Multicast Routing in VP-based ATM Networks [1]

Ren-Hung Hwang   Hwa-Shing Huang

Department of Computer Science
& Information Engineering
National Chung Cheng University
Chia-Yi, Taiwan, R.O.C.
{rhhwang,hhs82}@cs.ccu.edu.tw

Abstract

The introduction of new features, such as quality of service (QOS) requirements, multipoint connections, virtual path (VP) concepts, into ATM-based broadband networks poses many technical challenges. In particular, new routing algorithms need to be developed to adapt to these new features. In this paper, we focus on designing *adaptive multicast* routing algorithms which takes these new features into consideration. We also propose a new performance metric, referred to as *fractional reward loss*, for evaluating multicast routing algorithms.

[1]This research was supported in part by National Science Council of Taiwan, R.O.C., under contract NSC-83-0408-E-194-006-T.

1 Introduction

Asynchronous transfer mode (ATM) has been proposed by International Telecommunications Union (ITU), formerly CCITT, as the target technology for future broadband networks [1]. Based on the cell switching technology, an ATM network is able to handle multiple classes of traffic with different traffic characteristics and quality of service (QOS) requirements. Furthermore, by allowing statistically multiplexing of cells, an ATM network is very flexible and efficient for providing such broadband services.

Many characteristics of ATM networks make it very suitable for future broadband networks. First, to reduce processing complexity at switches, ATM networks provides a connection-oriented, cell-switching environment. In addition, a virtual path (VP) concept has been proposed to further simplify call setup procedure, traffic control, and resource management [2, 3]. Second, in order to carry a broad range of traffic classes, such as real-time, multimedia applications, a guaranteed quality-of-service (QOS) is provided by ATM networks when a connection is set up. To cope with traffic heterogeneity, different applications may request for different QOS's. Finally, to meet the requirement of some new applications, such as multiparty video conferencing, multi-point (multicast) connections will be provided through ATM switches with multicast capability.

While tremendous research has been focused on many aspects of ATM networks, routing in broadband networks has been a much less well-studied problem, especially in the presence of multicast connections. Although the problem of routing in both packet-switched and circuit-switched networks has been well studied, it has been shown that routing in ATM networks will be very different from routing in traditional networks [4]. Especially, according to the ITU specifications on future Broadband ISDN [1], the need to provide a guaranteed QOS in ATM networks has resulted in the need for a *call-level* admission control mechanism and the reservation of resources (e.g., *effective bandwidth* [5]) by a call on a link-by-link (or VP-by-VP) basis. Therefore, it is observed in [4] that routing in ATM networks is more similar to routing in circuit-switched networks than that of packet-switched networks, even the ATM networks is packet(cell)-switched networks. On the other hand, many characteristics of ATM networks, such as carrying multiple classes of service with different QOS requirements and the use of VP, have made routing in ATM networks becomes considerably more complicated than that of circuit-switched networks.

In this paper, we focus on designing multicast routing algorithms for ATM networks. We assume that each multicast connection is associated with a QOS requirement. The QOS of a multicast connection is defined to be the end-to-end QOS to be satisfied from the source node to any of the destination nodes. As in [6--8], we limit our study to the multicast routing problem in homogeneous

VP networks. A *homogeneous* VP network carries only one type of traffic and all connections in the network require a homogeneous end-to-end QOS guarantee. In this study, we also assume that each VP is associated with dedicated resources (bandwidth, buffer space, ..., etc.). The amount of resources allocated to each VP is determined periodically over a relatively much longer period of time as compare to the multicast connection setup time.

We are interested in developing adaptive multicast routing algorithms in which routing decision is made based on network state information. Four multicast routing algorithms are proposed in this paper. We first develop a baseline multicast routing algorithm based on the Least-Loaded Routing (LLR) concept, a well-known routing approach in circuit-switched networks. The second algorithm is based on the LLR concept and the dynamic VP capacity sharing concept which was originally proposed in [7] to improve the performance of point-to-point routing algorithms for VP-based ATM networks. In [9], an *intermediate exit* feature of a VP switch was proposed for multicast routing in VP-based ATM networks. Our third algorithm is designed for VP networks with the intermediate exit feature. The last algorithm combines the dynamic VP capacity sharing concept and intermediate exit feature. Our simulation results show that the intermediate exit feature improves the performance of the multicast routing algorithm significantly while the dynamic VP capacity sharing concept only improves the performance slightly.

Point-to-point routing in ATM networks which takes QOS requirement and the VP concept into account has been studied in [6--8]. In these works, adaptive routing algorithms based on approaches commonly used in circuit-switched networks were developed for *homogeneous* VP networks. In [6], the authors developed several adaptive routing algorithms based on the LLR concept. Besides the *homogeneous* VP networks assumption, the authors also assumed that a VP only provides single class of QOS to all virtual channels (circuits) carried on that VP. A more general network model was studied by [7, 8] in which multiple classes of QOS are provided on a VP and two different VP resource reservation strategies were studied. In [7], the authors formulate the routing problem as a Markov decision process (MDP) and propose several MDP-based routing algorithms. Hwang then, in [8], proposes LLR-based routing algorithms under the same network model.

Multicast routing in VP-based ATM networks has been studied in [9] in which the routing problem is formulated as a cost optimization problem. The objective of [9] is, for a given multicast connection, to minimize a cost function which consists of bandwidth cost, connection establishment cost, and switching cost. Since all these costs are deterministic, the resulting algorithm is static not adaptive, i.e., is independent of the current network state (e.g., traffic loads, VP occupancy). Our research differs from this work in the following respects. First, we are considering *adaptive* multicast routing algorithms which determine multicast routing tree based on network state information. Second, we assume that each multicast connection is associated with a QOS requirement. Finally, our objective is to maximize the long run overall network resources utilization.

The remainder of this paper is organized as follows. In section 2, we introduce the homogeneous VP networks we are interested and examine the multicast routing algorithms in presence of VP and QOS requirements. In section 3, we propose four multicast routing algorithms. Performance evaluation of these four routing algorithms is shown in section 4. Section 5 summaries the paper.

2 Network Model and Problem Definition

2.1 A Homogeneous Virtual Path Network

In the ITU specifications on future Broadband ISDN (B-ISDN), the Virtual Path (VP) concept has been proposed to simplify traffic control, reduce nodal complexity, and ease resource management [2, 3]. A VP typically carries a bundle of virtual circuits (VC's) and is routed over a set of adjacent physical links. Since VC's belong to the same VP have the same end-points, all VC's on a VP can be transported, processed, and managed in a similar manner. Thus, nodal processing complexity can be reduced by the use of VP. For example, all VC's on a VP can share the same routing table such that establishment of a VC as well as maintenance of routing tables can be simplified.

Processing requirements for setting up a VC can be further substantially decreased by reserving resources on VP's. In ATM networks, certain QOS guarantee is provided when a VC is established. Without reserving dedicated resources on VP's, call setup need to be done on a link-by-link basis in order to determine whether the QOS requirement of the new arriving call can be guarantee as well as continue to meet the QOS guarantees of all existing calls. However, when resources are reserved on VP's, resource management and QOS guarantee are maintained at the VP level. Therefore, a new VC can be established by executing an admission control function at the source node of a VP without requiring any call processing at transit nodes along the VP. In this paper, we refer to the resources reserved on a VP as the capacity of the VP [7]. Reserving capacity on VP's can be done deterministically or statistically [7]. The amount of capacity to be reserved on each VP can also be determined periodically [10, 11] or dynamically (on-demand) [12, 13]. The issues of how to reserve capacity on VP's and when to reserve is beyond the scope of this paper. In this paper, we assume that capacity is reserved on VP's deterministically, i.e., dedicated resources is reserved for each VP and no multiplexing is allowed between different VP's. (However, multiplexing among VC's within a VP is still allowed [7].) We also assume that the amount of capacity reserved on each VP is determined periodically, over a relatively much longer period of time than the VC set up time. In other words, from the routing point of view, each VP is allocated with a fixed amount of dedicated capacity.

