Proportional Bandwith Distribution in IP Networks Implementing the Assured Forwarding PHB

Maria-Dolores Cano, Fernando Cerdan
Department of Information Technologies and Communications
Polytechnic University of Cartagena
Campus Muralla del Mar s/n, 30202 Cartagena, Spain
{mdolores.cano, fernando.cerdan}@upct.es

Abstract

Recent demands for new applications are giving rise to an increasing need of Quality of Service (QoS). Nowadays, most IP-based networks tend to use the DiffServ architecture to provide end-to-end QoS. Traffic conditioners are a key element in the deployment of DiffServ. In this paper, we introduce a new approach for traffic conditioning based on feed-back signaling among boundary nodes and traffic conditioners. This new approach is intended to provide a poportional distribution of excess bandwidth to end-users. We evaluate through extensive simulations the performance of our proposal in terms of final throughput, considering contracted target rates and distribution of spare bandwidth. Results show a high level of fairness in the excess bandwidth allocation among TCP sources under different network conditions.

1. Introduction

Inside the Differentiated Service (DiffServ) architecture [1], the Assured Forwarding PHB (AF-PHB) [2, 3] is one of the current PHBs (Per Hop Behavior) with the status of standard. The idea behind the AF-PHB is to assure a minimum throughput, the contracted target rate, to an end-user while enabling consuming excess bandwidth if the network load is lower than the maximum link utilization. Excess bandwidth is defined as the remaining available bandwidth once all connections have a throughput equal to their contracted target rates. In the AF-PHB service, this excess bandwidth should be shared in a fair way. The term *fair* can be understood in two different ways: i) a fair excess bandwidth sharing means an even distribution of the spare bandwidth among all sources composing the aggregate; or ii) a fair excess bandwidth means sharing the spare bandwidth proportionally to the contracted target rate

of each source. In this work we follow the second approach. Notice that we use the term throughput without considering retransmitted packets, which is usually called goodput.

In the AF-PHB, there are four independently forwarded AF instances. Within each AF instance, an IP packet is assigned one of three different levels of precedence. Packets that conform to the contracted target rate are called *in-of-profile* (*in*), while non-conformant packets are called *out-of-profile* (*out*). In this case only two levels of precedence are used. When network congestion occurs, DiffServ nodes try to protect packets with a lower drop precedence value from being lost by preferably discarding packets with higher drop precedence.

Most related literature has focused on traffic conditioners for the AF-PHB Service, presenting different proposals to complete the AF goals. The first goal, assuring the contracted target rate of the final user, has been achieved for many of the published schemes. As regard to the second goal, studies done in [4, 5, 6] introduce algorithms for achieving proportional fairness in the AF-PHB Service. These proposals have in common the use of three colors for each AF-PHB instance. To the best of our knowledge, there is only one approach to offer a proportional distribution of excess bandwidth using two levels of precedence. In [7], the authors propose TATC (Traffic Aware Traffic Conditioner). This algorithm allocates back *out-of-profile* bandwidth to *in-profile* bandwidth in proportion to the target rates, what presumably leads to higher assured bandwidth for flows with high target rates. On the other hand, algorithms such as TSW (Time Sliding Window) [8] or ETSW (Enhanced Time Sliding Window) [9] were employed to compare the performance of EBM [5] in terms of excess bandwidth sharing. Although neither TSW nor ETSW were thought to carry out a proportional distribution of excess bandwidth, the widespread use of TSW turned them into classical references. The lack of contributions that use two levels of precedence

IEEE COMPUTER SOCIETY

motivates us to employ an enhanced version of TSW [9, 10] to perform comparisons with our proposal.

