JOURNAL OF INDUSTRIAL AND
MANAGEMENT OPTIMIZATION
Volume **10**, Number **1**, January **2014**

doi:10.3934/jimo.2014.10.193

pp. **193–206**

A CONTINUOUS-TIME QUEUEING MODEL WITH CLASS CLUSTERING AND GLOBAL FCFS SERVICE DISCIPLINE

Willem Mélange, Herwig Bruneel, Bart Steyaert
Dieter Claeys and Joris Walraevens

Department of Telecommunications and Information Processing
Ghent University - UGent
Sint-Pietersnieuwstraat 41, 9000 Ghent, Belgium

Abstract. In this paper the focus is on "class clustering" in a continuous-time queueing model with two classes and dedicated servers. "Class clustering" means that customers of any given type may (or may not) have a tendency to "arrive back-to-back". We believe this is a concept that is often neglected in literature and we want to show that it can have a considerable impact on multiclass queueing systems, especially on the system considered in this paper. This system adopts a "global FCFS" service discipline, i.e., all arriving customers are accommodated in one single FCFS queue, regardless of their types. The major aim of our paper is to quantify the intuitively expected (due to the service discipline) negative impact of "class clustering" on the performance measures of our system. The motivation of our work are systems where this kind of inherent blocking is encountered, such as input-queueing network switches, road splits or security checks at airports.

1. **Introduction.** In general, queueing phenomena occur when some kind of customers, desiring to receive some kind of service, compete for the use of a service facility (containing one or multiple servers) able to deliver the required service. Most queueing models assume that a service facility delivers exactly one type of service and that all customers requiring this type of service are accommodated in one common queue. If more than one service is needed, multiple different service facilities can be provided, i.e., one service facility for each type of service, and individual separate queues are formed in front of these service facilities. In all such models, customers are only hindered by customers that require exactly the same kind of service, i.e., that compete for the same resources.

In some applications, it may not be physically feasible or desirable to provide separate queues for each type of service that customers may require, and it may be necessary or desirable to accommodate different types of customers (i.e., customers requiring different types of service) in the same queue. In such cases, customers of one type (i.e., requiring a given type of service) may also be hindered by customers of other types. For instance, if a road or a highway is split in two or more subroads leading to different destinations, cars on that road heading for destination A may be hindered or even blocked by cars heading for destination B, even when the subroad leading to destination A is free, simply because cars that go to B are in

2010 *Mathematics Subject Classification.* Primary: 60K25, 90B20; Secondary: 60J28.
Key words and phrases. Queueing theory, class clustering, global FCFS, continuous time.
The reviewing process of the paper was handled by Wuyi Yue and Yutaka Takahashi as Guest Editors.

front of them. In other words, there is a first-come-first-served (FCFS) order on the main road. This blocking also takes place in weaving sections on highways albeit to a lesser extent [14, 15]. We refer to [17, 18] for a general overview and validation of modelling traffic flows with queueing models. Similarly, in switching nodes of telecommunication networks, information packets with a given destination of node A may have to wait for the transmission of packets destined to node B that arrived earlier, even when the link to node A is free, if the arriving packets are accommodated in so-called input queues according to the source from which they originate (the well-known HOL-blocking effect, see [9, 11, 10, 16, 3]). Analogously, at a security checkpoint (e.g., at an international airport or train station) people are usually body-searched by someone of the same gender. As a result, when a group of friends of the same gender arrive, the people of the opposite gender behind them may have to wait until the whole group has been checked, even when the other security person is available, at least when it is not allowed to overtake at the security checkpoint (which is often the case for security reasons). In general, these applications can be modelled by a queueing system with different types of traffic, servers which are dedicated to these different classes, and a FCFS scheduling in the shared queue. Therefore, customers of one type can be blocked by customers of the other type that are waiting in front of them, even when their server is available. We will refer to this scheduling as "global FCFS" in the remainder.

In [12], we already got some insight in the impact of this kind of phenomenon on the performance of the involved systems. In this paper, we shift focus to the effect of class clustering, i.e., to the way customers of any given type have a tendency to "arrive back-to-back". Class clustering is a concept that often is neglected in literature to keep the model as simple as possible, but in this paper we want to demonstrate that it is not always possible to treat this concept negligently. It may be already intuitively clear that the tendency of customers to arrive back-to-back will have some impact. Especially when looking at the extremes. When customers have a great tendency to arrive back-to-back, we will have long periods with only one type of customer and thus long periods when only one of the two servers is working. On the other hand, when customers have a tendency to not arrive back-to-back and the customers thus arrive with alternating types, a lot of the time both servers will be able to work. In the first case there will be more blocking than in the second. It is this effect that we quantify in this paper. We model the basic elements of the problems at hand to be able to determine the combined impact of class clustering and global FCFS. We therefore propose a continuous-time queueing model that is still rich enough to capture the essential aspects of the problem at hand but simple enough to determine explicit closed-form formulas for not only the distribution of the system occupancy, but also for the distribution of the system delay and for the stability condition.

