x \geq 0 \text{ and integer} \right\} \\
&\geq \max_{\lambda} \left\{ \min_x cx + \lambda(Ax - b) \mid Dx \leq e, x \geq 0 \right\} \\
&= \max_{\lambda} \left\{ \min_x (c + \lambda A)x - \lambda b \mid Dx \leq e, x \geq 0 \right\} \\
&= \max_{\lambda} \left\{ \max_{\mu} \mu e - \lambda b \mid \mu D \leq c + \lambda A, \mu \leq 0 \right\} \\
&= \max_{\lambda, \mu} \{ \mu e - \lambda b \mid \mu D \leq c + \lambda A, \mu \leq 0 \} \\
&= \max_{\lambda, \mu} \{ \mu e - \lambda b \mid \mu D - \lambda A \leq c, \mu \leq 0 \} \\
&= \min_y \{ cy \mid Ay = b, Dy \leq e, y \geq 0 \} \\
&= z_{\text{LP}}
\end{aligned}$$



# Which is the better lower bound?

- ◆ Generally,  $z_{LP} < z^*$ , so where in the gap does  $z_{LR}$  fall?
- ◆ An IP is said to have the *integrality property* if its LP relaxation naturally has an all-integer solution (the convex hull)

**Theorem:** Let (P) be an integer program and (P-LR <sub>$\lambda$</sub> ) its Lagrangian subproblem for a given  $\lambda$ . If (P-LR <sub>$\lambda$</sub> ) has the integrality property for all  $\lambda$ , then

$$z_{LP} = z_{LR}$$

- ◆ We know the Lagrangian relaxation of UFLP must have integer solutions

**Corollary:** For the UFLP,  $z_{LP} = z_{LR}$

# Upper bound

- ◆ For UFLP, we have  $z_{LR}(\lambda) \leq z_{LR} = z_{LP} \leq z^* \leq z(x, y)$
- ◆ How can we find a “good” upper bound? Any feasible solution yields an upper bound, but which one is good?
- ◆ Any heuristic method mentioned previously would work.
- ◆ But we would like to convert a solution to (UFLP- $LR_\lambda$ ) into a solution to the original —how?
- ◆ If the solution to (UFLP- $LR_\lambda$ ) is feasible for the original, then we’re lucky!!
- ◆ Remember it’s infeasible since the linking constraint is violated.

$$\exists i \in I, \quad \text{s.t.} \quad \sum_{j \in J} y_{ij} \neq 1$$

- ◆ This means the  $i$ -th customer’s demand is under/over-fulfilled
- ◆ Remedy!

# Updating the multipliers

- ◆ What makes a good value of  $\lambda_i$ ? It should be chosen to entice

$$\sum_{j \in J} y_{ij} = 1 \quad \forall i \in I$$

- ◆ On the objective, it appears

$$\lambda_i \left( 1 - \sum_{j \in J} y_{ij} \right)$$

- If  $\sum_{j \in J} y_{ij} < 1$ , then  $\lambda_i$  is too small; it should be increased
- If  $\sum_{j \in J} y_{ij} > 1$ , then  $\lambda_i$  is too large; it should be decreased
- If  $\sum_{j \in J} y_{ij} = 1$ , then  $\lambda_i$  is just right; it should not be changed

$$\lambda_i^{n+1} \leftarrow \lambda_i^n + t^n \left( 1 - \sum_{j \in J} y_{ij} \right)$$

# Initialization and Termination

- ◆ Initialization: choose  $\lambda$  according to
  - Set  $\lambda_i = 0$ , for all  $i$
  - Set it to some random number
  - Set it according to some other ad-hoc rule
  
- ◆ Terminate if one of the following happens:
  - The upper bound and lower bound are less than some pre-specified tolerance, say 0.1%, either in absolute or percentage terms
  - A certain number of iterations, say 1200, have passed
  - The displacement  $|\lambda^{n+1} - \lambda^n|$  is smaller than pre-specified tolerance