In this paper we introduce a new algorithm for proportional excess bandwidth sharing for the AF Service, so end-users get excess bandwidth in proportion to their target rates. The proposed traffic conditioner, which performs marking and policing functions, is placed next to the source of traffic (where the contract is established), but out of the reach of the final user. Basically, our proposal marks IP packets with one of two drop precedences (*in* and *out*, what simplifies the scheme) using the CB marker [10], and then our New Policy Function (NPF) is applied. NPF is based on adapting the source throughput to network conditions by discarding packets if necessary. To carry out this task, NPF needs a very simple signaling between the edge network node and traffic conditioners. This signaling is not a problem given that this path corresponds to the user local loop (short distances). The traffic conditioner with NPF is evaluated through extensive simulations, and results show that the AF Service goals are widely satisfied.

The rest of the paper is organized as follows. Section 2 describes the characteristics of our NPF proposal. Section 3 details the simulation topology and the simulation tool employed to conduct the performance evaluation. In Section 4 we present simulation results, and compare fairness and user contracts guarantees with results obtained for the improved Time Sliding Window (TSW) algorithm. We end with conclusions in Section 5.

2. The NPF algorithm

Let us denote by c the link capacity and by b the sum of all contracted target rates of those sources that join in a boundary node. For two-color based traffic, we can define α_{ideal} as indicated in (1): the quotient link capacity c minus b divided by b. Notice that the upper part of this fraction represents the excess bandwidth. Observe that α_{ideal} is a fixed value unless a user changes his/her contracted target rate, a user cancels his contracted target rate or there is a new user with a new contracted target rate.

$$\alpha_{ideal} = \frac{link\ capacity - \sum_{i=1}^{n} target\ rate_i}{\sum_{i=1}^{n} target\ rate_i} = \frac{c-b}{b} \quad (1)$$

Suppose that we measure in the time interval (t_1, t_2) the ratio number of *out* packets divided by number of *in* packets that leave the boundary node. We call this value α_m (2). For simplicity, we assume all packets

have similar size, but equation (2) can be also calculated with the sum of packet sizes. Then, if link utilization is about hundred percent, α_m and α_{ideal} should be almost equal. That is, if we subtract b from the link capacity c we obtain the excess bandwidth, and the excess bandwidth is represented by all packets marked as *out*. Similarly, b is represented by all packets marked as *in*. In consequence, the ideal situation yields to α_m equal to α_{ideal}.

$$\alpha_m = \frac{\sum_{t_1}^{t_2} out\ packets}{\sum_{t_1}^{t_2} in\ packets} \quad (2)$$

Notice that there is a traffic conditioner placed next to each source but out of the reach of the final user. If we measure the ratio α_m at the output of each traffic conditioner, we can use α_{ideal} to achieve a fair proportional distribution of the excess bandwidth. Given the ideal value α_{ideal}, we compare it with the corresponding α_m values obtained for each source of traffic (α_m^i, where i is the source number). If α_m^i is less than α_{ideal} then the source is not consuming its corresponding excess bandwidth. If both values coincide, then the source consumes exactly its corresponding spare bandwidth. Finally, if α_m^i is greater than α_{ideal} then this source i is consuming bandwidth beyond its fair quota. Therefore, when it is detected an α_m^i value greater than α_{ideal} this source i has to be penalized to decrement its throughput. Fig.1 shows the general procedure of NPF.

Let us give an example. Suppose there are two sources, s_1 and s_2, with contracted target rates of 1 and 10 Mbps respectively. The link capacity is 33 Mbps, so α_{ideal} is in this case equal to 2. For a fair excess bandwidth distribution s_1 should get 2 Mbps of the excess bandwidth, whereas s_2 should obtain 20 Mbps, both values in proportion to their service profiles. If we measure α_m^1 and α_m^2 and both are equal to 2, then s_1 gets 2 Mbps of excess bandwidth (two times its contracted target rate), and the same applies to s_2 that would get 20 Mbps (again, two times its contracted target rate). On the contrary, if α_m^1 is equal to 3, then s_1 gets 3 Mbps of excess bandwidth, so is stealing bandwidth that proportionally corresponds to s_2. If α_m^1 is equal to 1, then s_1 only obtains 1 Mbps of spare bandwidth, thus s_2 has to decrement its throughput because is consuming more bandwidth than allowed.

