Optimization of Buffer Sizes in a Queue System with Customers of Multiple Classes and Different Delay and Loss Requirements

Lobna Eid AL-Tyeb, Safia M. Ezzat*

Abstract

In this paper researcher analyse the performance of a queue management policy on a network switch, which receives two classes of traffic with different quality requirements in terms of delays and losses. It is shown how the state probabilities of this system can be obtained recursively, and from them the average delays of both types of traffic, as well as their loss rates. All these quantities are involved in calculating the performance of the system through an objective function that represents a weighting of the cost caused by the delays produced in priority traffic (in real time) and the losses caused by non-priority traffic (in non-real time).

Keywords: Buffer size, delays, Real-time traffic, non-real-time traffic

*Statistics Dept, Faculty of Commerce, Al-Azhar University, Girls' Branch ,Cairo, Egypt
Corresponding author's email: lobnaalatyeb@azhar.edu.eg

Introduction

Today's broadband communications networks are designed to flexibly and efficiently support and accommodate a wide variety of services such as voice, video, data, and their multimedia combinations. To meet these objectives, various network standards have been designed, among which ATM stands out as one of the most important. In this standard, the information that travels through the network is organized into small fixed-length packets called cells. The different types of applications that make use of the network may vary in their service requirements. Thus, for example, real-time applications, such as videoconferencing, require extreme performance in terms of throughput, delay, delay variation and the loss rate. The increasing generalization of these applications has made the urgent need to provide network services with guaranteed performance and to develop the algorithms that support these services [see Zhang, H. (1995)]. One of the fundamental mechanisms to be able to provide services with guaranteed performance is the choice of the packet service discipline in the switch. In a packet switched network, packets from different connections interact with each other on each switch, and without proper control these interactions can affect network performance. The service discipline of the switching node controls the order in which the packets are served and determines how the packets from the different connections interact. One way to achieve this objective is by implementing, in the switch buffers, priority mechanisms capable of controlling the time or space dedicated to each cell. Due to its simplicity, speed and lower cost of implementation, in high-speed switches priority mechanisms are preferred that control the space (capacity) available in the buffer, in which they must take into account the arrival time of each cell and control its residence time on the switch.

In recent years, different authors[i] have proposed a series of new service disciplines that provide guaranteed performance per connection for high-speed packet-switched networks. The interaction produced between the different packets that arrive at a switch gives rise to a distortion in the distribution of outgoing traffic, which can cause serious damage to those services that require a certain level of synchrony or regularity in the transmission of their cells. The only way to restore the traffic pattern inside the networks is by using non-conservative work schemes, Zhang (1995). These schemes, although they manage to control the distortion of the traffic pattern in the connections, and therefore are capable of providing performance

with limited delay variation, have the disadvantage that they increase the average delay of the packets and underuse the capacity of the transmission channel. When using conservative work schemes, one has to resort to disciplines based on timestamps since they have the flexibility to provide the required services. Examples of this type widely referenced in the literature are WFQ (Weighted Fair Queuing). In this work, a novel buffer management scheme based on rounds, conservative work and guaranteed rate is presented, which takes into account the arrival pattern of two kinds of traffic. In particular, we have considered that these traffic classes correspond, respectively, to real-time traffic (voice, video) and non-real-time traffic (data). The mechanisms based on rounds basically consist of switching the service between the different input channels to the switch. There are multiple policies for running the rounds. Knightly, E et all (1995) introduce the simple attend to one cell of each channel, from the first to the last, and start over from the first channel. A variation of this policy consists in assigning priorities to the channels according to the traffic class, serving in each round more cells in the channels of higher priority than in the channels of lower priority; for example, three cells of a video channel could be served for each cell of a data channel. Other policies specify thresholds: as long as there are fewer than n cells waiting on channel A, channel B is served; once there are n cells in A, the service is dedicated exclusively to A until there are fewer than m cells in A. The mechanism that we present incorporates an additional buffer that is shared by the two traffic classes in the input buffers of each channel. This system is managed by a policy that, to a certain extent, takes into account the arrival times of the cells without the need to use time stamps that specify these times. This policy provides a reduction in the delays of high priority cells (real time), without significantly affecting the performance given to low priority traffic (non-real time).

The system design is shown in the following figure 1:

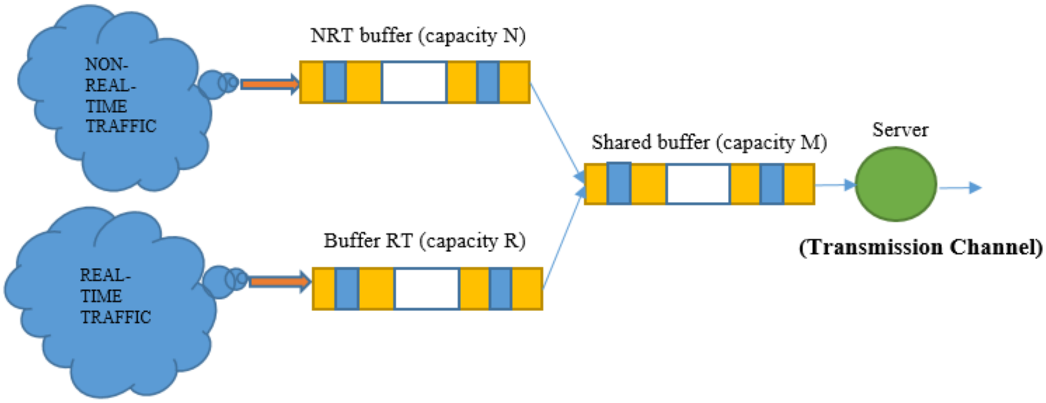


Figure 1.

Real-time traffic (RT) accesses its own buffer, which has a limited capacity for R cells[ii]. Likewise, non-real-time traffic (NRT) accesses a second buffer with capacity N. Cells coming out of both buffers are mixed in a third shared buffer, with capacity for M cells. The output of this third buffer is the transmission channel, over which both traffics are multiplexed. System management occurs as follows:

1. At any given time, there can be only one NRT cell in the shared buffer, and at most M-1 RT cells.
2. If there are no NRT cells in the system, the first NRT cell to reach the system goes directly and instantly to the shared buffer. If there is already a NRT cell in the shared buffer, any new NRT cells arriving in the system are incorporated into the NRT buffer.
3. Every time a RT cell arrives, it directly accesses the shared buffer if there is less than M-1 RT cells; otherwise, the RT buffer is incorporated.

