Delay Analysis of Priority Queues with Modulated Traffic

P.G. Harrison and Yu Zhang
Department of Computing
Imperial College London, SW7 2AZ, UK

{yuzhang, pgh}@doc.ic.ac.uk

Abstract

Differentiated services and other scheduling strategies are now widespread in the traditional, 'best effort' Internet. These offer quality of service guarantees for important customers at the same time as supporting less critical applications of lower priority. Since response time, or delay, is a crucial performance metric for delay-sensitive applications, time delays in priority queues have been studied extensively in recent years. We consider a DiffServ node which is modelled as a non-pre-emptive priority queue with modulated arrivals and derive an expression for the probability distribution of the response time using the generating function method. We consider two service classes: *expedited traffic* forms the high priority class and is modelled as a Poisson process whereas best effort traffic is in the low priority class and modelled as a Markov modulated Poisson process. The distribution of service time is general. This queue has many real-world applications; in the example considered here, it could model a DiffServ router which provides service differentiation for signalling or management traffic together with standard data streams. Mean delays are derived as explicit expressions and show very close agreement with simulation. Higher moments can be computed in the same way with more routine algebra.

1 Introduction

In addition to traditional data services, multimedia and real-time applications are becoming indispensable services offered by the best-effort Internet. The future Internet is expected to offer a certain quality of service (QoS) guarantee to some important applications, which today are 'best effort'. As is well understood, packets may suffer some delay and loss at the network nodes during their traversal across a packet-switched network. Therefore, packet loss and end-to-end delay are two crucial performance metrics for Internet QoS. Among these two, the end-to-end delay is more important for real-time applications which are generally highly delay sensitive.

Models and techniques for providing and evaluating quality of service are extensively studied in both academia and industry. The Differentiated Services (DiffServ) model, proposed as a simple and scalable mechanism to fulfil this QoS requirement, aims to offer service differentiation for different classes of flows at each network node [1, 2, 3, 4]. In a DiffServ supported network, traffic is categorised into different classes at the ingress edge nodes. An edge node marks each class of packets with a Per-Hop Behavior (PHB) by writing a DiffServ Code Point (DSCP) into each IP packet's header. For instance, in an IPV6 Diffserv domain, a DSCP is written into the 'Type Of Flow' field in a packet header. Packets belonging to the same class are marked with the same DSCP and experience the same forwarding behavior in the core nodes. Several PHB groups have been defined. In descending order of priority, they are known as Expedited Forwarding (EF) and Assured Forwarding (AF), in addition to Best Effort Forwarding (BEF).

The implementation of the PHBs is by means of buffer management. A priority queues (PQ) is a simple and efficient way to offer differentiated services, and thus a fundamental and essential element in a Diffserv node. It consists of a set of buffers served with priorities. Each class of traffic with a certain QoS requirement enters a separate buffer that is granted a specified priority. The traffic in a higher priority buffer is served before that of all lower priorities. Different priority schemes can be considered in a priority queue. Basically, there are two types of schemes depending on whether or not on-going service can be interrupted; namely non-pre-emptive and pre-emptive priorities. Non-pre-emptive priority mechanisms, where a high priority packet cannot interrupt an on-going service of a lower priority packet, are more popularly deployed but their performances are more complex to analyze. Because of the importance of priority queues in modelling today's computer and communications systems, as well as other Internet elements, their behavior has been studied extensively in the literature in recent years [5, 6, 7, 8].

In this paper, we present a novel approach to the analysis of a non-pre-emptive priority queue with modulated arrivals and derive an expression for the delay probability distribution by the generating function method. We consider two priority classes where the high priority traffic is modelled as a Poisson process and the low priority class as a Markov Modulated Poisson process (MMPP). The distribution of service time is general. This queue has many real-world applications; in particular, it can model a router, which provides service differentiation for signalling and management traffic, together with standard data streams.

The rest of the paper is organised as follows. The main methodology of our approach is presented in section 2, studying the delay performance of a First in First Out (FIFO) queue with modulated traffic, giving a known result by a new mothod. The extension of the approach to a priority queue is addressed in Section 3. Section 4 presents some numerical and simulation results and Section 5 concludes the paper.

2 Main Methodology

To make our approach easy to understand, we illustrate it by first studying an MMPP/G/1 FIFO queue. Its extension to a corresponding priority queue is described in the next section.

2.1 MMPP traffic

In an MMPP/G/1 queue, packets are generated according to a Markov Modulated Poisson process (MMPP), which is a doubly stochastic process where the intensity of a Poisson process is variable, defined by the state of an independent (modulating) Markov chain. Since it is a nice analytical model for both network traffic and real time traffic such as voice streams [9, 10], techniques for analyzing a queueing system with modulated traffic are widespread in the literature. In this subsection, we give the main definitions and derivations relating to MMPP traffic.

We first define the following generating function[1]

$$\mathbf{g}(z,t) = \sum_{n=0}^{n=\infty} z^n \mathbf{p}(n,t) \tag{1}$$

where the element $p_{ij}(n,t)$ is the conditional probability $Pr(K(t) = n, J(t) = j \mid J(0) = i)$, $K(t)$ denotes the number of packets generated by the MMPP in $(0,t)$, $J(0)$ and $J(t)$ are the phases of MMPP at times 0 and t respectively.

Thus, $\mathbf{g}(z,t)$ is the matrix generating function of the number of packets generated by the MMPP in (0,t), conditional on its phase at time 0 (given by the row numbers). It is given by the following expression (our proof can be found in Appendix A but the result is well-known []):

$$\mathbf{g}(z,t) = exp\{[\mathbf{R} + (z-1)\mathbf{\Lambda}]t\} \tag{2}$$

where $\mathbf{R}$ and $\mathbf{\Lambda}$ are the parameters that characterize the MMPP; $\mathbf{R}$ is the generator matrix of the modulating chain and $\mathbf{\Lambda}$ is the diagonal matrix of arrival rates at each state of that chain. They are defined by

$$\mathbf{R} = \begin{bmatrix} -\sum_{i\neq 1} r_{1i} & r_{12} & \dots & r_{1m} \\ r_{2,1} & -\sum_{i\neq 2} r_{2i} & \dots & r_{2m} \\ . & . & \dots & . \\ . & . & \dots & . \\ r_{m1} & r_{m2} & \dots & -\sum_{i\neq m} r_{mi} \end{bmatrix} \tag{3}$$

where $r_{ij} \geq 0$ is the rate at which the modulated process transits from phase i to phase j;

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ . & . & \dots & . \\ . & . & \dots & . \\ 0 & 0 & \dots & \lambda_m \end{bmatrix} \tag{4}$$

where λ_i is the arrival rate when the modulating chain is in phase i.

Further, we define $\mathbf{K}^\sharp(z)$ to be the matrix probability generating function of the number of arrivals occurring during a random time B, conditional on the MMPP phase at the beginning

[1] A matrix is written in bold and a vector is characterized by an arrow on top.

of the service time (row) and giving the MMPP phase at the end (column). We find that

$$\begin{aligned}\mathbf{K}^{\sharp}(z) &= \sum_{n=0}^{\infty} z^n \int_0^{\infty} \mathbf{p}(n,t)dB(t) && (5)\\ &= \int_0^{\infty} (\sum_{n=0}^{\infty} z^n \mathbf{p}(n,t))dB(t) && (6)\\ &= \int_0^{\infty} exp\{[\mathbf{R} + (z-1)\mathbf{\Lambda}]t\}dB(t) && (7)\\ &= B^*(-\mathbf{R} - (z-1)\mathbf{\Lambda}) && (8)\end{aligned}$$

where $B^*(.)$ is the Laplace-Stieltjes transform (LST) of the random time's probability distribution. The element $K^{\sharp}_{i,j}(z)$ refers to the conditional probability generating function of the number of packets generated in time B when the MMPP ends in phase j, given that it started in phase i.