The structure of the rest of the paper is as follows: in section 2, we start by giving a brief description of the mathematical model. In section 3, we first discuss the stability condition of our system. It is clear that only when the stability condition is met, our analysis is justified. In section 4, we start with focus on the system occupancy, i.e., the number of customers in the system. The distribution of the system occupancy is determined and some related performance measures are calculated. Next in section 5, the focus is shifted to the system delay. Its distribution is determined and some related performance measures are calculated. Section 6 is devoted to a discussion of the results derived in previous sections and some

numerical examples are provided. Some conclusions and directions for future work are given in section 7.

2. **Mathematical model.** We consider a continuous-time queueing model with infinite waiting room. There are two servers working at rate μ (exponential service times) and two types (classes) of customers. Each of the two servers is dedicated to a given class of customers. In this case, server A always serves customers of type 1 and server B always serves customers of type 2. The customers are served in their order of arrival, regardless of the class they belong to (global FCFS).

The customers enter the system according to a Poisson arrival process with mean arrival rate λ. In this paper, the major aim is to estimate the impact of the degree of class clustering in the arrival process on this two-class two-server system. To explicitly model this, we assume a first-order Markovian type of correlation between the types of two consecutively arriving customers, which basically means that the probability that the next customer belongs to a given class depends on the class of the previous customer. We denote by α the probability that the next customer has *the same type as the previous one*, and by $1-\alpha$ the probability that the next customer belongs to *the opposite type as the previous one*. The parameter α can then be considered as a measure of the degree of class clustering in the arrival process, and will therefore be referred to as the "cluster parameter" in the sequel. It is easily seen that the size of a cluster of customers of the same type, i.e., the number of consecutive customers of any given type between two customers of the opposite type, is geometrically distributed with parameter α and mean value $1/(1-\alpha)$. From a conceptual point of view, the only price we pay with this choice is that we can only study cases where both classes of customers are equiprobable and thus both types of customers account for half of the total load of the system. This is for instance an accurate modelling for (i) uniform input-queueing switches, (ii) road splits with equally balanced traffic in both directions and (iii) security checks at airports in periods when there are a lot of travellers with leisure as purpose of travelling [4], leading to about a $50\% - 50\%$ ratio of males and females.

3. **Stability condition.** We start the section with introducing the average amount of work that enters the system per unit time:

$$\gamma \triangleq \frac{\lambda}{\mu}.$$

The stability condition can then be expressed as

$$\gamma < t_S + 2t_D, \tag{1}$$

where t_S represents the fraction of time when one server is working and t_D is the fraction of time when both servers are working. Indeed, the system is stable when the average amount of work per unit time that enters the system (γ) is smaller than the average amount of work the system can execute per unit time, i.e., the average amount of work the system would execute per unit time when it would be constantly provided with new customers. When only one server is able to work, only one time unit work per unit time can be executed. However when both servers can work, two time units work per unit time can be executed, thus explaining (1). We can also rewrite (1) as

$$\gamma < 1 + t_D, \tag{2}$$

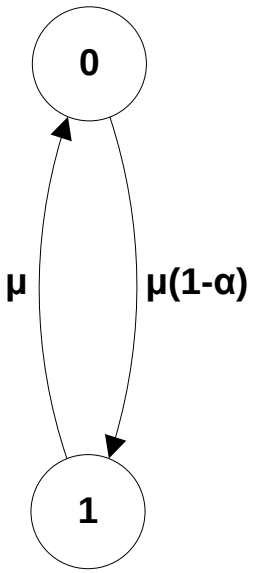

FIGURE 1. Two-state Markov chain to determine the stability condition

which we can interpret as follows: there will always be at least one server working when the system is constantly provided with new customers and only a fraction of time a second server is working. To determine the fractions of time t_S and t_D, we conceive that the number of working servers forms a simple two-state Markov chain (see Fig. 1). When looking at the stability condition, we assume that the system is constantly provided with new customers and thus independent of the number of customers in our system. The first state in our Markov chain is when only one server is working or, in other words, when the first two customers in our system are of the same type (state 0). The second state is when both servers are working, i.e. when the first two customers in our system are not of the same type (state 1). The rate to go from state 0 to state 1 is $(1-\alpha)\mu$; namely a rate μ to finish the ongoing service multiplied with the probability $1-\alpha$ that the next two first customers of our system are of opposite types (or the probability that both servers will be able to work). The rate to go from state 1 to state 0 equals μ; namely a rate 2μ to finish one of the services multiplied with the probability $\frac{1}{2}$ that the next customer to be served is of a different type of the departed customer (or the probability that only one server is able to work). The time t_S is then the fraction of time the Markov chain sojourns in state 0 and is given by

$$t_S = \frac{1}{2-\alpha}. \tag{3}$$

Similarly, the time t_D is the fraction of time the Markov chain sojourns in state 1 and is given by

$$t_D = \frac{1-\alpha}{2-\alpha}. \tag{4}$$

Equations (1) and (2) lead to

$$\gamma < \frac{3-2\alpha}{2-\alpha}. \tag{5}$$

We assume this stability condition to be fulfilled in the remainder of the paper.

