Journal of High Speed Networks 00 (2004) 1–21
IOS Press

DiffServ-Aware Multicasting

Baijian Yang [a,*] and Prasant Mohapatra [b]
[a] *Department of Industry and Technology, Ball State University, Muncie, IN 47304 USA*
E-mail: byang@bsu.edu
[b] *Department of Computer Science University of California, Davis, CA, USA*
E-mail: prasant@cs.ucdavis.edu

Abstract. Advances in the areas of QoS and IP multicasting have necessitated the need of integration of these two important features of Internet. Differentiated services (DiffServ) has been proposed as a scalable Solution for supporting QoS in the Internet. Coexistence of multicasting and DiffServ is promising since the DiffServ model can provide a scalable framework and may reduce the computational complexity to locate a QoS-satisfied multicast tree. In this paper, we have first identify the problems of provisioning multicasting in DiffServ domains. Next, we have proposed an efficient DiffServ-Aware Multicasting (DAM) scheme which has three novel features: weighted traffic conditioning (WTC), receiver-initiated marking (RIM) scheme, and Heterogeneous DSCP Headers encapsulation (HDE). The proposed technique solves many problems with the integration of DiffServ and multicasting while accommodating heterogeneous QoS requirements. The framework is scalable, flexible, and feasible. Moreover, existing QoS techniques can be incorporated into this framework. Performance evaluation through analysis and simulations demonstrate conformance of the QoS requirements and the potential benefits of DAM.

Keywords: Differentiated service, DiffServ-Aware Multicasting, Quality of Service, receiver-initiated marking, weighted traffic conditioning

1. Introduction

The next generation Internet needs the support of two important aspects in addition to all the features of the current generation Internet. These aspects are: additional capacity and the support for Quality of Service (QoS). Capacity of the current generation Internet is likely to get outgrown by the bandwidth-consuming network traffic such as transmission of continuous media, interactive games, and the evolving peer-to-peer information sharing applications. The other issue is QoS support, which includes requirements of minimum bandwidth, delay, loss rate, jitter etc., which are being aggressively demanded by the evolving applications, and the transformation of Internet into a commercial infrastructure.

Simply increasing the network capacity through advanced technology is not the solution of the capacity problem. Historically, the users have always managed to consume the entire system capacity soon after it was enlarged [1]. IP Multicasting techniques [2–4] are attractive solutions for this capacity shortage problem since they can reduce the bandwidth consumption by sharing network resources. On the other hand, they bring extra overheads in routers for building and maintaining multicast delivery trees. IP multicasting techniques are most suitable for the bandwidth consuming applications which work in group communication models, such as video/audio conferences, distant education, and information dissemination. These applications also need QoS support, which provides assurance or guarantee of the required resources for meaningful comprehension of the information.

Several approaches have been proposed to provide QoS in the internet. Among them, Differentiated Service (DiffServ) [5] model has received more attention because of its scalability and implementation simplicity. In DiffServ model, traffic entering a network are classified and conditioned at the boundaries of the network and assigned to a set of behavior aggregates. Each traffic aggregate is assigned a DiffServ Code Point (DSCP). The DSCP is encoded in the packet header (such as in the TOS bytes in IPv4 header). Within the core of the network, packets

---

[*] Corresponding author.

are forwarded based on the per hop behaviors (PHBs) associated with the DSCP. A DiffServ domain is defined as a contiguous set of DiffServ-aware nodes with common service provisioning policy and a set of PHB groups implemented at each node. In order to support differentiated services, Service Level Agreements (SLAs) must be set up between the DiffServ domains, which specifies a negotiated service profile between two adjacent DiffServ domains. The resource allocation and management are handled by dedicated nodes, called Bandwidth Brokers (BBs), in each of the domains. In summary, DiffServ facilitates scalability by eliminating the hop-by-hop signaling and avoiding per micro-flow or per customer state maintenance within the core routers.

Current proposal of DiffServ has two basic classes of services: Assured Forwarding (AF) and Expedited Forwarding (EF). AF assigns out-of-profile (in excess of the SLA) traffic to a higher drop probability. It is used to support assured services in which the customers are likely to get the negotiated SLA without any guarantees. EF exercises strict admission control and drop all out-of-profile traffic. Since EF guarantees a minimum service rate and has the highest priority, it is used to support premium services.

IP Multicasting and QoS are closely related since most applications that are suited for multicasting normally desire QoS support. It is thus essential to design techniques to support QoS-aware multicasting in the Internet. The basic approaches for QoS-aware multicasting fall into three categories:

- *QoS-aware-route-based Multicast* [6–9]: The primary goal of the algorithms that belong to this class is to determine a routing path that can satisfy the QoS requirements for multicasting. The QoS-aware routing is done with minimum signaling overhead. Examples of this class include QoSMIC [7] and QMRP [8]. Although these schemes can find efficient QoS-aware routes for multicasting, they are not sufficient for provisioning QoS. The path search, if succeeds, can only guarantee that the resulting path meets the QoS requirements when there is no congestion. QoS services such as RSVP must be supported to maintain this QoS satisfied path. Further, lack of global QoS support makes QoS-aware-route-based schemes inefficient.
- *QoS-enabled Multicasting by Resource Reservation Protocol (RSVP)* [1]: To provide QoS, this approach builds a QoS-satisfied multicast tree using RSVP [10,11]. The RSVP extension to multicast is not more complicated than the case of unicast. However, the *per flow* soft state maintenance in every on-tree router make this scheme poor in terms of scalability.
- *Provisioning Multicasting in DiffServ Domain* [12]: The strength of this scheme is that it separates QoS issues from multicasting routing. That is, multicast delivery trees are built and maintained by normal multicast protocol without going through a complicated QoS path searching process. The QoS is provided by the underlying DiffServ architecture. Since DiffServ is a scalable approach, it is desirable to incorporate it with multicast to provide QoS enabled multicast.

Compared to the first two approaches, the third approach of integration of multicast with DiffServ is better in terms of scalability and ease of implementation. Moreover, DiffServ multicasting is not orthogonal to the other two approaches. DiffServ deals with packet forwarding, which is independent of the routing process. So QoS-aware-route-based protocols like QMRP and QoSMIC can be also plugged into the DiffServ-aware multicasting schemes.

Two major problems need to be addressed to support DiffServ multicasting. One of them is the Neglected Reservation Sub-tree Problem (NRS-Problem) [12], and the other is associated with marking/remarking schemes. We have proposed a DiffServ-aware multicasting technique that provides solutions for both the problems. The proposed DiffServ-Aware Multicasting (DAM) framework has two components. First, a Weighted Traffic Conditioning (WTC) table is located in every DiffServ edge router with a goal to maintain SLA integrity. By looking up the WTC table, multicast traffic is conditioned based on the amount of outgoing traffic rather than that of incoming traffic. The second component is the Receiver-Initiated Marking (RIM) scheme to support heterogeneous QoS requirements within a multicast group. Both the components are designed to respond to the group membership changes. In short, DAM can support QoS-enabled multicasting in DiffServ domain without violating the SLAs of a heterogeneous and dynamically changing group.

The rest of the paper is organized as following. In Section 2, DiffServ multicast network model is given and NRS-problem is discussed. DAM and its building blocks – weighted traffic conditioning model and receiver-initiated marking scheme are proposed in Section 3. The implementation details are summarized is illustrated in Section 4. Performance analysis and simulation results are presented in Section 5, followed by the concluding remarks in Section 6.

2. DiffServ multicasting model and problems

In this section, we describe the DiffServ multicasting model followed by the related issues and problems associated with SLA and packet marking.

2.1. DiffServ multicasting model

Differentiated services in the Internet is provisioned through independent DiffServ domains that contain nodes that are DiffServ-compliant. A DiffServ domain consists of Edge Routers (ERs) and Core Routers (CRs). The main idea behind the DiffServ architecture is to push the complexity of traffic conditioning and policing to the ERs. The core routers just take care of forwarding packets based on their per-hop behavior (PHB), thus retaining the simplicity of implementation. Details of the DiffServ architecture is described in [18]. Figure 1 shows two DiffServ domains A and B, serving a multicast message originating from node S and destined to nodes R1, R2 and R3. The assumptions for the network are as follows:

- Each edge router supports multicast routing protocol.
- Each core router is directly connected with edge routers or other core routers. The core routers can be multicast capable or incapable. In this paper, we only focus on the case where core routers are capable of multicasting. However, the framework proposed in this paper can be easily extended to the situations where all core routers are not multicast capable.
- Each domain is both multicast and DiffServ capable. That is, we do not consider the case in which intermediate domains are incapable of either multicast or DiffServ. Later we indicate how our scheme can be adapted for domains that are not DiffServ capable.
- For the core-based multicast routing protocols, the root of the tree (core in CBT, RP in PIM [2,3]) is either located at the edge of the DiffServ domain or at the edge routers.

