

Analysis of a Burstiness Curve for FBM Traffic and Token Bucket Shaping Algorithm

L. Janowski* and Z. Papir*

*Department of Telecommunications
AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland
e-mail: {papir, janowski}@kt.agh.edu.pl

Abstract— The paper presents a burstiness curve – a trade-off between Token Bucket descriptors - derived for a FBM traffic model using an envelope analysis. The burstiness curve is defined by traffic parameters (mean, variance, and Hurst parameter) and the considered Token Bucket packet drop probability. The Obtained results allow for more control to be imposed on Token Bucket parameters, when shaping the traffic of a particular service on the network.

I. INTRODUCTION

Since the self-similar (SS) nature of telecommunication traffic in LAN networks has been revealed by Willinger [1] one of the most important questions in teletraffic engineering is how to control SS traffic. The question is crucial because self-similarity was discovered not only in Ethernet [1] but also in WWW traces [2] and VBR video traffic as well [3].

One of the main properties of the self-similar traffic is burstiness [1]. A bursty traffic does not possess a stable mean value. Significant differences in the mean value are one of the reasons why bursty traffic is more difficult to control than a shaped one. Random Early Detection queue algorithm is one of the ways to decrease burstiness of TCP traffic [4]. Token Bucket (TB) algorithm is another method of decreasing traffic burstiness that is based on traffic shaping. Various versions of both the RED and the TB algorithms are widely used in existing networks for their easy implementation.

The influence of self-similar traffic on network control algorithms is significant, therefore open questions still do exist. To answer them many different SS models were employed to render the influence of the SS on network control algorithms. Some models describe self-similarity just for finite time scales (MMPP model) [5], another ones exhibit exact self-similarity as the Fractional Brownian Motion (FBM) model [6] and more general ones exhibit a short range dependence as well as the self-similarity [7].

The Token Bucket algorithm is used in different network techniques and layers [9, 10]. It works like a low bound filter and does not depend on any protocol. A traffic flow after the TB shaping does not exhibit bursts longer than maximal burst size being one of the TB parameters, and a mean bit rate does not exceed some level that is another TB parameter. From an administrator's point of view it means that the traffic

generated by any service being limited by the TB algorithm gets easy to shape. The administrator needs to tune just the TB parameters to make the service traffic more conformant.

Misadapting TB parameters can dramatically increase the drop probability of packets coming from the service. However, according to premises presented in [11, 12] 'burstiness curves' can be defined that allow for some trade-off between a mean bit rate and a burst size at a constant packet drop probability. In the paper a closed-form formula for a FBM traffic burstiness curve is derived.

The second section of the paper provides basic definitions of self-similarity and its FBM model. The third section reminds main properties and parameters of the Token Bucket algorithm and the related burstiness curve. The fourth section presents a traffic envelope approach and its application to derive the burstiness curve for the FBM traffic model. The last section concludes the paper.

II. SELF-SIMILARITY

Self-similar process is defined as follows [13]:

Definition 1. Process $X = (X(t))_{t \geq 0}$ is self-similar (SS) if:

$$\exists H > 0 \forall c > 0 \quad X(ct) \stackrel{d}{=} c^H X(t), \quad (1)$$

where $\stackrel{d}{=}$ means distribution equal (it is $\forall f(x)$ -bound and limited $E\{f(X(ct))\} = E\{f(c^H X(t))\}$). X is called H -ss process and H is called a Hurst parameter being a measure of a self-similarity degree.

The Hurst parameter defines intensity of decreasing a variance for different time scales. As follows from (1):

$$\text{Var}\{X(ct)\} = c^{2H} \text{Var}\{X(t)\}. \quad (2)$$

The Prop. (2) implies that the variance of the SS process on a longer time scale ($c > 1$) is larger for a greater Hurst parameter ($H > 1/2$). On the other hand, long-term process variability is related to its burstiness; the process gets more bursty for an increasing Hurst parameter [1].

