Two-differentiated Marking Strategies for TCP flows in a Differentiated Services Network

Sung-Hyuck Lee, Seung-Joon Seok, Seung-Jin Lee, Sung-Kwan Youm
and Chul-Hee Kang*

Department of Electronics Engineering, Korea University
5-Ka, Anam-dong Sungbuk-ku, 136-701 Seoul, Korea
{starsu, ssj, linuz, skyoum}@widecomm.korea.ac.kr, *chkang@korea.ac.kr

Abstract. The saw-tooth like behaviors of TCP impact Assured Forwarding Service flows in a differentiated services (diffserv) network. Therefore, we argue the use of TCP-friendly building blocks (or modules) and fairness modules in the diffserv architecture regarding this issue, and propose *Two Markers System (TMS)* that is able to properly mark packets and fairly share the bandwidth to each flow for their targeted sending rates. TMS has two marking modules that are placed on the source and the edge of a diffserv network. For sources of the network*, the virtual source making modules* play important roles of reducing TCP impacts in the assured services and suitable marking packets. Next, in the edge of the network, *the edge embedded marking module* conducts new fairness policy based on the marking rate of flows from sources, so called *"marking rate-based fairness"*. Finally, we present simulation results to illustrate the effectiveness of TMS scheme over several parameters. That is, *Two Markers System* reduces TCP impacts over assured service and fairly shares the bottleneck link bandwidth of a network.

1 Introduction

The diffserv approach is based on a set of simple mechanisms that treat packets differently according to the marking of the DS field in the IP header. Before entering in a DS domain, the field is marked with a certain value (or codepoint) that determines the treatment that should be supplied to the packet inside the domain. However, because of the limited amount of bits available for use in the DS field, the IETF's Diffserv Working Group has defined a small set of building blocks, called per-hop behaviors (PHBs) which are used by routers to deliver a number of services. Among the initial PHBs being standardized are the Expedited Forwarding (EF) and the Assured Forwarding (AF) PHBs. The EF PHB specifies a forwarding behavior in which packets see a very small amount of loss and a very low queuing delay. In order to ensure every packet marked with EF receives this service, EF requires every router to allocate enough forwarding resources so that the rate of incoming EF packets is always less than or equal to the rate which the router can forward them. The AF PHB group, on the other hand, specifies a forwarding behavior in which packets see a very small amount of loss, and consists of four, independently forwarded classes that have two or three drop preference levels. The idea behind AF is to preferentially drop best-effort packets and packets which are outside of their contract when congestion occurs.

In this paper, we consider a form of a better-than-best-effort service called the "Assured Service". The Assured Service follows expected capacity profiles that are statistically provisioned. Packets are treated preferentially according to the dropping probability applied to the best-effort queue. The assurance of service comes from the expectation that the traffic is unlikely to be dropped as long as it stays within the negotiated capacity profile. The building blocks of this service include a traffic marker at the edge of the domain, and a differentiated dropping algorithm in the core of the network. A packet of a flow is marked IN (in profile) if the temporal sending rate of the arrival time of the packet is within the contract profile of the flow. Otherwise, the packets are marked OUT (out-of-profile). The temporal sending rate of a flow is measured using TSM (Time Sliding Window) or token bucket control module. A differentiated dropping algorithm such as RIO (Random Early Detection with IN/OUT) is provided in the core routers of the network. In particular, the OUT packets are preferentially dropped upon evidence of congestion at the bottleneck before the IN packets. After dropping all incoming OUT packets, IN packets are discarded. With this dropping policy, the RIO network gives preference to IN packets and provides different levels of service to users based on their service contracts.