Given the network model as described above, for each edge router, the total amount of traffic that flows into domain i is denoted as FI_i, and total amount of traffic leaving domain i is denoted as FO_i. The capacity of upstream links and downstream links of router i are denoted as CI_i and CO_i respectively. Using the flow conservation property, the unicast flow equation of this model is:

$$\sum FI_i = \sum FO_i, \quad \text{where } FI_i \leqslant CO_i, FI_i \leqslant CI_i. \tag{1}$$

In other words, for unicast traffic, if capacity constraints are met and if no packets are dropped, the amount of input traffic to each DiffServ domain equals to the amount of output traffic of that domain. However, equation (1) does not hold for multicast cases since additional traffic may be generated within a DiffServ domain, which violates the SLA discussed in the following section.

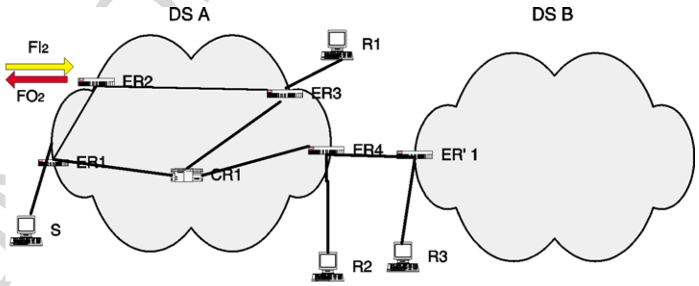

Fig. 1. DiffServ multicasting model.

2.2. NRS problem and SLAs violations

In the context of DiffServ, network resources are consumed based on the pre-negotiated SLA. However, in a DiffServ-aware multicasting environment, it is possible that the actual resources consumed exceed the pre-negotiated SLA. Since the multicast tree could branch at any node, the amount of outgoing traffic from a domain may exceed the incoming traffic to the domain and thus consume additional resources. This problem is denoted as Neglected Reserved Sub-tree (NRS) problem [12], which violates the SLAs and adversely affects any existing traffic flows.

There are two basic types of NRS problems.

Case 1: Figure 2(a) shows a multicast tree originating from source S and destined to R1 and R2. The branching point is at the egress node ER3 of DiffServ domain A. Assume that the existing multicast flow has subscribed the service level L. The border routers are equipped with meters and they normally do traffic conditioning to ensure that the traffic going to the downstream DiffServ domain B conform to the SLA between two domains. Consider R2 joining the multicast group at ER'1. The extra traffic generated between ER3 and ER'1 by this new multicast traffic for R2 may exceed the service L that domain B has subscribed from domain A. Thus if the SLA is not renegotiated, the over-subscribed packets with service L will be dropped randomly without discriminating between the flows. The consequence is that both unicast and multicast traffic with service L are adversely affected.

Case 2: Consider Fig. 2(b) where R2 joins the multicast group at CR1. The branching point is the interior core node CR1 in DiffServ domain A. Since traffic meters are normally not available in core routers of a DiffServ domain, the extra traffic in DiffServ domain A will consume more than that was subscribed. In other words, the extra multicast traffic may "steal" the traffic quota from lower service levels on the output link.

NRS problem can be solved by assigning a Lower than Best Effort (LBE) PHB to the newly branched multicast traffic [13]. In this approach, the resources and processing of existing traffic are protected while maintaining the

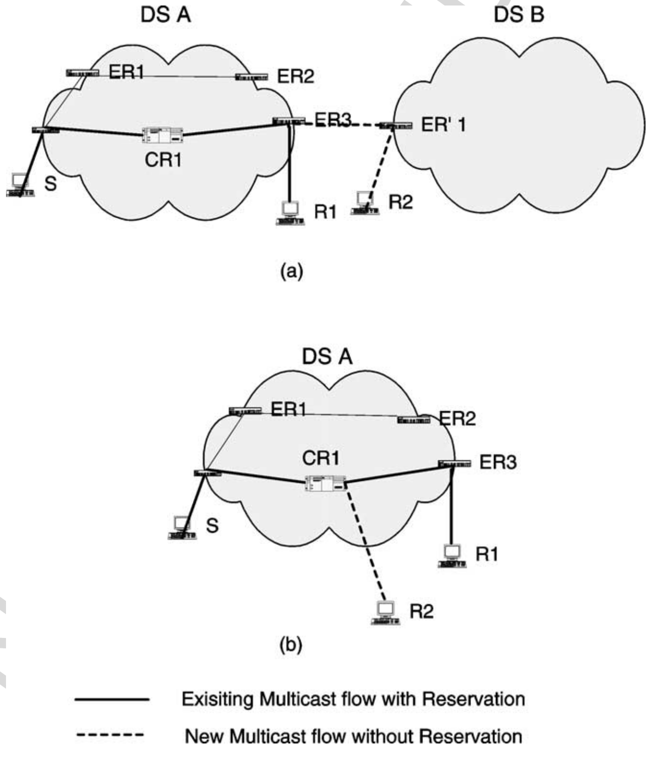

Fig. 2. Illustration of the NRS-problem.

simplicity of the DiffServ model. In order to get higher level of services, the joining node has to explicitly negotiate with the bandwidth brokers (BBs). Upon succeeding, the BBs will reconfigure the routers accordingly. However, this approach did not address how networking resource management should be done in DiffServ multicasting environment.

Alternatively, NRS problem can be eliminated by installing RSVP along the path, which requires per-flow soft state maintenance in the core and thus leads to poor scalability. Although recently proposed RSVP Aggregation scheme [14] can dramatically reduce the amount of states stored in the RSVP capable routers, it may not have significantly performance improvements over multicast sessions due to the problems of complicated aggregation of path message distribution and heterogeneous resource requirements.

2.3. *DiffServ multicasting marking problems*

In the DiffServ network, packets are marked as "In" or "Out" on the basis of the profile negotiated through the SLA. In unicast communication, the marking of the packets is usually done on the aggregate basis of bandwidth requirements. As DiffServ is uni-directional, the current marking scheme is normally sender-based, which do not consider the QoS requirements of the receivers. However, such a marking would not be adequate in multicast communications. Packet marking in DiffServ multicasting environment differs from that of the unicast case in the following three aspects:

1. IP multicast works in a group communication mode and receivers in a multicast group may have different QoS requirements. When multicast packets are duplicated in a router, every outgoing branch may have to be marked differently. It also implies that the marking should be based on the requirements and the capabilities of the receivers.
2. Group membership in multicast operation is dynamic. When a new host joins a multicast group, a new branch may be generated. Simply coping the DSCP code from existing branch may lead to SLA violations.
3. When heterogeneous marking are allowed in a DiffServ domain, marking of the lower level of services to the subtrees that branch after admission control will bring unfairness between multicast flows and unicast flows. Consider the scenario in Fig. 3, a multicast flow enters a DiffServ domain at ingress router E1, and is destined to E3 and E2 requesting *EF* and *AF1*, respectively. At the incoming interface of E1, assume that 80% of the *AF1* packets are in-profile and marked with lower drop probability of DSCP *AF11*, while the remaining *AF1* packets are marked with higher drop probability of DSCP *AF12*. Further assume that network congestion only happens on the link from core router CR to edge router E2. Marking the multicast packets as *AF11* DSCP at the multicast branching node CR is not fair because this new subtree originating from the core router is unaffected by the traffic conditioning that the unicast message encounters at E1. In this example, unicast *AF1* class packets traveling from E1 to E2 will be dropped more severely than that of multicast packets, since 20% of the unicast *AF1* packets are marked with a higher drop probability *AF12*, but all the *AF1* multicast packets are granted *AF11* DSCP. Thus it is unfair to the existing unicast flows if markings at the branching points are not handled properly.

In this paper, we have proposed a fair marking scheme, which accommodates heterogeneous QoS requirements of the receivers without violating the SLAs or affecting the quality of the existing flows.

3. DiffServ-Aware Multicasting (DAM)

We propose a DiffServ-Aware Multicasting (DAM) technique, which is composed of three novel components: Weighted Traffic Conditioning (WTC) model, Receiver-Initiated Marking (RIM) scheme, and Heterogeneous DSCP Headers Encapsulation (HDHE). In this section, we outline the components and the algorithm for DAM in detail. The WTC model aims to maintain the negotiated SLAs in DiffServ multicasting environment, and the RIM scheme is proposed primarily to accommodate heterogeneous QoS requirements of the receivers in multicast groups, while HDHE provides a mean to ensure the fairness among multicast flows and non-multicast flows.