Many different models displaying self-similarity are widely used. In the paper the simplest self-similar model proposed by Norros [6] is considered. The reason is that the results obtained from the FBM model are close to an observed network behaviour [14, 15]; furthermore, mathematical properties of the FBM model allow computing its envelope process effectively.

Definition 2. Process $Z(t)$ is a normalized FBM if:

1. $Z(t)$ has stationary increments;
2. $Z(0) = 0$, and $E\{Z(t)\} = 0$ for all t ;
3. $E\{Z^2(t)\} = |t|^{2H}$ for all t ;
4. $Z(t)$ has continuous paths;
5. $Z(t)$ is Gaussian, i.e. its finite-dimensional distributions are multivariate Gaussian distributions.

Accumulated FBM traffic model is defined as follows:

$$A(t) = \bar{a}t + \sigma Z(t), \quad (3)$$

where \bar{a} is a mean value, σ^2 is a variance for $t=1$ and $Z(t)$ meets Def. (2).

Replacing $X(t)$ by $A(t)$ in (2) results in:

$$\text{Var}\{A(ct)\} = \sigma^2 |ct|^{2H} = c^{2H} \text{Var}\{A(t)\}. \quad (4)$$

Equation (4) proves that the FBM model fits (2) describing the variability of the self-similar process and indicates its efficiency for bursty traffic modelling.

In the paper the FBM model was selected from number of SS models. The basic prerequisite to choose the FBM model is knowing its marginal distribution [6, 16, 17]. The distribution allows to computation of an envelope process that is needed to determine the burstiness curve. The disadvantage of the FBM model is its fixed short range dependence, however, results obtained from it are close to network behaviour [15, 18].

III. TOKEN BUCKET AND BURSTINESS CURVE

The Token Bucket includes a pool of tokens (permits). A packet cannot enter transmission without seizing tokens its bit length in number [19]. If a packet arrives to the TB then token number is verified. If there are enough tokens then the packet is sent immediately and token number is reduced by a packet bit length. Otherwise, the packet either waits for tokens or is dropped from service. The token pool is refilled at a constant rate; if the bucket is full, tokens are lost.

The Token Bucket uses two parameters to control the connection. The first parameter is the bucket size $b[\text{bit}]$ and the second one a token accumulating rate $r[\text{bps}]$. Therefore, the Token Bucket takes control over two important packet source parameters: burst size and mean value.

Dropping incoming packets undergoes two restrictions. The bucket size has to be greater than the maximum length of packets in transmission in order to prevent a systematic dropping of the longest packets. On the other hand, dropping packets can occur only in a so called

congestion interval that is a contiguous time period with the token bucket being not fully replenished. Therefore, in order to derive a drop probability relationship only congestion intervals need to be considered. For the congestion interval a maximum accumulated amount of data sent through the TB is limited to:

$$L(t) = b + rt, \quad (5)$$

where t is a congestion interval duration.

The intriguing property of the Token Bucket is that there is no unique pair of TB parameters (r, b) which would respond with a same drop probability. Pairs (r, b) of TB parameters for the same drop probability create so called a burstiness curve $b = b(r)$ [11].

The burstiness curve is a decreasing function $b = b(r)$ according to a simple reasoning based on common understanding the TB algorithm [12]. Suppose that the TB(r_1, b_1) results in the drop probability μ_1 and the TB($r_2 > r_1, b_2 = b_1$) in the drop probability μ_2 correspondingly. As the TB refilling rate grows $r_2 > r_1$ at the same bucket size $b_2 = b_1$ the TB packet drop probability would decrease $\mu_2 < \mu_1$. By definition, however, the burstiness curve has to preserve the constant packet drop probability that could be attained only whenever for $r_2 > r_1$ and $\mu_2 = \mu_1$ it is valid that $b_2 < b_1$.

