

Penalty Shaper to Enforce Assured Service for TCP Flows

Emmanuel Lochin¹, Pascal Anelli², and Serge Fdida¹

¹ LIP6 - Université Paris 6, ² IREMI - Université de la Réunion, France
{emmanuel.lochin, serge.fdida}@lip6.fr, pascal.anelli@univ-reunion.fr

Abstract. Many studies have explored the TCP throughput problem in DiffServ networks. Several new marking schemes have been proposed in order to solve this problem. Even if these marking schemes give good results in the case of per-flow conditioning, they need complex measurements. In this paper we propose a Penalty Shaper (PS) which is able to profile a set of TCP flows so as to improve its conformance to a desired target. The main novelty of this shaper is that the shaping applies a penalty delay which depends on the out-profile losses in a DiffServ network. This penalty shaping can be used with any classic conditioner such as a token bucket marker (TBM) or a time sliding window marker (TSWM). We made an evaluation of the Penalty Shaper on a real testbed and showed that the proposed scheme is easily deployable and allows for a set of TCP flows to achieve its target rate.

Key words: Edge to Edge QoS, Assured Service, TCP.

1 Introduction

The Differentiated Services architecture [1] proposes a scalable means to deliver IP Quality of Service (QoS) based on handling of traffic aggregates. This architecture advocates packet tagging at the edge and lightweight forwarding routers in the core. Core devices perform only differentiated aggregate treatment based on the marking set by the edge devices. Edge devices in this architecture are responsible for ensuring that user traffic conforms to traffic profiles. The service called Assured Service (AS) built on top of the AF PHB is designed for elastic flows. The minimum assured throughput is given according to a negotiated profile with the user. Such traffic is generated by adaptive applications. The throughput increases as long as there are available resources and decreases when a congestion occurs. The throughput of these flows in the assured service breaks up into two parts. First, a fixed part that corresponds to a minimum assured throughput. The packets of this part are marked like inadequate for loss (colored green or marked IN). Second, an elastic part which corresponds to an opportunist flow of packets (colored red or marked OUT). These packets are conveyed by the network on the principle of "best-effort" (BE). In the event of congestion, they will be dropped first. Thanks to an Penalty Shaper, we show

that it is possible to provide service differentiation between two source domains, on a set of TCP flows, based on its marking profile. In this paper we evaluate the solution with long-lived TCP flows. The proposed solution provides the advantage of neither needing RTT(Round Trip Time) evaluation nor loss probability estimation. The solution takes care of the behavior of TCP flows only. Consequently, as it is easily deployable, it has been experimented on a real testbed. This paper is organized as follows. Section 2 presents related work. In section 2.1 we present the algorithm. Testbed and scenarios are presented in section 3. Section 4 presents the results obtained and their analysis. Finally, section 5 concludes the paper.

2 Related work

There have been a number of studies that focused on assured service for TCP flows but also on the aggregate TCP performance. In [2], five factors have been studied (RTT, number of flows, target rate, packet size, non responsive flows) and their impact has been evaluated in providing a predictable service for TCP flows. In an over-provisioned network, target rates are achieved regardless of these five factors. This result is corroborated by [3]. However the distribution of the excess bandwidth depends on these five factors. When responsive TCP flows and non-responsive UDP flows share the same class of service, there is unfair bandwidth distribution and TCP flow throughputs are affected. The fair allocation of excess bandwidth can be achieved by giving different treatment to out-of-profile traffic of two types of flows [3]. Recently [4] demonstrates the unfair allocation of out-of-profile traffic and concludes that the aggregate that has the smaller/larger target rate occupies more/less bandwidth than its fair-share regardless of the subscription level. In [5], a fair allocation of excess bandwidth has been proposed based on a traffic conditioner. The behavior of the traffic conditioner has a great impact on the service level, in terms of bandwidth, obtained by TCP flows. Several markers have been proposed to improve throughput insurance [6–9]. These algorithms propose to mark aggressive TCP flows severely out-of-profile so that they are preferentially dropped. Even if these marking strategies work well in simulation, their main disadvantage is their implementation complexity. Indeed, these algorithms need to measure a flow’s RTT, its loss probability, have a per-state information of the flows or require a complex signaling algorithm.

