Annals of Telecommunications manuscript No.
(will be inserted by the editor)

Jitter Analysis of an IPP Tagged Traffic Stream in an {IPP,M}/M/1 Queue

Geza Geleji · Harry Perros

Received: date / Accepted: date

Abstract We study a single server queue with two different arriving streams, a tagged arrival process and a background arrival process. The tagged traffic is assumed to be an Interrupted Poisson Process (IPP) and the background traffic is Poisson. The service time is exponentially distributed and customers are served in a FIFO manner. We obtain numerically the PDF of the inter–departure time of the IPP tagged arrival process, from which we calculate its jitter, defined as a percentile of the inter–departure time. Numerical results of the 95th percentile and the squared coefficient of variation of the tagged inter-departure time are given as a function of the arrival rate of the background traffic.

Keywords queueing · multi–class · jitter · percentile

1 Introduction

The Next Generation Network (NGN) is an evolution of the vertically separate integrated networks, whereby a single IP-based network will carry all services. The IP network will run over different transport technologies, such as wireless networks, optical networks, Ethernet, and SONET/SDH, which should be able to provide quality–of–service (QoS) assurances. QoS is expressed in terms of the one way end–to–end delay, jitter, and packet loss.

Jitter is a well understood concept, but there is no agreed upon statistic for measuring it. In view of this, various metrics have been proposed in the

G. Geleji · H. Perros
North Carolina State University
Department of Computer Science
Campus Box 8206
Raleigh, NC 27695–8206
Tel.: ++1-919-515-2040
E-mail: ggeleji@ncsu.edu, hp@csc.ncsu.edu

literature. One commonly used metric is based on the one way end–to–end delay of the individual packets. Specifically, if we consider the differences of the end–to–end delay of successive packets, then jitter can be defined using a statistic, such as, the running average of differences over a fixed number of packets, and a percentile value x such that 95% of the time these differences are less than x. Jitter has also been expressed as the difference between the end–to–end delay of a packet and the end–to–end propagation delay, and it is known as the delay variation. An alternative metric for jitter is based on the inter–arrival times of the successive packets at the destination, and it is typically defined as a percentile, such as the 95th percentile, of the inter–arrival time. This latter definition is the one used in this paper. Note that the 95th percentile is recommended by ITU–T as a simple, and fairly accurate method for calculating Inter–Packet Delay Variation in real time [12].

In our work, we study how the jitter of a traffic stream is affected by the presence of another traffic stream. For this, we consider a single queue with two different arriving processes, a tagged and a background stream, and we derive the probability density function (PDF) of the inter–departure time of the tagged traffic.

The problem of characterizing the departure process of a single class of customers in a multi–class queue arises in the analysis of non product–form queueing networks and more recently in the characterization of the jitter of a traffic flow. The exact Laplace transform of the class-dependent inter-departure time distribution in a multi-class queue, where each arrival process is Poisson and the service time has a class-dependent general distribution was obtained by Stanford and Fischer [8]. Dasu [3] considered a two–class single server queueing system where the tagged arrival process is a generalized phase process [2], the background arrival process is Poisson, and the service time follows a phase–type distribution. For this model, he obtained a closed–form expression of the Laplace transform of the inter–departure time of the tagged traffic. This is a very complex expression even in the case where the tagged inter–arrival time follows an Erlang distribution and the service time is exponentially distributed. For this case, he obtained the second moment by numerical differentiation of the Laplace transform. To the best of our knowledge, the above two references are the only ones where exact results have been reported. Several approximations have also been reported under a variety of assumptions. Whitt [11] developed two moment approximations of the departure process of a single class of customers in a multi–class GI/G/m queue. In Kumaran et al. [5], the tagged and the background arrival processes were assumed to be matrix exponential (ME), and the service time distribution was also an ME. The authors obtained an approximation for the tagged departure process. In Mitchell et al. [6], an approximation of the tagged departure process was also obtained for heavy and light traffic under similar assumptions as the previous paper. The above references are for continuous–time models. In addition, the problem of determining the jitter has been also considered in the discrete–time domain for ATM networks, see for instance Sohraby and Privalov [7].

In this paper, we consider a single server queue with two different arriving streams, a tagged arrival process and a background arrival process. The tagged traffic is assumed to be an Interrupted Poisson Process (IPP) and the background traffic is Poisson. The service time is exponentially distributed and customers are served in a FIFO manner. The assumptions are less general than those considered in Dasu [3], but for this model, we obtain the exact[1] PDF of the inter–departure time of successive tagged customers from which we can easily compute its 95th percentile, or any other percentile. This is done by analyzing numerically a series of homogeneous, absorbing Markov processes.

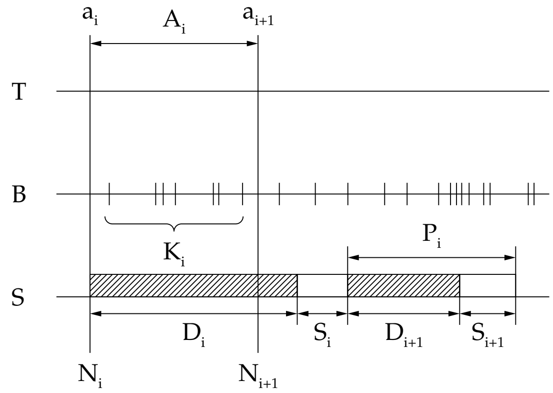
The paper is organized as follows. In Section 2, we derive the PDF of the tagged inter–departure time of the {IPP,M}/M/1 queue. In Section 3 we present a set of numerical results obtained using our model, and in Section 4, we conclude our paper.

2 The PDF of the Tagged Inter–Departure Time

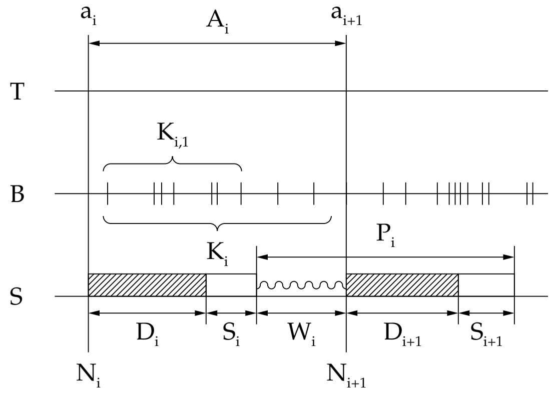
Let us consider a single server FIFO queue with two different arriving traffic streams, an IPP tagged arrival process and a Poisson background arrival process. The tagged and background streams are interspersed and the queue serves the customers in their order of arrival, without giving priority to either stream, in an exponentially distributed amount of time that is independent of the stream to which the customer belongs. The successive service times are independent and identically distributed.

