Analysis of a batch-service queue with variable service capacity, correlated customer types and generally distributed class-dependent service times

Jens Baetens[1,*], Bart Steyaert[1], Dieter Claeys[2,3], Herwig Bruneel[1]

Abstract

Queueing models with batch service have been studied frequently, for instance in the domain of telecommunications or manufacturing. Although the batch server's capacity may be variable in practice, only a few authors have included variable capacity in their models. We analyse a batch server with multiple customer classes and a variable service capacity that depends on both the number of waiting customers and their classes. The service times are generally distributed and class-dependent. These features complicate the analysis in a non-trivial way. We tackle it by examining the system state at embedded points, and studying the resulting Markov Chain.

We first establish the joint probability generating function (pgf) of the service capacity and the number of customers left behind in the queue immediately after service initiation epochs. From this joint pgf, we extract the pgf for the number of customers in the queue and in the system respectively at service initiation epochs and departure epochs, and the pgf of the actual server capacity. Combined with additional techniques, we also obtain the pgf of the queue and system content at customer arrival epochs and random slot boundaries, and the pgf of the delay of a random customer. In the numerical experiments, we focus on the impact of correlation between the classes of consecutive customers, and on the influence of different service time distributions on the system performance.

Keywords: Batch Service, Two-Class, Variable Service Capacity, Correlated customer types

1. Introduction

In a manufacturing environment, various machines may occur that are capable of processing multiple products in a single group or batch. These machines can be modelled by a batch-service queueing system, which differs from a multi-server system in that a newly arrived customer cannot join a batch of which the service period has started, even if the maximum capacity of the batch has not been reached yet. Batch-service queueing systems are not only common in manufacturing environments but also in transportation systems where a single vehicle can move a large number

*Corresponding author: jens.baetens@ugent.be
[1]Ghent University, Dept. of Telecommunications and Information Processing, SMACS Research Group, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium
[2]Industrial Systems Engineering (ISyE), Flanders Make, Belgium
[3]Dept. of Industrial Systems Engineering and Product Design, Ghent University

of items at the same time and telecommunication networks where packets are grouped into so-called bursts before transmission in order to reduce overhead by limiting the number of headers. The number of customers that can be grouped together either follows the general batch-service rule (which uses constant minimum and maximum service capacities) or varies from time to time (in)dependently of the state of the system. In this paper, we will use a variable service capacity that depends on the content of the queue.

Another feature we incorporate in this paper is that the server distinguishes two types of customers and that the service time distributions depend on the type of the customers being processed. Customer differentiation, for instance, occurs in manufacturing where a furnace can process multiple product classes with different parameters such as duration or temperature of the heating phase. Using a batch server that allows for customer differentiation has mostly been studied in the context of priority queueing and polling systems. These methods use a dedicated queue for each type of customer processed by the server. However, using multiple dedicated queues or allowing for reordering of the customers in the queue is not always feasible, due to certain constraints such as space or the impact of the increased complexity on the cost and reliability of the system. Another option is to use a single queue that is shared between the classes without reordering of customers. This results in a global FCFS service discipline. The benefit of this option is that we can guarantee a flow throughout the queueing system since one class cannot block another class, and strict fairness rules since customers can not skip ahead in the queue.

In this paper, we analyse a discrete-time batch-service queueing model with two different customer classes, with a variable service capacity that depends on the number of waiting customers and their respective classes. Upon coming available, the single server in this system will start serving a new batch, that contains the sequence of same-class customers at the head of the queue. This type of model can for instance be found in postal distribution centers where a sorter can pick items from the front of a conveyor and sort them according to their destination area. Consecutive letters with the same destination can be sorted simultaneously and the sorting time of such a group is only slightly sensitive to the number of letters that are grouped together since the most significant part of the processing time stems from moving the items to their corresponding box which is independent of the size of the group. This model can also model a batch process in the pacemaker loop of a manufacturing system. The pacemaker loop is a key concept in Lean manufacturing: it is the part of the production system closest to the customer. In the pacemaker loop, buffering is avoided as much as possible to create a flow and, consequently, shorter lead times. However, when buffering is needed in the pacemaker loop, for example in front of a batch process, it should be FIFO buffers to create a consistent flow.

