A Hop Constrained Min-Sum Arborescence with Outage Costs

Rakesh Kawatra
Minnesota State University, Mankato, MN 56001
Email: Kawatra@mnsu.edu

Abstract

The hop constrained min-sum arborescence with outage costs problem consists of selecting links in a network so as to connect a set of terminal nodes N={2,3,……n} to a central node with minimal total link cost such that (a) each terminal node j has exactly one entering link; (b) for each terminal node j, a unique path from the central node to j exists; (c) for each terminal node j the number of links between the central node and j is limited to a predefined number h_j*, and (d) each terminal node has an associated outage cost, which is the economic cost incurred by the network user whenever that node is disabled due to failure of a link. We suggest a Lagrangian based heuristic to solve the integer programming formulation of this network problem.*

1. Introduction

The hop constrained min-sum arborescence (HCMA) problem is frequently encountered in the network design, routing and scheduling problems. It consists of finding links to connect a set of geographically remote nodes to a central node such that for each remote node j there is exactly one link entering node j, and for each remote node, a unique path exists from the central node to node j. The solution is subject to hop constraints which limits the number of links between the central node and any terminal node to a predefined number h. These hop constraints are often used to contain the maximum delay between any terminal node and the central node.

The hop constraints can also be used to model reliability constraints when designing telecommunication networks as pointed out by LeBlanc and Reddoch [6]. They also suggest that in many networks hop constraints can be used to avoid degradation of the signal quality. Gouveia [3] presented an integer programming model for spanning trees with hop constraints and suggested a Lagrangian based heuristic for solving the problem.

In this paper, we study the HCMA problem where each terminal node has an associated outage cost. The outage cost associated with a terminal node is the economic cost incurred by the network user whenever that node is disabled due to failure of a link. It is a measure of the opportunity cost of the equipment and its user and does not include cost of repairing or replacing the link. The location of the central node and terminal nodes, and the limit h on the number of hops between the central node and each terminal node is given. Also given are the annual costs of installing the links and the outage cost associated with each terminal node. We formulate the problem as an integer programming problem and use a Lagrangian based heuristic method to solve it. We used the lower bound given by the Lagrangian method to estimate the quality of our heuristic solutions. Subgradient optimization method is used to find good lower bounds. Computational results are presented to demonstrate the performance of the Lagrangian based heuristic for different network structures.

2. Model formulation

The hop constrained min-sum arborescence problem with node outage cost is formulated as an integer-programming problem. Our objective is to minimize the total annual cost consisting of links costs and the expected node outage costs.

We use the following notations in the paper:

N: the set of terminal nodes 2,3...n;
Node 1: central node;
T_1 = family of spanning arborescences (directed trees or branchings) with root at node 1, i.e., the family of directed networks which do not contain a cycle and such that for every terminal node t, there is a path from node 1 to node t.

C_{ij}: annual cost of installing a link (i, j);
D_t: node outage cost associated with terminal node t;

Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS'03)
0-7695-1874-5/03 $17.00 © 2002 IEEE

Q: link failure rate;

h_t: the limit on the maximum number of links between the central node and terminal node t.

Decision Variables

X_{ij}: a binary variable such that $X_{ij} = 1$ indicates that link (i, j) is in the solution; otherwise $X_{ij} = 0$;

Y^t_{ij}: a variable such that $Y^t_{ij} = 1$ if the directed link from node i to node j in on the path from node 1 to node t; otherwise $Y^t_{ij} = 0$.

The hop constrained min-sum arborescence problem with node outage costs is formulated as:

$$Z_{IP} = \text{Minimum}\left\{\sum_{i=1}^{n}\sum_{j=2}^{n} C_{ij}X_{ij} + \sum_{t=2}^{n} Q*D_t \sum_{i=1}^{n}\sum_{j=2}^{n} Y^t_{ij}\right\} \quad (1)$$

subject to

$$X \in T_1 \quad (2)$$

$$\sum_{j=2}^{N} Y^t_{ij} - \sum_{j=1}^{N} Y^t_{ji} = \begin{cases} +1 \text{ if } i=1 \\ -1 \text{ if } i=t \\ =0 \text{ otherwise} \end{cases}$$

$$\text{for all } i \in N \cup [1],\ t \in N \quad (3)$$

$$Y^t_{ij} \le X_{ij} \quad \text{for all } i \in N \cup [1], \text{ and } j, t \in N \quad (4)$$

$$\sum_{i=1}^{N}\sum_{j=2}^{N} Y^t_{ij} \le h_t \quad \text{for all } t \in N \quad (5)$$

$$X_{ij} \in \{0,1\} \text{ for all } i \in N \cup [1], \text{ and } j \in N \quad (6)$$

$$Y^t_{ij} \in \{0,1\} \quad \text{for all } i \in N \cup [1], \text{ and } j, t \in N \quad (7)$$