The parameters of the tagged IPP are as follows: the arrival rate in the ON state is λ, and in the OFF state is zero. The transition rate from the ON state to the OFF state is σ_1, and from the OFF state to the ON state is σ_2. We assign the index 1 to the ON state, and 2 to the OFF state. The rate of the background Poisson process is μ, and the rate of service is θ. In Figure 1, we show the two possible sequences of the significant queueing events involving two successive tagged arrivals into the queue. Let a_i and a_{i+1} denote, respectively, the instant of arrival of the first and the second tagged customer during the observed period. Let N_i and N_{i+1} denote the respective queue lengths at these two instants. Let A_i be the inter–arrival time, K_i the number of background arrivals occurring between these two tagged arrivals, and $K_{i,1}$ the number of background arrivals occurring between the arrival of the first tagged customer and its respective departure. Let W_i be the time that passes between the departure of the first tagged customer and the arrival of the second tagged customer (only relevant if positive). Let D_i be the amount of time it takes to serve all the customers that were already in the queue at the moment of arrival of the first tagged customer at a_i, and S_i the service time of the first tagged customer. The interpretation of D_{i+1} depends on the context. In case 1 (Figure 1a), D_{i+1} represents the amount of time it takes to serve all the background customers that have arrived between the two tagged

[1] i.e., arbitrary precision



(a) Case 1: The second observed tagged arrival (a_{i+1}) occurs *before* the customer that arrived at a_i departs. That is, $A_i < D_i + S_i$.



(b) Case 2: The second observed tagged arrival (a_{i+1}) occurs *after* the customer that arrived at a_i has departed. That is, $A_i > D_i + S_i$.

Fig. 1: Order of events with respect to successive arrivals of two tagged customers. Queueing events are shown on three simultaneous timescales marked by capital letters on the left. "T" marks the timescale belonging to the tagged arrival process, "B" the timescale of the background arrival process, and "S" the timescale of the service process.

arrivals. Quite clearly, this is the service time of K_i background customers. In case 2 (Figure 1b), D_{i+1} is the time it takes to serve all the background customers that are in the queue at the moment just before the second tagged arrival at a_{i+1}. Note that at this instant, there can be no tagged customers in the queue, since the last one has already been served at time $a_i + D_i + S_i$ and the next one is just about to arrive. Let S_{i+1} be the service time of the second tagged arrival at time a_{i+1}. Finally, let P_i be the time that passes between the departure of the two tagged customers arriving at a_i and a_{i+1}. This is the inter–departure time, whose PDF we compute in this paper.

The computation of P_i is broken down into a series of absorbing Markovian processes that are solved sequentially. Our approach to obtaining the PDF of an arbitrary tagged inter–departure time is to take an arbitrary tagged arrival, a_i, and observe how long it takes for it to receive service and depart $(D_i + S_i)$. Then we observe a second tagged arrival, a_{i+1}, following the first one, compute the time it takes for it to get served and then derive the PDF of the time between the departure of these two tagged customers. For this, we first need to obtain the queue length distribution (denoted by N_i) as seen by an arbitrary tagged arrival; this is done in Section 2.1. We use this queue length distribution in Section 2.2 to compute the probability that case 2 occurs, that is, the second tagged customer arrives after the departure of the first one (Figure 1b).

Having obtained these two quantities, we proceed to calculate the inter–departure time $P_i = W_i + D_{i+1} + S_{i+1}$ for case 2, which follows a phase–type distribution, in Section 2.4. The cumulative distribution function (CDF) of this phase–type distribution is obtained by analyzing the absorption time of an absorbing Markov process with a single absorbing state. The initial probability distribution of this Markov process is determined by the joint probability distribution of the queue length and the state of the tagged arrival IPP (i.e., ON or OFF) at the instant of departure, $a_i + D_i + S_i$, of the first tagged customer. This is obtained in Section 2.3.

In case 1, the second tagged customer arrives before the first one departs. In this case, the inter–departure time $P_i = D_{i+1} + S_{i+1}$, where S_{i+1} is the service time of the second tagged customer, and D_{i+1} is the service time of K_i background customers that arrived between the tagged arrivals (Figure 1a). The distribution of K_i is obtained in Section 2.5 by modeling the evolution of the system between the two tagged arrivals.

2.1 The Queue Length Distribution at a Tagged Arrival

In our analysis, we need to know the queue length distribution in the queueing system under study, at the moment of a tagged arrival, i.e. an arrival from the IPP stream. The tagged IPP merged with a Poisson process forms a two–state Markov–Modulated Poisson process (MMPP–2). Let λ_1 and λ_2 be the rate of arrivals in state 1 and 2, respectively, of the MMPP–2, and let σ_1 and σ_2 be the rate of transition from state 1 to state 2 and from state 2 to state 1, respectively. Then the steady–state probability vector of such a process may be expressed as:

$$\Pi = (\Pi_1, \Pi_2) = \left(\frac{\sigma_2}{\sigma_1 + \sigma_2}, \frac{\sigma_1}{\sigma_1 + \sigma_2} \right) \tag{1}$$

The parameters of the MMPP–2 are set as follows: $\lambda_1 = \lambda + \mu$; $\lambda_2 = \mu$. That is, state 1 of the MMPP–2 corresponds to the ON state of the tagged

IPP, and state 2 of the MMPP–2 corresponds to the OFF state. Naturally, the background Poisson process is active in both states; hence the μ term. The queue length distribution at the instant of a tagged arrival was obtained by analyzing the queueing system as an MMPP–2/M/1 queue, using Neuts' matrix geometric method [9, Section 10.6].

The infinitesimal generator of the system is

$$Q = \begin{pmatrix} \Sigma - \Lambda & \Lambda & 0 & \cdots \\ \Theta & \Sigma - \Lambda - \Theta & \Lambda & \cdots \\ 0 & \Theta & \Sigma - \Lambda - \Theta & \cdots \\ 0 & 0 & \Theta & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{2}$$

$$\Sigma = \begin{pmatrix} -\sigma_1 & \sigma_1 \\ \sigma_2 & -\sigma_2 \end{pmatrix} \Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \Theta = \begin{pmatrix} \theta & 0 \\ 0 & \theta \end{pmatrix}. \tag{3}$$

Let $\pi = (\pi_0, \pi_1, \pi_2, \ldots)^T$ be the stationary distribution matrix returned by Neuts' method, where $\pi_i = (\pi_{i,1}, \pi_{i,2})^T$, and $\pi_{i,j}$ is the time average probability that the arrival process is in state j and the system contains i customers. Now, the queue length as seen by a tagged arrival may be expressed as follows:

$$\pi_a = \frac{1}{\Pi_1} \pi \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{4}$$

2.2 The Probability that Case 2 Occurs ($A_i > D_i + S_i$)

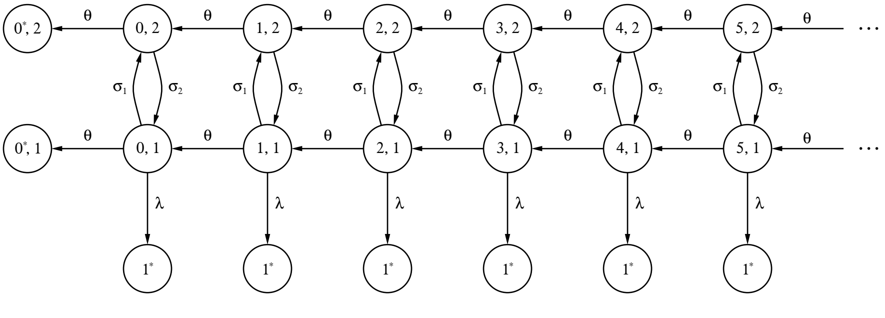


Fig. 2: The probability of a case 2 tagged departure.

