NEW STATE DEPENDENT MULTICAST ROUTING FOR SINGLE RATE LOSS NETWORKS *

Chi-Chung Cheung, Danny H.K. Tsang
Department of Electrical and Electronic Engineering
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
email: (ecccc, eetsang)@ee.ust.hk

Sanjay Gupta
GSM Products Division, Motorola
1501 West Shure Drive (IL27-3205), Arlington Heights, Illinois 60004, U.S.A.
email: guptasn@cig.mot.com

Abstract

In this paper, we investigate the dynamic multicast routing problem for single rate loss networks and briefly discuss the dynamic multicast routing algorithm called Least Load Multicast Routing (LLMR). We propose a new multicast routing algorithm called Maximum Mean Number of New Calls Accepted Before Blocking Multicast Routing (MCBMR), which can more accurately reflect the current and future loading of a network. Simulation results show that this algorithm, compared with LLMR, not only has a smaller network revenue loss, but also results in smaller call blocking probabilities for all classes of traffic.

1 Introduction

Multicasting refers to the ability of a set of more than two nodes or end-users in a communication network to communicate simultaneously with each other. Applications that require multicast capability (either point-to-multipoint (PTM) as in distributional video or multipoint-to-multipoint (MTM) as in video conferencing, online collaboration and others) will be an integral part of future broadband services. Given the popularity of multicast end-user services and applications, we study the problem of multicast call routing in single rate loss networks. A single rate loss network is one where each connection request requires that a unit capacity be allocated to it on each link that it is routed.

If a route cannot be assigned to a connection request, it is assumed that the call is lost and it does not retry. Examples of single rate loss networks include telephone networks and homogeneous VP-based ATM networks [4].

In the past, the multicast routing problem has been formulated as a Steiner tree problem [5], which is a NP-complete problem [6]. The link cost is set as either end-to-end delay [7] or the copies of a packet delivered [8] in packet switching networks. Moreover, the routing decision is made beforehand and does not change for a long time.

Recent research efforts are focused on dynamic multicast routing, which does not decide routes to use beforehand, but takes into account the state of the network at the instant a connection request arrives to assign a route. The scenario is a circuit-switched network or a virtual-circuit based packet switched network where all connection requests have the same bandwidth requirement for each link routed. The arrival process of a connection request is assumed to be a random process. Each connection requests to set up a multicast connection among a set of randomly chosen nodes. Each connection request, if accepted, departs from the network after a random amount of time. The objective is to either maximize the network revenue or minimize the network revenue loss, which will be defined in Section 2.

An obvious approach to perform an optimum routing algorithm is to find an appropriate state descriptor for the network and model the evolution of the network state as a Markov process (after making suitable simplifying assumptions). However, the routing problem with the above solution

*This research was supported in part by the Hongkong Telecom Institute of Information Technology grant: HK-TIIT 93/94.EG01.

is computationally impossible because of too many states to be considered. Consider the case of a symmetrical network with 10 links in which each link can carry a maximum of 9 connections. Assume that the state descriptor used is the occupancy of each link. In this unreasonably small example, with an incomplete state descriptor, the size of the state space is 10^{10} , large enough to render the above mentioned approach useless.

Based on the above problem, the following heuristic approach is employed: when a multicast connection request arrives, only the links that connect to the destination nodes will be considered. We assign a link cost function to each link and then search for a minimum spanning tree for the connection request. If a minimum spanning tree is available, we accept the request and route the call using the tree; otherwise, we will allow one alternative node and restart the searching. If a minimum spanning tree is still not available, the call request will be rejected; otherwise, we accept the request and route the call using the tree. Some previous research works in the area of dynamic multicast routing schemes in single rate loss networks are reported in [3] and [1] as Least Load Multicast Routing (LLMR). An analytical model is provided in [2] for symmetrical networks. The LLMR algorithm is to find a connected tree for the multicast call by applying minimum spanning tree searching with a link cost function equal to the negative value of the number of free circuits, i.e. the current residual capacity. Note that finding the minimum spanning tree is equivalent to find a multicast connected tree with maximum number of free circuits. By considering the loading of links involved, LLMR chooses the links with the least loading to establish the connection.

In this paper, we will employ the same heuristic approach, the minimum spanning tree searching, because the heuristic approach is simple and we concentrate instead on developing a new link cost function to optimize the network performance. We consider a new link cost function, the mean number of new calls accepted before blocking (MCB), which can more accurately reflect the current and future loading of a link, compared with LLMR. The paper is organized as follows. We begin by presenting in Section 2 the new link cost function, which can result in better network performance compared with LLMR. We will discuss why it can more accurately reflect the loading of a link. The network model and the multicast routing algorithm will also be presented in this section along with simple examples. We will also describe how to determine the new link cost function. Section 3 shows the numerical results of our proposed algorithm and

compares its performance with LLMR. Finally, we conclude in Section 4 by summarizing the paper and outlining avenues for future research.