2.2 MMPP/G/1 queues

The queue length and response time delay probability distributions are now well known for MMPP/G/1 queues, usually obtained by standard matrix-analytic methods [11, 12]. The approach presented in this paper is an alternative way to find the delay distribution, but more importantly, this approach is easily extended to priority queues.

Consider now the possible delays a packet may experience when passing through a node in a network. As shown in Fig.1, a *tagged packet*, upon entering the queue, finds the server busy serving some other packet and, at the same time, there may (or may not) be a queue of packets waiting to be served. If this tagged arrival occurs after a partial period U of the on-going service, we find the total time it has to wait until its service starts is the sum of the residual time of the on-going service V plus the total service time of all the packets queueing in front of it at the time of its arrival. More specifically, the queueing time consists of three parts [13]: the first part is the time taken for the server to serve all the queueing packets which were already in the system at the instant (time 0, say) the on-going service began (A); the second part comes from the delay from serving those queueing packets that entered the system during the interval (0, U); and the last part is the residual period V of the ongoing service time. In the event that the tagged packet sees the server idle at the time of arrival, its queueing delay is zero. The total time a packet spends in the system, its response time or delay, is the queueing delay plus its own service time.

2.2.1 Steady State Queueing Length distribution

In order to determine how long it takes the server to finish serving the A packets, we observe the queue just after every service begins and compute the queue length distribution at those times at equilibrium.

Let $t_n : n \geq 0$ denote the successive epochs of a service period beginning. Define X_n and J_n to be respectively the number of packets in the queue (excluding the one in service, if any) and

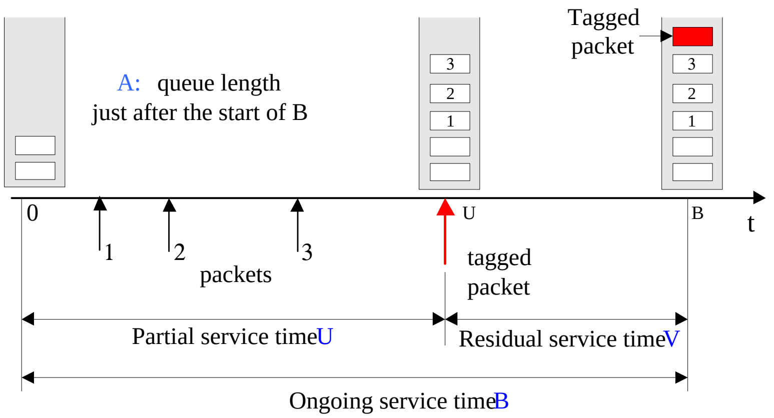

Figure 1: Basic idea of our approach to analysing the delay distribution.

the phase of the MMPP at time t_n^+, both quantities being observed just after a service period begins. The sequence $\{(X_n, J_n, t_{n+1} - t_n) : n \geq 0\}$ defines a semi-Markov chain on the state space $\{1, \dots m\} \times \{0, 1, \dots\}$, where m is the number of phases of the MMPP. The Semi-Markov chain is characterized by the transition probability matrix:

$$\tilde{\mathbf{Q}} = \int_0^\infty \begin{bmatrix} \tilde{\mathbf{T}}_{11}(t) & \tilde{\mathbf{T}}_{12}(t) & \dots & \tilde{\mathbf{T}}_{1m}(t) \\ \tilde{\mathbf{T}}_{21}(t) & \tilde{\mathbf{T}}_{22}(t) & \dots & \tilde{\mathbf{T}}_{2m}(t) \\ . & . & \dots & . \\ . & . & \dots & . \\ \tilde{\mathbf{T}}_{m1}(t) & \tilde{\mathbf{T}}_{m2}(t) & \dots & \tilde{\mathbf{T}}_{mm}(t) \end{bmatrix} dB(t) \tag{9}$$

where $\tilde{\mathbf{T}}_{ij}(t)$ is the infinite matrix:

$$\tilde{\mathbf{T}}_{ij}(t) = \begin{bmatrix} p_{ij}(0,t) + p_{ij}(1,t) & p_{ij}(2,t) & p_{ij}(3,t) & \dots & p_{ij}(n,t) & \dots \\ p_{ij}(0,t) & p_{ij}(1,t) & p_{ij}(2,t) & \dots & p_{ij}(n-1,t) & \dots \\ 0 & p_{ij}(0,t) & p_{ij}(1,t) & \dots & p_{ij}(n-2,t) & \dots \\ 0 & 0 & p_{ij}(0,t) & \dots & p_{ij}(n-3,t) & \dots \\ . & . & . & \dots & . & \dots \end{bmatrix} \tag{10}$$

where $p_{ij}(n,t) = Pr(K(t) = n, J(t) = j | J(0) = i)$ is the conditional probability that the MMPP is in phase j at time t and there are n packets generated in $(0,t)$, given that the MMPP starts in phase i at time 0.

Let the row vector $\vec{\psi}$ be the stationary phase-queue-length distribution (when it exists), and

$$\vec{\psi} = [\psi_1(0),\, \psi_1(1),\, \dots, \psi_2(0),\, \psi_2(1), \dots, \dots, \psi_m(0),\, \psi_m(1),\, \dots] \tag{11}$$

where $\psi_m(n)$ denotes the joint probability that the queue length is n and the MMPP is in phase m at equilibrium. Note that, $\psi_i(0)$ $(1 \leq i \leq m)$ actually includes two cases: where the server is either idle or busy with no other customers waiting in the queue.

The joint phase-queue-length mass function at equilibrium (when it exists) is the solution of

$$\vec{\psi} = \vec{\psi}\mathbf{Q} \tag{12}$$

Now let the row vector $\vec{N}^{\sharp}(z)$ be the (vector) generating function of the probability vector $\vec{\psi}$, where $\vec{N}^{\sharp}(z) = [N_1^{\sharp}(z), \ldots, N_m^{\sharp}(z)]$. Then,

$$N_i^{\sharp}(z) = \sum_{n=0}^{n=\infty} z^n \psi_i(n) \tag{13}$$

Taking the inner-product of eq.(12) with $[z^0\ z^1\ \ldots, z^0\ z^1 \ldots, \ldots, z^0\ z^1\ \ldots]$ gives $\vec{N}^{\sharp}(z)$ on the left hand side. Using eq.(5), eq.(9) and eq.(10) on the right hand side, we obtain

$$\vec{N}^{\sharp}(z)(z\mathbf{I} - \mathbf{K}^{\sharp}(z)) = \vec{\psi}(0)\mathbf{P}(0)(z-1) \tag{14}$$

where $\vec{\psi}(0) = [\psi_1(0), \psi_2(0), \ldots, \psi_m(0)]$. $\mathbf{P}(0) = \int_0^{\infty} \mathbf{p}(0,t)dB(t)$ is the probability matrix that no packet arrives during a service time, jointly with the phase at the end of the service, conditional on the phase of the MMPP at the start of the service.