In [11], authors presented that the use of a simple token bucket marker for the above assured service results in TCP realizing the minimum assured rate. The authors attributed the cause of such behavior to TCP's complex response primarily to packet losses. TCP reacts to congestion by halving the congestion window (cwnd) and increases the window additively when packets are delivered successfully. Exponential decrease (halving the congestion window) is required to avoid congestion collapse and TCP treats a packet drop as an indication congestion[7]. However, in the differv network these additive-increase and multiplicative-decrease make it hard to protect the reservation rate. When TCP reacts to an OUT packet drop by halving its congestion window and increases additively, it may not reach its reservation rate.

In [1], [2], Feng et al. proposed adaptive priority marking schemes that focused on TCP-friendly mechanism in RIO scheme. The authors presented two different Marking mechanisms: source-integrated that the control engine, so called, packet marking engine (PME) is integrated with the host, and source transparent marking that PME is potentially external to the user. They showed results that the proposed scheme decreased TCP impacts over the Assured Service. They, however, didn't consider marking scheme for aggregated flows.

In [11], through the medium of simulation work, Ibanez and Nichols showed that TCP didn't realize the minimum assured rate over the Assured Service. The overall performance was affected by the bursty losses being experience by the OUT packets. They concluded that it was ambiguous how the Assured Service could be characterized quantitatively for TCP application, and an Assured Service couldn't provide consistent rate guarantees.

In [8], Ikjun and Narasimha proposed and evaluated several schemes to improve the realization of throughput guarantees in the Internet that employed a differentiated services architecture. They showed that excess bandwidth of a network was not distributed proportional to target rates, and the impact of differences in RTTs could be reduced by the aggregation of sources and fair sharing of bandwidth at the edge routers.

In this paper, we take the problem into consideration between the transport control protocol (TCP) and the differentiated drop policies of the network in realizing the reserved throughputs, and propose newly modified scheme called, Two Markers System for improving the realization of target rates in a differentiated services network. In addition, the issue of fairness among users is important in this diffserv context since we expect that the differentiation in service will be based on some kind of pricing mechanism, thus it is necessary that the resource allocation be commensurate with how the service is priced. In this paper, we also argue how fairly the proposed scheme allocates bandwidth.

The rest of this paper is organized as follows: Section 2 proposes Two Markers System to reduce TCP influences over assured services and consider fairness for a bottleneck link of a differentiated services network, and then, explores the proposed algorithms. Section 3 presents results using the above algorithms in simulated environments for responsive traffic flows such as TCP, and performs analysis for simulated results. Section 4 concludes our work

2 Two Markers System

In this section, we focus on several strategies used to mark packets in order to consider TCP dynamics and adapt fairness for sharing a bottleneck link of a network, and propose a modified marking scheme, so called, Two Markers System (TMS); the first marker (TMS_I) is located at sources of a network to adapt TCP congestion control algorithm, and the second marker (TMS_II) at edge to fairly mark the aggregated flow.

Marking strategies can be classified into three categories based on the criteria used for the marking. Module of devices can mark packets: (i) based on the state of all individual flows of an aggregation, called *flow marking*, (ii) based only on aggregation state, without any knowledge about individual flows, called *aggregation marking*, and (iii) based on a certain knowledge of individual flows, called *flow aware aggregation marking*. We will take newly improved marking strategy into consideration to complement the issue of the first and the second marking strategy.

2.1 Concepts

What's the Two Markers System (TMS)? This system has two marking modules which are located in the source and at the edge of differentiated services network, and each marking module plays different roles to achieve the reservation rate and the target rate of Assured Services. Figure 1 illustrates the "Two Markers System" framework.