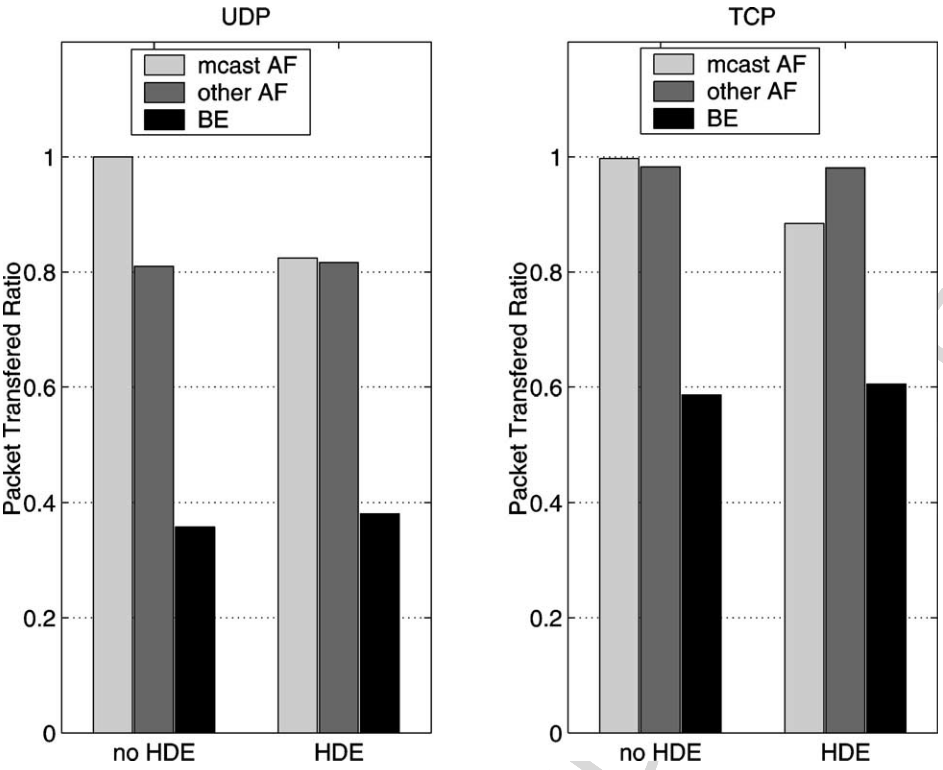

Fig. 3. Illustration of the unfair marking problem.

3.1. Weighted traffic conditioning (WTC) model

To motivate the proposed model, we first itemize the causes of the NRS problem in DiffServ environments.

1. Equation (1) is not satisfied in DiffServ multicasting environment.
2. BBs are unaware of the traffic replications due to multicasting.
3. By default, the DSCP as well as data will be copied at the branching point of the multicast delivery tree.
4. Traffic conditioning and policing in DiffServ are performed at the ingress of the domain on an aggregated basis.

The fundamental idea of WTC is to ***count*** the admitted multicast traffic as multiple unicast traffic while conditioning the traffic aggregate at the edge routers. This approach (discussed later in detail) is different from some of the unicast based multicasting schemes, such as [15], where one multicast flow is replaced by multiple unicast flows and each intermediate multicast receiver acts as a proxy server by sending multiple unicast flows to its downstream receivers. In our approach, however, it is not necessary to convert a multicast flow into multiple unicast flows (although it can also be applied to the unicast based multicast techniques). The counting is done only on a logical basis. Thus, the WTC model retains the bandwidth saving feature of multicasting. Further, WTC scheme can work in a Single Path (SP) mode and Multiple Paths (MP) mode of multicast. In the WTC scheme, the amount of traffic that leaves a DiffServ domain is counted at the entry of that domain. So WTC is independent of the way in which the traffic leaves a domain, facilitating it to work in both SP and MP multicasting.

The major goal of WTC is to keep the integrity of equation (1) while conforming to the fundamental idea of DiffServ and IP multicasting. Another reason for introducing WTC to the DiffServ-aware multicasting is because of the pricing schemes. Currently proposed pricing schemes in DiffServ networks [16] have not taken multicast

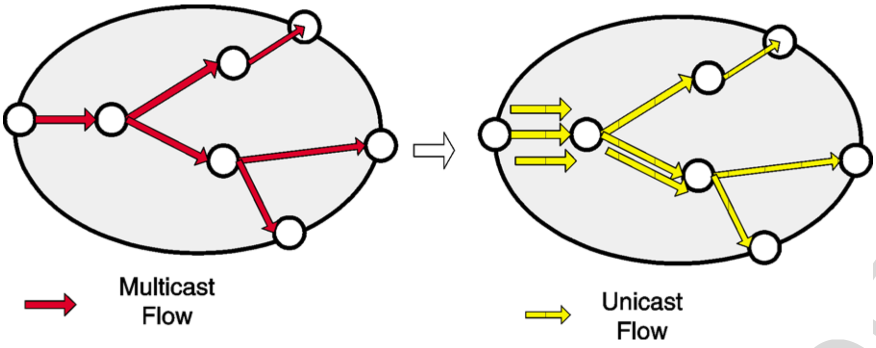

Fig. 4. Counting a multicast flow as multiple unicast flows.

duplication into consideration. Without an appropriate traffic metering scheme, they cannot be expanded in the context of DiffServ multicasting. The proposed WTC approach inherently facilitate the pricing structure.

The WTC scheme can be illustrated with the example shown in Fig. 4. If one multicast flow with premium service enters domain A and is replicated twice within this domain, the amount of that flow should be counted three times as many as the original amount at the boundary of the domain. Using this approach, there will not be any SLA violations since the amount of traffic counted at the ingress point of a DiffServ domain equals to the actual amount of traffic flowing out of that domain. Another issue is that the number of branching in multicast trees is a dynamic process due to member join and leave process. So we have adopted an approach (described in Section 3.4) to handle this dynamics. It must also be noted here that the proposed counting approach overestimates the bandwidth requirements within each domain for each of the multicast flows. However, such overestimation helps in maintaining the SLAs.

3.2. Receiver-initiated marking (RIM) scheme

Inherently QoS-aware multicasting is a receiver-based approach. Receivers join and leave at their own will and many have heterogeneous QoS needs. In the context of DiffServ-aware multicasting, packets should be marked according to receivers' QoS requirements. This concept is quite different from the popular DiffServ model which is unidirectional (usually sender-based) in nature. The QoS specification is made only in one direction; from the sender to the receiver. When a new receiver joins a multicast group, its QoS requirement could belong to one of the following four levels:

1. It has no QoS requirements.
2. It requests for whatever is the highest level of the available QoS at the node at which the new member joins.
3. It explicitly specifies a QoS requirement that is lower than or equal to the highest available QoS at the node at which it joins.
4. It explicitly specifies a QoS requirement that is higher than the available QoS at the node at which it joins.

Among these four types, level 1 and level 2 define relative QoS requirements, i.e., a new receiver only needs to indicate whether it has QoS requirements or not when it seeks to join a multicast group. If a receiver specifies QoS requirements explicitly, it indicates that the receiver wants absolute QoS requirements, which can be further classified as level 3 and level 4. Different levels of QoS requirements demand the packet marking scheme to appropriately handle them. To meet end-to-end QoS requirements, packet marking should be done in a consistent manner. In other words, a multicast sub-tree should be grafted at a node where its upper stream is marked at a level equal to or higher than the markings of the sub-tree.

The basic rules of this RIM scheme are described as follows, where the DSCP "DEFAULT" can be either BE or LBE.

- *Level 1*: Mark the new branch as DEFAULT.
- *Level 2*: If the highest available QoS is DEFAULT, do the same as in the case of level 1. Otherwise signal network management entities (e.g., BB or the ingress router) for admission control. If successful, copy the "highest" available DSCP at the joining node, otherwise, mark it as DEFAULT.
- *Level 3*: Signal network management entities for admission control. If successful, update the WTC look-up table, and mark the new branch with a DSCP that corresponds to the best available QoS, otherwise, mark it as DEFAULT.
- *Level 4*: Traverse retracing path toward the a root of the multicast tree until an on-tree node having a DSCP equal to or higher than the requested QoS requirement is found. Signal network management entities for admission control. If successful, mark the new branch with a DSCP that corresponds to the best available QoS and remark intermediate path with this new DSCP. If unsuccessful, either try selecting a new path or simply mark it as DEFAULT.

The admission control mentioned above includes authentication, authorization and allocation. Since there are a number schemes proposed for SLAs renegotiation and the BB implementation [17,20,21], the actual resource allocation schemes could be different. The proposed RIM and WTC techniques are independent of these implementation variations.