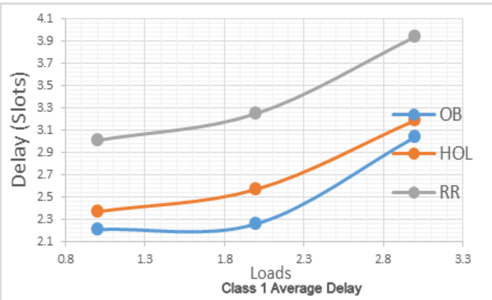
4. Every time a RT (NRT) cell is transmitted, the first RT buffer cell (NRT), if any, instantly queues the shared buffer.
5. When an input buffer is full, cells that reach it are rejected.

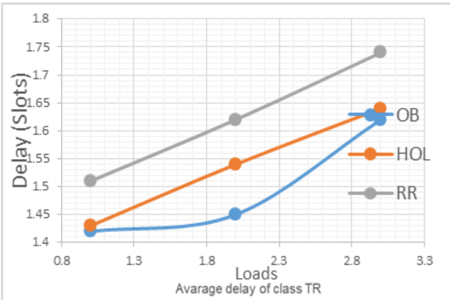
As can be seen from this description, the mechanism for managing these queues is very simple and, therefore, its operation in practice is very fast, which constitutes one of its main advantages from a technical point of view. Furthermore, it should be noted that the operations of selection and introduction of cells in the output buffer do not generate any additional delay when there are cells in said buffer or the output link is busy.[iii]

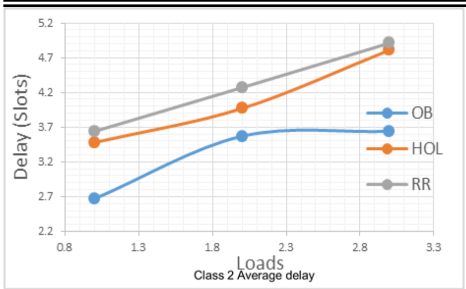
The system control parameters are the sizes of the buffers, N, R and M. These values must be chosen carefully, so that the requirements of the network traffic are met. These requirements are specified in minimizing the delay for the RT traffic and minimizing the losses for the NRT traffic, without this being achieved at the cost of excessively increasing the losses for the RT traffic and the delay for the NRT traffic. If we value in α the cost of the delay per unit time and per cell for the RT traffic and in β the cost of the loss of a cell in the NRT traffic, our objective is to find the values of N, R and M that minimize the long-term average cost per unit of time for the system:

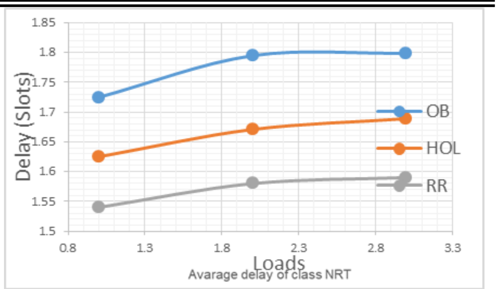
$$\varphi(M,N,R) = \lim_{t\to\infty} \frac{\propto E[Y_{RT}(t)] + \beta E[L_{NRT}(t)]}{t}$$

where $E[Y_{RT}(t)]$ is the total mean delay accumulated by RT clients up to t, and $E[L_{NRT}(t)]$ is the mean number of NRT cells lost up to t. This minimum must be found with the restriction that the average waiting time for NRT clients does not exceed a threshold W1 and the average loss rate for RT clients does not exceed the threshold g_1. We have carried out various simulations comparing this management policy with two other policies based on rounds. For comparison, we have used an RR (Round Robin) policy, consisting of serving k RT class cells for each NRT class cell, and a Head of the Line policy (HOL), consisting of emptying the RT queue each time it serves to a predetermined number j of NRT cells. These simulations were carried out with the YATS (*Yet another Tinny Simulator*) ATM network simulator, and the classes of network objects that would implement these management policies had to be designed. The code for these objects is available at [YATS]. The objective of the simulations has been to check the behaviour of the three policies under different load conditions and service relationships for each class of customer. In all cases, the control parameters of the different management mechanisms described have been arranged in such a way that the service rates dedicated to each type of traffic are as similar as possible in all cases, so that the comparison makes sense. In particular, the results shown below correspond to the case where, on average, two RT cells are served for each NRT cell. We have denoted as OB ($Output\ Buffer$) the results obtained with the new proposed policy.









The graphs on the left correspond to high load conditions, and those on the right to medium load conditions of the network. In the case of high load, the simulations were carried out with an initial load of 80% (with 60% corresponding to RT traffic and 20% to NRT); To appreciate the behaviour of the different policies when a type of traffic increases its flow, the effect of the increase in the rate of each of the traffic classes by 11.1% was analysed; going to generate a throughput of 86.6% when the increase is in RT traffic and 82.2% when the increase is in NRT traffic. The graphs on the right show the behaviour at half load; now the RT traffic flow is 42% and that of the NRT 14%, generating a total load of 57.1%. As in the previous case, the effect of the increase in the rate of each of the traffic classes by 16.6% was analyzed, going to a flow of 64.2% when the increase is RT traffic and 59.5% when the increase is NRT traffic. As can be seen in these graphs, the policy presented is, in general, better than HOL (it almost always produces delays less than or equal to this for the two classes of traffic in all load conditions), and reduces waiting times for RT traffic compared to RR policy[iv], albeit at the cost of increasing NRT traffic wait times. It should also be noted that in the simulations it is observed that the new policy reduces the maximum delays in both types of traffic, thus causing less distortion in the traffic pattern at the exit of the system. This empirical verification of the good properties of this management policy invites us to carry out an analytical study of it, which allows us to have the optimal values of the system control parameters in each case.[v]