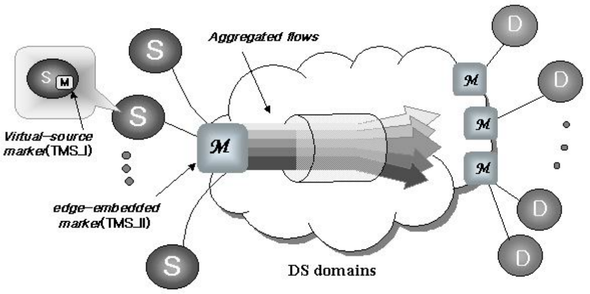


Fig. 1. The "Two Markers System" framework

First, a virtual source-marking module (TMS_I) carries out two main roles. One is to control a flow and congestion, and the other is to give the marking probabilities to the edge-marking module (TMS_II). Therefore, TMS_I decreases TCP impacts in the underlying AF services, and helps the edge-marking module to properly mark packets. So to speak, TMS_I can be not a marker used to mark packets in order to classify the service in the core of a network, but an indicator, also called *virtual source marker* that notifies TMS_II in the edge of a network of the marking rate. Second, the edge embedded marking module (TMS_II) monitors incoming traffic flows from users at the edge of the network against the profile that the users have agreed with the service provider. TMS_II measures the number of the marked packets (or marking information) from sources and re-marks aggregated flows for the profile that the users have agreed with the service provider. In this case, we elaborate a fairness strategy for sharing excess bandwidth of a bottleneck link. In [2], it showed that a source-integrated packet marking engine (PME) properly kept up marking rate rather than a source-transparent PME, because the measurement of throughputs against the reservation rate at the source is accomplished more exactly than at the edge of a network. We also assume that TMS_I properly marks packets at sources, and each host pours the packets into the edge of the network.

Therefore, a fiducial point of fairness strategy is the marking rate (or marking information) of traffic flows from users. TMS_II re-marks (or marks) aggregated flows for total target rate of all the flows.

Note that in this case, sources' target rates are different from the edge's target rate. Let R_i denote the reservation rate of a flow i and C represent the capacity of a bottleneck link in the network. The target rate of each source is a reservation rate (R_i) of which user informs the edge, in order to be assured of bandwidth for delivering data. On the other hand, a flow's target rate (T_i) in the edge of a network is the reservation rate (R_i) plus variable value that is the marking rate to the excess bandwidth (E). The excess bandwidth at this bottleneck link is then given

$$E = C - R_i = C - \sum_{i=1}^{n} R_i \quad . \tag{1}$$

Therefore, a flow's target rate in the edge of the network is given by

$$T_i = R_i + X_m \times E = R_i + X_{mi}(C - \sum R_i) \quad . \tag{2}$$

Where, X_{mi} is proportional to the probability of marking from a source. That is, it is determined depending upon fraction between the marking rate of each flow and the average marking rate (4). Development for X_{mi} is as follows: First, the probability of marking for each flow i is the number of the marked packets (N_{mi}) by throughput Thr_i, or In-profile window by congestion window.

$$m_i = \frac{N_{mi}}{Thr_i} = \frac{iwnd}{cwnd} \quad . \tag{3}$$

where, *iwnd* and *cwnd* represent window of the marked packets and congestion window, respectively. Next, the average marking rate $E[m_i]$ of all the flows at the edge of a network is

$$E[m_i] = \frac{1}{n}\sum_{i=1}^{n} m_i$$

$$= \frac{1}{n}\sum_{i}^{n} (\frac{iwnd}{cwnd})_i \quad . \tag{4}$$

where, $Mark_p$ also represents the probability of marking in TMS_I algorithm of the. Finally, X_{mi} is developed by using the fraction between (3) and (4), that is, X_{mi} represents a relative value against the datum point, $E[m_i]$. The fraction of X_{mi} varies from 0 to 1 depending upon m_i and $E[m_i]$. For simplicity, we assume that X_{mi} has uniform distribution, so coefficient 1/2 in (5) represents average value of uniform distribution.

$$X_{mi} = \frac{m_i}{2E[m_i]} = \frac{nm_i}{2\sum_i^n m_i} \quad . \qquad \textbf{(5)}$$

2.2 Suitable-Marking strategy: Two Markers System-I algorithm

In this subsection, we focus on the marking algorithm, so called the suitable-Marking strategy that is implemented in the virtual source marker (TMS_I). Congestion window (cwnd) consists of two windows in this algorithm[8]: *In-profile window (iwnd)* and *Out-of-profile window* (ownd). *In-profile window* represents the number of marked packets per-RTT needed to achieve the reservation rate that is guaranteed by a contract with Bandwidth Broker. O*ut-of-profile window* signifies the number of unmarked packets per-RTT that is outstanding. Each window has its respective threshold (issthresh and ossthresh).