3.3. *Heterogeneous DSCP encapsulation (HDE)*

In HDE, when a multicast flow enters a DiffServ domain and is supposed to be branched with heterogeneous QoS requirements at a core router, the markings for each of these branches are encapsulated in the packet header at the ingress router of the domain. Thus the traffic conditioning done at the edge routers will be equally applicable to all the multicast branches that will egress out of the DiffServ domains with different QoS requirements. Thus all of these branches encounter the same traffic conditioning as that of any existing unicast message. As the number of branchings within any DiffServ domain is not expected to be too high (or a limit could be imposed), the HDE

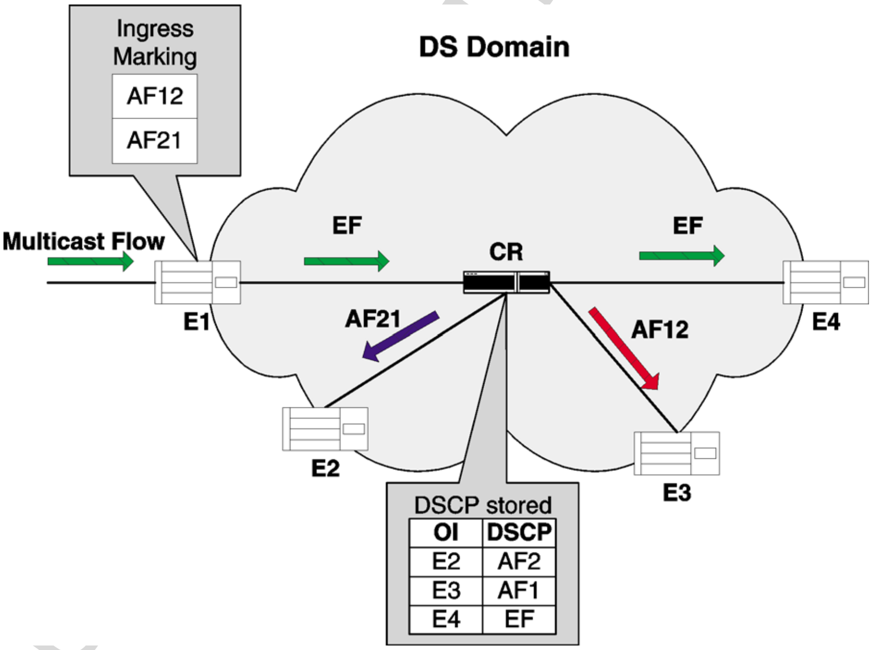

Fig. 5. Illustration of heterogeneous DSCP encapsulation (OI: Outgoing Interface).

scheme will not pose significant overheads in terms of the header length. Furthermore, we need to capture the markings of only the heterogeneous Af traffic. The out-of-profile EF traffic will get dropped at the ingress router.

Consider Fig. 5 as an example. A multicast flow enters a DiffServ domain at edge router E1. One branch that leaves through E4 is marked as *EF*, another branch that flows out from E3 is marked as *AF1*, and the last branch exiting from E2 is marked as *AF2*. Suppose at E1, this flow is marked *AF12* for *AF1* class, and *AF21* for *AF2* class, this information is inserted in the packet. When this packet is duplicated at the branching node CR, the DSCP stored in CR indicates the class of service, and the actual DSCP code should be either copied or calculated from the the information stored in the header. In our example, the branch from CR to E3 belongs to class *AF1*, and ingress *AF1* marking for this packet is *AF12*, thus this branch will be marked as *AF12*. For the same reason, the branch from CR to E2 will be marked as *AF21*.

3.4. *DiffServ Aware Multicasting (DAM) technique*

The primary goal of this technique is to quantify the amount of each multicast traffic flow at its ingress node and have the packets appropriately marked. It is clear that the WTC scheme demands edge routers to maintain and update flow-specific information. The proposed protocol, by taking receivers' QoS requirements into account, can reduce the load of updating WTC look-up table.

In DAM, receivers' QoS requirements will be piggybacked on the multicast JOIN packet. If a receiver requests no QoS requirements or the network fails to allocate the requested resources, the new branch will be marked as DEFAULT. In such cases, there is no need to update the WTC look-up table. The weighted multicast flow traffic conditioning is needed only when the new branch needs to be marked higher than the DEFAULT. So for the rest of this section, we only consider QoS requirements at levels 2, 3 and 4 (as defined earlier in Section 3.2).

When a receiver wants to join a multicast group, the existing multicast delivery tree may or may not exist in its DiffServ domain. There may be three possible cases that need to be considered, as shown in the Fig. 6. The "join router" in the discussion refers to the nearest on-tree node whose highest DSCP is either higher than or equal to the receivers' QoS requirements.

- **Case 1**: Join at the egress node of the DiffServ domain. In this scenario, the first hop of the receiver, which is also the egress point of the DiffServ domain, is already in the multicast delivery tree. Without further ado, the new receiver can join directly. The extra traffic generated on the output link by the multicast replication will not affect other hosts, subnets or domains. Therefore, it is the responsibility of Subnet Bandwidth Manager (SBM) or hosts to ensure that their QoS requirements do not exceed what they subscribed to in conformance with the SLA. Therefore, the Admission Controller (AC) will not be signaled and the multicast flow weight will not be updated.
- **Case 2**: Join at the ingress or interior node of the same DiffServ domain. When the edge routers get a "JOIN" message and find out that it is not in the multicast delivery tree, it will then forward this JOIN request back toward a root of the multicast tree. When it finally reaches a join point, which could be either an ingress node or an interior node of the same DiffServ domain, multicast flow weight may need to be updated at the WTC look-up table. The procedures that should be followed can be enumerated as:
    1. Mark the new branch as DEFAULT. If the branching router is a core router, it sends a REQUEST message to the upstream ER in its domain.
    2. ER sends an Admission Control Request (ACR) to the BB similar to the case when a unicast flow wants to send packets to this DiffServ domain.
    3. Upon receiving ACR, the AC validate the request based on the SLA and resource availability. AC will send Admission Control Answer (ACA) message to the requesting ER. The ACR message will be positive if it is successful, otherwise it will be negative.
    4. If the response from the AC is positive, the ER marks this new branch with the DSCP that corresponds to the best available QoS and sends an UPDATE message to the downstream routers in the path from this edge router down to the new receiver. All the inter-domain ingress routers on this path should also update their WTC look-up table.

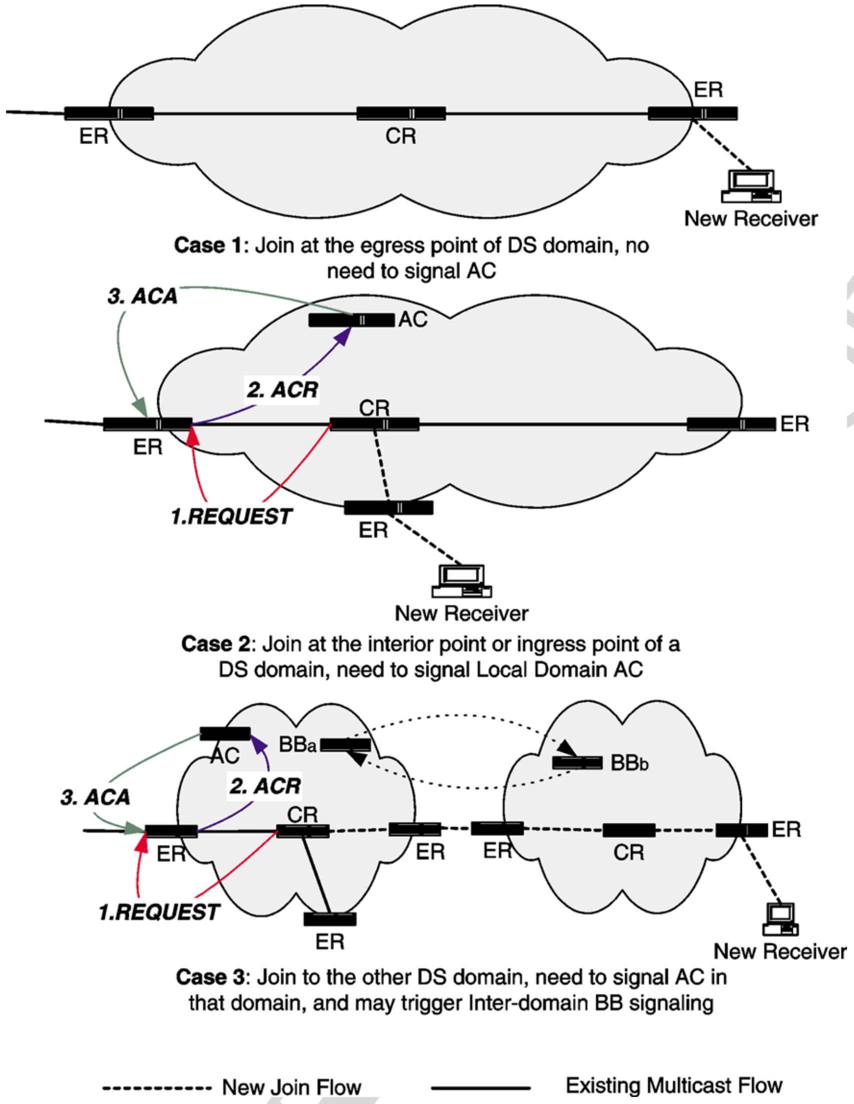

Fig. 6. Three cases of joining multicast tree in a DiffServ domain.

- **Case 3**: Join at another DiffServ domain. In this case, the JOIN message will be forwarded to other DiffServ domains. Basically routers in the joining DiffServ domain will perform the same actions as described in case 2.

For the cases of "LEAVE" or "PRUNE", the same approach is followed with minor differences, such as, decreasing the counter or removing the entry rather than increasing or generating the corresponding elements.

The DAM algorithms of multicast "JOIN" with QoS requirement are formalized in Algorithms 1 and 2. The DSCP field generation for multiple outgoing interfaces with heterogeneous QoS requirements is described in Algorithm 3.

