Burst ratio in a single-server queue

Andrzej Chydzinski¹ · Dominik Samociuk¹

Published online: 5 June 2018
© The Author(s) 2018

Abstract

In contemporary packet networks, the possibility of packet loss is a negative, but inevitable aspect of the network design. One of the most important characteristics of the packet loss process is the burst ratio—a characteristic describing the tendency of losses to occur in long series, one after another. In this paper, we study the burst ratio in the queueing system with the finite buffer for packets. This is motivated by the fact, that most packet losses in wired networks occur due to queueing of packets in routers, and overflowing routers' buffers. We firstly derive the exact formula for the burst ratio. Then we study its behaviour as the buffer size grows, and obtain a simplified formula for large buffers. Thirdly, we present numerical results for different system parameterizations as well as the comparison with simulation results. Then we show results of measurements of the burst ratio in the networking laboratory. Finally, we draw conclusions on the influence (or lack of it) of several factors on the burst ratio.

Keywords Queueing system · Networking · Packet losses · Burst ratio · Loss ratio

1 Introduction

In the most popular networks, based on the TCP/IP protocol stack, a significant portion of packets is lost in network nodes (routers). This is connected with the TCP protocol design, which increases the end user transmission rate, until the buffer in one of the routers on the transmission path gets full and loses arriving packets.

These losses do not cause any practical damage when we are, for instance, downloading an e-book from the Internet. This is due to the fact, that every packet lost in the network is retransmitted by the TCP protocol, which guarantees delivery of complete data.

A different situation we face however, when we need the packets to be delivered immediately, in real time, like e.g., in the Internet telephony, Internet television, videoconferencing, etc. In such applications, packet losses may cause a significant deterioration of the quality of transmission perceived by end users.

When we want to characterize packet losses that occur at a network node, the first, natural characteristic we think of is the loss ratio. It is defined as the number of packets lost

at the node, divided by the total number of packets arriving to the node, both measured in a long time interval. In this paper, the loss ratio is denoted by L . (Sometimes, instead of the loss ratio, the loss probability is used, but given the long observation time, these characteristics are equal).

The second very important characteristic of the loss process is the burst ratio, defined in [1]. The burst ratio is equal to the average observed length of series of lost packets, divided by the theoretical average length of series of lost packets, expected for a pure random loss process. By a pure random loss process we mean the Bernoulli process, i.e., the sequence of binary, independent and identically distributed random variables $A_1, A_2, A_3, \dots$, for which $\mathbf{P}(A_i = 0) = L$ and $\mathbf{P}(A_i = 1) = 1 - L$.

In this paper, the burst ratio will be denoted by B , while the average observed length of series of lost packets, by $\bar{G}$.

From the basic course of the probability theory, we know that the average length of series of losses, expected for a random loss, is equal to:

$$\frac{1}{1 - L}.$$

Therefore, the burst ratio can be also presented as:

$$B = \frac{\bar{G}}{\frac{1}{1-L}} = \bar{G}(1 - L). \quad (1)$$

✉ Andrzej Chydzinski
andrzej.chydzinski@polsl.pl

¹ Institute of Informatics, Silesian University of Technology,
Akademicka 16, 44-100 Gliwice, Poland

To illustrate the burst ratio, let us consider the following simple example of stream of packets passing through a network node:

$f f X f f f f f X X f f f f X X X f$,

where f denotes a packet which was successfully forwarded, while X , a packet lost at the node. As we can see, in this stream we have three subseries of lost packets: X , XX , and XXX , of lengths 1, 2 and 3, respectively. Therefore, the average length of series of lost packets is $\bar{G} = (1 + 2 + 3)/2 = 2$. We also have the loss ratio in the stream of $L = 6/18 = 1/3$. On the other hand, the theoretical average length of series of lost packets (i.e., the average length of series of zeroes in the Bernoulli process) for $L = 1/3$ is $1/(1 - L) = 1.5$. Therefore, the burst ratio in our stream equals $B = 2/1.5 = 1.333$.

Naturally, $B > 1$ means that the losses have a higher tendency to group together, than in a pure random loss. It is also possible that $B < 1$. In such case, the losses are more scattered over the time axis, than they would be in the pure random loss.

Now, the burst ratio is a very important characteristic, especially when real-time multimedia transmissions are considered, due to the following reason. Some fraction of lost packets can be quite well tolerated by end users, due to compensation mechanisms of multimedia codecs and due to human perception characteristics, but not when long bursts of lost packets occur frequently. In fact, the burst ratio may have, in some cases, a higher impact on the transmission quality, than the loss ratio.

This effect can be presented using numbers. For instance, the following impairment factor for the packetized voice transmission is used in [2]:

$$I = I_e + (95 - I_e) \frac{100L}{100L/B + R},$$

where I_e and R are some constants, unimportant in this context.¹ Using default values, $I_e = 0$ and $R = 4.3$ (see [2], p. 9), we may deduce, for instance, that the quality of the voice transmission is worse in the case with $L = 2\%$ and $B = 2$, than in the case with $L = 3\%$ and $B = 1$.

Given the importance of the burst ratio, it is natural to ask, how its value can be analytically computed and which model of the packet loss should be used for this purpose. It seems natural to choose a model that can mimic, as precisely as possible, the actual process of losing packets at a router. As already mentioned, in wired networking, by far the most frequent reason of packet loss is the queueing mechanism, and buffer overflows in particular. Namely, a packet gets deleted,

when upon its arrival at the router, the buffer of the proper output interface is full, and the packet cannot enter the queue. Therefore, the most natural way to compute the burst ratio is by using the model of packet queueing with limited buffer capacity.

In this paper, we study the burst ratio in the well-known and commonly used $M/G/1/N$ queueing model, i.e., with packet arrivals modeled by the Poisson process, arbitrary distribution of the service time and the buffer of size N . (The arbitrary distribution of the service time will allow us to model an arbitrary distribution of the packet size in the arrival stream).

After introducing all the necessary definitions and notations (Sect. 2), we derive the exact formula for the burst ratio in the $M/G/1/N$ queue (Sect. 3). Then we study its behaviour as $N \rightarrow \infty$ and obtain the limiting formula, which can be used as an approximation for large buffers (Sect. 4). Thirdly, we present numerical results for different system parameterizations, as well as their comparison with simulation results (Sect. 5). In particular, using numerical results, we check the dependence of the burst ratio on the variance of the service time, the system load and the buffer size. We also check the convergence speed to the limiting formula, as the buffer grows. Then we present and discuss experimental values of the burst ratio, obtained in the laboratory network (Sect. 6). Finally, we conclude the paper, by summarizing the impact (or lack of it) of different factors on the burst ratio (Sect. 7) and commenting on the possibilities of the future work.

1.1 Related work

To the best of the authors' knowledge, there are no published results on the burst ratio in any queueing model, $M/G/1/N$ in particular.