Program code 1 and 2 illustrate the modified TCP congestion control algorithm. Some of the earlier work [2], [8] has focused on similar issues in networks where marking policies are different from the one studied here. Our study extends the earlier work and proposes new approaches to improving the realization of bandwidth guarantees. In the starting point of this algorithm, how can we determine the probability of marking ($Mark_p$)? It is generally determined depending upon fraction between the measured and target throughputs. The window size increases or decreases according to throughput against the reservation rate. When either window is above its threshold, it increases linearly.

In Program code 2, whenever OUT packets loss occurs, O*ut-of-profile window (*ownd) reduces the half of its window size. On the other hand, the loss of IN packets indicates that congestion window reduce the half of its window size. A virtual source-marking module (TMS_I) properly keeps up marking rate rather than an edge-embedded marking module. While the established edge marker, for example, is fairly effective in maintaining the observed throughput close to the reservation rate, it often marks more packets than required. A sender recognizes packets loss, and determines whether the lost packet was sent as a marked or an unmarked packet. The loss of IN-marked packets represents an acute congestion situation in the network, and congestion window reduces the half of its window size. That is, if the loss of IN-marked packets occurs frequently, the congestion window size will be maintained low. A virtual source marking module, however, will decrease TCP impacts over AF services, because it properly regulates marking rate according to the condition of a network. Thus, since the probability that IN-marked packets are excessively dropped at the core of the network is reduced, congestion window will not be often decreased unlike the established edge marker.

The goals of the TMS_I provide modified congestion control that reduces TCP dynamics over AF services and a new datum point for fairly sharing the bottleneck bandwidth. We deal with usage of the datum point of the edge-embedded marker in subsection 3.3. Therefore, the strategy providing the datum point alleviates somewhat of the problem of introducing unfairness within aggregated flows on *aggregation marking*.

Program code 1: TMS_I algorithm (cwnd open)

```
When new data is acknowledged
  iwnd = $Mark_p$ * cwnd;
  ownd = cwnd - iwnd;
    if(Thr < Rr)
      if(iwnd < issthresh)
          iwnd += $Mark_p$;
      else
          iwnd += 1/2cwnd;
      if(ownd < ossthresh)
          ownd += (1- $Mark_P$);
      else
          ownd += 1/2cwnd;
    else
      if( 0 < iwnd)
          if(iwnd < issthresh)
              iwnd -= 1-$Mark_p$;
          else
              iwnd -= $Mark_p$;
      else            /* ownd = cwnd */
```

```
            if(ownd < ossthresh)
                ownd += 1;
            else
                ownd += 1/cwnd;
        cwnd = iwnd +ownd;
        $Mark_p$ = iwnd/cwnd;
```

Program code 2: TMS_I algorithm(cwnd close)

```
When the third duplicate ACK in a row is
  iwnd = $Mark_p$ * cwnd;
  ownd = cwnd - iwnd;
    if(out packet loss)
        ossthresh = ewnd/2;
        ownd = ossthresh;
    else
        cwnd = cwnd/2;
   issthresh = $Mark_p$ * cwnd;
   ossthresh = cwnd - issthresh;
```

2.3 Marking rate-based Fairness strategy: Two Markers System-II algorithm

The edge-embedded marker(TMS_II) elaborates a new marking strategy based on marking rates from sources in order to fairly share bottleneck bandwidth. In this case, the edge-embedded marker closely cooperates with sources to accomplish the fairness strategy, called “marking rate-based fairness”. Notice that fairness is defined as how well the throughput of all individual flows of aggregated traffic realizes their target rates(Ti).

