Delays in IP Routers, a Markov Model

Tadeusz Czachórski[1]([✉]), Adam Domański[2], Joanna Domańska[1],
Michele Pagano[3], and Artur Rataj[1]

[1] Institute of Theoretical and Applied Informatics, Polish Academy of Sciences,
Baltycka 5, 44–100 Gliwice, Poland
`tadek@iitis.pl`
[2] Institute of Informatics, Silesian Technical University,
Akademicka 16, 44–100 Gliwice, Poland
`adamd@polsl.pl`
[3] Department of Information Engineering, University of Pisa,
Via Caruso 16, 56122 Pisa, Italy
`m.pagano@iet.unipi.it`

Abstract. Delays in routers are an important component of end-to-end delay and therefore have a significant impact on quality of service. While the other component, the propagation time, is easy to predict as the distance divided by the speed of light inside the link, the queueing delays of packets inside routers depend on the current, usually dynamically changing congestion and on the stochastic features of the flows. We use a Markov model taking into account the distribution of the size of packets and self-similarity of incoming flows to investigate their impact on the queueing delays and their dynamics.

Keywords: Markov queueing models · Self-similarity · IP packets length distribution · IP routers delays

1   Introduction

Queueing theory has its origins in models proposed by Erlang and Engset a hundred years ago for evaluation of telephone and telegraph systems. These models were based on Markov chains, which since then accompany modelling and evaluation of telecommunication systems. With the increasing complexity of models they encounter natural limitations as state explosion and numerical problems with solving very large systems of equations. On the other hand, the increase of computer power and size of memory, as well as the development of better software help us to overcome these problems.

This is why we are trying here to refine Markov models of router queueus. It is well known that the distribution of the size of packets and self-similarity of the input traffic have an impact on the transmission quality of service (determined by transmission time, jitter, and loss probability); they influence also dynamics of changes of number of packets waiting in routers to be forwarded. These issues are usually investigated with the use of discrete-event simulations which in case of

self-similar traffic demand very long runs and are time consuming, especially if we study transient states. Here we introduce to a Markov model details which were previously reserved to simulation models: a real distribution of IP packets and self-similar nature of packet flows. To obtain numerical results we use standard software: HyperStar [20] to approximate measured distributions with phase-type ones, enabling the use of Markov chains and Prism [11] to study transient states of a complex Markov model. We use also existing Markovian models of self-similar traffic [1]. With this purely engineering approach, we are able to construct more realistic than existing before models of IP queues and delays. The article is a continuation of [5] where we considered the queue length distributions at IP routers. Here we concentrate on the distribution of delays in these queues. The numerical study is based on more recent data.

2   Distribution of IP Packets

CAIDA, Center for Applied Internet Data Analysis [3], routinely collects traces on several backbone links. These monthly traces of one hour each are provided to interested researchers on request in pcap files containing payload-stripped, anonymised traffic. We used measurements of CAIDA coming from the link Equinix Chicago collected during one hour on 18 February 2016 having 22 644 654 packets belonging to 1 174 515 IPv4 flows, [4].

In a Markov model we should represent any real distribution with the use of a system of exponentially distributed phases (PH). Numerical PH fitting, e.g. with the use of Expectation Maximisation Algorithms, is a frequently investigated problem [2], and various tools exist, HyperStar [20], which we have chosen, is reported to be efficient at fitting spikes as in case of our distribution.