The exact shape of a burstiness curve seems to be rather difficult to compute for any traffic source model. However, in the paper the burstiness curve for the FBM traffic model is considered as a model which exhibits a SS like a real network traffic. The adopted approach is a kind of conversion of the FBM stochastic process by 'enveloping' (bounding) it by a deterministic time function, guaranteeing a low discrepancy probability.

IV. ENVELOPE PROCESS AND BURSTINESS CURVE FOR FBM TRAFFIC

The burstiness curve for any stochastic process is difficult to derive, therefore, in the paper it is proposed to compute the burstiness curve for the FBM process using an envelope analysis. The envelope analysis, based on a process quantile behaviour with time, converts probabilistic computations into deterministic ones.

A. Envelope process

In most cases deterministic models are more handy for performance evaluation than statistical ones. A method of changing probabilistic computations into a deterministic form is proposed in [8]. The idea of the proposed solution is to construct an envelope process. The envelope process is some deterministic function (of time) having values exceeded by stochastic process for any t with some low excess probability ε . The probability ε , a measure of a majorizing precision, is one of the envelope function parameters.

For the FBM process the envelope process is given by the formula [8]:

$$\bar{A}(t) = \bar{a}t + k\sigma t^H, \quad (6)$$

where \bar{a} and σ^2 are Brownian motion process parameters defined in (3, 4), $k = k(\varepsilon)$ is some constant dependent on the probability ε , and H is Hurst parameter.

The probability that the cumulative process $A(t)$ defined in (3, 4) exceeds its envelope process $\bar{A}(t)$:

$$A(t) > \bar{A}(t), \quad (7)$$

is determined by k :

$$\Pr\left\{\frac{A(t) - \bar{a}t}{\sigma t^H} > k\right\} = \Phi(k) = \varepsilon, \quad (8)$$

where $\Phi(k)$ is the residual distribution function of the standard Gaussian distribution [3].

Note that for low probabilities ε the FBM process is well bounded by the envelope process (6):

$$\bar{A}(t) = \bar{a}t + \Phi^{-1}(\varepsilon)\sigma t^H, \quad (9)$$

where $\Phi^{-1}(\varepsilon) = k$.

Example

The measured parameters of some traffic source are: a mean bit rate $\bar{a} = 1.2$ Mbit/s, a Hurst parameter $H = 0.8$, and a variance $\sigma^2 = 4$ for $t = 1$.

The proper FBM model has the form:

$$A(t) = 1.2t + 2Z(t). \quad (10)$$

The envelope process for the FBM traffic given by (9) follows:

$$\bar{A}(t) = 1.2t + 2\Phi^{-1}(\varepsilon)t^{0.8}. \quad (11)$$

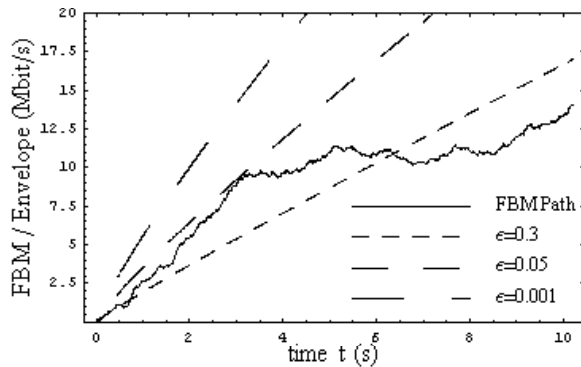


Figure 1. Envelope processes for different ε values

Fig. 1 presents some paths of the analysed FBM traffic (10) and its envelope (11) for a number of different values of the excess probability ε .

The main conclusion available from Fig. 1 is that the lower the excess probability ε is the less accurate is the enveloping.

B. Burstiness curve

The burstiness curve as defined in Sect. 2 is a trade-off function $b = b(r)$ between a maximum burst size and a token refilling rate resulting in the same packet drop probability.