# Branch and Bound

- ◆ If the Lagrangian procedure stops because the 2nd or 3rd criterion, there is no guarantee that the solution found is optimal
- ◆ If we stop and accept the best feasible solution we found without a guarantee of optimality, this means Lagrangian is treated as a heuristic
- ◆ Switching to branch and bound for an accurate solution
- ◆ At each node of the branch-and-bound tree, fixing  $x_j = 1$  or  $0$  for branching. Then solve a Lagrangian relaxation, instead of an LP relaxation

# Relaxation for Inequalities

- ◆ Generally inequality or equality constraints  $c(x) \leq, \geq, =$ 
  - For  $\leq$  constraints,  $\lambda$  is restricted to be *non-positive*.
  - For  $\geq$  constraints,  $\lambda$  is restricted to be *non-negative*.
  - For  $=$  constraints,  $\lambda$  is unrestricted in sign.

$$\lambda_i^{n+1} \leftarrow \lambda_i^n + t^n c(x^n)$$

$$\lambda_i^{n+1} \leftarrow \lambda_i^n + t^n \max\{c(x^n), 0\}$$

$$\lambda_i^{n+1} \leftarrow \lambda_i^n + t^n \min\{c(x^n), 0\}$$

How should we  
update the multipliers?

# Alternate Relaxation

$$\begin{aligned} \text{minimize} \quad & \sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} h_i c_{ij} y_{ij} + \sum_{i \in I} \sum_{j \in J} \lambda_{ij} (x_j - y_{ij}) \\ & = \sum_{j \in J} \left( \sum_{i \in I} \lambda_{ij} + f_j \right) x_j + \sum_{i \in I} \sum_{j \in J} (h_i c_{ij} - \lambda_{ij}) y_{ij} \end{aligned}$$

$$\begin{aligned} \text{subject to} \quad & \sum_{j \in J} y_{ij} = 1 && \forall i \in I \\ & x_j \in \{0, 1\} && \forall j \in J \\ & y_{ij} \geq 0 && \forall i \in I, \forall j \in J \end{aligned}$$

# Alternate Relaxation

$$(x\text{-problem}) \quad \text{minimize} \quad \sum_{j \in J} \left( \sum_{i \in I} \lambda_{ij} + f_j \right) x_j$$

$$\text{subject to} \quad x_j \in \{0, 1\} \quad \forall j \in J$$

$$(y\text{-problem}) \quad \text{minimize} \quad \sum_{i \in I} \sum_{j \in J} (h_i c_{ij} - \lambda_{ij}) y_{ij}$$

$$\text{subject to} \quad \sum_{j \in J} y_{ij} = 1 \quad \forall i \in I$$

$$y_{ij} \geq 0 \quad \forall i \in I, \forall j \in J$$



## Capacitated fixed-charge location problem (CFLP)

$$\text{minimize} \quad \sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} c_{ij} h_i y_{ij}$$

$$\text{subject to} \quad \sum_{j \in J} y_{ij} = 1 \quad \forall i \in I$$

$$\sum_{i \in I} h_i y_{ij} \leq b_j \quad \forall j \in J$$

$$y_{ij} \leq x_j \quad \forall i \in I, \forall j \in J$$

$$x_j \in \{0,1\} \quad \forall j \in J$$

$$y_{ij} \geq 0 \quad \forall i \in I, \forall j \in J$$

# Lagrangian Relaxation (CFLP-LR<sub>λ</sub>)

$$\begin{aligned} \text{minimize} \quad & \sum_{j \in J} f_j x_j + \sum_{i \in I} \sum_{j \in J} c_{ij} h_i y_{ij} + \sum_{i \in I} \lambda_i \left( 1 - \sum_{j \in J} y_{ij} \right) \\ & = \sum_{j \in J} \left[ f_j x_j + \sum_{i \in I} (c_{ij} h_i - \lambda_i) y_{ij} \right] + \sum_{i \in I} \lambda_i \end{aligned}$$

$$\begin{aligned} \text{subject to} \quad & \sum_{i \in I} h_i y_{ij} \leq b_j & \forall j \in J \\ & y_{ij} \leq x_j & \forall i \in I, \forall j \in J \\ & x_j \in \{0, 1\} & \forall j \in J \\ & y_{ij} \geq 0 & \forall i \in I, \forall j \in J \end{aligned}$$