A new tool for generating realistic Internet traffic in NS-3*

Doreid Ammar Thomas Begin Isabelle Guerin-Lassous

Université Lyon 1 / LIP
UMR ENS Lyon - INRIA - CNRS - UCBL

Email: {firstname.lastname}@ens-lyon.fr

ABSTRACT

This paper provides a contribution for NS-3 consisting of a new tool for generating Internet traffic. This tool is based on the Poisson Pareto Burst Process (PPBP), a Long-Range Dependent (LRD) model for network traffic. The PPBP model provides a simple and accurate network traffic generator that matches statistical properties of real-life IP networks. We have implemented this model in NS-3. We evaluate the computing performance of this PPBP implementation. Our results show a moderate overhead introduction in terms of memory needs and a roughly identical cost in CPU time as compared to a simple Poisson traffic generator.

Keywords

network simulator 3, traffic generator model, Long Range Dependent, Poisson Pareto Burst Process.

1. INTRODUCTION

Discrete-event simulation is a widely used technique for evaluating the performance of existent and non-existent computer network systems. On one hand, it represents an easier way to setup and deploy instruments for various study cases, and it also allows a simple method to reproduce tests. On the other hand, network simulation must be realistic and accurate, yet enabling both ease of use and computationally efficiency to allow scalable scenarios.

When simulating a computer network, there is an obvious critical need for realistic traffic generators. One would expect a traffic generator that (1) matches some key statistical properties of real-life IP networks so as to ensure the overall simulation accuracy, while (2) enabling both ease of use and minimization of costs in terms of memory consumption and CPU time.

*This work has been partly supported by the project *Semantic Networking* within the common laboratory INRIA - Bell Labs.

Simple traffic models like Poisson processes usually include few controllable parameters, which tends to ease their setup. However, this simplicity may result from unrealistic assumptions, such as ignoring some properties exhibited by real-life traffic (e.g., long-range dependence, self-similarity), and therefore leads the researchers to irrelevant results. It may then be crucial in some cases to equip the simulator with traffic generators enabled to generate processes with long-range dependence.

Heavy-tailed on-off sources is a common way to form a long-range dependent traffic flow [9]. In this paper, we focus on another model, namely the Poisson Pareto Burst Process (PPBP) model [10], that also generates long-range dependent traffic. The PPBP model can be viewed as the asymptotic case of heavy-tailed on-off sources when the number of sources is very large, and this eases its parameters setup and handling.

Several tools were developed to model and simulate the Internet traffic. Some of these works are based on parameterization of traffic models from live network measurements, such as RApid [6]. Other works focus on stochastic processes models that qualitatively reproduce certain aspects of the Internet Traffic. Proposed models include the fractional Brownian motion input model [1], the fractional Gaussian noise input process [7] and the Poisson Pareto Burst Process that relies on a $M/G/\infty$ queue [3, 4, 8].

For testing new solutions in NS-3, one often needs to generate traffic and, if possible, traffic models that fairly reproduce real-life traffic. As far as we know, very few traffic generators are available in NS-3. In this work, we propose a new tool for generating traffic based on the PPBP model for the NS-3 practitioners.

The remainder of this paper is organized as follows. In Section 2 we give an overview of the PPBP process and we detail the PPBP NS-3 module that we have implemented. Section 3 is devoted to the model validation and performance results. Section 4 concludes this paper.

2. ARCHITECTURE OVERVIEW

This section briefly presents the Poisson Pareto Burst Process model. It presents formulae relating the parameters of the PPBP, and describes the implementation of the model in NS-3.

2.1 The Poisson Pareto Burst Process (PPBP)

The PPBP is a process based on the overlapping of multiple bursts with heavy-tailed distributed lengths [10]. Events in this process represent points of time at which one of an infinite population of users begins or stops transmitting a traffic burst. The PPBP is closely related to the $M/G/\infty$ queue model as proposed by Cox [3, 4].