It is easy to see that $\vec{\psi}(0)\mathbf{P}(0)$ gives the stationary probabilities that the server is idle when the MMPP is in each phase. It is called the level 0 probability vector, or the boundary vector. We denote it by $\vec{\phi} = [\phi_1, \phi_2, \ldots, \phi_m]$. Steady state arguments now immediately show that $\vec{\psi}(0)\mathbf{P}(0)\vec{e}^T = 1 - \rho$, which is the stationary idle probability of the server, where $\rho = \vec{\pi}(\mathbf{\Lambda}/\mu)\vec{e}^T$ is the traffic intensity, where $\vec{e}^T$ is the column vector $[1, \ldots, 1]^T$.[2]

Thus,

$$\vec{N}^{\sharp}(z) = (z-1)\vec{\phi}(z\mathbf{I} - \mathbf{K}^{\sharp}(z))^{-1} \tag{15}$$

There are several methods to determine numerically the level 0 probability vector [14, 15, 16, 17]. In the approach of [14], the roots inside the unit disk ($|z| < 1$) of the characteristic equation $det\{z\mathbf{I} - \mathbf{K}^{\sharp}(z)\} = 0$ are computed. The algorithm in [16, 17] computes the level 0 vector by calculating the stationary probabilities of the starting phase of a busy period. Since the computation of $\vec{\phi}$ is not the emphasis of our paper, we just compute it by the algorithm of [17], which is included in Appendix C for completeness. Having obtained $\vec{\phi}$, the generating function of the joint phase-queue-length probability mass function, $\vec{N}^{\sharp}(z)$ is completely specified.

2.2.2 Waiting Time Distribution

We now determine the matrix generating function of the waiting time distribution's LST for an arbitrary packet in an MMPP/G/1 queue. First we define some further random variables corresponding to a tagged packet which arrives at the system in phase j.

- B denotes the full service time (independent of j).
- U denotes the partial service time.
- V denotes the residual service time.
- A denotes the number of packets queueing at the beginning of the service time during which the tagged packet arrives; the generating function of its joint phase-queue-length distribution is $\vec{A}^{\sharp}(z) = \vec{N}^{\sharp}(z)$.

[2] Appendix B has a more rigorous proof.

- X_j denotes the number of packets that arrived during the partial service time U.
- Q_j denotes the queueing time for the tagged packet.
- W_j denotes the total time the tagged packet spends in the system. It is the sum of the queueing time and the service time B of a packet.

We also use the indicator function I defined by $I_b = 1$ when b is true and $I_b = 0$ when b is false.

Consider the scenario that a tagged packet enters the queue when the MMPP is in phase j, after a partial service time U. We call such a packet a phase j packet. As illustrated above, the total time this tagged packet has to wait before it gets served is the sum of the residual time V and the total service time of all the packets queueing in front of it at the time of arrival. Let n denote the total number of packets queueing in front of it. Among these packets, there are A packets which have been in the queue since the beginning of the on-going service and X packets which arrived during the partial service time U, where $A + X = n$. Thus, for a phase j packet,

$$Q_j = \left(V + \sum_{h=1}^{A-1} B_h + \sum_{k=1}^{X} B'_k \right) I_{A>0}$$

where the random variables B_h and B'_k are independent and identically distributed as a full service time B.

Using conditional expectations and the property that $E[E[Z_1|Z_2, Z_3]|Z_3] = E[Z_1|Z_3]$ for all random variables Z_1, Z_2, we have,

$$\begin{aligned} E\left[e^{-sQ_j} \mid U, V\right] &= E\left[E\left[e^{-s\left(V+\sum_{i=1}^{A-1} B_i+\sum_{j=1}^{X} B'_j\right)I_{A_j>0}} \mid A, X_j, U, V\right] \middle| U, V\right] \\ &= E\left[e^{-sVI_{A>0}} E\left[e^{-sB}\right]^{(A+X_j)I_{A>0}} \middle| U, V\right] \\ &= Pr(A>0) E\left[e^{-sV} B^*(s)^{(A+X_j)} \middle| U, V\right] + Pr(A=0) \\ &= Pr(A>0) e^{-sV} \left[\vec{A}^{\sharp}(B^*(s)) e^{[\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]U}\right]_j + Pr(A=0) \end{aligned}$$

by equation 1. Now, the joint probability density function of U and V is $f(u, v) = \mu b(u + v)$, where $b(w)$ is the probability density function of the service time B and μ is the mean service rate, i.e. the reciprocal of the mean service time $-B^{*\prime}(0)$. Thus, deconditioning on U, V and

$J(U)$ $(=j)$ yields,

$$\begin{aligned}
E\left[e^{-sQ}\right] &= E\left[E\left[e^{-sQ} \mid U, V\right]\right] \\
&= Pr(A=0) + Pr(A>0)\vec{A}^{\sharp}(B^*(s))\mu \int_0^{\infty}\int_0^{\infty} e^{-sv\boldsymbol{I}+[\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]u} b(u+v)\,du\,dv \\
&= Pr(A=0) + Pr(A>0)\vec{A}^{\sharp}(B^*(s))\mu \int_0^{\infty}\int_0^{w} e^{-sw\boldsymbol{I}+[s\boldsymbol{I}+\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]u} b(w)\,du\,dw \\
&= Pr(A=0) + Pr(A>0)\vec{A}^{\sharp}(B^*(s))\mu[s\boldsymbol{I}+\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]^{-1} \times \\
&\qquad \int_0^{\infty}\left(e^{[\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]w} - e^{-sw\boldsymbol{I}}\right) b(w)\,dw \\
&= Pr(A=0) + \\
&\qquad Pr(A>0)\vec{A}^{\sharp}(B^*(s))\mu[K^{\sharp}(B^*(s)) - B^*(s)\boldsymbol{I}][s\boldsymbol{I}+\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]^{-1}
\end{aligned}$$

Equation 15 now yields, with further straightforward simplification and multiplication by $B^*(s)$ to get the LST of the response time distribution,

$$E[e^{-sW}] = \begin{cases} \vec{\phi}B^*(s)s[s\mathbf{I}+\mathbf{R}+(B^*(s)-1)\mathbf{\Lambda}]^{-1}\,\vec{e}^T \quad (s>0) \\ 1 \quad (s=0) \end{cases} \tag{16}$$

Notice that the waiting time distribution of the MMPP/G/1 queue given in eq.(16) is exactly the same as that in [16, 17].

3 Two-Class Priority Queue with MMPP Arrivals

The approach of the previous section generalises well to the study of a non-pre-emptive priority queue. In our study, the priority queue is assumed to have a high priority Poisson flow with rate λ_h and an MMPP flow as its low priority input, which is characterized by a phase transition matrix $\mathbf{R}_l$ and arrival rate matrix $\mathbf{\Lambda}_l$. The high and low priority packets are assumed to have the same service time distribution with a mean service rate of μ.