We assume that an edge router receives information of the reservation rate that each user wants to be assured of, and it contracts with the service provider(or bandwidth broker) for bottleneck link bandwidth of the network. In this mechanism, the edge router sets the target rate different from the source: Recall the numerical expression (2) to mind. As stated in subsection 3.1, the target rate of each flow at the edge of a network includes excess bandwidth in proportion to the probability of marking from the sources, because the sources suitably mark packets against their target rates, that is reservation rates. Thus, the magnitude of marking rate represents increase or decrease in demand for bandwidth. If the number of marked packets, for example, exceeds the threshold value, that is the average marking rate, $E[m_i]$, the edge-embedded marker considers that the flow wants more bandwidth than others in order to achieve its reservation rate. Therefore, the flow is marked more and has a higher target rate than others.

Program code 3 illustrates the TMS_II algorithm that considers the “marking rate-based fairness” policy and a congestion situation.

Program code 3: TMS_II algorithm

```
if($Th_t$ < C)
     For i -> i=1 to i =n
     Ti = $R_i$+($X_{mi}$ (C- $R_t$));
        if($Thr\_{}_i$ < $T_i$)
            $M_i$ += α|$T_i$ - $Thr\_{}_i$|;
        else
            $M_i$ -= α|$T_i$ - $Thr\_{}_i$|;
  else if($Th_t$ == C)
       exit;
  else
       For i -> i = 1 to i = n
       $T_i$ = $R_i$
          if($Thr\_{}_i$ < $T_i$)
              $M_i$ -= β|$T_i$ - $Thr\_{}_i$|;
          else
              $M_i$ -= γ|$T_i$ - $Thr\_{}_i$|;
/*   $M_i$: marking rate of ith flow
       $T_i$: target rate of ith flow,
       $R_i$: reservation rate of ith flow,
       Excess bandwidth = C – $R_{tr}$
       $X_m$ = (0.5 * $Mark_p$)/E[$m_i$]
       α≤β≤γ, α, β, and γ is experimental values   */
```

First, TMS_II observes the total throughput of aggregated flows and compares it with the bottleneck link bandwidth(C) of the network. This comparison has two objectives: One is to consider a congestion situation, the other is to ensure the coherence of the suitable-marking strategy. The first and seventh lines in program code3 show that the edge-embedded marker fairly allocates the excess bandwidth of the bottleneck link. The eighth and ninth lines represent that the edge-marking rate is equal to the marking probability of the sources(or marking information), and marking mechanism for a congestion situation appears from the tenth to the sixteenth lines. Then, if the total throughput doesn’t reach the capacity C, it again measures whether the throughput of a flow i realizes the target rate (T_i), the crux of the fairness strategy, or not. Note that if the total throughput is more than the capacity C, the target rate of each flow becomes its reservation rate. Now, changes of the marking rates for each flow follow: According to a network situation, TMS_II applies different scaling to marking rate. That is to say, it uses α when there is enough bandwidth to achieve the contract rate at bottleneck link of a network, and β or γ when congestion occurs. We consider that β is bigger than α and γ is bigger than β. The idea behind difference is to quickly reduce loss of the IN-marked packets in oversubscribed state. When there is an acute congestion at the bottleneck link of a network and loss of the IN-marked packets occurs, the assured service isn't guaranteed due to TCP dynamics. Therefore, the differentiated scaling mechanism can have important meaning in oversubscribed network.

In short, sources inform an edge-embedded marker of suitable marking rate, then edge allocates fairly bandwidth to flows based on the marking rate of each source. Therefore, we can say that marking rate-based fairness strategy alleviates the problem of leading to an inefficient utilization of the reserved bandwidth on flow marking.

