Research Journal of

Information Technology

ISSN 1815-7432

Academic Journals Inc.

www.academicjournals.com

Research Journal of Information Technology 4 (1): 1-11, 2012
ISSN 1815-7432 / DOI: 10.3923/rjit.2012.1.11

A Modified Packet Marking Algorithm to Improve Bandwidth Fairness in Diffserv Networks

[1]A.H. Altalhi, [2]M.S.B.M. Azmi, [3]A.M. Al-Kharasani and [3]S.A. Ali
[1]Department of Information Technology, College of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
[2]Department of Software Engineering, College of Information Technology, University Tenaga Nasional, 43100, Selangor, Malaysia
[3]Department of Networking, Faculty of Science Computer and Information Technology, University Putra Malaysia, 43400, Selangor, Malaysia

Corresponding Author: A.H. Altalhi, Department of Information Technology, College of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia Tel: +966-2-695-1317 Fax: +966-2-640-0000

ABSTRACT

DiffServ routers provide Per Hop Behaviors to aggregate traffic for different services levels. An unfairness problem occurs in DiffServ networks, however: fairness deteriorate as the number of aggregates increases in the network. Researchers have thus, created various algorithms to overcome fairness deterioration. This article studies a packet marker based on a Time Sliding Window (TSW) and proposes a new hybrid marker algorithm to improve bandwidth fairness on DiffServ networks by providing a proportional fair share of excess network bandwidth among TCP aggregates. The simulation experiments have compared the new hybrid marker algorithm to various algorithms using NS-2. The results show that the proposed hybrid marker is sensitive not only to the number of flows in the aggregate but also to the number of aggregates in the network.

Key words: Quality of service, diffServ networks, traffic maker, tswTCM, fairness

INTRODUCTION

On IP-based networks the IETF has defined two models for Quality of Service (QoS): the Integrated Service (IntServ) (Braden *et al*., 1994) and the Differentiated services (DiffServ) (Blake *et al*., 1998). Moreover, these models take in numerous schemes that provide special traffic handling.

The earliest attempt to revise the internet form of the best effort towards a QoS acquiescent network is the IntServ service. Furthermore, the scalability issue in the IntServ model puts forward a major concern of the model, especially in the core network (Carpenter and Nichols, 2002). The IETF thus provides an efficient and scalable edge-to-edge QoS within a single domain more accurately by introducing the DiffServ model, which proposes avoiding this issue by providing guarantees to the aggregates more than to individual flows (Sani and Othman, 2011; Su and Atiquzzaman, 2003).

The DiffServ services propose the concept based on effortless functionalities, where the core network is maintained flow moves to the edge of the network (Nichols *et al*., 1998). Nevertheless, the DiffServ service supports QoS in the network by providing lower flexibility and fairness among the aggregate that share the network and their flows share. Moreover, in DiffServ network nodes, Per Hob Behaviors (PHBs) are implemented using queuing and scheduling schemes and head through useful fundamental elements including markers and shapers. Though these elements were useful in queuing and scheduling performance improvement, they caused an un-fairness problem in the DiffServ. Moreover, the DiffServ distinguishes traffic by aggregating individual flows that require special handling based on the Service Level Agreement (SLA) between the client and service supplier, according to a pre-defined forwarding module. A DiffServ Code Point (DSCP) is then allocated to each packet in a behavior aggregate and the flows with the same DSCP belong to one PHB (Blake *et al.*, 1998). Furthermore, the major DiffServ components are traffic conditioning and PHB. The former component is only done at the edge of the network, designed to classify the traffic in various terms, including coloring, policing and shaping. The latter PHB component principally contains a variety of mechanisms, including scheduling, queuing and dropping schemes.

Also, the DiffServ split the PHB component schemes into different categories (Heinanen and Guerin, 1999a; Heusse *et al.*, 2003; Heinanen *et al.*, 1999; Sani and Othman, 2011) such as, Best Effort (BE), Expedited Forwarding (EF) and Assured Forwarding (AF). The classic category is the Best Effort (BE), which does not ensure a high QoS level and therefore guarantees any QoS level. Another category, Expedited Forwarding (EF), ensures a moderate QoS level by maintaining a small delay in traffic flows. Conversely, the last category, Assured Forwarding (AF), has four classes and three levels of drop priority in each class ensure a lowest throughput level. Additionally, packets use three different colors to mark traffic, which describe traffic priorities using green, yellow and red. The highest received packet priority is marked as green, the lowest priority is red and the higher dropping priority packets are yellow (Chrysostomou *et al.*, 2009; Heinanen and Guerin, 1999b; Heinanen *et al.*, 1999). Moreover, two main parts of the DiffServ network have separated: the edge and core routers. The former part, the edge router, is responsible for traffic classification and conditioning from diverse sources and injects the treated traffic into the core network. The latter part, the core router, is responsible for a traffic forwarding mechanism, which implements active queue management schemes, such as RED based on in/out (RIO) and First in/out (FIO) (Chrysostomou *et al.*, 2009; Clark and Fang, 2002).

