A Random Early Demotion and Promotion Marker for Assured Services

Fugui Wang, Prasant Mohapatra, *Senior Member, IEEE*, Sarit Mukherjee, and Dennis Bushmitch

***Abstract*—The differentiated services (DiffServ) model, proposed to evolve the current best-effort Internet to a quality-of-service-aware Internet, provides packet level service differentiation on a per-hop basis. The end-to-end service differentiation may be provided by extending the per-hop behavior over multiple network domains through service level agreements between domains. The edge routers of each of the domains monitor the aggregate flow of the incoming packets and demote packets when the aggregate incoming traffic exceeds the negotiated interdomain service agreement. A demoted packet may encounter other edge routers on its path that have sufficient resources to route the packet with its original marking. In this paper, we propose a random early demotion and promotion (REDP) technique that works at the aggregate traffic level and allows 1) fair demotion of packets belonging to different flows, and 2) easy and fair detection and promotion of the demoted packets. Using early and random decisions on packets REDP ensures fairness in promotion and demotion. It uses a three color marking mechanism, reserving one color for differentiating between a demoted packet and a packet with the original out-of-profile marking. We experiment with the proposed REDP scheme using the $ns2$ simulator for both TCP and UDP streams. The results demonstrate the fairness of REDP scheme in demoting and promoting packets. Furthermore, we show a variety of results that demonstrates that REDP provides better assured services compared to the previously proposed RIO scheme with or without the provision of promotion.**

***Index Terms*—Assured services, demotion, differentiated services, Internet, promotion, quality of service, random early detection.**

I. Introduction

THE CURRENT Internet uses the best-effort service model. In this model the network allocates bandwidth among all the contending users as best as it can and attempts to serve all of them without making any explicit commitment to rate or any other service quality. With the proliferation of multimedia and real-time applications, it is becoming more desirable to provide certain quality of service (QoS) guarantee [1] for Internet applications. Furthermore, several enterprises are willing to pay an additional price to get preferred service in return from the Internet service providers. The Internet Engineering Task Force (IETF) [2] has proposed a few service models and mechanisms to ensure Internet QoS. Notably among them are the Integrated Services (IntServ) [3] model and the Differentiated Services (DiffServ) model [4]. Like the circuit-switched service in the current telephone system, IntServ could provide per-flow QoS guarantee. However, IntServ has two major drawbacks [5]. First, the amount of state information increases proportionally with the flow leading to poor scalability at the core routers. Second, implementation of IntServ requires changes in the Internet infrastructure. DiffServ, on the other hand, provides simple and predefined per-hop behavior (PHB) level service differentiation in the Internet core routers. Per-flow or flow aggregate marking, shaping, and policing are done at the edge router. Thus it does not suffer from the scalability problem and has less requirement from the Internet infrastructure.

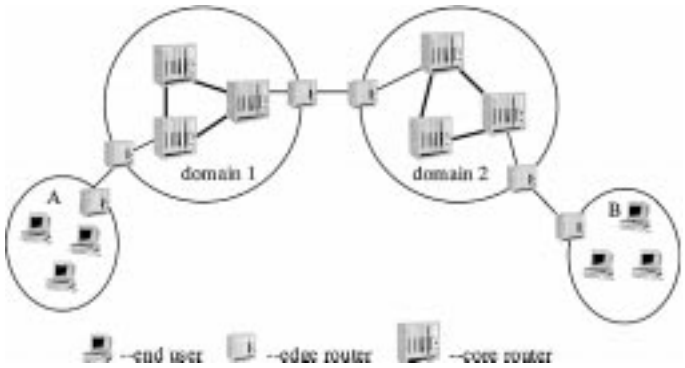

Fig. 1. A typical DiffServ architecture.

Today's Internet comprises of multiple interconnected autonomous domains, or administrative domains [6]. Each domain has core routers that are interconnected by backbone networks. End users or the other domains are interconnected to the each other through edge routers. A typical DiffServ architecture is shown in Fig. 1. Before entering a DiffServ domain, a packet is assigned a DiffServ Code Point (DSCP) by the marker located in the edge router [7]. When the packet reaches a DiffServ aware router, the DSCP of the packet will be checked to determine the forwarding priority of the packet. The DSCP of a packet may be changed when it crosses the boundary of two domains. For example, in Fig. 1, a packet sent from host A to host B may be marked as high priority DSCP when it enters domain 1. At the boundary of domain 1 and domain 2, if domain 1 has not negotiated enough traffic forwarding rate with domain 2 for that priority, the marker at domain 2 may have to remark that packet as a low priority DSCP before it would forward the packet to domain 2. Currently, IETF has defined one class for expedited forwarding (EF) [8] and four classes for assured forwarding (AF) [9].

Manuscript received October 15, 1999; revised April 15, 2000.

F. Wang and P. Mohapatra are with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824 USA (e-mail: wangfugu@cse.msu.edu; prasant@cse.msu.edu).

S. Mukherjee and D. Bushmitch are with the Panasonic Information and Networking Technologies Lab, Princeton, NJ 08540 USA (e-mail: sarit@research.panasonic.com; db@research.panasonic.com).

Publisher Item Identifier S 0733-8716(00)09233-7.

EF was originally proposed by Van Jacobson in [10] as premium services. It is expected that premium traffic would be allocated only a small percentage of the network capacity and would be assigned to a high-priority queue in the routers. It is ideal for real-time services such as IP telephony, video conferences and the like.

AF was first proposed by Clark in [11] as assured services. Originally, it was proposed to use the RED-in/out (RIO) [12] approach to ensure the "expected capacity" specified by the traffic profile. The basic idea is, upon each packet arrival, if the traffic rate is within the traffic profile, the packet is marked as "in," otherwise, it is marked as "out." In a DiffServ aware router, all the incoming packets are queued in the original transmission order. However, during network congestion, the router preferentially drops the packets that are marked as "out." By appropriate provisioning, if we could make sure that the aggregate "in" packets would not exceed the capacity of the link, the throughput of each flow or flow aggregate could be assured to be at least the rate defined in the traffic profile. To ensure service differentiation, currently, the AF PHB [9] defined by IETF specifies four traffic classes with three drop precedence levels (or three colors) within each class. In all, there are twelve DSCP's for AF PHB. Within an AF class, a packet is marked as one of the three colors—*green*, *yellow*, and *red*—where *green* has the lowest drop probability and *red* has the highest drop probability. It is expected that with appropriate negotiation and marking, end-to-end minimum throughput could be assured or at least assured to some extent.

An Internet connection can span through a path involving one or more network domains. If we want to guarantee the end-to-end minimum throughput of the connection, we have to make sure that the aggregate traffic along the path does not exceed any of the interdomain negotiated service level agreements (e.g., the traffic rate) after this flow joins. This is very hard to ensure since the interdomain service agreement is not usually renegotiated at the initiation of each new connection. For assured services, the interdomain traffic rates are usually negotiated statically or updated periodically to avoid signaling overhead and scalability problem [4]. The negotiation is usually based on statistical estimation. So, the instantaneous aggregate flow rate may be higher or lower than the negotiated rate. In case of higher incoming flow rate, the intermediate marker demotes some of the "in" packets to "out" so that the aggregate rate of "in" packets conform to the negotiated rate. The demotion, although exercised at an aggregate flow level, should affect all the connections proportional to their current usage (i.e., fair demotion). On the other hand, if the incoming flow is lower, ideally the intermediate marker should reallocate the excess capacity and promote a "previously demoted" packet. This promotion should be fair across all connections as well.