Figure 1 presents the cumulative distribution function (cdf) of IP packet lengths obtained from this trace and its approximation with the use of an hyper–Erlang distribution having three Erlang distributions with a variable number of phases, up to 3000. It demonstrates the quality of fitting as a function of the number of phases. To limit the number of states in the Markov model to follow, we have chosen the modest maximum number of phases to 300. The resulting Erlang distributions in parallel have 15, 4 and 300 phases, and its density function is (for $x \geq 0$)

$$f_B(x) = 0.05233 \frac{(0.01417)^{15} x^{15} e^{-0.01417x}}{14!} + 0.51162 \frac{(0.06067)^4 x^3 e^{-0.06067x}}{3!}$$
$$+ 0.43604 \frac{(0.20277)^{300} x^{299} e^{-0.20277x}}{299!}. \tag{1}$$

The largest approximation errors are at both extremities of the distribution, for small and large packets (e.g. the cdf is not equal 1 for the size of 1500 bytes). The mean of this distribution, i.e. mean packet size is 734.241 bytes. The same character has the distribution of service times, as the time to send a packet is proportional to its size, only phase parameters are rescaled. In numerical examples we assume that the buffer volume is equal to 64 mean packet size.
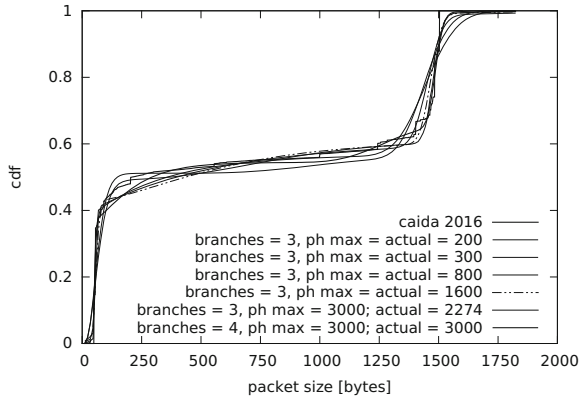
Fig. 1. The influence of the complexity of Markov model of a TCP packet size on the quality of the model

3   Self-similar Traffic

Since the mid 90s, with the collection of high-quality traffic measurements on several Ethernet LANs at the Bellcore Morristown Research and Engineering Center [12] and the statistical analysis of the collected data [13], self-similarity has become an important research domain [14]. In the following years the same statistical features have been confirmed by traffic measurements over different network and application scenarios. Moreover, various works highlighted the relevant impact of the long memory properties, typical of self-similar processes, on queueing dynamics; indeed, ignoring these phenomena leads to an underestimation of important performance measures, such as queue lengths at buffers and packet loss probability [8,10]. Therefore, it is necessary to take into account these features in realistic models of traffic.

Unfortunately, pure self-similar processes lack analytical tractability and only asymptotic results, typically derived in the framework of Large Deviation Theory, are available for simple queueing models (see, for instance, [15] and references therein). Therefore, many researchers investigated the suitability of Markovian models to describe traffic flows that exhibit self-similarity [6,21]. Different models have been proposed, but all works highlighted an important common conclusion: matching self-similarity is only required within the time scales of interest for the analysed system, e.g. [16].

As a result, more traditional and well investigated traffic models, such as Markov Modulated Poisson Processes (MMPPs), may still be used for modelling self-similar traffic. In this paper we focus on the model originally proposed in [1], and detailed in [6]. The model is simple: pseudo self-similar traffic can be generated as the superposition of a number of ON-OFF sources, a special case of two-state MMPPs, also known as Interrupted Poisson Processes, since the rate is zero when the modulating chain is in one of the two states (OFF state); we used five ON-OFF sources.

4   Remarks on Buffer Occupation and Loss Probability

Let us consider now the buffer occupation and the associated loss probability. In the majority of queueing models a system capacity is expressed as the maximum number of customers that may be inside the system, waiting in the queue or being served. This approach is quite natural in case of fixed-size packets (for instance, in case of ATM cells), but can be misleading in IP networks, due to the high variability of packets size, as described in Sect. 2, and to the fact that the amount of memory in a router is typically expressed in bytes. However, the queue length distribution when i packets are in the buffer permits us to determine if there is still place for the next one. Assuming that the lengths of the packets are independent, it is straightforward to calculate the steady-state conditional queue distribution $Q_i(x) = P(Q < x|i$ packets are enqueued$)$ (for $i \geq 2$) as the i-fold convolution of the original distribution. Hence, we can easily calculate the probability that the queue length with i packets exceeds the volume V of the buffer and use this value as $p_{loss}(i)$, i.e. the probability that a packet is refused when there are already $i - 1$ packets in the buffer. The rate of the input flow is thus $\lambda(i) = \lambda(1 - p_{loss}(i))$.