3 Simulation and analysis

In this section, we present the simulation results with two marking algorithms we have described in the previous section. The simulation was done with the network simulator-2(ns-2). For the sake of simulation, we used a network with the configuration shown in Figure 2.

In the simulation, we have 10 sources (1 through 10 counting downwards) that communicate with one of ten different destinations. We conducted two scenarios: oversubscribed and non-oversubscribed network. In the first scenario, the aggregate reservation rate is 30Mbps, and the bottleneck capacity is set to 40Mbps so that the bottleneck is not oversubscribed. In the second scenario, the aggregate reservation rate is 50Mps, and the bottleneck capacity is also set to 40Mbps so that the bottleneck link experiences congestion. We assume that the RTT without queuing delay of each flow is randomly pocked from 80 to 220 ms. The sources are all TCP-Reno sources(unless specified otherwise). For the RIO implementation, the routers use RED with the values of 200 packets, 400packets, and 0.02 for *min_in, max_in*, and P_{max_in} and 50 packets, 100 packets and 0.5 for *min_out, max_out*, and P_{max_out}.

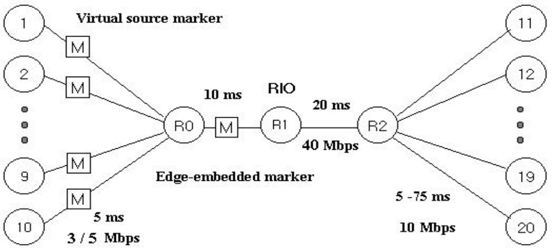
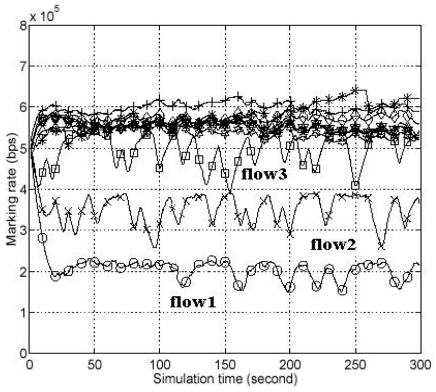


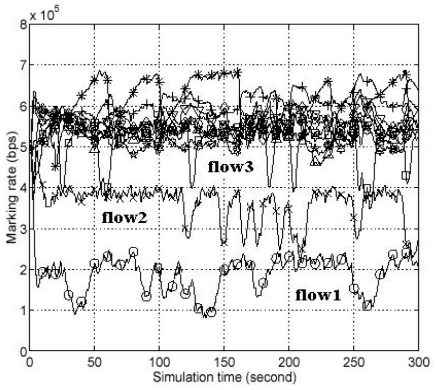
Fig. 2. Simulation topology

First of all, we investigate the simulation results with suitable-marking algorithm as the role of the virtual source marker. Since TCP windowing restricts the aggregated flows from competing for the excess bandwidth, the established edge marker continuously overmarks its packets as shown in Figure 3(a). It shows somewhat constant marking curves. However, since TMS_II determines marking rate based on the suitable-marking algorithm of sources, Figure 3(b) shows that the number of marked packets to accomplish the target rate is fewer than in the case of Figure 3(a). In the graphs below, flow 1 and flow2 demonstrate a striking contrast between *excess marking* and *suitable-marking*. In runtime from 100(s) to 200(s), while flow 1 and flow 2 in the Figure 3(b) mark each packet of a flow complying with the changing condition of the network, those in the Figure 3 (b) do not so.

