

Traffic conditioner: upper bound for the spacer overflow probability

Nathalie OMNÈS^{a,*},

^a*Mitsubishi Electric ITE TCL, 1, allée de Beaulieu, CS 10 806, 35 708 Rennes cedex 7, France*

Annie GRAVEY^b,

^b*Département informatique, ENST Bretagne, Technopole de Brest Iroise, B.P. 832, 28 285 Brest Cedex, France*

Raymond MARIE^c

^c*Equipe ARMOR, IRISA, Campus universitaire de Beaulieu, 35 042 Rennes Cedex, France*

Abstract

The current challenge for telecommunication networks is to offer new types of services. To achieve this, somehow traffic has to be conditioned. A spacer is a token bucket traffic conditioner that reshapes flows according to a periodic profile. Packets are delayed until they can be inserted according to this profile.

This short paper concentrates on packet discard by overflow in the spacer. An upper bound for the overflow probability is given, valid even when it is fully loaded. It shows the ratio of discarded packets is of the same order than the ratio of out-of-profile packets.

Key words: Traffic conditioning, leaky bucket, analytical model, upper bound, loss probability.

* Corresponding author.

Email addresses: omnes@tcl.ite.mee.com (Nathalie OMNÈS), annie.gravey@enst-bretagne.fr (Annie GRAVEY), Raymond.Marie@irisa.fr (Raymond MARIE).

1 Introduction

Although Internet Protocol (IP) is currently widely used, it does not provide a straightforward solution to the applications' needs. Hence it does not permit to offer all the quality of service guarantees on a large scale, and in particular to real-time services such as distributed games. To overcome this problem, more services should appear in the future.

The *IETF* (Internet Engineering Task Force) has standardized in *RFC 2475* (see [7]) an architecture for *DiffServ* (Differentiated Services). For the sake of scalability, several types of nodes are distinguished. *DiffServ* ingress and egress nodes are located on the boundary of a *DiffServ* domain, while *DiffServ* interior nodes only implement simple functionalities. This is achieved by marking packets in the ingress node with a *DiffServ* codepoint. Complexity is thus pushed in border routers, which are responsible for offering several types of services.

A type of service specifies some significant characteristics of packet transmission, which may include throughput, delay, jitter and loss. To control these characteristics, the network somehow needs to control incoming flows, which is achieved by traffic conditioning functions in ingress and egress nodes.

A traffic conditioner measures the temporal properties of a stream of packets to determine whether each packet is in-profile or out-of-profile. Token buckets are commonly used for identifying out-of-profile packets (see [1], [2, subsection 5.3.3], [7, subsection 2.3.2]). They allow out-of-profile packets to be either queued until they are in-profile, or discarded, or marked with a new codepoint, or forwarded unchanged while triggering some accounting procedures. Generally speaking, token buckets allow burstiness in the output stream, up to a given maximum burst size.

A leaky bucket can be seen as a particular token bucket that enforces a rigid output pattern, allowing no burstiness in the output stream (see [2, subsection 5.3.3]). It needs two parameters to identify a constant bit rate source : r , the peak rate, and b , the bucket depth. The peak rate is not sufficient for identifying out-of-profile packets as data go through several layers to reach the physical one, as well as through several network devices. Unless these devices are synchronous, periodicity gets corrupted by them.

A spacer is a token bucket whose output is periodic until no more packets are available, thus allowing no burstiness. In [1, corollary 1.2.1 p.16], it is shown that spacers and leaky buckets are equivalent in case of constant size packets. To shape the outgoing flow, packets in advance are delayed until they can be inserted according to a periodic profile. Consequently late packets do not suffer from any further delay. It has been proved in [1, section 1.5.3] that