The bandwidth provisioning Under-Provisioned Network (UPN) condition is the process by which a backbone Internet services provider verifies the amount of bandwidth needed by each links toward a superior performance level required (Sani and Othman, 2011). Furthermore, whenever the total of all target rates is larger than or equal to the available bandwidth of a bottleneck link, the UPN condition arises.

The scheduling and queuing schemes used in routers become the role of the PHB. Moreover, the excess core bandwidth in a DiffServ network is shared between flows instead of aggregates in the existing time-sliding window Three Color Marker (tswTCM) scheme. The unfairness issue also arises when a small service client injects more yellow packets than a large service client. Conversely, marking yellow and red packets without taking the aggregate service profile in which the probability concern of marking yellow packets is proportional to the service profile and achieves a proportional fair share of excess bandwidth (Fang *et al.*, 2000; Ibanez and Nichols, 1998; Su and Atiquzzaman, 2001, 2003) must be considered together to solve the unfairness issue and enhance performance.

This article proposes a new hybrid marker algorithm, called HI^{2}tswTCM, between the I^{2}tswTCM (Elshaikh *et al.*, 2008) and PaItswTCM (Sudha *et al.*, 2009) to optimize the fairness efficiently over a large range of provision levels.

Traffic markers algorithm: This section provides an overview of several traffic marker algorithms, which are essential models in the DiffServ edge router. Moreover, the marker algorithm contains a set of traffic senders according to the service profile. The traffic can be split into two parts (Sani and Othman, 2011): service profile conforms and non-conforms. The former part is marked with low drop precedence to receive better service, whereas the latter illustrates how the non-conforming traffic is marked with high drop precedence to receive best-effort service.

The time-sliding window Three Color Marker (tswTCM) algorithm by Fang *et al.* (2000) may take advantage of the Time-Sliding Window (TSW). This is done using the rate estimator. The tswTCM measures the arrival rate of any incoming packets. Therefore, packet marking is based on the comparison of the measured average arrival rate with a Committed Information Rate (CIR). Furthermore, in this marker algorithm the packets are classified and marked with green based on the calculated probability and an estimated arrival rate less than the committed information rate.

Improved time-sliding window Three Color Marker (ItswTCM) algorithm is an effortless marker algorithm (Su and Atiquzzaman, 2003). The core idea of this scheme is simple: packets are marked when they exceed their committed information rate. Consequently, the CIR rate should be considered when injecting more yellow packets. This significantly improves the sharing fairness when exceeding bandwidth for low to medium provision levels. Furthermore, when the average arrival rate is less than the CIR, as in the tswTCM scheme, all packets are marked as a green to ensure that the SLA is not missed. Therefore, when the CIR has exceeded the aggregation, all packets are marked as green. Moreover, if traffic throughput exceeds its CIR but is less than (C*CIR), all packets are marked as yellow; otherwise, packets are marked in yellow with probability (P) and packets are marked in red with probability (1-P). The pseudo code (1) describes the ItswTCM algorithm.

Algorithm 1: ItswTCM algorithm

```
Avgrate ← estimated average sending rate of traffic stream.
C ← constant with C >1.
if avgrate < = CIR then
| packet is green
else
    if avgrate < = C * CIR then
    | the packet is mark as yellow
    else
        P = ((C * CIR) / avgrate)^2
        with probability P packet is yellow
        with probability (1-P) the packet is red
    endif
endif
```

The Double Improved time-sliding window Three Color Marker (I^2tswTCM) algorithm was proposed (Elshaikh *et al.*, 2008). This algorithm introduced a new adaptive method of the C, first, introduced in ItswTCM as a constant value, by changing its value based on the estimated rate and available resources results from improving fairness of excess bandwidth sharing. The new value of C is obtained from Eq. 1 as follows:

$$C = \sum_{i=1}^{n} CIR_i + \gamma \quad (1)$$

where, $0<\gamma<1$

The pseudo code (2) described the I^2tswTCM algorithm.