The previous studies on the burst ratio were based on black-box models of the packet loss process. In such models, the actual reason and mechanism of loss is not modeled. Instead, only the bare statistical properties of the loss process are modeled by a more or less advanced stochastic process, usually Markovian. The most popular process of this type is the two-state Markov chain, [4]. This model can be parameterized by two numbers, p and q , where p is the probability of losing a packet and changing the state, if the previous packet was accepted, while q represents the probability of accepting a packet and changing the state, if the previous packet was lost. A slightly more advanced packet loss model is the Gilbert model, [5–7]. It also uses the Markov chain with two states, say *good* and *bad*, but unlike in the previous model, the change of the state is not directly translated into packet acceptance or loss. Instead, the packet loss occurs with some probability, b , only when the chain is in state *bad*. Therefore, the Gilbert model requires three parameters, p , q and

¹ I_e is called the equipment impairment factor at zero packet-loss, while R is called the packet-loss robustness factor. See [3] for their definitions and values.

b. Another modification is the Gilbert-Elliot model, [8,9], in which it is allowed to lose a packet in state *good* as well, with probability g . The four-state Markov chain model, [10], extends further the Gilbert-Elliot model, by allowing to lose or accept packets in one of the following states: *good-accept*, *good-lose*, *bad-accept*, *bad-lose*. Finally, in the general, k -state Markov chain model, [4,7,11], at least k consecutive packets are lost, when the chain is in state k .

In all the aforementioned models, the burst ratio can be computed directly, using the pure Markovian structure of the loss process. In particular, the burst ratio in the two-state Markov model was studied in [1,12,13]. The main formula for the burst ratio was derived in [1], while in [12,13], the formula for the burst ratio in the case of several concatenated channels, each modeled by the two-state Markov chain, was obtained and compared with simulation results.

Generally speaking, the advantage of Markovian loss models is their simplicity and basic ability to capture the correlation between consecutive packet losses. They are unable, however, to mimic precisely the structure of consecutive losses, caused by the queueing mechanism and buffer overflows. For instance, in the two-state Markovian model, the size of the series of lost packets has geometric distribution. This makes possible to mimic the average size only, while mimicking a large variance is impossible. (More on disadvantages of Markov chain models in packet loss modeling can be found in [14]).

As for the packet loss ratio, it is considered to be one of the most important performance indicators for Internet Service Providers (ISPs), [15]. Therefore, several papers have been published on the loss ratio measurements in the Internet, see e.g., [15–19]. Analytical formulas for the loss ratio in queueing models with finite buffers and different arrival processes can be found in [20–22]. In particular, in [20] the formula for the case of Poisson arrivals is shown, in [21]—the formula for the case of Markov-modulated arrivals, while in [22]—for the case of batch Markovian arrivals.

Another characteristic of the loss process, studied by some researchers, is the probability that in a block of n consecutive packets, j packets are lost. More on this subject can be found in [14,23].

As for the methodology used in this paper, it is based on Lemma 3.2.1 of [24], which gives the general solution of a class of systems of equations, using recurrent sequences of type (5), called potentials. For other examples of the usage of this method in solving queueing systems, we refer the reader to [25,26].

2 Queueing model and notations

In this study, we consider a single-server queueing system (i.e., with one service station), whose arrival process is given

by the Poisson process of rate λ . The service time is distributed according to distribution function $F(t)$, which is not further specified and can assume any form. The capacity of the buffer (waiting room) is finite and equal to N , including the packet in service. If a packet arrives when the buffer is full (i.e., there are N packets already present in the system), it is deleted and lost. If not stated otherwise, it is assumed that the buffer size is non-zero. (The system with no buffering space will be considered separately).

In the Kendall's notation of queueing systems, such system is denoted by $M/G/1/N$.

The queueing discipline is irrelevant in this study, therefore either FIFO (First In First Out) or FCLS (First Come Last Served) can be assumed.

The queue length at time t is denoted by $X(t)$. We assume that $X(t)$ includes the service position, if occupied. It is also assumed that if at $t = 0$ the queue is not empty, then the time origin corresponds to a service completion epoch.

In the analysis, we will be using frequently the probability that k packets arrive during one service time, denoted by a_k . Naturally, we have:

$$a_k = \int_0^\infty \frac{e^{-\lambda u} (\lambda u)^k}{k!} dF(u), \quad k = 0, 1, 2, \dots \quad (2)$$

The load offered to the queueing system, ρ , is defined as:

$$\rho = \lambda \int_0^\infty t dF(t).$$

$\mathbf{P}(\cdot)$ denotes probability, while γ_i denotes the number of packet losses during the i -th buffer overflow period. The buffer overflow period is the time interval, at the beginning of which the queue size grows from $N - 1$ to N , while at the end decreases from N to $N - 1$, remaining equal to N in the middle. All packets arriving during an overflow period are lost.

3 Burst ratio in the finite-buffer queue

Having defined the queueing model, we can formulate and prove the main result of the paper. It states the following.

Theorem 1 *In the $M/G/1/N$ queueing system the burst ratio is equal to:*

$$B = \frac{\sum_{k=0}^{N-1} r_k}{[1 - g(0)] \left(1 + \rho \sum_{k=0}^{N-1} r_k\right)} \sum_{l=1}^{\infty} l g(l), \quad (3)$$

where

$$g(l) = \frac{\sum_{k=1}^N R_{N-k} a_{k+l} - \sum_{k=1}^{N-1} R_{N-1-k} a_{k+l}}{\sum_{k=0}^N R_{N-k} a_k - \sum_{k=0}^{N-1} R_{N-1-k} a_k}, \quad (4)$$

$$R_0 = 0, \quad R_1 = \frac{1}{a_0},$$

$$R_{k+1} = R_1 \left(R_k - \sum_{i=0}^k a_{i+1} R_{k-i} \right), \quad k \geq 1, \quad (5)$$

$$r_0 = 1, \quad r_1 = \frac{1}{a_0} - 1,$$

$$r_{k+1} = \frac{1}{a_0} \left(r_k - \sum_{i=0}^{k-1} a_{i+1} r_{k-i} - a_k \right), \quad k \geq 1. \quad (6)$$

Proof of Theorem 1 The proof will be presented in three parts. In the first part, the distribution of the number of losses in the first buffer overflow period will be found. In the second part, the average length of series of lost packets will be obtained in the steady state. In the third part, the formula for the loss ratio will be recalled and exploited.

We start with finding the distribution of the number of consecutive packet losses in the first buffer overflow period, i.e., the distribution of γ_1 . Obviously, this distribution depends on the number of packets present in the system at $t = 0$.

Assume first that the system is initially not empty, i.e., $X(0) > 0$. Using the continuous formula of total probability, with regards to the first departure time, we obtain:

$$\mathbf{P}(\gamma_1 = l | X(0) = n)$$

$$= \sum_{k=0}^{N-n-1} \int_0^\infty \mathbf{P}(\gamma_1 = l | X(0) = n+k-1) \frac{e^{-\lambda u} (\lambda u)^k}{k!} dF(u)$$

$$+ \int_0^\infty \frac{e^{-\lambda u} (\lambda u)^{N-n+l}}{(N-n+l)!} dF(u), \quad 0 < n < N. \quad (7)$$

The first integral in (7) covers the case, in which the buffer does not get full before the first departure time. In such a case, there must be no more than $N-n-1$ new arrivals during the first service time. The second integral in (7) covers the case, in which the buffer does get full during the first service time. This can happen only if there are $N-n+l$ new arrivals before the first departure time, where l is the number of packets lost in the first overflow period.