As described in Algorithm 1, core routers only perform simple tasks like setting up new routing entry and passing messages. This design conforms to the basic concept of DiffServ architecture since it keeps core routers simple and fast. The complexity of DAM is pushed to the ERs as illustrated in Algorithm 2. The major tasks of ERs are updating their WTC look-up tables and signaling BBs in their domain. Algorithm 3 outlined a DSCP

Algorithm 1 **DAM at Core Router i**

```
switch (the received message)

case JOIN(r_ip,qos,G)
 if (i is on-tree node of G)
  dscp := DEFAULT
  create a routing entry R(in,out,G,dscp)
  forward REQUEST(r_ip,qos,G) to upstream
 else
   send JOIN(r_ip,qos,G)

case REQUEST(r_ip,qos,G)
  reverse forward REQUEST(r_ip,qos,G)

case UPDATE(r_ip,newdscp,G)
  R.dscp := newdscp
  unicast UPDATE(r_ip,newdscp,G) to downstream node
```

Algorithm 2 **DAM at Edge Router j**

```
switch (the received message)

case JOIN(r_ip,qos,G)
 if (j is on-tree node of G)
  dscp: = DEFAULT
  create a routing entry R(in,out,G,dscp)
   if (qos > the highest dscp of G at j)
    forward REQUEST(r_ip,qos,G) to upstream
   else
    if (r_ip and R.out in the same subnet)
     R.dscp := mapping (qos)
    else
     send ACR(r_ip,j,qos,G) to local BB
 else
  send JOIN(r_ip,qos,G)

case REQUEST(r_ip,qos,G)
 if (qos > the highest dscp of G at j)
  forward REQUEST(r_ip,qos,G) to upstream
 else
  send RAR (r_ip,j,qos,G) to the local BB

case UPDATE(r_ip,newdscp,G)
 if (R.in from other DiffServ Domain)
   update WTC table
 newdscp = mapping (R.in,newdscp)
 R.dscp := newdscp
 if r_ip and j not in same subnet
   unicast UPDATE (r_ip,newdscp,G) to downstream node

case ACA(r_ip,newdscp,G)
 if (ACA == positive)
  if (R.in from other DiffServ Domain)
    update WTC table
  R.dscp := newdscp
  unicast UPDATE (r_ip,newdscp,G) to downstream node
 else
  discard received message
```

Algorithm 3 `DSCP Generation in DAM forwarding`

```
for (every multicast oif) {
 dscp := oif.dscp
 for (every dscp in the packet's hdhe) {
  if (packet.dscp.class == dscp.class)
   dscp := packet.dscp
 }
}
```

generation procedure when duplicate multicast packets at the branching nodes. All the algorithms are independent of the multicast routing protocol, which make DAM a flexible approach for implementation.

4. Implementation issues

In this section, we describe the implementation details of the proposed DAM technique. The configuration of traffic conditioners and the marking process are also discussed.

4.1. WTC

In order to perform weighted traffic conditioning, ERs should maintain a look-up table and they should be informed of the number of replications of each multicast flow. The look-up table would contain the following fields: multicast group ID, DSCP, and the number of replications.

The architecture of traffic conditioners at the edge routers should be thus changed to facilitate weighted multicast metering as illustrated in Fig. 7. When packets enter the edge router, based on their destination address, they will enter at either the unicast classifier component or the multicast classifier component. In the unicast case, the traffic conditioning structure remains unchanged. If a multicast flow f enters a domain, since its destination IP address is a class D address, it goes to the multicast classifier unit and then checks the look-up table for the weight. As shown in the example of Fig. 8, the look-up results indicate that this flow has two DSCP codes: *D1* and *D2*, with weight, *w1* and *w2,* respectively. It means that the flow has $w1 + 1$ branches marked as *D1* and $w2 + 1$ branches marked as *D2* leaving this domain. Thus packets of flow f should be shaped and conditioned based on the look-up weight results. The other entries in the Table corresponds to AF and EF packets as indicated.

The overhead of maintaining WTC is analyzed in Section 5, which can be reduced by using the receiver-initiated marking scheme discussed in the next section.

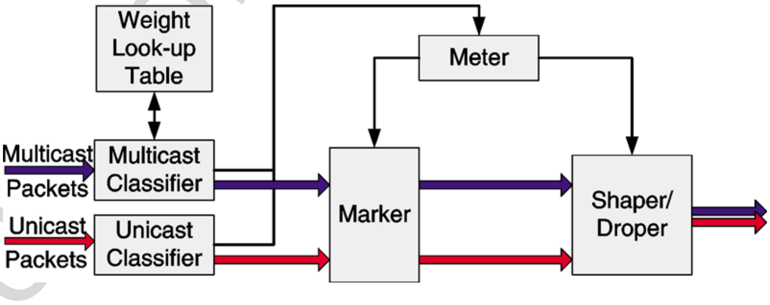

Fig. 7. Logical view of traffic conditioner with WTC component.

4.2. Marking

Figure 9 described a token bucket implementation scheme of DAM marking at the edge routers. For simplicity, we assume that the DiffServ domain supports two classes of forwarding schemes, EF and AF, respectively. Further, we assume that each packet consumes only one token.

To facilitate the receiver initiated marking scheme, every multicast-capable router needs to have one more DSCP field setup for the multicast flow. This extra field was also suggested in [12].

Destination Address	**DSCP**	**Weight**
226.35.7.28	EF	1
226.35.7.28	AF1	3
226.35.7.28	AF2	2
IP of flow f	D1	w1
IP of flow f	D2	w2
...	...	...

Fig. 8. An example of multicast flow weight look-up table.

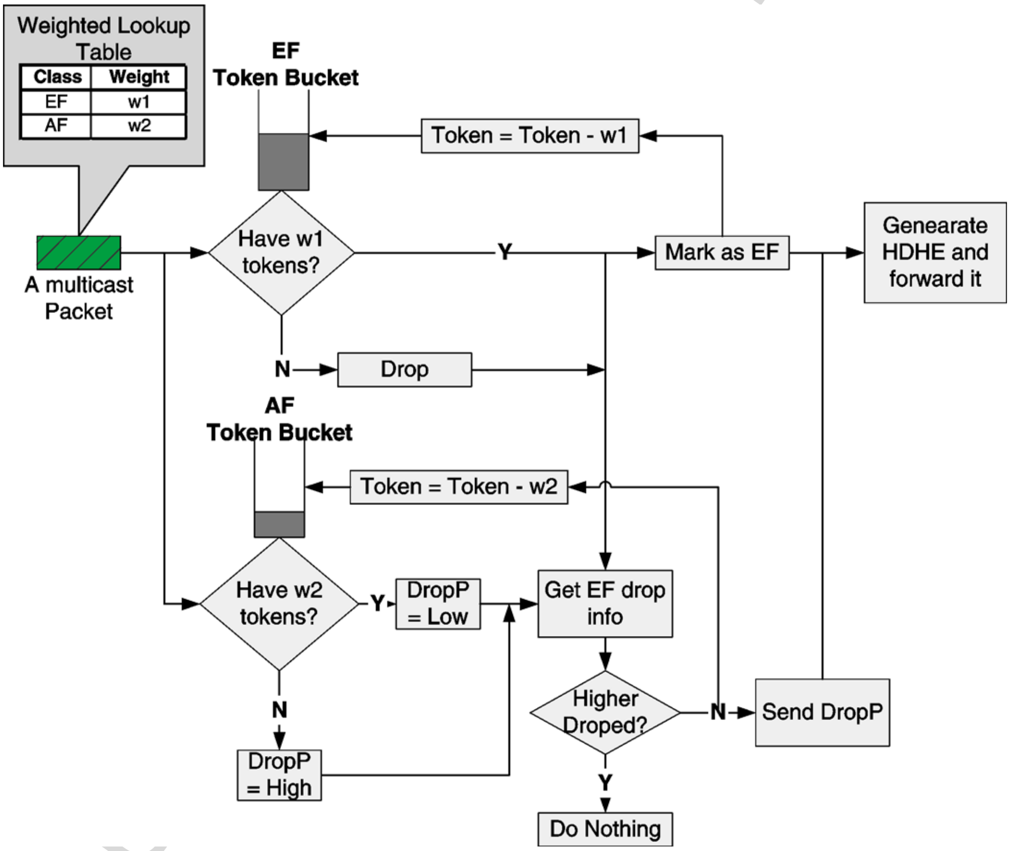

Fig. 9. Token bucket marking implementation at the edge routers.

5. Performance analysis

We provide both qualitative and quantitative performance evaluation of the proposed multicasting approach through analyses and simulations.

5.1. Qualitative analysis

In this section, the proposed DAM technique will be evaluated qualitatively using the following performance matrices: scalability, flexibility, feasibility, and complexity.