As shown in Figure 1, we distinguish between cases 1 and 2 depending upon whether the second tagged customer arrives before or after the first tagged customer has departed. This distinction is necessary because the inter–departure time between two successive tagged departures is structured differently. As will be seen later on, it is essential to know with what probability do these two cases occur. The probability that case 2 occurs is computed by analyzing the absorption probabilities of the continuous–time absorbing Markov process shown in Figure 2.

This process depicts the evolution of the system from the moment that a tagged customer A arrives to the moment that it either departs, or a second tagged customer B arrives. States (i, j) depict the number of customers i in the queue which are ahead of customer A, and the state $j \in \{1, 2\}$ of the IPP. Any background customers that arrive after A are not of interest, since we are only concerned with the service of the customers that were already present in the queue at the moment of arrival of A. The initial distribution of this process is the queue length distribution of the system as seen by customer A upon its arrival, calculated in the previous section. The IPP has to be in the ON state ($j = 1$) in order for a tagged customer to arrive. Consequently, $p(i, 2) = 0$ for $i \geq 0$. The remaining probabilities $p(i, 1)$, $i \geq 0$, of the initial distribution are the queue–length probabilities seen by an arbitrary tagged customer upon arrival.

The process starts therefore at a state $(i, 1)$ with probability $p(i, 1)$ and evolves until absorption. There are three absorbing states: (1^*), $(0^*, 1)$, and $(0^*, 2)$. Let us assume for example that the process is in state $(1, 1)$. This means that there is one customer (tagged or background) ahead of customer A, which is in service, and the IPP is in the ON state. There are two possible transitions: *a)* the customer in service departs, which shifts the state to $(0, 1)$, and *b)* tagged customer B arrives, which shifts the process to the absorbing state (1^*). Absorbing states $(0^*, 1)$ and $(0^*, 2)$ indicate that the tagged customer A departed prior to B's arrival and the state of the IPP at the instant of departure was 1 (ON) or 2 (OFF), respectively. Absorbing state (1^*) indicates that the tagged customer A departed after B's arrival.

Based on this absorbing Markov process, the probability that case 2 occurs is equal to the sum of the probabilities that the process is absorbed in states $(0^*, 1)$ and $(0^*, 2)$. Also, from the individual absorbing states we have the state of the IPP when A departs prior to B's arrival. The probability that case 1 occurs is the complementary of the probability that case 2 occurs and is also equal to the probability that the process will be absorbed in state (1^*).

If we arrange the states of this absorbing Markov process so that the transient states, ordered as follows: $(0, 1)$, $(0, 2)$, $(1, 1)$, $(1, 2)$, ..., are followed by the absorbing states $(0^*, 1)$, $(0^*, 2)$, and (1^*), then the infinitesimal generator of the process is (I is the identity matrix, Σ and Θ are as defined in Eq. 3)

$$P = \begin{pmatrix} Q & R \\ 0 & I \end{pmatrix}, \text{ with} \tag{5}$$

$$Q = \begin{pmatrix} \Sigma - \star & 0 & 0 & \cdots \\ \Theta & \Sigma - \star & 0 & \cdots \\ 0 & \Theta & \Sigma - \star & \cdots \\ 0 & 0 & \Theta & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad R = \begin{pmatrix} \Theta & \Lambda \\ 0 & \Lambda \\ 0 & \Lambda \\ 0 & \Lambda \\ \vdots & \vdots \end{pmatrix}, \tag{6}$$

$$\varLambda = \begin{pmatrix} \lambda \\ 0 \end{pmatrix}, \quad \hat{\varLambda} = \begin{pmatrix} \lambda & 0 \\ 0 & 0 \end{pmatrix}, \quad \text{and} \quad \star = \hat{\varLambda} + \varTheta. \tag{7}$$

In order to compute the absorption probabilities [9, Section 9.6.2] [4, Section 11.2], the number of transient states in the absorbing Markov process has to be finite. For this reason, it is necessary to truncate the Q and R matrices to a finite number of states. The absorption probability matrix $B = [b_{ij}]$, where b_{ij} is the probability of absorption in absorbing state j on condition that the process was started in transient state i, is computed as follows:

$$B = (I - \dot{Q})^{-1}\dot{R}, \tag{8}$$

where $\dot{Q}$ and $\dot{R}$ are discretized versions [9, Section 10.1.1] of Q and R, respectively. It is important to note that the matrix $(I - \dot{Q})$ has a block tridiagonal structure, so B may be efficiently computed using the Thomas algorithm [1].

2.3 Case 2: The Joint Probability Distribution of the Queue Length and the State of the IPP at the Instant of Departure of the First Tagged Customer

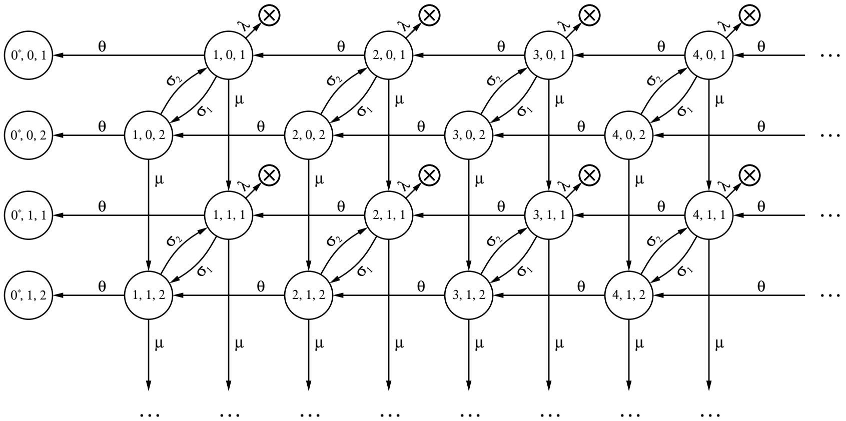


Fig. 3: Case 2: The absorbing Markov process for computing the joint distribution of the queue length and the state of the arrival process at the departure time of the first tagged customer.

In order to compute the queue length distribution N_{i+1} at a_{i+1} in case 2, we need to know the joint distribution of the queue length and the state of the IPP at the departure of the first tagged customer at $a_i + D_i + S_i$. This is achieved using the absorbing Markov process presented in Figure 3. The model has a three–dimensional state space (i, j, k), where i represents the number of customers in the queue immediately before a_i plus the tagged customer arriving at a_i, j represents the number of background customers that have arrived since then, and k is the state of the IPP. Starting from an initial state $(i, 0, k)$, $i \geq 1$, $k \in \{1, 2\}$, the process keeps track of the evolution of the system from the instant a_i until the first tagged departure, which is assumed to conform to case 2.