In this paper we propose a new technique for the marking process at the edge routers. The proposed process is motivated by the observation that some of the "in" packets may get marked as "out" at nodes where the aggregate incoming traffic rate exceeds the available bandwidth. However, later in the connections' path, the available bandwidth may be enough to route these "out" packets (that were originally "in"). Therefore, there is a need to identify these demoted packets instead of clubbing them together with the packets that were marked "out" at the point of origination. Our technique addresses two important aspects of the marking process at the edge routers. First, in the case of demotion, it ensures that the proportion of packets demoted for each microflow is fair (with respect to their rates). Second, it proposes a mechanism to identify the demoted "in" packets and promotes them fairly across connections when a domain has excess capacity. The fairness is achieved early and by randomly making marking decisions on the packets. Identification of a previously demoted packet is ensured using the AF PHB specified packet markings. In order to support this, the marker uses a three-color (red, yellow, and green) marking process, where yellow is used as an indicator for temporary demotion. We have experimented with the marker on the $ns2$ [13] simulator. Our results show the effectiveness of the technique for both TCP and UDP traffic. The marking scheme is very fair in demoting and promoting packets and provides better performance to the in-profile traffic compared to the traditional leaky bucket scheme and the RIO scheme.

The rest of the paper is organized as follows. Section II discusses the demotion and possible promotion at the boundary of two domains. The benefit of providing promotion is also discussed in detail in this section. Section III proposes an REDP marker to fairly demote and promote packets at the domain boundary. Section IV studies the performance of REDP marker using $ns2$ network simulator. Section V discusses how to further improve the benefit from promotion by supporting three drop precedences in the core routers. Parameter sensitivity of the proposed marker is discussed in Section VI, followed by the conclusions in Section VII.

II. Interdomain Marking

A packet in the Internet travels from a source to its destination by getting routed through one or more network domains. According to the architecture of DiffServ defined by the IETF, neighboring domains negotiate service level agreement (SLA) with each other, which specifies how much traffic of each service level could be passed from one domain to the other domain. More technical details, such as the committed rate, maximum burst size, etc., are specified by the traffic conditioning agreement (TCA). Traffic conditioners (TC) are implemented at the edge routers to ensure that the aggregate traffic of any level should not exceed the traffic profile of the TCA. A simple example of TC is a leaky bucket marker used for RIO as shown in Fig. 2. The TCA between the upstream domain and the downstream domain specifies that r bits/s "in" traffic from the upstream domain could enter the downstream domain with a maximum burst size of b. The leaky bucket is fed with a constant rate of r bits/s. When a packet arrives from the upstream domain, if the packet has been marked as "out," TC simply forward it as "out." If the packet has been marked as "in," TC checks the leaky bucket to see whether there are enough tokens for this packet. If there is, the packet is forwarded as "in" and the packet size worth of tokens are deducted from the leaky bucket. Otherwise, the "in" packet is demoted to "out" and forwarded.

A more sophisticated two-rate–three-color marker (TR-TCM) [14] based on a similar mechanism was proposed recently as a possible candidate for the three color AF.

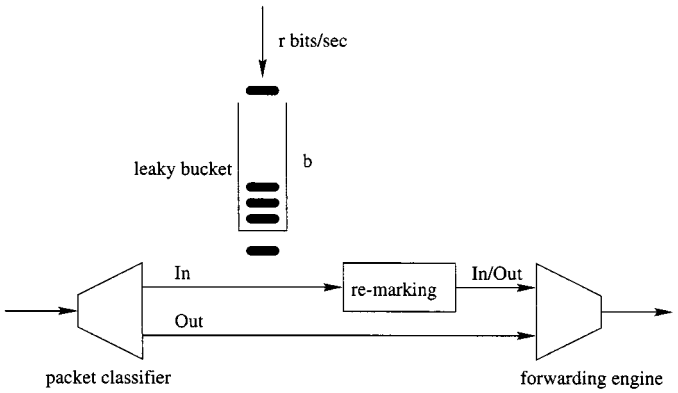

Fig. 2. A leaky bucket intermediate marking model.

The common idea behind the marker is that when the aggregate traffic of certain service level exceeds the rate defined in the traffic profile, the packet is demoted to a lower service level. However, in this model, if the traffic rate of the service level is lower than the rate defined in the traffic profile, lower service level packets are not promoted to higher level. The scheme does not promote a packet because of the problem in identifying the packets to promote. For example, assume that flow-1 has subscribed for certain throughput of assured services, and flow-2 has subscribed for best effort services. Both of them pass through several domains. Assume that some of the "in" packets of flow-1 are demoted while crossing the first domain. While crossing the second domain, if the TC has some extra "in" tokens and if promotions were allowed, the best effort traffic and the demoted traffic of flow-1 compete for getting promoted. In all fairness, the demoted packets of flow-1 should be promoted first. However, there are no identification marks in these demoted packets. By promoting the packets of both flows randomly, the assurance of the in-profile packets cannot be improved. The simulation result in Section IV-B of this paper supports this argument.

Usually, an end-to-end connection would cross multiple Diff-Serv domains. For assured services, static TCA based on statistical estimation is preferred for simplicity and ease of pricing. Since there is no end-to-end signaling and negotiation, demotion is unavoidable. If we use the marking model as has been proposed in the literature, once an "in" packet is demoted, it will be treated as an "out" packet for all of the remaining domains. Assuming the number of domains along the end-to-end connection is n, and the probability that a packet gets demoted through each domain boundary is p, the end-to-end demotion probability of a packet would be $1-(1-p)^n$. However, some of the demotion decisions could be reverted if we can identify the demoted packets and promote them as soon as we have excess resources available in the downstream domain. Based on these motivations, we propose a three-color demotion-promotion scheme in the following section.

III. REDP Marker

In this section, we propose a new technique called random early demotion and promotion (REDP) for managing the inter-domain flow control. The main aspects of the REDP scheme are

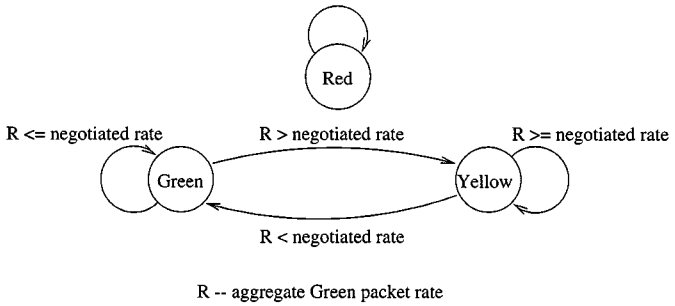

Fig. 3. State diagram of demotion and promotion within three colors.

the provision of promotion of the demoted packets and making the demotion/promotion process fair. The provision of promotion enhances the performance of the assured traffic, whereas the early randomness in packet marking decision ensures the fairness of the proposed scheme. These performance measures are quantified and justified in the next section. Here we describe the marking process and the framework of the REDP scheme. Notice that the initial marking of packets at the host markers can be done on a per-flow basis. However, the intermediate marking must be done on the aggregate level for the ease of scalability.

We use a variation of the tricolor marking model for the REDP scheme. Therefore, each packet can be marked as *green, yellow,* or *red*. Suppose an end user submits an expected rate r. Initially, the local domain configures a leaf marker for the flow. A packet from this flow is marked as *green* if it is in-profile and *red* if it is out-of-profile. None of the packets is marked as *yellow*. Intermediate markers are implemented in the TC of domain boundaries. While crossing a domain boundary, a *green* packet is demoted to *yellow* if the aggregate *green* packet rate[1] exceeds the negotiated rate at the intermediate marker. A *yellow* packet is promoted to *green* if the aggregate *green* packet rate is lower than the negotiated rate. A *yellow* packet is never demoted to *red* and a *red* packet is never promoted to *yellow*. Thus, *yellow* is specifically used to memorize the demoted *green* packets. When we are able to promote, we only try to promote the *yellow* packets. In other words, we would only promote the assured packets that were demoted. The motivation for reserving the *yellow* packets to remember the previous state of a high priority packet came from the fact that different traffic classes, not the three-color scheme per traffic class, gives effective isolation between TCP and UDP flows [19]. Two colors per class is enough for service differentiation within a class [19]. The third color can be used more effectively in the manner proposed in this paper. The state diagram of the demotion–promotion algorithm is shown in Fig. 3.