Algorithm 2: I^2tswTCM algorithm

```
Avgrate ← estimated average sending rate of traffic stream.
X ← Bandwidth Link Capacity
γ ← constant with 0<γ<1

S_CIR ← Σ_{i=1}^{N} CIR_i

C ← (X / S_CIR ) + γ
if avgrate < = CIR then
| packet is green
else
    if avgrate < = C * CIR then
    | the packet is mark as yellow
    else
        P = ((C * CIR) / avgrate)^2
        with probability P packet is yellow and with probability (1 - P) the packet is red
        endif
    endif
```

In addition, the pseudo code (2) describes the I^2tswTCM algorithm. The Provision aware improved time-sliding window Three Color Marker (PaI^2tswTCM) algorithm has recently been proposed (Sani and Othman, 2011). The core idea in this algorithm is controlling the amount of yellow and red packets injected to the link under UPN conditions. Furthermore, the PaI^2tswTCM attempts to improve fairness in the UPN condition. Consequently, this algorithm is contains two main parts: the first activates if the network provision level is below the 70%, while the second activates if the network provision level exceeds the 70%. The pseudo code (3) describes the PaI^2tswTCM algorithm. This article thus proposes a new hybrid marker algorithm, called HI^2tswTCM, between the I^2tswTCM (Elshaikh *et al.*, 2008) and PaItswTCM (Sudha *et al.*, 2009) to optimize fairness efficiently over a large range of provision levels.

Algorithm 3: PaItswTCM algorithm

```
Avgrate ← estimated average sending rate of traffic stream.
C ← constant with C > 1.
X ← Bandwidth Link Capacity

S_CIR ← Σ_{i=1}^{N} CIR_i

if (S_CIR < = (0.7 * X)) then
```

Algorithm 3: Continued

```
Avgrate ← estimated average sending rate of traffic stream.
|           if avgrate < = CIR then
|           | packet is green
    else
|   |       if avgrate < = C * CIR then
|   |       | the packet is mark as yellow
|   |       else
|   |       | P = ((C * CIR) / avgrate)^2
|   |       |   with probability P packet is yellow
|   |       |   with probability (1-P) the packet is red
|   |       |-  endif
|   |-  endif
else
|   if avgrate < = CIR then
|   | packet green
|   else
|   |       if avgrate < = ((C * CIR/avgrate)^2 * CIR) then
|   |       the packet is mark as yellow
|   |       else
|   |       |   the packet mark as red
|   |       |-  endif
|   |- endif
|-  endif
```

Proposed algorithm: HI^{2}tswTCM: Finding an efficient optimization solution for unfairness bandwidth sharing in various markers of DiffServ markers is the main objective of this article. Therefore, to attain this objective, a new marker algorithm, called the Hybrid Modified Double Improved time-sliding window Three Color Marker (HI^{2}tswTCM) algorithm is proposed. In the proposed algorithm, the marker mechanism is marks data packets in green to ensure desired service rate. In addition, yellow packets are injected whenever there is excess bandwidth to be shared, after considering their CIRs and the red packets injected into the link are controlled under UPN conditions. Furthermore, the HI^{2}tswTCM merely makes an effort to improve the fairness in the UPN conditions. Accordingly, the HI^{2}tswTCM has two parts: network provision levels either below or above 70%. The former part, is enabled when the network provision level is below 70% and the constant value C, as mentioned in I^{2}tswTCM, will rely upon the value of equal to 0.6 (Elshaikh *et al*., 2008). Moreover, this part has two cases: in the first case, all packets are marked in green if the average arrival rate is less than the CIR, which guarantees the service rate. The second case occurs when the CIR is exceeded and the packets are marked in yellow and red. The latter HI^{2}tswTCM algorithm part is enabled when the provision level is over 70% and the constant value C will rely upon the value of equal to 0.2. Moreover, all packets are marked in green whenever the average arrival rate is less than the CIR and the packets are marked yellow if the rate is between the CIR and $(((C*CIR)/avgrate)^2)$. Otherwise, they are marked red. The pseudo code (4) describes the HI^{2}tswTCM algorithm.

