Path Level Reliability in Overlay Multicast Tree for Realtime Service

Chae Y. Lee and Jung H. Lee
Dept. of Industrial Engineering, KAIST,
373-1 Kusung Dong, Taejon, Korea
{chae, shazam}@kaist.ac.kr

Abstract

Overlay Multicast is a promising approach to overcome the implementation problem of IP multicast. Real time services like internet broadcasting are provided by overlay multicast technology due to the complex nature of IP multicast and the high cost to support multicast function. Since multicast members can dynamically join or leave their multicast group, it is necessary to keep a reliable overlay multicast tree to support real time service without delay. In this paper, we consider path level reliability that connects each member node. The problem is formulated as a binary integer programming which maximizes the reliability of multicast tree. Tabu search based algorithm is presented to solve the NP-hard problem.

1. Introduction

Multicast method in communication has become an important part of many next generation applications, including video-on-demand (VOD) and IPTV. Many emerging Applications in the Internet are characterized by high volume data rate and multi receivers. Multicast is an efficient way to transmit same information to a group of receivers simultaneously [1].

In traditional IP multicast, Routers play an important role in replication and transmission of data packets to receivers. However, most commercial ISPs didn't deploy routers supporting IP multicast yet because of the implementing cost and the complexity of the technology [2, 3].

Overlay multicast is widely studied as an alternative to IP multicast. Differently from IP multicast, it didn't require changing legacy routers. Overlay multicast can be implemented on application layer. After source node transmits packets to one or several end node, receiving end node replicates and sends packets one after the other [4].

There are two major research issues in overlay multicast. One is a topic on minimizing the delay from a multicast source to each member. Lee, Park and Baek [5] studied minimizing maximum delay in dynamic overlay network. The problem is formulated as a degree-bounded minimum spanning tree. A tabu search heuristic is developed. The other one is about the reliability of overlay multicast tree. Cho and Lee [6] were interested in multicast tree rearrangement to recover node failures. Lee and Kim [4] studied a problem of designing reliable overlay multicast trees with multiple sessions. The objective is to build a reliable multicast tree for each session that satisfies common constrains of an overlay network. Link level reliability is considered in building overlay multicast tree.

In this paper, we are interested in path level reliability in overlay multicast tree. An overlay multicast tree can be broken due to join and leave of multicast members. Each member's probability of being serviced without failure depends on the reliability of the path through which data packets are delivered. Sojourn probability of each member node on the path from source to destination are considered.

The rest of this paper is organized as follows. In section 2, we formulated the construction of overlay multicast tree that considers path level reliability. In section 3, a tabu search based algorithm is developed to solve the NP-hard problem. Computational results and conclusion will be provided in section 4 and 5.

2. Path Level Reliability and its Formulation

Due to a characteristic of overlay multicast tree, we need to consider joining or leaving of member nodes. When a member node joins or leaves a multicast group, it needs to be connected into or disconnected from its tree. Joining of a member node can be handled by periodically updating the tree or connecting it to a member node with enough capacity. However, updating procedure is not simple when a member node leaves a multicast group. If a member node leaves a multicast group, all descendent nodes of the member are disconnected. Therefore, we need to consider reliability to build a sustainable overlay multicast tree.

Link reliability is considered in [4] to increase reliability of multicast tree. Sojourn time of two member nodes are considered to have the link reliability. A tabu search heuristic is employed to maximize the minimum link reliability. However, the path reliability from a source to a member node is more critical than the link reliability in the overlay multicast.

In this paper, we consider path level reliability in overlay multicast. Given a multicast tree, the path reliability from a source to a member node is obtained. The path reliability is then utilized to obtain reliability of overlay multicast tree. Path reliability is obtained as the multiplication of every node reliability in the path. Tree reliability is also obtained by the combination of the path reliability. The path based tree reliability of the overlay multicast tree given in figure 1 is computed as $p_1^4 * p_2 * p_3^3 * p_4 * p_5$.

The reliability of each node i is discussed to be p_i.

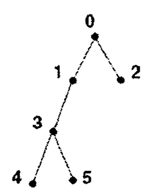

Figure 1. An overlay multicast tree with six nodes

It is clear from the above example that the tree reliability is highly dependent on the nodes directly connected to the source and those in the upper level of a tree.