The leaky bucket is a deterministic flow control network element that can be used as a traffic marker. Like the drop-tail queue, a simple leaky bucket demotes all packets that arrive when there are no tokens available. As argued in [16], much of the Internet traffic is highly periodic, either because of periodic sources (e.g., real-time audio or video) or because window flow control protocols have a periodic cycle equal to the connection round trip time (e.g., a network-bandwidth limited TCP

[1]Many microflows may pass from the upstream domain to the downstream domain through the intermediate marker. Aggregate *green* packet rate is the sum of the rate of all of the *green* packets of these microflows.

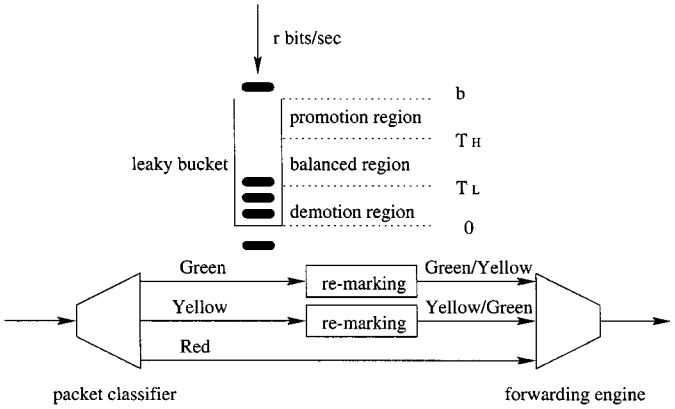

Fig. 4. REDP marker.

bulk data transfer). This phase effect could bring unfairness in the demotion and promotion among different microflows as addressed in [15]. The following (concocted) example explains the unfairness of the phase effect (or synchronization). Suppose all packets of two streams are originally marked as *green*. They have the same rate and same packet size and are aggregated in the marker. Suppose the packet from the streams (1 and 2) are interleaved in the following pattern, 1 212 121 212. . ., and the marker has to demote 50% of the packets from the aggregate flow, i.e., every other packet must be demoted. Then all the packets from one flow will be demoted, while all the packets from the other flow will remain unaffected. Phase effect could also bring about unfairness in promotion. Detail discussions on the phase effect have been reported in [16]. Introducing randomness in the packet selection process of the flow control mechanism could solve the problem. An example is the random early detection (RED) gateway [17] that reduces the unfairness of the drop-tail queue. We apply a similar concept to the leaky bucket marker by introducing randomness and early decisions on the packet marking process. In addition, as discussed earlier, we allow the promotion of the *yellow* packets based on bandwidth availability. We call this marker the random early demotion and promotion (REDP) marker.

An REDP marker is implemented using a leaky bucket. A promotion threshold is set in the leaky bucket. If the tokens in the leaky bucket exceed the promotion threshold and an arriving packet is *yellow*, it is promoted to *green*. Similarly, a demotion threshold is used in the leaky bucket. If the number of tokens in the leaky bucket is less than the demotion threshold, an arriving *green* packet is demoted to *yellow*. Using this scheme, we can also detect whether the aggregate rate of the arrival of *green* packets is lower or higher than the negotiated rate.

The marking model is shown in Fig. 4. Two thresholds, T_L and T_H, divide the leaky bucket into three regions—*demotion region*, *balanced region*, and *promotion region*. Initially, the token count is set within the *balanced region*. Three situations can arise during the marking process.

1) *Balanced:* If the arriving rate of *green* packets is equal to the token filling rate r, the token consumption rate is the same as the token filling rate. Therefore, the number of token in the bucket remains in the *balanced region*. Each packet is forwarded without changing the color.
2) *Demotion:* If the arriving rate of *green* packets exceeds r, the token consumption rate exceeds the token filling rate. The number of tokens decreases and the token level falls into the *demotion region*. In the *demotion region*, each arriving *green* packet is randomly demoted to *yellow* with a probability of P_{demo}, where P_{demo} is a function of the token count (TK_{num}). A simple example of the function could be

$$P_{demo} = (T_L - TK_{num})\mathrm{MAX}_{demo}/T_L$$

where MAX_{demo} is the maximum demotion rate. When the leaky bucket runs out of tokens, each arriving *green* packet is demoted to *yellow*.
3) *Promotion:* If the arriving rate of *green* packets is less than r, the token filling rate exceeds the token consumption rate. The number of tokens increases and the level reaches the *promotion region*. In the *promotion region*, each arriving *green* packet will still be forwarded as *green*, consuming a certain number of tokens. Each arriving *yellow* packet will be randomly promoted to *green* with a probability of P_{promo}, where P_{promo} is a function of the token count in the leaky bucket (TK_{num}). An example of the function is

$$P_{promo} = (TK_{num} - T_H)\mathrm{MAX}_{promo}/(b - T_H).$$

The REDP scheme removes the phase effect of periodical flows by detecting the arriving rate of the *green* packets early and by promoting or demoting packets randomly. During demotion, it keeps the number of demoted packets of each flow approximately proportional to the number of *green* packets of that flow. Similarly during promotion, it keeps the number of promoted packets of each flow approximately proportional to the number of *yellow* packets of that flow.A

The DiffServ core routers could support either two or three drop precedences. If it supports two drop precedences (e.g., RIO), *green* is deemed as "in." Both *yellow* and *red* are deemed as "out." If it supports three drop precedences, *green* has the lowest drop probability and *red* has the highest drop probability.

IV. Performance Study

In this section, we analyze the performance and effectiveness of the REDP scheme. In the previous section, we claimed that the REDP has two major advantages. First, the demotion and promotion performed by REDP is fair across the connections. Second, by allowing the promotion of demoted packets, REDP improves the performance of the assured traffic. We quantify these two measures in this section through experiments using the $ns2$ simulator. Both UDP and TCP sources are analyzed to show the performance improvement.

A. Fairness of Demotion and Promotion

Note that the demotion and promotion algorithm employed in the REDP marker uses the same mechanism (i.e., random and early decisions) to ensure fairness. Here, to avoid repetition, we only show the fairness of demotion. Fig. 5 depicts the simulation topology used to study the fairness of demotion. Hosts

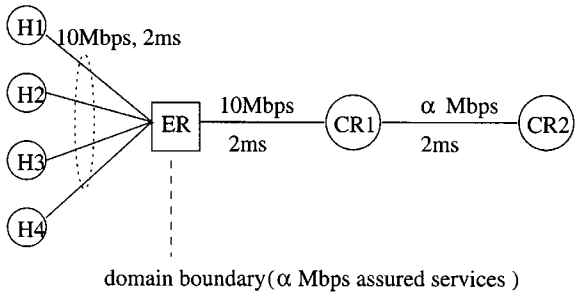

Fig. 5. Simulation topology used to study the fairness of demotion.