The HI^{2}tswTCM ensures that further yellow packets are injected in proportion to its CIR and PIR more than other marker algorithms, because the HI^{2}tswTCM allows the injection of further yellow packets, which will helps ensure the behavior of the TCP traffic sources from the unbehaved

UDP traffic. If all unbehaved packets marked yellow exceed the service rate, they may suffer from a high drops in case of congestion. However, the behaved packets marked green respond to congestion by reducing the packet rate.

Algorithm 4: HI^{2}tswTCM algorithm

```
Avgrate ← estimated average sending rate of traffic stream.
X ← Bandwidth Link Capacity
γ ← constant with 0.2 or 0.6
N
S_CIR ← CIR_i
i = 0
if S_CIR = (0.7 * X) then
| γ ← 0.6
else
| γ← 0.2
C ← (X / S_CIR ) + γ
if avgrate < = CIR then
packet is green
else
    if avgrate < = C * CIR then
    | the packet is mark as yellow
    else
        P = ((C * CIR)/avgrate)^2
        with probability P packet is yellow
        with probability (1-P) the packet is red
        endif
    endif
```

Simulation setup: This section describes a simulation topology and performance criteria. Several simulation experiments were performed using a network simulator (ns-2) under DiffServ networks (McCanne and Floyd, 1997).

Simulation topology: In the simulation topology (Fig. 1), there are three edge routers (i.e., E1, E2 and E3) and all edge routers are attached to a Core Router (CR). The edge routers E1 and E2 are responsible for monitoring and marking packets for aggregates 1 and 2, respectively. The CR maintains active queue management and provides service differentiation between E1 and E2 according to the drop precedence carried in the packet headers. Moreover, the network has two groups of aggregate input sources, aggregate 1 and aggregate 2. The aggregate 1 is responsible for sending packets through the E1 to destination D1. Similarly, the aggregate 2 sends packets through edge router E2 to destination D2.

Simulation parameters: The simulation used the multi-RED queue management scheme. Moreover, the notation (x; y; z) represents the minimum threshold, maximum threshold and weight parameter of the RED queue, respectively. The core router setting were 0,40,0.2, 40,80,0.1 and 80,120,0.02 for the red, yellow and green packets, respectively (Floyd and Jacobson, 1993). The following general value is given in the RED configuration parameters (Clark and Fang, 2002; Floyd and Jacobson, 1993; Pieda *et al*., 2000). The simulation parameters used here are TCP Reno with

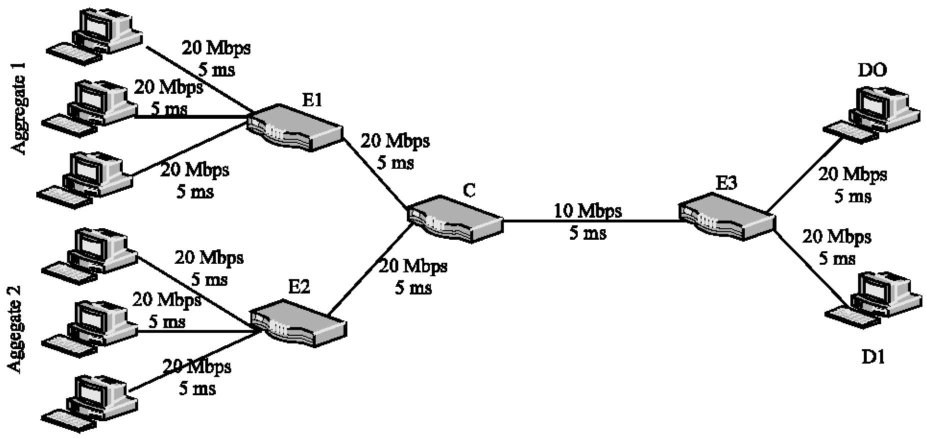

Fig. 1: Simulation topology

FTP application. For each aggregate, the maximum segment size (MSS) is 1500 bytes and the Round Trip Time (RTT) is 40 milliseconds (ms). For ItswTCM and PaItswTCM, the constant value of C is set to 2 (Sani and Othman, 2011; Su and Atiquzzaman, 2003). The bandwidth link capacity used as the X value in this article is set to 10 Mbps between the core router CR and edge router E3 and the used capacity value is set to 20 Mbps for other bandwidth links.