The initial state distribution $(i, 0, k)$, $i \geq 1$, $k \in \{1, 2\}$ is determined by the queue length distribution as seen by customer A at time a_i, conditioned on the fact that its departure conforms to the case 2 assumption. The probabilities may be obtained by combining the results from the first two Markovian models (Sections 2.1 and 2.2). The probability of starting in the remaining states (i, j, k), $i \geq 0$, $j > 0$, $k \in \{1, 2\}$ is zero. The process evolves until tagged customer A departs, or tagged customer B arrives before A's departure. Given we are in, say, state $(2, 1, 1)$, the process can shift to $(2, 1, 2)$ if the IPP state changes from ON to OFF, or to $(1, 1, 1)$ if a departure occurs, or to $(2, 2, 1)$ if a background customer arrives. It can also be absorbed in the state represented by smaller circles marked with "X" (these absorbing states are of no interest to this analysis and for presentation purposes they are all marked the same), if tagged customer B arrives before A's departure. Another set of absorbing states $(0^*, j, k)$, $j \geq 0$, $k \in \{1, 2\}$, represent the case where A departs prior to B's arrival. Absorption in these states can only occur from states $(1, j, k)$, $j \geq 0$, $k \in \{1, 2\}$.

The states are enumerated in the following way: $(1, 0, 1)$, $(1, 0, 2)$, $(2, 0, 1)$, $(2, 0, 2)$, $(3, 0, 1)$, $(3, 0, 2)$, …, $(1, 1, 1)$, $(1, 1, 2)$, $(2, 1, 1)$, $(2, 1, 2)$, $(3, 1, 1)$, $(3, 1, 2)$, …, $(1, 2, 1)$, $(1, 2, 2)$, $(2, 2, 1)$, $(2, 2, 2)$, $(3, 2, 1)$, $(3, 2, 2)$, …, (X), $(0^*, 0, 1)$, $(0^*, 0, 2)$, $(0^*, 1, 1)$, $(0^*, 1, 2)$, … (note that the transient states precede the absorbing states). The Q and R blocks (see Equation 5) of the infinitesimal generator matrix of this continuous–time absorbing Markov process are as follows:

$$Q = \begin{pmatrix} \tilde{\Sigma} & \tilde{M} & 0 & 0 & \cdots \\ 0 & \tilde{\Sigma} & \tilde{M} & 0 & \cdots \\ 0 & 0 & \tilde{\Sigma} & \tilde{M} & \cdots \\ 0 & 0 & 0 & \tilde{\Sigma} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{9}$$

$$R = \begin{pmatrix} \tilde{\Lambda} & \tilde{\Theta} & 0 & 0 & 0 & \cdots \\ \tilde{\Lambda} & 0 & \tilde{\Theta} & 0 & 0 & \cdots \\ \tilde{\Lambda} & 0 & 0 & \tilde{\Theta} & 0 & \cdots \\ \tilde{\Lambda} & 0 & 0 & 0 & \tilde{\Theta} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{10}$$

$$\text{with} \quad \tilde{\Lambda} = \begin{pmatrix} \Lambda \\ \Lambda \\ \Lambda \\ \Lambda \\ \vdots \end{pmatrix}, \quad \tilde{\Theta} = \begin{pmatrix} \Theta \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix}, \tag{11}$$

$$\tilde{\Sigma} = \begin{pmatrix} \Sigma - \star & 0 & 0 & 0 & \cdots \\ \Theta & \Sigma - \star & 0 & 0 & \cdots \\ 0 & \Theta & \Sigma - \star & 0 & \cdots \\ 0 & 0 & \Theta & \Sigma - \star & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{12}$$

$$\tilde{M} = \begin{pmatrix} M & 0 & 0 & 0 & \cdots \\ 0 & M & 0 & 0 & \cdots \\ 0 & 0 & M & 0 & \cdots \\ 0 & 0 & 0 & M & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{13}$$

$$M = \begin{pmatrix} \mu & 0 \\ 0 & \mu \end{pmatrix}, \quad \star = M + \hat{\Lambda} + \Theta. \tag{14}$$

Σ and Θ are as defined in Eq. 3; Λ and $\hat{\Lambda}$ are as defined in Eq. 7. In order to make Q and R finite, the matrices $\tilde{\Lambda}$, $\tilde{\Theta}$, $\tilde{\Sigma}$, and $\tilde{M}$ have to be truncated at a maximum value of i, i.e., the number of customers already in the queue upon A's arrival. Q and R should be truncated at the maximum value of j, i.e., the number of background customers that may arrive after A. If the maximum value of i is N, and the maximum value of j is K, then, as a result of these truncations, $\tilde{\Lambda}$ will have $2N$ rows and 1 column, $\tilde{\Theta}$ $2N$ rows and 2 columns, while both of $\tilde{\Sigma}$ and $\tilde{M}$ $2N$ rows and $2N$ columns. Q will have K block rows and K block columns, R K block rows and $2K + 1$ columns.

2.4 The Inter–Departure Time for Case 2

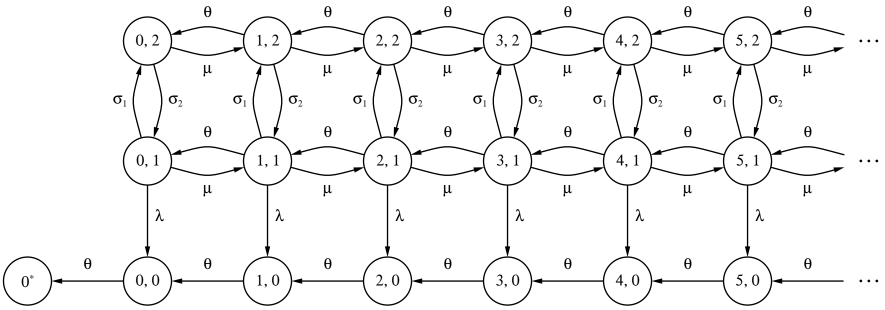


Fig. 4: Case 2: The absorbing Markov process from which the case 2 inter–departure time is obtained.

Given the state of the system at $a_i + D_i + S_i$ under the case 2 condition, the inter–departure time between the two tagged customers under observation follows a phase–type distribution which is the time to absorption in the absorbing Markov process shown in Figure 4. The process has a two–dimensional transient state space (i, j), $i \geq 0$, $j \in \{0, 1, 2\}$ and a single absorbing state, 0^*. The set of transient states consists of two structural subsets: (i, j), $i \geq 0$, $j \in \{1, 2\}$, and $(i, 0)$, $i \geq 0$. The former models the evolution of the queue from $a_i + D_i + S_i$ until a_{i+1}, i.e., the arrival of the second tagged customer. This event is denoted by a transition from the first subset to the second at rate λ. When the process is in a state belonging to the second subset, only services may occur, at a uniform rate of θ. The process will get absorbed after serving all the customers that were present in the queue at the instant of the transition from the first subset to the second. Obviously, the second subset represents the behavior of the queue from a_{i+1} until $a_{i+1} + D_{i+1} + S_{i+1}$.

