An Efficient Distributed Routing Algorithm for Delay Constrained Multicast

Hieu T. Tran and Richard J. Harris
School of Electrical and Computer Engineering, RMIT University,
Melbourne, Australia
Email: {hieu, richard}@catt.rmit.edu.au

***Abstract*— For dynamic QoS multicast routing, efficient handling of node's join and leave is an important problem. This paper addresses the problem of optimally connecting a new node to an existing multicast tree under source-to-destination delay and inter-destination delay variation constraints (MRDC-JOIN). We assume a QoS framework where a delay dependent cost function is associated with each network link. After showing that MRDC-JOIN is NP-complete, we present centralized and distributed versions of the algorithm MRDC-JOIN. By means of simulation, we show that the algorithm is capable of handling join requests efficiently in terms of success rate and tree cost in comparison with a centralized static multicast routing algorithm [1].**

***Index Terms*— Multicast, QoS, source-to-destination delay, inter-destination delay variation, mathematical programming/optimization, simulations.**

I. Introduction

QoS-based multicast routing has attracted a lot of interest due to increasing demand in group-based real-time applications that require stringent quality of service, such as video conferencing and multi-player Internet games. For applications involving group communication, multicast is more efficient than unicasting as it allows for the dissemination of a single copy of data to a group of destinations instead of sending a separate copy to each destination as in a unicast communication.

As far as real-time applications are concerned, delay is one of the most important QoS parameters. Commonly, the delay from a source to all destinations should be bounded [2], [3]. In addition, the requirement for inter-destination delay variation (the difference between delays from the source to different destinations) also arises for applications that require a certain level of synchronization among various destinations [2], [3]. Furthermore, minimizing the cost of the multicast connection is also an important issue.

This work was supported by the Australia Telecommunication Co-operative Research Centre (ATcrc).

When a node wishes to join an existing multicast session, a route from the existing multicast tree to the node must be computed. This paper deals with the problem of optimally connecting a node to an existing source-based multicast tree such that end-to-end delay and inter-destination delay variation constraints of the overall tree are (still) satisfied (the existing tree currently satisfies the constraints). We assume a QoS framework where a delay dependent cost function is associated with each network link and link delays are integer valued.

The rest of the paper is organized as follows. The underlying problem MRDC-JOIN is formulated in Section II. Section III reviews a number of related works. Section IV shows that MRDC-JOIN is NP-complete. Heuristics for the sub-problem of MRDC-JOIN and the test results are described in Section V. The simulation results show that our heuristics outperform one of the currently best known existing heuristics (H_MCOP [4]). Section VI presents centralized and distributed versions of a heuristic algorithm for the problem MRDC-JOIN, and discusses simulation results using the algorithm. Section VII concludes the paper.

II. Problem Formulation

We represent a network as a connected directed graph $G(V, E)$. V and E are the set of n nodes and m links of the network respectively. The set of destinations is denoted by M, which is a subset of V.

The source-based tree, that is rooted at a source node s and spans M, is represented as a directed tree $T(V_T, E_T)$ where V_T and E_T represent the set of nodes and the set of links in the tree. In the tree T, an unique path from a node u to a node v, if one exists, is denoted by p_{uv}^T. A *branch point* of paths p_{su}^T and p_{sv}^T, denoted by B_{uv}^T, is the node where the two paths split.

We consider a QoS framework similar to those described in [5]–[7] where it is possible to impose a requirement, d_l, on link delay for a cost $c_l(d_l)$. We assume that link cost functions are non-increasing and delays are integer. We further assume that link delay variation, Δ_l, and the link propagation delay, δ_l, are constant. If the delay requirement imposed on link l is d_l then, at any time, link delay is guaranteed to be within $[d_l - \Delta_l, d_l]$.