3.1 High priority packets

Consider a high priority tagged packet. As shown in Fig.2, this packet sees the server busy with some packets queueing (the queue could also be empty). Since it is a high priority packet, it can only see the high priority packets which are queueing ahead of it, if there are any. Low priority packets, even any that came earlier, are transparent to the tagged packet. In this case, the queueing delay of this tagged packet also consists of three parts, similar to the MMPP/G/1 case. The first part is the time for the server to serve all the high priority packets that were already in the queue at the time the on-going service began; the second part comes from serving those high priority packets that enter the system during the partial service time U; and the last part is the residual period V.

Since the high priority packets are Poisson arrivals, in the remainder of this subsection, all the parameters are scalars, unless stated otherwise. The derivations follow the same method as described in section 2, but are much simpler since the matrix operations become scalar. The reader can also refer to our previous paper [13] for reference.

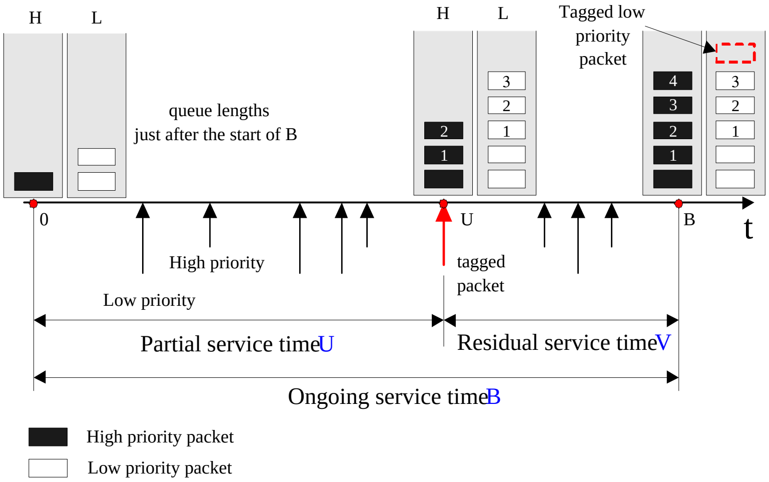

Figure 2: Basic idea of our approach to study the delay distribution of a priority queue.

The generating function of the number of high priority packets that arrive in a service time B is now just $K^\sharp(z) = \int_0^\infty e^{(z-1)\lambda_h t} dB = B^*((1-z)\lambda_h)$. Let $A'^\sharp(z)$ denote the probability generating function for the number of high priority packets A' in the queue at the beginning of a service time. As in [13], this is

$$A'^\sharp(z) = N'^\sharp(z) = \frac{(1-\rho_h)(1-z)}{B^*(\lambda_h(1-z)) - z} \tag{17}$$

where $\rho_h = \lambda_h/\mu$.

Let X' denote the number of high priority packets that arrive during the partial service time U. Following the same approach as before, we find (with a slight abuse of notation using $QL > 0$ to replace the indicator function)

$$E[e^{-sQ}|U, V, A', X', QL > 0] = e^{-sV}(B^*(s))^{A'}(B^*(s))^{X'} \tag{18}$$

Deconditioning on A' and X', we have

$$E[e^{-sQ}|U, V, QL > 0] = e^{-sV} A'^\sharp(B^*(s)) e^{(B^*(s)-1)\lambda_h U} \tag{19}$$

Further deconditioning on U and V yields

$$E[e^{-sQ}|QL > 0] = \frac{A'^\sharp(B^*(s))\mu[B^*(-(B^*(s)-1)\lambda_h) - B^*(s)]}{s + (B^*(s)-1)\lambda_h} \tag{20}$$

In a non-pre-emptive priority queue, the probability that a high priority packet sees the server busy upon entering the queue is equal to $\rho = \frac{\lambda_h}{\mu} + \vec{\pi}\frac{\mathbf{\Lambda}_l}{\mu}\vec{e}^T$. The row vector $\vec{\pi}$ is the stationary phase probability of the low priority MMPP flow. Correspondingly, it finds the system empty with a probability of $1-\rho$. Hence, the generating function of the total time that a high priority packet stays in the system is obtained as:

$$\begin{aligned} E[e^{-sW}] &= B^*(s)\{E[e^{-sQ}|QL>0]\rho + E[e^{-sQ}|QL=0](1-\rho)\} && (21)\\ &= \frac{B^*(s)[(1-\rho)+\rho\mu A'^{\sharp}(B^*(s))[B^*(-(B^*(s)-1)\lambda_h)-B^*(s)]]}{s+(B^*(s)-1)\lambda_h} && (22)\\ &= \frac{\vec{\pi}[(\mathbf{I}-(\lambda_h\mathbf{I}+\mathbf{\Lambda}_l)/\mu)s+(1-B^*(s))\mathbf{\Lambda}_l]B^*(s)\,\vec{e}^T}{s+(B^*(s)-1)\lambda_h} && (23)\end{aligned}$$

Thus, the LST of the delay distribution of a high priority packet in a MMPP/G/1 non-preemptive queue is

$$E[e^{-sW}] = \begin{cases} \frac{\vec{\pi}[(\mathbf{I}-(\lambda_h\mathbf{I}+\mathbf{\Lambda}_l)/\mu)s+(1-B^*(s))\mathbf{\Lambda}_l]B^*(s)\,\vec{e}^T}{s+(B^*(s)-1)\lambda_h} & (s>0)\\ 1 \quad (s=0) \end{cases} \tag{24}$$

3.2 Low priority packets

For a low priority packet, its queueing delay is the sum of four parts, as shown in Fig.2. The first part is the sum of the 'high-priority busy periods'[3] generated by all the packets (A) in the queue at the time the on-going service began; the second part comes from the sum of the busy periods generated by all the packets (X) that enter the system during the partial service time U; the third part arises from the busy periods generated by the high priority packets that arrive during V; and the last part is the residual period V.

Notice that $\mathbf{K}^{\sharp}(z)$ is the matrix generating function of the number of packets, of both high and low priorities, that arrive during a service time B, conditional on the starting phase at the beginning of the service. It is expressed as

$$\mathbf{K}^{\sharp}(z) = B^*(-\mathbf{R}_l-(z-1)(\lambda_h\mathbf{I}+\mathbf{\Lambda}_l)) \tag{25}$$

As expected, we get $\vec{A}^{\sharp}(z)$ as

$$\vec{A}^{\sharp}(z) = \vec{N}^{\sharp}(z) = \vec{\phi}_{hl}(1-z)[B^*(-\mathbf{R}_l-(z-1)(\lambda_h\mathbf{I}+\mathbf{\Lambda}_l))-z\mathbf{I}]^{-1} \tag{26}$$

where $\vec{\phi}_{hl}$ is the level 0 vector of the priority system, representing the probabilities that the server is idle, in each phase. It can be computed by the algorithms given in Appendix C.

Therefore, similarly to deriving the queueing time distribution in a single priority class queue, we determine the queueing time distribution of a phase j low priority packet as

$$E[e^{-sQ_j}|U,V,QL>0] = e^{-(s+(1-M^*(s)\lambda_h))V}\textstyle\sum_{i=1}^m A_i{}^{\sharp}(M^*(s))g_{ij}(M^*(s),U) \tag{27}$$

where the random variable V_h denotes the number of high priority arrivals during V. $M^*(s)$ is the LST of the high priority busy period distribution, where $M^*(s) = B^*[s-(M^*(s)-1)\lambda_h]$; see Appendix D for a short proof.