In the PPBP model, bursts arrive according to a Poisson process with rate λ_p, and their length follows a Pareto distribution characterized by Hurst parameter H, typically between 0.5 and 0.9, and a mean T_{on}. Each burst is modeled by a flow with a constant bit-rate r. Then, the overlapping bursts form in aggregate a Long-Range Dependent traffic provided the burst length have infinite variance [10].

Based on Little's law [2], the average number of active bursts is given by:

$$E[n] = T_{on} \times \lambda_p \quad (1)$$

Since each burst gives birth to a flow with a constant bit-rate, it is then straightforward to compute the overall rate of the PPBP, λ:

$$\lambda = T_{on} \times \lambda_p \times r \quad (2)$$

2.2 Implementation

Throughout the design of the PPBP module, a particular attention was paid to its flexibility, computing performance and usability.

The PPBP module allows any NS-3 practitioners to easily generate a single flow according to the PPBP process as described in Section 2.1. A new PPBP source can be introduced within a simulation scenario with just a couple of lines of code. In practice, the practitioners simply need to setup the PPBP source by defining values for the Hurst parameter, H, the mean length of a burst, T_{on}, the bit-rate of each individual burst, r and the arrival rate of burst, λ_p, (or alternately, the average number of active bursts, $E[n]$) so that, overall, the rate of the PPBP source, λ, gets the desirable value (see Eq. (2)).

The PPBP module consists of two main functions. The first function allows to keep track of the current number of active bursts at time t, n_t, taking into account that their arrival process follows a Poisson process and that their length is determined by a Pareto distribution. The second function generates the packets departure at a constant bit-rate $n_t \times r$, and thus needs to query the first function.

3. VALIDATION AND RESULTS

3.1 Experimental results

We now evaluate the computing performance of the implemented PPBP module as well as, for sake of comparison, that of a simple Poisson traffic generator. To do this, we consider a very simple network made up of only two nodes connected by one link. The link capacity is set to 10 Mbps. At one node, we generate for a period of time of 250 sec a flow resulting from the PPBP module (or from a Poisson source model). For the PPBP generator, we select $H = 0.7$ as recommended in [5], $T_{on} = 200$ ms and $r = 1$ Mbps. The packets size is set to 1470 bytes. Meanwhile, we measure

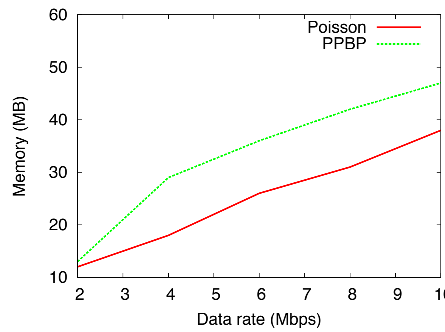

Figure 1: Memory consumption for processing packets departure

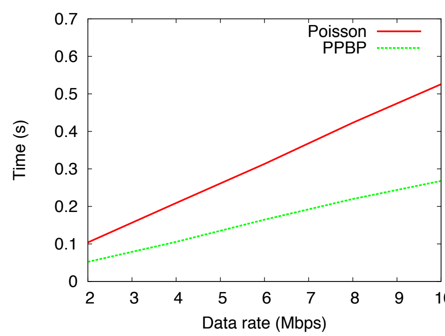

Figure 2: CPU time for processing packets departure

the memory consumption and the CPU time needed to generate the packets departure. We repeat this experiment for various levels of the flow's data rate between 1 and 10 Mbps.

Figure 1 shows the additional overhead, in terms of memory consumption, that is incurred by using the PPBP model instead of a simple Poisson source. It appears that the total amount of consumed memory through the simulation is always greater using the PPBP model. This can easily be explained as the PPBP module, unlike a single Poisson source, must keep track of the current number of bursts and of their size. For instance, the memory overhead brought by the PPBP model comparing to a simple Poisson source reached 60%, corresponding to 11 MB of extra memory consumption when the generated flow has a data rate, λ, equal to 4 Mbps.