H1, H2, H3, H4 each has a leaf marker implemented inside. Each of the hosts has a 0.5 Mb/s assured service profile. So initially each host could have up to 0.5 Mb/s packets marked as *green*. The remaining packets are marked as *red*. Each flow originates from a host and passes through multiple domains and terminates at CR2. After successfully crossing one[2] or several domains, the packets reach the edge router ER. ER is at the boundary of the two domains. Suppose, at ER, we don't have enough SLA to pass all the *green* packets to the downstream domain. Then some of the *green* packets need to be demoted to *yellow*. The goal of our experiment is to evaluate the fairness of different marking schemes. In other words, we investigate if equal proportion of the *green* packets of each flow would be demoted by the different marking schemes. In the following discussions, we analyze and compare the fairness of the following three schemes using our simulation: leaky bucket, REDP, and per flow marking.[3]

All the marking schemes are implemented in the ER of Fig. 5. The token filling rate of the leaky bucket is α Mb/s, where $\alpha < 2$ Mb/s. After being remarked by the marker, a packet is forwarded to the core router CR1 and then terminates at core router CR2. The link capacity between ER and CR1 is larger than the aggregate bandwidth of the four flows. The link capacity between CR1 and CR2 is exactly α Mb/s. Assured service is implemented in CR1 through the RIO scheme. In the core routers, all the *green* packets are treated as "in," both *red* and *yellow* packets are treated as "out." We implemented a simple RIO queue [12] in the $ns2$ simulator. Both "in" and "out" packets are buffered in the same queue. We use two sets of RED parameters for "in" and "out" packets. The RED parameters for "in" packets is: 45 packets, 60 packets, and 0.1 for $\min_{\text{in}}$, $\max_{\text{in}}$, and $P\max_{\text{in}}$, respectively, and 20 packets, 40 packets, and 0.5 for $\min_{\text{out}}$, $\max_{\text{out}}$, and $P\max_{\text{out}}$, respectively, where $\min_{\text{in}}$ and $\max_{\text{in}}$ represent the upper and lower bounds for the average queue size for "in" packets, and $P\max_{\text{in}}$ is the maximum drop probability for an "in" packet when the queue size is in the $[\min_{\text{in}}, \max_{\text{in}}]$ range. The $\min_{\text{out}}$, $\max_{\text{out}}$, and $P\max_{\text{out}}$ are the corresponding parameters for the "out" packets. This configuration ensures that the aggregate *green* packets from CR1 to CR2 is exactly the link capacity. So almost all the *green* packets would be forwarded, and almost all the *yellow* and *red* packets would be dropped. By computing and analyzing the throughput of each flow at CR2, we derive the fairness of the demotion for different markers. Theoretically, if the demotion is fair, each flow should get approximately the same throughput, that is, $\alpha/4$. We have used both UDP and TCP sources in our simulation to demonstrate the effectiveness of REDP for these two popular transport-layer protocols.

1) Fairness of Demotion for UDP Sources: In this simulation, we have assumed four UDP sources—udp1, udp2, udp3, udp4—starting from hosts H1, H2, H3, and H4, respectively. The sending rate of each flow is 0.6 Mb/s. Originally, 0.5 Mb/s is marked as *green* and the remaining 0.1 Mb/s is marked as *red*. In the first simulation, we choose $\alpha = 1.6$ Mb/s. So at edge router ER, 2 Mb/s *green* packets arrive but only 1.6 Mb/s of them could be marked as *green* before entering the downstream domain. If the marker implemented in ER is ideally fair, each flow should have 400 kb/s packets forwarded as *green* and 100 kb/s packets demoted as *yellow*. Because the bandwidth of the bottleneck link, from CR1 to CR2 is exactly 1.6 Mb/s, only the *green* packets could pass this link. All the *yellow* and *red* packets will be dropped here. So ideally, each flow should get 400 kb/s throughput. The simulation is executed for 50 s and we use the last 40 s to calculate the throughput of each flow. Calculation of the throughput is done in the same way for all the other results in this paper.

The throughput for different flows using different makers is shown in Fig. 6(a). If we use a leaky bucket marker in ER, the throughput of the four flows are highly biased. Flow2 only get about 200 kb/s, while flow3 and flow4 get about 500 kb/s each. This is because of the synchronization problem. The four UDP flows have the same rate and are sending data periodically. Most of the time when a *green* packet of flow2 comes, the leaky bucket happens to run out of tokens. If we use a per-flow based marker in ER, each flow gets close to 400 kb/s. If we use our REDP marker in ER, each flow also gets approximately 400 kb/s.

Fig. 6(b) shows another set of result with a different demotion ratio. Here, $\alpha = 1.2$ Mb/s, so ideally 300 kb/s of each flow could be passed as *green* and 200 kb/s should be demoted to *yellow*. From the result, we can observe that the throughputs are highly biased if we use leaky bucket marker in the ER. However, the REDP scheme removes the synchronization or phase effect and is very fair as is demonstrated by comparing its results to the ideal per-flow marking process.

Depending on the sending rate of each flow, the phase effect could be less or more serious. Next, we change the sending rate of each flow to 0.79, 0.73, 0.53, and 0.61 Mb/s, respectively, and repeated the simulation. Note that from each flow, 0.5 Mb/s is marked as *green* and the rest is marked as *red*. Fig. 7 shows the results. Fig. 7(a) and (b) shows that the throughputs of the four flows are still biased in the case of the leaky bucket marker. However, the degree of variance is higher. In both cases, the

[2]In the real world, it is unlikely that a *green* packet will get demoted when it reaches the first intermediate marker because the leaf markers are configured based on the capacity of the first intermediate marker. In this experiment, in order to simplify the simulation topology, we assume that the demotion happens when packets reach the first intermediate marker, that is, when it reaches edge router ER in Fig. 5.

[3]Per-flow marking is implemented in the following way. Assume that all the intermediate markers know the original submitted rate of each flow. Tokens assigned to each flow are proportional to its original submitted rate. This model, although should be the fairest among these three, needs per-flow monitoring and signaling. It may not be practical as an intermediate marker because of the scalability problem. Here we only use it as an ideal case to evaluate the fairness of REDP marker.

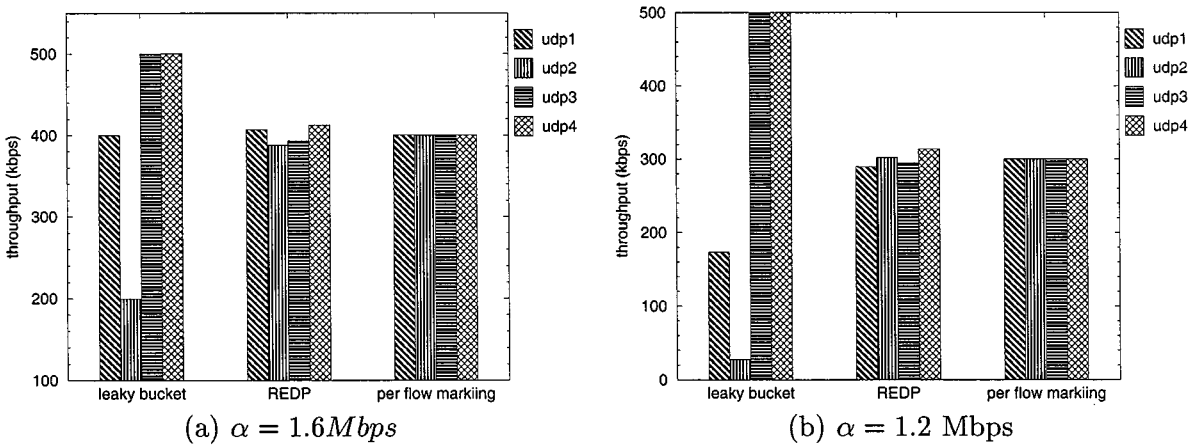

Fig. 6. Demotion fairness comparison: UDP sources, four flows have same sending rate.