Performance metrics: The Fairness Index FI, is the main performance metric, which is the ratio between the numbers of data packets successfully received at the destination. Equation 2 presents the Fairness Index; according to this equation, the closer value of FI is set to 1 (Jain, 2008).

$$FI = \frac{(\sum_i [\![X_i)^2]\!]}{N * \sum_i X_i^2} \quad (2)$$

where, $0<FI<1$ and

$$X_i = \frac{\text{Excess bandwidth obtained by aggregate i}}{\text{CIR of aggregate i}}$$

N is the total number of aggregates; we use a proportional fair share rather than just equal shares.

Simulation results and discussion: This section shows simulation results of the proposed algorithm compared with other related algorithms, including ItswTCM (Su and Atiquzzaman, 2003), I^2tswTCM (Elshaikh *et al.*, 2008) and PaItswTCM (Sani and Othman, 2011).

Figure 2 shows the behavior fairness index with respect to the provision level where aggregates sharing the link have equal numbers of flows. The effect of different constant values used in I^2tswTCM, as mentioned in the pseudo code (2), compared to values used in the proposed algorithm,

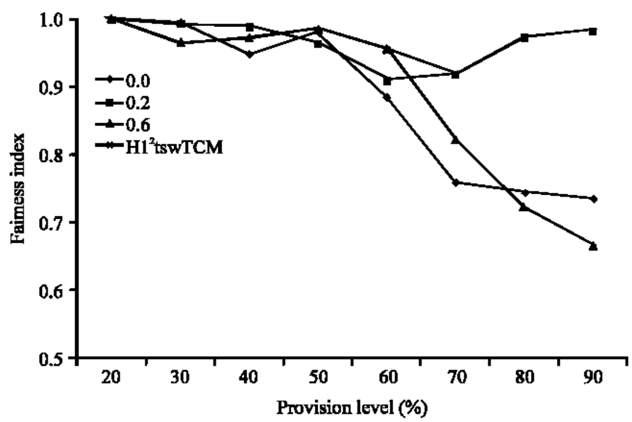

Fig. 2: Effects of constant value

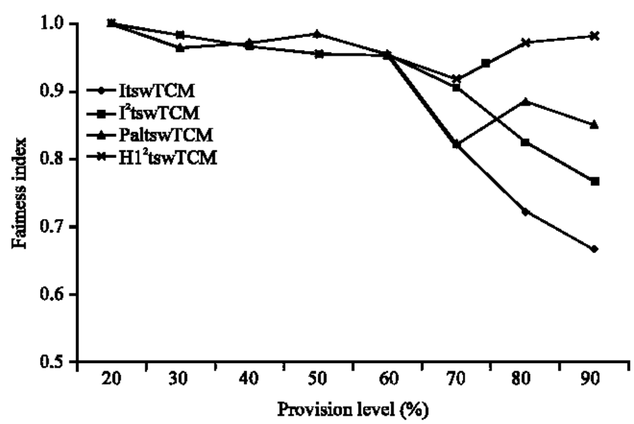

Fig. 3: Scenario 1, with 16 flows

as mentioned in the pseudo code (4), can be seen in Fig. 2, which also shows similar curves in all cases but with differing performance for each fairness index value. By setting a different value, the proposed algorithm HI^{2}tswTCM performs better than the I^{2}tswTCM algorithm in fairness, especially in the provision level range between 70-90%.

In all scenarios, we studied the fairness for the different marker algorithms in three different scenarios. The CIR for aggregate 1 = 1 Mbps, while the CIR for aggregate 2 increases from 1 Mbps to 8 Mbps, corresponding to the network moving from a provision level of 20 to 90%, respectively. The provision level is the ratio of total provisioned bandwidth to the bandwidth at the bottleneck link, which equals to 10 Mbps.

Figure 3 shows the performance results of the first scenario, which has an equal number of flows, set to 16 for each aggregate sharing the bottleneck link. Furthermore, the proposed HI^{2}tswTCM algorithm, shown in Fig. 3, performed better compared to the other marker algorithms in fairness for a long range of network provisions, up to 90%. This indicates that the proposed HI^{2}tswTCM increases the fairness in all provisioned networks and it's suitable to operate with better fairness, achieving almost a value of 1.