Another advantage of using VP's for traffic control is to segregate traffic types which requiring different QOS's. Future B-ISDN is expected to handle many types of broadband service, such as video conferencing, broadcast audio, video-on-demand, ..., etc. Traffic of different types of broadband service may have very different traffic characteristics and QOS requirements. How to multiplex different broadband traffic over a physical link while providing different QOS guarantees to each traffic is still an open question. However, the problem can be much simplified if different types of traffic are separated by assigning a VP with dedicated resources to each type of traffic. Therefore, in this paper we assume that VP's are used for segregating different types of traffic. Furthermore, we will limit our study to a homogeneous VP network in which all VP's carry the same type of traffic and all multicast connections require the same end-to-end QOS guarantee.

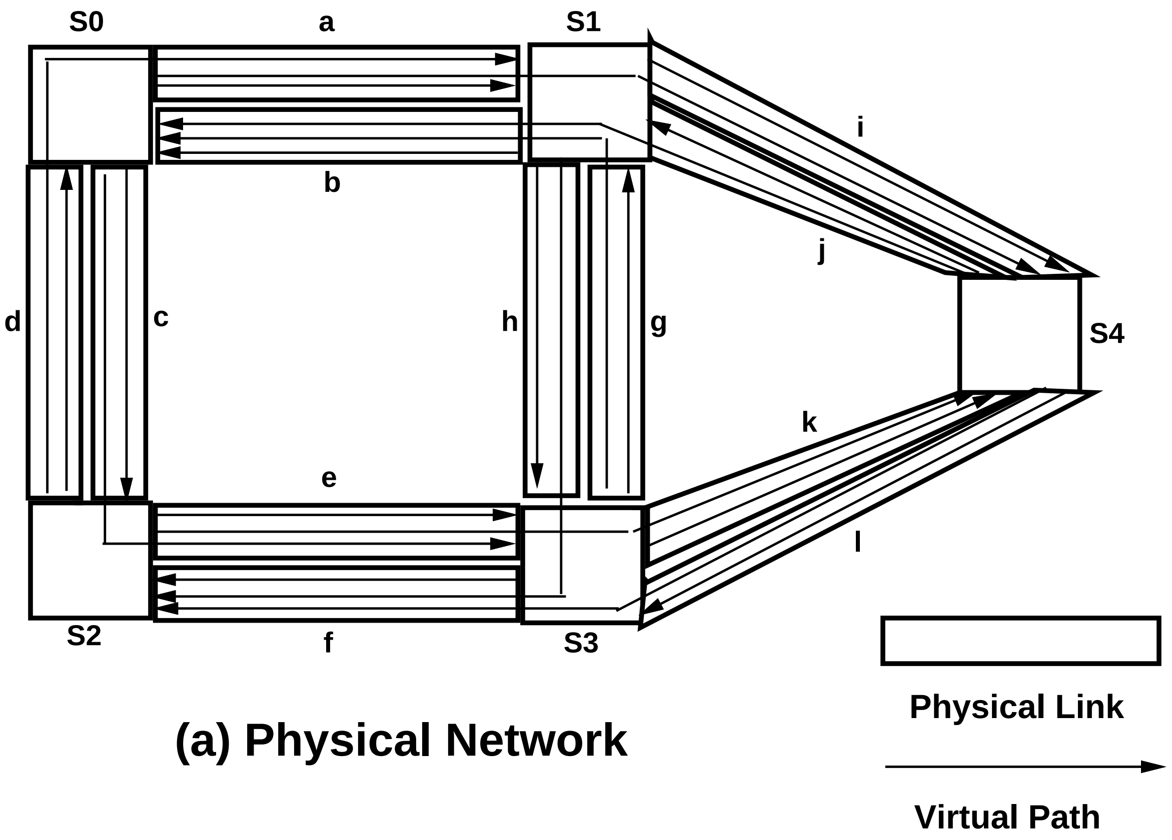
A homogeneous VP network can be modelled as a fully connected network with N nodes and $N(N-1)$ links by viewing the VP end-points as nodes and VP's as links. For example, let us consider the physical network shown in Figure 1(a), which consists of 5 switches (nodes) and 12 unidirectional physical links. If a VP is assigned to each source-destination pair, as shown in Figure 1(b), an 5-node fully connected VP network can be formed which consists of 20 unidirectional *logical links* (VP's). The capacity of each link in the homogeneous VP network is the amount of resources reserved on the corresponding VP. (Note that the capacity may be different for each link and can be zero.)

2.2 Problem Definition

Let us model a homogeneous VP network as a directed graph $G = (V, E)$ with node set V and unidirectional link (VP) set E. As stated in the previous section, the network is fully connected and each link (VP) is allocated a fixed amount of dedicated capacity.

We can characterize a multicast connection request by following parameters:

- The source node of the connection, s, $s \in V$.
- The set of destination nodes, D, $D \subset V$.
- End-to-end QOS requirement $Q(\delta, \epsilon)$ (to be defined).
- Reward (revenue) for carrying the connection in the network, $r_{(s,D)}$ (to be defined).



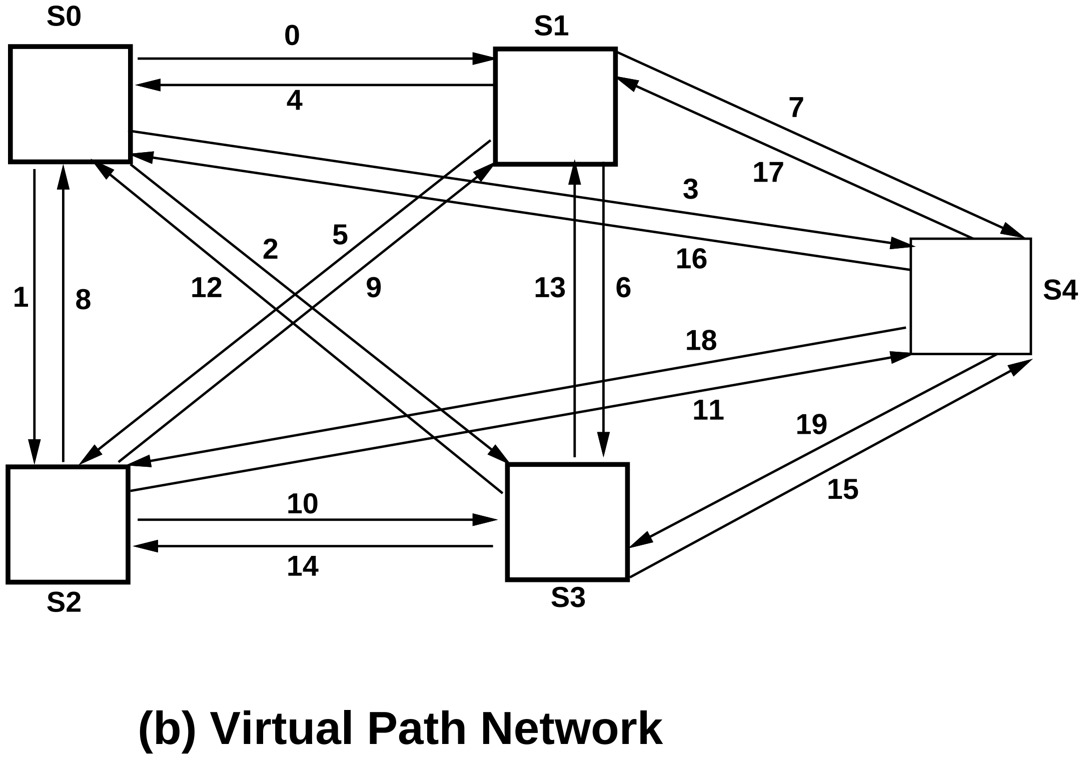


Figure 1: Example: A 5-node network and its corresponding VP network.

Connections are classified according to their source node and set of destination nodes such that connections within the same class have the same parameters $(s, D, Q(\delta, \epsilon), r_{(s,D)})^2$. Let W be the set of all possible classes of multicast connections.