The initial probability of the absorbing state 0^* and of the states belonging to the second structural subset, $(i, 0)$, $i \geq 0$, is zero. The initial probability of being in state (i, j), $i \geq 0$, $j \in \{1, 2\}$, is equal to the probability that at instant $a_i + D_i + S_i$, there were exactly i customers in the queue and the arrival IPP was in state j. We have already computed this probability distribution in Section 2.3.

The infinitesimal generator matrix of the Markov process shown in Figure 4 is

$$Q = \begin{pmatrix} S & R \\ 0 & 1 \end{pmatrix}, \text{ with} \tag{15}$$

$$S = \begin{pmatrix} \check{\Sigma}' & \check{M} & 0 & 0 & \cdots \\ \check{T} & \check{\Sigma} & \check{M} & 0 & \cdots \\ 0 & \check{T} & \check{\Sigma} & \check{M} & \cdots \\ 0 & 0 & \check{T} & \check{\Sigma} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad R = \begin{pmatrix} \bar{T} \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix}, \tag{16}$$

$$\check{\Sigma}' = \begin{pmatrix} -\sigma_1 - \lambda - \mu & \sigma_1 & \lambda \\ \sigma_2 & -\sigma_2 - \mu & 0 \\ 0 & 0 & -\theta \end{pmatrix}, \tag{17}$$

$$\check{\Sigma} = \begin{pmatrix} -\sigma_1 - \lambda - \mu - \theta & \sigma_1 & \lambda \\ \sigma_2 & -\sigma_2 - \mu - \theta & 0 \\ 0 & 0 & -\theta \end{pmatrix}, \tag{18}$$

$$\check{M} = \begin{pmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \check{T} = \begin{pmatrix} \theta & 0 & 0 \\ 0 & \theta & 0 \\ 0 & 0 & \theta \end{pmatrix}, \text{ and } \bar{T} = \begin{pmatrix} 0 \\ 0 \\ \theta \end{pmatrix}. \tag{19}$$

The CDF of the time to absorption, which is the CDF of P_i in case 2, is given by

$$F_{P_i^{(2)}}(t) = P(P_i^{(2)} < t) = 1 - \alpha e^{St} \underline{1}, \tag{20}$$

where α is the initial distribution on the transient states of the process as discussed above, e^x is the matrix exponential and $\underline{1}$ is a column vector of ones.

2.5 Case 1: The Distribution of the Number of Background Arrivals Between Two Successive Tagged Arrivals

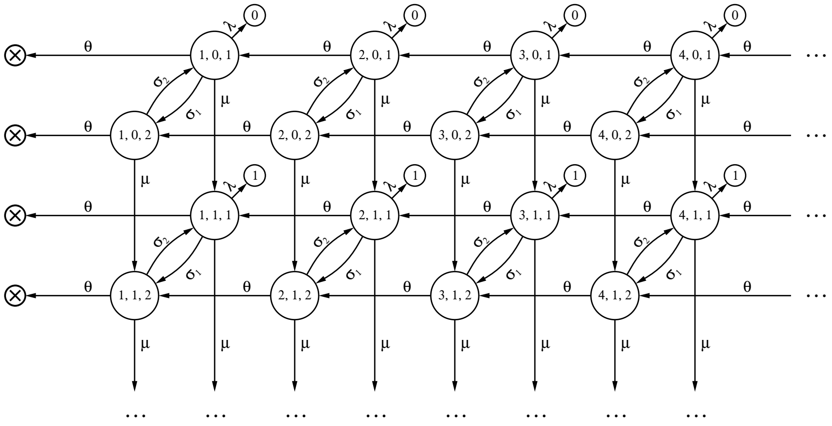


Fig. 5: Case 1: Calculation of the distribution of the number of background arrivals between the two tagged arrivals.

In case 1, the distribution of the first tagged inter–departure time, P_i, is determined by K_i, the number of background arrivals occurring between the first two tagged arrivals. The probability distribution of K_i may be computed using the absorbing Markov process shown in Figure 5. The state of the process is given as the triplet (i, j, k), where $i \geq 1$ is the queue length immediately after the first tagged arrival at a_i, $j \geq 0$ denotes the number of background arrivals since a_i, and $k \in \{1, 2\}$ is the state of the IPP arrival process. For instance, in state $(2, 1, 1)$, there are 3 customers in the queue, of which one (tagged or background) is in front of the tagged customer and another one (a background customer) is behind it. A background arrival will shift the process to state $(2, 2, 1)$, a change in the IPP state will shift it to $(2, 1, 2)$, and if a tagged customer arrives, the process will get absorbed in the state marked with (1). It is also possible for the tagged customer to depart before the second tagged customer arrives. For instance, if the process is in state $(1, 1, 2)$ and a departure occurs, it will shift to absorbing state (X_2) (the (X_1) and (X_2) absorbing states are not used in the analysis, and for presentation purposes they are all indicated as (X)).

The initial state distribution $(i, 0, k)$, $i \geq 1$, $k \in \{1, 2\}$ is the joint distribution observed immediately after time a_i, calculated in Section 2.1. The initial probability of the remaining states if zero. Of interest are only the absorbing states (j), $j \geq 0$ due to the arrival of the second customer, since for states (X_1) and (X_2), the assumption of $A_i < D_i + S_i$ is not fulfilled.

The matrices defining the Markov process are as follows, assuming that the states are enumerated in the order $(1,0,1)$, $(1,0,2)$, $(2,0,1)$, $(2,0,2)$, $(3,0,1)$, $(3,0,2)$, …, $(1,1,1)$, $(1,1,2)$, $(2,1,1)$, $(2,1,2)$, $(3,1,1)$, $(3,1,2)$, …, (X_1), (X_2), (0), (1), (2), …:

$$Q = \begin{pmatrix} \tilde{\Sigma} & \tilde{M} & 0 & 0 & \cdots \\ 0 & \tilde{\Sigma} & \tilde{M} & 0 & \cdots \\ 0 & 0 & \tilde{\Sigma} & \tilde{M} & \cdots \\ 0 & 0 & 0 & \tilde{\Sigma} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{21}$$

$$R = \begin{pmatrix} \tilde{\Theta} & \tilde{\Lambda} & 0 & 0 & 0 & \cdots \\ \tilde{\Theta} & 0 & \tilde{\Lambda} & 0 & 0 & \cdots \\ \tilde{\Theta} & 0 & 0 & \tilde{\Lambda} & 0 & \cdots \\ \tilde{\Theta} & 0 & 0 & 0 & \tilde{\Lambda} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{22}$$