- **Scalability**. DAM is scalable because it is built on top of the DiffServ model by pushing the complexity to the edges of the networks. Compared to the traditional DiffServ approach, DAM adds only one extra overhead, which is the multicast flow weight look-up tables maintained at the edge routers. However, this overhead is not significant and it will not adversely affect its scalability since the WTC look-up table is used only during traffic classification at the ingress point of the DiffServ domain. This WTC look-up table keeps per-flow per-DSCP records, which can be viewed as a variant of unicast Multi-Field (MF) classification [18]. After multicast weighted traffic conditioning is done, traffic can be aggregated without discriminating whether they are multicast traffic or not. In terms of traffic treatment, DAM is a *per-aggregation* based scheme rather than a *per-flow* based scheme.
- **Flexibility**. DAM is independent of the underlying routing protocol. Therefore, it can work in heterogeneous networking environments. This routing independent feature also allows the coexistence of DAM with QoS-enabled multicast routing techniques like layered multicasting, multi-channel multicasting, multi-path multicasting and multiple-unicast multicasting. Further, DAM is isolated from the implementation details of BB architecture. So any changes made in the evolving BB design will have little impact on it.
- **Feasibility**. DAM is implemented in such a way that it requires routers to make only slight modifications. Every DiffServ capable router should add a "DSCP" field in the forwarding table. For DiffServ edge routers, it needs to modify DiffServ traffic classifier and maintain WTC look-up tables. However, these changes are unavoidable to support correct traffic metering and pricing scheme in DiffServ multicasting. Also, DAM can be easily adapted to the situations where core routers are not multicast capable by implementing only the Algorithm 2 at the edge routers.
- **Complexity**. DAM adopts simple algorithms at both edge routers and core routers. The major overhead of these algorithms is the weighted traffic conditioning at edge routers. In DAM, QoS is provided by the underlying DiffServ architecture. It avoids the complicated procedures of searching QoS-satisfied paths.

5.2. Quantitative analysis

In DiffServ architecture, senders or receivers are not directly attached to the core routers. So multicasting can save bandwidth only when a core router is the bottleneck, which is the case in actual networking environments. The analysis in this section assume that after traffic conditioning at the edge routers, core routers still may get congested.

The quantitative analysis mainly focus on EF UDP traffic within a single DiffServ domain. Other results are shown in Section 5.3. Notations used in this section are listed in Fig. 10.

5.2.1. Amount of EF output vs. input at the DiffServ edge router

The following equations show the relationship between the amount of EF input and actual EF outputs from a DiffServ domain for a multicast operation.

$$T_{wtc} = \begin{cases} (1 + a * b) * T, & \text{where } (1 + a * b) * T \leqslant T_c \\ Tc, & \text{where } (1 + a * b) * T > T_c, \end{cases}$$

$$T_{Normal} = \begin{cases} (1 + a * b) * T, & \text{where } T \leqslant T_c \\ (1 + a * b) * T_c & \text{where } T > T_c. \end{cases}$$

T	input EF traffic amount at an edge router in a DiffServ domain
T_c	maximum EF Input capacity of an edge router
a	ratio of EF multicast traffic over total EF traffic at an edge router
b	average duplication times within a DiffServ domain
N	number of DiffServ domains that a multicast tree spans
Pa	probability of having sufficient resources in a DiffServ domain
Pm	probability of having a branch marked higher than or equal to requested QoS in a DiffServ domain

Fig. 10. Notations.

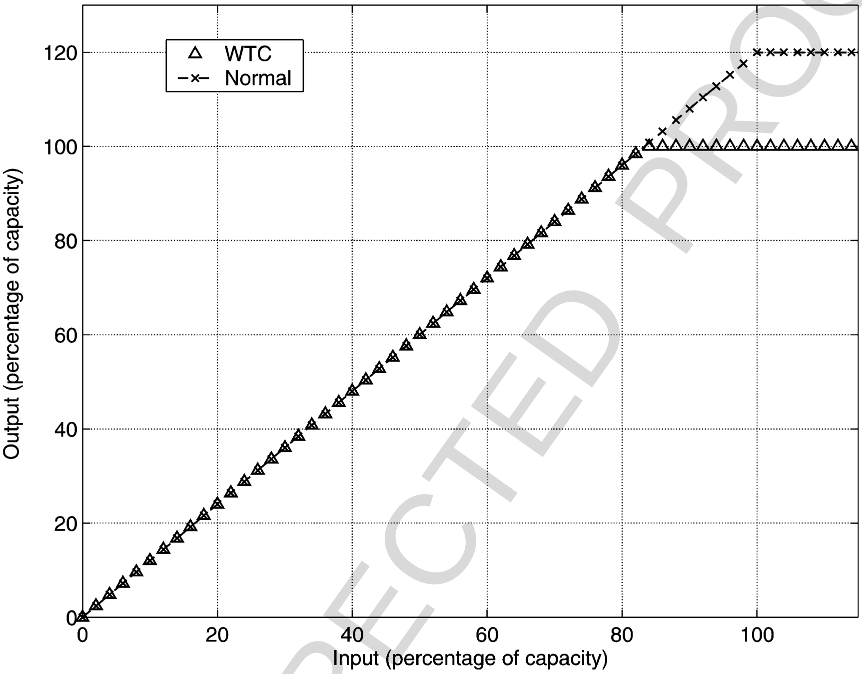

Fig. 11. Performance of EF at a DiffServ edge router ($a = 0.1$, $b = 2$).

For the Normal DiffServ multicasting, multicast flows marked as EF are duplicated in the domain. Therefore, the actual amount of outgoing traffic will be $(1 + a{*}b{*}T)$, rather than T. However, the edge routers are unaware of the actual traffic amount. They only drop packets when T exceeds their capacity. For the proposed WTC scheme, packets will be dropped based on actual accounting of outgoing EF traffic. That is, packets will be dropped when $(1 + a{*}b{*}T)$ is greater than T_c.

The numerical results are plotted in Fig. 11. It shows that WTC approach fully conform to the SLAs since the actual output does not exceed its EF input capacity. But the normal multicast DiffServ method violates the SLA.

5.2.2. *Message overhead*

DAM is associated with three types of signaling messages: non-AC/BB signaling, AC signaling, and BB signaling. The first type of messages includes "JOIN", "REQUEST" and "UPDATE". It can be observed from the Algorithms 1 and 2 that the number of these non-AC/BB signaling is a linear function of the number of DiffServ domains that a multicast tree spans. AC signaling is required in DAM for avoiding the NRS problems. Logically,

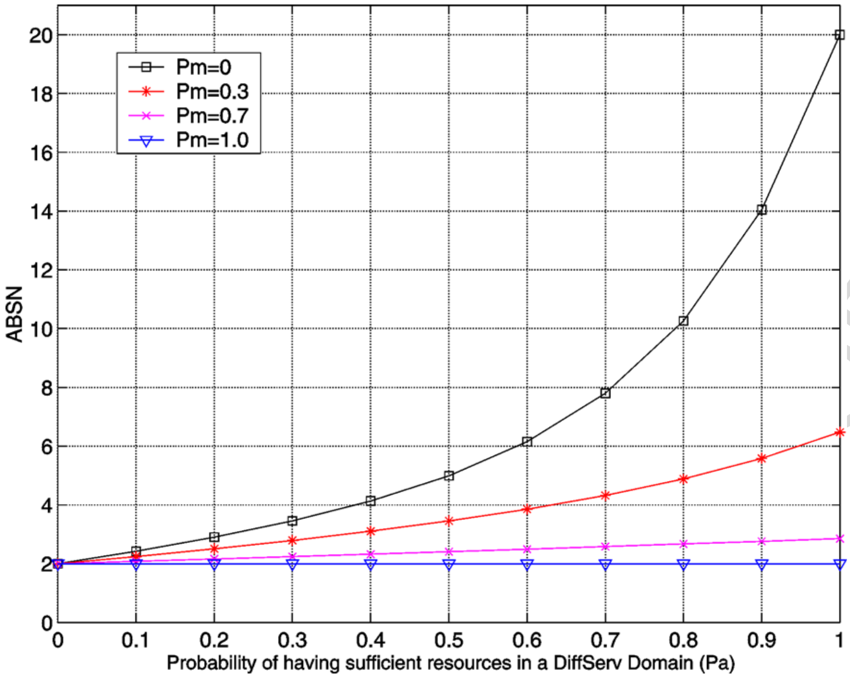

Fig. 12. Average BB signaling of DAM.

each new branch of a multicast flow requires one round of signaling. The last type of messages is related to BB. In DiffServ architecture, BBs will be signaled for every SLA negotiation or resource allocation request. For any approach based on DiffServ, it is essential to reduce the BB signaling overhead. So we focus on quantifying BB signaling overhead in this part.

Current Q-bone BB design adopts a "nailed up" model [17]. BBs will stop signaling the next domain if the negotiation between two intermediate domains produce a negative result. Thus the signaling will be stopped at the ith (except the first and last) domain when previous $i-1$ domains have sufficient resources and the ith domain does not. Assuming P_a is uniformly distributed, the Average BB Signaling Number (ABSN) for a unicast EF flow crossing N DiffServ domains is given by,

$$ABSN(N, P_a) = \begin{array}{l} 2 + 2(N-1)P_a^{N-1} \\ +2\sum_{i=2}^{N-1}(P_a^{i-1}(1-P_a)(i-1)). \end{array}$$

For DAM, BB signaling starts from the joining domain instead of the domain in which the multicast tree root is located. The joining domain is the one which has a branch marked higher than or equal to the requested QoS and is the nearest to the new receiver. So the average of BB signaling can be described by,

$$ABSN_{DAM} = \begin{array}{l} 2P_m + (1-P_m)^{N-1}ABSN(N, P_a) \\ +\sum_{i=2}^{N-1}(P_m ABSN(i, P_a)(1-P_m)^{i-1}). \end{array}$$

We further define the Relative BB Signaling Cost (RBSC) as: $RBSC = ABSN(N, P_a)/(ABSN_{DAM})$.