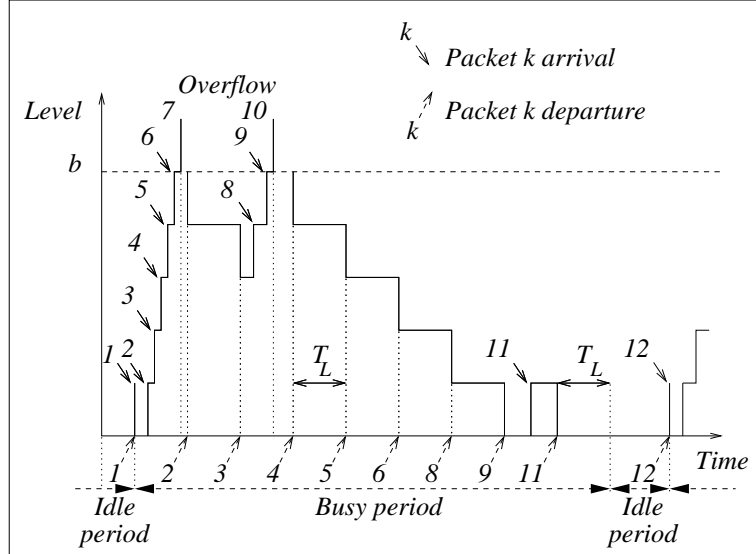


Fig. 1. Spacer occupancy level vs. time

spacing does not increase delay nor bandwidth requirements. It does not even increase buffer requirements since interior nodes only need to implement very small buffers. Furthermore, limiting the flow to the a given rate at the access facilitates management in the core network, which is a key issue (see [2,4–6]).

Let us consider Figure 1. Each step upwards correspond to a packet arrival, while each step downwards to a departure. Packets 2 to 6 are in advance, so they are delayed in the spacer. On the contrary, packet 12 is late, so it is retransmitted as soon as it enters the spacer, and it induces an idle period. Such a period begins when the spacer is empty and waiting for a new packet to transmit. It ends when a new packet arrives. We see that the output traffic is shaped.

The spacer does not need to absorb all bursts, as they correspond to packets that have suffered from a long delay in the network devices they crossed. The larger the burst is, the greater the delay perceived by the receiver. Therefore the spacer lops long bursts by discarding packets when they arrive as the spacer is full. This is represented in Fig. 1, where packets 7 and 10 are lost by overflow.

The spacer cannot be reduced to a queuing analysis in a straightforward manner as it consists of a $G/D/1$ system, which we aim at studying even if the load $\rho = 1$. This is a particularly delicate point. It cannot either be analyzed as a "(max,+)" system unless we give up the random behavior of the core network, which would take off an essential property.

Our aim in this article is to study the impact of the spacer capacity on the overflow probability (see [3,4]). To do so, we state working assumptions and define the variables that are central to our modelling in section 2. We then

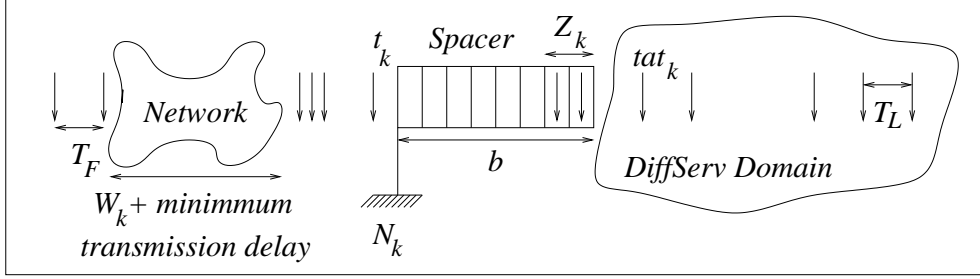


Fig. 2. Variables

study the overflow probability in section 3, where our main result is an upper bound presented in proposition 3. We also prove the upper bound in a deterministic context. We finally conclude in section 4. Main proofs and intermediate lemmas are presented in the appendix.