It is worth mentioning that our approach introduces some kind of approximation: indeed, on one side we consider not the real length of the packets laying in the queue, but just the length distribution with which they have been generated. On the other side, the loss probability will depend also on the length of the arriving packet; so, if the queue is almost full for most of the time, it is likely it will mainly contain short packets and so the real queue length (in bytes) might be less, leading to an upper bound of the real p_{loss}. Instead, in case of lower utilisation, the queue length distribution seen by the arriving packet should be much closer to $Q_i(x)$ and hence our approximation works better.

5   Numerical Solutions, Transient States, Network Dynamics

Queueing models are usually limited to the analysis of steady states and popular Markovian solvers, as e.g. PEPS [17] are adapted to it. However, the intensities of real network traffic are perpetually changing; users send variable quantities of data, and traffic control algorithms interfere to avoid congestion (congestion window used in TCP is a good example).

Theoretically, for any continuous time Markov chain with transition matrix $\mathbf{Q}$ the Chapman-Kolmogorov equations

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \boldsymbol{\pi}(t)\mathbf{Q}, \tag{2}$$

have the analytical transient solution $\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)e^{\mathbf{Q}t}$, where $\boldsymbol{\pi}(t)$ is the probability vector and $\boldsymbol{\pi}(0)$ is the initial condition. However, it is not easy to compute the expression $e^{\mathbf{Q}t}$ when $\mathbf{Q}$ is a large matrix. An efficient approach is to use a projection method, where the original problems is projected to a space

(e.g. Krylov subspace) where it has considerably smaller dimension, solve it there and then re-transform this solution to the original state space [22]. It is implemented among others in a well known probabilistic model checker Prism [11]. We used Prism supplementing it with a preprocessor based on [18,19] to ease the formulation of more complex queueing models.

6   Response Time Distribution

Having the queue distribution $p(n)$, the response time (waiting time plus service time) probability density function (pdf) $f_R(x)$ is obtained as

$$f_R(x) = \sum_n p(n) f_B(x)^{*(n+1)}$$

where $f_B(x)$ is the pdf of service time distribution and $*(i)$ denotes i-fold convolution.

Figure 2 presents the comparison of response time distribution given by simulation and our model. Simulation was based on real traffic and packet size traces. In Markov model we used ON-OFF sources with the corresponding to measurements Hurst parameter (average of estimations made by several methods) and the described Hyper-Erlang distribution. In linear time scale the errors of the model are almost invisible. Therefore we use logarithmic time scale. The discrepancies are caused, amongst others, by the insufficient precision of approximation by the function in Eq. 1. It gives an under-representation of actual sizes of small packets, and a respective over-representation of large packets.

In numerical examples we use the validated above model to illustrate the impact of self-similarity, utilisation factor ϱ, and packet size distribution on the response time. In the examples the input flow starts at $t = 0$ and the queue is initially empty. Figures 3 and 4 present (i) the evolution of the mean response time as a function of time – time is normalized to the mean service time of
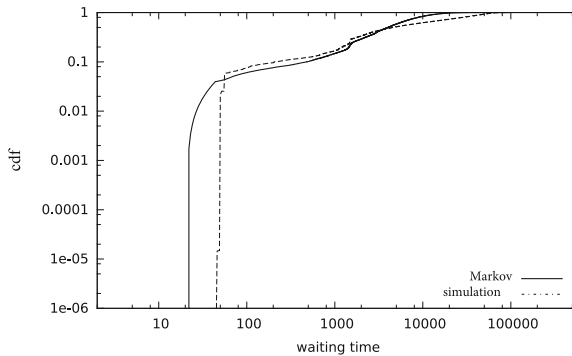