$$\text{with } \tilde{\Lambda} = \begin{pmatrix} \Lambda \\ \Lambda \\ \Lambda \\ \Lambda \\ \vdots \end{pmatrix}, \quad \tilde{\Theta} = \begin{pmatrix} \Theta \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix}, \tag{23}$$

$$\tilde{\Sigma} = \begin{pmatrix} \Sigma - \star & 0 & 0 & 0 & \cdots \\ \Theta & \Sigma - \star & 0 & 0 & \cdots \\ 0 & \Theta & \Sigma - \star & 0 & \cdots \\ 0 & 0 & \Theta & \Sigma - \star & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{24}$$

$$\tilde{M} = \begin{pmatrix} M & 0 & 0 & 0 & \cdots \\ 0 & M & 0 & 0 & \cdots \\ 0 & 0 & M & 0 & \cdots \\ 0 & 0 & 0 & M & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \tag{25}$$

with Σ, Θ as defined in Eq. 3, M as defined in Eq. 14, Λ and $\hat{\Lambda}$ as defined in Eq. 7, and $\star = M + \hat{\Lambda} + \Theta$. Again, the infinite matrices need to be truncated.

Given the probability distribution of K_i, one may easily compute the distribution of $P_i^{(1)}$ as follows. With probability $P\{K_i = 0\}$, $P_i^{(1)}$ is the distribution of the service time of a single customer which is exponentially distributed with rate θ. With probability $P\{K_i = k\}$, $P_i^{(1)}$ is the service time of $k+1$ customers, and follows an Erlang$(k+1, \theta)$ distribution. From the CDFs of these Erlang distributions, the CDF of $P_i^{(1)}$ may be easily computed:

$$F_{P_i^{(1)}}(t) = \sum_{k=0}^{n} P\{K_i = k\} F_{E_{k+1,\theta}}(t) \tag{26}$$

Let us remind the reader that since $P\{K_i = k\}$ has only been computed for a finite set of k values due to the truncation, the summation in the above formula consists of a finite number of terms as well.

Finally, the distribution of P_i may be computed by taking the weighted average of the CDFs of the two P_i random variables obtained independently for cases 1 and 2:

$$F_{P_i}(t) = P\{\text{Case } 1\}F_{P_i^{(1)}}(t) + P\{\text{Case } 2\}F_{P_i^{(2)}}(t) \tag{27}$$

Due to the fact that $F_{P_i}(t)$ is monotonically increasing, numerically solving the equation $F_{P_i}(t) = 0.95$ to obtain the 95th percentile of the inter–departure time is fairly straightforward.

2.6 The heavy–traffic model

As the traffic load gets higher, the probability that case 2 occurs decreases, and the probability that case 1 occurs increases. In the limiting case, therefore, where the traffic intensity of the queue tends to 1, only case 1 may occur, and in this case instead of the three–dimensional state space of the absorbing Markov process in Section 2.5, a similar one with a two–dimensional state space will suffice (see Figure 6). This greatly simplifies the computation of the CDF of the inter–departure time.

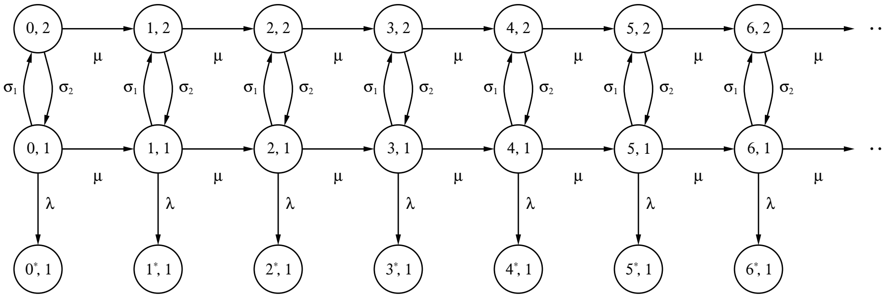


Fig. 6: Heavy traffic model: the distribution of the number of background arrivals between two tagged customers.

3 Numerical Results

In this section, we provide numerical results and also discuss the computational complexity of the proposed algorithm. The numerical results were obtained for the case where the tagged traffic accounts for a small percentage of the utilization in relation to the background traffic. This is realistic in cases where we are concerned with a single flow in the presence of multiple flows in an output port buffer of a router.

Figure 7 shows the 95th percentile of the inter–departure time distribution of the tagged process as a function of the arrival rate of the background traffic. The three curves shown therein correspond to three different values of the squared coefficient of variation, C^2, 5, 10, and 20. For each curve, we have constructed a tagged arrival IPP whose rate of arrival in the ON state is 0.2, the transition rates between the two states are symmetric, and the squared coefficient of variation matches the C^2 value associated with the curve. It is easy to see that the mean arrival rate of such a process is 0.1. The service rate of the queue is set equal to 1.0. Thus the traffic intensity due to the tagged arrival process is 0.1. The arrival rate of the background traffic was varied from 0.05 to 0.9, which means that the total traffic intensity due to both arrival streams was varied from 0.15 to 1.0. The case where the rate of the background traffic is 0.9 corresponds to full utilization. For this case, we obtained the results separately, using the heavy traffic model (Section 2.6). We observe that for C^2 equals 10 and 20, the 95th percentile of the inter–departure time shows an increasing tendency as the rate of the background traffic increases, whereas for $C^2 = 5$ it decreases.

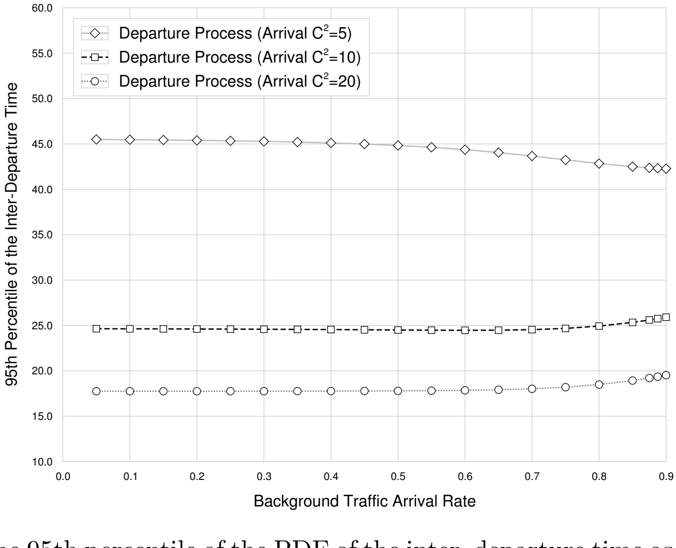


Fig. 7: The 95th percentile of the PDF of the inter–departure time as a function of the arrival rate of the background traffic.