[3] The elapsed times between a packet's starting service and the departure of the last high-priority packet to arrive during its service period are all stochastically identical 'high priority busy periods' in that any other low priority packets present have no effect.

Deconditioning on U and V, we obtain, analogously to the method of section 2, the LST of the delay distribution for a low priority packet:

$$E[e^{-sW}] = \begin{cases} \vec{\phi}_{hl} B^*(s)(s + (1 - M^*(s))\lambda_h)[s\mathbf{I} + \mathbf{R}_l + (M^*(s) - 1)\mathbf{\Lambda}_l]^{-1}\, \vec{e}^T \ \ (s > 0) \\ \vec{\pi}\, \vec{e}^T \quad (s = 0) \end{cases} \tag{28}$$

3.3 Mean delay

Mean delays are the single most important performance metrics for both high and low priority packets. In this section, we derive the mean delays iasn explicit expressions by differentiating the generating functions of their delay distributions. Higher moments can be obtained by further differentiation at a cost of greater algebraic tedium.

3.3.1 High priority packets

According to eq.(24), we define

$$Q(s) = \frac{T(s)}{G(s)} \tag{29}$$

where $T(s) = \vec{\pi}[(\mathbf{I} - (\lambda_h \mathbf{I} + \mathbf{\Lambda}_l)/\mu)s + (1 - B^*(s))\mathbf{\Lambda}_l]\, \vec{e}^T$, and $G(s) = s + (B^*(s) - 1)\lambda_h$.

Since $G(0) = 0$, we rearrange the terms of eq.(29) as $Q(s)G(s) = T(s)$, and get the first moment of $Q(s)$ by differentiating it twice with respect to s. At $s = 0$, we obtain, indicating an nth derivative by the parenthesised superscript ${}^{(n)}$,

$$Q^{(1)}(0) = 0.5(T^{(2)}(0) - Q(0)G^{(2)}(0))[G^{(1)}(0)]^{-1} \tag{30}$$

Since the mean delay is $\bar{W} = -Q^{(1)}(0) + \mu^{-1}$, we have

Proposition 1 *The mean sojourn time for high priority packets is*

$$\bar{W} = \mu^{-1} + \frac{\lambda B^{(2)}(0)}{2(1 - \rho_h)} \tag{31}$$

where $\lambda = \lambda_h + \vec{\pi}\mathbf{\Lambda}_l \vec{e}^T$, $\rho_h = \lambda_h/\mu$, $B^{(2)}(0)$ *is the second moment of the service time distribution.*

3.3.2 Low priority packets

The derivation of the mean delay for low priority packets is much more complex. According to eq.(28), we define

$$\vec{Q}(s) = \vec{T}(s)[\mathbf{G}(s)]^{-1} \tag{32}$$

where $\vec{T}(s) = \vec{\phi}_{hl}(s + (1 - M^*(s))\lambda_h)$ and $\mathbf{G}(s) = s\mathbf{I} + \mathbf{R}_l + (M^*(s) - 1)\mathbf{\Lambda}_l$.

Some derivations and expressions are shown in table (1) for later use in the calculations.

Since $G(0) = R_l$ is a singular matrix, the mean delay cannot be derived through simple differentiation. Upon rearrangement of terms we have

$$\vec{Q}(s)\mathbf{G}(s) = \vec{T}(s) \tag{33}$$

Table 1: Some derivations and expressions

$$\vec{Q}(0)=\vec{\pi}$$
$$\mathbf{G}(0)=\mathbf{R_l}\quad \mathbf{G}^{(1)}(0)=\mathbf{I}+\mathbf{\Lambda}_l M^{*(1)}(0)\quad \mathbf{G}^{(2)}(0)=\mathbf{\Lambda}_l M^{*(2)}(0)$$
$$\vec{T}(0)=0\quad \vec{T}^{(1)}(0)=\vec{\phi}_{hl}(1-\lambda_h M^{*(1)}(0))\quad \vec{T}^{(2)}(0)=\vec{\phi}_{hl}(-\lambda_h M^{*(2)}(0))$$
$$M^{*(1)}(0)=\frac{B^{*(1)}(0)}{1+B^{*(1)}\lambda_h}\quad M^{*(2)}0=\frac{B^{*(2)}(0)(1-\lambda_h M^{*(1)}(0))^2}{1+B^{*(1)}\lambda_h}$$

Note: 1. $M^{*(1)}(0)$, $M^{*(2)}(0)$, $B^{*(1)}(0)$, $B^{*(2)}(0)$ are the first 2 moments of $M^*(s)$ and $B^*(s)$ respectively. 2. the superscript (i) denotes the ith differentiation with respect to s.

Differentiating eq.(33) twice with respect to s, postmultiplying both sides with $\vec{e}^T$ and setting $s=0$, we obtain

$$2\vec{Q}^{(1)}(0)\mathbf{G}^{(1)}(0)\vec{e}^T+\vec{Q}(0)\mathbf{G}^{(2)}(0)\vec{e}^T=\vec{T}^{(2)}(0)\vec{e}^T \tag{34}$$

Substituting $\mathbf{G}^{(1)}(0)$, $\vec{Q}(0)$, $\mathbf{G}^{(2)}(0)$ and $\vec{T}^{(2)}(0)$ into eq.(34), we have

$$-\vec{Q}^{(1)}(0)\mathbf{\Lambda}_l M^{*(1)}(0)\vec{e}^T=\vec{Q}^{(1)}(0)+0.5M^{*(2)}(0)\vec{\pi}\mathbf{\Lambda}_l\vec{e}^T+0.5M^{*(2)}(0)\lambda_h\vec{\phi}_{hl}\vec{e}^T \tag{35}$$

Since $M^{*(1)}(0)=\frac{B^{*(1)}(0)}{1+B^{*(1)}\lambda_h}=-\frac{1}{\mu(1-\rho_h)}$, eq.(35) can be rewritten as

$$\vec{Q}^{(1)}(0)(\mathbf{\Lambda}_l/\mu)\vec{e}^T=\vec{Q}^{(1)}(0)(1-\rho_h)+0.5M^{*(2)}(0)(1-\rho_h)\vec{\pi}\mathbf{\Lambda}_l\vec{e}^T+0.5M^{*(2)}(0)\lambda_h(1-\rho_h)\vec{\phi}_{hl}\vec{e}^T \tag{36}$$

Returning to eq.(33), differentiate once with respect to s, then add $\vec{Q}(0)\vec{e}^T\vec{\pi}$ to both sides and rearrange terms. Setting $s=0$, we obtain

$$\vec{Q}^{(1)}(0)=\vec{Q}^{(1)}(0)\vec{e}^T\vec{\pi}(\mathbf{G}(0)+\vec{e}^T\vec{\pi})^{-1}+(\vec{T}^{(1)}(0)-\vec{Q}(0)\mathbf{G}^{(1)}(0))(\mathbf{G}(0)+\vec{e}^T\vec{\pi})^{-1} \tag{37}$$

Note that $\vec{e}^T\vec{\pi}(\mathbf{G}(0)+\vec{e}^T\vec{\pi})^{-1}=\vec{e}^T\vec{\pi}(\mathbf{R}_l+\vec{e}^T\vec{\pi})^{-1}=\vec{e}^T\vec{\pi}$ [15]. Substituting $\vec{T}^{(1)}(0)$, $\mathbf{G}^{(1)}(0)$, $\mathbf{G}(0)$ into eq.(37), and postmultiplying by $(\mathbf{\Lambda}_l/\mu)\vec{e}^T$ on both sides, we have

$$\vec{Q}^{(1)}(0)\mathbf{\Lambda}_l/\mu=\vec{Q}^{(1)}(0)\vec{e}^T\rho_l+(\vec{\phi}(1-\lambda_h M^{*(1)}(0))-\vec{\pi}(\mathbf{I}+\mathbf{\Lambda}_l M^{*(1)}(0))(\mathbf{R}_l+\vec{e}^T\vec{\pi})^{-1}(\mathbf{\Lambda}_l/\mu)\vec{e}^T \tag{38}$$

where $\rho_l=\vec{\pi}\frac{\mathbf{\Lambda}_l}{\mu}\vec{e}^T$.