Now assume that the system is initially empty, i.e., $X(0) = 0$. In this case we simply have:

$$\mathbf{P}(\gamma_1 = l | X(0) = 0) = \mathbf{P}(\gamma_1 = l | X(0) = 1). \quad (8)$$

Formula (8) follows from the fact that, starting from an empty queue, the system must pass through the state $X(0) = 1$, before the buffer can get full.

Using the abbreviation:

$$q_n(l) = \mathbf{P}(\gamma_1 = l | X(0) = n), \quad (9)$$

from (7) we obtain:

$$q_n(l) = \sum_{k=0}^{N-n-1} a_k q_{n+k-1}(l) + a_{N-n+l}, \quad 0 < n < N, \quad (10)$$

while from (8) we get:

$$q_0(l) = q_1(l). \quad (11)$$

Substituting $\tilde{q}_n(l) = q_{N-n}(l)$ in (10) and (11) we have:

$$\sum_{k=-1}^{n-1} a_{k+1} \tilde{q}_{n-k}(l) - \tilde{q}_n(l) = w_n(l), \quad 0 < n < N, \quad (12)$$

and

$$\tilde{q}_N(l) = \tilde{q}_{N-1}(l), \quad (13)$$

respectively, with:

$$w_n(l) = a_n \tilde{q}_1(l) - a_{n+l}. \quad (14)$$

According to Lemma 3.2.1 of [24], every solution of system (12) has the form:

$$\tilde{q}_n(l) = c(l) R_n + \sum_{k=1}^n R_{n-k} w_k(l), \quad n \geq 1, \quad (15)$$

where R_k is given in (5) and $c(l)$ is a function which does not depend on n . (Sequence R_k is called *potential* for distribution a_k). The task now is to find unknown $c(l)$ and $\tilde{q}_1(l)$. Substituting $n = 1$ into (15) we obtain:

$$c(l) = a_0 \tilde{q}_1(l) \quad (16)$$

and

$$\tilde{q}_n(l) = \tilde{q}_1(l) \sum_{k=0}^n R_{n-k} a_k - \sum_{k=1}^n R_{n-k} a_{k+l}. \quad (17)$$

Then, putting $n = N$ and $n = N-1$ into (17) and exploiting (13) yields:

$$\tilde{q}_1(l) = g(l), \quad (18)$$

where $g(l)$ is defined in (4). Replacing back $q_n(l) = \tilde{q}_{N-n}(l)$, from (17) and (18) it follows the form of the distribution of the number of losses in the first buffer overflow period:

$$q_n(l) = g(l) \sum_{k=0}^{N-n} R_{N-n-k} a_k$$

$$- \sum_{k=1}^{N-n} R_{N-n-k} a_{k+l}, \quad 0 \leq n < N, \quad l = 0, 1, 2, \dots \quad (19)$$

This finishes the first part of the proof.

In the second part of the proof, we want to derive the average length of series of packets, lost one after another, in the steady state. For this purpose, we can use formula (19), but with some modifications.

Formula (19) concerns only the first overflow period. In order to compute the distribution for other overflow periods, we may notice that just after an overflow period is finished, the system must assume, at least for a brief time, the queue size of $N - 1$. Moreover, just after an overflow period is finished, the service and arrival processes are reset (the former by the definition of the queueing model, the latter by the properties of the Poisson process). Therefore, the distribution of the number of packets lost in every overflow period, except for the first one, is the same as the distribution of the number of packets lost in the first overflow period with the initial queue size of $N - 1$. We can formally note this as:

$$\mathbf{P}(\gamma_i = l) = \mathbf{P}(\gamma_1 = l | X(0) = N - 1) = q_{N-1}(l), \quad i \geq 2. \quad (20)$$

Using (19) with $n = N - 1$, we obtain $q_{N-1}(l) = g(l)$, which leads to:

$$\mathbf{P}(\gamma_i = l) = g(l), \quad i \geq 2. \quad (21)$$

As distribution (21) is the same for every $i \geq 2$, it is also the steady-state distribution of the number of losses in the overflow period.

Now, pay attention that the number of losses in the overflow period is not always equal to the length of series of packets lost one after another. The only difference is that the former can be zero (no losses in the overflow period), while the latter cannot be zero (we count the series only after at least one packet loss). To obtain the distribution of the length of series of lost packets, we have to exclude the case $l = 0$ and normalize the remaining distribution $g(l)$, $l = 1, 2, \dots$, using factor $1/(1 - g(0))$.

Therefore, defining $G(l)$, $l = 1, 2, \dots$, as the probability that the length of lost series of packets in the steady state is l , we have:

$$G(l) = \frac{g(l)}{1 - g(0)}, \quad l = 1, 2, \dots, \quad (22)$$

with $g(l)$ given in (4). Finally, the average length of series of packets lost one after another is:

$$\bar{G} = \sum_{l=1}^{\infty} l G(l) = \frac{1}{1 - g(0)} \sum_{l=1}^{\infty} l g(l), \quad (23)$$

what finishes the second part of the proof.

In the third part of the proof, we have to recall some known results on the $M/G/1/N$ queue, needed to present the loss

ratio in a compact form. One of the ways to compute the loss ratio is based on the steady-state probability that the queue is empty, i.e.,:

$$P_0 = \lim_{t \rightarrow \infty} \mathbf{P}(X(t) = 0). \quad (24)$$

It is known that for the $M/G/1/N$ system it holds:

$$P_0 = \frac{1}{1 + \rho \sum_{k=0}^{N-1} r_k}, \quad (25)$$

where r_k is defined in (6) (see formulas (1.18a), (1.12) and (1.11b) from [20], chapter 5).

On the other hand, the loss ratio can be computed as (see eg. [27], p. 10):

$$L = 1 - \frac{1 - P_0}{\rho}, \quad (26)$$

which in the case of the $M/G/1/N$ system gives:

$$L = 1 - \frac{\sum_{k=0}^{N-1} r_k}{1 + \rho \sum_{k=0}^{N-1} r_k}. \quad (27)$$

Now we can easily finish the proof. Namely, combining (1) with (28), (23) and (27) finishes the proof of Theorem 1. $\square$

3.1 Special case: no buffer

In the queueing literature, a system with no buffering space is also considered. (In our notation, the lack of buffering space means $N = 1$, as the necessary service position is included in N in our notation).

The burst ratio in such system can be computed quickly by observing that an arriving packet is dropped if and only if the service position is occupied. We know that the distribution of the number of packets arriving during one service period is a_k , $k = 0, 1, \dots$. Therefore, the probability that the length of series of lost packets equals l is:

$$G(l) = \frac{a_l}{1 - a_0}, \quad l = 1, 2, \dots \quad (28)$$

Its average value is then:

$$\bar{G} = \frac{1}{1 - a_0} \sum_{l=1}^{\infty} l a_l. \quad (29)$$

On the other hand, from (27) we get:

$$L = 1 - \frac{1}{1 + \rho}. \quad (30)$$

Using (1), we obtain the final result.