Figure 12 shows ABSN with varying P_a and P_m. We observe that $ABSN_{DAM}$ decreases when P_a decreases, which means that when network resources become scarce, the BBs are less likely to be signaled. This feature relieves the signaling overhead of the network when congestion happens. Figure 12 also shows that $ABSN_{DAM}$ increases when P_m decreases. In the worst case, when P_m equals to 0, the $ABSN$ is the same as the unicast case.

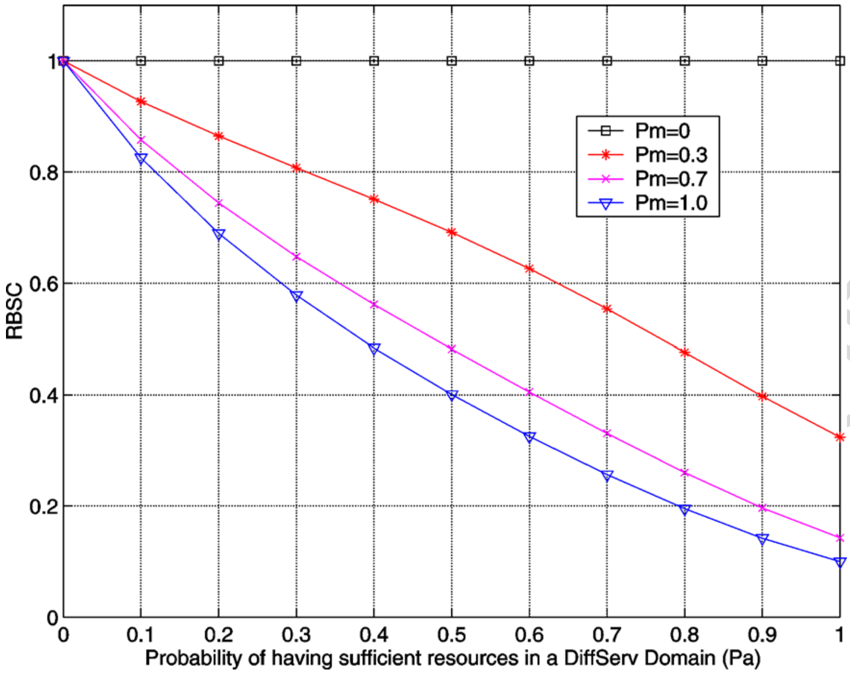

Fig. 13. Relative BB signaling cost of DAM.

Figure 13 shows the variation of $RBSC$ with respect to P_m and P_a, which indicates that the relative BB signaling cost drops when P_m increases. The BB signaling overhead analysis reveals two features of DAM. First, BBs will not be over-burdened when congestion occurs. Second, the relative BB signaling cost is less than or equal to the unicast case.

5.3. *Simulations results*

We have implemented the normal DiffServ multicasting, and DAM on the NS simulator [19]. The goal of the simulation is to evaluate the above approaches by comparing the packets transmission ratio, which is defined as the ratio of the number of packets transmitted to the total number of packets. This study is focused on a single DiffServ domain. In the simulation, we assume that the multicast traffic is based on UDP. For each scenario, unicast UDP and TCP traffic are studied. The network topology of the simulation is illustrated in Fig. 14. The bandwidth of each link is 10 Mb. Consider that S1 is delivering multicast packets at a rate of 1 Mb per second through ER1, CR and ER3 to a multicast receiver R1. Host R2 wants to join the multicast group. Existing unicast traffic flow aggregations are: ER1 to ER4 – 4 Mb BE; ER3 to ER4 – 2 Mb EF; and ER2 to ER4 – 6 Mb AF.

If the multicast flow is EF traffic and the maximum rate of EF traffic which are allowed to enter the domain at ER1 is 2 Mb, then DAM produces the same results as the normal DiffServ approach. EF traffic has the highest priority level, and no packets are dropped unless the amount of EF traffic exceeds the link capacity. Thus for the EF multicast flow, we mainly study its impact on other traffic classes, such as AF traffic and BE traffic.

Figure 15 illustrated the EF traffic simulation results. For AF class traffic in this domain, we have studied both UDP traffic and TCP traffic. Both results indicate that normal DiffServ multicasting noticeably reduces the packet transmission ratio of BE traffic, while DAM approach has little impact on the existing traffic. The results agree with the quantitative analysis presented earlier in Section 5.2.

If the multicast flow belongs to AF traffic and the maximum rate of AF traffic that is allowed to enter the domain at the edge router ER1 is 5 Mb, then a part of the AF traffic will be marked down to BE if DAM technique is adopted.

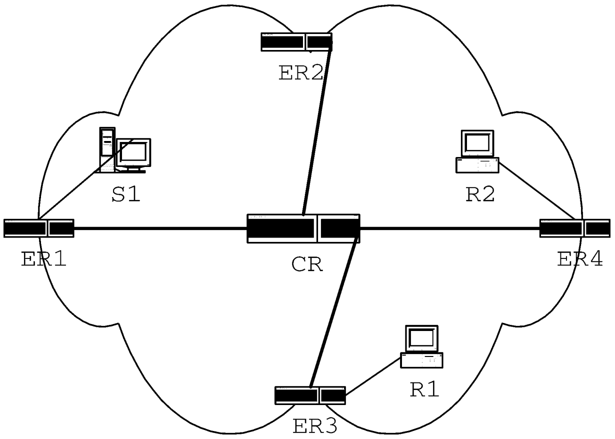

Fig. 14. Network topology of the simulation environment.

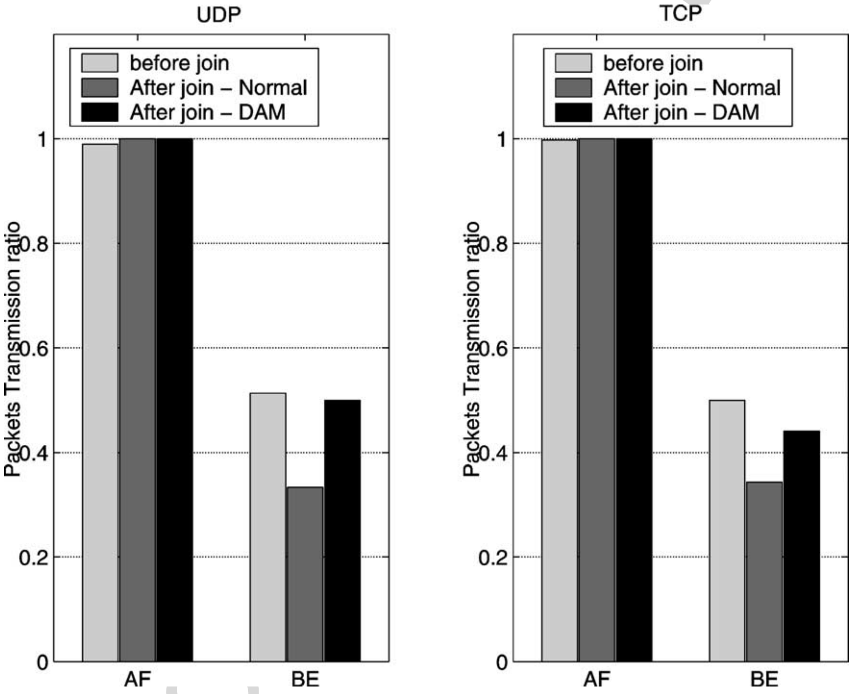

Fig. 15. EF multicast results.

The simulation results of AF multicast traffic are shown in Fig. 16. When the existing AF traffic is UDP based, the normal DiffServ multicasting approach produces better results for the new AF multicast flow at the cost of severely dropping the BE traffic. While DAM forms a compromise between the new multicast traffic and the existing BE flows. The packet transmission ratio of the multicast flow is about 0.9 with DAM without any significant impact on the BE traffic. When other AF traffic belongs to TCP, the results demonstrate the same trends except that unicast TCP AF traffic remain unchanged in terms of packet transferred ratio. This behavior is due to the TCP congestion control mechanism.