2 Assumptions and formal description

The assumptions made all along this paper are the following :

- i) Packets are of constant size.*
- ii) Packets arrive in sequence.*
- iii) The spacer is empty before packet 1 arrives.*
- iv) All packet losses are due to overflow.*
- v) The spacer load is smaller than or equal to 1.*

We assume that the packets are of constant size for the sake of simplicity. This is the case for telephony over *IP* (Internet Protocol) and it also corresponds to the case of an *ATM* (Asynchronous Transfer Mode) network (see [2]).

Let us consider Figure 2. T_F is a constant defined as the interval between the generation of two consecutive packets at the source level. As packets are of constant size, the parameters r (the peak rate) and b (the bucket depth) of the leaky bucket can be replaced by T_F and τ_F , the delay variation tolerance, also called the jitter (see [1, corollary 1.2.1 p.16], [2, p. 466-467]). If S denotes the packet size in bits, we have $T_F = S/r$. Furthermore, it is assumed that the interval between arrivals of two consecutive packets into the spacer lies between $T_F - \tau_F$ and $T_F + \tau_F$.

Let tat_k be packet k 's theoretical arrival time in the spacer, t_k its arrival time, and T_L the interdeparture time of packets from the spacer. The sequence of theoretical arrival times is used to generate the output of the spacer. Actually, packet k is taken out of the spacer when tat_k is reached. Furthermore, the spacer is in an idle period at time t if and only if the time elapsed since the last packet to enter has left is greater than or equal to T_L . Note that tat_k is

Table 1

Basic notations

tat_k	packet k 's theoretical arrival time in the spacer
t_k	packet k 's arrival time in the spacer
b	capacity of the spacer
T_F	interval between generation of two consecutive packets at source level
τ_F	delay variation tolerance of the flow entering the spacer
T_L	interdeparture time of packets from the spacer
Z_k	occupancy level of the spacer at t_k^-
W_k	packet k 's waiting time in the network.
N_k	Number of packets that were lost by overflow before packet k 's arrival.

moved forward when a packet arrives late, and backward when a packet is lost. It is computed using the following rule :

$$tat_k = \begin{cases} t_1 & \text{for } k = 1 \\ tat_{k-1} & \text{if packet } k-1 \text{ is lost and } t_k \leq tat_{k-1} \\ tat_{k-1} + T_L & \text{if packet } k-1 \text{ is not lost and } t_k \leq tat_{k-1} + T_L \\ t_k & \text{otherwise.} \end{cases} \quad (1)$$

Furthermore, packet k is conforming to parameters (T_F, τ_F) (or in-profile) if and only if

$$tat_k - t_k \leq \tau_F. \quad (2)$$

As we assume that the spacer load is smaller than or equal to 1, we have $T_L \leq T_F$. If $T_L = T_F$ then the spacer is fully loaded.

We further denote the spacer occupancy level just before packet k 's arrival by Z_k , the waiting time packet k is subject to in the network by W_k , and the number of packets that were lost by overflow before packet k 's arrival by N_k . W_k is defined as the difference between the observed and the minimum transmission delay in the network. Due to the periodicity of the initial stream, we have $t_k = t_1 - W_1 + (k-1)T_F + W_k$, $\forall k \geq 1$. All these basic notations are depicted in Table 1 and Figure 2.

3 Study of the overflow probability

With the help of the variables defined in section 2, we are now going to state necessary conditions for overflow, give a formal expression for the sequence of theoretical arrival times, and finally study the probability for a packet to be lost in the spacer.

3.1 Preliminaries

Packet k is lost only if it reaches the spacer in advance. In particular, it implies it reaches the spacer before a particular packet, denoted by $P(k)$, leaves the spacer. Therefore we have

$$(Z_k = b) \implies (t_k < \text{tat}_{P(k)}). \quad (3)$$

Proof Let CT_i denote the i^{th} packet that is not lost by overflow in the spacer. If $Z_k = b$, let J be a random variable taking its value in \mathbb{N} such that $\text{tat}_{CT_{J-1}} \leq t_k < \text{tat}_{CT_J}$. At t_k^- , $k-1$ packets arrived, N_k of which overflowed. Furthermore, if $\text{tat}_{CT_{J-1}} \leq t_k < \text{tat}_{CT_J}$, then $J-1$ packets have left. Therefore there are $k-1-N_k-J+1$ packets in the spacer. This leads to the following condition on J : $Z_k = k - N_k - J = b \implies J = k - N_k - b$. In particular, $P(k) = CT_{k-N_k-b}$. \square