We extend our previous work on batch servers with variable service capacity by including general service time distributions, which do not have to be equal for different classes. This extension complicates the analysis significantly by removing the memoryless property in the service process. In addition, we include correlation, or clustering, between the classes of consecutive customers. In the context of a postal distribution center, for example, this feature models that letters arrive in bags and that letters from the same bag are more likely to be addressed to the same destination region. In the context of a batch process in a pacemaker loop, clustering follows naturally from the scheduling of the pacemaker process, that is the first process in the loop. Customer orders during an interval (e.g. a day) are gathered by the production control department and scheduled for the next interval using a so-called 'Heijunka box'.

An important point in solving the problem is to choose proper embedded points to eliminate the need to keep track of the state of the system at the start of a service, which is required to

know how the system transitions occur after service completion. Contrary to our previous work on batch servers with variable capacity, we start the analysis by looking at the joint probability generating function (pgf) of the queue occupancy and the service capacity after service initiation epochs. This joint pgf allows us to derive a wide range of different performance measures at different time instances. We also include a detailed and extensive analysis of the delay of a random customer.

In the following section 2, we review the literature about batch-service queueing systems. The discrete-time two-class queueing model with batch service is described in detail in Section 3. This system consists of a single First-Come-First-Served (FCFS) queue of infinite size, and a single batch server with a variable capacity. In Section 4, we establish the system equations, from which we deduce the stability condition, and derive a closed-form expression for the steady-state joint pgf of the number of customers in the queue after service initiations epochs, and the number of customers being served, which we consider to be the central result of the paper. Based on this result, we obtain the steady-state pgfs for the queue and system occupancy after service initiation epochs, departure epochs, customer arrival epochs and random slot boundaries as well. We also derive an expression for the steady-state pgf describing the number of customers in a served batch, called the real service capacity in the remainder. We finish the analysis by studying the delay of a random customer. Note that determining the steady-state probability distributions from the obtained pgfs is tedious and often inefficient and therefore usually not done since the performance measures, such as the expected value, can easily be derived from the pgf by using the moments of the pgf. In Section 5, we illustrate this for some specific scenarios by calculating the mean value of the previously mentioned system properties. These results allow us to gain some insight into the system behaviour. Finally, some conclusions and possible extensions are presented in Section 6.

2. Literature review

Since batch servers can be used to model a wide range of applications, queueing systems with a batch server have been studied thoroughly. In the introduction, we presented three options for the service capacity of the batch server. The most commonly used service discipline is the general batch-service rule where two constants are used for the minimum and maximum service capacity. These types of batch servers have been studied by Gupta and Sivakumar [1, 2], Tadj et al. [3], Janssen and van Leeuwaarden [4], Arumuganathan and Jeyakumar [5], Banerjee and Gupta [6], Banerjee et al. [7, 8], Chang and Takine [9], Chaudry and Templeton [10], Claeys et al. [11, 12, 13, 14], Goswami et al. [15], Olbert et al. [16], Pradhan and Gupta [17], and Wang and Odoni [18].

Secondly, the number of customers that can be processed by the batch server can also be stochastic. Chaudry and Chang [19] analysed the number of customers in the system at different time instances in the $\mathrm{Geo}/\mathrm{G}^Y/1/N+B$ model, where Y denotes the stochastic service capacity, B the maximum service capacity and N the size of the buffer. In their paper, this model was used to model a transportation service with multiple vehicles that can have a different available capacity. The model has been extended by Chang and Choi [20], Sikdar and Gupta [21], and Sikdar and Samanta [22] by considering a server that takes a vacation when a service is finished and no customers are waiting to be processed, in order to model a broadband network using Asynchronous Transfer Mode(ATM) technology. Yi et al. [23] published an extension of the work of Chang and Choi [20] by using a minimum threshold before a service is initiated, i.e. by studying the queue-length distribution of the $\mathrm{Geo}/\mathrm{G}^{a,Y}/1/K$ model . Furthermore, Pradhan et al. [24] modelled the testing of blood samples for infectious diseases by considering a $M/G_r^Y/1$ queue and a batch server with a service time distribution that depends on the number of customers in the batch. Germs and

Van Foreest [25] calculated the rejection probability of a random customer for the $M^X/G^Y/1/K+B$ queue with a buffer size of K and maximum service capacity B for three different rejection policies. The previous models study a batch service queueing system with variable service capacity that is independent of the number of waiting customers. The third type of batch-service queueing systems, namely those with variable service capacity dependent on the content of the queue, have also been studied by Germs and Van Foreest [26] where they analysed the $M(n)^{X(n)}/G(n)^{Y(n)}/1/K+B$ batch-service queueing system. In this model, the arrival rate, service time and variable service capacity distributions depend on the number of customers waiting in the queue with finite size.