2 The Mean Number of New Calls Accepted Before Blocking (MCB)

Before proceeding further, we need to introduce the parameter to measure the performance of a network in multicast routing. In point-to-point communication, the end-to-end call blocking probability is the most common measure of quality of a network. However, there are more than one class of call connection requests (a class here means a group of multicast call connections with the same destination size) and thus the above measure cannot be directly applied in multicast routing. For evaluating the performance of multicast routing algorithms, we use a performance measure called normalized revenue loss, which was proposed in [1]. We set the corresponding revenue of a call connection request with destination set of size d to $d - 1$ (i.e. the number of direct links used) and thus the expression of the normalized revenue loss is

$$\frac{\sum_{d=2}^D (d-1)\lambda_d B_d}{\sum_{d=2}^D (d-1)\lambda_d}, \quad (1)$$

where D is the maximum destination size of multicast call connections, λ_d is the call arrival rate of class d and B_d is the call blocking probability of class d .

From the above equation, in order to maximize the normalized network revenue or minimize the normalized revenue loss, we need to minimize the call blocking probabilities of all classes. The intuitive reason of Least Load Multicast Routing (LLMR) to consider the number of free circuits as the link cost function is that this cost can reflect the current loading or resources of a link. The number of free circuits of a link indicates currently how many circuits are available and, based on this information, LLMR can evenly distribute the loading of a network to minimize the call blocking probability. However, the LLMR approach does not take into account the future call arrivals of the network and thus sometimes cannot make a good decision. In this section, we propose a new link cost function, the mean number of new calls accepted before blocking, which can more accurately reflect the current and future loading of a link. Its value reflects more accurately the actual available network resources that one can make use of before blocking. Consider the following case: a multicast

call request arrives to a single rate loss network and, during the routing decision, a link will be chosen from two links whose information is shown in Table 1. Both the call interarrival time and the call holding time of the two links are exponentially distributed. The current link offered loads of *link 1* and *link 2* are 45 circuits/unit and 40 circuits/unit respectively. Based on the number of free circuits available, LLMR will choose *link 1* because the number of free circuits of *link 1* (5 circuits) is more than that of *link 2* (2 circuits). However, by considering the value of *MCB* (the mean number of new calls accepted before blocking), the new link cost function shows that *link 1* on average can accept 60.4 new calls before blocking while *link 2* can accept 75.5 new calls before blocking. Therefore, based on the values of *MCB*, *link 2* will be chosen instead of *link 1*. This example shows that the number of free circuits alone cannot accurately reflect the actual available resources of links.

<i>link information</i>	<i>link 1</i>	<i>link 2</i>
link capacity	50 cir.	50 cir.
call holding time	1 unit	1 unit
current link offered load	45 cir./unit	40 cir./unit
number of free circuits	5	2
<i>MCB</i>	60.4	75.5

Table 1: The link information of two links

2.1 The Network Model and The Proposed Routing Algorithm

Before proceeding further, we would like to introduce necessary definitions and notations. We consider a single rate loss network to be an undirected graph $G = (V, E)$ where V and E are the set of nodes and links respectively. Throughout we denote by $R(e)$, $e \in E$, the number of free circuits, or residual capacity, of link e . Denote by $\hat{M}(e)$, $e \in E$, the mean number of new calls accepted before blocking for link e .

Consider a connection request c with destination set $S(c)$ (each node $s \in S(c)$ is referred to as a *destination* of c). Let $d(c)$ be the number of nodes involved in a connection request c , i.e. destination size. A connection c requires a *connected, acyclic* graph (tree) $T(c) = (V(c), E(c))$, where $S(c) \subseteq V(c)$ and a unit bandwidth is reserved on each link $e \in E(c)$. Nodes $s \in V(c) - S(c)$ are referred to as *alternative* nodes. A link, $e \in E(c)$, is referred to as

an *alternative* link if it connects to an *alternative* node. Let $t(e)$ denote the reservation threshold for link e , i.e., link e can only be used as an *alternative* link if $R(e)$ exceeds the reservation threshold $t(e)$.

As direct links are always tried first, our proposed algorithm will attempt to build a connected tree $T(c)$ which contains all nodes in $S(c)$ and its cost $\alpha(T(c)) = \sum_{e \in E(c)} \alpha(e)$ is minimum compared with all possible connected tree. The cost function of link e , $\alpha(e)$, is defined as the negative value of $\hat{M}(e)$. The finding of a minimum spanning tree is thus equivalent to finding a connected tree with maximum value of *MCB*. If a connected tree cannot be established (i.e. all the destination nodes cannot be connected by direct links only), our proposed algorithm will attempt to build a connected tree with one additional node (i.e. an *alternative* node) and the cost function of link e (either direct or *alternative*) is defined as

$$\alpha(e) = \begin{cases} -\hat{M}(e) & R(e) > t(e) \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

We select a connected tree with minimum cost from all possible cases. If a connected tree still cannot be established, the call connection request will be rejected. Our proposed algorithm is given in Figure 1.