Figure 8 shows the C^2 of the inter–departure time of the tagged process as a function of the arrival rate of the background traffic, for the same three tagged arrival processes used in Figure 7. As the rate of arrival of the background traffic increases, the C^2 of the inter–departure times of the tagged process drops. We have also conducted simulation experiments using an Erlang–4 distribution as the inter–arrival process of the tagged traffic (the C^2 of this process is less than 1) and observed that as the background traffic arrival rate increases, the C^2 of the inter–departure time of the tagged process increases (which is consistent with the results shown in Dasu [3]). This trend is in contrast with the IPP tagged arrival process, where as shown in Figure 8, the C^2 decreases. Finally, we note that if the tagged arrival traffic is Poisson (the C^2 of this process is 1), then the tagged departure process is also Poisson. Therefore, the C^2 of the inter–arrival time does not change as the tagged stream passes through the queue with Poisson background traffic. Just as in the case of Figure 7, the data points corresponding to a background arrival rate of 0.9 (full utilization) were obtained using the heavy traffic model.

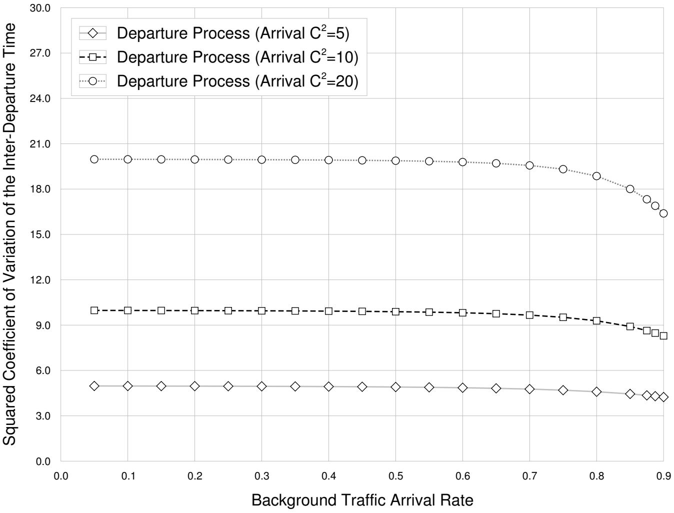


Fig. 8: The squared coefficient of variation (C^2) of the PDF of the inter–departure time as a function of the arrival rate of the background traffic.

Figure 9 shows the 95th percentile of the probability distribution of the inter–arrival time and the inter–departure time of the tagged arrival process as a function of the C^2 of its inter–arrival time. There are three curves on the graph. One gives the 95th percentile of the inter–arrival time of the tagged arrival IPP process (i.e., before it enters the queue), and the other two give the 95th percentile of the inter–departure time of the tagged arrival process with two different arrival rates of the background traffic: 0.6 and 0.8.

Note that increasing the rate of background traffic transforms the tagged stream towards a more Poisson–like traffic stream, as far as the 95th percentile of the inter–arrival time is concerned. When the arriving tagged stream is Poisson ($C^2 = 1$), the departing tagged stream is Poisson as well, irrespective

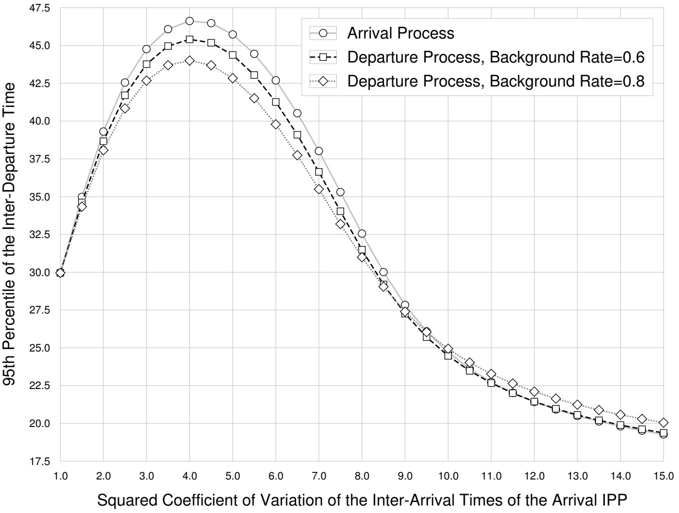


Fig. 9: The 95th percentile of the PDF of the inter–departure time as a function of the squared coefficient of variation (C^2) of the PDF of the inter–arrival time of the tagged IPP.

of the rate of background traffic. When the squared coefficient of variation of the tagged arrival process is in the $[2, 8]$ range, the 95th percentile of the process is greater than that of and equivalent Poisson process; consequently, passing the stream through a node with Poisson background traffic decreases the 95th percentile (i.e., draws it towards the Poisson process). When the C^2 value of the arrival process is greater than 9, the 95th percentile of the tagged inter–arrival time distribution is less than that of the Poisson process. In such cases, the background traffic has the effect of increasing the 95th percentile of the inter–arrival time distribution of the tagged process.

All of the results presented in this section have been verified by simulation.

3.1 Computational Complexity

The computational complexity of our method is dominated by the time required to compute the matrix exponential in Equation 20. Our implementation uses GNU Octave, which computes the matrix exponential by the means of a diagonal Padé approximation, the complexity of which is given as $O(N^3)$ by Ward [10]. In our case, N stands for the order of the S matrix being exponentiated, which is proportional to the number of states in Figure 4. In turn, this is proportional to the maximum queue length that the model can handle in the W_i time interval of case 2.

Other relatively complex steps include the solutions of the systems of linear equations presented in Sections 2.3 and 2.5 (see also Equation 8). If the maximum number of customers in the queue at a tagged arrival is U, and the maximum number of background arrivals between two tagged arrivals is V, then both systems have $O(UV)$ unknowns. Using Gaussian elimination, we obtain a time complexity of $O\left((UV)^3\right)$, which is prohibitively high for practical problems. In our work we have used Thomas’ algorithm [1], which is capable of solving systems with block tridiagonal matrices in linear time, and therefore the time complexity presented by these subproblems is $O(UV)$. It is practical to let $N = U = V$, in which case this can be expressed as $O(N^2)$. In the paper, we chose N so that the probability of the queue length exceeding N was reasonably small.

Finding a specific percentile of the inter–departure time requires the solution of Equation 27 equated to the percentile value. Doing this numerically requires $O(-\log P)$ steps, where P is the desired precision of the result, i.e., the difference between the upper and the lower bound obtained through the bisection method. For example, setting $P = 10^{-4}$ would return the percentile to approximately four fractional decimal digits of accuracy. Since each step requires the computation of Equation 27, which is a complex task in itself, the overall running time complexity of our algorithm may be stated as $O(-N^3 \log P)$. In practice, however, our method is substantially faster than simulations that produce results of comparable accuracy. Only for very high utilizations (above 97.5%) does our model require more time to solve than simulation.