In the above model, constraints (3) are flow conservation constraints. Constraints (4) ensures that if there is no direct link from node i to node j, then there cannot be direct flow from node i to node j. Constraints (5) are the hop constraints which for every terminal node t, $t \in N$, limit the number of links between the central node and node t to a predefined number h_t.

3. Solution methods

We propose a Lagrangian based heuristic method to solve this problem. In our approach we first form a Lagrangian relaxation of the problem which is solved optimally. Next, we use a branch exchange heuristic to generate a feasible solution from the infeasible Lagrangian solution. We use subgradient optimization method to find good Lagrangian multipliers. The best values of the lower bound and the feasible solution are retained when the subgradient algorithm stops. The lower bound given by the Lagrangian relaxation is used to obtain a quantitative estimate of the quality of the solution given by the branch exchange heuristic.

3.1 Lagrangian relaxation

In this study we use a Lagrangian relaxation approach to generate lower bounds for the hop constrained min-sum arborescence problem with node outage costs. Lagrangian relaxation can be used to obtain tight lower bounds for a variety of integer programming problems. (See Fisher [2] for an application-oriented survey of Lagrangian relaxation).

We form a relaxation of the hop constrained min-sum arborescence problem by multiplying each constraint (4) by a nonnegative Lagrange multiplier μ_{ijt} and each constraint (5) by nonnegative multiplier θ_t and adding the products to the objective function. This results in the following relaxation of problem Z_{IP}:

$$L(\mu,\theta) = \text{Minimize}\left\{Q(\mu) + \sum_{t=2}^{n} M_t(\mu,\theta) - \sum_{t=2}^{n}\theta_t h_t\right\}$$

s.t. (2), (3),(6) and (7),

where

$$Q(\mu) = \text{Minimize}\left\{\sum_{i=1}^{n}\sum_{j=2}^{n} X_{ij}\left(C_{ij} - \sum_{t=2}^{n}\mu_{ijt}\right)\right\}$$

s.t. (2) and (6),

and

$$M_t(\mu,\theta) = \text{Minimize}\left\{\sum_{i=1}^{n}\sum_{j=2}^{n} (\mu_{ijt} + \theta_t + Q*D_t) Y^t_{ij}\right\}$$

s.t. (3) and (7).

3.1.1 Procedure for evaluating $Q(\mu)$

Solving $Q(\mu)$ requires finding the min-sum arborescence $\hat{X}(\mu)$ rooted at node 1, which for a given vector of Lagrange multipliers μ can be accomplished using Fischetti and Toth's algorithm [1]. For this algorithm, we use the length of

$$\text{arc (i, j)} = \left(C_{ij} - \sum_{t=2}^{n}\mu_{ijt}\right).$$

3.1.2 Procedure for solving $M_t(\mu,\theta)$

The function $M_t(\mu,\theta)$ is evaluated by solving a single-commodity flow problem. In this problem, one unit of a commodity t is to be shipped from the central node to node t. Since the links are uncapacitated, the flow $\hat{Y}_t(\mu,\theta)$ will be along the shortest path from the central node to node t, which can be found using Dijkstra's algorithm [5] with $\mu_{ijt} + \theta_t + Q * D_t$ as the cost of directly shipping one unit of commodity t from node i to node j. While solving $M_t(\mu,\theta)$ we stop the Dijkstra's algorithm as soon as a shortest path to node t is found.

3.2 Improving the Lagrange multipliers

It is well known that for any μ and θ, the value of the Lagrangian relaxation L(μ,θ) provides a lower bound to Z_{IP}. We wish to find the tightest bound which can be achieved, i.e., we wish to solve the Lagrangian dual problem to obtain the best lower bound, $L(\mu^*,\theta^*) = \max_{\mu,\theta\geq 0} \{L(\mu,\theta)\}$.