Equation 3 depends on the sequence $(\text{tat}_k)_{k \geq 1}$, which needs to be expressed in a more tractable way than equation 1. To achieve this, we firstly need to introduce DA_k , which quantifies the theoretical arrival times shift due to the idle period that occurs just before packet k 's arrival, if any. We secondly introduce δ_k , which equals 1 when packet k arrives in a idle period and 0 otherwise.

Proposition 1 $\forall k \geq 2$, if packet k does not overflow, then $\text{tat}_k = t_1 + (k-1)T_L + \sum_{j=2}^k \delta_j DA_j - N_k T_L$. Packet k arrives in an idle period if and only if $\delta_k = 1$.

The proof which implies the introduction of further variables is presented in the appendix 5.1. The proposition shows that $\sum_{j=2}^k \delta_j DA_j$ is the total duration of the idle periods that occurred before packet k 's arrival. Hence $\sum_{j=2}^k \delta_j DA_j$ represents the theoretical arrival times shift (delay) due to idle periods occurring between packet 1 and packet k arrivals, while $N_k T_L$ is the theoretical arrival times shift (advance) due to packets lost by overflow in the spacer. We

can sum up this into $X_k = \sum_{j=2}^k \delta_j DA_j - N_k T_L$, which represents the theoretical arrival times shift between packet 1 and packet k arrivals. Let us now give a new necessary condition for packet k to overflow.

Proposition 2 $(Z_k = b) \implies (W_1 - W_k > (k - 1)(T_F - T_L) + bT_L - X_k)$.

Proof Using proposition 1, we have $tat_{P(k)} = t_1 + (P(k) - 1)T_L + \sum_{j=2}^{P(k)} \delta_j DA_j - N_{P(k)}T_L$. But the number of packets that entered the spacer before packet k is $k - 1 - N_k$. It also equals $P(k) - N_{P(k)} + b - 1$ as $P(k)$ is the next packet to leave the spacer, and the spacer is full when packet k arrives. Therefore $P(k) - N_{P(k)} = k - N_k - b$. Furthermore, $\delta_j = 0, \forall j = P(k) + 1, \dots, k$, as these packets do not arrive in an idle period. This leads to $tat_{P(k)} = t_1 + (k - b - 1)T_L + \sum_{j=2}^k \delta_j DA_j - N_k T_L$. Finally, we obtain $(Z_k = b) \implies (t_k < tat_{P(k)}) \implies (W_1 - W_k > (k - 1)(T_F - T_L) + bT_L - X_k)$. \square

3.2 Upper bound in a stochastic context

Thanks to proposition 2, we can now give an upper bound for the overflow probability, as shown in the proposition below. This is our essential result, as it proves that the loss ratio is proportional to the out-of-profile packets ratio, even if the spacer is fully loaded.

Proposition 3 *Let k be a packet of the flow. If*

- i) $\mathbb{P}(|W_i - W_j| > \tau_F) < \varepsilon, \forall i, j \in \mathbb{N}^*$,*
- ii) $(W_{k+1} - W_k)_{k \geq 1}$ is a sequence of independent random variables, and*
- iii) $b = \max\left(2, \left\lceil \frac{\tau_F}{T_L} \right\rceil\right)$ where $\lceil x \rceil$ is the smallest integer greater than x , then*

$$\mathbb{P}(Z_k = b) < \varepsilon.$$

Proof Let us define A_i as is the index of the first packet of the i^{th} idle period. The key points are to show firstly that $(Z_k = b) \implies (W_{A_i} - W_k > bT_L)$ and secondly that $\{A_i = j\} \in \sigma(W_2 - W_1, \dots, W_j - W_{j-1})$. Then, on the i^{th} idle period, $\{A_i = j\}$ and $\{W_j - W_k > bT_L\}$ are independent by assumption *ii*). Finally, assumption *iii*) implies $bT_L \geq \tau_F$, and assumption *i*) leads to the upper bound. A detailed version of this proof is presented in the appendix 5.2. \square