2.1 The Penalty Shaper (PS)

Let $r(i)_{AS}$ be the assured rate of the flow i (*i.e.* in-profile packets throughput), n the number of AS TCP flows in the aggregate at the bottleneck level and C the link capacity. Precisely, this capacity corresponds to a bottleneck link in the network. If a number of i flows cross this link, the total capacity allocated for assured service R_{AS} is : $\sum_{i=1}^n r(i)_{AS}$. Let C_{AS} be the resource allocated to the assured service.

$$R_{AS} < C_{AS} \tag{1}$$

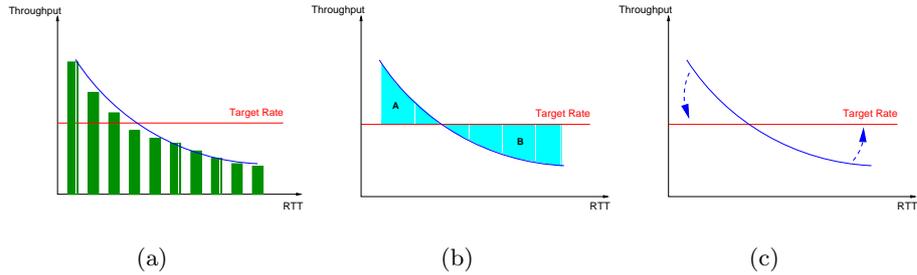


Fig. 1. TCP throughput of 10 TCP flows in function of the RTT

Equation (1) means an **under-subscription** network. In this case, there is excess bandwidth in the network. If $R_{AS} \geq C_{AS}$, this is an **over-subscription** network and there is no excess bandwidth. This configuration is the worst case for the AS. This service must provide an assurance until the over-subscription case is reached. Afterwards, not enough resources are available and the service is downgraded.

$$TCP \text{ Throughput} = \frac{C * \text{Maximum Segment Size}}{RTT * \sqrt{p}} \quad (2)$$

The preferential dropping taking place at the core routers provides a good indication of the state of congestion. If the network is far from being congested, the in-profile packets will rarely be dropped and their dropping probability will be insignificant. If the network is going to be congested, almost all of the out-profile packets will be dropped. In a well-dimensioned network, inequity from (1) should be respected. When there are losses in the network, it corresponds to the losses of out-profile packets, and not in-profile packets. It means that a light congestion appears in the network and some out-profile packets must be dropped. In order to increase the loss probability of the opportunist flows, new conditioners presented in section 2 are based on increasing the out-profile part of the most aggressive traffic. Then, the loss probability raises and the TCP throughput of the opportunist traffic decreases. It's a logical behaviour because the latter has a reject probability higher than the non-opportunist traffic. [10] gives a model of TCP throughput represented by the equation (2). With C a constant and p the loss probability. Changing the p value from the equation (2) thanks to a marking strategy is complex. Indeed, it is necessary to evaluate the loss probability of the network and estimate an RTT for each flow. In order to illustrate this point, figure 1 presents the principle of these marking strategies. Figure 1 (a) symbolizes the throughput obtained by ten flows with different RTTs. The smaller the RTT, the higher is the throughput. The flows with a small RTT occupies more bandwidth than necessary as shown by area A in figure 1 (b). The aim of a marking strategy is to distribute fairly excess bandwidth