Four points need to be clarified. (i) A link cannot be selected if it does not have any free circuits. Note that the value of $\hat{M}(e)$ will be zero if link e does not have any free circuits. (ii) A connected tree cannot be established if there exists some unconnected destination nodes. (iii) We allow at most one *alternative* node when no connected tree with direct links only is found. It is because each additional *alternative* node requires one additional link to form a connected tree and this wastes network resources. In addition, the routing complexity increases exponentially as the number of *alternative* nodes increases. (iv) The dynamic add/drop procedures are not shown in the above algorithm because they are quite straightforward. If a node requests to join a connected tree, it should choose a link with minimum link cost to connect from the node to the tree. For simplicity, if there is no direct link available, the request will be denied. For the procedure to drop a node, if there is only one link connected from the node to its connected tree, the link will be released; otherwise, nothing is changed.

2.2 Illustrative Example

To illustrate the routing algorithm outlined above, we consider a 4-node network and a connection request with destination set $\{r, a, b, c\}$ with node r

begin

Construct a minimum spanning tree (MST)
 $T(c) = (S(c), E(c))$ with $\alpha(\epsilon) = -\hat{M}(\epsilon)$.

if a MST cannot be established,

Set $\alpha_{min} = 0$.

for each $s \in V(c) - S(c)$ (alternative node),

Construct a MST $T(c) = (S(c) + s, E(c))$
 with the link cost defined in (2).

if a MST is established and $\alpha_{min} > \alpha(T(c))$,

Set $\alpha_{min} = \alpha(T(c))$ and $T_{best} = T(c)$.

endif

end for

if a MST cannot be established,

the call connection request is rejected;

else

the tree is T_{best} and done.

endif

endif

end

Figure 1: The MCBMR algorithm

as the source node. The steps in the connection establishment procedure are as shown in Figure 2 — the number next to a link denotes its link cost, the negative value of MCB, and links that are included in the set $E(c)$ are drawn with “thick” lines. Node c is the first one to be added as link (r, c) has a link cost of -3 that is smaller than the link costs of links (r, a) and (r, b) which is followed by node a (link (c, a) is chosen due to its smaller link cost over links (r, a) , (r, b) and (c, b)). Finally, we choose link (a, b) because the link cost of (a, b) is the smallest compared with links (r, b) and (b, c) . After iteration 4, the procedure is completed and a tree is established for the connection request.

Figure 3 shows the operation of selecting an alternative node. Given that each link’s residual capacity exceeds the trunk reservation threshold, except link (r, a) . Since there are no free circuits on link (r, a) (assume a link in state 0 is full), two available two-link alternative routes are considered to connect nodes r and a : (r, b, a) and (r, c, a) . The cost of (r, b, a) is -7, which is larger than that of (r, c, a) (its cost is -9). Thus we choose node c as the alternative node and (r, c, a) as the two-link alternative route.

Figure 2: Example to illustrate our proposed algorithm without alternative nodes

Figure 3: Example to illustrate our proposed algorithm with an alternative node

2.3 The Implementation Issues

To obtain the value of the mean number of new calls accepted before blocking (MCB), the obvious approach is to measure the value directly. However, the method of direct measurement cannot frequently update the MCB values and can updated them only when a link is blocked. Thus the direct measurement method cannot accurately determine the current MCB values. In this section, we introduce an indirect measurement method to obtain the MCB values.

Consider a fully connected single rate loss network. When a multicast call request arrives, a set of direct links will be considered and, if available, a connected tree is established among them for the call. We assume all multicast call connection requests are Poisson arrivals and the holding times of accepted calls are exponentially distributed. With the link independence assumption, the queueing model of a link can be modeled as a general M/M/N/N queue as shown in Figure 4, where $1/\mu$ is the mean holding time of calls and N is the total number of circuits of the link. In addition, given that there are k occupied circuits in the link, we define λ_k and M_k as the state dependent arrival rate at state k and the mean number of new calls that can be accepted starting from state k before reaching state $k + 1$, respectively. By examining the possible state transitions from state k , it is not difficult to see that M_k satisfies the

first-order difference equation

$$M_k = \frac{\lambda_k}{\lambda_k + k\mu} + \frac{k\mu}{\lambda_k + k\mu} [M_{k-1} + M_k] \quad (3)$$

with the initial condition $M_0 = 1$. Equation (3) can be simplified as

$$M_k = \frac{k\mu}{\lambda_k} M_{k-1} + 1 \quad (4)$$

and thus M_k can be easily obtained by the above recursive equation. Let $\hat{M}_i$ be the mean number of new calls that can be accepted starting from state i before blocking, i.e. the mean number of new calls accepted starting from state i before reaching state N . It is easy to observe that $\hat{M}_i$ equals to the sum of M_k , $k = i, i+1, \dots, N-1$, i.e.

$$\hat{M}_i = \sum_{k=i}^{N-1} M_k. \quad (5)$$

Therefore, we can compute the link cost by knowing the state dependent arrival rates, which can be determined by measuring the mean state sojourn time.