Assumption *ii*) means that the sequence of waiting times $(W_k)_{k \geq 1}$ has independent increments. Intuitively, this is valid firstly in a large network which is in a stationary state. Secondly, it is valid if the network load varies with time

and the period of the flow is large compared to the network load variation scale. Thirdly, it is valid if the period of the flow is small compared to the network load variation scale, as the network is then almost seen as stationary by this flow. Therefore assumption *ii)* applies to a large range of networks, including high-speed packet networks such as the Internet.

3.3 Upper bound in a deterministic context

The theory of Network Calculus has already solved this problem in a deterministic context. In particular, in [1, Fig. 1.10 p.30], the buffer bound is given. We will now show that our modelling leads to the same result.

Proposition 4 *If $b = \max(2, \lceil \tau_F/T_L \rceil)$ and the flow is conforming with parameters (T_F, τ_F) , then no packet is lost by overflow in the spacer.*

Proof Assume we are in the first busy period, and there exists an index k such that packet k is the first packet to overflow. We have $k \geq b+2$. By equation 3, we have $Z_k = b \implies t_k < \text{tat}_{P(k)}$. As packet k is the first to overflow, $P(k) = k - b$. Furthermore, during the first busy period, $\text{tat}_{k-b} = \text{tat}_1 + (k - b - 1)T_L$. Hence $Z_k = b \implies W_1 - W_k > bT_L \geq \tau_F$. But applying equations 1 and 2, $t_1 + (k - 1)T_F - t_k \leq \tau_F$, therefore $W_1 - W_k \leq \tau_F$. Finally, k doesn't exist. Applying the same reasoning, it is straightforward to prove by induction that no packet overflows during the j^{th} busy period. \square

4 Conclusion

Traffic conditioning at the ingress of a *DiffServ* domain can be achieved by a spacer, which is a particular token bucket. We have seen in this article that it is possible to model it with simple mathematical tools. Applying our model, we have presented in proposition 3 an upper bound for the overflow probability. It proves that the probability of losing packets is of the same order than the probability that the network jitter exceeds the jitter tolerance. Although this result is intuitively valid, it had never, to our knowledge, been proved. Our proof takes into account the random nature of the core network, and is valid even if the spacer is fully loaded. This is important, as introducing either a random context or a load equal to 1 may have unexpected consequences. From a practical point of view, this result can be used to offer statistical guarantees to stringent flows.

As the spacer size increases linearly with the jitter, it seems crucial to reduce this jitter. This can be achieved by implementing traffic control functions, and

in particular appropriate schedulers. Future work shall include application to variable size packets.

5 Appendix

5.1 Proof of proposition 1

By definition of DA_k and δ_k we have $DA_k = W_2 - W_1 + T_F - T_L + N_2T_L$ for $k = 2$, $DA_k = W_k - W_1 + (k - 1)(T_F - T_L) - \sum_{j=2}^{k-1} \delta_j DA_j + N_k T_L$ for $k \geq 3$, and $\delta_k = \mathbb{1}_{DA_k > 0}$, for $k \geq 2$.

Let E_k be the last packet that entered the spacer before packet k . Packet k arrives in an idle period if and only if packet k 's arrival time, t_k , is greater than $tat_{E_k} + T_L$. Any packet reaching the spacer in an idle state is immediately retransmitted. Therefore we have, $\forall k \geq 1$, $tat_k = t_k$ if $t_k > tat_{E_k} + T_L$, and $tat_{E_k} + T_L$ otherwise.

We now prove proposition 1 by induction. We also prove that, if $E_k < k - 1$, then $\delta_{E_k+1} = \delta_{E_k+2} = \dots = \delta_{k-1} = 0$.