A packet TB drop probability μ for the envelope process can be estimated provided that a packet can be dropped whenever it makes the accumulated traffic exceed the envelope process (the TB algorithm is assumed to be tuned to serve the envelope traffic):

$$\mu \leq \varepsilon. \quad (11)$$

The inequality (11) cannot be more accurate because there is no information about the amount of tokens left in the bucket if an incoming packet exceeds the envelope process. For this reason the worst case has to be assumed in further considerations:

$$\mu = \varepsilon \quad (12)$$

It follows that the envelope excess probability ε could be equated to the packet TB drop probability.

The properly tuned TB does not drop the incoming packets if:

$$b + rt \geq A(t) \quad (13)$$

for all t .

Taking into account the envelope approximation of the FBM incoming traffic (6) the (13) gets converted into (14).

$$rt + b \geq \bar{a}t + k\sigma t^H. \quad (14)$$

The k parameter is obtained when solving (8) for the considered excess probability ε (typically 0.05, 0.01).

Wrapping (14) gets:

$$(r - \bar{a})t - k\sigma t^H + b \geq 0 \quad (15)$$

for all t .

Since the Hurst parameter $1/2 < H < 1$ the (15) is satisfied for the stable case only $r - \bar{a} > 0$.

To proceed further it is enough to notice that the (15) has to be met for the worst case and therefore, the minimum value of the left side of the (15) in turn must be equal to zero (as of a weak inequality).

The time when the left side of (15) yields a minimum value is t^* :

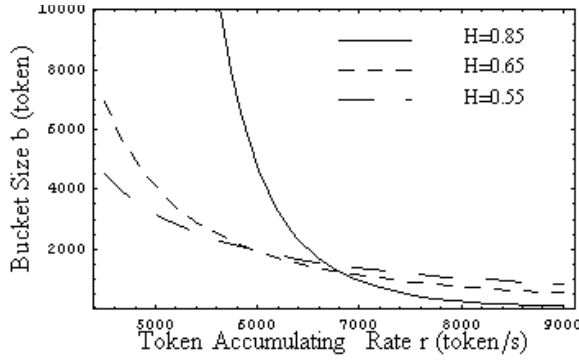


Figure 2. Envelope processes for different ϵ values

$$t^* = \left(\frac{k\sigma H}{r - \bar{a}} \right)^{\frac{1}{1-H}}. \quad (16)$$

Inserting the t^* into (15) and converting it into the worst case equation the bucket size b as a function of traffic parameters \bar{a}, σ, H , drop probability ϵ , and token accumulating rate r is obtained:

$$b = (\bar{a} - r) \left(\frac{\Phi^{-1}(\epsilon)\sigma H}{r - \bar{a}} \right)^{\frac{1}{1-H}} + \Phi^{-1}(\epsilon)\sigma \left(\frac{\Phi^{-1}(\epsilon)\sigma H}{r - \bar{a}} \right)^{\frac{H}{1-H}} \quad (17)$$

For known traffic parameters \bar{a}, σ, H , and the considered drop (excess) probability ϵ , the burstiness curve can be plotted from (17). Some examples are drawn in Fig. 2.

Fig. 2 proves that serving a traffic with a higher Hurst parameter requests much faster token refilling rate r for the same token bucket b and the same traffic parameters. It means, that if slow Token Bucket rate is needed after Token Bucket traffic shaping, then the burst size has to be larger for a higher Hurst parameter. The importance of this result is that (17) enables us to compute a new value of TB size, if the bucket accumulating rate is changed and if the traffic source is characterized by the self-similarity.

V. CONCLUSIONS

The envelope process allows to derive the Token Bucket burstiness curve for the FBM model. The result is a closed-form analytic equation that can be used to control optimal TB parameters for different applications like video streaming or edge router traffic aggregating in IP QoS networks following DiffServ concept [11]. An important result is that for traffic characterized by higher Hurst parameter, the burstiness curve is a faster decreasing function, however, the bucket size values are much higher for slower token accumulating rates than for traffic characterized by small Hurst parameter.