Corollary 1 *In the system without buffer, the burst ratio is equal to:*

$$B = \frac{1}{(1 - a_0)(1 + \rho)} \sum_{l=1}^{\infty} l a_l. \quad (31)$$

3.2 Computational remarks

It can be noticed, that for small and moderate buffer sizes, formula (3) is easy in computation. Firstly, distribution a_k is well known in the queueing theory and easy to derive symbolically for several classes of distribution functions F . In other cases, it can be obtained via numerical integration. Secondly, there is an infinite sum in (3), but it is rare that we have to compute more than a hundred of its components. This is due to the fact, that in practice (e.g., computer networking) we deal with small to moderate loss ratios (say 0.5 or smaller). The number of non-negligible components in the considered infinite sum gets large for large loss ratios (e.g., 0.9999). In practice however, we rarely meet such queueing systems—it makes very little sense to exploit a system which losses 99.99 percent of its customers (packets, jobs, etc.).

For large N , recursions (5) and (6) can get numerically unstable. Fortunately, this problem has minor practical consequences for calculations of the burst ratio in networking. Most importantly, we will show that (3) converges, as N grows, to the limiting burst ratio, and the convergence is rather fast. Therefore, a good approximation of the burst ratio can be obtained using the limiting value, even for a relatively small buffer size, far smaller than makes (5) and (6) unstable. Alternatively, two other approaches can be adopted. Firstly, the precision of numbers used in calculations can be increased—most popular software packages for computations offer arbitrary-precision numbers. Secondly, the solution of system (12), based on a different type of recursion, (e.g., derived using the censored Markov chains), can be searched for.

4 Burst ratio for a large buffer

In this section we will compute the limit of the burst ratio when the buffer size grows to infinity. The limit will be denoted by B_{∞} . This limit is interesting not only from the theoretical point of view, but has also some practical applications. As we will see, B_{∞} has a simpler form than (3), so it can be used for quick approximations of the burst ratio. Moreover, as shown via numerical examples, such approximation is often very good, even for as small buffers as $N = 100$.

Before the theorem is presented and proven, we have to make one assumption. Let $f(s)$ denote the Laplace transform of the service time distribution, i.e.,:

$$f(s) = \int_0^{\infty} e^{-st} dF(t). \quad (32)$$

From now on we assume that if $\rho \neq 1$, then there exists a solution of the equation $f(\lambda - \lambda x) = x$, denoted by x_0 , which is positive and different than 1. As we can check, this assumption is not very restrictive. In fact, it can be easily shown that if $\rho > 1$, then the solution always exists and is smaller than 1. Namely, for $\rho > 1$ we have:

$$\begin{aligned} b(x) &= d f(\lambda - \lambda x) / d x = -\lambda f'(\lambda - \lambda x), \\ b(1) &= -\lambda f'(0) = \rho > 1. \end{aligned} \quad (33)$$

Moreover, $f(\lambda) > 0$. Therefore, the graph of the function $y = f(\lambda - \lambda x)$ is below the graph of $y = x$ for $x = 1 -$, and above the graph $y = x$ for $x = 0 +$. As function $y = f(\lambda - \lambda x)$ is continuous, it must cross the graph $y = x$ somewhere in interval $(0, 1)$.

For $\rho < 1$ the solution x_0 may not exist. Fortunately, this happens only for some special types of functions F , eg. with infinite variance. In most practical cases of F , x_0 does exist.

If $\rho = 1$, then we assume that $x_0 = 1$. Now we can formulate the theorem.

Theorem 2 *In the M/G/1/N queueing system the limiting burst ratio as $N \rightarrow \infty$ is equal to:*

$$B_{\infty} = \frac{\min\{1, \rho^{-1}\}}{1 - h(0)} \sum_{l=1}^{\infty} l h(l), \quad (34)$$

where

$$h(l) = \frac{1}{x_0^{l+1}} f(\lambda - \lambda x_0) - \frac{1}{x_0^{l+1}} \sum_{j=0}^l x_0^j a_j. \quad (35)$$

Proof of Theorem 2 The proof consists of three parts, devoted to $\rho > 1$, $\rho < 1$ and $\rho = 1$, respectively.

We start with case $\rho > 1$. Unfortunately, in this case it is hard to prove the theorem by direct computation of the limit of (3) as $N \rightarrow \infty$. Therefore, we have to use an alternative technique.

Namely, let us consider another queueing system, of $G/M/1$ type in the Kendall's notation. In the new system, the buffer is infinite, the interarrival time is distributed according to F , the service time is distributed exponentially with parameter λ , and the new system is not empty at the beginning, i.e., $X(0) > 0$. Moreover, the new system has one special feature: when it gets empty, the service is switched to another, independent queue (secondary queue), containing a very large (inexhaustible) number of packets. The secondary queue is served until a new packet of the primary stream arrives. Upon such arrival, the service of the secondary queue is immediately interrupted and switched back to the main queue, and so

on. Let δ_1 denote the number of packets from the secondary queue, served during the first period in which the primary queue is empty.

It is easy to see, that the load of such $G/M/1$ system is smaller than 1. Namely, we have $\rho' = 1/\rho < 1$. Due to this, the $G/M/1$ system is stable, and in the steady state the probability $\mathbf{P}(X(t) > Q)$ can be made arbitrary small, if Q is large enough. Because of that, the distribution of δ_1 , in the defined $G/M/1$ system with $X(0) = 1$, can be arbitrary close to the distribution of γ_1 in the $M/G/1/N$ system with $X(0) = N - 1$, if N is large enough. Thus the limiting distribution of γ_1 as $N \rightarrow \infty$ in the $M/G/1/N$ system is equal to the distribution of δ_1 in the $G/M/1$ system, i.e.,:

$$\begin{aligned} \lim_{N \rightarrow \infty} \mathbf{P}(\gamma_1^{M/G/1/N} = l | X(0) = N - 1) \\ = \mathbf{P}(\delta_1^{G/M/1} = l | X(0) = 1). \end{aligned} \quad (36)$$

Therefore, we are left with the task of finding the distribution of δ_1 in the $G/M/1$ system. Using the law of total probability, with respect to the first arrival time, we have:

$$\begin{aligned} \mathbf{P}(\delta_1 = l | X(0) = n) \\ = \sum_{k=0}^{n-1} \int_0^\infty \mathbf{P}(\delta_1 = l | X(0) = n - k + 1) \frac{e^{-\lambda u} (\lambda u)^k}{k!} dF(u) \\ + \int_0^\infty \frac{e^{-\lambda u} (\lambda u)^{n+l}}{(n+l)!} dF(u), \quad n \geq 1. \end{aligned} \quad (37)$$

The first integral in (37) covers the case, in which the primary queue does not get empty before the first arrival. This requires no more than $n-1$ service completions before the first arrival. The second integral in (37) covers the case, in which the primary queue does get empty before the first arrival. This requires $n+l$ service completions, where l is a non-negative integer.