To set up a multicast connection, a tree $T = (V_T, E_T)$, referred to as a multicast tree, is constructed where V_T is the set of nodes on the tree, $\{s\} \bigcup D \subseteq V_T$, and E_T is the set of links on the tree. The problem of multicast routing is to construct a multicast tree, T, such that the long run overall network resources utilization can be maximized under the constraint that an end-to-end QOS, $Q(\delta, \epsilon)$ must be satisfied from the source node to any of the destination nodes.

The QOS requirement of a multicast connection may be specified in many ways, depending on the application's requirement. For example, file transfer applications may require no cell loss while video and audio applications may tolerate a small amount of cell losses but require low cell delay and jitter. In this paper, we assume the QOS criterion of a multicast connection is specified by $Q(\delta, \epsilon)$, which means that no more than a fraction, ϵ, of cells received by any of the destination nodes should experience more than D seconds of queueing delay. (A lost cell is considered as a cell with infinite queueing delay.)

In order to guarantee the end-to-end QOS from the source node to any of the destination nodes for a multicast connection, certain amount of bandwidth must be reserved, explicitly or implicitly, on each link on the multicast tree when the tree is constructed. The amount of bandwidth required on each link depends on the traffic characteristics, the end-to-end QOS requirement, and the length of the path traversing the link. In our numerical example, we assume a traffic source can be characterized by a two-state fluid-flow model [14,5] and is described by three parameters: its peak rate, r (bps); utilization, ρ, i.e., the fraction of time the source is at ''on'' state; and the mean ''on'' period, b (seconds) [5]. We also assume that the traffic characteristics of a multicast connection are unaffected as the traffic travels along the tree [15,7].

The QOS required on each link on a multicast tree is determined by the equal allocation policy [15]. For example, consider the multicast tree in Figure 2 in which s is the source node and $\{d_1, d_2, d_3\}$ is the set of destination nodes. If the QOS required by the multicast connection is $Q(\delta, \epsilon)$, then the QOS required on VP_1 is also $Q(\delta, \epsilon)$ since the path from s to d_1 consist of only one VP. However, if the path consists of more than one VP, e.g., the path from s to d_3, then the end-to-end QOS requirement needs to be decomposed into several ''local'' QOS requirements which must be satisfied on each VP on the path. The problem of how to decompose the end-to-end QOS requirement into local QOS requirements optimally is beyond the scope of this paper. We assume that the equal allocation policy [15] is adopted which assigns an equal amount of the end-to-end

[2]Recall that we assume all connections have the same QOS requirement. Here, we also assume that connections with the same source node and set of destination nodes have the same reward parameter.

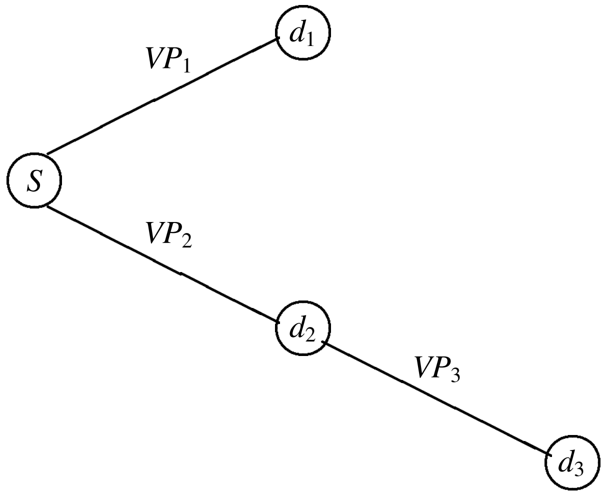


Figure 2: A simple multicast tree.

QOS to each VP on the path. That is, a QOS of $Q(\delta/2, \epsilon/2)$ is required on VP_2 and VP_3 in the example in Figure 2.

Providing multiple classes of QOS on each VP are required since a VP may be part of many multicast trees and paths that traversing the VP may consist of different number of VP's. The ADAPTIVE scheme proposed in [7] is adopted in this paper for providing multiple classes of QOS on a VP. In the ADAPTIVE scheme, either the most stringent QOS is provided to all VC's on the VP or multiple classes of QOS are provided via allocating dedicated resources to each class of QOS. Which one of the approaches is adopted depending on which approach is more efficient, in terms of the maximum number of VC's can be supported. The algorithm for determining which approach to adopt and the maximum number of VC's of each QOS class can be supported on a VP is given in the appendix A (see also [7]).

Finally, we propose a new performance metric, referred to as *fractional reward loss* [16], for evaluating multicast routing algorithms. In the past, performance of multicast routing algorithms has been evaluated by comparing the costs of the multicast trees found by the multicast routing algorithms to the that of the ''optimal solution'', i.e., the minimum cost of all possible multicast trees. However, without carefully defining the cost function of each link, we hardly can gain much insight from such comparison. Thus we define a new performance metric for evaluating multicast routing algorithms for loss networks. Formally, the fractional reward loss of a network is defined as

$$\text{fractional reward loss} = \frac{\sum_{w \in W} r_w \lambda_w B_w}{\sum_{w \in W} r_w \lambda_w}$$

where λ_w is the arrival rate of class w traffic and B_w is the blocking probability of class w traffic. Minimizing the fractional reward loss is equivalent to maximize the expected revenues produced by the network.

We are interested in adaptive multicast routing algorithms, i.e., those algorithms that use network state information to make routing decisions. We assume that global network state information is available instantaneously whenever it is needed by the routing algorithms. The objective of a routing algorithm is to minimize the fractional reward loss. Thus, a connection is carried on the most ''efficient'' multicast tree (based on some routing rules using current state information). Otherwise, the connection is rejected (even if there is enough network resources to carry it).

3 Multicast Routing Algorithms

When resources are reserved on VP's, routing in VP networks is very similar to routing in circuit-switched networks. Each virtual path can be viewed as a logical link with allocated capacity. Thus, as shown in Figure 1, the logical view of a VP network can be fully connected even though the physical network itself is sparse. Furthermore, when distinct VP's are used for separating different types of traffic, routing in homogeneous VP networks is also very similar to routing in traditional single-rate circuit-switched networks. Therefore, in this section, we study how routing algorithms in circuit-switched networks can be applied to VP networks.

Existing adaptive point-to-point routing algorithms in circuit-switched networks can be classified into two categories: *Least-Load path Routing*-based (LLR-based) [17--19] and *Markov Decision Process*-based (MDP-based) [20, 21]. The LLR approach tries to route an incoming call to the direct link, the link between the source node and destination node, first. If the call is blocked (because of no free circuits), the ''least busy'' path is then tried. In our approach, we will follow the LLR routing policies defined in [18, 22, 23, 6] in which the least busy path is defined as the path that has the maximum number of free circuits. (Note that this definition is different from the Lease-Busy-Alternative (LBA) policy define in [24].) The MDP approach, on the other hand, formulates the routing problem as a Markov decision process. Although the MDP approach has been shown to be more efficient than the LLR approach in other context [16, 25], the computational complexity required for the MDP approach is much higher. Thus, we will only focus on LLR-based multicast routing in homogeneous VP-based ATM networks.

In the past, the multicast routing problem has been formulated as a cost optimization problem, also known as the Steiner tree problem [26], and is known to be NP-complete [27, 28]. Thus most previous researchers have focused on developing heuristic algorithms that take polynomial time and produce near optimal results (e.g., [29, 30]). In formulating the multicast routing problem as a Steiner tree problem, all researchers have assume that each link is given a predefined cost (i.e., how the link costs are defined is not their concern). The development of multicast routing algorithms in VP networks is much more complicated than these previous works because of the

QOS constraint, (i.e., each multicast connection is associated with an end-to-end QOS requirement which needs to be satisfied between any source-destination pair), and the coupling with the physical network configuration (e.g., the dynamic VP capacity sharing concept [7] and the intermediate exit feature [9]). In the following, we propose a new approach, which is based on the LLR approach, on designing adaptive multicast routing algorithms.