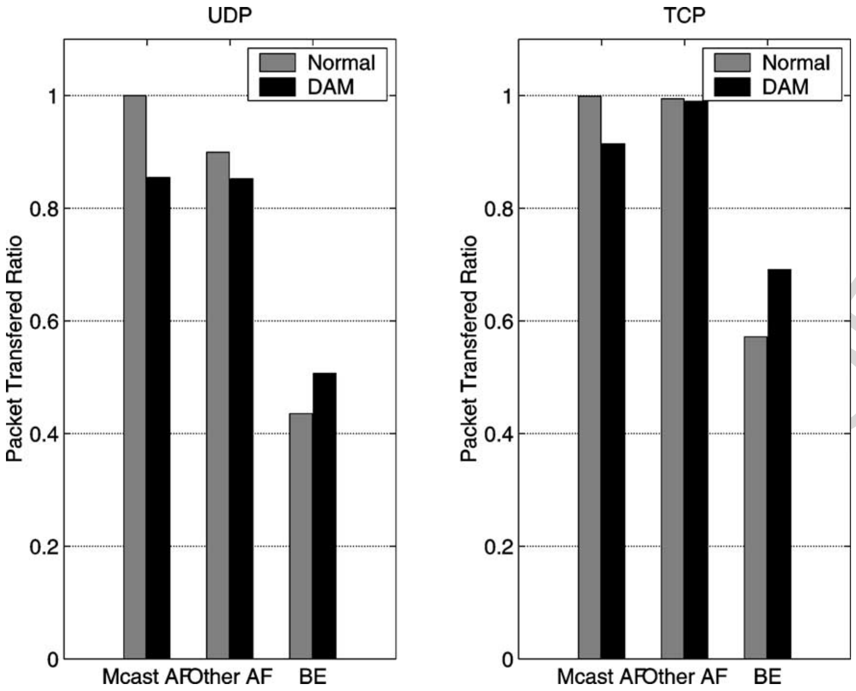

Fig. 16. AF multicast results.

Both EF and AF simulation results indicate that WTC is necessary when applying per-aggregation-based resource management schemes in DiffServ domains. SLA violation problems can be avoided in DAM without per-flow resource management approaches like RSVP.

As discussed in Section 3.3, HDE approach is adopted in DAM to improve fairness. For the same network topology, we assume one EF multicast flow which originates from ER1 is remarked to AF at CR for a sub-tree through ER4. Comparisons have been made with unicast AF flows traveling through ER1 to ER4 to study the packets transfered ratio. Traffic distribution on link CR to ER4 is: 2 Mb EF, 1 Mb AF multicast, 6 Mb unicast AF and 6 Mb unicast BE. Figure 17 clearly illustrated that the multicast AF traffic presented nearly the same performance as that of unicast UDP AF traffic when HDE is used. For unicast TCP AF traffic case, around 10% of the multicast AF packets are dropped with HDE, while nearly no multicast AF packets get dropped without HDE. Packet transferred ratio is not affected for TCP AF traffic. But the average transmission rate increases about 4% with HDE. Both the results show that HDE scheme ensures fairness between AF multicast flows and AF unicast flows.

In short, DAM technique avoids the SLA violation problem and unfairness issue introduced in DiffServ multicasting environments.

6. Conclusions

In this paper, we proposed a DiffServ-aware multicasting (DAM) technique to provide QoS in multicasting. In DAM, the NRS problem is solved by Weighted Traffic Conditioning (WTC) at the edge routers, and the heterogeneity in QoS requirements of the receivers are handled by receiver-initiated marking (RIM). Fairness is achieved with Heterogeneous DSCP Headers Encapsulation (HDHE). DAM protocol can be easily integrated with the existing DiffServ model for the Internet. DAM is scalable and also independent of the routing protocol.

DAM requires very little modifications in the currently proposed DiffServ architecture. The changes include adding a “DSCP” field in every multicast capable router, adding multicast traffic classifier and implementing the

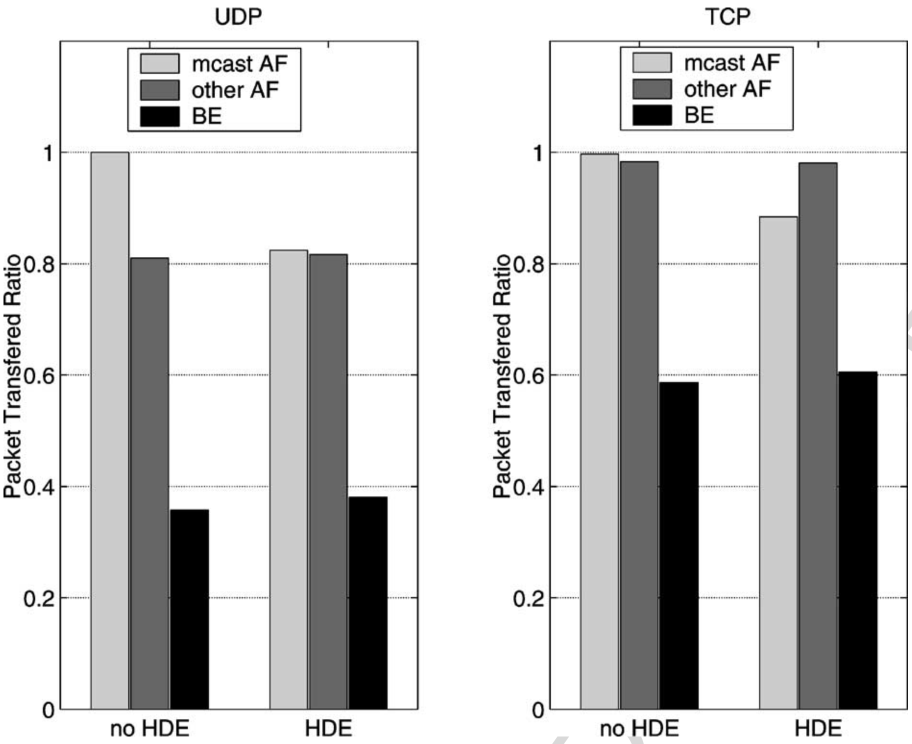

Fig. 17. AF Fairness results.

weighted look-up table at DiffServ edge routers. These added features introduce little overhead, but greatly enhance the functionalities of DiffServ multicasting. Through analyses and simulations, we have shown that DAM conforms to the SLAs between DiffServ domains while requiring a simple and scalable resource management scheme.

Acknowledgement

This research was supported in part by the National Science Foundation through the grants CCR-0296070 and ANI-0296034.

References

[1] P. Ferguson and G. Huston, *Delivering QoS on the Internet and in Corporate Networks*, John Wiley & Sons Inc., 1998.

[2] S. Deering, Multicast routing in Internetworks and extended LANs, in: *Proceedings of ACM SIGCOMM*, 1988, pp. 55–64.

[3] D. Kosiur, *IP Multicasting: The Complete Guide to Interactive Corporate Networks*, John Wiley & Sons Inc., 1998.

[4] D. Thaler and M. Handley, On the aggregatability of multicast forwarding state, in: *IEEE INFOCOM*, 2000.

[5] K. Nicholds, V. Jacobson and L. Zhang, A two-bit differentiated services architecture for the Internet, Internet draft. URL:http://diffserv.lcs.mit.edu/Drafts/draft-nichold- diff-svc-arch-00.txt.

[6] K. Carlberg and J. Crowcrof, Building shared trees using a one-to-many join mechanism, *ACM Computer Communication Review* (Jan) (1997), 5–11.

[7] M. Faloutsos, A. Banerjea, and R. Pankaj, QosMIC: Quality of Service sensitive multicast Internet protocol, in: *SIGCOMM*, 1998, pp. 144–153.

[8] S. Chen, K. Nahrstedt and Y. Shavitt, A QoS_Aware Multicast Routing Protocol, *IEEE Journal on Selected Areas in Communication* (2000), 1594–1603.

[9] B. Wang and J. Hou, Multicast routing and its QoS extension: problems, algorithms and protocols, *IEEE Network* (Jan/Feb) (2000), 22–36.

[10] J. Wroclawski, The Use of RSVP with IETF Integrated Services, Internet draft, RFC 2210.

[11] K. Fujikawa and I. Sheng, Simple Resource ReSerVation Protocol (SRSVP), Internet draft, URL: http://search.ietf.org/internet-drafts/draft- fujikawa-ric-srsvp-00.txt.

[12] R. Bless and K. Wehrle, Group communication in differentiated services networks, in: *IQ Workshop at CCGRID 2001*, 2001, pp. 618–625.

[13] R. Bless and K. Wehrle, A lower than best-effort per-hop behavior, Internet draft, URL: http://www.ietf.org/internet-drafts/draft-bless-diffserv-lbe-phb-00.txt.

[14] F. Baker, C. Iturralde and F. Le Faucheur, Aggregation of RSVP for IPV4 and IPV6 Researvations, RFC 3175, Sep, 2001.

[15] R. Cohen and G. Kaempfer, A unicast-based approach for streaming multicast, in: *IEEE INFOCOM*, 2001.

[16] P. Marbach, Pricing differntiated services networks: bursty traffic, in: *IEEE INFOCOM*, 2001.

[17] R. Neilson, J. Wheeler, F. Reichmeyer and S. Hares, A disscussion of bandwidth Broker Requirements for Internet2 qbone deployment, version0.7, Qbone draft, URL: http://www.merit.edu/working.groups/i2-qbone- bb/doc/BB_Req7.pdf.

[18] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, An architecture for Differentiated Services, RFC 2475, IETF, Dec, 1998.

[19] *Network Simulator – NS (version 2)*, http://www-mash.cs.berkeley.edu/ns/.

[20] A. Terzis, L. Wang, J. Ogawa and L. Zhang, A two-tier resource management model for the Internet, in: *GLOBECOM*, Vol. 3, 1999, pp. 1779–1791.

[21] M. Bilmer, T. Braun, Evaluation of bandwidth broker signaling, in: *ICNP Proceedings*, 1999, pp. 145–152.