Denoting:

$$h_n(l) = \mathbf{P}(\delta_1 = l | X(0) = n), \quad (38)$$

from (37) we obtain:

$$h_n(l) = \sum_{k=0}^{n-1} a_k h_{n-k+1}(l) + a_{n+l}, \quad n \geq 1. \quad (39)$$

Rearranging (39) we get then:

$$\sum_{k=-1}^{n-1} a_{k+1} h_{n-k}(l) - h_n(l) = v_n(l), \quad n \geq 1, \quad (40)$$

with

$$v_n(l) = a_n h_1(l) - a_{n+l}. \quad (41)$$

Applying Lemma 3.2.1 of [24], the general solution of system (40) has the form:

$$h_n(l) = d(l) R_n + \sum_{k=1}^n R_{n-k} v_k(l), \quad n \geq 1, \quad (42)$$

where $d(l)$ does not depend on n . Putting $n = 1$ into (42) yields $d(l) = h_1(l) a_0$. Then, from (42) we obtain:

$$h_n(l) = h_1(l) \sum_{k=0}^n R_{n-k} a_k - \sum_{k=1}^n R_{n-k} a_{k+l}. \quad (43)$$

Now we are reduced to finding $h_1(l)$. In order to do this, let us define the following function:

$$H(x, l) = \sum_{n=2}^{\infty} x^n h_n(l). \quad (44)$$

As $h_n(l)$ is a probability, for every $l \geq 0$ and $n \geq 0$ it holds $0 \leq h_n(l) \leq 1$. Due to this fact, the series in (44) converges if $|x| < 1$. Thus function $H(x, l)$ is well defined for every $|x| < 1$. On the other hand, using (43) and (5) yields:

$$H(x, l) = x \frac{h_1(l)x - \sum_{j=1}^{\infty} x^j a_{j+l}}{f(\lambda - \lambda x) - x}. \quad (45)$$

As it was shown at the beginning of this section, under the assumption $\rho > 1$, the denominator of (45) has a root, x_0 , such that $0 < x_0 < 1$. Therefore, x_0 must also be the root of the nominator. We have:

$$\begin{aligned} h_1(l) = \frac{1}{x_0} \sum_{k=1}^{\infty} x_0^k a_{k+l} = \frac{1}{x_0^{l+1}} f(\lambda - \lambda x_0) \\ - \frac{1}{x_0^{l+1}} \sum_{j=0}^l x_0^j a_j = h(l), \end{aligned} \quad (46)$$

where $h(l)$ is given in (35).

Now we can get back to the $M/G/1/N$ system. As has been argued, (46) presents the limiting distribution of the number of consecutive packet losses during the overflow period as $N \rightarrow \infty$. In order to calculate the limiting burst ratio, we have to compute the limiting loss ratio first. Under the assumption $\rho > 1$, we have $P_0 \rightarrow 0$ as $N \rightarrow \infty$. Therefore, from (26) we can conclude $L \rightarrow 1 - \frac{1}{\rho}$ as $N \rightarrow \infty$.

Finally, excluding the case in which there are no losses during the overflow period and normalizing the remaining distribution $h(l)$, $l = 1, 2, \dots$, using factor $1/(1 - h(0))$, we obtain:

$$B_{\infty} = \frac{\rho^{-1}}{1 - h(0)} \sum_{l=1}^{\infty} l h(l), \quad (47)$$

which finishes the proof of (34) in the case of $\rho > 1$.

Now, let us assume that $\rho < 1$. In this case it holds $x_0 > 1$, and potential (5) have a simple form, namely:

$$R_n = \frac{1}{1-\rho} - \frac{b}{x_0^n} + o(x_0^{-n}), \quad b = -\frac{1}{\lambda f'(\lambda - \lambda x_0) + 1}. \quad (48)$$

(see formula (22) in [28]). Using (48), we can directly show that $h(l)$ is the limit of (4) as $N \rightarrow \infty$.

Moreover, it is well known that for $\rho < 1$, the loss ratio decreases to 0, as $N \rightarrow \infty$. Therefore, in this case we have:

$$B_\infty = \frac{1}{1-h(0)} \sum_{l=1}^{\infty} lh(l), \quad (49)$$

which finishes the proof of (34) for $\rho < 1$.

Finally, in the case $\rho = 1$ the proof is straightforward, as it is easy to see that $x_0(\rho) \rightarrow 1$ as $\rho \rightarrow 1$. $\square$

4.1 Computational remarks

Obviously, (34) is easy in numerical computations. As was argued in the previous section, in most practical cases the infinite sum in (34) consists of a small number of non-negligible components. Root x_0 can be found easily, with arbitrary precision, using one of the numerical methods of solving equations.

5 Numerical examples

The purpose of this section is four fold. Firstly, we want to demonstrate the practical usability of Theorems 1 and 2, by showing numerical results for different system parameterizations. Secondly, we want to observe the dependence of the burst ratio on crucial system parameters. In particular, the influence of the variance of the service time, the load of the system and the size of the buffer, on the burst ratio, will be tested. Thirdly, we want to check the convergence of (3) to (34), i.e., to check, for what buffer sizes we can approximate (3) by (34). Finally, we want to compare the values of burst ratio obtained from analytical formulas with values obtained in simulations.

In all numerical examples, it is assumed that packets arrive at the router's queue according to the Poisson process, form the queue in the buffer, and are served (transmitted) by an output link of 1Gb/s throughput. The packets in the arrival stream may have different sizes, following some distribution. As the throughput of the output link is constant, different packet sizes are equivalent to different transmission times, i.e., distribution F in our model. For example, a packet of size 1500 bytes needs 12 μ s of the service time, a packet of

Table 1 Loss ratio and burst ratio for different packet size distributions

Packet size distribution	Loss ratio	Burst ratio
Constant, (a)	0.005008	1.35233
Uniform, (b)	0.006489	1.51717
Exponential, (c)	0.009901	1.98019
Two-point, (d)	0.052439	6.18027

size 40 bytes—only 0.32 μ s, etc. Therefore, instead of the service time distribution, the packet size distribution can be given in the system parameterization.

5.1 Burst ratio versus the variance of the service time

To study the influence of the variance of the service time, we can calculate the burst ratio in the following four cases of the packet size distribution:

- (a) constant packet size of 770 bytes,
- (b) uniform distribution of the packet size in range [40, 1500] bytes;
- (c) exponential distribution of the packet size with the average value of 770 bytes;
- (d) two-point distribution; packets of size 40 bytes or 9216 bytes, with probabilities 0.920445 and 0.079555, respectively.

As we can see, the average packet size is 770 bytes in all four cases, but the standard deviation is different and equals 0, 422, 770 and 2483 bytes in distributions (a), (b), (c) and (d), respectively. The packet arrival rate is assumed to be $\lambda = 162337.66$ packets per second, which is equivalent to the total arrival rate of 1Gb/s. Therefore, we have $\rho = 1$. The buffer size is $N = 100$ packets.

In Table 1, the loss ratio and the burst ratio for packet size distributions (a)–(d) are presented. Similarly, in Fig. 1 the distribution of the length of the series of lost packets, $G(l)$, is depicted for packet size distributions (a)–(d). As we can see, the higher the variance of the service time, the larger the loss ratio and the burst ratio. This is not very surprising—to lose a long series of packets, we need a long service time. It is worth noticing however, that the burst ratio may assume very high values, in relatively simple examples. In example (d), the average length of series of packets lost one after another is over 6. Such value would have a deep impact on several protocols and applications.

Analyzing Table 1, one may conjecture that a high value of the burst ratio always comes together with a high value of the loss ratio. Such conjecture is wrong. This will be shown

Fig. 1 Distribution of the length of series of packets lost one after another. The top-left corner: constant packet sizes, the top-right corner: uniform packet sizes, the bottom-left corner: exponential packet sizes, the bottom-right corner: two-point packet sizes

in the next example, in which the influence of the system load on the burst ratio will be checked.