In our development of multicast routing algorithms, we will follow the terminology and empirical rules in LLR routing in circuit-switched networks. For example, a destination node will be connected to the source node by first trying the VP between the source node and that destination node, which we refer to as the direct VP between the two nodes. If the destination node cannot be connected through the direct VP (due to QOS constraint), an alternate path, which consists of two VP's, with the maximum free capacity is then tried. As stated in section 2, a more stringent QOS is required on each VP on the alternate path in order to meet the end-to-end QOS requirement. Therefore, in order to accommodate alternate calls (i.e., calls routed on alternate paths), two classes of QOS are established on each VP; class 1 for alternate calls and class 2 for direct calls. Since the alternate path consists of at most two VP's, the length of each path in the multicast tree, which connects the source node to a destination node, is also limited to be at most two. The rational for such constraint is that the longer the path the more stringent QOS, and thus more capacity, is required to set up the call on each VP on the path.

In the following, we propose four multicast routing algorithms. The first algorithm extends the LLR concept to the multicast routing problem in the homogeneous VP-based ATM networks. The second algorithm adds the dynamic VP capacity sharing concept [7] to improve the performance of the first algorithm. The third algorithm considers the ATM networks with the intermediate exit feature [9]. Finally, the last algorithm combines both the dynamic VP capacity sharing concept and the intermediate exit feature into the design of the algorithm.

Before we describe our algorithm, let us define some additional terminologies. We denote a virtual path from node a to node b by $VP(a,b)$ and define $VP(a,b) = (\{a,b\},\{e(a,b)\})$ where $e(a,b)$ is the unidirectional edge from a to b in the directed graph G. Furthermore, we denote the addition of the virtual path $VP(a,b)$ to a multicast tree T by $T = T + VP(a,b)$ where the operation $T + VP(a,b)$ is defined as $(V_T \bigcup \{a,b\}, E_T \bigcup \{e(a,b)\})$.

Algorithm 1: The deterministic reservation multicast algorithm (MLLR_D):

The *MLLR_D* algorithm simply extends the LLR concept into the design of a multicast routing algorithm. When a multicast connection request arrives at the source node, following steps are performed.

1. The *MLLR_D* tries to connect the source node to each destination node via direct VP's first. Let T be the resulting multicast tree.

2. For those destination nodes which cannot be connected to the source node via direct VP's, *MLLR_D* tries to connect them to the destination nodes already on the multicast tree via a single VP. If there exists more than one such VP, then the one with maximum free capacity is chosen. Note that in order to satisfy the end-to-end QOS requirement, the two VP's on the path which connects the source node to the destination node must provide the class 1 QOS. For example, consider the multicast tree in Figure 2 in which the source node is connected to destination node d_2 first via VP_2, and then via VP_3 to d_3. Class 2 QOS is required on VP_2 when d_2 is connected to the multicast tree. But now, since the end-to-end QOS requirement of d_3, the QOS required on VP_2 must be changed to class 1 QOS when d_3 is connected to the multicast tree. (Of course, class 1 QOS is also required on VP_3.) We will refer to the path from the source node to d_3 as a direct-alternate path, since the path is an alternate path for d_3, but it also consists of VP_2 which originally is the direct path for d_2.

3. For those destination nodes that still cannot be connected to the multicast tree, *MLLR_D* tries to connect them to the source node via alternate paths. If there exists more than one alternate paths, the path with maximum free capacity is tried. Specifically, a destination node will be tried to connect the multicast tree via a single VP with maximum free capacity first. (At this step, it is still possible that a destination node can be connected to the multicast tree via a single VP because of the addition of alternate paths to the multicast tree.) If it is not possible to find such a VP, then alternate paths with length of two are tried. If the destination node is still cannot be connected to the multicast tree via alternate paths, the multicast connection request is blocked and all capacity reserved is released.

The detailed description for the *MLLR_D* algorithm is given in Figure 3.

To avoid network performance degradation under high traffic load, a trunk reservation concept is adopted in LLR-based routing algorithms [31]. Previous studies have shown that dynamic alternate routing methods based on LLR may lead to an unstable, and inefficient, network state in which most calls are carried on two link paths. Therefore, trunk reservation was introduced to guard against such performance degradation. The idea is to allow alternate calls to use free capacity of link (VP) ℓ only if free capacity of the link can accommodate more than $t(\ell)$ direct calls, where $t(\ell)$ is the trunk reservation threshold.

In the *MLLR_D* algorithm, trunk reservation is implemented in step 3 to prevent performance degradation. Furthermore, we found that multiple trunk reservation thresholds are needed on each VP; one threshold for each connection class. Finding optimal trunk reservation thresholds is very complicated because there are multiple classes of connections and classes of QOS on a VP. Therefore, in our numerical results, we have to tune the trunk reservation thresholds on each VP to yield the best performance for each multicast routing algorithms.

Algorithm 2: The deterministic reservation multicast algorithm with dynamic VP capacity sharing(MLLR_DS):

The *MLLR_DS* algorithm is developed based on the dynamic VP capacity sharing concept, proposed in [7]. The original idea is to allow a call blocked on the direct VP to be carried on another VP if these two VP's share the same source node and the set of links in the former VP is a subset of that of the latter. For example, consider VP 1 and VP 3 in Figure 4. Let the first node be the source node and the middle node be one of the destination nodes. If there is no free capacity to connect the destination node to the source node, instead of connecting the destination node via an alternate path which might consist of VP 3 and some other VP, the destination node could be still be connected via VP 1 if VP 1 could ''borrow'' some capacity from VP 3. The ''borrowed'' capacity would be ''returned'' as soon as one of existing connections on VP 1 terminates.

The *MLLR_DS* algorithm differs from *MLLR_D* by adding following step between step 1 and step 2 in the *MLLR_D* algorithm. This extra step is to allow dynamic VP capacity sharing between two VP's.

1(a): If a destination cannot be connected via the direct VP and there exists another VP with free capacity that can be shared by the direct VP, the destination is connected to the multicast tree via that VP.

The detailed description for the *MLLR_D* algorithm is given in Figure 3.

Algorithm 3: The deterministic reservation multicast algorithm with intermediate exits (MLLR_DE):

In [9], an *intermediate exit* feature of a VP was proposed for multicast routing in VP-based ATM networks. The idea is to have a VP/VC switch act as a termination point as well as a cross-connect point. For example, consider the simple network shown in Figure 4. Assume that a multicast request, which arrives to the first node as the source node, is to establish a multicast tree to the second node and the last node. With the intermediate exit feature, we can establish the connection on VP 3. When a cell, sent from the source node, arrives at the second node, the cell is duplicated; one of the cells is forwarded to the last node while the other cell is left at the second node.

The advantage of the intermediate exit feature is that less capacity is required to establish a multicast connection. For example, in Figure 4, we need at least two VP's, VP 1 and VP 3, to establish the multicast tree without the intermediate exit feature. In order to take the advantage of this feature, destinations that can be connected with direct VP's which can provide intermediate exit feature, such as VP 3 in Figure 4, are tried first. Specifically, when a multicast connection request arrives at the source node, following steps are performed.

1. The *MLLR_DE* tries to connect the source node to destination nodes via direct VP's with intermediate exit feature first.

2. For those destination nodes which are transit nodes of existing VP's on the multicast tree, remove them from the set of destinations to be connected and add them to the multicast tree (via notifying those intermediate switches to perform the copy-and-forward function). Note that no additional VP capacity is required for adding those destination nodes to the multicast tree.

3. For the remaining destination nodes, simply follow step 1 to 3 in the *MLLR_D* algorithm.

The detailed description for the *MLLR_D* algorithm is given in Figure 3.

Algorithm 4: The deterministic reservation multicast algorithm with dynamic VP capacity sharing and intermediate exits (MLLR_DSE):

The MLLR_DSE algorithm is a combination of MLLR_DS and MLLR_DE. In MLLR_DSE, i.e., both dynamic VP capacity sharing and intermediate exit features are considered. When a multicast connection request arrives at the source node, following steps are performed.