In what follows we will assume that both classes of traffic arrive at the system according to two Poisson processes, with parameters λR and λN respectively. Likewise, we will assume that the service time follows a general probability distribution, identical for both types of traffic, although later we will particularize the results for deterministic service. This is reasonable for a communications network, where the speed of the channel (which is the system server)[vi] is constant and identical regardless of the type of information that is transmitted. More debatable is the Poisson process hypothesis for arrivals, which is used basically because it allows sufficient simplicity in the analytical treatment of the problem. However, the technique used is generalizable to models with Erlangen arrivals or with phase-type distributions, which allow covering more general types of traffic. In any case, the management and control mechanism presented has demonstrated its efficiency in simulations with non-Poisson input traffic. In particular, various tests have been carried out when cells of both classes follow bursty traffic patterns, generated by mixing several ON-OFF sources. In each case, during the OFF periods, of exponential durations with different means depending on the source of origin, no cells are generated; during ON periods, regularly spaced cells of also exponential durations are generated. In the simulations, it can be observed that, with these traffic patterns, the broad behaviour of the delays produced in the switch following the three indicated policies does not differ from that observed for Poisson traffic, although the values are, of course, different[vii].

2. DESCRIPTION OF THE SYSTEM STATUS

The state of the previous queuing system at time t can be described by the triple (N_{TR} (t), N_{NRT} (t), S_{NRT} (t)) where:

N_{TR} (t): is the total number of TR clients in the system at time t.

N_{NRT} (t): is the total number of NRT clients in the NRT buffer at time t.

S_{NRT} (t): is the position occupied by the (only) NRT client in the shared queue, if there are any of these clients in said queue; otherwise this variable is 0.

Since the buffers are finite, this system necessarily reaches equilibrium. We will then call:

(NRT, NNRT, SNRT) = $\lim_{t\to\infty} N_{TR}(t), N_{TNR}(t), S_{TNR}(t)$

According to the system management policy, only states of the form are possible:

(k, 0,0), $0 \leq k \leq R + M - 1$

(k, n, m), $m-1 \leq k \leq R + M-1$, $0 \leq n \leq N$, $1 \leq m \leq M$

that constitute a total of

M + R + M (N +1)$(R + \frac{M+1}{2})$ possible states.

Our first objective will be to determine the probabilities at equilibrium, π_{ijk}, of the different states of the system. To do this, in the same way as in the classical treatment of the M / G / 1 tails, we could calculate these probabilities from the Markov chain embedded in the output instants. To do this, it is enough to calculate the transition probabilities between two successive exit instants of the system. This is a simple but laborious task. Thus, calling ak the probability that k RT clients will arrive during a service time, and cj the probability that j NRT will arrive we would have, for example:

$p_{000,\,k00} = a_k\, c_0,\ 0 \le k \le R + M - 3$

The calculation is more complicated when there are arrivals of both classes of customers between two departures; If there was no NRT client in the shared queue at the first exit, the order of arrival must be determined to find out in which position the first NRT client to access it is located in this queue. If we call:

φ_{rs} (j) = Prob (if r RT clients and s NRT clients arrive between two successive exits, the first of the NRT clients arrives in the j-th position, $1 \le j \le r + s$)

it can be proved that:

$$\varphi_{rs}(j) = \frac{\binom{r+s-j}{s-1}}{\binom{r+s}{s}}$$

and we would have, for example,

$$p_{000,knm} = \frac{\lambda_R}{\lambda_{R+\lambda_N}} a_k\ C_{n+1}\ \varphi_{kn+1}(m),\ 1 \le k \le R+M-3,\ 1 \le m \le M-2,\ 0 \le n \le N-1$$

If we denote by E_m the set of states for which $S_{NRT} = m$ (ordering within this set the states in increasing order of N_{TR}

and N_{NRT}), it can be shown that the transition matrix is of the form:

$$P = \begin{array}{cc} & \begin{matrix} E_0 & E_1 & E_2 & & E_{M-1} & E_M & & \end{matrix} \\ \begin{matrix} E_0 \\ E_1 \\ E_2 \\ E_3 \\ \\ \\ E_M \\ \\ \\ \end{matrix} & \begin{bmatrix} A_{00} & A_{01} & A_{02} & \cdots & A_{0M-1} & A_{0M} & \\ A_{01} & A_{12} & A_{12} & \cdots & A_{1M-1} & A_{1M} & \\ 0 & A_{21} & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & A_{32} & \cdots & 0 & 0 & \\ \cdots & . & . & . & . & . & \\ . & . & . & . & . & . & \\ 0 & 0 & 0 & \cdots & A_{MM-1} & 0 & \\ & & \vdots & & & & \ddots \\ & & & & & & \cdots \end{bmatrix} \end{array}$$

Where the matrices A_{ij} in turn can be decomposed into sub-boxes of triangular or very hollow matrices. Although it is possible to obtain the stationary probabilities π_{ijk} from this matrix by solving the system:

$$\sum\nolimits^{\pi P\ =\pi}_{\pi_{ijk=1}}$$

it is a complex numerical problem. Therefore we will proceed in another way to find the π_{ijk} probabilities.

3. CALCULATION OF THE STATIONARY PROBABILITIES OF THE SYSTEM

The process $\{(N_{TR}(t), N_{NRT}(t), S_{NRT}(t)), t \geq 0\}$ is regenerative. This process regenerates itself in those moments in which an arrival occurs that finds the empty system. If we define a cycle as the time elapsed between two arrivals that find the empty system, and we call:

T = Length of one cycle

T_{ijk} = Total time, during a cycle, in which the system is in the state (i, j, k). So, according to the theory of regenerative processes:

$$\pi_{ijk} = \frac{E[T_{ijk}]}{E[T]} \tag{1}$$

In particular, since arrivals are Poisson, we have

$E[T_{000}] = \frac{1}{(\lambda_R + \lambda_R)E[T],}$ and therefore: (2)

Following the line of argument of Tijms (1986)[viii] it is possible to construct recursive equations to calculate the probabilities π_{ijk} using the previous expressions. For this we define:

N_{ijk} = Number of clients served during a cycle that leave the system in state (i, j, k) upon exit.

A (i, j, k), (a, b, c) = Average time, during a service that started with the system in the state (i, j, k), in which the system remains in the state (a, b, c).