Figure 4: General M/M/N/N queueing model

3 Numerical Results

In this section, we give numerical results to illustrate the performance of the MCBMR algorithm when compared with Least Load Multicast Routing (LLMR) proposed in [3] and [1]. We consider a fully connected network with 8 nodes. The capacity of each link is randomly generated between 40 and 60 bandwidth units or circuits. The traffic matrix of multicast connection requests is randomly generated. There are three classes of call connection requests in the network and their destination sizes are 2, 3 and 4. The loadings of the three classes are equal and the call arrivals of each class follow a Poisson Process. All call holding times are exponentially distributed with unit mean. All processing time, including call setup and release time, are negligible. When a call connection request is rejected, it will not retry and is cleared immediately from the network. For each set of simulation point, the length of each run is 10^6 units of mean

interarrival time of connection request and the initial 10% is discarded to avoid the effect of transient state. The vertical lines about each point represent the 95% confidence intervals. We define the network loading as the ratio of the total offered load to the total network capacity and set 80% as the engineered load (i.e. 0% overload). For alternative routing, we set the trunk reservation threshold of each link as 5 circuits. We use a performance measure called normalized revenue loss introduced in Section 2.

Figures 5 and 6 show the mean call blocking probabilities for LLMR and MCBMR under the same range of overload conditions for the cases of no alternative nodes and at most one alternative node respectively. We find that MCBMR always outperforms LLMR in terms of achieving lower call blocking probabilities for all three classes. It is because LLMR just captures the current information (the current loading of links) but MCBMR makes use of all information (the current state of the network and the state dependent arrival rates of links) to forecast the future congestion.

The relative improvement of the normalized revenue loss of MCBMR over LLMR are shown in Figures 7 and 8. We find that the relative improvement can be up to 50% and, as we expect, it decreases when the overload increases. It is because, when the network utilization becomes saturated, the network resources are insufficient to accept incoming calls and hence no algorithm can improve the network performance under this condition.

4 Conclusions

In this paper, we have investigated the dynamic multicast routing problem for single rate loss networks. We proposed a new dynamic multicast routing algorithm called Maximum Mean Number of New Calls Accepted Before Blocking (MCBMR) multicast routing. The goal is to search for a connected tree with maximum mean number of new calls that can be accepted before blocking. Through simulations, we find that MCBMR always outperforms LLMR in all cases.

References

- [1] C.C. Cheung, H.W. Chu, D.H.K. Tsang, and S. Gupta. Least loaded first multicast routing in single rate loss networks. In *Proceedings of the IEEE International Conference of Communications*, pages 513–517, 1996.

- [2] C.C. Cheung, D.H.K. Tsang, and S. Gupta. State dependent multicast routing for single rate loss networks. In *Proceedings of the International Conference on Telecommunications*, pages 1425–1430, 1997.
- [3] R. H. Hwang. Adaptive multicast routing in single rate loss networks. In *Proceedings of the IEEE INFOCOM*, pages 571 - 578, 1995.
- [4] S. Gupta, K. W. Ross, and M. El Zarki. *On Routing in ATM Networks*. Prentice-Hall and Manning, 1995. Routing in Communications Networks.
- [5] S. Hakimi. Steiner's Problems in Graphs and its Implications. In *Networks*, vol. 1, pages 113 - 133, 1971.
- [6] R. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85 - 103, 1972.
- [7] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos. Multicast Routing for Multimedia Communications. In *IEEE/ACM Transactions on Networking*, vol. 1, pages 286 - 292, June 1993.
- [8] R. Tode, Y. Sakai, M. Yamamoto, H. Okada, and Y. Tezuka. Multicasting Routing Algorithm for Nodal Load Balancing. In *Proceedings of the IEEE INFOCOM*, pages 2086 - 2095, 1992.

Figure 5: Mean call blocking probabilities of MCBMR and LLMR (no alternative node is allowed)

Figure 6: Mean call blocking probabilities of MCBMR and LLMR (at most one alternative node is allowed)

Figure 7: Relative improvement of revenue loss of MCBMR over LLMR (no alternative node is allowed)

Figure 8: Relative improvement of revenue loss of MCBMR over LLMR (at most one alternative node is allowed)