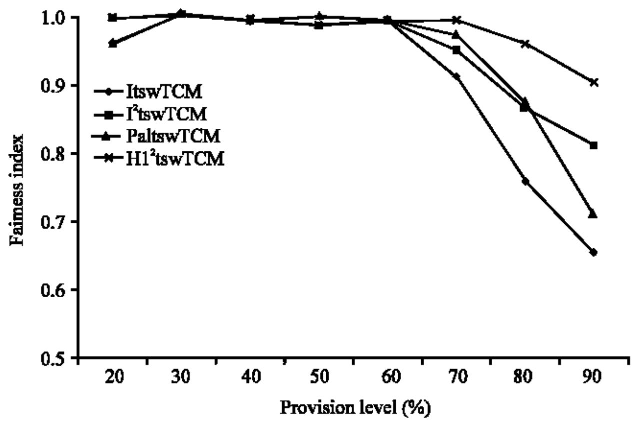

Fig. 4: Scenario 2, with differing numbers of flows

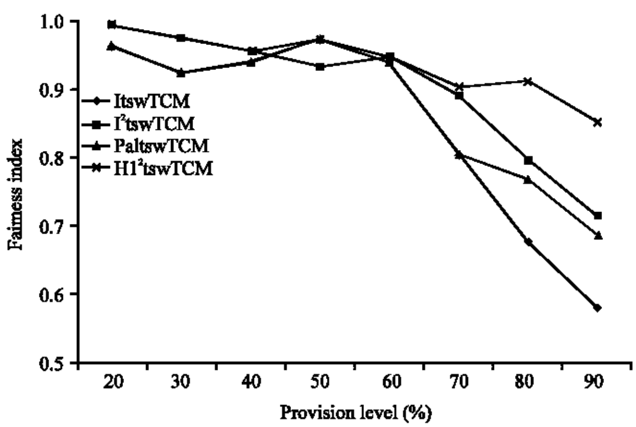

Fig. 5: Scenario 3, where the numbers of flows is different

Figure 4 illustrates the second scenario, showing the fairness index with respect to the provision level where the aggregates sharing the link have a different number of flows and compares the performance evaluation between several marker algorithms to the proposed HI^2tswTCM algorithm. In this scenario, aggregate 1 has fewer flows than aggregate 2 (i.e., aggregate1 has 16 flows and aggregate2 has 32 flows). Figure 4 shows that the proposed HI^2tswTCM algorithm performs better than other algorithms in the 70-90% provision level for the reason stated above.

Figure 5 shows the third scenario, in which the performance results show that all aggregates sharing the bottleneck link have a different number of flows. In this scenario, aggregate 2 has fewer flows than aggregate1 (i.e., aggregate 1 has 32 flows and aggregate 2 has 16 flows). Figure 5 shows that the HI^2tswTCM algorithm performs better than the other marker algorithms for a long range of network provisions, up to 90%.

The performance results presented in Figure 6 show the scenario where three aggregate share the bottleneck link, which has 16 flows in each aggregate. The CIR of aggregate1 and 2 is fixed at 1 Mbps and aggregate 3 varies between 1 to 7 Mbps. The simulation results show that the fairness achieved by the HI^2tswTCM marker is over a large range, between 70 and 90% of the provision level. This reveals the insensitivity of the marker to the number of aggregates in the network.

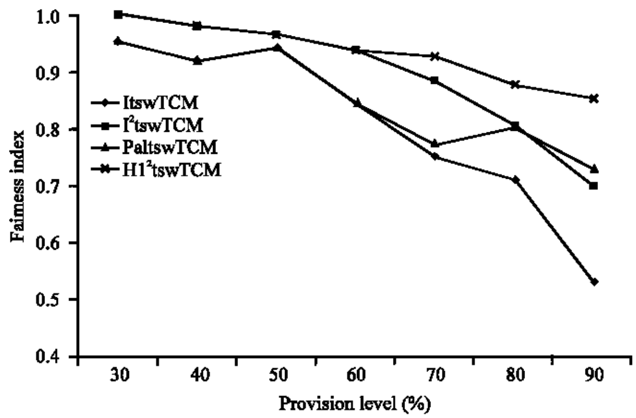

Fig. 6: Three aggregates, with equal flows

CONCLUSIONS

This article has proposed a new Hybrid Marker algorithm (HI^2tswTCM) to enhance fairness when DiffServ networks have excess bandwidth among TCP aggregates. This hybrid marker more efficiently improves fairness in bandwidth sharing. To improve the proportional fair share performance of excess bandwidth, injecting more yellow packets into the network through recalculation based on the better values of C is important. The HI^2tswTCM also injects extra yellow packets more than in ItswTCM, I^2tswTCM and PaItswTCM. Lastly, the yellow packets play a significant role in improving performance for the proportional fair share of excess bandwidth in a DiffServ network. Several simulation experiments were performed, showing that the proposed HI^2tswTCM has superior performance results than other marker algorithms for network provision levels between 70-90%.