Noting now that:

- the first service of a cycle begins with a customer present (which can be TR or NRT),
- during a service the system can reach the state (a, 0,0) only if it started in a state (i, 0,0) with $i \leq a$,
- during a service, the system can reach state (a, b, c) with $1 \leq c \leq M$ only if the service started with the system in some state (i, j, c), with $i \leq a$, $j \leq b$ (since the position c of the NRT client in the shared queue is not altered during a

service time), or if the system started in some state (i,0,0) with i≤ a, i <c,

We can write the following equations:

$if\ \ c-1 \le a \le R+M-1, 1 < c \le M, 0 \le b \le N$:

$$E[T_{abc}] = \frac{\lambda_R}{\lambda_R+\lambda_N} A_{(1,0,0)(a,b,c)} + \sum_{i=c-1}^{a}\sum_{j=0}^{b} E[N_{ijc}]A_{(i,j,c)(a,b,c)} + \sum_{i=1}^{a} E[N_{i00}]A_{(i,0,0)(a,b,c)} \qquad (3)$$

$Si\ \ 0 \le a \le R+M-1, 0 \le b \le N$:

$$E[T_{ab1}] = \frac{\lambda_R}{\lambda_R+\lambda_N} A_{(0,0,1)(a,b,1)} + \sum_{i=0}^{a}\sum_{j=0}^{b} E[N_{ij1}]A_{(i,j,1)(a,b,1)}$$

$Si\ \ 0 \le a \le R+M-1,$:

$$E[T_{a00}] = \frac{\lambda_R}{\lambda_R+\lambda_N} A_{(1,0,0)(a,0,0)} + \sum_{i=0}^{a} E[N_{i,0,0}]A_{(i,0,0)(a,0,0)}$$

Since arrivals and departures in this system occur one at a time, when the system is in equilibrium the mean number E [N_{abc}] of departures leaving the system in the state (a, b, c) is equal to the mean number arrivals that find the system in this state. Now, since arrivals are produced according to a Poisson process, (provided that the system has available space to accommodate each new arrival), this last average number is$\lambda^*_{abc}\ \ [T_{abc}]$

Where :

$$\lambda^*_{abc} = \begin{cases} \lambda_R + \lambda_N & \text{Si a <R+M-1y b + c < N + 1} \\ \lambda_R & \text{Si a <R+M-1y b + c} \ge \text{N + 1} \\ \lambda_N & \text{Si a <R+M-1y b + c< N + 1} \\ 0 & \text{Si a <R+M-1y b + c} \ge \text{N + 1} \end{cases}$$

$$E\,[N_{abc}] = \lambda^{*}_{abc}\quad E[T_{abc}] \tag{4}$$

Substituting these values in the system of equations (3), calling $p_R = \lambda_R / \lambda_N + \lambda_R$, $p_N = \lambda_N / \lambda_N + \lambda_R$ dividing all the terms by the average duration of the cycle, E [T], and taking into account (1) and (2), we arrive at:

if $1 \le a \le R+M-1,\ 1 < c \le M,\ 0 \le b \le N$:

$$\pi_{abc} = p_R\,\lambda^{*}_{abc}\,\pi_{000}\,A_{(1,0,0)(a,b,c)} + \sum_{i=c-1}^{a}\sum_{j=0}^{b}\lambda^{*}_{abc}\,\pi_{abc}\,A_{(i,0,0)(a,b,c)} + \sum_{i=1}^{a}\lambda^{*}_{abc}\,\pi_{i00}\,A_{(i,0,0)(a,b,c)}$$

if $0 \le a \le R+M-1,\ 1 < c \le M,\ 0 \le b \le N$:

$$\pi_{ab1} = p_N\,\lambda^{*}_{abc}\,\pi_{000}\,A_{(0,0,1)(a,b,1)} + \sum_{i=0}^{a}\sum_{j=0}^{b}\lambda^{*}_{abc}\,\pi_{ij1}\,A_{(i,j,1)(a,b,1)}$$

(5)

if $0 \le a \le R+M-1$,

$$\pi_{a00} = p_R\,\lambda^{*}_{abc}\,\pi_{000}\,A_{(1,0,0)(a,0,0)} + \sum_{i=0}^{a}\lambda^{*}_{abc}\,\pi_{i00}\,A_{(i,0,0)(a,0,0)}$$

If in this system all equations are divided by π_{000}, calling $\theta_{abc} = \pi_{abc} / \pi_{000}$, the values π_{abc} can be easily obtained recursively. The probabilities π_{abc} are then easily obtained simply by observing that: from where:

$$\pi_{000} + \sum_{(a,b,c)\neq(0,0,0)} \pi_{abc} = 1 \Rightarrow \sum_{(a,b,c)\neq(0,0,0)} \theta_{abc} = \frac{1}{\pi_{000}}$$

$$\pi_{000} = \frac{1}{1+\sum_{(a,b,c)\neq(0,0,0)} \theta_{abc}}$$

$$\pi_{000} = \frac{\theta_{abc}}{1+\sum_{(a,b,c)\neq(0,0,0)} \theta_{abc}} \tag{6}$$

Obviously, the terms $A_{(i,j,k)(a,b,c)}$ have yet to be determined.

Let's start by calculating $A_{(i,j,c)(a,b,c)}$ for $1 \leq c \leq M$, $0 \leq i \leq a$, $0 \leq j \leq b$. For this we define the random variable:

$$\chi_{(i,j,l)(a,b,c)}\ \chi_{(i,j,l)(a,b,c)}(t) = \begin{cases} 1 & \text{if at time t the system is in state (a, b, c), and a service that started at time 0 with the system in state (I, j, c) has not finished yet, otherwise} \\ 0 & \end{cases}$$