Upon equating right hand sides of eq.(36) and eq.(38) and rearranging, we have

$$\begin{aligned}\vec{Q}^{(1)}(0) &= \frac{1}{1-\rho_l-\rho_h}\{(\vec{\phi}_{hl}(1-\lambda_h M^{*(1)}(0))-\vec{\pi}(\mathbf{I}+\mathbf{\Lambda}_l M^{*(1)}(0))(\mathbf{R}_l+\vec{e}^T\vec{\pi})^{-1}(\mathbf{\Lambda}_l/\mu)\vec{e}^T\\ &\quad -0.5M^{*(2)}(1-\rho_h)\vec{\pi}\mathbf{\Lambda}_l\vec{e}^T-0.5M^{*(2)}(0)\lambda_h(1-\rho_h)\vec{\phi}_{hl}\vec{e}^T\}\end{aligned}$$

Letting $\rho=\rho_h+\rho_l$, we have

Proposition 2 *The mean sojourn time for low priority packets is*

$$\begin{aligned}\bar{W} &= \mu^{-1}+\frac{1}{1-\rho_l-\rho_h}\{(\vec{\phi}_{hl}(1-\lambda_h M^{*(1)}(0))-\vec{\pi}(\mathbf{I}+\mathbf{\Lambda}_l M^{*(1)}(0))(\mathbf{R}_l+\vec{e}^T\vec{\pi})^{-1}(\mathbf{\Lambda}_l/\mu)\vec{e}^T\\ &\quad -0.5M^{*(2)}(1-\rho_h)\vec{\pi}\mathbf{\Lambda}_l\vec{e}^T-0.5M^{*(2)}(0)\lambda_h(1-\rho_h)(1-\rho)\}\end{aligned}$$

Table 2: Simulation Scenarios for MMPP/M/1 Priority Queues

Service rate= 10

$Case1$. $\lambda_H = 0.1$ $\lambda_{L1} = 8.5$ $\lambda_{L2} = 8.5$ $r_{L12} = 0.002$ $r_{L21} = 10$.
$Case2$. $\lambda_H = 0.1$ $\lambda_{L1} = 3.5$ $\lambda_{L2} = 2.5$ $r_{L12} = 0.5$ $r_{L21} = 1$.
$Case3$. $\lambda_H = 2.5$ $\lambda_{L1} = 1.5$ $\lambda_{L2} = 2.5$ $r_{L12} = 0.5$ $r_{L21} = 1$.

Table 3: Performance Evaluation of MMPP/M/1 Priority Queue

	Case1		Case2		Case3	
	*	**	*	**	*	**
Mean Queue Length (High)	0.0187	0.0187	0.0132	0.0133	0.3928	0.3945
Mean Delay (High)	0.1870	0.1869	0.1322	0.1329	0.1572	0.1578
Mean Queue Length (Low)	6.1034	6.1243	0.4728	0.4734	0.3661	0.3721
Mean Delay (Low)	0.7174	0.7205	0.1504	0.1495	0.2051	0.2030

* Simulation Results; ** Numerical Results

4 Analytical and Simulation Results

In this section, we present numerical examples to evaluate the accuracy of the mean delays derived from the proposed method. Notice that we are checking the accuracy of the algorithms implemented, and indeed of the simulation, since the methods are exact.

Three different priority queues are evaluated; their detailed parameters are listed in Table 2. Service times are chosen to have exponential, Erlang-2 and constant distributions. The mean service rate μ is fixed at 10. The results of the mean delay and mean queue length for MMPP/M/1, MMPP/Erlang-2/1 and MMPP/D/1 priority queues are shown in table (3), table (4) and table (5), respectively.

The columns marked ** are the analytical results, those marked * being from simulation. The level 0 vectors $\vec{\phi}_{hl}$ for each case are calculated by the algorithm in Appendix C, where n is set at 50 and k is set at 70. The analytical results for the mean queue length are computed by Little's Formula. Discrete Event Simulation (DES) is used to verify the numerical results. The very well matched results verify the accuracy of the proposed approach, the equations derived and their implementation. Conversely, the simulation itself is also verified for correctness in a mutual validation process between two models.

5 Conclusion

The delay performance of a DiffServ type of node has been investigated by modeling it as a two-class, non-pre-emptive priority queue with modulated traffic. Laplace-Stieltjes transforms

Table 4: Performance Evaluation of MMPP/Erlang-2/1 Priority Queue

	Case1		Case2		Case3	
	*	**	*	**	*	**
Mean Queue Length (High)	0.0169	0.0165	0.0125	0.0125	0.3561	0.3583
Mean Delay (High)	0.1657	0.1651	0.1251	0.1247	0.1428	0.1433
Mean Queue Length (Low)	4.8319	4.8059	0.4384	0.4345	0.3273	0.3250
Mean Delay (Low)	0.5683	0.5654	0.1383	0.1372	0.1786	0.1773

* Simulation Results; ** Numerical Results

Table 5: Performance Evaluation of MMPP/D/1 Priority Queue

	Case1		Case2		Case3	
	*	**	*	**	*	**
Mean Queue Length (High)	0.0144	0.0143	0.0117	0.0117	0.3216	0.3223
Mean Delay (High)	0.1430	0.1434	0.1163	0.1165	0.1288	0.1289
Mean Queue Length (Low)	3.4681	3.4867	0.3975	0.3952	0.2806	0.2779
Mean Delay (Low)	0.4079	0.4102	0.1255	0.1248	0.1535	0.1516

* Simulation Results; ** Numerical Results

of the delay distributions were derived for both high and low priority classes of packets. We also derived the mean delays as explicit expressions by considering the moments of the LSTs, obtained by differentiation at the origin. Comparison of the numerical results and simulation suggests that the proposed schemes are correctly implemented. Besides, the proposed approach can be used to explore some more complex priority queues, such as those in which the service time distributions differ for the high and low priority packets. The method can also be extended to study the priority queue with modulated traffic in both the high and low priority input streams. These ideas are currently under study. More complex queues will probably require approximations to be made, whereupon the validation against simulation will be essential.