Given a source-to-destination delay bound D^e, and an inter-destination variation delay bound D^i, a *feasible delay partition* $S^T_{D^e D^i} = \{d_l\}_{l \in E_T}$ is a set of delay requirements imposed on the links of the multicast tree T such that:

$$D_{p^T_{su}} \leq D^e \quad \forall u \in M \tag{1}$$

$$D_{p^T_{bu}} - (D_{p^T_{bv}} - \Delta_{p^T_{bv}}) \leq D^i$$
$$\forall u, v \in M, \quad u \neq v, b = B^T_{uv} \tag{2}$$

$$\delta_l + \Delta_l \leq d_l \quad \forall l \in E_T \tag{3}$$

where, $D_{p^T_{uv}} = \sum_{l \in p^T_{uv}} d_l$, and $\Delta_{p^T_{uv}} = \sum_{l \in p^T_{uv}} \Delta_l$.

The constraints above have the following interpretation: (1) requires that maximum delay from the source to a destination must not exceed the source-to-destination delay bound, (2) ensures that the difference in delays from the source to any pair of destinations will not be greater than the inter-destination delay variation bound, (3) simply requires that the delay requirement imposed on a link is feasible.

Given the notation described above, the Multicast Routing with Delay Constraints – JOIN (MRDC-JOIN) problem can now be formally stated as follows:

Problem MRDC-JOIN: Given $G(V, E)$, $\{\delta_l, \Delta_l, c_l(d)\}_{l \in E}$, a source-to-destination delay bound D^e, an inter-destination delay variation bound D^i, a multicast tree $T \subseteq G$ rooted at a source node $s \in V$, spanning a group $M \subseteq V - \{s\}$, a feasible partition $\{d_l\}_{l \in T}$, find an in-tree node $t \in T$ and a route from t to a given node v, $p \subseteq G \setminus T \cup \{t\}$, and a set $\{d_l\}_{l \in p}$ such that $\sum_{l \in p} c_l(d_l)$ is minimized while the partition $\{d_l\}_{l \in E_{T \cup p}}$ is feasible.

III. Related Work

In a later section, we shall show that the sub-problem of MRDC-JOIN is, in essence, a combined delay partition and (unicast) routing problem with a single additional constraint on the total delay variation of the route.

The optimal delay partition and routing problem for convex cost functions was investigated in [5] and a pseudo-polynomial algorithm was proposed. Reference [6] extended that problem to general cost functions and developed a number of pseudo-polynomial algorithms and fully polynomial approximation versions of those algorithms. Similar and related problems were also addressed in [8], [9] and reference therein.

Ours is different from previous work because not only propagation delay and delay variation are taken into account but there is also an additional constraint on the total delay variation of the route. It is this constraint that makes MRDC-JOIN NP-complete as explained in the next section.

A number of papers (e.g., [2], [3]) considered multicast routing with end-to-end delay and inter-destination delay variation constraints but all assumed fix link delays and/or fix link costs. Compared to the assumptions made by previous work, our assumptions are more realistic, hence, our solution will more likely find application in real-life problems.

IV. MRDC-JOIN is NP-Complete

We transform Restricted Shortest Path problem (RSP), which is known to be NP-complete [8], to MRDC-JOIN. Let us consider an arbitrary instance of RSP: given a graph $G(V, E)$, each edge e is associated with a weight w_e and length l_e, find a shortest path from node s to v such that total path weight is no greater W. We construct a network $G'(V', E')$ that consists of $G(V, E)$ and a new edge (s, u) (see Figure 1). Let $D^e = \sum_{e \in E} w_e$, $D^i = W$. Let the current multicast tree compose of edge (s, u) and $\delta_{(s,u)} = 0$, $\Delta_{(s,u)} = 0$, and $d_e = D^e$. For all $e \in G$, set $\delta_e = 0$, $\Delta_e = w_e$, $c_e(d) = l_e$.