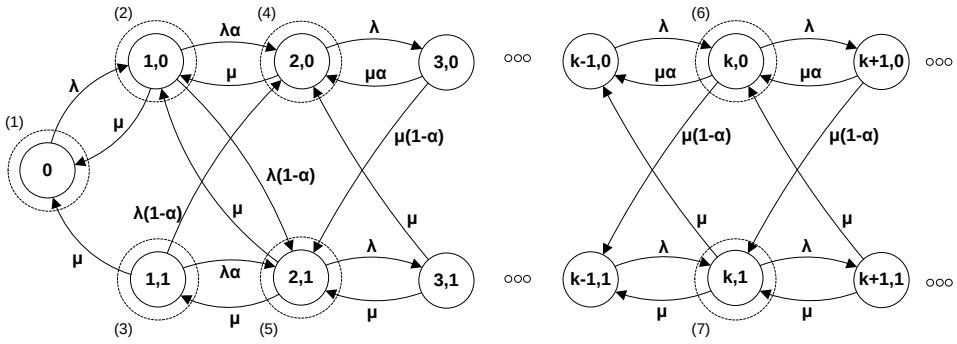

FIGURE 2. The state diagram of our system

4. System occupancy.

4.1. **State diagram, balance and boundary equations.** This system can be described by a continuous-time Markov process. We describe the state of the system by the pair (n, m) where n represents the number of customers in the system and m indicates whether the two customers at the front of the system are of the same type ($m = 0$) or not ($m = 1$). For example, the state $(n, 0)$ means that the types of the first two customers are of the same type and a total of n customers resides in the system. This is thus a Quasi-birth-and-Death (QBD) process with two phases (m) and where the levels represent the number of customers in the system. Note, we are only interested in the (difference of) types of the first two customers. We stress that the exact type of the first two customers is of no importance, only whether they are different. The first reason for this is that the cluster parameter α denotes the probability that the next customer is of the same type as the previous customer, regardless of the type of this customer. Secondly, the service times of both types of customers have the same distribution, namely an exponential one with mean $\frac{1}{\mu}$. When we have only one customer in the system ($n = 1$) we interpret the state differently. When the system is in the state $(1, 0)$, the customer left in the system has the same type as the last customer that arrived in the system. Similarly, $(1, 1)$ means that the customer left in the system has a different type than the last customer that arrived in the system. Indeed, the last customer may have already left the system because customers are able to overtake each other in the service units (by having a shorter service time). Notice also that it is not necessary to split state (0) in a $(0, 0)$ and a $(0, 1)$ state because we always enter state $(1, 0)$ from both states. Thus, due to our choice for modelling the system as a symmetric one (the arrival process is only determined by the total arrival rate λ and the cluster parameter α and the service times are equally distributed as well), the state diagram of Fig. 2 emerges.

Define $p(n, m)$ as the steady-state probability to be in state (n, m). Then from Fig. 2 (see transitions to and from states (6) and (7)), we find the following balance equations for $p(n, m)$, for the repeating portion of our Markov chain ($n > 2$),

$$(\lambda + \mu)p(n, 0) = \alpha\mu p(n + 1, 0) + \mu p(n + 1, 1) + \lambda p(n - 1, 0), \tag{6}$$

$$(\lambda + 2\mu)p(n, 1) = (1 - \alpha)\mu p(n + 1, 0) + \mu p(n + 1, 1) + \lambda p(n - 1, 1). \tag{7}$$

The following boundary equations can be obtained similarly (observe transitions to and from states $(1)-(5)$ in Fig. 2, resp.)

$$\lambda p(0) = \mu p(1,0) + \mu p(1,1) \tag{8}$$
$$(\lambda+\mu)p(1,0) = \mu p(2,0) + \mu p(2,1) + \lambda p(0) \tag{9}$$
$$(\lambda+\mu)p(1,1) = \mu p(2,1) \tag{10}$$
$$(\lambda+\mu)p(2,0) = \alpha\mu p(3,0) + \mu p(3,1) + \alpha\lambda p(1,0) + (1-\alpha)\lambda p(1,1) \tag{11}$$
$$(\lambda+2\mu)p(2,1) = (1-\alpha)\mu p(3,0) + \mu p(3,1) + (1-\alpha)\lambda p(1,0) + \alpha\lambda p(1,1). \tag{12}$$

with $p(0)$ the steady-state probability to be in state (0).

4.2. **Analysis of the distribution and moments of the system occupancy.** The approach to solve the balance equations is inspired on ideas from [1, 2, 5, 6] and proceeds as follows. We start by searching for basic solutions (with $n > 2$) of the balance equations (6) and (7), assuming the form

$$y(m)x^n. \tag{13}$$

With the solutions in this set we then construct a linear combination which also satisfies the boundary equations and the normalization equation. This is a valid assumption because Quasi-birth-and-death processes have a well known matrix-geometric relation. See also Section 1.6. in Neuts [13] for a discussion on results of the form above and their relationship with matrix-geometric results. Our assumption is thus equal to assuming that the associated rate matrix of the QBD is diagonalizable (what can be proved using theorems in [8]) and is given by

$$R = \begin{bmatrix} y_0(0) & y_0(1) \\ y_1(0) & y_1(1) \end{bmatrix}^{-1} \begin{bmatrix} x_0 & 0 \\ 0 & x_1 \end{bmatrix} \begin{bmatrix} y_0(0) & y_0(1) \\ y_1(0) & y_1(1) \end{bmatrix}. \tag{14}$$