Second, we study the proper values of scaling factors α, β, and γ through simulation, respectively. In Figure 4 (a), M_i is adjusted in steps such as $\alpha = 0.01$, $\beta = 0.05$, and $\gamma = 0.10$. As the figure shows, A2 is slow in reacting to changes in the network, and the marking rate lags behind the changes in the network load, slowly rising in response to an increased traffic load and slowly falling in response to a decreased traffic load. To investigate the other side, the simulation was repeated while allowing Mi to be updated in scaling factors of $\alpha = 1.0$, $\beta = 1.0$, and $\gamma = 1.5$. That is, when more bandwidth is needed to achieve the target rate, packets are marked more. Otherwise, packet marking is quickly turned off. Figure 4(b) shows the results from this simulation. As expected, M_i adapts very quickly to the changes in the network load, thus allowing the flow to achieve its target rate during periods of increased traffic load. This rapid response also allows the edge-embedded marker to turn off packet marking quickly when it detects that the available bandwidth is sufficient to satisfy the target rate.



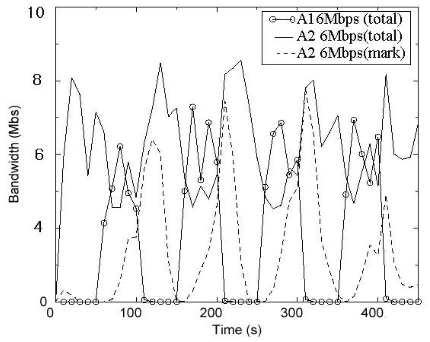
a) The established marker



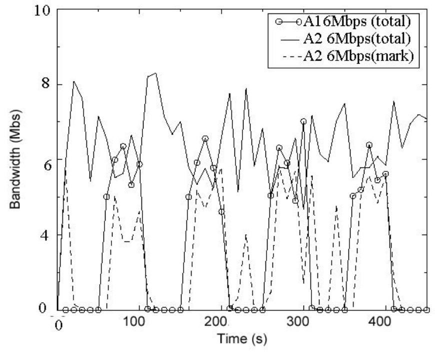
b) Proposed TMS marker

Fig. 3. Suitable-marking algorithm: When the established marker is applied, graph of a) represents somewhat constant marking rate. However, when suitable marking algorithm is applied, graph of b) represents variable marking rate according to a certain situation of the network. For example, flow 1 in graph of b) properly marks packets in comparison with graph of a)

Finally, we present the simulation results with marking rate-based fairness algorithm described in the previous section. We set that reservation rate of each flow is 5Mbps, and compare two marking schemes: One is the aggregation marking, the other is marking strategy of TMS. As stated above, the target rate of a flow i in TMS varies in proportion to the probability of marking from the sources. Figure 5 represents the results that the throughputs of all individual flows of aggregated traffic realize their target rates. Each flow in Figure 5 (a) often fails to achieve their target rates, because it receives bandwidth with no information about its state. That is to say, aggregation marking marks packets based on only aggregation state, without any knowledge about individual flows. However, in Figure 5 (b), each flow satisfies its reservation rate and shares the excess bandwidth of the bottleneck link according to the probability of his marking if the total throughput is less than the capacity of a network.



a) α = 0.01, β =0.05, and γ =0.10



b) α = 1.0, β =1.0, and γ = 1.5

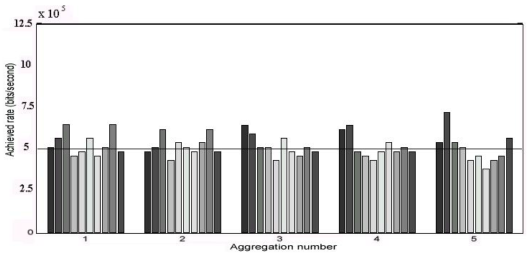
Fig. 4. Marking Packet through scaling factor

Two Markers System shows the improvement of performance about several parameters we used to simulate. However, an implementation of virtual source marking modules at source causes an issue of scalability in a differentiated services network through modifying TCP source, in addition, α, β, and γ should be set at their optimal values for the burstiness problems occurred owing to coarse scaling value.