1. The MLLR_DSE tries to connect the source node to destination nodes via direct VP's with intermediate exit feature first.

2. For those destination nodes which are transit nodes of existing VP's on the multicast tree, remove them from the set of destinations to be connected and add them to the multicast tree.

3. The MLLR_DSE then tries to connect the source node to the remaining destination nodes via direct VP.

4. If a destination still cannot be connected via the direct VP and there exists another VP with free capacity that can be shared by the direct VP, the destination is connected to the multicast tree via that VP.

5. Step 2 and 3 in MLLR_D are then performed which tries to connect the remaining destination nodes via direct-alternate and alternate paths respectively.

4 Performance Evaluation

In this section, we study the performance of the four routing algorithms via a simple 5-node network, as shown in Figure 1(a), through simulation. We assume links a, b, e and f all have a bandwidth of 4.632 Mbps, each with an input buffer space of 148.224 Kbits (i.e., 32 msec worth of data). The remaining links are assumed to have bandwidth 3.088 Mbps, each with input buffer space 98.816 Kbits. The VP configurations are shown in Figure 1(b). Each VP is assigned a dedicated bandwidth of 1.544 Mbps and a buffer space of 49.408 Kbits. The ADAPTIVE scheme, proposed in [7] is used to establish two classes of QOS on a VP.

We assume that homogeneous voice sources, described by a two-state fluid flow model with parameters $(32000, 0.3513, 0.352)$, with end-to-end QOS $Q(0.032, 10^{-6})$ are offered to the network. All multicast connections are classified into 20 classes. The arrival process to each class is assumed to be Poisson process and the connection holding times are assumed to be exponentially distributed random variables with mean of 1. The reward parameter for the each class of traffic, $r_{(s,D)}$, is set in proportion to the number of its destinations, i.e., $r_{(s,D)} = |D| \times r$ where $|D|$ is the number of destination nodes and r is the unit reward for point-to-point connections. Two cases of traffic patterns are studied in comparing the performance of these four multicast routing algorithms. Let λ_i, $i = 1, 2, 3, 4$, be the total arrival rate for all classes of connections with i destinations. In the first traffic configuration, the ratio of the four arrival rates , $\lambda_1 : \lambda_2 : \lambda_3 : \lambda_4$, is set to 20:1:1:1 [3]. In the second case, the ratio is 1:1:1:1.

The performance of these four multicast routing algorithms under the two traffic configurations is compared in Figure 7 and Figure 8. (Recall that the performance metric of interest to us is the *fractional reward loss* defined in section 2.) In these figures, the fraction reward losses yielded by the four multicast routing algorithms are plotted under different traffic loads. The vertical lines about each point indicate the 95 percent confidence interval.

From Figure 7 and Figure 8, following observations can be made:

1. Routing algorithms with intermediate exit yield significantly lower fractional reward loss than algorithms that do not. These algorithms even perform much better than those algorithms that allow dynamic VP capacity sharing. For example, MLLR_DE performs much better than MLLR_DS. Thus, we can conclude that providing the feature of intermediate exit at ATM switches is very important for multicast routing.

2. Providing the dynamic VP capacity sharing feature to algorithms already with the intermediate exit feature does not improve the network performance very much. For example,

[3]Note that, in real networks, we expect point-to-point connections will still be the major traffic.

MLLR_DE and MLLR_DSE have almost the same performance under the first traffic configuration and yield the same performance under the second traffic configuration. The reason is that VP's which provide the feature of intermediate exit also provide capacity for loaning. Therefore, these two features conflict with each other. For example, consider VP3 in Figure 4 under the situation that a multicast request tries to connect the first node to the rest two nodes. In order to provide free ride for the intermediate node, VP3 had better not loan its capacity to VP1.

3. The relative performance of these four routing algorithms is almost the same under different traffic configurations.

In conclusion, multicast routing algorithms with dynamic VP capacity sharing concept yield better than algorithms that do not, but the improvement is not as significant as that of the algorithms with intermediate exit feature. In addition, providing the dynamic VP capacity sharing feature to algorithms already with the intermediate exit feature does not improve the network performance very much.

5 Summary

In this paper, adaptive multicast routing in VP-based ATM networks with reserved VP capacity was studied. Many characteristics of ATM networks, such as end-to-end QOS guarantee, multiple classes of traffic, and the use of virtual path, have made the routing problem much more complicated than that of traditional networks. We have, in this paper, designed four multicast routing algorithms, based on the Least-Loaded Path Routing concept, which take into account these characteristics. We also proposed the fractional reward loss concept for evaluating the multicast routing algorithms [16]. Our numerical results indicate that routing algorithms which utilize the intermediate exit feature [9] seems to yield the best performance.

References

[1] ITU-T SG. XVIII, ''I-Series Recommendations.''

[2] ITU-T SG. XVIII, ''Draft Recommendation I.311: B-ISDN General Network Aspects.'' Geneva, June 1990.

[3] ITU-T SG. XVIII, ''Draft Recommendation I.371: Traffic Control and Congestion Control in B-ISDN.'' Geneva, June 1992.

[4] R.-H. Hwang, ‘‘Routing in High-Speed Networks,’’ Ph.D. Dissertation, Dept. of Computer Science, University of Massachusetts at Amherst, 1993.

[5] R. Guerin, H. Ahmadi, and M. Naghshineh, ‘‘Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks,’’ *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 968--981, September 1991.

[6] S. Gupta, K. Ross, and M. E. Zarki, ‘‘Routing in Virtual Path Based ATM Networks,’’ *GLOBECOM’92*, pp. 571--575, December 1992.

[7] R.-H. Hwang, J. F. Kurose, and D. Towsley, ‘‘MDP Routing in ATM Networks Using the Virtual Path Concept,’’ *INFOCOM’94*, pp. 1509--1518, June 1994.

[8] R.-H. Hwang, ‘‘LLR Routing in VP-based ATM Networks,’’ *INFOCOM’95*, April 1995.

[9] M. H. Ammar, S. Y. Cheung, and C. M. Scoglio, ‘‘Routing Multipoint Connections Using Virtual Paths in an ATM Network,’’ in *INFOCOM’93*, (San Francisco), pp. 98--105, April 1993.

[10] J. L. Burgin, ‘‘Management of Capacity and Control in Broadband ISDN,’’ *Intl. J. Digital and Analog Cabled Systems*, vol. 2, pp. 155--165, 1989.

[11] J. Y. Hui, M. B. Gursoy, N. Moayeri, and R. D. Yates, ‘‘A Layered Broadband Switching Architecture with Physical or Virtual Path Configurations,’’ *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 1416--1426, December 1991.

[12] S. Ohta and K. Sato, ‘‘Dynamic Bandwidth Control of the Virtual Path in an Asynchronous Transfer Mode Network,’’ *IEEE Transactions on Communications*, vol. 40, pp. 1239--1247, July 1992.

[13] S. Ohta, K. Sato, and I. Tokizawa, ‘‘A Dynamically Controllable ATM Transport Network Based on the Virtual Path Concept,’’ *GLOBECOM’88*, pp. 39.2.1--5, 1988.

[14] D. Anick, D. Mitra, and M. M. Sondhi, ‘‘Stochastic Theory of a Data-Handling System with Multiple Sources,’’ *Bell Systems Technical Journal*, vol. 61, pp. 1871--1894, October 1982.

[15] R. Nagarajan, J. F. Kurose, and D. Towsley, ‘‘Local Allocation of End-to-End Quality-of-Service in High-Speed Networks,’’ *Modelling and Performance Evaluation of ATM Technology, IFIP Workshop TC6*, p. 2.2, January 1993.

[16] R.-H. Hwang, J. F. Kurose, and D. Towsley, ‘‘State Dependent Routing for Multirate Loss Networks,’’ *GLOBECOM’92*, pp. 565--570, December 1992.