Substituting (13) in (6) and (7), and dividing by x^{n-1} yields

$$(\alpha\mu x^2 - (\lambda+\mu)x + \lambda)y(0) + \mu x^2 y(1) = 0, \tag{15}$$
$$(1-\alpha)\mu x^2 y(0) + (\mu x^2 - (\lambda+2\mu)x + \lambda)y(1) = 0. \tag{16}$$

Equations (15) and (16) form a linear homogeneous system of equations for a given x. Since the system is homogeneous, x must be chosen such that the determinant of the system is zero. This determinant is given by

$$D(x) = (\alpha\mu x^2 - (\lambda+\mu)x + \lambda)(\mu x^2 - (\lambda+2\mu)x + \lambda) - (1-\alpha)\mu^2 x^4. \tag{17}$$

It is easy to see that $x = 1$ is a zero of this determinant and equation (17) can thus be rewritten as

$$D(x) = (x-1)(\mu^2(2\alpha-1)x^3 - \mu(\lambda(\alpha+1)+2\mu)x^2 + (\lambda^2+3\lambda\mu)x - \lambda^2). \tag{18}$$

We can thus find the zeroes of the determinant in closed form since we have to determine the zeroes of a polynomial of degree 3. It is also possible to prove that the determinant $D(x)$ has two solutions x_0 and x_1 in the interval (-1,1) and two outside this interval (of which one is $x = 1$). After determination of these x_j in the interval (-1,1), we can determine the $y_j(m)$ corresponding with each given x_j, i.e. determine the nontrivial solution of the homogeneous system given in (15) and (16). To find a particular solution we use the initial values $y_j(1) = 1$. Having found

basic solutions (13) of the balance equations (6) and (7), we can express the general solution as a linear combination of these basic solutions (for $n > 1$ and $m = 0, 1$),

$$p(n,m) = \sum_{j=0}^{1} C_j y_j(m) x_j^n. \tag{19}$$

To have a fully specified distribution we still need to specify five unknowns (C_0, C_1, $p(0)$, $p(1,0)$ and $p(0,1)$). To determine these unknowns, we use the boundary equations (8) to (12) and the normalizing condition,

$$p(0) + p(1,0) + p(1,1) + \sum_{n=2}^{\infty} \sum_{m=0}^{1} p(n,m) = 1. \tag{20}$$

Notice that we can drop one of the boundary equations since the boundary equations are dependent (the normalization condition replaces this boundary equation).

Knowing the probability mass function (pmf), we can derive some performance measures of practical importance using (8)-(12) and (19). For example, the mean system occupancy can be found as

$$\begin{aligned} \overline{N} &= p(1,0) + p(1,1) + \sum_{n=2}^{\infty} \sum_{m=0}^{1} np(n,m) \\ &= \sum_{j=0}^{1} \sum_{m=0}^{1} \frac{C_j y_j(m) x_j^2((m+1)\mu(1-x_j)^2 + \lambda(2-x_j))}{(1-x_j)^2\lambda}. \end{aligned} \tag{21}$$

Using Little's Law, the mean system delay equals

$$\begin{aligned} T &= \frac{\overline{N}}{\lambda} \\ &= \sum_{j=0}^{1} \sum_{m=0}^{1} \frac{C_j y_j(m) x_j^2((m+1)\mu(1-x_j)^2 + \lambda(2-x_j))}{(1-x_j)^2\lambda^2}. \end{aligned} \tag{22}$$

Finally the tail probability is given by

$$E[u > i] = \begin{cases} \sum_{j=0}^{1} \sum_{m=0}^{1} \frac{C_j y_j(m) x_j^2((m+1)\mu(1-x_j)+\lambda)}{(1-x_j)\lambda}, & i = 0 \\ \sum_{j=0}^{1} \sum_{m=0}^{1} C_j y_j(m) \frac{x_j^{i+1}}{1-x_j}, & i > 0 \end{cases} \tag{23}$$

5. System delay.

5.1. Analysis of the distribution and moments of the system delay.

Define $s_{n,m}(t)$ as the probability density function of the system delay of a customer given that the customer sees the state (n,m) on arrival. Using the PASTA property and (19) we get for the probability density function of the system delay

$$\begin{aligned} s(t) = \frac{Prob[t < S < t + dt]}{dt} &= \sum_{n,m} p(n,m) s_{n,m}(t) \\ &= p(0)s_0(t) + p(1,0)s_{1,0}(t) + p(1,1)s_{1,1}(t) \\ &\quad + \sum_{j=0}^{1} \sum_{m=0}^{1} C_j y_j(m) \sum_{n=2}^{\infty} s_{n,m}(t) x_j^n. \end{aligned} \tag{24}$$