From this example we extract that it is possible to know the behavior of the sources by comparing α_m^i and α_{ideal}. We've seen that with the relation $\alpha_m^i = \alpha_{ideal}$ the system works well, but the inequality identifies
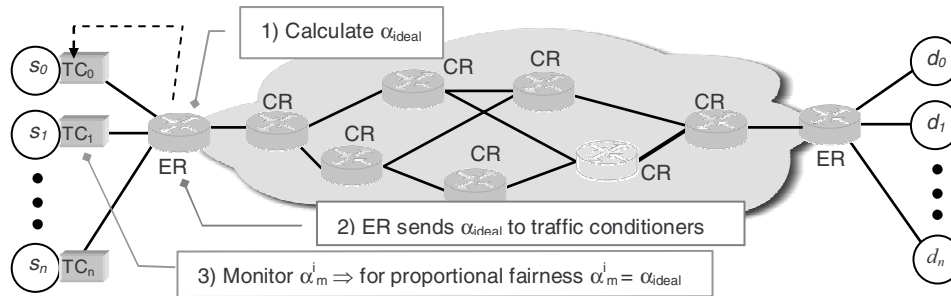
Fig 1. General procedure using NPF. Edge routers (ER) calculate α_{ideal}. α_{ideal} is sent to traffic conditioners (TC) where the policy NPF algorithm is applied: monitor α^i_m to control the source throughput. Core routers (CR) only perform the AF PHB.

unfairness in the excess bandwidth distribution. In fact, if $\alpha^i_m > \alpha_{ideal}$ then source i has to decrement its throughput. Because we are working with TCP sources, packet losses make sources to slow down. Since this is our goal, we employ packet discarding in the corresponding traffic conditioner i when the condition $\alpha^i_m > \alpha_{ideal}$ is detected.

There are different options to be applied for packet discarding. One of them consists of dropping packets if $\alpha^i_m > \alpha_{ideal}$ independently on the type of packet (*in* or *out*). The question associated to this option is that discarding *in* packets may cause problems for assuring contracted target rates. The solution that we have adopted is to discard only *out* packets when the condition $\alpha^i_m > \alpha_{ideal}$ is true. The proposed NPF algorithm operates as shown in Fig. 2.

```
αⁱₘ = ratio (out packets/in packets)
if (αⁱₘ ≤ αideal)
  do not discard the packet
else
    if packet is in
        do not discard the packet
  else
        discard the packet
```
Fig. 2. NPF algorithm at the traffic conditioner.

Before applying NPF as a policy function, packets are marked in the traffic conditioner. In this case, marking is done with the Counters-Based algorithm (CB) introduced in [10] and used in [11]. CB performs comparatively better than other marking schemes like TSW or Leaky Bucket. Its main advantages are an easy configuration and high accuracy in guaranteeing the contracted target rates in heterogeneous scenarios. CB uses two counters and includes a simple mechanism to avoid accumulation of "credits" when a source stops transmitting data, for instance when a time out expires. The pseudocode of CB is written in Fig. 3.

```
Initially:
   counter₁=1
   counter₂=link_rate/target rate
For each unit of time:
   counter₂--
   if counter₂≤0
     counter₁++
     counter₂=link_rate/target rate
   if there is a packet arrival
     if counter₁>0
       packet marked as in
       counter₁--
     else
       packet marked as out
```
Fig. 3. Counters-Based marking algorithm.

3. Simulation setup

The simulation tool for the sliding window protocol of TCP Reno sources was developed in [12] and widely used in [13][14]. Some of its features are: for a worst case study TCP sources are long-lived, that is, they have unlimited data to send; destinations only send acknowledgements, which are never lost or delayed; and the maximum window size equals the product bandwidth delay as usual for WAN environments. We employ a large IP packet size of 9,188 bytes, which corresponds to classical IP over ATM (Asynchronous Transfer Mode).