We are going to show that the optimal solution to this problem instance of MRDC-JOIN is also the optimal solution to RSP and vice versa. First, assume the optimal solution of MRDC-JOIN is a path p and a partition $\{d_e\}_{e \in p}$. Then, we must have:

$$\sum_{e \in p} d_e \leq D^e \tag{4}$$

$$\sum_{e \in p} \Delta_e \leq D^i + \sum_{e \in p} d_e - D^e \tag{5}$$

Because link cost is constant and the constraint (5) becomes less strict as $\sum_{e \in p} d_e$ increases, path p and a delay partition having total of D^e must also be an optimal solution. In other words, p is the least-cost path among all paths having $\sum_{e \in p} \Delta_e \leq D^i$. Since $D^i = W$ and $\Delta_e = w_e$, p is also the optimal solution to RSP.

Conversely, suppose p is the optimal path of RSP problem, it is easy to see that partitioning D^e over p will give a feasible solution to MRDC-JOIN. Let assume that p is not the optimal solution for MRDC-JOIN, there must exist a lower cost path p' which is different from p (remember that delay partition *does not* affect the path cost). However, according to previous result, p' must

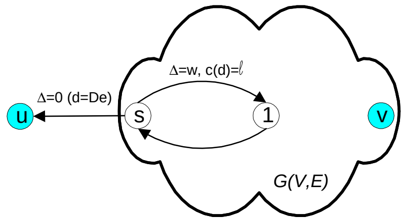

Fig. 1. Problem transformation: adding an edge to $G(V,E)$, assigning link delays, and setting constraints.

also be the optimal solution to RSP, which contradicts assumption that p' has lower cost than p.

Since the transformation is obviously polynomial and MRDC-JOIN is in NP, MRDC-JOIN is NP-complete.■

V. Optimal unicast routing and delay partitioning with delay variation constraint

We first solve the (NP-complete) problem of optimal delay partition and routing with a delay variation constraint that is a sub-problem of MRDC-JOIN. By thinking of each link as a set of links of different delays and costs, the problem can be considered as a multi-constrained optimal path (MCOP) problem [4], hence, it can be solved by heuristics such as H_MCOP [4]. The computational complexities of H_MCOP and the k-shortest path version of H_MCOP are $\mathcal{O}(n\log n + mD)$ and $\mathcal{O}(n\log n + kmD\log kn + (k^2+1)mD)$, respectively.

As will be seen from Section V-C, the performance of H_MCOP for this kind of problem is very poor, which motivates us to find better heuristics. For the integer cost function case, we develop a heuristic that is based on OPQR [6], and for the convex cost function case, we derive from Dynamic-OP-MP [5] a heuristic that is capable of obtaining better results in less time.

A. Integer Cost Functions

In order to accommodate total delay variation requirement Dv, our heuristic algorithm OPQRwVC employs a double-pass approach (see Figure 2). First, in the reverse pass, a Reverse-Dijkstra algorithm is invoked to compute the minimum total delay variation, $\triangle^r_{uv}$, from all other nodes u to node v starting from v (line 1). This information is then used in the forward pass to determine if the heuristic should consider a node, i.e. only a node with foreseen total delay variation that meets the requirement will be considered (line 13). Lines 14-19 update $D(u,i)$, which is the minimum total delay at a node u for a given cost i, and the corresponding total forward delay variation. OPQRwVC also ensures that the link delay is feasible (see lines 11-12). Once the delay requirement is met (line 20), line 21 makes sure that D has been fully partitioned and returns the route and the corresponding delay partition (line 22).

In comparison with OPQR, OPQRwVC requires an additional Reverse-Dijkstra run at the beginning and a simple check in each iteration, therefore, its complexity is $\mathcal{O}(n\log n + m + mU(U+\log D))$.