Some reasoning in the Laplace domain yields the following recursive relations

$$s^*_{n,0}(\theta) = \frac{\mu}{\mu+\theta}\left(\alpha s^*_{n-1,0}(\theta) + (1-\alpha)s^*_{n-1,1}(\theta)\right), \tag{25}$$

$$s^*_{n,1}(\theta) = \frac{2\mu}{2\mu+\theta}\left(\frac{1}{2}s^*_{n-1,0}(\theta) + \frac{1}{2}s^*_{n-1,1}(\theta)\right) \tag{26}$$

where $s^*_{n,m}(\theta)$ are the Laplace transformation of the above defined $s_{n,m}(t)$. Equation (25) can be understood as follows: the delay of a customer arriving when the system is in state $(n,0)$ equals the sum of an exponentially distributed service time with rate μ and the delay of a (virtual) customer arriving in a state with one less customer, i.e. state $(n-1,m)$, where $m=0$ with probability α and $m=1$ with probability $1-\alpha$. A similar reasoning leads to equation (26). After introducing

$$G_{j,m}(\theta) = \sum_{n=2}^{\infty} s^*_{n,m}(\theta)x_j^n \tag{27}$$

and

$$H_j(\theta) = \sum_{m=0}^{1} y_j(m)G_{j,m}(\theta), \tag{28}$$

we get by multiplying (25) and (26) with x_j^n and summing for all $n>2$

$$(\mu+\theta)(G_{j,0}(\theta) - x_j^2 s^*_{2,0}) = \alpha\mu x_j G_{j,0}(\theta) + (1-\alpha)\mu x_j G_{j,1}(\theta), \tag{29}$$

$$(2\mu+\theta)(G_{j,1}(\theta) - x_j^2 s^*_{2,1}) = \mu x_j G_{j,0}(\theta) + \mu x_j G_{j,1}(\theta), \tag{30}$$

Multiplying (29) with $y_j(0)$, (30) with $y_j(1)$ and adding both yields

$$\begin{aligned}\theta H_j(\theta) - (\mu+\theta)x_j^2 y_j(0)s^*_{2,0} - (2\mu+\theta)x_j^2 y_j(1)s^*_{2,1} = \\ G_{j,0}(\theta)((\alpha\mu x_j - \mu)y_j(0) + \mu x_j y_j(1)) \\ + G_{j,1}(\theta)((1-\alpha)\mu x_j y_j(0) + (\mu x_j - 2\mu)y_j(1)).\end{aligned} \tag{31}$$

After using (15) and (16) we get

$$\begin{aligned}\theta H_j(\theta) - (\mu+\theta)x_j^2 y_j(0)s^*_{2,0} - (2\mu+\theta)x_j^2 y_j(1)s^*_{2,1} = \\ G_{j,0}(\theta)\lambda(1-\frac{1}{x_j})y_j(0) + G_{j,1}(\theta)\lambda(1-\frac{1}{x_j})y_j(1).\end{aligned} \tag{32}$$

Using (28) yields

$$H_j(\theta) = \frac{1}{\theta - \lambda(1-\frac{1}{x_j})}\sum_{m=0}^{1}((m+1)\mu+\theta)y_j(m)x_j^2 s^*_{2,m}(\theta) \tag{33}$$

And finally, by (24),

$$s^*(\theta) = p(0)s^*_0 + p(1,0)s^*_{1,0} + p(1,1)s^*_{1,1} + \sum_{j=0}^{1} C_j H_j(\theta). \tag{34}$$

To fully determine $s^*(\theta)$, we need the following

$$s_0^* = \frac{\mu}{\mu+\theta} \tag{35}$$

$$s_{1,0}^* = (1-\alpha)\frac{\mu}{\mu+\theta} + \alpha(\frac{\mu}{\mu+\theta})^2 \tag{36}$$

$$s_{1,1}^* = \alpha\frac{\mu}{\mu+\theta} + (1-\alpha)(\frac{\mu}{\mu+\theta})^2 \tag{37}$$

$$s_{2,0}^* = \alpha(\frac{\mu}{\mu+\theta})^3 + (1-\alpha)(\frac{\mu}{\mu+\theta})^2 \tag{38}$$

$$s_{2,1}^* = \frac{1}{2}\frac{2\mu}{2\mu+\theta}\frac{\mu}{\mu+\theta} + \frac{1}{2}\frac{2\mu}{2\mu+\theta}(\frac{\mu}{\mu+\theta})^2 \tag{39}$$

which are easily deduced. Note that the tagged customer is able to start his service when he is the second customer in the system if the first customer is of a different type. Otherwise, the customer has to wait until he is the first customer in the system. After inserting (35)-(39) into (34) and after some simplifications, we get

$$s^*(\theta) = \sum_{j=0}^{1}\sum_{m=0}^{1} \frac{C_j\mu^2x_j^2y_j(m)((m+1)\mu + (1-\alpha+m\alpha)\lambda(1-\frac{1}{x_j}))}{(\mu-\lambda(1-\frac{1}{x_j}))^2}\frac{1}{\theta-\lambda(1-\frac{1}{x_j})} \tag{40}$$

and by taking the inverse Laplace transform, we find

$$s(t) = \sum_{j=0}^{1}\sum_{m=0}^{1} \frac{C_j\mu^2x_j^2y_j(m)((m+1)\mu + (1-\alpha+m\alpha)\lambda(1-\frac{1}{x_j}))}{(\mu-\lambda(1-\frac{1}{x_j}))^2}e^{\lambda(1-\frac{1}{x_j})t}. \tag{41}$$

Thus, the probability density function $s(t)$ is a hyperexponential distribution.