An overlay multicast tree can be modeled with a graph G= (V, E), where V represents the set of multicast member node and E represents links between nodes. A node m∈V is a multicast member which needs to be serviced from the source node. A path is required to connect node m to the source node s. Let x_{ij} be a binary variable for link (i, j). If there is a link between node i and j in multicast tree, x_{ij} = 1. Otherwise, x_{ij} = 0. And Let y_{ijm} be a binary variable for a piece of path to the member node m. If a piece of path (i, j) is used for multicast service for member node m in multicast, y_{ijm} =1. Then the following flow conservation equations hold for every node in multicast network.

$$\sum_{j \neq i} y_{ijm} - \sum_{j \neq i} y_{jim} = \begin{cases} +1, & \text{if } i = s, \text{ for } i, j, m \in V \\ -1, & \text{if } i = m, \text{ for } i, j, m \in V \\ 0, & \text{otherwise} \end{cases}$$

A piece of path (i, j) to node m can be chosen only if link between i and j is selected. This constraint represent following equation.

$$y_{ijm} \le x_{ij}, \qquad \text{for m} \in V \text{ and } (i,j) \in E$$

Then, we have following equation to build a spanning tree for each session with n members.

$$\sum_{(i,j) \in E} x_{ij} = n - 1,$$

Now, we consider end-to-end delay constraint. Let d_{ij} be the delay between node i and j and L be the delay bound.

$$\sum_{(i,j) \in E} d_{ij} y_{ijm} \le L, \qquad \text{for } m \in V$$

If d_{ij}=1 for all (i, j), above equation is constraint of hop count for each member.

To satisfy degree constraint of node i, we have following constraints.

$$\sum_{j \neq i} x_{ij} + \sum_{j \neq i} x_{ji} \le D_i, \quad \text{for } i, j \in V$$

The last constraint we will consider is about link capacity. When multicast service havs service data rate r and each link (i, j) can support up to C_{ij}, we have

$$r x_{ij} \le C_{ij}, \qquad \text{for } (i,j) \in E$$

Now, our objective is building an overlay multicast tree with higher path level reliability. Let p_i be the sojourn probability of node i in overlay multicast service. In this paper, path level reliability of overlay multicast tree is represented as $\sum_{m \in V} \sum_{(i,j) \in E} y_{ijm} \log p_j$. To present path level reliability, we introduced a logarithmic function, which is usually employed for representing various utilities. The objective value is represented as a negative value due to an adopted logarithmic function. Overlay multicast tree with bigger value of path level reliability is more reliable. Thus, our objective function is given as

$$\text{Maximize } \sum_{m\in V}\sum_{(i,j)\in E}\left(y_{ijm}\log p_j\right)$$

The above objective function will maximize the path level reliability of overlay multicast tree. From the above discussion, we have the following binary integer programming formulation.

$$\text{Maximize } \sum_{m\in V}\sum_{(i,j)\in E}\left(y_{ijm}\log p_j\right)$$

subject to

$$\sum_{j\neq i} y_{ijm} - \sum_{j\neq i} y_{jim} = \begin{cases} +1, & \text{if } i=s, \text{ for } i,j,m\in V \\ -1, & \text{if } i=m, \text{ for } i,j,m\in V \\ 0, & \text{otherwise} \end{cases}$$

$$y_{ijm} \le x_{ij}, \qquad \text{for m} \in V \text{ and } (i,j)\in E$$

$$\sum_{(i,j)\in E} x_{ij} = n-1,$$

$$\sum_{(i,j)\in E} d_{ij} y_{ijm} \le L, \qquad \text{for } m\in V$$

$$\sum_{j\neq i} x_{ij} + \sum_{j\neq i} x_{ji} \le D_i, \qquad \text{for } i,j\in V$$

$$r\,x_{ij} \le C_{ij}, \qquad \text{for } (i,j)\in E$$

$$x_{ij}, y_{ijm} \in \{0,1\}$$

Note that above overlay multicast tree construction is a NP-hard problem [7]. Since this is NP-hard problem, we can not solve the proposed integer programming by any conventional method. We propose a tabu search heuristic to solve overlay multicast tree construction problem. Tabu search heuristic is a promising approach which has been successfully applied to many combinatorial optimization problems.

