Numerical Optimization

Lecture 9: Case Study: Facility Location

Ye Shi
Email: yeshi@shanghaitech.edu.cn

Slides were made by Prof Hao Wang.

本节内容

建模：

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

求解：

- 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 $$\sum_{j\in J} f_j x_j + \sum_{i\in I}\sum_{j\in J} c_{ij} h_i y_{ij}$$ → 最优值记为z^*

subject to
$$\sum_{j\in J} y_{ij} = 1 \qquad \forall i \in I$$
$$y_{ij} \le x_j \qquad \forall i \in I, \forall j \in J$$
$$x_j \in \{0,1\} \qquad \forall j \in J$$
$$y_{ij} \ge 0 \qquad \forall i \in I, \forall j \in J$$

assignment constraint

linking constraint

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

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$_\lambda$)

- Relaxing the assignment constraint

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

subject to
$$\cancel{\sum_{j\in J} y_{ij} = 1 \qquad \forall i \in I}$$
$$y_{ij} \le x_j \qquad \forall i \in I, \forall j \in J$$
$$x_j \in \{0,1\} \qquad \forall j \in J$$
$$y_{ij} \ge 0 \qquad \forall i \in I, \forall j \in J$$

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:

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

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

- 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 λ
- It turns out that for any λ, $z_{LR}(\lambda)$ 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.

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

Maximize the lower bound

- How do we choose the "best" λ?

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

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

- Let λ^* 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

$$\begin{array}{ll} \text{minimize} & cx \\ \text{subject to} & Ax = b \\ & Dx \leq e \\ & x \geq 0 \text{ and integer} \end{array}$$

$$\begin{array}{ll} \text{minimize} & cx + \lambda(Ax - b) \\ \text{subject to} & Dx \leq e \\ & x \geq 0 \text{ and integer} \end{array}$$

Lagrangian Relaxation

$$
\begin{aligned}
z_{\mathrm{LR}} &= \max_{\lambda}\left\{ \min_{x}\ cx + \lambda(Ax-b) \,\middle|\, Dx \le e, x \ge 0 \text{ and integer} \right\} \\
&\ge \max_{\lambda}\left\{ \min_{x}\ cx + \lambda(Ax-b) \,\middle|\, Dx \le e, x \ge 0 \right\} \\
&= \max_{\lambda}\left\{ \min_{x}\ (c+\lambda A)x - \lambda b \,\middle|\, Dx \le e, x \ge 0 \right\} \\
&= \max_{\lambda}\left\{ \max_{\mu}\ \mu e - \lambda b \,\middle|\, \mu D \le c + \lambda A, \mu \le 0 \right\} \\
&= \max_{\lambda,\mu}\{ \mu e - \lambda b \,|\, \mu D \le c + \lambda A, \mu \le 0 \} \\
&= \max_{\lambda,\mu}\{ \mu e - \lambda b \,|\, \mu D - \lambda A \le c, \mu \le 0 \} \\
&= \min_{y}\{ cy \,|\, Ay = b, Dy \le e, y \ge 0 \} \\
&= z_{\mathrm{LP}}
\end{aligned}
$$

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

Theorem: Let (P) be an integer program and (P-LR$_\lambda$) its Lagrangian subproblem for a given λ. If (P-LR$_\lambda$) has the integrality property for all λ, 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 $z_{LR}(\lambda) \le z_{LR} = z_{LP} \le z^* \le z(x, y)$
- 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$_\lambda$)—how?
- If the solution to (UFLP-LR$_\lambda$) is feasible, then we're lucky!!
- Remember it's infeasible since the linking constraint is violated.

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

- This mean the i-th customer is assigned to 0 or more than 1 facility
- Remedy!

Updating the multipliers

- What makes a good value of λ_i? It should be chosen to entice

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

- On the objective, it appears

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

  - If $\sum_{j \in J} y_{ij} = 0 \; (< 1)$, then λ_i is too small; it should be increased
  - If $\sum_{j \in J} y_{ij} > 1$, then λ_i is too large; it should be decreased
  - If $\sum_{j \in J} y_{ij} = 1$, then λ_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 λ 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 tolerate

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 $c(x) \le , \ge , =$
  - For $\le$ constraints, λ is restricted to be *non-positive*.
  - For $\ge$ constraints, λ is restricted to be *non-negative*.
  - For $=$ constraints, λ 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?