from area A to area B. As a result, it corresponds to rotate the curve on figure 1 (c). As opposed to the marking strategy adopted by new conditioners, we propose a delay based shaper. This shaper applies a delay penalty to a flow if there are out-profile packets losses in the network and if it outperforms its target rate. The basic idea is that the penalty is a function of the out-profile packet losses. Instead of raising the p value, from equation (2), of the most opportunist flow, the Penalty Shaper raises a delay penalty to the flow. It results in a growth of the RTT. Mathematically, as shown in equation (2), increasing RTT value is similar to increasing p value in term of TCP throughput. [11] has shown that limiting out-profile packets is a good policy to achieve a target rate and a good solution to avoid TCP throughput fluctuations. Indeed, by avoiding packets dropping we avoid TCP retransmission. This is an efficient solution to optimize the bandwidth usage. That's the reason why we choose to operate on the raise of the RTT value instead of the raise of the p value. Moreover, we don't need complex measurements. Our goal is to reduce out-profile losses by applying a delay penalty to the flows that are the most opportunist in the network. Therefore, when a RIO¹ [12] router in the core network is dropping out-profile packets, it marks the ECN flag [13] of the in-profile packets enqueued in the RIO queue. In a well-dimensioned network, there is no in-profile packet loss. Then, the edge device can be aware that there is a minimum of one flow or set of flows which are opportunists in the network. This opportunist traffic is crossing the same path. The edge device evaluates its sending rate thanks to a Time Sliding Window (TSW) algorithm [14]. If its sending rate is higher than its target rate, it considers that its traffic may be opportunist. Then, it applies a penalty to the incoming traffic while the network feedback that there are out-profile packets losses. This penalty allows a raise of the RTT and consequently, decrease the TCP throughput. The algorithm presented in figure 2 explains how the penalty is calculated and applied. As explained on figure 4, once incoming TCP traffic is shaped, it passes through a marker such as a TBM. The PS mechanism could be placed on the client side rather than on the edge router. There isn't hypothesis on its localization. The PS is just used to enforce desired target rate and not used for marking traffic. In our simulation, the edge router uses a token bucket marker mechanism in order to mark in-profile and out-profile packets. Concerning the flows, traffic profile consists of a minimum throughput, characterized by two token bucket parameters, namely the token rate r and the size of the bucket b . The conformity control of an aggregate compared to the profile is thus done naturally by a token bucket as proposed in [15]. Figure 4 illustrates the ingress edge router mechanism.

The way TCP flows are conditioned by a marking strategy at the DiffServ network influences the drop probability of these TCP flows and consequently their behavior in the core network. Moreover, it is necessary to define a scalable conditioning in order to give an ISP² an exploitable solution. Many are the conditioners presented in section 2 which will never leave the framework of

¹ RED with IN and OUT

² Internet Service Provider

```

K = 10 ms
T = 1 sec
FOR each observation period T
  TSW gives an evaluation of the throughput : throughput_measured
  IF throughput_measured < target_rate OR there are no out-profile losses
  THEN reduce the penalty delay
    current_penalty = current_penalty - K
  ELSE raise the penalty delay
    current_penalty = current_penalty + K
  END IF
END FOR

```

Fig. 2. PS algorithm

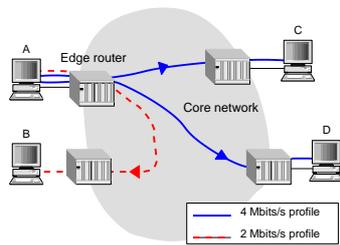


Fig. 3. Traffic conditioning sample

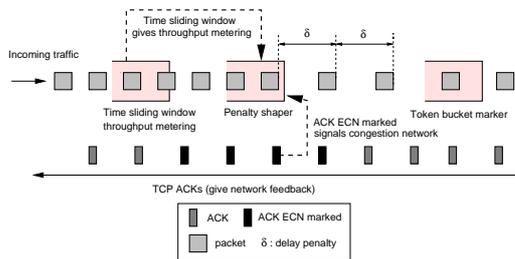


Fig. 4. Ingress edge router mechanism

simulation because of their conditioning constraints. In order to avoid confusion with the aggregate principle defined in [1], in the remainder of this paper, we call TCP client aggregates a set of TCP flows emitted from one source to one destination. These aggregates can have one or several TCP flows. We chose to make the traffic conditioning in the following way : each client emitting one or more flows towards one or more destinations will have one traffic profile per destination. As shown on figure 3, client A forces the edge router to setup three different traffic conditioners. Two conditioners with a profile rate of 4Mbits/s and one conditioner with a profile rate of 2Mbits/s . The main advantage of this solution is that the conditioning can be made on flows with similar RTTs (i.e. in the same order of magnitude). This solution doesn't depend on the complex problem of RTT estimation necessary to the functioning of the conditioners presented in section 2. The solution of traffic shaping coupled to a conditioner/marker such as the TBM should be easily deployable and scalable.