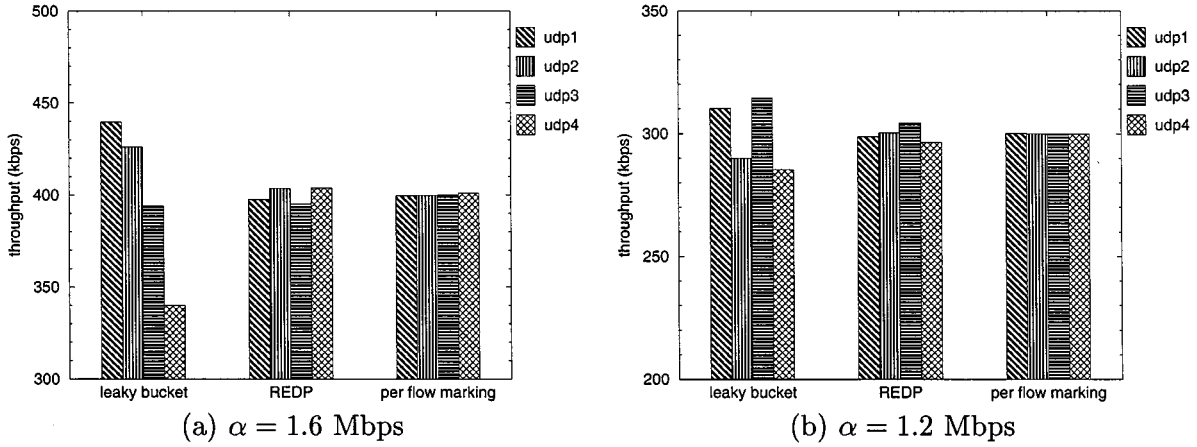

Fig. 7. Demotion fairness comparison: UDP sources, four flows have different sending rates.

REDP scheme achieves better fairness over the leaky bucket marker. The fairness of REDP is almost as good as the per-flow marking while it works on the aggregate flow level and thus does not have any scalability problem.

Phase effect is very common for UDP sources. We have done several simulations and have observed this effect frequently. Depending on the rate of each flow, the SLA, and the packet size, etc., the effect could vary. By using REDP marker, we incorporate a random component in the path. This randomness removes the deterministic phase effect so that it does fair demotion and promotion.

2) Fairness of Demotion for TCP Sources: According to the analysis in [16], TCP sources also have phase effect because of its sliding window flow control algorithm. A TCP source will not send the next burst of packets until it receives the ACK of the current burst of packets. So, the period is the roundtrip time (RTT) of the connection. In [16], the authors have shown that flows with similar RTT could get biased throughputs if they share a common link.

The topology of Fig. 5 is again used for this simulation, where the four UDP sources are changed to four TCP sources. The delay of the link between H1 and ER is changed to 1 ms, between H3 and ER is changed to 1 ms, and between CR1 and CR2 is changed to 10 ms. So the RTT of each flow is 26, 28, 26, 28 ms, respectively. The throughput of each flow is shown in Fig. 8. From the figure we could observe the phase effect when we use the leaky bucket marker. Both per-flow marking and REDP could increase the fairness of demotion. However, the fairness improvement of REDP marker over leaky bucket marker is not as obvious as using UDP sources. This is because TCP has its own flow control and congestion control algorithm [18].[4] Adding random components along the path could improve the fairness of TCP sources, but would not completely solve the problem.

B. Benefit from Promotion

Depending on the actual network traffic, a packet demoted at the boundary of a domain may or may not get dropped in that domain. If it is not dropped in that domain, it is preferable to promote it as soon as there are excess tokens in any of the downstream edge markers. This ensures that a packet does not get dropped under minor and transient congestions in the downstream domains. The proposed REDP marker could do both demotion and promotion.

The topology shown in Fig. 9 is simulated to study the benefit of promotion. ER1 and ER2 are two edge routers, each of

[4]We believe if the TCP sources are modified according to [18], a "better" fairness can be obtained through REDP. We did not carry out the experiments since the source does not remain traditionally TCP compliant.

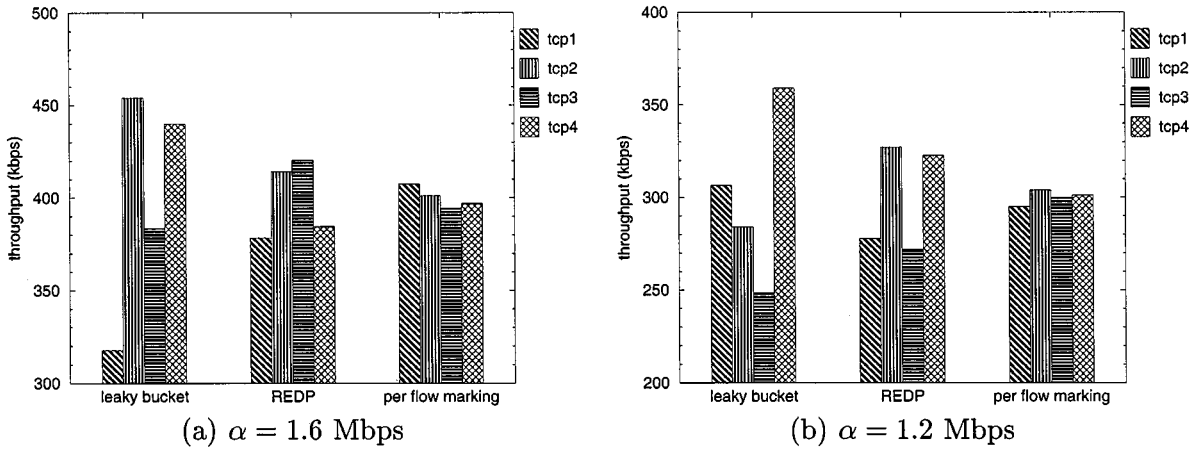

Fig. 8. Demotion fairness comparison: TCP sources.

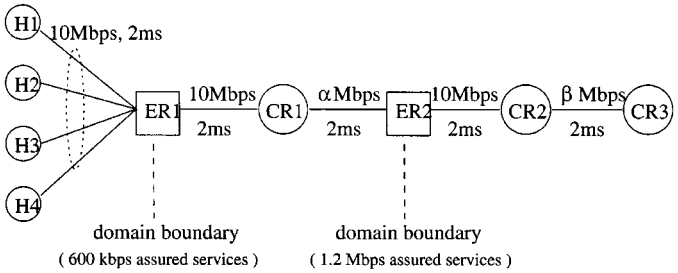

Fig. 9. Simulation topology used to study the benefit from allowing promotion.

which has a marker implemented in it. CR1, CR2, and CR3 are core routers with built-in RIO mechanism for flow control. Similar to the previous simulation, H1, H2, H3, H4 each has a flow starting from it and sinking at CR3. Each flow crosses two domain boundaries. At the first domain boundary defined by ER1, there is not enough SLA to forward all the *green* packets as *green*. Some of the *green* packets are demoted. Let us assume that at the second domain boundary, ER2, there is some excess SLA. So we have three choices.

1) *No Promotion:* We only use two colors, *green* and *red* (or "in" and "out"). In case of deficient SLA, *green* is demoted to *red*. In case of excess SLA, *red* is not promoted to *green*.
2) *Two-Color Promotion:* We only use two colors. In case of deficient SLA, *green* is demoted to *red*. In case of excess SLA, *red* is promoted to *green*. In this case, there is no distinction between a packet that is originally marked as *red* and a demoted packet that is also marked *red*.
3) *Three-Color Promotion:* We use three colors. In case of deficient SLA, *green* is demoted to *yellow*. In case of excess SLA, *yellow* is promoted to *green*. No demotion or promotion is done between *yellow* and *red*. In the core routers, *green* is treated as "in," both *yellow* and *red* are treated as "out."

We implement all these three alternatives and compare the performance in the following experiments.