The second main characteristic of the model considered in this paper is customer differentiation. This has mostly been studied in the context of priority queueing or polling systems. In manufacturing, Reddy et al. [27] study an industrial repair shop that repairs the most critical machines first. In telecommunication systems, priority queueing systems have been studied by, for instance, Zhao et al. [28] or Walraevens et al. [29]. Polling systems are also common in telecoummnications and have been studied by Boxma et al. [30], Dorsman et al. [31] and Fowler et al. [32]. Queueing systems where queues are shared between different classes have been studied by, for instance, Bruneel et al. [33, 34], Mélange et al. [35], Maertens et al. [36], Reveil et al. [37].

In our previous paper, see Baetens et al. [38], we studied the system occupancy in a simplified model where the service times of both classes were identical to a single slot and where there was no correlation between the classes of consecutive customers, and in Baetens et al. [39], we analysed the delay of customers in that model. In Baetens et al. [40], we analysed an extension of that model by including geometrically distributed service times and customer based-correlation. The analysis of the delay of a random customer was briefly outlined in our extended abstract, see Baetens et al. [41]. As mentioned in the introduction, applications of the batch-service queueing system in this paper can be found in the pacemaker loop in Lean manufacturing which requires FIFO-buffers in order to create a consistent flow, and a 'Heijunka box' which introduces correlation between the classes of consecutive customers by presorting the arrivals over short time periods. For a more detailed explanation of these terms, we refer to Duggan [42] or Rother [43].

3. Model description

In this paper, we consider a discrete-time two-class (called class A and B) queueing system with an infinite queue size and a batch server whose capacity is stochastic. The batch server uses a global FCFS service discipline, which means that newly arrived customers are added at the tail of a single queue. When the server is idle or has finished processing a batch and finds a non-empty queue at a slot boundary, a new service is initiated immediately. The number of customers grouped together in this service is determined by the length of the sequence of consecutive same-class customers at the head of the queue. More specifically, the variable service capacity will be equal to n, if and only if one of the following two cases occur:

- All n customers present in the queue are of the same class.
- There are more than n customers awaiting processing, and the first n customers in the queue are of the same class while the $(n+1)$-th customer is of the opposite class.

Since all customers in a single batch are of the same class, we define the class of a batch as the class of the customers in the batch. We note that in theory, the server capacity can become very large when there is a very large sequence of consecutive customers of the same type. However, in

Lean manufacturing, one of the key principles is to keep the "interval", also called EPEI (Every Product Every Interval) low, which means smaller batches and leads to less inventory and shorter lead times, see Duggan [42] or Rother [43]. Hence, in practice, the server capacity will mainly be limited by the batch size.

We model the aggregated numbers of customer arrivals in consecutive slots by a sequence of independent and identically distributed (i.i.d.) random variables. These random variables follow a general distribution with common probability mass function (pmf) $e(n)$ and pgf $E(z)$. The mean number of customers that arrive in a single slot is denoted as λ. The classes of consecutive customers within the aggregated arrival stream are governed by an irreducible time-homogeneous two-state Markov chain $\{\zeta_j; j \geq 1\}$ with transition probabilities

$$\alpha := Pr[\zeta_{j+1} = A|\zeta_j = A] \ , \qquad \beta := Pr[\zeta_{j+1} = B|\zeta_j = B] \ .$$

The steady-state probabilities of this Markov chain readily follow from elementary theory of Markov chains (see e.g. Ross [44], Chapter 4):

$$\lim_{j\to\infty} Pr[\zeta_j = A] = \frac{1-\beta}{2-\alpha-\beta} \ , \qquad \lim_{j\to\infty} Pr[\zeta_j = B] = \frac{1-\alpha}{2-\alpha-\beta} \ .$$

The service time of a batch of customers only depends on the class of the batch and not on the service capacity. Given that class, the service time is generally distributed and independent of the capacity of the server, the classes of previous batches and previous service times. The service time of a batch of class A (B) is characterized by the pmf $s_A(n)$ ($s_B(n)$) and the pgf $S_A(z)$ ($S_B(z)$).