REFERENCES

Blake, S., D. Black, E. Davies, Z. Wang and W. Weiss, 1998. An architecture for differentiated services. IETF draft, RFC 2475, pp: 1-10.

Braden, B., D. Clark and S. Shenker, 1994. Integrated services in the internet architecture: An overview. RFC, Publishers, 1633 United States.

Carpenter, B. and K. Nichols, 2002. Differentiated services in the Internet. Proc. IEEE, 90: 1479-1494.

Chrysostomou, C., A. Pitsillides and Y.A. Sekercioglu, 2009. Fuzzy explicit marking: A unified congestion controller for best-effort and diff-serv networks. Computer Networks: The Int. J. Comput. Telecommun. Network., 53: 650-667.

Clark, D.D. and W. Fang, 2002. Explicit allocation of best-effort packet delivery service. IEEE/ACM Trans. Networking, 6: 362-373.

Elshaikh, M.A., M. Othman, S. Shamala and J.M. Desa, 2008. A new fair marker algorithm for DiffServ networks. Comput. Commun., 31: 3064-3070.

Fang, W., N. Seddigh and B. Nandy, 2000. RFC2859: A Time Sliding Window Three Colour Marker (TSWTCM). The Internet Society, http://tools.ietf.org/html/rfc2859.

Floyd, S. and V. Jacobson, 1993. Random early detection gateways for congestion avoidance. IEEE/ACM Transa. Networking, 1: 397-413.

Heinanen, J. and R. Guerin, 1999a. RFC2697: A single rate three color marker. The Internet Society, USA. http://tools.ietf.org/html/rfc2697.

Heinanen, J. and R. Guerin, 1999b. RFC2698: A two rate three color marker. The Internet Society, USA. http://tools.ietf.org/html/rfc2698.

Heinanen, J., F. Baker, W. Weiss and J. Wroclawski, 1999. RFC 2597: Assured forwarding PHB group. June 1999. http://portal.acm.org/citation.cfm?id=RFC2597.

Heusse, M., P. Starzetz, F. Rousseau, G. Berger-Sabbatel and A. Duda, 2003. Bandwidth allocation for DiffServ based quality of service over 802.11. Proc. Conf. Global Telecommun., 247: 992-997.

Ibanez, J. and K. Nichols, 1998. Internet draft: Preliminary simulation evaluation of an assured service. http://tools.ietf.org/id/draft-ibanez-diffserv-assured-eval-00.txt.

Jain, R., 2008. The Art of Computer Systems Performance Analysis. Wiley India Pvt. Ltd., India, ISBN:8126519053, Pages: 716.

McCanne, S. and S. Floyd, 1997. UCB/LBNL/VINT network simulator-ns: Version 2. Software and Documentation, UCB/LBNL/VINT Network.

Nichols, K., S. Blake, F. Baker and D.L. Black, 1998. Definition of the differentiated services field (ds field) in the IPv4 and IPv6 Headers. RFC 2474.

Pieda, P., J. Ethridge, M. Baines and F. Shallwani, 2000. A network simulator differentiated services implementation. Open IP, Nortel Networks, pp: 1-43, http://marco.uminho.pt /disciplinas/ST/st0607/DSnortel.pdf

Sani, Y. and M. Othman, 2011. A provision-aware fair bandwidth distribution marker algorithm for DiffServ networks. J. Network Comput. Applic., 34: 715-721.

Su, H. and M. Atiquzzaman, 2001. Performance modeling of differentiated service network with a token bucket marker. Proceedings of the 11th Workshop on Local and Metropolitan Area Networks, March 18-21, 2001, IEEE., pp: 81-82.

Su, H. and M. Atiquzzaman, 2003. ItswTCM: A new aggregate marker to improve fairness in DiffServ. Comput. Commun., 26: 1018-1027.

Sudha, S., B. Rao and N. Ammasaigounden, 2009. A modi?ed I2TSWTCM to improve bandwidth fairness in Diffserv. Proceedings of the 3rd International Conference on Industrial and Information Systems, December 8-10, 2009, IEEE., pp: 1-4.