5.2 Burst ratio versus the system load

In this subsection, we chose one packet size distribution, say (c), and vary the Poisson arrival rate, so that the load of the system varies from the severe underload ($\rho = 0.5$) to the severe overload ($\rho = 2$). To lose some packets in the case of a very small load, a small buffer is required as well. Therefore we assume $N = 10$.

The resulting loss ratio and burst ratio as functions of ρ are depicted in Fig. 2. As was to be expected, the loss ratio changes drastically when ρ changes from 0.5 to 2. Namely, for $\rho = 0.5$ we have $L = 0.000489$, while for $\rho = 2$, we get $L = 0.5002$. At the same time, the burst ratio changes very little, keeping its value between 1.499 and 1.818. We checked other service time distributions as well and obtained similar results—the behaviour of the burst ratio observed in Fig. 2 is not specific to distribution (c).

The fact that the burst ratio depends little on the system load is a consequence of its definition (i.e., the average observed length of series of lost packets, divided by the theoretical average length of series losses, expected for a random loss). If the load grows, then the average length of series of losses, expected for a random loss, grows as well. Therefore,

the growth of the denominator compensates the growth of the nominator. This was perhaps the intention of the inventor of the burst ratio, i.e., that it should reflect the length of series of losses caused by other reasons, not the load.

5.3 Burst ratio versus the buffer size and the convergence to the limit

In this subsection, we chose one packet size distribution, say (d), and change the buffer size (i.e., $N = 10, 20, 50, 100, 200, 500$). First of all, this is meant to check the dependence of the burst ratio on the buffer size. Secondly, we can check the convergence rate of B to its limit, B_∞ . The calculations are carried out for an underloaded system ($\rho = 0.8$), critically loaded system ($\rho = 1$) and an overloaded system ($\rho = 1.2$).

The results are presented in Table 2. The first row is obtained using Corollary 1, while the last row—using Theorem 2.

Firstly, we can see that the burst ratio grows with the buffer size. However, the influence of the buffer size on the burst ratio is moderate—in the presented results, its value changes by 33–67%, when the buffer size changes from zero to ∞ . Note that the results for no buffering space are consistent with the other results—the burst ratio reaches the smallest value, but only a little smaller than for $N = 10$.

Fig. 2 Dependence of the loss ratio (on the left) and the burst ratio (on the right) on the system load

Table 2 Burst ratio for different buffer sizes and system loads

	$\rho = 0.8$	$\rho = 1$	$\rho = 1.2$
No buffer	3.79808	3.96355	4.03497
$N = 10$	3.89041	4.47650	5.10242
$N = 20$	4.54017	5.24337	5.77931
$N = 50$	4.95339	5.88887	6.45808
$N = 100$	5.04321	6.18027	6.69954
$N = 200$	5.05246	6.34416	6.75991
$N = 500$	5.05255	6.44921	6.76360
$N = \infty$	5.05255	6.52229	6.76360

Secondly, we can observe that the burst ratio converges quite quickly to the limiting value. Depending on the required accuracy, we can approximate the exact value of B by B_∞ , for as low value of N , as a hundred packets. We should be careful, however, in cases where the load is close to 1. In such cases the convergence is slower, as it follows from Table 2.

5.4 Simulation versus analytical results

In this set of experiments, we verified our analytical results using simulations. For this purpose, the Omnet++ simulator, [29], was used. Ten different scenarios were used in simulations. In every case, the output link of 1 Gb/s capacity was simulated, but other system parameters were changed. In particular, all packet size distributions from Sect. 5.1 were simulated. These distributions were combined with different buffer sizes (from 10 to 500 packets) and with different arrival rates (from 500 Mb/s to 1.5 Gb/s). During every simulation, empirical values of the average length of the lost series of packets, as well as the number of lost packets, were collected, while about 10^7 packets were passing through the queue in the network node.

The simulation results are presented in the third column of Table 3, with 95% confidence intervals. For comparison,

the analytical values, computed using Theorem 1, are given in the second column.

As we can see, for all the scenarios, the computed burst ratio agrees very well with the simulated value, and it lays inside the confidence interval. The confidence intervals are rather small—in simulations, it is easy to collect a large number of samples. The largest (but still quite small) confidence interval was obtained in the scenario with $\rho = 0.5$. This is easy to understand by noticing that in such an underloaded system, packet losses occur rarely. Therefore, relatively few series of losses were observed in this scenario (about 3000 series).

6 Real network measurements

6.1 Equipment

In addition to numerical calculations and discrete-event simulations, we conducted laboratory experiments with the real traffic, using prepared for this purpose testbed, presented in Fig. 3, consisting of:

- traffic generator—Spirent SPT-N4U, [30], with MX2-10G-S12 load module [31], equipped with twelve fiber optic ports, each configurable to 1 Gb/s or 10 Gb/s,
- device under test—layer 3 switch/router Cisco 3750X [32], equipped with twelve 1 Gb/s and two 10 Gb/s ports,
- three servers for collecting data and computing characteristics.

The traffic was generated on ports 1–4 of Spirent SPT-N4U, for the purpose of tests set to 1 Gb/s each. The generated traffic was based on the most popular protocol stack - Ethernet in layer 2, IPv4 in layer 3 and TCP/UDP in layer 4. Spirent's default TCP congestion control, i.e., New Reno with 32768 bytes of the maximum window size, was used. In layer 7, the HTTP traffic was emulated, with Apache web-

Table 3 Burst ratios computed using Theorem 1 and obtained in simulations (with 95% confidence intervals)

System parameters	Analytical burst ratio	Simulated burst ratio
Constant pkt. sizes, $N = 100$, $\rho = 1$	1.35233	1.35402 ± 0.00630
Uniform pkt. sizes, $N = 100$, $\rho = 1$	1.51717	1.52993 ± 0.00808
Exponential pkt. sizes, $N = 100$, $\rho = 1$	1.98019	1.98302 ± 0.01281
Two pkt. sizes, $N = 100$, $\rho = 1$	6.18027	6.18538 ± 0.02623
Uniform pkt. sizes, $N = 200$, $\rho = 1.2$	1.38468	1.38584 ± 0.00152
Uniform pkt. sizes, $N = 500$, $\rho = 1.2$	1.38468	1.38399 ± 0.00151
Exponential pkt. sizes, $N = 10$, $\rho = 0.5$	1.49926	1.48352 ± 0.02949
Exponential pkt. sizes, $N = 10$, $\rho = 1.5$	1.65691	1.65731 ± 0.00206
Two-point pkt. sizes, $N = 20$, $\rho = 0.9$	4.91447	4.91201 ± 0.01157
Two-point pkt. sizes, $N = 50$, $\rho = 0.9$	5.47531	5.46945 ± 0.02245

Fig. 3 Laboratory setup

server on port 80 on the server side, and Mozilla browser on the client side.

From Spirent ports 1–4, the traffic was forwarded to ports 1–4 of the device under test. In this device, the whole incoming traffic was duplicated, using SPAN functionality, to output ports 8 and 13.

The copy of the incoming traffic sent to output port 13 was captured by Server 1, for further analysis.

Output port 8 was the main port under study, loaded with the whole incoming traffic from ports 1–4. At this port the losses, caused by buffer overflows, actually occurred.

The output traffic from port 8, thinned by the losses, was forwarded to port 9, and then duplicated again, using SPAN, to output ports 5 and 14.