OPQRwVC$(G, \{\Delta_l, \delta_l, c_l(d)\}_{l\in E}, t, v, D, Dv, U)$

1: $\{\triangle^r_{uv}\}_{u\in V} \leftarrow$ Reverse-Dijkstra$(G, \{\Delta_l\}_{l\in E}, v)$
2: **for all** $u \in V - \{t\}$ **do**
3: $D(u,0) \leftarrow \infty$, $\triangle_{tu} \leftarrow \infty$
4: $D(t,0) \leftarrow 0$, $\triangle_{tt} \leftarrow 0$
5: **for** $i = 1,2,...,U$ **do**
6: **for all** $u \in V$ **do**
7: $D(u,i) \leftarrow D(u,i-1)$, $\triangle(u,i) \leftarrow \triangle(u,i-1)$
8: **for all** $l \in \{(u',u)|(u',u)\in E\}$ **do**
9: **for** $j = 1,2,...,i$ **do**
10: $d_l(j) \leftarrow \min\{d|c_l(d) \le j\}$
11: **if** $d_l(j) < \delta_l + \Delta_l$ **then**
12: $d_l(j) \leftarrow \delta_l + \Delta_l$
13: **if** $\triangle_{tu'} + \Delta_{u'u} + \triangle^r_{uv} \le Dv$ **then**
14: **if** $D(u,i) > d_l(j) + D(u', i-j)$ **then**
15: $D(u,i) \leftarrow d_l(j) + D(u',i-j)$
16: $\triangle_{tu} \leftarrow \triangle_{tu'} + \Delta_{u'u}$
17: **else**
18: **if** $D(u,i) = d_l(j) + D(u',i-j)$ **then**
19: $\triangle_{tu} \leftarrow \min\{\triangle_{tu}, \triangle_{tu'} + \Delta_{u'u}\}$
20: **if** $D(v,i) \le D$ **then**
21: add $D - D(v,i)$ to delay partition of the outgoing link of t
22: return the corresponding route and partition.
23: return FAIL

Fig. 2. The heuristic algorithm for the case of integer link cost functions

B. Convex Cost Functions

The optimal delay partition and routing problem could be solved exactly and efficiently when cost functions are convex by a dynamic algorithm DYNAMIC-OP-MP [5]. In fact, its computational complexity is $\mathcal{O}(mD\log D)$. The reasons for this efficiency and a detailed description of the algorithm are given in [5].

Using DYNAMIC-OP-MP as a basis, we develop a heuristic algorithm OPQRwVC-CC with the same approach used in OPQRwVC. Like DYNAMIC-OP-MP, OPQRwVC-CC has one main function and 3 sub-routines, i.e., INIT, UPDATE, and GET, our sub-routines are almost the same as the original ones except that information regarding the “forward” total delay variation at each node for each total delay is also maintained and updated along with *base* information. In the main function, the foreseen total delay variation is checked before best route information is updated at each node. Overall, the complexity of OPQRwVC-CC is $\mathcal{O}(n \log n + m + mD \log D)$.

C. Performance Evaluation

We compare the performance of H_MCOP ($k = 1..3$), OPQRwVC, and OPQRwVC-CC using the *success rate* (SR), which refers to the fraction of route requests for which feasible routes are found by a given algorithm, and *average route cost* (Avg. Cost).

In our simulation, a 50-node network is generated based on Waxman's model [10]. Each link in the network is assigned two random values representing propagation delay and delay variation. Link cost function of a link is randomly picked from a pool of 3 convex cost functions. The cost functions are chosen to be convex on purpose, as that allows for direct comparison of all algorithms. Any two nodes t and v in the network that are at least 4 hops away from each other will be included in the simulation. Selection of delay constraints is carefully and systematically conducted to ensure that the constraints are sufficiently tight. Table I shows the SRs and average route costs obtained by the algorithms.