Fig. 2. Comparison of response time distribution given by simulation and Markov model, logarithmic time scale
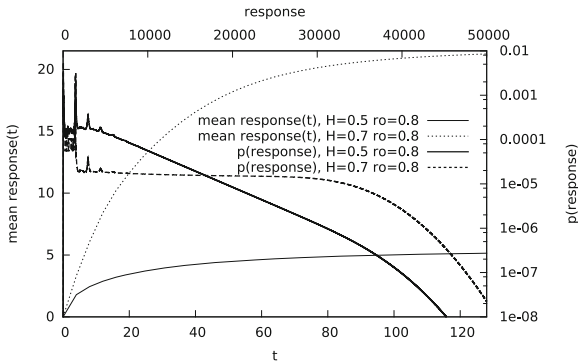
Fig. 3. Mean response time as a function of time and steady state distribution of response time for hyper-Erlang representation of service time distribution, $H = 0.5, 0.7$, $\varrho = 0.8$
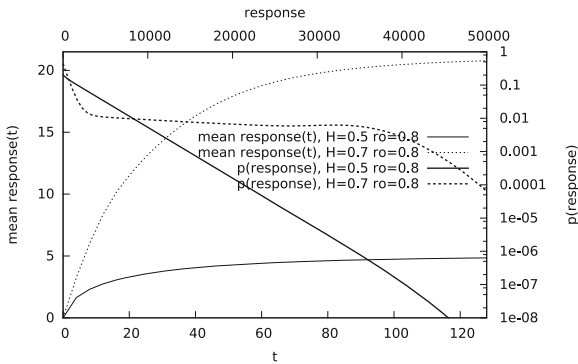

Fig. 4. Mean response time as a function of time and steady state distribution of response time for exponential service time distribution, $H = 0.5, 0.7$, $\varrho = 0.8$

a packet and we consider $t \in [0, 120]$ (ii) steady state distribution of response time – time unit here is the time to serve one byte and we consider the interval $[0, 50000]$. In Fig. 3 we considered our Hyper-Erlang representation of service time distribution, $\varrho = 0.8$, and the input traffic is either Poisson ($H = 0.5$) or self-similar ($H = 0.7$). In Fig. 4 the hyper-Erlang is replaced by an exponential distribution with the same mean.

From the comparison of the simulation results, it is easy to notice the effect of self-similarity that worsen both the transient and steady-state behaviour of the system, confirming that the use of just 5 ON-OFF sources is enough to capture correlation on all the relevant time scales (at least for the considered buffer size). As far as the service time distribution is concerned, it significantly influences the steady-state performance, especially in case of self-similar traffic (and hence for actual traffic flows). In other words, self-similarity and actual packet size distribution are relevant factors that must be taken into account in looking for realist traffic models.

7   Conclusions

In this work we proposed an approach that unifies in a Markovian model (i) a real IP packet distribution which is a basis to define both the losses due to a finite buffer volume and the service time distribution (ii) self similar traffic. The presented numerical examples, based on real traffic data collected by CAIDA a few months ago, confirm that our approach is feasible and may be used also to study transient behaviour of router delays.

Quantitative results may be obtained with the use of well known public software tools. As further work, we plan to apply our approach to Active Queue Management mechanisms.