The copy of the thinned traffic sent to port 14 was captured by Server 2, for further analysis.

The thinned traffic from port 5 was forwarded back to Spirent SPT-N4U.

DPDK-dump application [33] was used on Servers 1 and 2 to capture the copies of incoming and outgoing traffic, respectively, and insert them into a separate SQLite database on each server.

These two databases were then transferred to Server 3, where the actual comparisons and computation of loss parameters were performed. Using a separate server for computations, we offloaded the first two servers, which were collecting high traffic volumes. It also made possible collecting data in another traffic scenario, when results from the previous scenario were still being analyzed in Server 3.

Table 4 Experimental versus theoretical burst ratios in traffic scenarios with different number of TCP connections

Number of TCP flows	Link load, ρ	Loss ratio, L	Experimental burst ratio, B_e	Theoretical burst ratio, B_t	Error, $\frac{ B_t - B_e }{B_e}$
24	0.941	0.00095	1.30067	1.34307	3.3%
32	0.943	0.00302	1.29924	1.34397	3.4%
48	0.947	0.00687	1.33702	1.34576	0.7%
64	0.949	0.00983	1.34827	1.34665	0.1%
96	0.954	0.01447	1.34524	1.34889	0.3%
128	0.957	0.01815	1.34836	1.35023	0.1%
192	0.958	0.02208	1.35891	1.35068	0.6%

In every scenario, 50 Mb/s of the UDP traffic was added, the buffer size of the link under study was $N = 100$ packets

6.2 Results

We measured the burst ratio in 20 traffic and buffer scenarios, divided into three groups.

In the first group, the buffer size for the output link under test (port 8) was set to 100 packets. On each Spirent port 1–4, 12.5 Mb/s of the UDP traffic was generated, resulting in 50 Mb/s of the total UDP traffic (5% of the tested link capacity). Moreover, on each Spirent port 1–4, several TCP connections were established. In particular, 7 distinct scenarios were tested, with 6, 8, 12, 16, 24, 32 and 48 TCP connections per port, thus resulting in the total number of long-lived TCP flows of 24, 32, 48, 64, 96, 128 and 192, respectively. Every TCP source on Spirent was assumed to have unlimited data to transmit and TCP used automatically the maximum available frame size, of 1518 bytes, for the great majority of sent packets. In order to introduce some variety to packet sizes, all UDP datagrams were assumed to be much smaller, of 552 bytes. In every scenario, two million packets were generated.

Note that in all scenarios, in which at least one TCP connection is used, it is impossible to set manually the load of the tested link, ρ , to an arbitrary value. The load is adjusted automatically by the New Reno algorithm, running within every TCP source.

The results for the first group of scenarios, with different number of TCP flows, are presented in Table 4. In particular, the average recorded load of the link is given in the second column, while the packet loss ratio in the third column. In the fourth column, the burst ratio computed in the experiment is shown, while in the fifth column - the burst ratio computed using Theorem 1, with the same parameters (load, packet size distribution and buffer size).

As we can see in Table 4, the experimental burst ratio was always greater than one, what is consistent with the results obtained in the previous section. Moreover, a very good agreement with theoretical results can be observed—the relative error is in range 0.1–3.4%. At first glance, this might be surprising, given the fact that the homogeneous

Poisson process constitutes rather coarse approximation of the IP traffic with several TCP flows, with the feedback congestion control in each of them. But, recalling the discussed fact, that the burst ratio depends much more on the service time distribution, than on the load, the obtained results make perfect sense.

In the second group of scenarios, we changed the distribution of the packet size, by increasing the fraction of small UDP datagrams, from 0 to 20%. Naturally, all the remaining parameters were unaltered—in each scenario, 128 TCP flows were used and the buffer size was always 100. The results are given in Table 5.

Increasing the fraction of small datagrams, we obviously increased the variance of the service time distribution. As we can see in Table 5, this also enlarged the burst ratio. This observation is consistent with observations made in the previous section. The growth of the burst ratio with the variance of the packet size can be observed for both experimental and theoretical B , though in the case of the theoretical value, the growth is a little slower. The relative error between the measured and computed value of B was in range 0.8–8.7%, which again is a good results, given the simplicity of the arrival process model.

Finally, in the third group of scenarios, we changed the buffer size from 50 to 1000, maintaining 128 TCP flows and 50 Mb/s of the UDP traffic in each case. The results are given in Table 6. As we can see, a high agreement between the experimental and theoretical value of B was obtained in every case.

7 Conclusions

We presented an analysis of the burst ratio in the queueing system with finite buffer. To the best of our knowledge, there are no published results on the burst ratio in a queueing model.

First of all, we derived the formula for the burst ratio in the $M/G/1/N$ queueing model, as well as its limit as the buffer

Table 5 Experimental versus theoretical burst ratios in traffic scenarios with different UDP traffic fractions

Fraction of the UDP traffic	Link load, ρ	Loss ratio, L	Experimental burst ratio, B_e	Theoretical burst ratio, B_t	Error, $\frac{ B_t - B_e }{B_e}$
0%	0.958	0.01741	1.23684	1.34023	8.4%
2%	0.957	0.01766	1.29645	1.34390	3.7%
4%	0.957	0.01771	1.33174	1.34810	1.2%
6%	0.957	0.01797	1.36354	1.35239	0.8%
8%	0.957	0.01813	1.39533	1.35677	2.8%
10%	0.956	0.01826	1.43087	1.36079	4.9%
15%	0.956	0.01877	1.46276	1.37238	6.2%
20%	0.956	0.01896	1.51731	1.38458	8.7%

In every scenario, 128 TCP flows were used, the buffer size of the link under study was $N = 100$ packets

Table 6 Experimental versus theoretical burst ratios in scenarios with different buffer sizes

Buffer size (packets)	Link load, ρ	Loss ratio, L	Experimental burst ratio, B_e	Theoretical burst ratio, B_t	Error, $\frac{ B_t - B_e }{B_e}$
50	0.957	0.01781	1.35314	1.34938	0.3%
100	0.957	0.01815	1.34836	1.35023	0.1%
200	0.957	0.01808	1.35205	1.35024	0.1%
500	0.954	0.01783	1.33970	1.34890	0.7%
1000	0.957	0.01767	1.35184	1.35024	0.1%

In every scenario, 128 TCP flows were used and 50 Mb/s of the UDP traffic

size grows to infinity. Then we showed several numerical calculations of the burst ratio, for different system parameterizations. We also compared the results obtained from the analytical formulas, with simulation results, using several different scenarios. Finally, we carried out measurements of the burst ratio on the real network, using realistic traffic structure (protocols, number of flows, TCP/UDP traffic proportions, buffer sizes etc.).

All the three approaches (analysis, simulations and lab measurements) enabled us to draw consistent conclusions about the dependence of the burst ratio on the system parameters. As we could see, the burst ratio strongly depends on the variance of the service time. Using a simple service time distribution, but with a high variance, one can obtain the burst ratio of a very high value. On the other hand, the value of the burst ratio depends very little on the system load, at least for its practically useful values, say from 0.5 to 2. Similarly, the value of the burst ratio does not depend very strongly on the buffer size—for relatively small buffers, it reaches the limit, and does not change anymore, as the buffer size grows.