The simulation topology is shown in Fig. 4. There are n TCP Reno sources (s_1, s_2, ..., s_n) transmitting at the link rate, which has been set to 33 Mbps. All sources send traffic to destinations (d_1, d_2, ..., d_n) through the edge node E_1. The bottleneck is placed between the edge nodes E_1 and E_2, since the sources transmit at link rate. For the AF PHB, edge nodes employ Dual Queuing [15] instead of RIO (RED (*In* and *Out* packets)) [8]. Dual Queuing (DQ) is a buffer management scheme that treats *in* and *out* packets

differently but avoiding interference between them. DQ places *in* and *out* packets in separate FIFO queues. To reduce complexity, both queues are served in a weighted round-robin fashion. The scheduler serves the queue buffering *in* packets with a probability that matches the total contracted traffic load (ρ), and with probability $1-\rho$ the queue buffering *out* packets. The maximum number of packets that can be stored are limited by the thresholds HBO_{in} (High Buffer Occupancy for *in* packets) and HBO_{out}. We select 20 packets for HBO_{in} and 10 packets for HBO_{out}. Using DQ (see Fig. 5), packets may not be served in the same order they arrive to the network node, but this does not represent a problem with TCP.
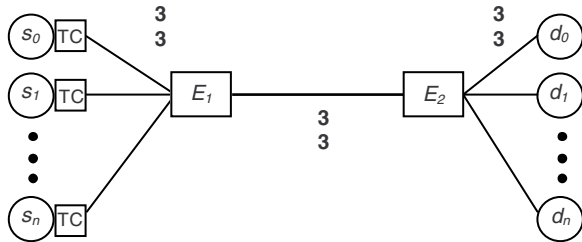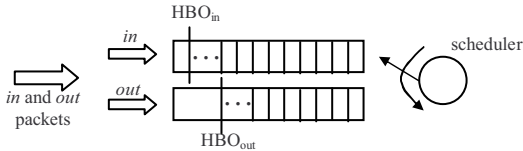

Fig. 4. Simulation topology.

Fig. 5. Dual Queuing buffer management.

The delay in the bottleneck is 2.5 ms, and 2 ms from the edge node E_2 to destinations. We use different values x_i for the delay between sources and E_1 to evaluate scenarios with different round trip times. Contracted target rates are also variable depending on the case under study. Moreover, we obtain results varying the network provision level from 20 to 90% in all cases. Simulation results have a confidence interval of 95% that has been calculated with a normal distribution function using 30 samples, with an approximate value of ±0.002 for fairness calculations and ±0.01 for achieved target rates.

4. Results

In this section we present and discuss simulation results. We evaluate the performance of the proposed traffic conditioner with NPF in terms of guarantees of achieving the contracted target rates and fairness in excess bandwidth sharing. Results are also compared with the classical TSW. To analyze the fairness of

different schemes we use the definition given in [16], where the fairness index *f* is calculated as follows (3):

$$f = \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N \cdot \sum_{i=1}^{N} x_i^2} \qquad (3)$$

$$x_i = \frac{total\ throughput\ i - contracted\ target\ rate\ i}{contracted\ target\ rate\ i} \qquad (4)$$

Where x_i is the excess throughput of source *i* divided by the contracted target of source *i* (4), and *N* is the number of sources that arrive to the boundary node. The closer to 1 in the *f* value, the fairer is the sharing of the spare bandwidth.

4.1. Case 1: same contracted target rates and round trip times

In this case, eight TCP Reno sources have contracted the same target rate and round trip time is set to 50 ms for all connections (x_i = 20.5 ms, for i=0..7 in Fig. 4). This is the simplest scenario we can consider. Table 1 illustrates the strong assurance in achieving contracted target rates for a provision level of 60% with NPF. We see that end-users achieve their corresponding targets. Comparing with the TSW, the main difference is that with TSW the measured *in* packets throughput does not guarantee the end-user target. In this case, this fact does not represent a problem because targets are guaranteed with the total throughput. However, it can be an inconvenience in more extensive topologies, because *out* packets have less priority and may be discarded at intermediated network nodes.