3. Tabu Search

Tabu search [8] is a successful meta-heuristic method for solving practical optimization problem. The main idea of tabu search includes three general components. 1) Initial solution, 2) Intensification with a short-term memory, 3) Diversification with a long-term memory.

3.1 Initial Overlay Multicast tree

We need an initial overlay multicast tree as a starting point of tabu search. We propose two algorithms for initial tree: Degree First Initial tree and Random Initial tree. For the Degree First Initial tree, Member nodes are sequenced in nonincreasing order of node degree. When more than 2 nodes make ties in degree, nodes are sequenced by sojourn probability. Then, a spanning tree is built by selecting a member node in sequence. A selected node organizes itself into a tree with minimum hop count. This process continues until all member nodes are connected to each tree. A tree built by above procedure may not satisfy delay bound or link capacity. To build a feasible tree, a node that does not satisfy constraints is selected. The selected node and its descendants are reconnected to an ascendant node which will satisfy constraints. For the random initial tree, Initial tree is built by randomly sequenced nodes. Tree is constructed by putting selected nodes in tree and reconnection process is repeated until all nodes satisfy given constraints.

3.2 Intensification

At first, we consider two types of moves: "node swap" and "node reconnection". In node swap move, worst effect node j with the lowest $\sum_{i\in V} y_{ijm}\log p_j$. The lowest $\sum_{i\in V} y_{ijm}\log p_j$ implies that a node with lower sojourn probability is placed in upper part of multicast tree and has more children node than others have. Worst effect node is selected as a target node. Target node is exchanged with a node that has higher sojourn probability with one less number of children nodes. Node reconnection is simply to reconnect the target node to other node that best improves tree's path level reliability.

Intensification procedure is based on a short-term memory which is embodied tabu list. After applying the node swap or node reconnection move, the target node that is moved is added to the tabu list. Nodes in tabu list are prohibited for a certain period to be selected again as a target node. Intensification procedure is repeated until no solution improvement is obtained consecutively for N_max times.

3.3 Diversification

Diversification with long-term memory is adopted to search broader solution space escaping from local optimality. It is triggered when no improvement is obtained during N_max iterations of intensification process. Nodes that are not placed near the source are chosen for nodes near the source in new tree. Nodes are sequenced by how distant they are placed from the source. Then a new tree is constructed by selecting a node in sequence. After constructing a new tree, tabu search is continued with the intensification process. Diversification process is repeated D_max times and the procedure is terminated.

4. Computational Result

Three different size of overlay multicast networks are considered. Problem with 10, 30 and 50 nodes are generated. For each case, 10 problems are generated by randomly selecting sojourn probability of nodes, node degree, link capacity and delay bound as in the Table 1.

• Number of nodes(n).	10, 30, 50 nodes
• Sojourn probability of nodes(p_i):	0.3~0.9
• Node degree(D_i).	3, 4, 5
• Link delay(d_{ij}):	1 hop
• Delay Bound(L):	0.3n, 0.5n
• Serive data rate(r):	384kbps, 600kbps
• Link capacity(C_{ij}):	0.5Mbps~1Mbps

Table 1. Parameters for Overlay Multicast tree

We first test out two initial tree constructing strategies: Degree First Initial Tree and Random Initial Tree. Problems with 30 nodes are tested with 10 examples. Test shows that Degree First Initial Tree gives that better solution. Thus, we adopt Degree First Initial Tree for the initial overlay multicast tree. Before applying tabu search, we need to tune the tabu parameters: tabu list size, N_max for intensification and D_max for Diversification. A tabu list size represent the number of iterations during which target node selected before is forbidden to be selected again as a target node. We tested 30 nodes problem and find that 0.2n is the best tabu size. We performed test for N_max and D_max and find out that the appropriate N_max is 0.3n and D_max is 8.

Table of 2 shows the result of computation. CPLEX [9] is employed to compare the performance of the proposed tabu based heuristics. From tables, result of tabu search is as good as one of CPLEX. CPLEX failed to obtain the optimal solution in reasonable time for problems with more than 50 nodes. This failure is due to exponential growth of branches in the process of CPLEX. CPLEX experienced memory problem when solving 100 nodes problem and failed to give out a solution.

5. Conclusion

To build a reliable overlay multicast tree, path level reliability is examined. The problem is formulated as a binary integer programming which is known as NP-hard. A tabu search heuristic is developed by adopting intensification and diversification. Node swap and reconnection is proposed for intensification. Diversification is implemented with long term memory. Computational result shows that the proposed tabu search gives outstanding performance.