1) UDP Sources: For the four hosts, assume that H1 and H3 each negotiate 500 kb/s assured services, and H2 and H4 use best-effort services. Four UDP flows, udp1, udp2, udp3, udp4 start from H1, H2, H3, H4, respectively, and sink at CR3. The rate of each flow is set at 500 kb/s. So initially udp1 and udp3 each has 500 kb/s packets marked as *green*, udp2 and udp4 each has 500 kb/s packets marked as *red*. At ER1, up to 600 kb/s *green* packets are allowed to be forwarded to the next domain. So 40% of the *green* packets are demoted here. We set $\alpha = 2$ Mb/s. So no congestion happens in this domain and the demoted packets will not be dropped in this domain. At ER2, up to 1.2 Mb/s *green* packets are allowed to be forwarded to the next domain. If we choose promotion, we could promote some of the *yellow* (for 3-color promotion) or *red* (for 2-color promotion) packets to *green*. We set $\beta = 1.2$ Mb/s. So within this domain, some of the *red* or *yellow* packets will be dropped. The link between CR2 and CR3 is the bottleneck. Fig. 10(a) shows the throughput of each flow under different marking schemes.

Without promotion, a demoted packet is treated as "out" for all of the remaining domains. So some of the packets will be dropped at the bottleneck link. Each of udp1 and udp3 gets about 400 kb/s throughput although they submitted 500 kb/s. If we use the 2-color promotion as described above, we can promote some of the *red* packets at ER2. However, since we cannot tell which one is initially marked as *red* and which one is demoted to *red*, both of them could be promoted to *green*, which would not improve the throughput assurance of udp1 and udp3. The simulation results support this point. We cannot improve the throughput assurance of flow1 and flow3 through 2-color promotion. In the 3-color promotion, we use *yellow* to memorize the demoted *green* packets. In ER2, only the *yellow* packets are promoted to *green*. So we could improve the bandwidth assurance of udp1 and udp3. Each of them gets a throughput of about 500 kb/s. This is where the REDP scheme benefits the most.

2) For TCP Sources: Now we change the four UDP sources to four TCP sources, keeping all of the other parameters unchanged. The simulation result is shown in Fig. 10(b). The result is similar to the previous simulation. No promotion and 2-color promotion have similar performance while the 3-color promotion improves the throughput assurance of tcp1 and tcp3. Thus, the concept of promotion used in the REDP scheme benefits both TCP and UDP traffics.

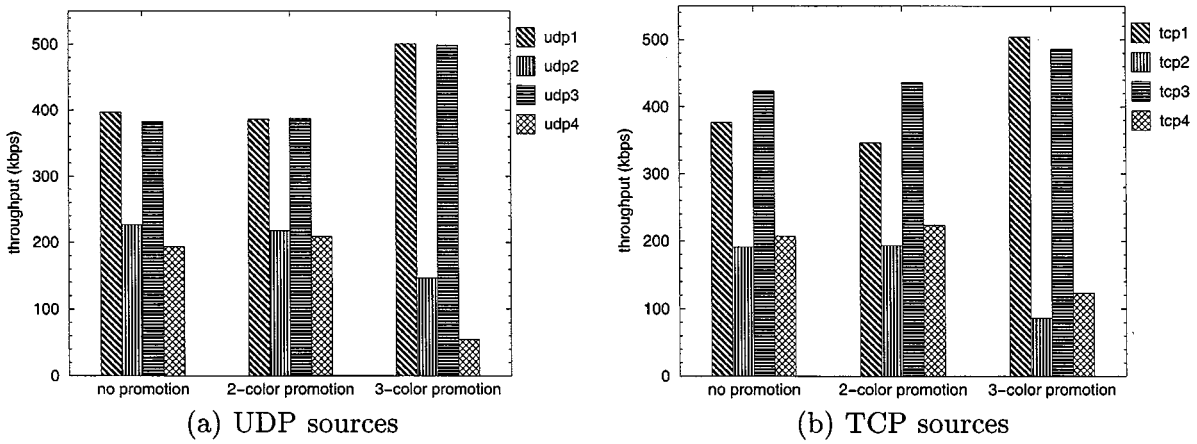

Fig. 10. The benefit from promotion.

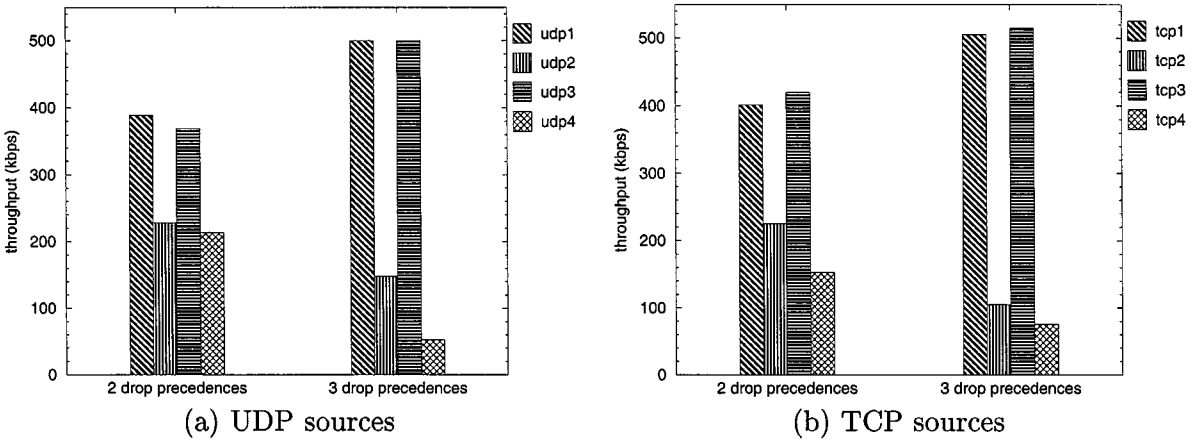

Fig. 11. Benefit from promotion: three drop precedences versus two drop precedences.

V. Supporting Three Drop Precedences to Improve Assurance

In the previous section, we analyzed the benefits of promotion in the REDP scheme. In the simulation, we chose $\alpha = 2$ Mb/s so that congestion does not happen in the first domain. What will happen if instead we choose $\alpha = 1.2$ Mb/s? Would we still get the same throughput assurance for flow1 and flow3? The answer is "no." Since in the core router, we only support two drop precedences, the *yellow* packets are treated the same as the *red* packets in the core router. If congestion happens in the first domain, some of the *yellow* packets will be dropped before they reach ER2. Promotion will not help to improve the throughput assurances of flow1 and flow3. The simulation result shown in Fig. 11 supports this answer. It is desirable to assign a lower drop probability to the *yellow* packets compared to the *red* packets, so that the *yellow* packets will be protected during network congestion. Thus, the core router needs to support three drop precedences. *Green* packets have the lowest drop probability and the *red* packets have the highest. The three drop precedences are supported through a similar way as RIO implementation except that we have three sets of RED parameters.

In our simulation, we choose the parameters for *red* packets as $\min_{red} = 20$, $\max_{red} = 35$, $P\max_{red} = 0.5$, the parameters for *yellow* packets as $\min_{yellow} = 35$, $\max_{yellow} = 45$, $P\max_{yellow} = 0.5$, and the parameters for *green* packets as $\min_{green} = 45$, $\max_{red} = 60$, $P\max_{red} = 0.1$. Fig. 11 shows the assurance gain by adding one more drop precedence in the core routers. For both UDP and TCP sources, if we set $\alpha = \beta = 1.2$ Mb/s, three drop precedences in the core router could greatly improve the throughput assurance of flow1 and flow3. Each of them gets a throughput of about 500 kb/s as shown in the figure.

VI. Parameter Sensitivity

For a leaky bucket marker, the only variable parameter is the size of the leaky bucket, b, which is also the maximum burst size. However, an REDP marker has additional parameters that determine the demotion and promotion processes. In this section, we briefly discuss how to select these parameters and their impact on the performance of REDP.