4. Analysis

In this section, we first determine the system equations that govern the system behaviour. The key is to select proper embedded epochs on which a Markov Chain can be defined that has the lowest possible dimension while still enabling us to derive a wide range of performance values.

As we will explain in detail in Section 4.1, we select the epochs immediately after service initiation epochs and keep track of the number of customers in service, the number of customers left behind in the queue, and the class of the batch in service. Then we analyse the conditions for stability, and we establish the steady-state joint pgf of the queue occupancy, that is the number of customers waiting in the buffer, after initiation of a service, and the number of customers in the batch being served. Based on that result, we derive the steady-state pgfs for the queue occupancy after service initiation, service departure and at random slot boundaries. Next, we deduce the steady-state pgfs for the sizes of the served batches, and the system occupancy, that is the total number of customers in the system, after service initiation, after service departure, and at random slot boundaries.

4.1. System equations

In this subsection we establish the equations that capture the behaviour of the system at consecutive service initiation epochs. A batch consists of all consecutive same-class customers that are at the head of the queue at the time of its service initiation. The queue occupancy after service initiation of the k-th batch is denoted by q_k. We also denote the class of the k-th batch in service by t_k.

First, we examine t_{k+1}. Let us distinguish between $q_k > 0$ and $q_k = 0$. If $q_k > 0$, the first customer left behind, that is the customer that becomes the head of the queue immediately after the k-th service is initiated, is of the opposite class as the batch in service, because otherwise it would also have been taken for service. Hence, if $q_k > 0$, the $(k+1)$-th batch is of the opposite class as the k-th batch. On the other hand, when $q_k = 0$, all customers have been taken into service and the class of the next customer is determined by the transition probabilities α and β. Summarized, we have

$$t_{k+1} = \begin{cases} \neq t_k & \text{if } q_k > 0 \\ A & \text{with probability } \alpha \text{ if } t_k = A \text{ and } q_k = 0 \\ B & \text{with probability } (1-\alpha) \text{ if } t_k = A \text{ and } q_k = 0 \\ B & \text{with probability } \beta \text{ if } t_k = B \text{ and } q_k = 0 \\ A & \text{with probability } (1-\beta) \text{ if } t_k = B \text{ and } q_k = 0 \ . \end{cases} \tag{1}$$

Second, we examine q_{k+1} and again consider two cases. The first case is depicted in Fig. 1 and shows the situation when the queue contains at least one customer at the end of the k-th service, that is $q_k + \hat{e}_k > 0$ with $\hat{e}_k$ the number of customers that arrive during the service time of the k-th batch. In this case, the $(k+1)$-th service starts immediately after the k-th service has ended. The situation when the queue is empty at the end of the k-th service is depicted in Fig. 2. In this case, the server is idle until there is at least one arrival in a slot. We also define $\tilde{e}_k$ as the number of arrivals in the slot before initiation of the k-th service, given that there is at least one arrival. We then obtain the following equations:

$$q_{k+1} = \begin{cases} (q_k + \hat{e}_k - c_{k+1})^+ & \text{if } q_k + \hat{e}_k > 0 \\ (\tilde{e}_{k+1} - c_{k+1})^+ & \text{else} \ , \end{cases} \tag{2}$$