Our NPF scheme performs as expected in this first scenario. Dropping *out* packets when it is detected a value of $\alpha^i_m > \alpha_{ideal}$, makes the TCP source to slow down. For a network provision level varying from 20% to 90%, Fig. 6 shows that both TSW and NPF present a fairness index above 0.95 in the whole range, but keeping NPF slightly higher values.

4.2. Case 2: variation in contracted target rates and same round trip times

In this section, the topology consists of eight TCP Reno sources whose contracted target rates are variable. For instance, for a network provision level of 60% sources s_0 to s_7 have contracted targets of 1-1-2-2-3-3-4 and 4 Mbps respectively. The round trip time is set to 50 ms (x_i = 20.5 ms for i=0..7 in Fig. 4). Even with variation of targets among the different

connections, NPF shows strong guarantees in achieving the contracts. With TSW, targets are reached thanks to the *out* packets, as it happened in case 1.

Regarding the fairness, we perceive in Fig. 7 that NPF presents a fairness index over the values obtained with TSW. It is important to remark the importance of a fairness value close to or over 0.8. Fig. 7 clearly shows that TSW only gets this value in a short range (35-45%), while NPF is over 0.8 in almost the entire range. This means that with NPF, all sources are getting their corresponding proportional part of the excess bandwidth. As it could be thought, none of the packet drops in NPF causes the source to completely stop transmitting data. On the contrary, it makes the sources to adapt to network conditions with the indications given by α_{ideal} and α^i_m.

Table 1. Throughputs obtained in case 1 (S ≡ source; TR ≡ Contracted target rate; TH$_{in}$ ≡ Obtained throughput for *in* packets; TH$_{total}$ ≡ Total throughput) (provision level 60%)

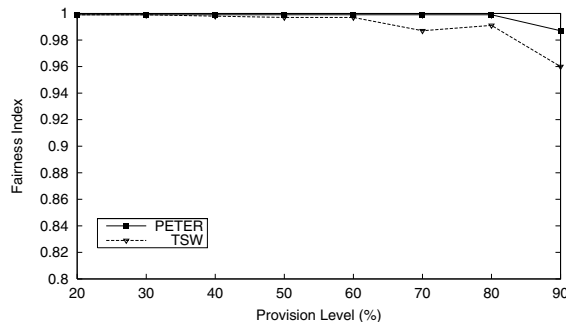| S	TR (Mbps)	TH$_{in}$ (Mbps)		TH$_{total}$ (Mbps)	
		NPF	TSW	NPF	TSW
0	2.5	2.49	2.19	3.95	3.96
1	2.5	2.45	2.25	3.92	4.18
2	2.5	2.48	2.22	3.94	4.03
3	2.5	2.38	2.18	3.89	3.96
4	2.5	2.46	2.28	3.85	4.10
5	2.5	2.42	2.17	3.85	3.91
6	2.5	2.42	2.20	3.90	4.17
7	2.5	2.40	2.20	3.91	3.97

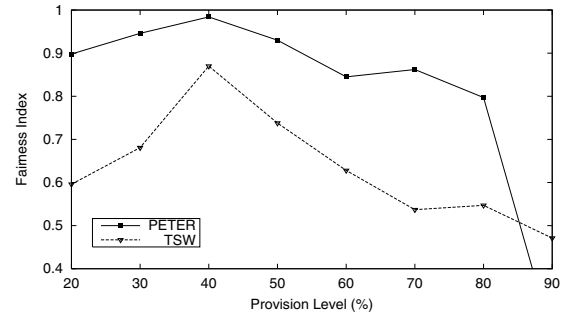Fig. 6. Fairness index vs. provivion level in case 1 with NPF and TSW.