ACKNOWLEDGMENT

This work has been partly supported by the European Union under the E-Next Project FP6-506869.

REFERENCES

- [1] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson. On the Self-Similar Nature of Ethernet Traffic (Extended Version). *IEEE Transactions on Networking*, Vol. 2, 1-15, February, 1994.
- [2] M. E. Crovella and A. Bestavros. Self-similarity in World Wide Web traffic: evidence and possible causes. *IEEE/ACM Trans. Networking* 5 (6) (1997), 835-846, December, 1997.
- [3] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. *Proceedings of the 20th Annual Conference on Local Computer Networks*, 397-406, 1995.
- [4] S. Floyd and V. Jacobson. Random Early Detection gateways for Congestion Avoidance. V.1 N.4, 397-413, August, 1993.
- [5] P. S. Ferreira, R. Valadas and A. Pacheco. Multiscale Fitting Procedure using Markov Modulated Poisson Processes. *Telecommunication Systems*, Vol. 23, No. 1-2, 123-148, June, 2003.
- [6] I. Norros. A Storage Model with Self-similar Input. *Queueing Systems*, Vol 16, 387-396, 1994.
- [7] J. C. Lopez-Ardao, P. Argibay-Losada and R. F. Rodriguez-Rubio. On Modeling MPEG Video at the Frame Level Using Self-Similar Processes. *MIPS 2004*, November, 2004.
- [8] N. Fonseca, G. Mayor, C. Neto. On the equivalent bandwidth of self-similar sources. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, Vol.10 No. 2, 104-124, April 2000
- [9] J. Glasmann, M. Czermin, A. Riedl. Estimation of Token Bucket Parameters for Videoconferencing Systems in Corporate Networks. *SoftCOM 2000*, 10-14, October, 2000.
- [10] A. Vidács, S. Molnár, G. Gordos and I. Cselényi. The Impact of Long Range Dependence on Cell Loss in an ATM Wide Area Network. *Proc. of GLOBECOM'98*, 8-12, November, 1998.
- [11] O. Narayan. Exact Asymptotic Queue Length Distribution for Fractional Brownian Traffic. *Advances in Performance Analysis*, Vol. 1, No. 1, 39-63, 1998.
- [12] M. Dabrowski and W. Burakowski. Assessment of token bucket parameters by on-line traffic measurements. *10th Polish Teletraffic Symposium PSRT*, September, 2003.
- [13] J. R. Gallardo, D. Makrakis and L. Orozco-Barbosa. Use of Alpha-Stable Self-Similar Stochastic Processes for Modeling Traffic in Broadband Networks. *Performance Evaluation*, Vol. 40, No. 1-3, 71-98, March, 2000.
- [14] W. Stallings. *High-Speed Networks: TCP/IP and ATM Design Principles*. Prentice Hall, 1998.
- [15] W. Willinger, A. Erramilli, and O. Narayan. Experimental queueing analysis with long-range dependent traffic. *IEEE/ACM Trans. on Networking*, 209-223, April, 1996.
- [16] L. Massouli and A. Simonian. Large Buffer Asymptotics for the Queue with FBM Input. *Applied Probability*, Vol. 36, No. 3, 894-906, 1999.
- [17] S. Valaee. A recursive estimator of worst-case burstiness. *IEEE/ACM Trans. on Networking*, 211-222, April 2001.
- [18] Y. Shu, F. Xue, Z. Jin, and O. Yang. The Impact of Self-similar Traffic on Network Delay. *Journal of Computer Science and Technology*, vol. 14, no. 6, 585-589, November, 1999; also in *Proc. IEEE 1998 Canadian Conference on Electrical and Computer Engineering*, Waterloo, Canada, 24-28, May, 1998.
- [19] Y. Bernet, S. Blake, D. Grossman and A. Smith. An Informal Management Model for DiffServ Routers. *RFC 3290*, May, 2002.