[17] C. Grandjean, ‘‘Call Routing Strategies in Telecommunication Networks,’’ *Fifth International Teletraffic Congress*, 1967.

[18] W. H. Cameron, J. Regnier, P. Galloy, and A. M. Savoie, ‘‘Dynamic Routing for Intercity Telephone Networks,’’ in *Tenth International Teletraffic Congress*, June 1983.

[19] G. R. Ash, ‘‘Use of a Trunk Status Map for Real-Time DNHR,’’ in *Eleventh International Teletraffic Congress*, (Kyoto, Japan), September 1985.

[20] K. R. Krishnan, ‘‘Markov Decision Algorithms for Dynamic Routing,’’ *IEEE Communications Magazine*, vol. 28, pp. 66--69, October 1990.

[21] Z. Dziong *et al.*, ‘‘On Adaptive Call Routing Strategy for Circuit Switched Networks - Maximum Reward Approach,’’ in *Twelfth International Teletraffic Congress*, (Torino), June 1988.

[22] K. R. Krishnan and T. J. Ott, ‘‘State Dependent Routing of Telephone Traffic: Theory and Results,’’ in *Proceedings of 25th CDC*, (Athens, Greece), December 1986.

[23] Z. Dziong and L. Mason, ‘‘An Analysis of Near Optimal Call Admission and Routing Model for Multi-Service Loss Networks,’’ in *INFOCOM’92*, (Florence, Italy), May 1992.

[24] D. Mitra, R. J. Gibbens, and B. D. Huang, ‘‘Analysis and Optimal Design of Aggregated Least-Busy-Alternative Routing on Symmetric Loss Networks with Trunk Reservations,’’ *13th International Teletraffic Congress*, pp. 477--481, 1991.

[25] T. J. Ott and K. R. Krishnan, ‘‘State Dependent Routing of Telephone Traffic and the Use of Separable Routing Schemes,’’ in *Eleventh International Teletraffic Congress*, (Kyoto, Japan), September 1985.

[26] S. Hakimi, ‘‘Steiner’s Problem in Graphs and its Implications,’’ *Networks*, vol. 1, pp. 113--133, 1971.

[27] R. Karp, ‘‘Reducibility among combinatorial problems,’’ in *in Complexity of Computer Computations*, (New York: Plenum Press), pp. 85--103, 1972.

[28] V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, ‘‘Multicasting for Multimedia Applications,’’ in *INFOCOM’92*, (Florence, Italy), pp. 2078--2085, May 1992.

[29] C.-H. Chow, ‘‘On Multicast Path Finding Algorithms,’’ in *INFOCOM’91*, (Bal Harbour), pp. 1274--1283, April 1991.

[30] B. M. Waxman, ‘‘Routing of Multipoint Connections,’’ *IEEE Journal on Selected Areas in Communications*, vol. 6, pp. 1617--1622, Dec. 1988.

[31] J. M. Akinpelu, ‘‘The Overload Performance of Engineered Networks with Nonhierarchical and Hierarchical Routing,’’ in *Tenth International Teletraffic Congress*, June 1983.

[32] F. P. Kelly, ‘‘Effective Bandwidths at Multi-type queues,’’ *Queueing Systems*, vol. 9, pp. 5--15, 1991.

[33] A. I. Elwalid and D. Mitra, ‘‘Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks,’’ *IEEE/ACM Transactions on Networking*, vol. 1, pp. 329--343, June 1993.

Appendix

A Multiple Classes of QOS on a VP

Supporting multiple classes of QOS on a VP is required due to more stringent QOS requirement of alternate calls. In [7], a simple cell transfer control scheme, referred to as ADAPTIVE, is proposed for providing multiple classes of QOS on a physical link or VP. The ADAPTIVE scheme is a combination of two schemes: the UNIFORM scheme and the SEPARATE scheme.

In the UNIFORM scheme, all cells are provided with the most stringent QOS. Since all cells are treated equally, no special cell control scheme is required. However, link utilization will be decreased because an unnecessarily stringent QOS is provided to *all* cells, regardless of their actual QOS requirements.

On the other hand, the SEPARATE scheme separates cells according to their QOS requirements and allocates dedicated buffer space and bandwidth to each class of QOS. Cells with the same QOS requirement are treated equally and are served by their dedicated bandwidth based on a first-come-first-serve discipline. The advantage of this approach is that cells will receive their actual QOS requirements. However, this advantage is offset by the reduction in the statistical cell multiplexing gains because of the partitioning of the resources.

In order to implement the SEPARATE scheme, we need to determine the amount of buffer space and bandwidth to dedicate to each class of QOS. Let us consider an example where two classes of QOS, say $Q(D_1, \epsilon_1)$ and $Q(D_2, \epsilon_2)$, are to be established on a link of capacity C. Let N_1 and N_2 be the maximum number of virtual circuits of these two classes that can be simultaneously supported without violating the QOS requirements. Assume without loss of generality that $Q(D_1, \epsilon_1)$ is more stringent than $Q(D_2, \epsilon_2)$. Our approach to determine the amount of buffer space and bandwidth to dedicate to each class of QOS is as follows. For a given value of N_1, we first determine the buffer size, B_1, and the amount of bandwidth, C_1, required by the class 1 traffic by solving the following equations:

$$\begin{aligned} B_1 &= D_1 \cdot C_1 & (1) \\ C_1 &= \hat{c}_1 N_1 & (2) \\ B_1 &> 0 & (3) \\ C_1 &> 0 & (4) \end{aligned}$$

where $\hat{c}_1$ is referred to as equivalent capacity [5], or effective bandwidth [32, 33], and is given by

$$\hat{c} = \frac{\alpha b(1-\rho)r - x + \sqrt{[\alpha b(1-\rho)r - x]^2 + 4x\alpha b\rho(1-\rho)r}}{2\alpha b(1-\rho)} \tag{5}$$

where $\alpha = \ln(1/\epsilon_2)$ and $x = B_1$. After a simple calculation, we have

$$C_1 = \frac{(\alpha b(1-\rho) + \rho N_1 D_1) r N_1}{\alpha b(1-\rho) + D_1 N_1}. \tag{6}$$

Similarly, the maximum number of class 2 calls that can be supported, N_2, is given by equation (5) using $\alpha = \ln(1/\epsilon_2)$ and $x = D_2 \cdot (C - C_1)$.

The ADAPTIVE scheme is to take the advantage of both schemes. That is, the ADAPTIVE scheme will adopt the UNIFORM scheme if it supports more virtual circuits than the SEPARATE scheme. Otherwise, the ADAPTIVE scheme will adopt the SEPARATE scheme. In Figure 9, we compare and contrast these three schemes via a simple example in which two sets of homogeneous voice sources, characterized by the same two-state fluid-flow model with parameters (32000, 0.3513, 0.352), and differing only in their QOS requirements are presented to a link. From Figure 9, we can observe that the SEPARATE scheme is a complement to the UNIFORM scheme while the ADAPTIVE scheme takes advantages of both schemes.

```
Remark:  given a connection request (s,D)
T := {{s},∅}
/* Try to connect destinations via direct VP first*/
For all d ∈ D do
    If the direct VP between (s,d), VP(s,d), has enough capacity
    to carry one more direct call Then
        D = D − {d}
        T = T + VP(s,d)
    End If
End For
/* Try to connect remaining destinations via direct alternate path */
For all d ∈ D do
    For all x ∈ D ∪ V_T, find x* and VP(x*,d) such that
    (1) e(s,x*) ∈ E_T, and
    (2) Both VP(s,x*) and VP(x*,d) are able to provide class 1 QOS
        to satisfy the connection's QOS requirement, and
    (3) VP(x*,d) has the maximum free capacity among all qualified VP's.
    If VP(x*,d) has enough capacity to carry one more alternate call Then
        D = D − {d}
        T = T + VP(x*,d)
    End If
End For
/* Try to connect remaining destination via alternate path */
For all d ∈ D do
    For all x ∈ V_T − D, find x* and VP(x*,d) such that
    (1) e(s,x*) ∈ E_T, and
    (2) The sum of the QOS's on VP(s,x*) and VP(x*,d) satisfies the
        connection's requirement, and
    (3) VP(x*,d) has the maximum free capacity among all qualified VP's.
    If there exists such VP(x*,d) with enough capacity to carry one more
    alternate call Then
        D = D − {d}
        T = T + VP(x*,d)
    Else
        For all x ∈ V − V_T, find x*,VP(s,x*) and VP(x*,d) s.t.
         (1) Both VP(s,x*) and VP(x*,d) are able to provide class 1 QOS
             to satisfy the connection's QOS requirement, and
         (2) The path which consists of VP(s,x*) and VP(x*,d) has the
             maximum free capacity among all qualified paths.
```