3 Experimental testbed

As shown in figure 5, we use the well-known dumbbell topology. This topology was used in many experimentations [7, 11, 16–18]. The choice of this platform can appear simple, but it characterizes any router being in a core network. Whether

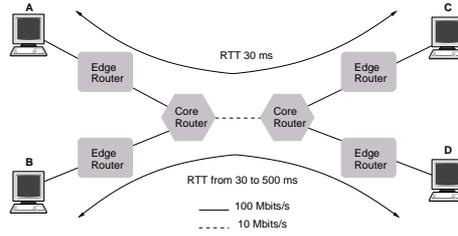


Fig. 5. Experimental testbed

it is on an edge router or on a core router, at the microscopic level, the behavior of each one is identical. At the macroscopic level, it is useless to serially chain several routers to carry out measurements because the end to end QoS will depend of the router under the worst conditions of traffic with regard to its output rate. Behind this router the flows are smoothed. Thus, a testbed made by a single router to build a bottleneck is sufficient to make the evaluation of the services proposed. The objective of this platform consist in evaluating the most significant QoS deviation. The testbed is composed of computers running Free-BSD. On the edge routers, the token bucket marker (TBM) from ALTQ³ development and the Penalty Shaper based on Dummynet⁴ On the core routers, a RIO queue, developed in ALTQ. We have added the marking functionality. Thus, the RIO queue is able to mark the ECN flag of the in-profile packets if it detects out-profile losses in its queue. Finally, we use two transmitting machines and two receivers for measurements. The acknowledgement management is important for reaching the targeted performance for TCP flows in DiffServ. The full-duplex link of the testbed allows forgetting TCP acknowledgment management. Indeed, no congestion occurs in the reverse path. The main parameters and hypothesis are : traffic generation is carried out in the following way: A to C (A, C) and B to D (B, D), after 120 seconds, `Iperf`⁵ gives an average throughput of the flow ; each AS flow is transmitted as TCP, packets have a size of 1024 bytes ; `Iperf` uses a TCP maximum window size $W_{max} = 64 \text{ packets}$; each set of flows between two hosts is conditioned by one TBM with or without PS ; b parameter of the TBM is set to one packet ; r parameter is set to the desired target rate ; the delay penalty is set to 10 ms and the observation period to 1 sec . It means that each second the algorithm gives an estimation of the throughput and evaluates the penalty delay ; we use a non-overlapping RIO with parameters : $(min_{out}, max_{out}, p_{out}, min_{in}, max_{in}, p_{in}) = (1, 63, 0.1, 64, 128, 0.02)$, the queue size corresponds to $2 * W_{max}$; we repeat each experiments five times and calculate the average throughput value (standard deviation is $\pm 0.15 \text{ Mbits/s}$).

³ <http://www.csl.sony.co.jp/person/kjc/>

⁴ http://info.iet.unipi.it/~luigi/ip_dummynet/

⁵ <http://dast.nlanr.net/Projects/Iperf/>

4 Performance evaluation of the Penalty Shaper

This section presents the results obtained in a real testbed with the PS. We evaluate the performance of the PS when client emitted TCP aggregate have the same or a different number of flows and identical or different RTTs. In all under-subscribed network scenarios : the total capacity allocated for the assured service is $R_{AS} = 8Mbits/s$ and the provisioned resource to the assured service is $C_{AS} = 10Mbits/s$ (that corresponds to the bottleneck capacity). So, there are $2Mbits/s$ of excess bandwidth.