Fig. 7. Fairness index vs. provivion level in case 2 with NPF and TSW.

4.3. Case 3: same contracted target rates and variation in round trip times

To evaluate the effect of having different RTT in the network, we consider an scenario with eight TCP Reno sources where all of them have the same contracted target rates but the RTT goes from 10 to 80 ms (in Fig. 4 $x_0 = 0.5$ ms; $x_1 = 5.5$ ms; $x_2 = 10.5$ ms; $x_3 = 15.5$ ms; $x_4 = 20.5$ ms; $x_5 = 25.5$ ms; $x_6 = 30.5$ ms; and $x_7 = 35.5$ ms*). It is well known the influence that the RTT has on the final throughput [17]. In heterogeneous scenarios, there is a bias against connections with larger RTT unless this effect is alleviated in some way. Results reported in Table 2 show that targets are clearly fulfilled with NPF. We find the same problem for TSW as in previous cases, because *in* packets do not guarantee by themselves the contracts. It is important to remark that the diversity in RTT does not influence the NPF performance.

Table 2. Throughputs obtained in case 3 (provision level 60%)

S	TR (Mbps)	TH$_{in}$ (Mbps)		TH$_{total}$ (Mbps)	
		NPF	TSW	NPF	TSW
0	2.5	2.49	1.85	3.93	3.58
1	2.5	2.48	1.85	3.57	4.26
2	2.5	2.48	2.00	3.68	4.25
3	2.5	2.49	2.18	3.76	4.21
4	2.5	2.50	2.25	3.85	4.00
5	2.5	2.49	2.40	3.76	4.27
6	2.5	2.49	2.47	3.47	4.15
7	2.5	2.49	2.44	3.15	3.55

The following figure, Fig. 8, reveals that NPF presents a fairness index above 0.8 for a network provisioning level in the range 20-76%. TSW also shows good values for the fairness index. From our point of view, this fine behavior of TSW is due to the interaction with the DQ buffer management scheme

employed in the edge nodes. Comparing the TSW performance with RIO (simulations not shown but conducted) and TSW with DQ, the latter presents better results in terms of fairness in the excess bandwidth sharing.
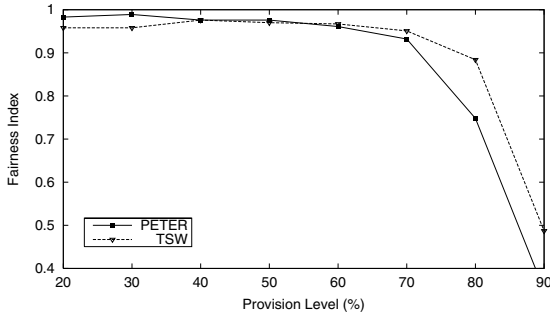

Fig. 8. Fairness index vs. provivion level in case 3 with NPF and TSW.

4.4. Case 4:variation in contracted target rates and round trip times

For a more realistic environment, we study in this case the performance of NPF and TSW in a scenario with eight TCP Reno sources with different targets and different round trip times. The round trip times are set as in case 3 from 10 to 80 ms. Targets also vary depending on the network load. For instance, for a 60% network provision level targets vary between 1 and 4 Mbps. From results shown in Table 3, NPF obtains a hard assurance of target rates for all connections, and so does TSW, but again with the help of *out* packets. Despite the heterogeneity of this scenario, Fig. 9 evidences the superiority of NPF to provide a fair excess bandwidth share. Meanwhile, TSW behaves worst as the heterogeneity increases.

4.5. Case 5: increment in number of sources

In this last case under consideration, we analyze the effect of incrementing the number of sources that arrive to the boundary node (E_1 in Fig. 4). We simulate a scenario with sixteen TCP Reno sources, where the first four connections (s_0 to s_3) have always a contracted target of 1 Mbps. The other twelve sources (s_4 to s_{15}) contract the same target rates to fill a network provision level from 20% to 90%. For example, for a 60% provision level sources s_4 to s_{15} contract 1.32 Mbps each. Although not shown, we obtain a good performance in assuring the contracts with our proposed traffic conditioner.