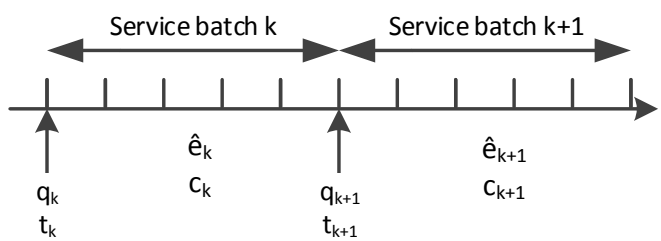


Figure 1: Relationship between q_k and q_{k+1} if $q_k + \hat{e}_k > 0$

where $(\ldots)^+ := \max(0, \ldots)$ and c_k is the theoretical capacity of the k-th service. The real capacity is the actual number of customers that are served in the k-th batch, while the theoretical capacity is the number of customers that would belong to this batch if an unlimited number of customers were available. The real service capacity can be derived from the theoretical capacity by taking the minimum of the theoretic service capacity and the number of customers in the queue before service initiation. Therefore, the theoretical service capacity is an upper bound for the real service capacity. The benefit of working with the theoretical service capacity is that it only depends on the type of the service, and not on the queue occupancy before service initiation. The pmf

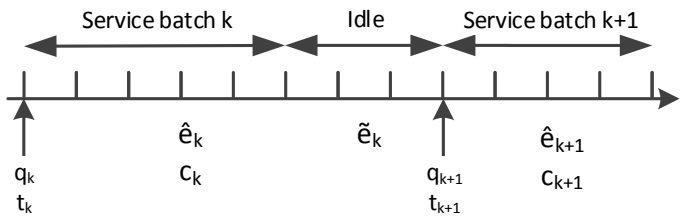


Figure 2: Relationship between q_k and q_{k+1} if $q_k + \hat{e}_k = 0$.

of the theoretical service capacity is $Pr[c_k = n|t_k = A] = (1-\alpha)\alpha^{n-1}$ in case of a class A batch or $Pr[c_k = n|t_k = B] = (1-\beta)\beta^{n-1}$ for a class B batch, with mean values of $1/(1-\alpha)$ and $1/(1-\beta)$ respectively. These conditional distributions of the theoretical capacity correspond to shifted geometric distributions, which possess the memoryless property. When c_k exceeds the number of waiting customers , then the real service capacity follows a shifted geometric distribution truncated by the number of waiting customers before service initiation. In Eq. 2, we also used the random variable $\tilde{e}_k$. The pmf $\tilde{e}(n)$ and pgf $\tilde{E}(z)$ of this random variable are given by

$$\tilde{e}(n) = \lim_{k\to\infty} Pr[\tilde{e}_k = n] = \frac{e(n)}{\sum\limits_{i=1}^{\infty} e(n)} = \frac{e(n)}{1-e(0)}$$

$$\tilde{E}(z) = \sum_{i=1}^{\infty} \tilde{e}(n) z^n = \frac{E(z) - E(0)}{1 - E(0)} \quad .$$

Using the theoretical capacity instead of the real capacity is not a trivial step in the analysis. This simplifies the analysis by eliminating the dependency between the size of the batch being processed and the number of customers in the queue before service initiation. We can eliminate this dependency because the only difference occurs when the queue would be empty after service initiation. When the theoretical capacity is larger than the queue occupancy before service initiation, this means that the queue is empty after service initiation.

From Eqs. 1 and 2, it follows that $\{(q_k, t_k), k \geq 0\}$ is a first-order 2-D Markov chain where q_k and t_k correspond respectively to the queue occupancy and class of the ongoing batch after initiation of the k-th service. Without significantly increasing the complexity of the analysis, we can add the number of customers in the ongoing batch to the Markov Chain, which allows us to obtain the pgf of a wide range of performance values.

4.2. Stability condition

Bruneel and Kim [45] have shown that the system will be stable when the mean number of arrivals in a certain time period is less than the mean number of customers that can be processed in the same time period. This method for calculating the stability condition has also been studied and rigorously proven by Baccelli and Foss [46] This approach has also been used by, for instance, Kuehn [47], Foss et al. [48], and Kim and De Veciana [49]. In a saturated system, i.e. a system where there are always many customers present, the server of the system is always active and alternates between serving a class A and a class B batch, whose sizes are equal to the respective theoretical capacities. Let us therefore examine an AB-period, which starts at the service initiation of a class