Searching for the optimal Lagrangian multiplier vectors μ^* and θ^* is very time consuming; however, approximate values can be found by using a subgradient optimization method [4]. This method begins with an initial vector of multipliers μ^0 and θ^0, which at iteration p is adjusted using the following rule:

$$\mu_{ijt}^{p+1} = \mu_{ijt}^{p} + s_p * V_{ijt}^{p} \qquad \forall\, j,t \in N,\ i \in N \cup [1]$$

$$\theta_t^{p+1} = \theta_t^{p} + S_p * U_t^{p} \qquad \forall\, t \in N$$

where

$$V_{ijt} = Y^t{}_{ij} - X_{ij} \qquad \forall\, j,t \in N,\ i \in N \cup [1]$$

$$U_t = \sum_{i=1}^{n}\sum_{j=2}^{n} Y_{ij}^{t} - h_t \qquad \forall\, t \in N$$

and $s_p = \dfrac{\lambda(Z^* - L(\mu^p,\theta^p))}{\left\|V_{ijt}^{p}\right\|^2 + \left\|U_t\right\|^2}$.

In the computation above, V and U are the sub-gradients of L(μ,θ), $\|\ \|^2$ denotes the Euclidean norm, Z^* is the best available overestimate of the optimal solution value, and λ is a scalar multiplier which satisfies the condition $0<\lambda\leq 2$. The value of λ is initially set equal to 2 and is reduced during the course of the search.

3.3 A Lagrangian based branch exchange heuristic

In our research, for simplicity we assumed $h_t = h$ for each terminal node t. Our heuristic can very easily be extended to allow different values of h_t for different terminal nodes. Since Q(μ) was solved independent of hop constraints, the optimal solution to Q(μ) may have more than h hops in the path from the central node to some of the terminal nodes, which is an infeasible solution to problem Z_{IP}. After every iteration of the subgradient optimization algorithm we use a branch exchange heuristic to generate a feasible solution to Z_{IP}. The best feasible solution is retained when the subgradient optimization algorithm is terminated. This branch exchange heuristic is an iterative procedure. In each iteration this heuristic identifies set of terminal nodes, called B^+, which are exactly (h+1) hops from the central node. Next the heuristic finds a subset C of set B^+ such that for each node j that belongs to the set C: the node or nodes that belong to the subtree rooted at node j and are furthest away from node j are exactly (h-1) hops away from this node. For each node j belonging to set C, the heuristic removes the link terminating at this node and replaces it with link (1,j). If set C is empty, then the heuristic identifies a terminal node j belonging to set B^+ for which at minimal increase in total annual cost of the network, the link terminating at node j can be replaced with link (i, j) where node i is at most (h-1) hops from the central node. If the replacing link violates any constraint, then it is ignored. This is continued until all the terminal nodes in the network satisfy the hop constraint. The branch exchange heuristic ends when the current solution is feasible.

For this heuristic we define the following additional notations:

g_j = the origin of the link incident to node j.

$Level_j$: number of hops between the central node and node j.

Leaf node: a terminal node with no links originating from it.

$Leaf_t$: set of leaf nodes belonging to the subtree rooted at node t.

$B^+ = \{t \mid Level_t = h+1 \text{ for all } t \in N\}$, i.e., the set of nodes that are (h+1) hops from the central node.

$B^- = \{t \mid Level_t < h \text{ for all } t \in N\}$, i.e., the set of nodes that are at most (h-1) hops from the central node.

Initially, $X = \hat{X}(\mu)$. Note that, since all of the values above are dependent upon the value of X, they must be recomputed each time X is modified by replacement of one or more links.

Step 1. If $B^+ = \phi$, then **STOP**; Else for each $j \in B^+$, find $\max lev_j = \max_{i \in Leaf_j} \{Level_i\}$.

Step 2. Find j^* s.t. $\max lev_j = h-1$.

Step 3. If $j^* = \{\phi\}$, then go to Step 4; Else Set $X_{g_{j^*} j^*} = 0$ and $X_{1 j^*} = 1$, and return to Step 1.

Ties if any are broken arbitrarily.