For $k = 2$, we have $tat_2 = tat_1 + T_L$ if $t_2 \leq tat_1 + T_L$ and $Z_2 < b$, and t_2 otherwise.

As $Z_2 \leq 1 < b$, $N_2 = 0$, $tat_1 = t_1$ and $t_2 = t_1 - W_1 + T_F + W_2$, we have $tat_1 + T_L \geq t_2 \iff W_2 \leq W_1 + T_L - T_F \iff \delta_2 = 0$. If $t_2 \leq tat_1 + T_L$, we have $tat_2 = tat_1 + T_L = t_1 + T_L + \delta_2 DA_2 - N_2 T_L$. Furthermore, if $t_2 > tat_1 + T_L$, $tat_2 = t_2 = t_1 - W_1 + T_F + W_2 = t_1 + DA_2 + T_L$. Hence in both cases we have $tat_2 = t_1 + T_L + \delta_2 DA_2 - N_2 T_L$.

If the result is true up to $k - 1$, as $Z_k < b$, we have $tat_k = tat_{E_k} + T_L$ if $t_k \leq tat_{E_k} + T_L$, and t_k otherwise.

But $Z_{E_k} < b$ and $1 \leq E_k \leq k - 1$, therefore $tat_{E_k} = t_1 + (E_k - 1)T_L + \sum_{i=2}^{E_k} \delta_i DA_i - N_{E_k} T_L$.

Firstly, if $t_k \leq tat_{E_k} + T_L$, we have $t_k \leq tat_{E_k} + T_L \iff t_1 - W_1 + (k - 1)T_F + W_k \leq t_1 + (E_k - 1)T_L + \sum_{j=2}^{E_k} \delta_j DA_j - N_{E_k} T_L + T_L \iff W_k \leq W_1 - (k - 1)T_F + \sum_{j=2}^{E_k} \delta_j DA_j + (E_k - N_{E_k})T_L$.

Let us show that, if $E_k < k - 1$, then $\delta_{E_k+1} = \delta_{E_k+2} = \dots = \delta_{k-1} = 0$. As packets $E_k + 1, E_k + 2, \dots, k - 1$ overflow, $t_{E_k+1} \leq tat_{E_k} \implies t_1 - W_1 + E_k T_F + W_{E_k+1} \leq t_1 + (E_k - 1)T_L + \sum_{j=2}^{E_k} \delta_j DA_j - N_{E_k} T_L \implies W_{E_k+1} - W_1 + E_k(T_F - T_L) - \sum_{j=2}^{E_k} \delta_j DA_j - N_{E_k+1} T_L \leq -T_L$, as $N_{E_k+1} = N_{E_k}$. Therefore

$t_{E_k+1} \leq \text{tat}_{E_k} \implies \delta_{E_k+1} = 0$. If $\delta_{E_k+1} = \delta_{E_k+2} = \dots = \delta_{E_k+i-1} = 0$ for $E_k + i \leq k - 1$, then $t_{E_k+i} \leq \text{tat}_{E_k} \implies t_1 - W_1 + (E_k + i - 1)T_F + W_{E_k+i} \leq t_1 + (E_k - 1)T_L + \sum_{j=2}^{E_k} \delta_j DA_j - N_{E_k} T_L \implies \delta_{E_k+i} = 0$, as $N_{E_k+i} = N_{E_k} + i - 1$. Hence $\delta_{E_k+1} = \delta_{E_k+2} = \dots = \delta_{k-1} = 0$.