4.1 Impact of the aggregates' aggressiveness in an under-subscribed network

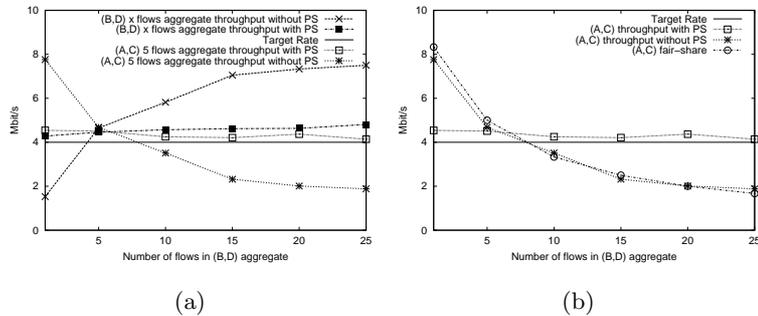


Fig. 6. TCP throughput versus aggregates' aggressiveness

The main feature of the solution is : even if there is a different number of flows in the aggregates, the PS is able to reach its target rate. Results are presented in figure 6. When two aggregates with different number of flows are in a network, the higher outperforms the smaller. This aggregate aggressiveness problem was first raised in [2]. In these tests, two aggregates are in competition and the RTT of both aggregates is set to $30ms$. The (A,C) aggregate has a fixed number of 5 flows and the (B,D) aggregate has a variable number of flows ranging from 1 to 25. The target rate of both aggregates is set to $r(A,C)_{AS} = r(B,D)_{AS} = 4Mbits/s$. When (B,D) has less/more than 5 flows, (A,C) is the most/less aggressive aggregate. Figure 6 (a) shows the throughput obtained by both aggregates. The PS is able to reach the desired target rate. For clarification, we draw on figure 6 (b) the throughput obtained by the (A,C) aggregate alongside the fair-share curve. This figure shows that the TBM stays close to the fair-share while the PS is near the the desired target rate.

4.2 Impact of the aggregate's size on a reference flow in an under-subscribed network

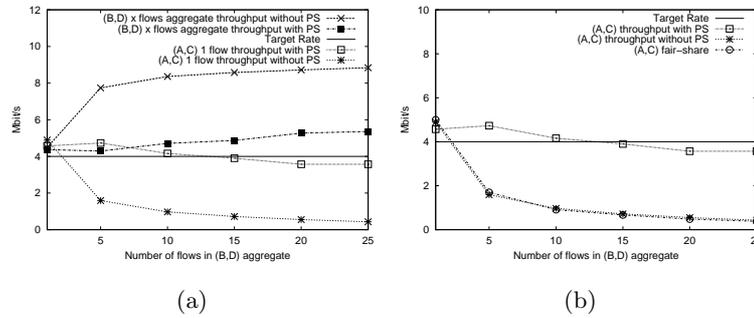


Fig. 7. TCP throughput of one flow versus an aggregate

We use the same test conditions as previously but (A,C) is characterized by one flow only. We observe in this test the isolation of one reference flow in competition with an increasing number of microflows in the (B,D) aggregate. Figure 7 (a) shows that the PS obtains a throughput close to the desired target rate while the TBM stays always close to the fair-share curve as shown on figure 7 (b).

4.3 Impact of the target rate in an under-subscribed network

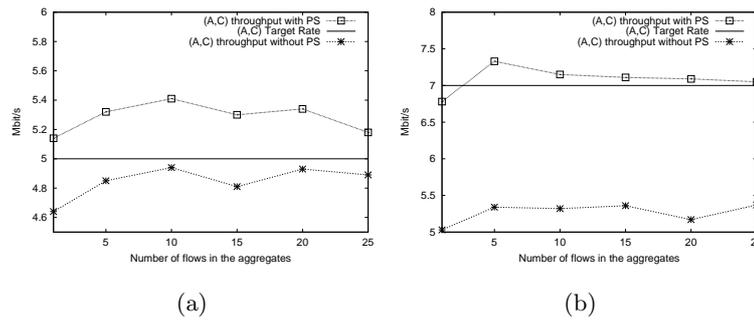


Fig. 8. TCP throughput with various target rates

| Test | # flows aggregate (A, C) versus (B, D) | A to C RTT=30ms | B to D RTT=30ms | A to C RTT=30ms | B to D RTT=30ms |
|----------|---|--------------------|--------------------|--------------------|--------------------|
| TBM only | 1 vs 1 | 4.64/5 | 4.29/3 | 5.03/7 | 4.36/1 |
| TBM+PS | 1 vs 1 | 5.14/5 | 3.61/3 | 6.78/7 | 1.61/1 |
| TBM only | 5 vs 5 | 4.85/5 | 4.56/3 | 5.34/7 | 4.05/1 |
| TBM+PS | 5 vs 5 | 5.32/5 | 3.39/3 | 7.33/7 | 1.70/1 |
| TBM only | 10 vs 10 | 4.95/5 | 4.38/3 | 5.32/7 | 4.26/1 |
| TBM+PS | 10 vs 10 | 5.41/5 | 3.47/3 | 7.15/7 | 1.64/1 |
| TBM only | 15 vs 15 | 4.81/5 | 4.48/3 | 5.36/7 | 4.23/1 |
| TBM+PS | 15 vs 15 | 5.30/5 | 3.57/3 | 7.11/7 | 1.94/1 |
| TBM only | 20 vs 20 | 4.93/5 | 4.41/3 | 5.17/7 | 3.86/1 |
| TBM+PS | 20 vs 20 | 5.34/5 | 3.41/3 | 7.09/7 | 1.92/1 |
| TBM only | 25 vs 25 | 4.89/5 | 4.38/3 | 5.37/7 | 4.37/1 |
| TBM+PS | 25 vs 25 | 5.18/5 | 3.69/3 | 7.05/7 | 1.92/1 |