Fig. 10 represents the fairness index versus the network provision level for the two schemes NPF and TSW. Even though this scenario benefit both schemes (no round trip time variation), we see that NPF performs significantly better than TSW for nearly the entire range of provision level, with negligible differences at medium provision levels. This example shows the robustness of DiffServ with the NPF algorithm, since increasing the number of sources does not represent a degradation of the final performance.

Table 3. Throughputs obtained in case 4 (provision level 60%)

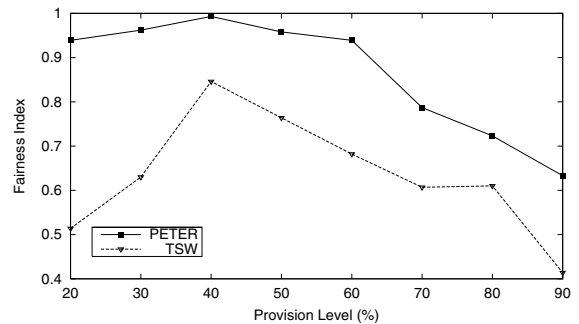| S	TR (Mbps)	TH_{in} (Mbps)		TH_{total} (Mbps)	
		NPF	TSW	NPF	TSW
0	4	3.99	2.91	4.87	4.43
1	4	3.99	3.17	5.80	5.68
2	3	2.99	2.42	4.70	4.49
3	3	3.00	2.58	4.27	4.40
4	2	1.98	1.72	3.00	3.84
5	2	1.98	1.82	3.10	3.83
6	1	0.99	0.88	1.40	2.98
7	1	0.98	0.90	1.33	2.47

Fig. 9. Fairness index vs. provivion level in case 4 with NPF and TSW.
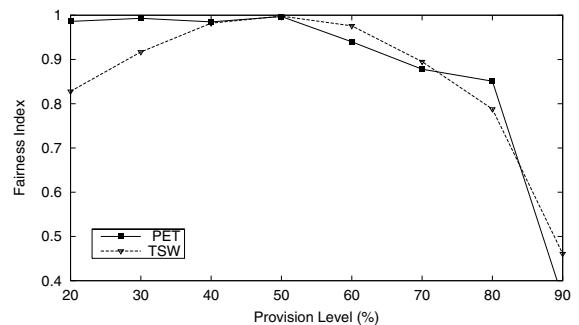

Fig. 10. Fairness index vs. provivion level in case 5 with NPF and TSW.

5. Conclusions

   In this paper we propose a new traffic conditioner for the AF PHB service that provides a strong assurance of contracted target rates and a fair distribution of spare bandwidth. We understand by fair distribution, the usage of the excess bandwidth proportionally to the contracted target rate of each source. The key to achive a fair share is the performance of the NPF algorithm. Once packets are marked with one of two levels of precedence (*in* or *out*), NPF is the policy function applied in the traffic conditioner. In the boundary node it is calculated the ratio excess bandwidth divided by the sum of all contracted target rates (α_{ideal}). This value is sent to the traffic conditioners, placed next to TCP sources but out of the reach of the final users. Traffic conditioners measure the relation number of *out* packets divided by number of *in* packets (α^i_m). For a fair share of excess bandwidth, it is shown that the relation $\alpha^i_m = \alpha_{ideal}$ has to be true, otherwise NPF acts moving the relation between α^i_m and α_{ideal} to the equality.