Figure 3: The algorithm for MLLR_D *(to be continued)*.

```
        If both VP(s,x*) and VP(x*,d) have enough capacity to carry
            one more alternate call Then
            D = D − {d}
            T = T + VP(s,x*) + VP(x*,d)
        Else
            block the connection request,
            abort call setup procedure
        End If
    End If
End For
```

Figure 3: The MLLR_D algorithm *(continued)*.

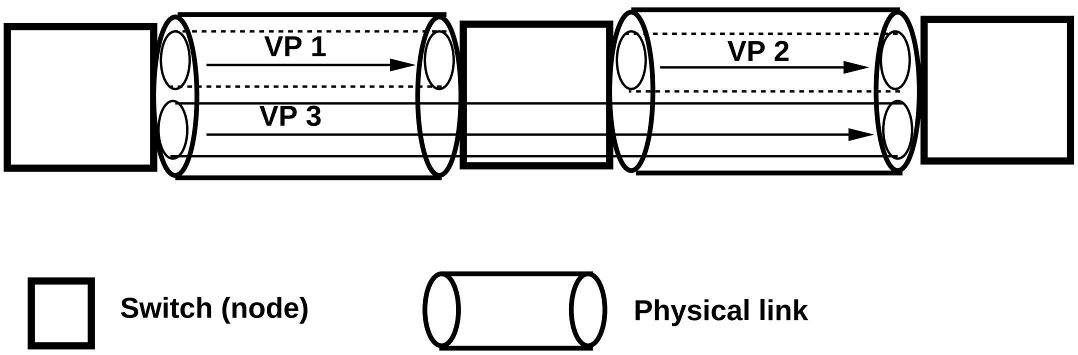


Figure 4: Example: A simple VP network

```
Remark:  given a connection request (s, D)
T := {{s}, ∅}
/* Try to connect destinations via direct VP first*/

        Same as in MLLR_D

/* Try to connect remaining destinations by sharing VP capacity */
For all d ∈ D do
    If there exists a VP(s, d') which has enough capacity
    to share with VP(s, d) Then
        D = D − {d}
        T = T + VP(s, d)
    End If
End For
/* Try to connect remaining destinations via direct alternate path */

        Same as in MLLR_D

/* Try to connect remaining destination via alternate path */

        Same as in MLLR_D
```

Figure 5: The algorithm for MLLR_DS.

```
Remark:  given a connection request (s, D)
T := {{s}, ∅}
/* Try to connect each destination via VP's with immediate exit feature */
For all d ∈ D do
    If VP(s,d), which provides immediate exit feature, has enough
    capacity to carry one more direct call Then
        D = D − {d}
        T = T + VP(s,d)
    End If
End For
/* Try to connect destinations via free ride on VP's */
/* with immediate exit feature */
For all d ∈ D do
    If d is the immediate node of some VP's already established Then
        D = D − {d}
        V_T = V_T + {d}
    End If
End For
/* Try to connect remaining destinations via direct VP */

        Same as in MLLR_D

/* Try to connect remaining destinations via direct alternate path */

        Same as in MLLR_D

/* Try to connect remaining destination via alternate path */

        Same as in MLLR_D
```

Figure 6: The algorithm for MLLR_DE.

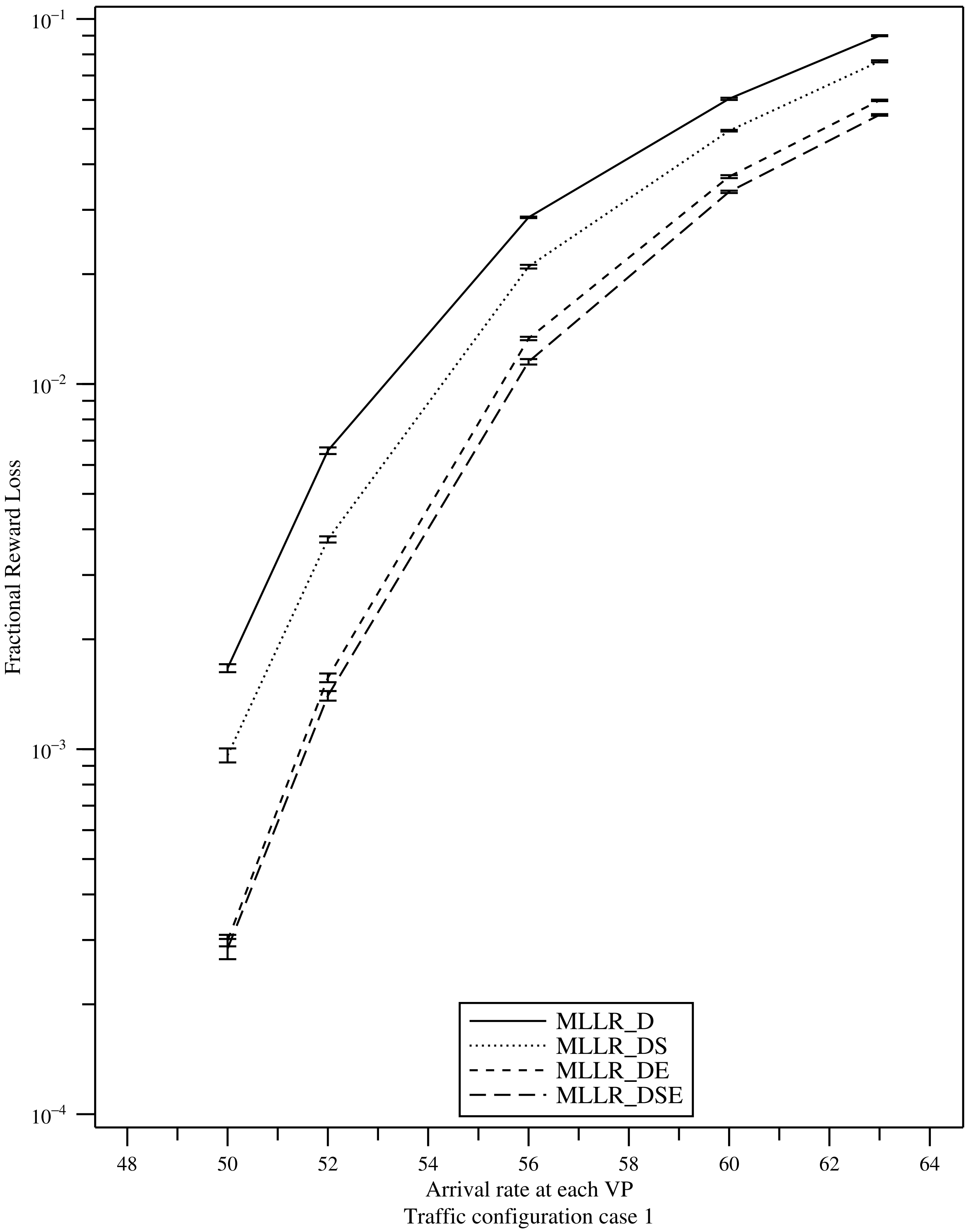


Figure 7: Performance of multicast routing algorithms under traffic configuration 1.