References

1. Andersen, A.T., Nielsen, B.F.: A Markovian approach for modeling packet traffic with long-range dependence. IEEE J. Sel. Areas Commun. **16**(5), 719 (1998)
2. Buchholz, P., Kriege, J., Felko, I.: Input Modeling with Phase-Type Distributions, Markov Models: Theory and Applications. Springer, Heidelberg (2014)
3. http://www.caida.org/home/
4. https://data.caida.org/datasets/passive-2016/equinix-chicago/20160218-130000. UTC/
5. Czachórski, T., Domanski, A., Domanska, J., Rataj, A.: A study of IP router queues with the use of Markov models. In: Gaj, P., Kwiecien, A., Stera, P. (eds.) CN 2016. CCIS, vol. 608, pp. 294–305. Springer, Heidelberg (2016). doi:10.1007/978-3-319-39207-3_26
6. Domańska, J., Domański, A., Czachórski, T.: Modeling packet traffic with the use of superpositions of two-state MMPPs. In: Kwiecień, A., Gaj, P., Stera, P. (eds.) CN 2014. CCIS, vol. 431, pp. 24–36. Springer, Heidelberg (2014). doi:10.1007/978-3-319-07941-7_3
7. Fischer, W., Meier-Hellstern, K.: The Markov-modulated Poisson process (MMPP) cookbook. Perform. Eval. **18**(2), 149–171 (1993)
8. Gorrasi, A., Restino, R.: Experimental comparison of some scheduling disciplines fed by self-similar traffic. Proc. IEEE Int. Conf. Comm. **1**, 163–167 (2003)
9. Grossglauser, M., Bolot, J.C.: On the relevance of long-range dependence in network traffic. IEEE/ACM Trans. Netw. **7**(5), 629–640 (1999)

10. Kim, Y.G., Min, P.S.: On the prediction of average queueing delay with self-similar traffic. In: Proceedings of IEEE Globecom 2003, vol. 5, pp. 2987–2991 (2003)
11. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: verification of probabilistic real-time systems. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 585–591. Springer, Heidelberg (2011). doi:10.1007/978-3-642-22110-1_47. www.prismmodelchecker.org/
12. Leland, W.E., Wilson, D.V.: High time-resolution measurement, analysis of LAN traffic: implications for LAN interconnection. In: INFOCOM 1991, Proceedings of the Tenth Annual Joint Conference of the IEEE Computer and Communications Societies. Networking in the 90s, vol. 3, pp. 1360–1366. IEEE (1991)
13. Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.: On the self-similar nature of Ethernet traffic (extended version). IEEE/ACM Trans. Netw. **2**(1), 1–15 (1994)
14. Loiseau, P., Gonçalves, P., Dewaele, G., Borgnat, P., Abry, P.V.-B., Primet, P.V.-B.: Investigating self-similarity and heavy-tailed distributions on a large-scale experimental facility. IEEE/ACM Trans. Netw. **18**(4), 1261–1274 (2010)
15. Mandjes, M.: Large Deviations of Gaussian Queues. Wiley, Chichester (2007)
16. Nogueira, A., Salvador, P., Valadas, R., Pacheco, A.: Markovian modelling of internet traffic. In: Kouvatsos, D.D. (ed.) Next Generation Internet: Performance Evaluation and Applications. LNCS, vol. 5233, pp. 98–124. Springer, Heidelberg (2011). doi:10.1007/978-3-642-02742-0_5
17. PEPS. www-id.imag.fr/Logiciels/peps/userguide.html
18. Rataj, A., Wozna, B., Zbrzezny, A.: A translator of java programs to TADDs. Fundam. Inform. **93**(1), 305 (2009)
19. Rataj, A.: More flexible models using a new version of the translator of java sources to times automatons J2TADD. Theor. Appl. Inform. **21**(2), 107–114 (2009)
20. Reinecke, P., Krauß, T., Wolter, K.: HyperStar: phase-type fitting made easy. In: 9th International Conference on the Quantitative Evaluation of Systems (QEST) 2012, Tool Presentation, pp. 201–202, September 2012
21. Robert, S., Boudec, J.Y.L.: New models for pseudo self-similar traffic. Perform. Eval. **30**(1–2), 57 (1997)
22. Stewart, W.: Introduction to the Numerical Solution of Markov Chains. Princeton University Press, Chichester (1994)