Table 1. Under-subscribed network (caption : goodput in $Mbits/s$ / target rate in $Mbits/s$)

In this section, two aggregates are in competition and the RTT of both aggregates is set to $30ms$. Both aggregates have the same number of flows. The number of flows varies from 1 to 25. Table 1 presents the results obtained in two scenarios. In the first one, (A, C) and (B, D) have respectively a target rate of $r(A, C)_{AS} = 5Mbits/s$ and $r(B, D)_{AS} = 3Mbits/s$ while in the second one, they have respectively a target rate of $r(A, C)_{AS} = 7Mbits/s$ and $r(B, D)_{AS} = 1Mbits/s$. So, the two scenarios illustrate both the case where the aggregates have near or distant target rates under under-subscription conditions. In figure 8 (a), we draw the throughput obtained by aggregate (A, C) when it requests a target rate of $r(A, C)_{AS} = 5Mbits/s$ (it corresponds to column 3 in table 1) and figure 8 (b) shows the throughput obtained by aggregate (A, C) when it requests a target rate of $r(A, C)_{AS} = 7Mbits/s$ (it corresponds to column 5 in table 1). All the aggregates reach their target rate even if the number of flows in the aggregate increases.

4.4 Impact of the RTT in an under-subscribed network

Even if there is a high number of flows in the aggregate and a high RTT difference, the PS is able to reach the target rate requested by an aggregate. Figure 9 illustrates this case. Figure 9 (a) shows the throughput of the (B, D) aggregate with an $RTT = 100ms$ versus an (A, C) aggregate with an $RTT = 30ms$, function of the number of flows. We focus on (B, D) aggregate throughput because in this scenario, (A, C) aggregate always reaches its target rate with or without the PS. The target rate for (A, C) and (B, D) is $r(A, C)_{AS} = r(B, D)_{AS} = 4Mbits/s$. Thanks to the Penalty Shaper, the (B, D) aggregate reaches the target rate and the increase of the number of flows in the aggregate doesn't influence the desired target rate. Finally, figure 9 (b) shows the throughput of a 10 flows aggregate (B, D) in competition with a 10 flows aggregate (A, C) . For the (A, C) aggregate, the RTT is equal to $30ms$ and for the (B, D) aggregate, we increase gradually the RTT from $30ms$ to $500ms$. It appears that the aggregate reaches the target rate when it is feasible (i.e. when target rate : $r(B, D)_{AS} > Wmax/RTT$). For information purposes, we did the same test

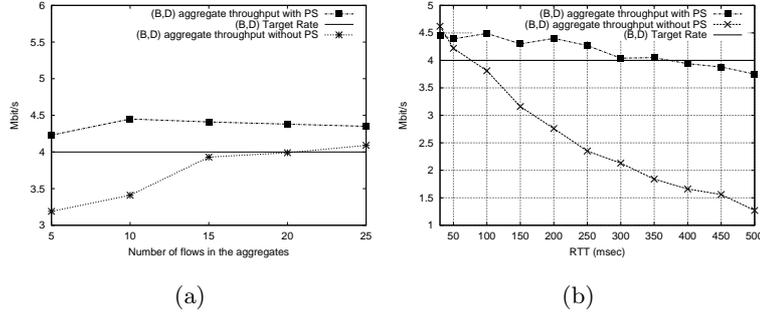


Fig. 9. TCP throughput versus RTT

with a number of flows ranging from 5 to 25 in both aggregates and obtained similar results. We present on figure 9 (b) only 10 versus 10 flows for space reasons.