Knowing the probability density function (pdf), we can again derive some performance measures of practical importance. For example, the mean system delay can be found as

$$\begin{aligned} T &= \int_0^\infty ts(t)dt \\ &= \sum_{j=0}^{1}\sum_{m=0}^{1} \frac{C_j\mu^2x_j^2y_j(m)((m+1)\mu + (1-\alpha+m\alpha)\lambda(1-\frac{1}{x_j}))}{(\mu-\lambda(1-\frac{1}{x_j}))^2}\frac{1}{(\lambda(1-\frac{1}{x_j}))^2}. \end{aligned} \tag{42}$$

While (42) looks a bit different than (22), it can be proven that both are identical by inserting the explicit values for the variables (C_j, $y_j(m)$ and x_j where $j = 0, 1$ and $m = 0, 1$) in function of the parameters (α, μ and λ). The cumulative distribution function (cdf) equals

$$\begin{aligned} S(t) &= \int_0^t s(u)du \\ &= \sum_{j=0}^{1}\sum_{m=0}^{1} \frac{C_j\mu^2x_j^2y_j(m)((m+1)\mu + (1-\alpha+m\alpha)\lambda(1-\frac{1}{x_j}))}{(\mu-\lambda(1-\frac{1}{x_j}))^2}\frac{1-e^{\lambda(1-\frac{1}{x_j})t}}{\lambda(1-\frac{1}{x_j})}. \end{aligned} \tag{43}$$

And the tail probability is given by

$$\begin{aligned} E[s > t] &= 1 - S(t) \\ &= 1 - \sum_{j=0}^{1} \sum_{m=0}^{1} \frac{C_j \mu^2 x_j^2 y_j(m)((m+1)\mu + (1-\alpha+m\alpha)\lambda(1-\frac{1}{x_j}))}{(\mu - \lambda(1-\frac{1}{x_j}))^2} \\ &\quad \frac{1 - e^{\lambda(1-\frac{1}{x_j})t}}{\lambda(1-\frac{1}{x_j})}. \end{aligned} \tag{44}$$

6. **Discussion of results and numerical examples.** In this section, we discuss the results obtained in the previous sections, from a quantitative and a qualitative perspective, by means of some numerical examples.

In some of the results we compare our system with two "extreme" queueing systems which we call "worst" and "best". These are two boundaries for our system (lower and upper). Worst is a well-known queueing system ($M \mid M \mid 1$) with an infinite waiting room, *one* type of customer and *one* server with a mean service time of μ. It is clear that this is a lower boundary for the system, since there is always at least one server working in the system, when there are customers in the system. Best is another well-known queueing system ($M \mid M \mid 2$) with an infinite waiting room, *one* type of customer and *two* servers with a mean service time of μ. This is an upper boundary for the system, since at most two servers are working in the system. The solution of both these queueing systems are standard and can be found in many books about queueing theory e.g. [7].

The first interesting result obtained is the stability condition (5) which shows that the maximum achievable throughput of this system, expressed in average amount of work per unit time, is very directly determined by the degree of class clustering in the arrival process, as described by the cluster parameter α. From the stability condition, we can already deduce that the achievable throughput decreases with the cluster parameter α. It is also interesting to look at the extrema. For the first extremum, $\alpha = 1$, only one type of customers arrives and only one of the servers is being used. The system behaves as the system $M \mid M \mid 1$ and the throughput never exceeds 1 time unit of work per unit time. For the second extremum, $\alpha = 0$, the types of customers arrive alternating. Note that the throughput cannot exceed $\frac{3}{2}$ time units of work per unit time instead of the (possibly expected) 2 time units of work per unit time, which is the maximum throughput of the system $M \mid M \mid 2$. Thus even in the optimal case of $\alpha = 0$, both servers are not working all the time when the system would be constantly provided with new customers. In other words, the system is also in this case non-workconserving. The reason for this is as follows: even when the customers arrive with alternating types, it is possible that the second customer (at the front of the system) has completed his service before the first customer has, since the second customer can have a shorter service time (the second customer overtakes the first customer). The third customer (now becoming the second) then still has to wait for service because the first customer occupies his server. The third customer then blocks the fourth customer that otherwise could have been served because his server is idle, ... So, the randomness of the service times is the culprit here.

We now move to the system delay and system occupancy in stable systems. Fig. 3 shows the mean system delay versus parameter γ for different values of the cluster parameter α. The figure illustrates the (negative) impact of global FCFS, even in

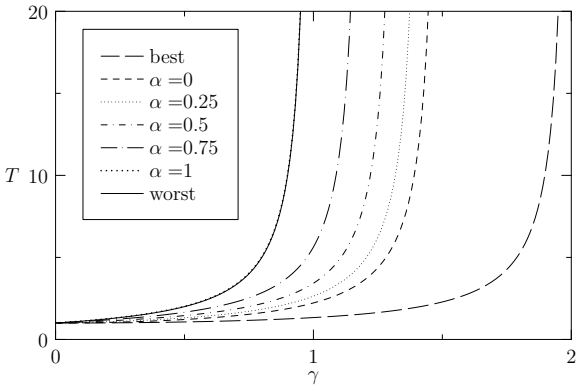

FIGURE 3. Mean system delay versus parameter γ for a given arrival rate of 1

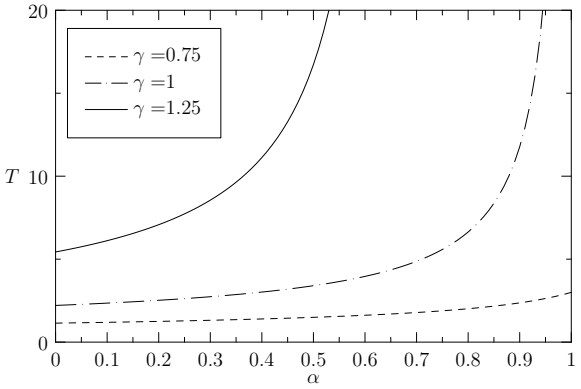

FIGURE 4. Mean system delay versus cluster parameter α for a given arrival rate of 1