T_L and MAX_{demo} are two parameters which determine the fairness of the demotion process. If $T_L = 0$, the demotion is same as the demotion of a leaky bucket marker. In order to ensure enough randomness for the demotion process, T_L need to be large enough. However, increasing T_L may result in a larger b, which will increase the maximum burst size of the output traffic. So we should select an appropriate T_L so that we can have both good fairness and acceptable burst size.

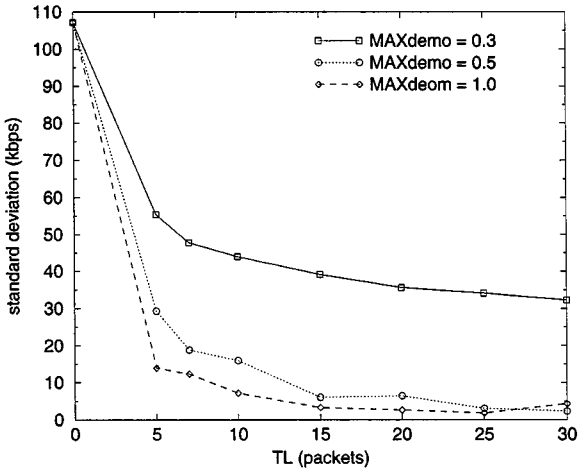

Fig. 12. Demotion fairness (4 UDP flows).

The range of MAX_{demo} is between 0 and 1. If $\mathrm{MAX}_{demo} = 0$, a *green* packet will not get demoted until the bucket runs out of tokens. So the demotion in REDP will become same as the demotion of a leaky bucket marker. Since T_L could not be set very large, selecting a large enough MAX_{demo} could improve the utilization of the *demotion region*, thereby improving the randomness of demotion.

A. Using Four UDP Flows

Fig. 12 shows the fairness of demotion under different (T_L, MAX_{demo}) selections, where the unit of T_L is in packets. We have used the topology shown in Fig. 5 for the simulation, but with a varying set of (T_L, MAX_{demo}). The overall demotion ratio is set to 40%. Ideally, each UDP flow should have 200 kb/s *green* packets demoted and 300 kb/s forwarded. In the simulation, we fix the bucket size $b = 60$ and change T_L from 0–30. Three MAX_{demo} values, 0.3, 0.5, and 1.0, are used in the simulation. Fig. 12 plots the standard deviation of the throughput of the four UDP flows for different values of MAX_{demo} with respect to T_L. The standard deviation of the throughputs defines the degree of fairness. The smaller the standard deviation, the fairer is the demotion. The following inferences could be derived from Fig. 12.

- For all three MAX_{demo} values, the fairness improves with the increase in T_L. However, when $T_L >= 15$, the fairness improvement becomes very slow as T_L increases. So $T_L = 15$ seems enough. This is also the value of T_L we have used in the former simultaions.
- For the same value of T_L, $\mathrm{Max}_{demo} = 0.5,\ 1.0$ have better fairness compared to $\mathrm{MAX}_{demo} = 0.3$, because a large enough MAX_{demo} value could improve the utilization of the *demotion region*.

B. Using More UDP and TCP Flows

So far, we have only used 4 TCP or UDP flows in all of the simulations to show the fairness of our REDP marker. In the following simulation we will use more flows to study the parameter sensitivity. It will also show that the REDP marker scales well with more flows. We use a simulation topology similar to the one shown in Fig. 5, except that now we have ten hosts, H1–H10, connected to the edge router ER. In our first simulation, we have 10 UDP sources starting from H1 to H10, respectively. All of the ten flows sink at CR2. The sending rate of the ten flows are 0.79, 0.73, 0.53, 0.61, 0.67, 1.0, 1.0, 1.0, 1.0, and 1.0 Mb/s, respectively. However, only 0.5 Mb/s of each flow is marked initially. So a total of 5 Mb/s marked packets enter ER. At ER, we can only mark 3 Mb/s packets, the remaining 2 Mb/s should be demoted. The link capacity between CR1 and CR2 is set to 3 Mb/s, so that all of the unmarked packets will be dropped. Ideally, each flow will get 300 kb/s throughput if the REDP marker is 100% fair. Fig. 13(a) shows the fairness under different T_L and MAX_{demo} values. Results in Fig. 13(a) we are very similar to those shown in Fig. 12. For $\mathrm{MAX}_{demo} = 0.5,\ 1.0$, $T_L = 15$ or 20 is enough. Further increase in T_L has negligible impact on the degree of fairness.

For the results shown in Fig. 13(b), we change the ten UDP flows to ten TCP flows. Similar to the simulation in Section IV-A-2, we let the roundtrip time of each flow be 26, 28, 26, 28 ms, ..., in order to show the phase effect of TCP flows. When $T_L = 0$, the standard deviation of the ten flows is about 63 kb/s. Increasing the value of T_L to 10 decreases the standard deviation to a value between 20 and 30. Further increasing T_L could only increase the fairness a little bit. It seems that it is more difficult to improve the fairness of TCP flows than that of the UDP flows. This is so because of the flow control and congestion control algorithm of TCP, as discussed earlier in Section IV-A-2. However, REDP marker does remove some of the phase effect of TCP flows. Another observation from Fig. 13(b) is that for the TCP flows the degree of fairness seems to be insensitive to the value of MAX_{demo}. A mixture of TCP and UDP flows may be even more complicated. According to the study in [19], mixing TCP and UDP traffic in the same AF class will have many problems and may not be fair to either. REDP won't be able to resolve this problem. Probably using two AF classes[5] to isolate UDP and TCP traffics may be a better idea.

C. Parameter Selection

Based on the observations from the above simulations, we suggest that MAX_{demo} and T_L should be set reasonably high. The process of promotion is symmetrical to that of demotion. So we could choose $b - T_H = T_L$, $\mathrm{MAX}_{promo} = \mathrm{MAX}_{demo}$.

$(T_H - T_L)$ determines the size of the *balanced region*, which also should be selected large enough. If the *balanced region* is too small, the leaky bucket may oscillate between the *demotion region* and the *promotion region*. This may cause unnecessary demotion and promotion. Again, large $(T_H - T_L)$ may increase the bucket size b, and will also delay the demotion and promotion processes. It is hard to configure a simple simulation to determine the best value for $(T_H - T_L)$. However, from our simulation experiences, we find $(T_H - T_L) = 10$ performs pretty good. These discussions may provide broad guidelines for parameters selection.

[5]IETF has defined four classes for AF.

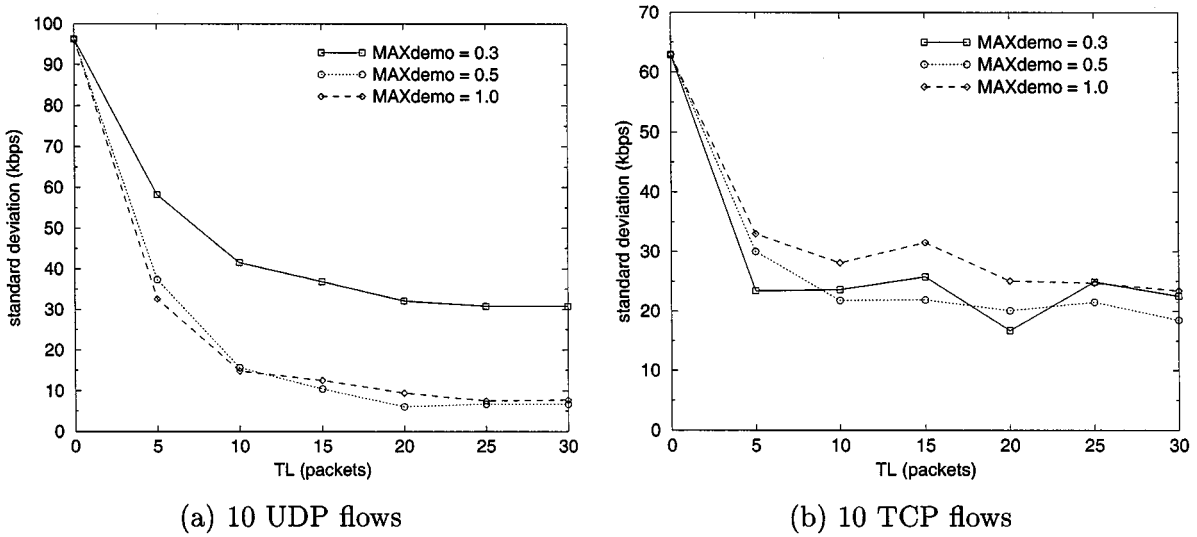

Fig. 13. Demotion fairness with more flows.