Step 4. If $B^+ = \phi$, then **STOP; ELSE** for each $j \in B^+$ and $i \in B^-$, find

$$\Delta_{ij} = \left\{C_{ij} - C_{g_j j} + (\text{Level}_i - \text{Level}_{g_j}) * Q * D_j\right\}$$

Step 5. Find $(i^*, j^*) = \arg\min_{i,j} \left\{\Delta_{ij}\right\}$;

Step 6. Set $X_{g_{j^*} j^*} = 0$ and $X_{i^* j^*} = 1$, and return to Step 1.

4. Numerical results

The effectiveness of the Lagrangian based heuristic was investigated by solving a randomly generated set of test problems. The data for the computational experiments were generated by drawing the coordinates of the nodes from a uniform distribution over a square of size 1000 by 1000. The annual cost of link (i, j) was chosen to be the Euclidean distance between point i and point j. The node outage costs were randomly generated from a uniform distribution U[0,1000]. We solved the problems for n = 20, 40, and 60; h = 3, 4, and 5; and Q = 0.02, 0.04, and 0.06. In our experiments we assumed $h_t = h$ for all $t \in N$. For each parameter set we solved 3 instances of the problem and computed the average gap. For purposes of the subgradient optimization method, we used the best heuristic solution obtained so far as the overestimate of the optimal objective function value. The initial value of the scalar λ was set to 2, and halved whenever $L(\mu^p, \theta^p)$ did not improve in 22 successive iterations. The Lagrange multipliers were initially set to 0. The stopping criterion in computation of the lower bounds was: stop if the total number of iterations exceeds 4000 or if the objective function value changes by less than 0.2 in 22 successive iterations. The Lagrangian based heuristic method was coded in Fortran 77 and run on IBM SP computer with a maximum processing speed of 888 MHz. Computational results of the experiment are presented in Table 1.

The computational results presented in Table 1 show that the average gap between the heuristic solution and the Lagrangian lower bound is within 14 percent. This gap provides an upper bound for the gap between the heuristic and optimal solutions.

Table 1. Computational results

N	h	Q	Average Gap
20	3	0.02	1.9%
20	3	0.04	2.6%
20	3	0.06	2.0%
20	4	0.02	0.4%
20	4	0.04	1.6%
20	4	0.06	1.6%
20	5	0.02	0.8%
20	5	0.04	1.6%
20	5	0.06	1.8%
40	3	0.02	8.2%
40	3	0.04	3.7%
40	3	0.06	4.3%
40	4	0.02	6.1%
40	4	0.04	4.0%
40	4	0.06	2.5%
40	5	0.02	4.9%
40	5	0.04	2.9%
40	5	0.06	2.5%
60	3	0.02	13.5%
60	3	0.04	7.5%
60	3	0.06	7.0%
60	4	0.02	11.3%
60	4	0.04	7.7%
60	4	0.06	7.3%
60	5	0.02	10.5%
60	5	0.04	6.7%
60	5	0.06	7.3%

Gap=(heuristic solution –lower bound)/(heuristic solution)

5. Conclusions

In this paper we presented an integer programming model of a hop constrained min-sum arborescence problem with node outage costs, in which the terminal nodes in the network must be connected to a central node with a constraint that limits the maximum number of links between the central node and each terminal node to a predetermined number h. We have suggested a Lagrangian based heuristic to find a low cost feasible solution. The lower bound found as a byproduct of the solution procedure is used to estimate the quality of the heuristic solution. Computational results for a variety of problems are reported. In our computational experiment, the average gap between the heuristic solution value and the optimal solution is shown to be within 14 percent.

6. References

[1] M. Fischetti and P. Toth, "An efficient algorithm for the min-sum arborescence problem on complete digraphs," ORSA Journal on Computing, vol. 5, no. 4, pp. 426-434, 1993.

[2] M. L. Fisher, "The Lagrangian relaxation method for solving integer programming problems", Management Science, vol. 27, 1-18, 1981.

[3] L. Gouveia, “Multicommodity flow models for spanning trees with hop constraints,” European Journal of Operational Research, vol. 95, 178-190, 1996.

[4] M. Held, P. Wolfe, and H. D. Crowder, "Validation of Subgradient Optimization", Mathematical Programming, vol. 6, 62-88, 1974.

[5] R. Larson, and A. Odoni, "Urban Operations Research", Prentice Hall, Englewood Cliffs, N. J., 1981.

[6] L. LeBlanc and R. Reddoch, “Reliable link topology/capacity design and routing in backbone telecommunication networks,” paper presented at the First ORSA Telecommunications SIG conference, 1990.