Problem	10 nodes			30 nodes			50 nodes		
	tabu search	CPLEX	GAP*	tabu search	CPLEX	GAP*	tabu search	CPLEX	GAP*
1	-2.47 (0.4)	-2.43 (1.7)	0.0161943	-8.73 (2.9)	-8.58 (1000.0**)	0.0171821	-29.05 (7.1)	n/a***	
2	-2.53 (0.7)	-2.49 (1.9)	0.0156103	-9.12 (2.1)	-9.02 (1000.0**)	0.0109645	-31.87 (8.1)	n/a***	
3	-2.55 (0.9)	-2.52 (1.6)	0.0117647	-8.9 (3.2)	-8.72 (1000.0**)	0.0202247	-29.68 (7.4)	n/a***	
4	-2.41 (0.5)	-2.38 (1.6)	0.0124481	-8.74 (2.6)	-8.63 (1000.0**)	0.0125858	-28.08 (7.2)	n/a***	
5	-2.39 (0.7)	-2.31 (1.9)	0.0334728	-9.06 (2.0)	-8.96 (1000.0**)	0.0110375	-28.64 (6.3)	n/a***	
6	-2.53 (0.7)	-2.40 (1.8)	0.0137628	-8.85 (1.9)	-8.77 (1000.0**)	0.0090395	-29.37 (6.8)	n/a***	
7	-2.81 (0.9)	-2.75 (1.1)	0.0213523	-8.88 (2.1)	-8.74 (1000.0**)	0.0157658	-29.46 (7.2)	n/a***	
8	-2.64 (0.3)	-2.53 (1.9)	0.0416667	-9.23 (2.3)	-9.12 (1000.0**)	0.0119177	-30.52 (7.5)	n/a***	
9	-2.49 (0.6)	-2.47 (1.6)	0.0080321	-8.79 (2.4)	-8.63 (1000.0**)	0.0182025	-28.65 (7.2)	n/a***	
10	-2.72 (0.6)	-2.69 (1.5)	0.0110294	-9.08 (1.9)	-8.89 (1000.0**)	0.0209251	-28.94 (7.5)	n/a***	

* GAP=(tabu search − CPLEX)/ tabu search
** Terminated by the time limit
*** failed to solve

Table 2. Computational Result of tabu search

References

[1] Yang-hua Chu, Aditya Ganjam, T. S. Eugene Ng, Sanjay G. Rao, Kunwadee Sripanidkulchai, Jibin Zhan and Hui Zhang, "Early Experience with an Internet Broadcast System Based on Overlay Multicast", Technical Report CMU-CS-03-214, Carnegie Mellon University, Dec., 2003.

[2] Christophe Diot, Brian Neil Levine, Bryan Lyles, Hassan Kassem and Doug Balensiefen, "Deployment Issues for the IP Multicast Service and Architecture", IEEE Network, Vol. 14, No. 1, 2000, pp. 78-88.

[3] Carlos A.S. Oliveira and Panos M. Pardalos, "Algorithms for the Streaming Cache Placement Problem on Multicast Networks", In Proceedings of Seventh INFORMS Telecommunications Conference (ITC'04), Boca Raton, Florida, Mar., 7-10, 2004.

[4] Chae Y. Lee and Ho Dong Kim, Reliable Overlay Multicast Trees for Private Internet Broadcasting with Multiple Sessions, Computers and Operations Research, accepted, 2005.

[5] Chae Y. Lee, Hyo Jung Park and Jin woo Baek, "A Multicast Tree to Minimize Maximum Delay in Dynamic Overlay Networks" 2006.

[6] Hee K. Cho and Chae Y. Lee, Multicast Tree Rearrangement to Recover Node Failure in Overlay Multicast Network, Computers and Operations Research, Vol. 33, No.3, pp. 581-594, 2006.

[7] S, Shi, J. Turner, and M. Waldvogel, Dimensioning server access bandwidth and multicast and multicast routing in overlay networks, in Proceedings of NOSSDAV, June, 2001.

[8] Fred Glover and Manuel Laguna, "Tabu Search", Kluwer Academic Publishers, 1997.

[9] CPLEX 8.1, http://www.ilog.com.