VII. Concluding Remarks

In the previously proposed studies of differentiated services in the Internet, the marking models in the edge routers demote packets when the available bandwidth is inadequate for the aggregate traffic flow. The demoted packets retain their new marking across all the domains they travel before reaching their destination. In this paper, a new approach for supporting differentiated services in the Internet is presented. A random early demotion and promotion (REDP) scheme is proposed that can be used for supporting differentiated services through an efficient marking process at the edge routers (the routers between the Internet domains). The primary features of the proposed REDP scheme are the provision of promotion of packets after getting demoted, and the fairness in the promotion and demotion processes. The promotion process is facilitated through a three-color marking process. The fairness is ensured through random and early decisions on the packets. We have simulated the REDP scheme using the $ns2$ simulator. Results indicate that the marker of the REDP scheme is very fair compared to a leaky bucket marker. The performance in terms of bandwidth allocation for assured traffic is significantly better than the leaky bucket and the previously proposed RIO scheme that uses a two-color marking scheme. The benefits of REDP can be greatly exploited during temporary and isolated over or under subscription in the Internet. We have also analyzed the benefits of promotion by using three drop precedences instead of two drop precedences, and have shown that the assured service gets better service guarantees with three drop precedences compared to the two drop precedences. All the results were obtained for both TCP and UDP traffics to demonstrate the wide applicability of the results and the REDP scheme. A parameter sensitivity study is also reported, results of which could be used as guidelines in determining the parameters for the REDP markers.

References

[1] P. Ferguson and G. Huston, *Quality of Service*. New York: Wiley, 1998.
[2] "IETF home page," http://www.ietf.org/.
[3] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation protocol (RSVP)—Version 1 functional specification," RFC 2205, Sept. 1997.
[4] Y. Bernet, J. Binder, S. Blake, M. Carlson, B. E. Carpenter, S. Keshav, E. Davies, B. Ohlman, and D. Berma, "A framework for differentiated services," Internet Draft, http://www.ietf.org/internet-drafts/draft-ietf-diff-serv-framework-02.txt, Feb. 1999.
[5] X. Xiao and L. M. Ni, "Internet QoS: The big picture," *IEEE Network Mag.*, pp. 8–18, Mar./Apr. 1999.
[6] F. Reichmeyer, L. Ong, A. Terzis, L. Zhang, and R. Yavatkar, "A two-tier resource management model for differentiated services networks," Internet Draft, Nov. 1998.
[7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," RFC 2475, Dec. 1998.
[8] V. Jacobson, K. Nichols, and K. Poduri, "An expedited forwarding PHB," Internet Draft, RFC 2598, <draftiet-f-diffserv-phb-ef-02.txt>, June 1999.
[9] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "Assured forwarding PHB group," Internet Draft, RFC 2597, <draft-ietf-diff-serv-af-0.6.txt>, June 1999.
[10] V. Jacobson, "Differentiated services architecture," in *Proc. Talk Int-Serv WG Munich IETF*, Aug. 1997.
[11] D. D. Clark, "Adding service discrimination to the internet," MIT Lab. Computer Science, Tech. Rep.
[12] D. D. Clark and W. Fang, "Explicit allocation of best effort packet delivery service," MIT Lab. Computer Science, Tech. Rep.
[13] "UCB/LBNL/VINT network simulator—ns(version 2)," http://www-mash.cs.berkeley.edu/ns/.
[14] J. Heinanen and R. Guerin, "A two rate three color marker," Internet Draft, <draft-heinanen-diffserv-trtcm-01.txt>, May 1999.
[15] H. Kim, "A fair marker," Internet Draft, <draft-kim-fairmarker-diffserv-00.txt>, Apr. 1999.
[16] S. Floyd and V. Jacobson, "On traffic phase effects in packet-switched gateways," *Internetworking: Research and Experience*, vol. 3, no. 3, pp. 115–156, Sept. 1992.
[17] ——, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, pp. 397–413, Aug. 1993.
[18] W. Feng, D. Kandlur, D. Saha, and K. Shin, "Understanding and improving TCP performance over networks with minimum rate guarantees," *IEEE/ACM Trans. Networking*, vol. 7, pp. 173–187, Apr. 1999.
[19] N. Seddigh, B. Nandy, and P. Pieda, "Study of TCP and UDP interactions for the AF PHB," Internet Draft, <draft-nsbnpp-diffserv-tcpudpaf-00.txt>, June 1999.

Fugui Wang received the B.E. degree in computer engineering from Tsinghua University, China, in 1994.

He is currently a graduate student in the Department of Computer Science and Engineering, Michigan State University. His research interests are in Internet, QoS, distributed systems.

Prasant Mohapatra (S'88–M'93–SM'98) received the B.S. degree from Regional Engineering College, Rourkela, India, in 1987, the M.S. degree from the University of Rhode Island in 1989, and the Ph.D. degree in computer engineering from the Pennsylvania State University in 1993.

Currently, he is an Associate Professor in the Department of Computer Science and Engineering at Michigan State University. Previously he was with the Department of Electrical and Computer Engineering, Iowa State University. During the summers of 1998 and 1999, he worked as a Visiting Scientist for Panasonic Technologies and Intel Corporation, respectively. His research interests include storage and server architecture, computer networks, distributed systems, Internet technology, and performance evaluation. In these areas he has published several papers in international journals and conferences. His research has been funded by the National Science Foundation, Intel Corporation, Panasonic Technologies, Rockwell International, and EMC Corporation.

Dr. Mohapatra is an Associate Editor of the IEEE TRANSACTIONS ON COMPUTERS, and has served on the program committees of several international conferences and NSF panels.

Sarit Mukherjee received the B.Tech. degree in computer science and engineering from the Indian Institute of Technology, Kharagpur, India, in 1987, and the M.S. and Ph.D. degrees in computer science from the University of Maryland, College Park, in 1990 and 1993, respectively.

He was an Assistant Professor of Computer Science and Engineering at the University of Nebraska-Lincoln from 1993 to 1997. Currently he is a lead scientist with Panasonic Information and Networking Technology Lab, Princeton, NJ, where he is leading the video networking group. His research interests include high speed network architectures and protocols, multimedia applications, network security and auto configuration.

Dr. Mukherjee is a member of ACM SIGCOMM.

Dennis Bushmitch received the B.S degree in electrical engineering from the New York Institute of Technology and the M.S. degree in electrical engineering from Polytechnic University, Brooklyn, NY, in 1994 and 1997, respectively.

He is presently a Ph.D. candidate at Polytechnic University. He is a Scientist with Panasonic Information and Networking Technology Lab, Princeton, NJ. He is also an Adjunct Lecturer at the New York Institute of Technology. His research interests include network traffic engineering, broadcast digital video networking, and multimedia applications. He holds two patents.

Mr. Bushmitch is a member of several honor societies.