the best case (the types of customers arrive alternating or $\alpha = 0$), the system performs much "worse" than the case where the system would be workconserving ($M \mid M \mid 2$). The figure also confirms some previously made observations. Our system behaves identical to the system $M \mid M \mid 1$ for $\alpha = 1$. When α increases, the stability region shrinks. When $\alpha = 0.5$, the type of the next customer is independent of the previous customer, and we can clearly see that neglecting the clustering in the arrival stream causes a considerable underestimation or overestimation of the performance of the system.

In Fig. 4, the mean system delay is shown versus cluster parameter α for different values of the parameter γ. Here, we notice that the impact of the cluster parameter is negligible for small values of γ. This is not surprising. For small values of γ, it is of lesser importance whether only one or both servers work since one server suffices to handle the incoming work. For bigger γ, it is a different story. Here the cluster parameter has a big impact and can even lead to unstable systems. This illustrates that the cluster parameter should not be neglected.

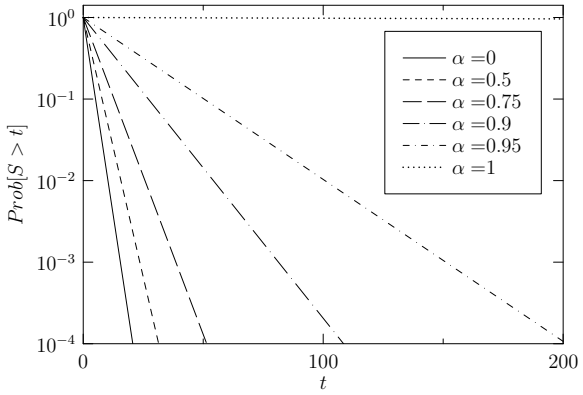

FIGURE 5. Tail probability of the system delay for a given arrival and service rate of 1

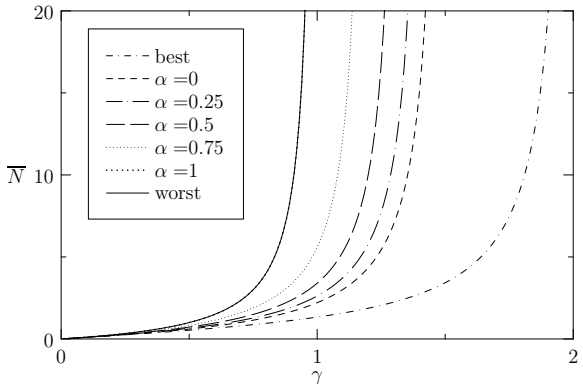

FIGURE 6. Mean system occupancy versus parameter γ

Fig. 5 represents the tail probability of the system delay for a given arrival and service rate of 1. The tail probability gives us the percentage of customers with a system delay greater than t units of time which can for example be of great importance for applications where the delay plays an important role. For example, given an arrival and service rate of 1 and cluster parameter of 0.75, circa 0.01% of the customers has a system delay greater than 51 units of time. This can be used for instance for dimensioning security checks at airports. Since excessive queueing delays cause missing of airplanes by passengers, which should be avoided as much as possible. Alternatively, this can be used for estimation of waiting times, so that passengers can be given the time they have to arrive beforehand. Note that the cluster parameter α should be measured, as this parameter has a crucial impact on performance. Notice also in Fig. 5 that the system is unstable for a given cluster parameter of 1.

In Fig. 6, the mean system occupancy versus parameter γ for different values of the cluster parameter α, is plotted. Here, similar conclusions are drawn as for the mean system delay (Fig. 3).

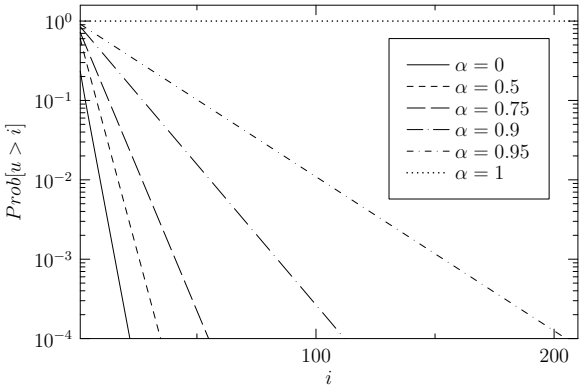

FIGURE 7. Tail probability of the system occupancy for a given arrival and service rate of 1

Finally, Fig. 7 shows the tail probability of the system occupancy for a given arrival and service rate of 1. The tail probability can be considered as an approximate value for the loss probability in a system with finite storage capacity equal to i places which can be used for dimensioning purposes. For example, given an arrival and service rate of 1 and cluster parameter of 0.75, the required buffer size is 56 for a loss ratio of 10^{-4}. Notice in Fig. 7 that the system is unstable for a given cluster parameter of 1 and the loss ratio of 10^{-4} is not achievable.