TABLE I

SUCCESS RATES AND AVERAGE ROUTE COST OF H_MCOP, OPQRWVC, AND OPQRWVC-CC

Algorithm	Success Rate(%)	Average Cost
H_MCOP ($k = 1$)	25.22	11.9455
H_MCOP ($k = 2$)	45.35	11.9336
H_MCOP ($k = 3$)	52.52	11.8156
OPQRwVC	100	14.9349
OPQRwVC-CC	100	11.7200

As can be seen from Table I, the success rates of both OPQRwVC and OPQRwVC-CC are 100% while those of H_MCOP are only 25.22% for $k = 1$ and up to 52.52% for $k = 3$. The average cost reduction obtained by OPQRwVC-CC is 27.4% and 2% in comparison with OPQRwVC and H_MCOP respectively. The 2% cost reduction of OPQRwVC-CC compared to H_MCOP does not seem to be significant, however, we need to remember the fact that H_MCOP does *not often* find feasible solutions in this experiment.

VI. PROPOSED ALGORITHM FOR MRDC-JOIN

The general idea behind our join algorithm is to determine the optimal (feasible) route from each of the nodes in the current multicast tree (a.k.a, *attachment candidate*) to the new node and simply pick the least cost one. This section will describe the algorithm and discuss the simulation results in the following subsections.

MRDC-JOIN$(G, \{\Delta_l, \delta_l, c_l(.)\}_{l \in E}, D^e, D^i, T, \{d_l\}_{l \in T}, v)$

1: **if** $v \in T$ **then**
2: **if** delay variation constraint is satisfied **then**
3: return OK
4: **else**
5: return FAIL
6: **for all** $t \in T$ **do**
7: Compute D, Dv, U
8: $G \leftarrow G \setminus T \cup \{t\}$
9: ROUTE$(G, \{\Delta_l, \delta_l, c_l(d)\}_{l \in E}, t, v, D, Dv, U)$
10: $G \leftarrow G \cup T \setminus \{t\}$
11: **if** route found **then**
12: return the best route found and its corresponding attachment node t and delay partition.
13: **else**
14: return FAIL

Fig. 3. The heuristic algorithm for the MRDC-JOIN problem

A. Algorithm Description

The centralized version of algorithm MRDC-JOIN is shown in Figure 3. Lines 1-5 handles the case where an in-tree node wishes to join the multicast session. The join is successful if the delay variation constraint is met, but unsuccessful otherwise. If the joining node is outside of the current multicast tree. The current tree *except* the attachment candidate currently under consideration must be temporarily "removed" from the network (line 8) to ensure that the new route will not form a loop with the existing tree. After route computation for a candidate is completed, the network is "restored" to the original (line 10). The ROUTE algorithm in line 9 can be any of the heuristics described in Section V.

Computational complexity of this version is roughly $|V_T|^2$ times the computational complexity of ROUTE, which is quite expensive. However, this complexity can be reduced by performing MRDC-JOIN in a distributed fashion.

In the distributed version, a node that wishes to join a multicast session first checks if it is already in the multicast tree. If that is the case, it will join the multicast group only if delay variation constraint is met. If the node is not currently on the tree, it unicasts a *join-request* message to the source. The source and any in-tree node that receives the request will forward the request message to its downstream nodes along the multicast tree. In the end, all in-tree nodes will receive this request.

Upon reception of the join-request, every in-tree node computes the delay constraints based on its knowledge of the downstream nodes and delay information piggy-backed on the forwarded join-request, computes the best route toward the new node, and if it finds a feasible route, it unicasts a *bid* to the new node. The new node collects all the bids, selects the best one and requests the node that sends the selected bid to connect the new node to the current tree via a *join* message.

The computational complexity of this distributed version is the same as that of ROUTE, although there will be more control overhead associated with the exchanging of messages.

B. Performance Evaluation

We generate 3 networks of 50 nodes, 100 nodes and 200 nodes using the same method described in Section V-C. A source node and a set of nodes are randomly selected from among the network nodes. The sequence in which the selected nodes are going to join the multicast session is randomized. The number of successful joins and total tree cost are recorded. The process is repeated 100 times for each test network. In order to account for the impact of SRs on route cost, tree cost in each run is scaled by $\frac{1}{SR}$. The success rates and the scaled costs are averaged over 100 runs.

