38 Constraint Shortest Path Problem

Problem Description

A well-known solution approach to optimization problems is Lagrangean relaxation. Lagrangean relaxation can be used to solve linear, non-linear, integer programming, and combinatorial optimization problems.

In this project we present the constraint shortest path problem, give a Lagrangean relaxation-based algorithm, and finally build a decision support system that would enable the user to solve the problem using the Lagrangean relaxation method. To learn more about the constraint shortest path problem and the Lagrangean relaxation method, we refer students to Ahuja *et al*. (1993).

Problem Formulation

The constraint shortest path problem (*P*) is a shortest path problem with additional capacity constraints (constraints (2)). Below we present a formulation of this problem.

$$\min: \sum_{(i,j)\in A} c_{ij}x_{ij} \qquad (P)$$

$$\text{Subject to:}$$

$$\sum_{\{j:(i,j)\in A\}} x_{ij} - \sum_{\{j:(j,i)\in A\}} x_{ij} = \begin{cases} 1 & \text{for } i = 1 \\ 0 & \text{for } i \in N - \{1,n\} \\ -1 & \text{for } i = n \end{cases} \qquad (1)$$

$$\sum_{(i,j)\in A} t_{ij}x_{ij} \le T \qquad (2)$$

$$x_{ij} \in \{0,1\} \qquad \text{for all } (i,j)\in A. \qquad (3)$$

Where, c_{ij} is the cost of using arc $(i,j)\in A$, *n* is the total number of nodes of the network, *T* is the capacity of the shortest path, and t_{ij} is the capacity usage coefficient for arc $(i,j)\in A$. The objective is to find the shortest path between nodes 1 and *n* of the network that does not violate the capacity constraints.

The Lagrangean relaxation method relaxes the problem by moving the complicating constraints to the objective function. The resulting optimization problem is the (unconstrained) shortest path problem. The solution to the relaxation problem is a lower bound for the capacitated shortest path problem.

The following is the Lagrangean relaxation problem (*LR*(μ)):

$$\min: \sum_{(i,j)\in A} c_{ij}x_{ij} + \mu\left(\sum_{(i,j)\in A} t_{ij}x_{ij} - T\right) = \sum_{(i,j)\in A} (c_{ij} + \mu t_{ij})x_{ij} - \mu T \qquad (LR(\mu))$$

$Subject\ to:$

$$(1)\ and\ (3)$$

is the Lagrangean multiplier.

The objective function value of ($LR(\mu)$) depends on the value of the multipliers μ. For any value of μ, ($LR(\mu)$) is a lower bound for P ($LR(\mu) \le P^*$). We are interested in identifying the highest lower bound for P^*. We do that by identifying the multiplier μ that maximizes the Lagrangean relaxation problem.

$$LR^* = \max_{\mu} LR(\mu).$$

For any value of μ, $LR(\mu) \le LR^* \le P^*$. To solve problem LR^* we use the following approach. Let μ^0 be any initial choice of the Lagrangean multipliers. We determine the subsequent values μ^k for k = 1,2,…, of the Lagrangean multipliers as follows:

$$\mu^{k+1} = \mu^k + \theta_k\left(\sum_{(i,j)\in A} t_{ij}x_{ij} - T\right). \qquad (4)$$

Where,

$$\theta_k = \frac{\lambda_k\left[UB - LR(\mu^k)\right]}{\left\|\sum_{(i,j)\in A} t_{ij}x_{ij} - T\right\|}. \qquad (5)$$

UB is an upper bound on the optimal objective function value P^*. Practitioners usually choose λ_k by starting with $\lambda_k = 2$ and then reducing λ_k by a factor of 2 if the best Lagrangean objective function value found so far has failed to improve in a specific number of iterations.

Lagrangean Relaxation Algorithm

Step 0

Set k = 0; $\mu^k = 0$; $\lambda_k = 2$;

- Find a feasible solution for problem P(1). Set UB equal to the corresponding objective function value.

Step 1

- Solve problem $LR(\mu_k)$(2).

- Keep the best lower bound found so far.

- Check if the solution to $LR(\mu_k)$ satisfies constraints (3) of P. If it does, this is a feasible solution to problem P as well. Calculate the corresponding objective function value.

- Keep the best upper bound found so far.

- If the best lower bound found is equal to the best upper bound, Stop. We have solved the capacitated shortest path problem.

Step 2

- If the solution to $LR(\mu_k)$ has not improved in the last g iterations (for example, g = 5), reduce λ_k by a factor of 2.

- Calculate θ_k.

- Calculate μ^{k+1} using (4).

- $k = k + 1$.

- If the solution to $LR(\mu_k)$ has not improved in the last G iterations (for example, G = 10), *Stop*. Report the best lower bound (solution of $LR(\mu_k)$) and upper bound (solution of P) found so far.

- Else go to Step 1.

(1) To find a feasible solution for problem *P*, one can follow a simple approach. Modify problem (*P*): replace c_{ij} with t_{ij} in the objective function and remove constraints (3). The modified problem is an un-capacitated shortest path problem. If the objective function value of the modified problem is less than or equal to T, the solution found is a feasible solution to problem *P*; otherwise problem *P* is infeasible. One can use Dijkstra's algorithm to solve the shortest path problem. For a description of this algorithm, we refer the students to Ahuja et al. (1993).

(2) Problem $LR(\mu_k)$ is the classical shortest path problem. One can use Dijkstra's algorithm to solve this problem.

Excel Spreadsheets

Build a spreadsheet that presents the following data about the arcs of the network: arc number, tail node, head node, cost, etc.

User Interface

1. Build a welcome form.
2. Build a data entry form. The following are suggestions to help you design this form. In this form insert two option buttons. These option buttons allow the user to select whether to read the data from a file or manually enter the data. Include a command button that, when clicked –on, performs these actions:
   a. If the user chose to read the data from a file, a text box should appear where the user types in the name of the file.
   b. If the user chose to enter the data manually, two text boxes appear where the user types in the total number of arcs (*m*) and nodes (*n*) of the network. Upon submission of this information, a table appears where the user enters the following data about each arc: head node, tail node, cost, and capacity.

   Insert a text box where the user can type in the capacity requirement (*T*) for this problem.
3. Build a form that allows the user to solve the problem and report the results. The following are suggestions to help you design this form. Insert a frame titled "Solve the Problem." This frame has two option buttons to allow the user to select either the Lagrangean relaxation-based algorithm described above or the Excel solver to solve the problem. Insert a command button that, when clicked on, solves the problem and opens

two frames. The first frame, titled “Reports,” has a number of option buttons that allow the user to open any of the reports described below. The second frame, titled “Sensitivity Analysis,” has a number of option buttons that allow the user to select a parameter for the sensitivity analysis. The user is interested in identifying the sensitivity of the solution with respect to capacity requirement T, arc cost, etc.

Design a logo for this project. Insert this logo in the forms created above. Pick a background color and a font color for the forms created. Include the following in the forms created: record navigation command buttons, record operations command buttons, and form operations command buttons as needed.

Reports

1. Give a graphical representation of the solution to the constraint shortest path problem.
2. Present the results from the Lagrangean relaxation algorithm. For each iteration, present the following: k, μ^k, λ_k , θ_k, the best upper bound, the best lower bound, and $LR(\mu^k)$.
3. Present the results from the sensitivity analysis.

Reference

Ahuja, R.K., Magnanti, T.L., Orlin, J.B., “Network Flows: Theory, Algorithms and Applications.” *Prentice Hall*, 1993.