7. **Conclusions and future work.** In this paper, we have studied a two-class, two-server queue with class-dedicated servers in continuous time assuming a Poisson arrival process where the types of customers are correlated in time according to a cluster parameter α. Due to the introduction of symmetry in the system we were able to propose a conceptual model that was still rich enough to capture the essential aspects of the problem at hand. This model allowed us to derive an explicit closed-form formula for the distributions of the system occupancy and the system delay. This allowed us to uncover the (negative) impact of the combination of global FCFS and class clustering. Class clustering is a concept that often is neglected, but we showed that it can have a considerable impact on a system and ignoring it can cause a considerable overestimation (or underestimation) of the performance.

The model examined in this paper can be generalized in various directions. First of all, more general models could be envisaged to describe the presence of class clustering in the arrival process of the system. A second direction to follow is to let the service times in both servers be different. This would have an effect on the overtaking of customers of different types and this affects blocking. Note that any change in modelling that destroys the current symmetry in the system will lead to a more complex Markov chain. Finally, letting the servers be able to serve one of the m first customers at the front of the queue and thus relaxing the global FCFS restriction is an interesting path to follow as well. We plan to tackle several of these generalizations in future work.

Acknowledgments. This research has been funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office. Furthermore,

the authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

REFERENCES

[1] I. Adan, T. de Kok and J. Resing, *A multi-server queueing model with locking*, EJOR, **116** (2000), 16–26.

[2] I. J. B. F. Adan, J. Wessels and W. H. M. Zijm, *A compensation approach for two-dimensional markov processes*, Advances in Applied Probability, **25** (1993), 783–817.

[3] P. Beekhuizen and J. Resing, *Performance analysis of small non-uniform packet switches,* Performance Evaluation, **66** (2009), 640–659.

[4] Z. Berdowski, F. van den Broek-Serlé, J. Jetten, Y. Kawabata, J. Schoemaker and R. Versteegh, *Survey on standard weights of passengers and baggage*, Survey. EASA 2008.C.06/30800/R20090095/30800000/FBR/RLO, (2009).

[5] D. Bertsimas, *An exact fcfs waiting time analysis for a general class of G/G/s queueing systems*, Queueing Systems Theory Appl., **3** (1988), 305–320.

[6] D. Bertsimas, *An analytic approach to a general class of G/G/s queueing systems,* Operations Research, **38** (1990), 139–155.

[7] P. P. Bocharov and C. D'Apice, "Queueing Theory," Walter de Gruyter, 2004.

[8] W. Grassmann, *Real eigenvalues of certain tridiagonal matrix polynomials, with queueing applications,* Linear Algebra and its Applications, **342** (2002), 93–106.

[9] M. Karol, M. Hluchyj and S. Morgan, *Input versus output queueing on a space-division packet switch*, IEEE Transactions on Communications, **35** (1987), 1347–1356.

[10] K. Laevens, *A processor-sharing model for input-buffered ATM-switches in a correlated traffic environment*, Microprocessors and Microsystems, **22** (1999), 589–596.

[11] S. Liew, *Performance of various input-buffered and output-buffered ATM switch design principles under bursty traffic: simulation study*, IEEE Transactions on Communications, **42** (1994), 1371–1379.

[12] W. Mélange, H. Bruneel, B. Steyaert and J. Walraevens, *A two-class continuous-time queueing model with dedicated servers and global fcfs service discipline*, In "Analytical and Stochastic Modeling Techniques and Applications," Lecture Notes in Computer Science, Springer Berlin / Heidelberg, **6751** (2011), 14–27.

[13] M. F. Neuts, "Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach," Corrected reprint of the 1981 original. Dover Publications, Inc., New York, 1994.

[14] D. Ngoduy, *Derivation of continuum traffic model for weaving sections on freeways*, Transportmetrica, **2** (2006), 199–222.

[15] R. Nishi, H. Miki, A. Tomoeda and K. Nishinari, *Achievement of alternative configurations of vehicles on multiple lanes*, Physical Review E, **79** (2009), 066119.

[16] A. Stolyar, *MaxWeight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic*, Annals of Applied Probability, **14** (2004), 1–53.

[17] T. Van Woensel and N. Vandaele, *Empirical validation of a queueing approach to uninterrupted traffic flows*, 4OR, A Quarterly Journal of Operations Research, **4** (2006), 59–72.

[18] T. Van Woensel and N. Vandaele, *Modeling traffic flows with queueing models: A review*, Asia-Pacific Journal of Operational Research, **24** (2007), 435–461.

Received September 2012; 1st revision January 2013; final revision June 2013.

E-mail address: wmelange@telin.UGent.be
E-mail address: hb@telin.UGent.be
E-mail address: bs@telin.UGent.be
E-mail address: dclaeys@telin.UGent.be
E-mail address: jw@telin.UGent.be