Therefore $t_k \leq \text{tat}_{E_k} + T_L \iff W_k \leq W_1 - (k - 1)T_F + \sum_{j=2}^{E_k} \delta_j DA_j + (E_k - N_{E_k})T_L \iff W_k \leq W_1 - (k - 1)(T_L - T_F) + \sum_{j=2}^{k-1} \delta_j DA_j - N_k T_L \iff \delta_k = 0$. This means an idle period occurs before t_k^- if and only if $\delta_k = 1$. So we have $\text{tat}_k = t_1 + (E_k - 1)T_L + \sum_{i=2}^{E_k} \delta_i DA_i - N_{E_k} T_L + T_L = t_1 + \sum_{i=2}^{k-1} \delta_i DA_i + (E_k - N_{E_k})T_L = t_1 + \sum_{i=2}^k \delta_i DA_i + (k - 1 - N_k)T_L = t_1 + (k - 1)T_L + \sum_{i=2}^k \delta_i DA_i - N_k T_L$.

Secondly, if $t_k > \text{tat}_{E_k} + T_L$, then $\delta_k = 1$, $\text{tat}_k = t_k = t_1 - W_1 + (k - 1)T_F + W_k$ and $\text{tat}_k = t_k = t_1 - W_1 + (k - 1)T_F + W_k = t_1 + (k - 1)T_L + \sum_{i=2}^k \delta_i DA_i - N_k T_L$.

5.2 Proof of proposition 3

Lemma 5 $A_i = \inf\{k > A_{i-1} : W_k - W_{A_{i-1}} + (k - A_{i-1})(T_F - T_L) + (N_k - N_{A_{i-1}})T_L > 0\}$. Furthermore, $\sum_{j=1}^i DA_{A_j} = W_{A_i} - W_1 + (A_i - 1)(T_F - T_L) + N_{A_i} T_L$.

Proof The result is straightforward for $i = 2$. If the result is correct for $i - 1$, then for $k > A_{i-1}$, $DA_k = W_k - W_1 + (k - 1)(T_F - T_L) - \sum_{j=1}^{i-1} DA_{A_j} + N_k T_L = W_k - W_1 + (k - 1)(T_F - T_L) - W_{A_{i-1}} + W_1 - (A_{i-1} - 1)(T_F - T_L) - N_{A_{i-1}} T_L + N_k T_L$, therefore packet k arrives in an idle period if and only if $(DA_k > 0) \iff (W_k - W_{A_{i-1}} + (k - A_{i-1})(T_F - T_L) + (N_k - N_{A_{i-1}})T_L > 0)$. Finally, $\sum_{j=1}^i DA_{A_j} = W_{A_i} - W_{A_{i-1}} + (A_i - A_{i-1})(T_F - T_L) + (N_{A_i} - N_{A_{i-1}})T_L + \sum_{j=1}^{i-1} DA_{A_j} = W_{A_i} - W_1 + (A_i - 1)(T_F - T_L) + N_{A_i} T_L$. \square

Lemma 6 $\forall k \in \mathbb{N}^*$, $N_{k+1} \in \sigma(W_k - W_{k-1}, W_{k-1} - W_{k-2}, \dots, W_2 - W_1)$.

Proof We prove this result by induction. We know that $N_2 = N_3 = \dots = N_{b+2} = 0$, so this is true for $k \leq b + 1$. We know $N_{b+3} \in \{0, 1\}$, and $N_{b+3} = 1 \iff t_{b+2} < \text{tat}_{CT(b+2-b-1)} + T_L \iff t_1 - W_1 + (b + 1)T_F + W_{b+2} < t_1 + T_L \iff W_1 - W_{b+2} > (b + 1)T_F - T_L$. Hence the result is true at index $b + 2$.

Assume this is true for all $i \leq k$. We have $N_{k+1} = N_k$ if packet k does not overflow, and $N_{k+1} = N_k + 1$ otherwise. Hence $N_{k+1} - N_k = 1 \iff t_k < \text{tat}_{CT(k-N_k-b-1)} + T_L \iff W_1 - W_k + (k - N_k - 2 - b)T_L + \sum_{j=2}^{k-N_k-b-1+N_{CT(k-N_k-b-1)}} \delta_j DA_j > (k - 1)T_F$. But $CT(k-N_k-b-1) < k$, so by applying the induction assumption we have $N_{CT(k-N_k-b-1)} \in \sigma(W_k - W_{k-1}, \dots, W_2 - W_1)$. Furthermore, $DA_2 \in \sigma(W_2 - W_1)$. It is straightforward

that $DA_j \in \sigma(W_j - W_{j-1}, \dots, W_2 - W_1)$, and so for $\delta_j = \mathbb{1}_{DA_j > 0}$. Finally, $N_{k+1} - N_k \in \sigma(W_k - W_{k-1}, \dots, W_2 - W_1)$. \square