Given the very simple structure of the Poisson model, compared to the real IP traffic, we demonstrated a surprisingly high agreement between the measured and theoretical values of the burst ratio—the relative error was always below 9% in all 20 experimental scenarios, while in 11 of them it was below 1%. This high agreement can be explained by the fact that the burst ratio does not depend much on the system load. It depends much more on the service time distribution,

which can be taken very precisely into account, when using Theorem 1.

Our study confirms also the fact that is well-known by computer networking practitioners. Namely, packet losses in computer networks have a tendency to group together, compared to what could be expected in the case of pure random loss (Bernoulli process)—we did not find any realistic example with the burst ratio smaller than 1.

Although our study was motivated by packet networks, all definitions, theorems and proofs were formulated in the language of the queueing theory. Therefore, they have universal sense and can be applied directly in other queueing systems, not only those present in computer networks.

There are several interesting possibilities of the future work. Firstly, we are going to derive the burst ratio in systems with more complex arrival processes. Good candidates are commonly used Markovian processes with an autocorrelated structure, e.g., the Markov modulated Poisson process (MMPP), or the batch Markovian arrival process (BMAP). For popular performance characteristics, like the queue size distribution, the transition from the Poisson process to general Markovian processes is well known. Similarly, it should be possible for the burst ratio. In the case of the BMAP arrivals, various admission strategies can be studied.

It is also interesting to investigate further the maximum in Fig. 2, i.e., to confirm analytically its existence, or give a counterexample. The complicated form of Theorem 1 makes finding this maximum hard, but not hopeless.

Finally, the solution of system (12), based on a different type of recursion, (e.g., derived using the censored Markov chains), can be searched for.

Acknowledgements This work was conducted within Project 2017/25/B/ST6/00110, founded by National Science Centre, Poland. The infrastructure was supported by PL-LAB2020 project, founded by the National Centre for Research and Development, Poland, contract POIG.02.03.01-00-104/13-00.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- McGowan, J. W. (2005). Burst ratio: A measure of bursty loss on packet-based networks, 16 2005. US Patent 6,931,017
- ITU-T Recommendation G.107: The E-model, a computational model for use in transmission planning. Technical report, (2014).
- ITU-T Recommendation G.113: Transmission impairments due to speech processing. Technical report (2007).
- Yajnik, M., Moon, S., Kurose, J., & Towsley, D. (1999). Measurement and modelling of the temporal dependence in packet loss. In *Proc. of IEEE INFOCOM* (Vol. 1, pp. 345–352).
- Gilbert, E. N. (1960). Capacity of a burst-noise channel. *Bell System Technical Journal*, 39(5), 1253–1265.
- Veeraraghavan, M., Cocker, N., & Moors, T. (2001). Support of voice services in IEEE 802.11 wireless LANs. In *Proceedings of IEEE INFOCOM* (Vol. 1, pp. 488–497).
- Jiang, W., & Schulzrinne, H. (2000). Modeling of packet loss and delay and their effect on real-time multimedia service quality. In *Proceedings of NOSSDAV* (pp. 1–10).
- Elliott, E. O. (1963). Estimates of error rates for codes on burst-noise channels. *Bell System Technical Journal*, 42(5), 1977–1997.
- Halinger, G., & Hohlfeld, O. (2008). The Gilbert-Elliott model for packet loss in real time services on the Internet. In *Proceedings of measuring, modelling and evaluation of computer and communication systems conference* (pp. 1–15).
- Clark, A. (2001). *Modeling the effects of burst packet loss and recency on subjective voice quality*. In *Proceedings of internet telephony workshop* (pp. 123–127).
- Sanneck, H. A., & Carle, G. (2000). Framework model for packet loss metrics based on loss runlengths. In *Proceedings of SPIE 3969, Multimedia Computing and Networking 2000* (pp. 1–11).
- Rachwalski, J., & Papir, Z. (2014). Burst ratio in concatenated Markov-based channels. *Journal of Telecommunications and Information Technology*, 1, 3–9.
- Rachwalski, J., & Papir, Z. (2015). Analysis of burst ratio in concatenated channels. *Journal of Telecommunications and Information Technology*, 4, 65–73.
- Yu, X., Modestino, J. W., & Tian, X. (2005). The accuracy of Gilbert models in predicting packet-loss statistics for a single-multiplexer network model. In *Proceedings of IEEE INFOCOM'05* (pp. 2602–2612).
- Benko, P., & Veres, A. (2002). A passive method for estimating end-to-end TCP packet loss. In *Proceedings of IEEE GLOBECOM'02* (pp. 2609–2613).
- Bolot, J. (1993). End-to-end packet delay and loss behavior in the internet. In *Proceedings of ACM SIGCOMM'93* (pp. 289–298).
- Coates, M., & Nowak, R. (2000). Network loss inference using unicast end-to-end measurement. In *Proceedings of ITC conference on IP traffic, measurement and modeling* (pp. 282–289).
- Duffield, N., Presti, F. L., Paxson, V., & Towsley, D. (2001). Inferring link loss using striped unicast probes. In *Proceedings of IEEE INFOCOM'01* (pp. 915–923).
- Sommers, J., Barford, P., Duffield, N., & Ron, A. (2005). Improving accuracy in end-to-end packet loss measurement. *ACM SIGCOMM Computer Communication Review*, 35(4), 157–168.
- Takagi, H. (1993). *Queueing analysis: Finite Systems*. Amsterdam: North-Holland.
- Chydzinski, A., Wojcicki, R., & Hryn, G. (2007). On the number of losses in an MMPP queue. In *Lecture Notes in Computer Science*, (Vol. 4712, pp. 38–48).
- Chydzinski, A., & Adamczyk, B. (2012). Transient and stationary losses in a finite-buffer queue with batch arrivals. *Mathematical Problems in Engineering*, vol. 2012, ID 326830, pp. 1–17
- Cidon, I., Khamisy, A., & Sidi, M. (1993). Analysis of packet loss processes in high-speed networks. *IEEE Transactions on Information Theory*, 39(1), 98–108.
- Chydzinski, A. (2007). Time to reach buffer capacity in a BMAP queue. *Stochastic Models*, 23, 195–209.
- Chydzinski, A. (2006). Transient analysis of the MMPP/G/1/K queue. *Telecommunication Systems*, 32(4), 247–262.
- Chydzinski, A. (2006). Queue size in a BMAP queue with finite buffer. In *Lecture notes in computer science* (Vol. 4003, pp. 200–210).
- Chydzinski, A., & Mrozowski, P. (2016). Queues with dropping functions and general arrival processes. *PLoS ONE*, 11(3), 1–23.
- Bratiychuk, M., & Borowska, B. (2002). Explicit formulae and convergence rate for the system $M^a/G/1/N$ as $N \rightarrow \infty$. *Stochastic Models*, 18(1), 71–84.
- <https://omnetpp.org>.
- Spirent URL: <https://www.spirent.com/>.
- MX2-10G-s12 load module. https://www.spirent.com/-/media/Datasheets/Broadband/PAB/SpirentTestCenter/Spirent_mX2_40-10_GbE_Dual-Speed_Test_Module_datasheet.pdf?la=en.
- CISCO 3750X Datasheet. https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-3560-x-series-switches/data_sheet_c78-584733.html.
- DPDK-Dump application for capturing traffic using DPDK. <https://github.com/marty90/DPDK-Dump>.