Appendix A: Derivation of the matrix generating function g(z,t)

We define $p_{ij}(n,t) = Pr(K(t)=n, J(t)=j|K(0)=0, J(0)=i)$, where $K(t)$ is the number of packets generated in $(0,t)$ and $J(t)$ is the phase of MMPP at time t. By the Chapman-Kolmogorov equations, we have

$$p'_{ij}(n,t) = p_{ij}(n,t)(R_{jj}-\lambda_j) + p_{ij}(n-1,t)\lambda_j + \sum_{h=1,h\neq j}^{m} p_{ih}(n,t)R_{hj} \tag{A.1}$$

In a matrix notation, we have

$$\mathbf{p}'(n,t) = -\mathbf{p}(n,t)\mathbf{\Lambda} + \mathbf{p}(n-1,t)\mathbf{\Lambda} + \mathbf{p}(n,t)\mathbf{R} \tag{A.2}$$

The matrix generating function $\mathbf{g}(z,t) = \sum_{n=0}^{\infty} z^n \mathbf{p}(n,t)$, defined for $|z| \le 1$, satisfies the differential equation

$$\mathbf{g}'(z,t) = \mathbf{g}(z,t)(-\mathbf{\Lambda} + z\mathbf{\Lambda} + \mathbf{R}) \tag{A.3}$$

with the initial condition $\mathbf{g}(z,0) = \mathbf{I}$. Therefore,

$$\mathbf{g}(z,t) = exp\{(\mathbf{R} + (z-1)\mathbf{\Lambda})t\} \tag{A.4}$$

Appendix B: Proof of the equation $\psi(0)\mathbf{P}(0)\vec{e}^T = 1 - \rho$

We know from eq.(14) that

$$\vec{N}^\sharp(z)(z\mathbf{I} - \mathbf{K}^\sharp(z)) = \vec{\psi}(0)\mathbf{P}(0)(z-1) \tag{B.1}$$

or, equivalently

$$\vec{N}^\sharp(z)(z\mathbf{I} - \int_0^\infty exp\{[\mathbf{R} + (z-1)\mathbf{\Lambda}]t\}dB(t)) = \vec{\psi}(0)\mathbf{P}(0)(z-1) \tag{B.2}$$

where $P_{ij}(0)$, the $(i,j)^{th}$ element of $\mathbf{P}(0)$, denotes the conditional probability that no packets arrived during a service period of length B and the phase ends in j, given that the phase started at i.

Upon differentiation of both sides of eq.(B.1), we get

$$\vec{N}^\sharp(z)(\mathbf{I} - \mathbf{K}^{\sharp(1)}(z)) + \vec{N}^{\sharp(1)}(z)(\mathbf{I} - \mathbf{K}^\sharp(z)) = \vec{\psi}(0)\mathbf{P}(0) \tag{B.3}$$

where the superscript (i) denotes differentiation i times with respect to z. Taking limits as $z \to 1$, we then have

$$\vec{N}^\sharp(1)(\mathbf{I} - \mathbf{K}^{\sharp(1)}(1)) + \vec{N}^{\sharp(1)}(1)(\mathbf{I} - \mathbf{K}^\sharp(1)) = \vec{\psi}(0)\mathbf{P}(0) \tag{B.4}$$

It is well known that $\vec{N}^\sharp(1) = \vec{\pi}$, which is the phase probability vector at equilibrium.

$\mathbf{K}^\sharp(1)$ is expressed as

$$\mathbf{K}^\sharp(1) = \int_0^\infty exp\{\mathbf{R}t\}dB(t) = \int_0^\infty (\mathbf{I} + \mathbf{R}t + \frac{\mathbf{R}^2t^2}{2!} \ldots .)dB(t) \tag{B.5}$$

From eq.(B.5), we see that $\mathbf{K}^\sharp(1)$ is actually the phase transition matrix of the MMPP after a service time, conditional on its starting phase at the beginning of the service.

Expanding the matrix exponential, $\mathbf{K}^{\sharp(1)}(z)$ can be calculated as

$$\begin{aligned}
\mathbf{K}^{\sharp(1)}(z) &= \int_0^\infty [exp\{(\mathbf{R}+(z-1)\mathbf{\Lambda})t\}]' dB(t) \\
&= \int_0^\infty [\mathbf{I}t + (\mathbf{R}+(z-1)\mathbf{\Lambda})t + \frac{(\mathbf{R}+(z-1)\mathbf{\Lambda})^2 t^3}{2!} + \frac{(\mathbf{R}+(z-1)\mathbf{\Lambda})^3 t^3}{3!} + \ldots]' dB(t) \\
&= \int_0^\infty [\mathbf{I}t + (\mathbf{R}+(z-1)\mathbf{\Lambda})t + \frac{(\mathbf{R}^2+(z-1)\mathbf{R\Lambda}+(z-1)\mathbf{\Lambda R}+(z-1)^2\mathbf{\Lambda}^2)t^2}{2!} \\
& + \frac{(\mathbf{R}^3+(z-1)\mathbf{R\Lambda R}+(z-1)\mathbf{\Lambda RR}+(z-1)^2\Lambda^2\mathbf{R}+(z-1)\mathbf{R}^2\mathbf{\Lambda}+(z-1)^2 R\mathbf{\Lambda}^2)t^3}{3!} \\
& + \frac{((z-1)^2\Lambda\mathbf{R\Lambda}+(z-1)^3\mathbf{\Lambda}^3)t^3}{3!} + \ldots]' dB(t) \\
& = \int_0^\infty [\mathbf{\Lambda}t + \frac{(\mathbf{R\Lambda}+\mathbf{\Lambda R}+2(z-1)\mathbf{\Lambda}^2)t^2}{2!} \\
& + \frac{(\mathbf{R\Lambda R}+\mathbf{\Lambda RR}+2(z-1)\mathbf{\Lambda}^2 R+\mathbf{R}^2\mathbf{\Lambda}+2(z-1)\mathbf{R\Lambda}^2)t^3}{3!} \\
& + \frac{(2(z-1)\mathbf{\Lambda R\Lambda}+3(z-1)^2\mathbf{\Lambda}^3)t^3}{3!} + \ldots] dB(t)
\end{aligned}$$

We note that $\vec{\pi}\cdot\mathbf{R}=0$, and then get

$$\vec{N}^\sharp(1)(\mathbf{I}-\mathbf{K}^{\sharp(1)}(1)) = \vec{\pi}\left(\mathbf{I}-\mathbf{\Lambda}\int_0^\infty\left(\mathbf{I}t+\frac{\mathbf{R}t^2}{2!}+\frac{\mathbf{R}^2t^3}{3!}+\ldots\right)dB(t)\right) \tag{B.6}$$

Therefore, eq.(B.4) can be rewritten as

$$\begin{aligned}
&\vec{\psi}(0)\mathbf{P}(0) = \\
&\quad \vec{\pi}(\mathbf{I}-\mathbf{\Lambda}\int_0^\infty(\mathbf{I}t+\frac{\mathbf{R}t^2}{2!}+\frac{\mathbf{R}^2t^3}{3!}+\ldots)dB(t)) + \vec{N}^{\sharp(1)}(1)(\mathbf{I}-\int_0^\infty(\mathbf{I}+\mathbf{R}t+\frac{\mathbf{R}^2t^2}{2!}\ldots.)dB(t))
\end{aligned}$$

Post-multiplying both sizes of eq.(B.7) by the column vector $\vec{e}^T=(1,\ldots,1)^T$, we get

$$\vec{\psi}(0)\mathbf{P}(0)\vec{e}^T = 1-\vec{\pi}(\mathbf{\Lambda}/\mu)\vec{e}^T = 1-\rho \tag{B.7}$$

where μ is the reciprocal of the mean service time, i.e. the service rate. Note that $\mathbf{R}\vec{e}^T=\vec{0}$ and let $\rho=\vec{\pi}(\mathbf{\Lambda}/\mu)\vec{e}^T$, denoting the traffic intensity. Then $1-\vec{\pi}(\mathbf{\Lambda}/\mu)\vec{e}^T=1-\rho$ stands for the equilibrium idle probability of the server. The vector $\vec{\psi}(0)\mathbf{P}(0)$ is actually the level 0 vector $\vec{\phi}$, which denotes the steady state probability that the system is empty at each phase.