Proof of proposition 3 Let k be a packet of the i^{th} busy period. We have by lemma 5 : $X_{A_i} = W_{A_i} - W_1 + (A_i - 1)(T_F - T_L)$. Furthermore, knowing packet k belongs to the i^{th} busy period, $X_k = W_{A_i} - W_1 + (A_i - 1)(T_F - T_L) + (N_{A_i} - N_k)T_L$. Hence $Z_k = b \implies W_k < W_1 - (k - 1)T_F + (k - b - 1)T_L + X_k \iff W_{A_i} - W_k > bT_L + (k - A_i)(T_F - T_L) + (N_k - N_{A_i})T_L > bT_L \geq \tau_F$. Using equation 2, we conclude that $\mathbb{P}(Z_k = b) \leq \mathbb{P}(W_k < W_1 - (k - 1)T_F + (k - b - 1)T_L + X_k) \leq \mathbb{P}(W_{A_i} - W_k > \tau_F) = \sum_{j \leq k-1} \mathbb{P}(W_j - W_k > \tau_F | A_i = j) \mathbb{P}(A_i = j)$.

By lemmas 5 and 6, $\{A_i = j\} \in \sigma(W_2 - W_1, \dots, W_j - W_{j-1})$, and $\{W_j - W_k\} \in \sigma(W_k - W_{k-1}, \dots, W_{j+1} - W_j)$. By *ii*) $\sigma(W_2 - W_1, \dots, W_j - W_{j-1})$ and $\sigma(W_k - W_{k-1}, \dots, W_{j+1} - W_j)$ are independent. Therefore $\mathbb{P}(W_j - W_k > \tau_F | A_i = j) = \mathbb{P}(W_j - W_k > \tau_F)$. Hence $\mathbb{P}(Z_k = b) \leq \sum_{j \leq k-1} \mathbb{P}(W_j - W_k > \tau_F) \mathbb{P}(A_i = j) < \varepsilon \sum_{j \leq k-1} \mathbb{P}(A_i = j) \leq \varepsilon$. \square

References

- [1] Jean-Yves Le Boudec, Patrick Thiran, *Network Calculus, A Theory of Deterministic Queuing Systems for the Internet*, Lecture Notes in Computer Science 2050, Springer 2001.
- [2] Andrew S. Tanenbaum, *Computer Networks*, 3rd edition, Prentice-Hall International, Inc., 1996.
- [3] Nathalie Omnès, Thèse n°2416, *Analyse d'outils de contrôle de la qualité de service dans les réseaux de paquets haut débit*, PhD, Université de Rennes 1, France, octobre 2001.
- [4] P. Boyer, F. Guillemin, M. Serval, J. P. Coudreuse, Spacing cells protects and enhances the utilization of ATM network links, *IEEE Network Magazine*, pp. 38-49, September 1992.
- [5] F. Guillemin, P. Boyer, J. Boyer, O. Dugeon and C. Mangin, Regulation of TCP over ATM via cell spacing, in *Proceedings of ITC'16*, Edinburgh, june 1999.
- [6] Gunnar Karlsson, Fredrik Orava, The DIY Approach to QoS, in *Proceedings of IWQoS'99*, IEEE Catalog Number 98EX354, 1999.
- [7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, *RFC 2475, An architecture for Differentiated Services*, IETF, December 1998.
- [8] E. Rosen, A. Viswanathan and R. Callon, *RFC 3031, Multiprotocol Label Switching Architecture IETF*, January 2001.