Background arrival rate		0.05	0.30	0.50	0.70
High accuracy	N	20	38	66	172
	$P(k \geq N)$	6.842E-13	1.712E-12	8.547E-12	1.83E-12
	T	0.147	0.543	2.287	115.619
	P_{95}	24.6412	24.5874	24.5031	24.5409
Low accuracy	N	10	16	30	76
	$P(k \geq N)$	8.283E-7	1.125E-5	9.545E-6	6.786E-6
	T	0.073	0.113	0.313	3.34
	P_{95}	24.6411	24.5872	24.5029	24.5408

Table 1: Accuracy and execution time of the numerical computations for a tagged process with $C^2 = 10$. N denotes the maximum queue length in the system, T the time, in seconds, required to carry out the computation, and P_{95} the computed 95th percentile of the tagged inter–departure time.

Background rate	0.05	0.30	0.50	0.70
T	5267.38	6271.61	8042.32	10187.96
P_{95}	24.6412	24.5927	24.501	24.5408
R_{95}	0.008629	0.008319	0.00838	0.007919

Table 2: Results and execution times of a set of simulation experiments. R_{95} stands for the radius of the 95% confidence interval of the 95th percentile, P_{95}. Note that the computed results lie within the confidence intervals.

Table 1 gives the parameters of two sets of computations carried out for a tagged traffic stream with a C^2 value of 10.0 and background arrival rates of 0.05, 0.30, 0.50 and 0.70. The maximum queue length allowed in the computation, N, the probability of overflow, $P(k \geq N)$, i.e., the probability that the queue length goes beyond the size of the truncated state space, the time it took to carry out the computation (T), and the result obtained for the 95th percentile of the inter–departure time (P_{95}) are reported. In the numerical solution of the $F_{P_i}(t) = 0.95$ equation, we have set the precision of the result to 10^{-4} in all cases. The rows marked with "high accuracy" show the results of computations in which the state space was large enough so that any further increase of the number of states would not produce a measurable change in the 95th percentile of the inter–departure time in the double–precision arithmetic of our implementation. This was achieved by making sure that the total probability of reaching a truncated state is on the order of 10^{-12}. In the "low accuracy" group, we have reduced the state space until a change was seen in the first 4 fractional decimal digits of the computed 95th percentile. This typically required the state space to be reduced to less than half of the "high accuracy" values. The timing values were measured on a PC equipped with an Intel Q6600 CPU running at 2.4 GHz and 2 GB RAM. Note that even though the CPU we have used has multiple cores, in order to be able to make a fair comparison, we did not take advantage of multi–threading in either the computations or the simulations.

Table 2 presents a set of simulation results corresponding to the computations shown in Table 1. Again, the squared coefficient of variation of the tagged arrival process was 10.0 and the rate of background traffic was 0.05, 0.30, 0.50 and 0.70. All of the experiments comprised 1001 batches of 10^6 tagged departures; results from the first batch were discarded. For each experiment, we report the time required to complete the simulation on the same hardware that we used for the computations, the value obtained for the 95th percentile of the inter–departure time as well as the radius of the 95% confidence interval of said result.

We note that the computed results lie within the confidence intervals obtained by simulation. However, the computations yielded more accurate results in a time that was several orders of magnitude less than the time required for the simulations.

4 Conclusion

We studied a single server queue with two different arriving streams, a tagged arrival process and a background arrival process. The tagged traffic is assumed to be an Interrupted Poisson Process (IPP) and the background traffic is Poisson. The service time is exponentially distributed and customers are served in a FIFO manner. We obtained numerically the PDF of the inter–departure time of the IPP tagged process, from which we calculated its jitter, defined as a percentile of the inter–departure time.

The numerical procedure requires the solution of a set of infinite Markov processes with absorbing states which can only be solved by truncating the rate matrices. The probability of reaching truncated states may be easily computed in advance. We have also compared results obtained from different levels of truncation and found that when the state space is large enough (but still reasonable in all of our experiments), the percentile values do not change at all in the double–precision floating point arithmetic of our implementation. Furthermore, this model can also be used to obtain accurate distributions and percentiles of various other performance metrics, such as the distribution of the queue length and the end–to–end delay. However, if the utilization of the queue is very high (above 97.5%), then our model may be rendered impractical by the fact that the time it takes to obtain accurate results surpasses that of simulation. For such cases, we have proposed a heavy traffic approximation based on the behavior of the queue at full utilization.

All of our computed results have been verified through simulation. We have found that in order to match the accuracy of the computations through simulation, very long simulation runs are necessary.

We obtained numerical results for the case where the tagged traffic accounts for a small percentage of the utilization in relation to the background traffic. This is realistic in cases where we are concerned with a single flow in the presence of multiple flows. In this case, the 95th percentile of the inter–departure time of the tagged process appears to be unaffected by the presence of the Poisson background traffic for utilizations up to 0.6. After that, it may slightly increase or decrease depending upon the C^2 of the inter-arrival time of the tagged process.

Finally, we note that our model can be easily extended to the case of an MMPP–k tagged arrival processes with an MMPP–k background processes, at a cost of a proportional increase in the state space.

References

1. Benkert, K., & Fischer, R. (2007). An efficient implementation of the Thomas–algorithm for block penta–diagonal systems on vector computers. *Lecture Notes in Computer Science, 4487,* 144–151.
2. Bitran, G. R., & Dasu, S. (1993). Approximating nonrenewal processes by Markov chains: Use of super-Erlang (SE) chains. *Operations Research, 41* (5), 903–923.
3. Dasu, S. (1998). Class dependent departure process from multiclass phase queues: Exact and approximate analyses. *European Journal of Operational Research, 108* (2), 379–404.
4. Grinstead, C. M., & Snell, J. L. (n.d.). Introduction to Probability. (2nd ed.). [On–line book. Accessed 12 Feb 2012.] URL `http://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/book.html`.
5. Kumaran, J., Mitchell, K., & van de Liefvoort, A. (2005). An analytic model of correlations induced in a packet stream by background traffic in IP access networks. *Proceedings of the 19th International Teletraffic Congress,* Beijing, China (pp. 687–696).
6. Mitchell, K., van de Liefvoort, A., & Place, J. (2000). Second–order statistics of an isolated departure stream from a shared buffer with correlated sources. *Proceedings of the 8th International Conference on Telecommunication Systems, Modeling and Analysis,* Nashville, TN (pp. 565–574).

7. Sohraby, K., & Privalov, A. (1999). End–to–end jitter analysis in networks of periodic flows. *Proceedings of IEEE INFOCOM '99. 18th Annual Joint Conference of the IEEE Computer and Communications Societies, 2,* New York, NY (pp. 575–583).
8. Stanford, D., & Fischer, W. (1989). The interdeparture–time distribution for each class in the $\Sigma_i M_i/G_i/1$ queue. *Queueing Systems, 4* (3), 179–191.
9. Stewart, W. J. (2009). *Probability, Markov chains, queues, and simulation.* Princeton, NJ: Princeton University Press.
10. Ward, R. C. (1977). Numerical computation of the matrix exponential with accuracy estimate. *SIAM Journal on Numerical Analysis, 14* (4), 600–610.
11. Whitt, W. (1983). The queueing network analyzer. *The Bell System Technical Journal, 62* (9), 2779–2815.
12. Network performance objectives for IP–based services, ITU–T Standard Y.1541, 2011.