4.5 A case study

| Test | (B, D)'s RTT | TCP (B, D) | TCP #1 (A, C) | TCP #2 (A, C) | TCP #3 (A, C) | UDP (A, C) |
|----------|--------------|------------|---------------|---------------|---------------|------------|
| TBM only | 50ms | 1.08/4 | 2.46/1 | 2.38/1 | 2.43/1 | 1.05/1 |
| TBM+PS | 50ms | 3.71/4 | 1.40/1 | 1.41/1 | 1.43/1 | 1.05/1 |
| TBM only | 100ms | 0.92/4 | 2.37/1 | 2.48/1 | 2.55/1 | 1.05/1 |
| TBM+PS | 100ms | 3.65/4 | 1.36/1 | 1.35/1 | 1.42/1 | 1.05/1 |

Table 2. Under-subscribed network (caption : goodput in $Mbits/s$ / target rate in $Mbits/s$)

In this part, we look at one (B, D) aggregate with five microflows in competition with three (A, C) aggregates constituted by ten microflows and one (A, C) UDP flow of $1Mbit/s$. Each (A, C) aggregate requests a target rate of $1Mbits/s$ and has an $RTT = 30ms$. Table 2 gives the results obtained when the (B, D) aggregate has an RTT equal to $50ms$ or $100ms$ and requests a target rate of $4Mbit/s$. The (B, D) aggregate is in the worst conditions to reach its requested target rate but we can see that thanks to the PS, it obtains a TCP throughput near the target rate.

4.6 Impact of the number of flows in an over-subscribed network

In both scenarios in table 3, the total capacity allocated to the assured service is $R_{AS} = 12Mbits/s$. There is no excess bandwidth, the network is over-subscribed and there are several in-profile packets losses. Measurements in table 3 show that

the PS allows to reach a TCP throughput closer to the target rate than a TBM. We don't discuss this case because we consider that it corresponds to a badly dimensioned service. Moreover, the PS algorithm, due to in-profile losses, is not really designed for this case.

| Test | # flows (A, C) versus (B, D) | A to C | B to D | A to C | B to D |
|----------|---------------------------------|----------|----------|----------|----------|
| | | RTT=30ms | RTT=30ms | RTT=30ms | RTT=30ms |
| TBM only | 1 vs 1 | 4.46/8 | 4.50/4 | 4.94/10 | 4.31/2 |
| TBM+PS | 1 vs 1 | 4.69/8 | 4.33/4 | 6.40/10 | 2.31/2 |
| TBM only | 5 vs 5 | 5.02/8 | 4.34/4 | 5.25/10 | 4.05/2 |
| TBM+PS | 5 vs 5 | 5.08/8 | 4.21/4 | 7.01/10 | 2.09/2 |
| TBM only | 10 vs 10 | 5.08/8 | 4.25/4 | 5.24/10 | 4.06/2 |
| TBM+PS | 10 vs 10 | 5.38/8 | 3.98/4 | 7.20/10 | 2.07/2 |
| TBM only | 15 vs 15 | 5.02/8 | 4.31/4 | 5.27/10 | 4.08/2 |
| TBM+PS | 15 vs 15 | 5.53/8 | 3.74/4 | 7.32/10 | 2.05/2 |
| TBM only | 20 vs 20 | 5.05/8 | 4.36/4 | 5.43/10 | 3.84/2 |
| TBM+PS | 20 vs 20 | 5.51/8 | 3.73/4 | 7.29/10 | 2.09/2 |
| TBM only | 25 vs 25 | 4.90/8 | 4.35/4 | 5.29/10 | 3.95/2 |
| TBM+PS | 25 vs 25 | 5.49/8 | 3.79/4 | 7.34/10 | 2.04/2 |

Table 3. Over-subscribed network (caption : goodput in *Mbits/s* / target rate in *Mbits/s*)

5 Conclusion and future works

In this paper, we have studied on a real testbed a Penalty Shaper (PS) which provides throughput assurance between TCP flows. This is the first proposal that use a delay penalty which depends on the out-profile losses in a DiffServ network. The number of flows in the aggregate can influence the targeted throughput and balance the disadvantage of a long RTT flow or a difficult target rate. The main consequence of these measurements is that we are able to obtain the guaranteed throughput if the profiled TCP aggregates in competition have the same or different number of flows. This is true whatever the differences between their RTTs and their target rates. In case of over-subscribed network, the PS is able to reach a throughput closer to the target rate than a simple token bucket marker. However, the PS is not designed for this type of network that corresponding to a badly dimensioned service. The proposed solution has the advantage of being easily deployable because it doesn't require complex measurements. The solution is scalable (it works at the edge of the network and is able to conditioning one or several flows) and being likely to be used with the most frequently used conditioners such as token bucket marker or time sliding window marker.