We compare the average success rates and average tree costs obtained by algorithm MRDC-JOIN with those computed by a centralized algorithm MRDC-GA [1]. Simulation results are shown in Table II.

TABLE II

COMPARISON OF SUCCESS RATES AND AVERAGE TREE COST OF MRDC-JOIN AND MRDC-GA

Test	**MRDC-JOIN**		**MRDC-GA**	
Network	**Avg. SR(%)**	**Avg. Cost**	**SR(%)**	**Cost**
50 nodes	87.27	76.21	100	52.1836
100 nodes	90	113.77	100	96.4997
200 nodes	88.87	276.39	100	208.217

The results show that MRDC-JOIN is capable of constructing a multicast tree quite efficiently. The costs given by MRDC-JOIN are less than 1.5 times the tree costs provided by MRDC-GA in all cases. In terms of success rates, MRDC-JOIN also performs very well as, on average, more than 87% join requests are successfully handled. The performance of MRDC-JOIN seems to get better when the network size increases.

VII. CONCLUSIONS

In this paper, we have formulated the fundamental problem of delay constrained multicast routing which is connecting a new node to an existing tree under source-to-destination and inter-destination delay variation constraints (MRDC-JOIN). We subsequently showed that MRDC-JOIN is NP-complete.

In simulation, our heuristics OPQRwVC and OPQRwVC-CC outperformed H_MCOP $(k = 1, 2, 3)$, i.e., 100% success rate compared to 52.52% of H_MCOP $(k = 3)$ and the average route costs were also better.

We presented centralized and distributed versions of algorithm MRDC-JOIN. The complexity of the distributed algorithm is the same as the heuristic employed in the algorithm. By means of simulation, we showed that the algorithm is capable of handling join requests efficiently in providing high success rates and good tree costs in all test networks.

For future work, we shall investigate the problem of handling more than one join request at a time and the possibility of incorporating the distributed version of MRDC-JOIN into a multicast routing protocol.

REFERENCES

[1] Hieu T. Tran and Richard J. Harris, "Qos multicast routing with delay constraints," in *International Network Optimization Conference (INOC)*, Evry, France, 2003.

[2] George N. Rouskas and Illia Baldine, "Multicast routing with end-to-end delay and delay variation constraints," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 346–356, 1997.

[3] C P Low and Y J Lee, "Distributed multicast routing, with end-to-end delay and delay variation constraints," *Computer Communications*, vol. 24, no. 9, pp. 848–862, 2000.

[4] T. Korkmaz and M. Krunz, "Multi-constrained optimal path selection," in *INFOCOM 2001*, 2001, vol. 2, pp. 834–843.

[5] Dean H. Lorenz and Ariel Orda, "Qos routing in networks with uncertain parameters," *IEEE/ACM Transactions on Networking*, vol. 6, pp. 768–778, 1998.

[6] Dean H. Lorenz, Ariel Orda, Danny Raz, and Yuval Shavitt, "Efficient qos partition and routing of unicast and multicast," in *IWQoS 2000*, Pittsburgh, PA, USA, 2000, pp. 75 – 83.

[7] Hieu T. Tran and Richard J. Harris, "Near-optimal allocation of delay requirements on multicast trees," in *IFIP Interworking 2002 - Converged Networking: Data and Real-time over IP*, Perth, Western Australia, 2002, pp. 325 – 339, Kluwer Academic Publishers.

[8] R. Hassin, "Approximation schemes for the restricted shortest path problem," *Mathematics of Operations Research*, vol. 17, no. 1, pp. 36 – 42, 1992.

[9] D. Raz and Y. Shavitt, "Optimal partition of qos requirements with discrete cost functions," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2593–2602, 2000.

[10] B.M. Waxman, "Routing of multipoint connections," *Selected Areas in Communications, IEEE Journal on*, vol. 6, no. 9, pp. 1617–1622, 1988.