A new auto-provisioned squat-based traffic management strategy for multiclass networks[1]

Xavier Hesselbach*, *Member, IEEE*, and Joan Antoni García-Espín**
**Technical University of Catalonia (UPC), Jordi Girona 1-3, 08034 Barcelona, Spain*
Tel: +34 934015987, e-mail: xavier.hesselbach@entel.upc.edu
*** i2CAT Foundation, C/Gran Capità, s/n, 08034 Barcelona, Spain*
Tel: +34 935532515, e-mail: joan.antoni.garcia@i2cat.net

Keywords: Multiclass, auto-provision, squat, kick out, effective bandwidth.

1. Introduction

In search of being able to offer better quality of service (QoS) in multiclass networks, a strategy that aims to allow resources between different classes of service (CoS) to be shared according to the class's needs and priority is proposed. The goal is to achieve an optimal use of the total bandwidth in a link. This paper presents shortly the proposed strategy and explains the "squatting" and "kicking" mechanisms. A model and simple results for performance to prove their utility are shown.

In a technical plane, Multi-Protocol Label Switching (MPLS) is considered nowadays one of the mechanisms that better performs the network service convergence of voice, video and data, required in most UMTS/HSDPA networks. The main reason is its inherent traffic engineering (TE) capabilities, which foster easy class of service tagging, traffic prioritization and resource optimization. This is especially important due to the fact that operators worldwide typically use four basic classes of traffic in their backbones: Conversational, streaming, interactive and background, each of them requiring different quality of service (QoS) parameters. IP Differentiated Services (DiffServ) [1] over the MPLS protocol stack have been proposed in many technical studies for guaranteeing IP-class QoS levels and keeping network utilisation at maximum. However, MPLS network functionalities have to be enhanced to fully support DiffServ, together with automatic, class-based network service provisioning.

Based in this scenario, this work is focused in a novel technique for idle resource squatting among classes, where low priority classes of traffic can utilize resources allocated to higher priority ones when remain unused. Similarly, a technique for letting high priority classes kick lower priority ones out of their allocated resources is described. In particular, the study has been carried out splitting the available bandwidth in a link among a set of classes of traffic coming from IP-DiffServ network. Other works [2], [3], have recently studied some traffic management algorithms under DiffServ, but the scheme here proposed enhances the per-link total bandwidth utilization.

Thus, the resources that are squatted or kicked out are bandwidth allocations. At this point, bandwidth is monitored in two stages: (i) the original bandwidth allocated per class –typically by means of a resource reservation protocol like RSVP- and (ii) the effective bandwidth being used by a class of traffic at a given time. It must be noted that the latter can be higher than the first of the observed class of traffic has squatted resources (bandwidth from higher priority classes).

2. Simple model and results

Let's shortly model the *squatting* strategy. Latter, following the same way, we will model the *kicking* mechanism. Considering N classes of service, where 1 is the maximum priority and N the minimum priority, a link with a bandwidth normalized to 1, and an allocated bandwidth per class 1/N, let's assume that all the classes increment the need for bandwidth linearly with a slope s, until a maximum $s{\cdot}max$, where $s{\cdot}\max \le 1/N$, except the class N (minimum priority) that will increment the demand without limitation.

We denote G_i the demanded bandwidth of class i, and S_i the permitted bandwidth (even considering squatted traffic):

$$\forall i \neq N, \qquad S_i = \begin{Bmatrix} s \cdot x & ; x \le max \\ s \cdot max & ; x > max \end{Bmatrix}$$

The normalized bandwidth available to the class N for squatting is $\left(\frac{1}{N} - s \cdot max\right) \cdot (N-1)$. Therefore, the total effective bandwidth usable to the class N is the owned plus the squatted:

[1] This work has been partially supported by the national Spanish project CICYT TSI2007-66637-C02 and the European NoE Euro-NF 2008-2010.

$$\frac{1}{N} + \left(\frac{1}{N} - s \cdot max\right) \cdot (N-1) = 1 - s \cdot max \cdot (N-1)$$

$$S_N = \begin{Bmatrix} s \cdot x & ; x \le \frac{1}{s} - max \cdot (N-1) \\ 1 - s \cdot max \cdot (N-1) & ; x > \frac{1}{s} - max \cdot (N-1) \end{Bmatrix}$$

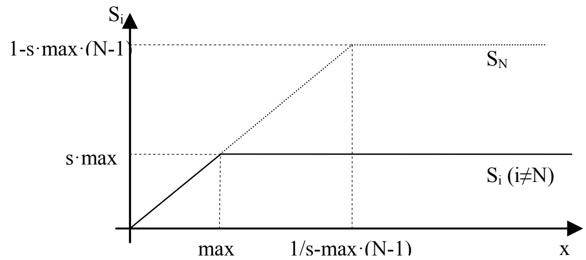

Figure 1. Per-class allocable bandwidth (normalized) using the squatting strategy.

The figure 1 shows that the class N can use an effective bandwidth much greater than the bandwidth allocated in the regular DiffServ behaviour. In general, this result can be extended to any class where higher priorities are running, but just counting the idle bandwidth in the higher classes.

Due to space limitations, we will leave the model details for the kicking strategy and its consequences for other work. However, it can be easily extended from the squatting that:

Assuming that all the classes are incrementing the bandwidth in the same way, linearly up to a value l, except the highest priority, the total normalized bandwidth for kicking is $\left(\frac{1}{N} - s \cdot l\right) \cdot (N-1)$ and, therefore, the total effective bandwidth usable to the class 1 is the owned plus the kicked out:

$$\frac{1}{N} + \left(\frac{1}{N} - s \cdot l\right) \cdot (N-1) = 1 - s \cdot l \cdot (N-1)$$

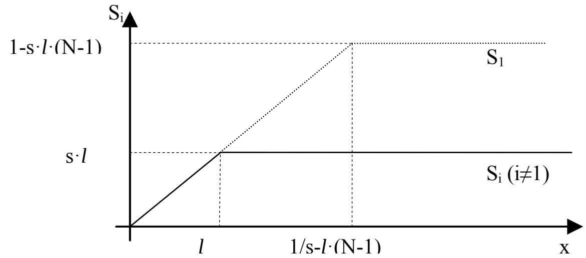

Figure 2. Per-class allocable bandwidth (normalized) using the kicking strategy.

As a summary, we can conclude that the class N can use the remaining bandwidth from other classes by means of squatting, and therefore, the effective bandwidth for this class is greater than the originally allocated. At the other side, using the kicking mechanism, analogue conclusion can be derived, but the benefits are now for the maximum priority class.

For the future works, thresholds in squatting and kicking should be defined to avoid resources beat down for any class and to guarantee a minimum always-reserved bandwidth. Also, consecutive squatting using several classes should be studied, and convergence and stabilization time must be analyzed. Additionally, the strategy when the owner of a squatted or kicked resource requires the bandwidth should be studied.

3. References

[1] Blake,S; Black. “An Architecture for Differentiated Services” RFC 2475. The Internet Society, 1998.

[2] Veres, S; Ionescu, Dan, Senior Member IEEE. “A Performance Model and Measurement Framework for DiffServ Implementations” IEEE Trans. on Instrumentation and Measurement, vol.56, issue 4, pp.1473–1480,august 2007.

[3] An Approach of End- to-End Diffserv/MPLS QoS Context Transfer in HMIPv6 Networks . Liu, Chuda, et al. Autonomous Decentralized Systems, 2007. IEEE ISADS '07. Page(s):245 - 254, 21-23 March 2007.