However

It is evident then that:

$$A_{(i,j,l)(a,b,c)} = E\left[\int_0^\infty \chi_{(i,j,c)(a,b,c)}(t)dt\right] = \int_0^\infty E[\chi_{(i,j,c)(a,b,c)}(t)]dt \tag{7}$$

$$E[\chi_t] = P(\chi_t = 1) = (1B(t))e^{-\lambda_R t}\left(\frac{\lambda_R t}{(a-i)!}\right)e^{-\lambda_R t}\frac{(\lambda_N t)^{b-i}}{(b-1)!} \tag{8}$$

where B (t) is the service time distribution function. In the particular case of deterministic service of duration D, we have:

$$B(t) = \begin{cases} 0 & t < D \\ 1 & t \geq D \end{cases} \tag{9}$$

Then in this case:

$$A_{(i,j,c)(a,b,c)} = \int_0^D e^{-\lambda_R t}\left(\frac{\lambda_R t}{(a-i)!}\right)e^{-\lambda_R t}\frac{(\lambda_N t)^{b-i}}{(b-1)!}dt \tag{10}$$

$$G(m,n) = \int_0^D e^{-\lambda_R t}\frac{(\lambda_R t)^m}{m!}e^{-\lambda_R t}\frac{(\lambda_N t)^n}{n!}dt \tag{11}$$

$$F(m,n) = e^{-(\lambda_R+\lambda_N)D}\ \frac{\lambda_R^m \lambda_N^n}{m!n!}\ \frac{D^{m+n}}{\lambda_R+\lambda_N}$$

If we call:

you can easily check the recursive relationship:

$$G(m,n) = \frac{\lambda_R}{\lambda_R+\lambda_n} G(m-1,n) + \frac{\lambda_N}{\lambda_R+\lambda_n} G(m,n-1) - F(m,n), m \geq 1, n \geq 1 \quad (12)$$

$$G(m,0) = \frac{1}{\lambda_R+\lambda_N}\left[\left(\frac{\lambda_R}{\lambda_R+\lambda_N}\right)^m - e^{-(\lambda_R+\lambda_n)D} \sum_{n=0}^{n}\left(\frac{\lambda_R}{\lambda_R+\lambda_N}\right)^h \frac{(\lambda_R\ D)^{m-h}}{(m-h)!}\right] m \geq 0 \quad (13)$$

$$G(0,n) = \frac{1}{\lambda_R+\lambda_N}\left[\left(\frac{\lambda_R}{\lambda_R+\lambda_N}\right)^n - e^{-(\lambda_R+\lambda_n)D} \sum_{n=0}^{n}\left(\frac{\lambda_R}{\lambda_R+\lambda_N}\right)^h \frac{(\lambda_R\ D)^{m-h}}{(m-h)!}\right] n \geq 0 \quad (14)$$

with initial values:

Thus, $A_{(i,j,c)(a,b,c)} =$ G(a-i,b-j),1≤ c ≤M, 0≤ i≤ a, 0≤j ≤b (15)

and its value can be calculated using recursively (12) with initial values (13) and (14).To now calculate A (i, 0,0) (a, b, c), with 1≤c<M we must note that for a period of service to go from state (i, 0,0) to (a, b, c), they must have reached i TR clients and b + 1 NRT clients, provided that the first NRT client has arrived when there are exactly c-1 RT clients in the system. Using the property of the Poisson process that if in a period (0, t) there are n occurrences of the same, they are uniformly distributed in the interval, the probability that the first arrival of a NRT client is the one that occupies the position c in the shared queue having reached RT clients and b + 1 NRT clients at (0, t), is given by:

$$\frac{(a-i)!\ (a+b+1-c)!}{(a-c+1)!(a+b+1-i)!}(b+1) \quad (16)$$

Now proceeding as for the calculation of $A_{(i,j,c)(a,b,c)}$, we arrive at:

$$A_{(i,0,0)(a,b,c)} = \frac{(a-i)!\ (a+b+1-c)!}{(a-c+1)!(a+b+1-i)!}(b+1)\ G\ (a-i,b+1) \quad (17)$$

A similar reasoning, taking into account now that for a NRT client to occupy the M position, it must have arrived when there are M-1 or more RT clients in the system, allows us to obtain:

$$A_{(i,0,0)(a,b,c)} = \frac{(a-i)!\,(a+b+2-M)!}{(a-M+1)!(a+b+1-i)!} \; G\,(a\text{-}i,\, b\text{+}1) \qquad (18)$$

Finally, also in a very simple way it follows that:

$$A_{(i,\,0,0)\,(a,\,0,0)} = G\,(a - i,\, 0) \qquad (19)$$

In this way, equations (15), (17), (18) and (19) together with (12) provide us with the precise recursive scheme to finally obtain the probabilities in equilibrium πijk from (5) and (6).

4. PROBABILITIES OF LOSS

A RT client is rejected by the system when there is no buffer space to host it upon arrival; this occurs when in total in the system there are R + M-1 hundred RT. Therefore, the probability of loss in the steady state for RT customers is given by:

$$\gamma_{TR} = \pi_{\,R+M-1,0,0} + \sum_{c=1}^{M} \sum_{b=0}^{N} \pi_{R+M-1,b,c} \qquad (20)$$

Given that during a cycle they try to access the system on average λRE [T] RT clients, if we call the number of RT clients rejected per cycle LTR (T), we have, making use of (2):

$$E[L_{TR}(T)] = \gamma_{TR}\,\chi_R E[T] = \frac{\gamma_{TR}\,\chi_N}{(\lambda_R+\lambda_N)\pi_{000}} \qquad (21)$$

Likewise, the probability of loss at steady state for NRT clients is given by:

$$\gamma_{TNR} = \sum_{a=0}^{R+M-1} \sum_{c=1}^{M} \pi_{aNc} \qquad (22)$$

and the average number of NRT clients rejected per cycle is:

$$E[L_{TR}(T)] = \gamma_{TNR}\chi_R E[T]= \frac{\gamma_{TNR}\chi_N}{(\lambda_R+\lambda_N)\pi_{000}} \quad (23)$$

5. WAITING TIMES

Let's locate ourselves in our system at the precise instant in which a service has just been completed (a cell is transmitted), which has left the system in state (a, b, c), and let ω_{ibc}, with $i \leq a$, be the time that the RT client in front of i - 1 clients still has to wait in queue. Obviously, if si, $i \leq M - 1$:

$$\omega_{ibc} = \begin{cases} \sum_{j=1}^{i-1} X_j \ si\ i < c\ \grave{O}\ c = b = 0 \\ \sum_{j=1}^{i} X_j \ si\ \ i \geq c \geq 1 \end{cases} \quad (24)$$

where Xj is the duration of a service time. Now, if i ≥ M, the waiting time for this client will not be affected by new RT clients arriving from now on, but it could be affected by NRT clients arriving from this moment. Indeed, there is the possibility that during the time it takes for our RT client to reach the shared buffer, by virtue of the management policy used, NRT clients that arrived after him but found little or no queue could access this buffer in their specific buffer and were able to enter the shared buffer soon[ix]. We can then establish the following recurrence relationships for the waiting time of our RT client:

• If c = 0:

$$\omega_{i00} = \begin{cases} X + \omega_{i-1,0,0} \ \text{ con prob. } \beta_0 \\ X + \omega_{i-1,0,0} \ \text{ con prob. } \beta_{j+1}, 0 \leq j < N \\ X + \omega_{i-1,0,0} \ \text{ con prob. } \beta_{N+1}^{*} \end{cases} \quad (25)$$

where X is the duration of a service period and Y β_k is the probability that k NRT clients will reach the system during a service period. On the other handβ_k^*, k or more NRT clients arrive.

- If c = 1:

$$\omega_{i01} = \begin{Bmatrix} X + \omega_{i,0,0} \text{ con prob. } \beta_0 \\ X + \omega_{i,j,M} \text{ con prob. } \beta_{j+1}, 0 \leq j < N-1 \\ X + \omega_{i,N-1,M} \text{ con prob. } \beta_N^* \end{Bmatrix} \tag{26}$$

$$\omega_{ib1} = \begin{Bmatrix} X + \omega_{i,b-1+j,M} \text{ con prob. } \beta_j, 0 \leq j < N-b, 0 < b < N \\ X + \omega_{i,N-1,M} \text{ con prob. } \beta_{N-b}^* \end{Bmatrix} \tag{27}$$

$$\omega_{k,N,1} = X + \omega_{k,N-1,M} \tag{28}$$

- If 1< $c \leq M$:

$$\omega_{ibc} = \begin{Bmatrix} X + \omega_{i-1,b+j,c-1} \text{ con prob. } \beta_j, 0 \leq j < N-b, 0 \leq b < N \\ X + \omega_{i,N-1,M} \text{ con prob. } \beta_{N-b}^* \end{Bmatrix} \tag{29}$$

$$\omega_{i,N,c} = X + \omega_{i-1,N,c-1} \tag{30}$$

From these equations the following recurrence relationships can be obtained for the mean waiting times:

- If i ≤ M-1

$$\mathrm{E}[\omega_{ibc}] = \begin{Bmatrix} (i-1)\mathrm{E}[X] \text{ If } i < c\grave{O}c = b = 0 \\ i\mathrm{E}[x] \text{ if } i \geq c \geq 1 \end{Bmatrix} \tag{31}$$

- If i ≥ M:

$E[\omega_{i00}] = E[X] + \beta_0 E[\omega_{i-1,0,0}] + \sum_{j=0}^{N-1} \beta_{j+1} E[\omega_{i-1,j,M-1}]$
$+\beta_{N+1}^{*} E[\omega_{i-1,N,M-1}]$

$E[\omega_{i01}] = E[X] + \beta_0 E[\omega_{i,0,0}] + \sum_{j=0}^{N-2} \beta_{j+1} E[\omega_{i,j,M}] + \beta_N^{*} E[\omega_{i,N-1,M}]$

$E[\omega_{ib1}] = E[X] + \sum_{j=0}^{N-b} \beta_j E[\omega_{i,b-1+j,M}] + \beta_{N-b}^{*} E[\omega_{i,N-1,M}],$
$0< b < N$ (32)

$E[\omega_{iN1}] = E[X] + +\beta_{N-b}^{*} E[\omega_{i,N-1,M}]$

$E[\omega_{ibc}] = E[X] + \sum_{j=0}^{N-b} \beta_j E[\omega_{i-1,b+j,c-1}] + \beta_{N-b}^{*} E[\omega_{i-1,N,c-1}],$
$0< b < N$

$E[\omega_{iNc}] = E[X] + E[\omega_{i-1,N,c-1}],$

Equations (32) together with the initial values given by (31) allow us to recursively obtain the expectations E $[\omega_{ijk}]$ for all possible states (i, j, k). The resolution of these equations will be more or less difficult depending on the probability distribution function of the service time. In the particular case of deterministic service time of duration D, we have:

$$E[X] = D \qquad (33)$$

$\beta_k = e^{-\lambda_N} D \frac{(\lambda_N D)^k}{k!}, \beta_N^{*} = 1 - \sum_{j=0}^{k-1} e^{-\lambda_N} D \frac{(\lambda_N D)^j}{j!}$

Now, the expectations obtained in (32) correspond to the average waiting times in queue measured from the moment a service ends. The global waiting time of a RT client that upon arrival finds the system in state (a, b, c) is:

- If a <M - 1:

$$W_{abc} = \begin{cases} X_{RES} + \sum_{i=1}^{a-1} X_i \ , If\ b = c = 0 \\ X_{RES} + \sum_{i=1}^{a} X_i \ , If\ c > 1 \end{cases} \tag{34}$$

where X_{RES} is the residual service time remaining for the end of the service of the client occupying the head of the shared queue upon arrival of the client RT

• If a ≥M - 1:

If new RT clients arrive during the residual service time of the client occupying the head of the queue, these do not affect the waiting time of the RT client that has just arrived. However, for the same reason noted above, they do affect NRT clients arriving during this time. If we call α_k the probability that during the residual service time after the arrival of the RT client k clients NRT will arrive, and y α_k^* the probability that k or more will arrive, we have:

$$W_{abc} = \begin{cases} X_{RES} + \omega_{a-1,j,M} & con\ prob.\alpha_{j+1,}\ \ 0 \le j \le N & (b = c = 0) \\ X_{RES} + \omega_{a-1,N,M} & con\ prob.\alpha_{N+1}^* & (b = c = 0) \\ X_{RES} + \omega_{a,0,0} & con\ prob.\alpha_{0,}\ \ 0 \le j \le N & (c = 1, b = 0) \\ X_{RES} + \omega_{a,j,M} & con\ prob.\alpha_{j+1,}\ \ 0 \le j \le N-2 & (c = 1, b = 0) \\ X_{RES} + \omega_{a,N-1,M} & con\ prob.\alpha_{N}^* & (c = 1, b = 0) \\ X_{RES} + \omega_{a,b+j-1,M} & con\ prob.\alpha_{j,}\ \ 0 \le j \le N-b & (c = 1, b > 0) \\ X_{RES} + \omega_{a,N,M} & con\ prob.\alpha_{N-b}^* & (c = 1, b > 0) \\ X_{RES} + \omega_{a-1,b+j,c-1} & con\ prob.\alpha_{j,}\ \ 0 \le j \le N-b & (c > 1) \\ X_{RES} + \omega_{a-1,N,c-1} & con\ prob\alpha_{N-b}^* & (c > 1) \\ X_{RES} + \omega_{a-1,0,0} & con\ prob.\alpha_{0,} & (b = c = 0) \end{cases} \tag{35}$$