   We extensively study the performance of our traffic conditioner for many different network conditions: variable target rates, variable round trip times, variability of both targets and delays, and increase of the number of sources that join in the boundary node. We observe that our scheme is able to guarantee contracted target rates in all cases and simultaneously offer a proportional distribution of excess bandwidth for the network provision level in the range 20-70%, where it is supposed that most networks operate. Moreover, it performs generally better than the improved TSW. We conclude that it is possible to contract an AF Service satisfying IP traffic guarantees, as shown in this paper.

6. References

[1] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998.

[2] J. Heinanen, F. Baker, W. Weiss, J. Wroclawski, "Assured Forwarding PHB Group", RFC 2597, June 1999.

[3] D. Grossman, "New Terminology and Clarifications for DiffServ", RFC 3260, April 2002.

[4] E. Kusmierek, R. Koodli, "Random Packet Marking for Differentiated Services", UMN Technical Report TR-00-020, Dept. of Computer Science & Eng., University of Minnesota, 2000.

[5] M. A. El-Gendy, K. G. Shin, "Equation-Based Packet Marking for Assured Forwarding Services", Proceedings of IEEE Infocom 2002, Vol. 2, pp. 845-854, New York, USA, June 2002.

[6] H. Su, M. Atiquzzaman, "ItswTCM: a new aggregate marker to improve fairness in DiffServ", Computer Communications 26, pp. 1018-1027, 2003.

[7] B. Nandy, N. Seddigh, P. Pieda, J. Ethridge, "Intelligent Traffic Conditioners for Assured Forwarding Based Differentiated Services Networks", Proceedings of Networking 2000, Paris, France, pp.540-554, May 2000.

[8] D. Clark and W. Fang, "Explicit Allocation of Best-Effort Packet Delivery Service", IEEE/ACM Transactions on Networking, Vol. 6, No. 4, pp. 362-373, August 1998.

[9] W. Lin, R. Zheng, J. C. Hou, "How to Make Assured Service More Assured", Proc. of the 7th International Conference on Network Protocols (ICNP'99), Toronto, Canada, pp. 182-191, October 1999.

[10] Maria-Dolores Cano, Fernando Cerdan, Joan Garcia-Haro, Josemaria Malgosa-Sanahuja, "Performance Evaluation of Traffic conditioner Mechanisms for the Internet Assured Service", in Quality of Service over Next-Generation Data Networks, Proceedings of SPIE Vol. 4524, pp. 182-193, 2001.

[11] Maria-Dolores Cano, Fernando Cerdan, Joan Garcia-Haro, Josemaria Malgosa-Sanahuja, "Counters-Based Modified Traffic Conditioner", Lecture Notes in Computer Science (QofIS'02), Vol. 2511, pp.57-67, 2002.

[12] F. Cerdan, O. Casals, "Performance of Different TCP Implementations over the GFR Service Category", ICON Journal, Special Issue on QoS Management in Wired & Wireless Multimedia Communications Network, Vol. 2, pp. 273-286, Baltzer Science.

[13] F. Cerdan, O. Casals, "Mapping an Internet Assured Service on the GFR ATM Service", Lecture Notes in Computer Science 1815 (Networking 2000), pp. 398-409, Springer-Verlag.

[14] V. Bonin, F. Cerdan, O. Casals, "A simulation study of Differential Buffer Allocation", Proceedings of 3rd International Conference on ATM, ICATM'2000, Germany, June 2000.

[15] Maria-Dolores Cano, Fernando Cerdan, Joan Garcia-Haro, Josemaria Malgosa-Sanahuja, "A new proposal for Assuring Services in Internet", Proceedings of the International Conference on Internet Computing IC'02, Vol. II pp. 379-384, Las Vegas, USA, June 2002.

[16] R. Jain, "The Art of Computer Systems Performance Analysis", John Wiley and Sons Inc., 1991.

[17] N. Seddigh, B. Nandy, P.Pieda, "Bandwidth Assurance Issues for TCP Flows in a Differentiated Services Networks", Proceedings of IEEE Globecom'99, Rio de Janeiro, Brazil, Vol. 3, pp. 1792-1798, December 1999.