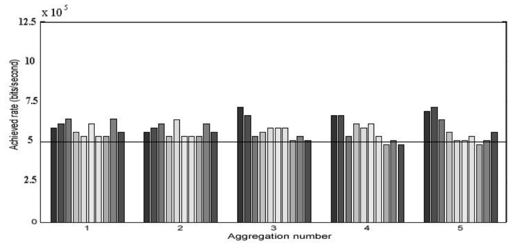
4 Conclusion

The strongest point of the Two Markers System proposed in this paper is that two marking modules are able to properly mark packets and fairly share the bandwidth for the target rates: suitable-marking and marking rate-based fairness strategies(markers). First, TMS_I using the suitable-marking algorithm properly adjusts the probability of marking(or the information of marking) according to the changes of the network, and notifies TMS_II, called the edge-embedded marker, of the marking rates that will be used to mark aggregated flows at the edge of the diffserv network. Next, TMS_II at the edge of the network marks packets based on the marking rates of sources, and allocates flows the bottleneck bandwidth in view of proportional fairness, called *marking rate-based fairness* modified *flow aware aggregation marking* strategy. We have simulated a TMS model to study the effects of several factors on the throughput rates of TCP flows in a RIO-based Differentiated Services network. Despite the scalability and burstiness issues, the simulation results, as expected, showed that Two Markers System reduced TCP impacts over assured service and fairly shared the bottleneck bandwidth.

In the near future, we will find the optimal values of scaling factor, α, β, and γ, in order to alleviate the burstiness problems of coarse scaling values, and study efficient scheme that informs edge-embedded marker of the information of marking.



a) Aggregation marking strategy



b) Marking rate-based fairness strategy

Fig. 5. Throughput for marking rate-based fairness

Acknowledgement

This research was supported in part by Information and Telecommunication Research Center(ITRC) contracts. We are deeply grateful to Asian Pacific Advanced Network-Korea Consortium(APAN-KR) members for technical discussion on Differentiated Services related issues.

References

1. W. Feng, D. Kandlur, D. Saha, andt K. Shin: Adaptive Packet Marking for Providing Differentiated Services in the Internet, In Proc. ICNP'98, Austin, TX, Oct. (1998) 108-117
2. W. Feng, D. Kandlur, S. Saha, and K. Shin: Understanding TCP Dynamics in an Integrated Services Internet, in Proc. NOSSDAV'97, St. Louis, MO, May (1997). 279-290
3. S. Floyd and V. Jacobson, Like-sharing and Resource Management Models for Packet Networks, IEEE/ACM Transactions on Networking, vol. 3 no. 4, August (1995) 365-386
4. J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski: Assured Forwarding PHB Group, RFC2597, June (1999)
5. K. Nichols, S. Blake, F. Baker, and D. Black: Definition of the Differentiated Service Field (DS Field) in the Ipv4 and Ipv6 Headers, RFC2474, Dec (1998)
6. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss: An Architecture for Differentiated Services, RFC2475, Dec. (1998)
7. V. Jacobson and M. Karels: Congestion Avoidance and Control, in Proc. SIGCOMM'88, Stanford, CA, August (1998) 314-329
8. I. Yeom and A. L. N. Reddy: Realizing Throughput Guarantees in a Differentiated Service Network, in Proc. ICMCS'99, Florence, Italy, June (1999) 372-376
9. N. Seddigh, B. Nandy, P. Pieda: Bandwidth Assurance Issues for TCP Flows in a Differentiated Services Network, in Proc. GLOBECOM'99, Rio De Janeiro, Dec. (1999)
10. C. Gbaguidi, H. J. Einsiedler, P. Hurley, W. Almesberger, and J. Hubaux: A Survey of Differentiated Services Proposals for the Internet, SSC Tech. Report SSC/1998/020, http://sscwww.epfl.ch, May (1998)
11. J. Ibanez, K. Nichols: Preliminary Simulation Evaluation of an Assured Service, IETF Internet Draft, draft-ibanez-diffserv-assured-eval-00.txt, August (1998)