In a similar way to what we did previously for the esperaijk waiting times, we can now find from (35) the average waiting time in queue for a RT client that upon arrival finds the system in the state (a, b, c):

$E[\, W_{a00}] = E[X_{RES}] + \alpha_0$ E$[\omega_{a-1,0,0}$]+$\sum_{j=0}^{N} \alpha_{j+1}E[\omega_{a-1,j,M}]$+$\alpha^*_{N+1}$E $[\omega_{a-1,N,M}]$

$E[\, W_{a01}] = E[X_{RES}] + \alpha_0$ E$[\omega_{a,0,0}$]+$\sum_{j=0}^{N} \alpha_{j+1}E[\omega_{a,j,M}]$+$\alpha^*_{N+1}$E $[\omega_{a,N-1,M}]$

$E[\, W_{abc}] = E[X_{RES}]$+$\sum_{j=0}^{N-b} \alpha_jE[\omega_{a-1,b+j,c-1}]$+$\alpha^*_{N-b}E[\omega_{a-1,N,c-1}]$,0≤ b< N, c >1. (36)

These expectations can be calculated recursively using as initial values the expectations obtained directly from (34):

$E[\, W_{a00}] = E[X_{RES}]$+(a-1)E(X),1≤ a< M-1. (37)

$E[\, W_{abc}] = E[X_{RES}]$+a E(X), if c >1

and the obvious condition: E [W_{000}] = 0.

As already happened with the mean values E [ωijk], obtaining the expectations in (36) depends on the difficulty of calculating the probabilities αk. In the particular case of deterministic service of duration D, it can be proved that:

$$a_{k}=\int_0^D e^{-\lambda_N}(D-t)\,\frac{[\lambda_N(D-t)]^k}{k!}\,\frac{1}{D}\,dt = \frac{1}{\lambda_N}[1-\frac{1}{D\,\lambda_N}(k+1-e^{-\lambda_N}\,\mathrm{D} \sum_{j=0}^{k}(j+1)\frac{(\lambda_N\ D)^{k-j}}{(k-j)!} \quad (38)$$

In this case, in addition E [XRES] = D / 2.

Let us now denote by W^{TR} (respectively, W^{NRT}) the random variable that measures the waiting time of an arbitrary RT (respectively, NRT) customer, once the system has reached equilibrium. Let us also call N_{TR} (respectively, N_{NRT}) the number of clients RT (respectively, NRT) present in the system when it is in equilibrium.

We are able to calculate the average waiting time of an arbitrary RT client. If we condition by the state of the system upon the arrival of this client, we have:

$$E[W^{TR}] = \sum_{(a,b,c)} E[W_{abc}]\, \pi_{abc} \qquad (39)$$

The value of this expectation is calculated using (36) and (37), with the stationary probabilities found in (5) and (6).

Finally, the mean wait time for NRT clients can be found using little's formula. In this particular system, Little's formula takes the form:

$$E[N_{TR}] + E[N_{NRT}] = \lambda_R (1-\gamma_{TR})\, E[W^{TR}] + \lambda_N (1-\gamma_{NRT})\, E[W^{NRT}] \qquad (40)$$

that is, the total average number of clients in the system at equilibrium must be equal to the sum of the effective arrival rates of each class of clients times their respective average waiting times. The mean numbers $E[N_{RT}]$ and $E[N_{NRT}]$ can be easily calculated from the probabilities π_{ijk} :

$$E[N_{TR}] = \sum_{a=0}^{R+M-1} a\, \pi_{a00} + \sum_{a=0}^{R+M-1} a \sum_{b=0}^{N} \sum_{c=1}^{M} \pi_{abc} \qquad (41)$$

$$E[N_{NRT}] = \sum_{a=0}^{R+M-1} \sum_{b=0}^{N} \sum_{c=1}^{M} (b+c)\, \pi_{abc}$$

and from (40) is easily solved:

$$E[W^{TNR}] = \frac{E[N_R] + E[N_D] - \lambda_R (1-\gamma_{TR})\, E[W^{TR}]}{\lambda_N (1-\gamma_{TNR})} \qquad (42)$$

6. SYSTEM PERFORMANCE OPTIMIZATION

As mentioned in section 1, the values of M, N and R should be chosen in such a way as to minimize the average cost per unit of time in the long term that the delays for RT clients and losses for NRTs entail:

$$\phi[M,N,R]=\frac{\lim_{t\to\infty} \alpha E\,[YRT\ (t)] + \beta E\,[LNRT)\ t)]}{t}$$