Appendix C: Computation of the level 0 vector $\vec{\phi}$

The following algorithm was introduced by [16, 17] to compute numerically the level 0 vector of an MMPP/G/1 queue.

1. Compute the stochastic matrix $\mathbf{G}$ iteratively as follows.
   Start with $G_0 = 0$, and for $k = 0, 1, 2, \ldots$ calculate
$$\mathbf{H}_{n+1,k} = [\mathbf{I} + \theta^{-1}(\mathbf{R} - \mathbf{\Lambda})]\mathbf{H}_{n,k} + \theta^{-1}\mathbf{\Lambda}\mathbf{H}_{n,k}\mathbf{G}_k, n = 0, 1, 2 \ldots \tag{C.1}$$
$$\mathbf{G}_{k+1} = \sum_{n=0}^{\infty} \gamma_n \mathbf{H}_{n,k} \tag{C.2}$$
   where $\mathbf{H}_{0,k} = \mathbf{I}$, $\theta = \max(\lambda_j - R_{jj})$ and $\gamma_n = \int_0^\infty e^{-\theta x}((\theta x)^n/n!)dB(x)$. The sequence $\mathbf{G}_k$ converges monotonically to $\mathbf{G}$.

   The (i, j) component of $\mathbf{G}$ is the probability that a busy period in phase i ends in phase j.

2. Calculate the stationary probability distribution of the Markov chain with transition matrix $\mathbf{G}$ by
$$\vec{g} = (g1, g2) = \frac{1}{G_{12} + G_{21}}(G_{21}, G_{12}) \tag{C.3}$$

3. $\vec{\phi} = (1 - \rho)\vec{g}$ where $(\phi)_j$ is the joint stationary probability of the system being empty and the phase of the MMPP being j at an arbitrary point in time. This is called the level 0 vector.

Appendix D: Derivation of the high priority busy period $M^*(s)$

Busy periods $M^*(s)$ are independent of each other and the busy period generated by the i^{th} packet in the queue depends only on its own service time B_i. Suppose Z_i packets arrive during the service of the i^{th} packet. Then we have

$$\begin{aligned} E[e^{-s(M_i - B_i)}|B_i] &= E[E[e^{-s(M_{i1} + \ldots + M_{iZ_i})}|Z_i, B_i]|B_i] \\ &= E[M^*(s)^{Z_i}|B_i] \\ &= g[M^*(s), B_i] = e^{B_i[(M^*(s)-1)\lambda_h]} \end{aligned}$$

Therefore, an arbitrary packet in the queue has a busy period distribution with LST

$$\begin{aligned} M^*(s) &= E[e^{-sB}E[e^{-s(M-B)}|B]] \\ &= E[e^{-sB}g[M^*(s), B]] \\ &= B^*[s + (1 - M^*(s))\lambda_h] \end{aligned}$$

References

[1] M. May, J. Bolot, A. Jean-Marie, and C. Diot, "Simple performance models of differentiated services schemes for the internet," in *Proceedings of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'99)*, vol. 3, pp. 1385–1394, 21–25 March 1999.

[2] L. Nguyen, T. Eyers, and J. Chicharo, “Differentiated service performance analysis,” in *Proceedings of Fifth IEEE Symposium on Computers and Communications 2000 (ISCC’00)*, pp. 328–333, 3–6 July 2000.

[3] C. Chassot, F. Garic, G. Auriol, A. Lozes, E. Lochin, and P. Anelli, “Performance analysis for an IP differentiated services network,” in *Proceedings of IEEE International Conference on Communications (ICC’02)*, vol. 2, pp. 976–980, 28 April-2 July 2002.

[4] B. Wydrowski, M. Zukerman, C. H. Foh, and B. Meini, “Analytical performance evaluation of a two class diffserv link,” in *Proceedings of the 8th International Conference on Communication Systems, 2002 (ICCS’02)*, vol. 1, pp. 373–377, 25-28 Nov. 2002.

[5] G. Bolch, L. Essafi, and H. de Meer, “Dynamic priority scheduling for proportional delay differentiated services,” in *Proceedings of the 2001 Aachen International Multiconference on Measurement, Modelling and Evaluation of Computer and Communication Systems*, Aachen, Germany, September 11-14, 2001.

[6] A. Andres, G. Bolch, and L. Essafi, “An adaptive waiting time priority scheduler for the proportional differentiation model,” in *Proceedings of the High Performance Computing Symposium 2001*, Seattle, USA, April 22nd - 26th, 2001.

[7] B. Choi, B. Shin, K. Choi, D. Han, and J. Jang, “Priority queue with two-state markov-modulated arrivals,” *Communications, IEE Proceedings*, vol. 145, pp. 152–158, June 1998.

[8] M. Lee, Y. Mun, and B. Kim, “Performance analysis of delay-loss priority mechanism using markov modulated arrival stream,” *Communications, IEE Proceedings*, vol. 144, pp. 311–315, Oct. 1997.

[9] S. Kang, H. K. Yong, D. Sung, and B. Choi, “An application of Markovian arrival process (MAP) to modeling superposed ATM cell streams,” *IEEE Transactions on Communications*, vol. 50, pp. 633–642, April 2002.

[10] L. Muscariello, M. Meillia, M. Meoand, M. Marsanand, and R. Cigno, “An MMPP-based hierarchical model of Internet traffic,” vol. 4, pp. 2143–2147, June 2004.

[11] M. F. Neuts, “Matrix-geometric solutions in stochastic model: An algorithmic approach,” *Baltimore, MD: Johns Hopkins University Press*, 1981.

[12] V.Ramaswami, “The n/g/1 queue and its detailed analysis,” *Adv. Appl Prob.*, vol. 12, pp. 222–261, March 1980.

[13] P.G.Harrison, “Teaching M/G/1 theory with extension to priority queues,” *IEE Proceeding on Comput. Digit. Tech*, vol. 147, pp. 23–26, January 2000.

[14] J. W. M. Guo-Liang Wu, “Computational methods for performance evaluation of a statistical multiplexer supporting bursty traffic,” *IEEE/ACM Transactions on Networking*, vol. 4, pp. 386–397, June 1996.

[15] Y. L. J. N. Daigle and M. N. Magalhaes, “Discrete time queues with phase dependent arrivals,” *IEEE Transactions on Communications*, vol. 42, pp. 606–614, Feburary/March./April 1994.

[16] H. Heffes and D. M. Lucantoni, “A markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance,” *IEEE Journal on Selected Areas in Communications*, vol. SAC-4, pp. 856–868, September 1986.

[17] W. Fischer and K. Meier-Hellstern, “The markov-modulated poisson process (MMPP) cookbook,” *Performance Evaluation*, vol. 18, pp. 149–171, 1992.