Figure 2 shows the CPU time spent for processing the packets departure using the PPBP module or alternately using a Poisson source model. Perhaps surprisingly, the PPBP module goes faster to generate the packet departure times than a Poisson process. This discrepancy can be explained by the fact that PPBP module draws into an exponential and a Pareto distributions if and only if the number of bursts changes, while a Poisson process has to generate a random value for every single packet departure. However, it is worthwhile to state that using a PPBP process as packet generator within a computer network model will typically require a larger number of transmitted packets than a Poisson source to get the system steady-state [5].

3.2 Autocorrelation proof

In Figure 3, we plot the autocorrelation function of a 50000 packets trace generated by the PPBP module with the parameters given above and a data rate of 4 Mbps. This function exhibits a "slow" rate of decay, akin to a long memory process. More specifically, the long-range dependence is observed up to 1.5 sec.

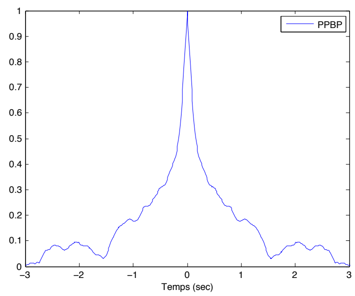

Figure 3: Autocorrelation function for a trace resulting from our PPBP generator

4. CONCLUSIONS

In this paper we have described and evaluated the implementation in NS-3 of a Poisson Pareto Burst Process (PPBP) model for generating Internet traffic. This traffic model is both a simple and accurate way for representing the aggregation of network traffic as it performs to capture the Long-Range Dependence (LRD) exhibited by Internet traffic. Our implementation in NS-3 was validated using a autocorrelation function plot that reveals the LRD property. In addition, our experimental results show that our PPBP module implementation introduces only a moderate computational overhead as compared to a single Poisson source. To conclude, we believe that our work could be useful to NS-3 practitioners as our PPBP module implementation, available at `http://perso.ens-lyon.fr/thomas.begin/NS3-PPBP.zip`, is easy to handle and may improve the accuracy of some computer network simulations.

Acknowledgment

We would like to thank Paulo Gonçalves for his help on the autocorrelation results.

5. REFERENCES

[1] R.G. Addie, M. Zukerman and T. Neame, *Fractal traffic: Measurements, modeling and performance evaluation.* Proceeding of Infocom 95, Boston (MA), April 1995, pp. 985-992.

[2] A. O. Allen, *Probability, statistics and queuing theory with computer science applications*, Vol. 2, Academic Press, 1990.

[3] D. R. Cox and V. Isham *Point Processes.* Chapman and Hall, 1980.

[4] D. R. Cox, *Long-range dependence: A review.* In H. A. David and H. T. David, editors, Statistics: An Appraisal, pages 55-74. Iowa State University Press, 1984.

[5] GROSS D., SHORTLE J.F., FISCHER M.J., MASI D.M. "Simulation input analysis: difficulties in simulating queues with Pareto service". In Proceedings of the 34th Winter Simulation Conference, pp. 407-415, 2002.

[6] Kun-chan Lan and John Heidemann, *A Tool for RApid Model Parameterization and its Applications*, Proceedings of the ACM SIGCOMM 2003 Workshops.

[7] I. Norros *A storage model with self-similar input.* Queueing Systems - Theory and Applications 16 (1994), pp. 387-396.

[8] Minothi Parulekar and Armand M. Makowski, *M|G|Infinity Input Processes: A Versatile Class of Models for Network Traffic.* INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Los Alamitos, CA, USA.

[9] Willinger, Walter and Taqqu, Murad S. and Sherman, Robert and Wilson, Daniel V. *Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level.* IEEE/ACM Trans. Netw., 1997.

[10] M. Zukerman, T. D. Neame and R. G. Addie, *Internet Traffic Modeling and Future Technology Implications.* Proceedings of INFOCOM 2003, San Francisco, USA, April 2003.