This minimum must be found with the restriction that E $[W^{NRT}]$ $<W_1$ and $\gamma_{TR}<g_1$. Since the process (N_{TR} (t), N_{NRT} (t), S_{NRT} (t)) is regenerative, the renewal theory allows us to calculate the previous limit of the form:

$$\phi[M,N,R,] = \frac{\alpha E\ [Y_{RT}\ \ (T)] +\ \beta E\ [L_{TNR}(T)]}{E[T]}\ (43)$$

that is, the average cost for the long-term system coincides with the average cost during a renewal period. Equation (2) allows us to calculate E [T], and equation (23) gives us E [L_{NRT} (T)]. We only need to determine E [Y_{RT} (T)], the accumulated waiting time for all RT clients served during a renewal period. This quantity is easily found as:

$$E\ [Y_{RT}\ (T) = \lambda_R\ (1-\gamma_{TR})\ E\ [T]\ E\ [W^{RT}]\ \ (44)$$

that is, the average number of RT clients that arrive during a renewal period, multiplied by the average waiting time of each one, already obtained in (39). Since all the terms that appear in (43) have been calculated numerically, we cannot find an explicit formula to obtain the values of M, N and R that minimize (43), and these have to be found by a search algorithm that goes successively increasing and / or decreasing the values of these parameters, also fulfilling the restrictions E [W_{NRT}] $<W_1$ and $\gamma_{TR}<g_1$. It should be noted that the recursive way of computing all the elements necessary for the calculation of the functions involved in (43) greatly simplifies the

necessary calculations in the search algorithm, since the terms obtained in one iteration can be reused in the next without need to recalculate them.[x]

7. CONCLUSIONS, CURRENT STATUS AND FUTURE DEVELOPMENTS

In this work we have analysed the performance of a queue management policy on a network switch as shown in Figure 1, which receives two classes of traffic with different quality requirements in terms of delays and losses. We have seen how, recursively, it is possible to numerically calculate the state probabilities of this system, and from them we have shown how to obtain the average delays of both types of traffic, as well as their loss rates. All these quantities are involved in the calculation of system performance through an objective function that represents a weighting of the cost caused by delays in priority traffic (in real time) and losses caused by non-priority traffic (in non-real time). The values of the optimal control parameters of the system are finally obtained by applying a suitable algorithm that minimizes the objective function. At the present time we are working on the development of an efficient algorithm suitable for this purpose, as well as on the search for approximate methods for calculating the terms of the objective function, which allow obtaining approximations of it without requiring as many calculations as those presented in this work. This task is particularly interesting when the predictable values of M, N, or R are large[xi].

Future extensions of this work include:

(1) Comparison of the management policy studied here with a greater variety of policies currently implemented in queuing systems with similar characteristics, evaluating the pros and cons of each of them and establishing a policy classification that allows choosing the best one in each particular case.
(2) Although in this work we have considered the case of real-time and non-real-time traffic, the policy studied is efficient to distribute the system resources between two real-time traffic and two non-real-time traffic. Interesting is the generalization of politics here shown to switches that receive more than two classes of traffic, and include the possibility that in the shared buffer there may be simultaneously two or more NRT clients.[xii]
(3) We have considered arrivals to the system according to a Poisson process. An open problem is the analysis of non-Poisson arrivals. The most immediate generalization of this model is to arrivals with phase-type distributions or MAP's (Markovian arrival processes).
(4) Finally, let us point out that the study of this management policy in discrete-time systems is also of interest, such as those used to model the transmission of cells in ATM.

References

1- Anusha, V., Parthasarathy, V., Prithvi, B., & Bellamkonda, A. P. (2015). An efficient and novel buffer clustering technique for minimal data loss, load balancing and effective queue management in multi core processors. Advances in Natural and Applied Sciences, 9(7), 100-109.

2- Chauhan, N., Banka, H., & Agrawal, R. (2021). Delay-aware application offloading in fog environment using multi-class Brownian model. Wireless Networks, 27(7), 4479-4495.

3- Ghosh, A. P., & Weerasinghe, A. P. (2010). Optimal buffer size and dynamic rate control for a queuing system with impatient customers in heavy traffic. Stochastic Processes and Their Applications, 120(11), 2103-2141.

4- Klimenok, V., Dudin, A., Vishnevskiy, V., Shumchenya, V., & Krishnamoorthy, A. (2016, November). Performance measures and optimization of queueing system with reserve server. In International Conference on Distributed Computer and Communication Networks (pp. 74-88). Springer, Cham.

5- Knightly, E. W., Wrege, D. E., Liebeherr, J., & Zhang, H. (1995, May). Fundamental limits and tradeoffs of providing deterministic guarantees to VBR video traffic. In Proceedings of the 1995 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems (pp. 98-107).

6- Liubogoshchev, M., Ragimova, K., Lyakhov, A., Tang, S., & Khorov, E. (2021). Adaptive Cloud-Based Extended Reality: Modeling and Optimization. IEEE Access, 9, 35287-35299.

7- Le, L., Jeffay, K., & Smith, F. D. (2006, July). A loss and queuing-delay controller for router buffer management. In 26th IEEE International Conference on Distributed Computing Systems (ICDCS'06) (pp. 4-4). IEEE.

8- Mehta, R. (2022). Discrete-time simulation for performance modelling of FIFO single-server queuing system. International Journal of Systems, Control and Communications, 13(2), 112-132.
9- Sivaraj, R., Pande, A., Zeng, K., Govindan, K., & Mohapatra, P. (2012, June). Edge-prioritized channel-and traffic-aware uplink carrier aggregation in LTE-advanced systems. In 2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM) (pp. 1-9). IEEE.

10- Sugirtham, N., & Jenny, R. S. (2021). Impact of Queuing Disciplines on the Performance of Multi-Class Traffic in a Network. Information Technology In Industry, 9(1), 698-703.
11- TIJMS, H.C. (1986): “Stochastic Modelling and Analysis: A computational approach”, John Wiley & Sons, Inc.605 Third Ave. New York, NYUnited States
12- Wynn, D. C., & Clarkson, P. J. (2018). Process models in design and development. Research in engineering design, 29(2),161-202.
13- Yang, S., Stol, N., Chi, H., & Li, Q. (2016). Optimized design of delay-line buffers with an input-feedback mechanism for asynchronous optical packet switching networks. Applied Optics, 55(31), 8705-8712.
14- Yashar Ganjali Gavgani, March 2007, “Buffer Sizing in Internet Routers” Ph.D dissertation Stanford University, California.
15- Yerima, S. Y., & Al-Begain, K. (2011). Novel radio link buffer management schemes for end-user multi-class traffic in high speed packet access networks. Wireless Personal Communications, 61(2), 349-382.

16- Zhao, M., Ang, C. W., Zhao, B., & Ng, W. S. (2019, October). A Tradeoff Study of Real-time Traffic Prediction Approaches for Intelligent Transportation System. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp. 1975-1980). IEEE.