In our measurements, all the edge routers have setup the PS and all the core routers have a RIO ECN-capable queue. In a next work, we will analyse the case (with a large topology) where all the routers have not implemented our mechanism. Second, the penalty value is another critical issue. In our algorithm, this penalty is arbitrarily defined. We work on a penalty which follows the AIMD principle of TCP. So the penalty will be automatically calculated. We are cur-

rently deploying this proposal on a real large scale testbed with various traffic such long-lived and short-lived TCP flows.

References

1. Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An architecture for differentiated services. Request For Comments 2475, IETF (1998)
2. Seddigh, N., Nandy, B., Piedad, P.: Bandwidth assurance issues for TCP flows in a differentiated services network. In: Proc. of IEEE GLOBECOM, Rio De Janeiro, Brazil (1999) 6
3. Goyal, M., Durresi, A., Jain, R., Liu, C.: Effect of number of drop precedences in assured forwarding. In: Proc. of IEEE GLOBECOM. (1999) 188–193
4. Park, E.C., Choi, C.H.: Proportional bandwidth allocation in diffserv networks. In: Proc. of IEEE INFOCOM, Hong Kong (2004)
5. Alves, I., de Rezende, J.F., de Moraes, L.F.: Evaluating fairness in aggregated traffic marking. In: Proc. of IEEE GLOBECOM. (2000)
6. Kumar, K., Ananda, A., Jacob, L.: A memory based approach for a TCP-friendly traffic conditioner in diffserv networks. In: Proc. of the IEEE International Conference on Network Protocols - ICNP, Riverside, California, USA (2001)
7. El-Gendy, M., Shin, K.: Assured forwarding fairness using equation-based packet marking and packet separation. *Computer Networks* **41** (2002) 435–450
8. Feroz, A., Rao, A., Kalyanaraman, S.: A TCP-friendly traffic marker for IP differentiated services. In: Proc. of IEEE/IFIP International Workshop on Quality of Service - IWQoS. (2000)
9. Habib, A., Bhargava, B., Fahmy, S.: A round trip time and time-out aware traffic conditioner for differentiated services networks. In: Proc. of the IEEE International Conference on Communications - ICC, New-York, USA (2002)
10. Floyd, S., Fall, K.: Promoting the use of end-to-end congestion control in the Internet. *IEEE/ACM Transactions on Networking* **7** (1999) 458–472
11. Yeom, I., Reddy, N.: Realizing throughput guarantees in a differentiated services network. In: Proc. of IEEE International Conference on Multimedia Computing and Systems- ICMCS. Volume 2., Florence, Italy (1999) 372–376
12. Clark, D., Fang, W.: Explicit allocation of best effort packet delivery service. *IEEE/ACM Transactions on Networking* **6** (1998) 362–373
13. Ramakrishnan, K., Floyd, S., Black, D.: The addition of explicit congestion notification (ECN) to ip. Request For Comments 3168, IETF (2001)
14. Fang, W., Seddigh, N., AL.: A time sliding window three colour marker. Request For Comments 2859, IETF (2000)
15. Heinanen, J., Guerin, R.: A single rate three color marker. Request For Comments 2697, IETF (1999)
16. Mellia, M., Stoica, I., Zhang, H.: TCP-aware packet marking in networks with diffserv support. *Computer Networks* **42** (2003) 81–100
17. Nandy, B., P.Piedad, Ethridge, J.: Intelligent traffic conditioners for assured forwarding based differentiated services networks. In: IFIP High Performance Networking, Paris, France (2000)
18. Wang, F., Mohapatra, P., Bushmitch, D.: A random early demotion and promotion marker for assured services. *IEEE Journal on Selected Areas in Communications* **18** (2000)