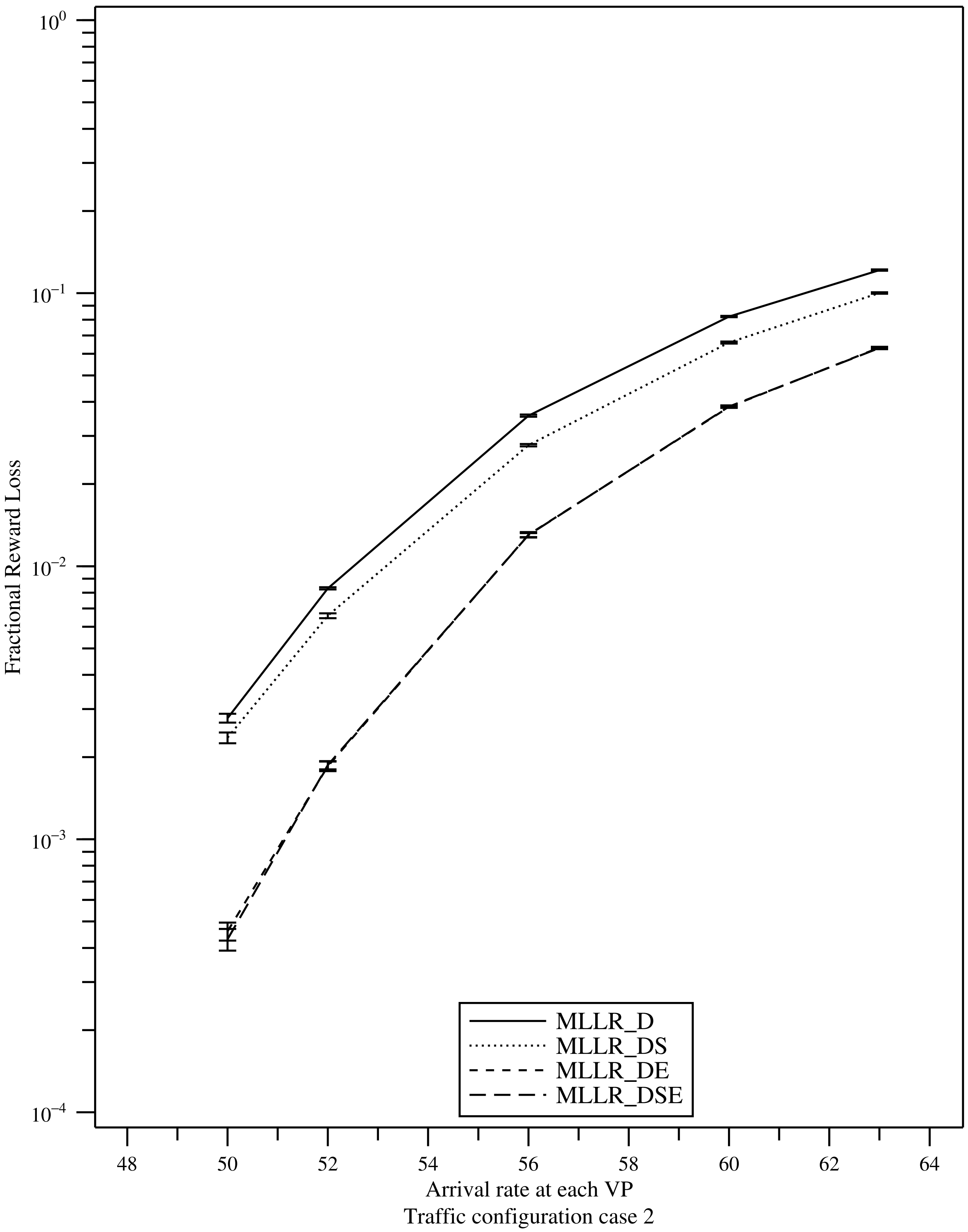
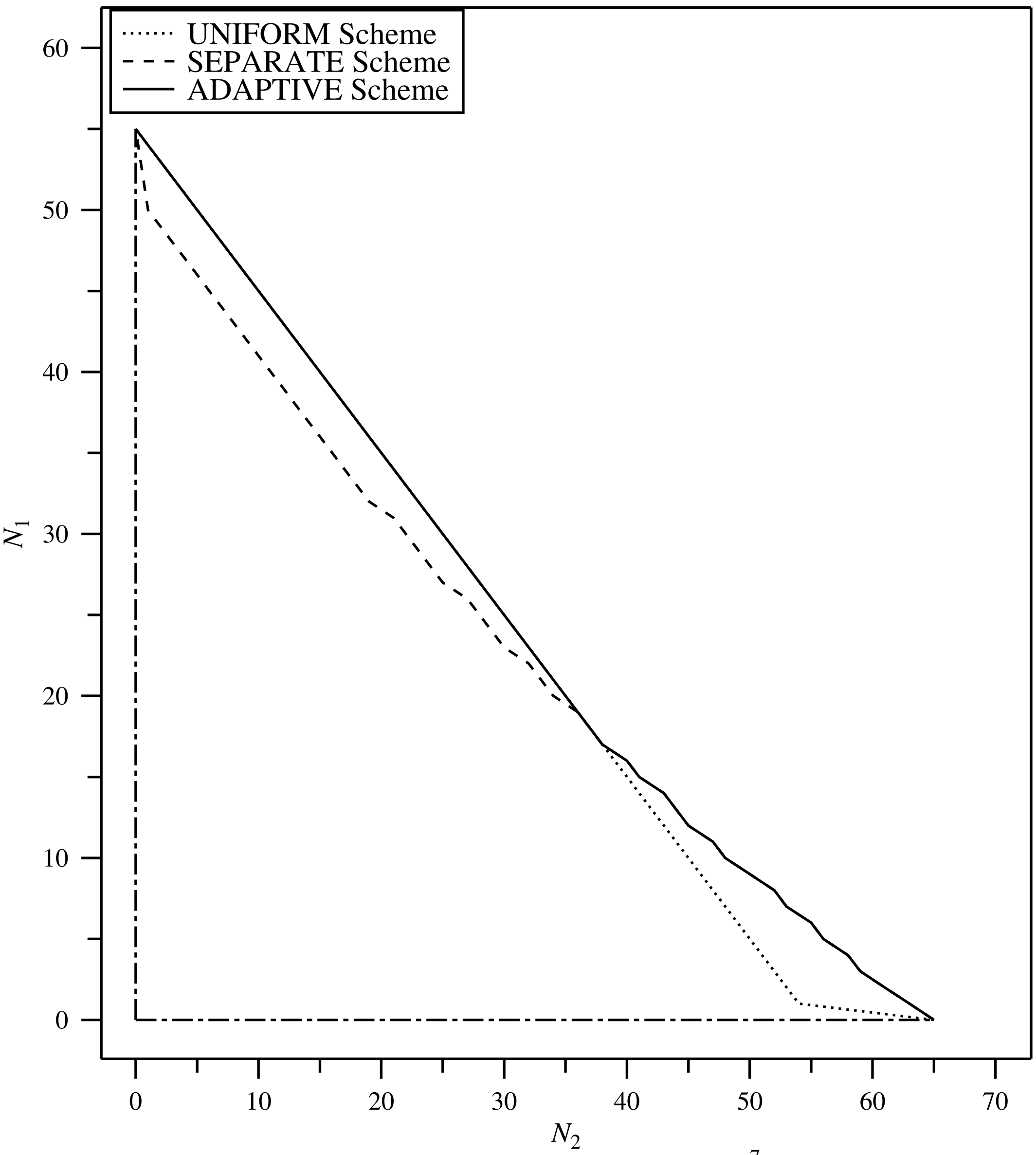


Figure 8: Performance of multicast routing algorithms under traffic configuration 2.



QOS of Class 1 = $Q(0.016, 5 \times 10^{-7})$
QOS of Class 2 = $Q(0.032, 10^{-6})$
Bandwidth: 1.544 Mbps

Figure 9: Comparison of the maximum number of VC's that can be supported under different cell control schemes.