A batch and ends at the service completion of the following class B batch. The mean amount of work that can leave the system is given by the sum of the average capacities of a class A and B batch, leading to the following stability condition:

$$\lambda(S'_A(1) + S'_B(1)) < \frac{1}{1-\alpha} + \frac{1}{1-\beta} \ , \tag{3}$$

where the left-hand side corresponds to the mean number of arrivals in an AB-period, and the right-hand side to the sum of the expected length of a class A and B batch in a saturated system, which is equal to the number of customers leaving the system during an AB-period. If either α or β equals 1, then the stability condition is reduced to $\lambda < \infty$, i.e., the system is always stable. This is as expected, since in this case all customers are of the same class, which means that no matter how many customers arrive, the queue will always be empty after service initiation epochs. Also, any increase in α and β leads on average to larger batches for the corresponding class and a larger maximum tolerable arrival rate λ.

Since α and β both merely capture the class clustering (the tendency of same-class customers to arrive back-to-back) of a single class, the combined level of class clustering is not easily quantifiable with these parameters. By introducing the parameters σ and τ we can define the arrival process relative to a process without correlation between the classes of consecutive customers. The parameter $\sigma = \frac{1-\beta}{2-\alpha-\beta}$ is defined as the probability that a random customer is of class A, and $\tau = \frac{1-\sigma}{1-\alpha} = \frac{\sigma}{1-\beta}$ as the ratio of the average size of a class-A (or B) batch, relative to the average size of a class-A (or B) batch in case of uncorrelated customer classes (i.e., each customer is of class A with probability σ, independent of the class of other customers). The global level of class clustering is thus captured by the value of τ, and $\tau = 1$ implies that the consecutive customer classes form an uncorrelated process. The stability condition can now be written as

$$\lambda < \frac{\tau}{\sigma(1-\sigma)} \frac{1}{(S'_A(1) + S'_B(1))} = \frac{\tau K}{S'_A(1) + S'_B(1)} \ . \tag{4}$$

In this condition, we defined $K = \frac{1}{\sigma} + \frac{1}{1-\sigma} = \frac{1}{\sigma(1-\sigma)}$ as the sum (or product) of the average capacity provided that there is no class clustering in the system. The variable $K \in [4, +\infty[$ gives an indication of the symmetry in the arrival process without correlation. The system is symmetric when $K = 4$ and higher values of K lead to more asymmetric systems. Based on Eq. 4, we can easily see that the maximum number of customers that the system can process is linear in τ (global level of class clustering) and K (global level of asymmetry in the customer classes).

4.3. Joint generating function

Assuming the stability condition is met, we can characterize the state of the system just after service initiation epochs by the type of the batch to be served, the number of customers left behind in the queue and the number of customers being served. Analysing the system after service initiation epochs avoids having to keep track of the residual service time, resulting in a reduced complexity. Using $\hat{c}_k$ as the number of customers being served (the real capacity), we can define the steady-state joint probabilities for the Markov chain $\{(t_k, q_k, \hat{c}_k), k \geq 0\}$ as

$$p_A(i,j) := \lim_{k\to+\infty} Pr[t_k = A, q_k = i, \hat{c}_k = j] \ , p_B(i,j) := \lim_{k\to+\infty} Pr[t_k = B, q_k = i, \hat{c}_k = j] \ ,$$

for all $i, j \geq 0$, with corresponding partial pgfs $P_A(z,x)$ and $P_B(z,x)$. The marginal steady-state pmf after service initiation epochs is then given by

$$p(i,j) \coloneqq \lim_{k \to +\infty} Pr[q_k = i, \hat{c}_k = j] = p_A(i,j) + p_B(i,j) \ ,$$

with pgf

$$P(z,x) \coloneqq \sum_{i,j \geq 0} p(i,j) z^i x^j = P_A(z,x) + P_B(z,x) \ .$$

We also define the steady-state probabilities for the 2-dimensional Markov chain $\{(t_k, q_k), k \geq 0\}$ as

$$q_A(i) \coloneqq \lim_{k \to +\infty} Pr[t_k = A, q_k = i] = \sum_{j=0}^{\infty} p_A(i,j) \ ,$$

$$q_B(i) \coloneqq \lim_{k \to +\infty} Pr[t_k = B, q_k = i] = \sum_{j=0}^{\infty} p_B(i,j) \ ,$$

for all $i \geq 0$, with corresponding partial pgfs $Q_A(z)$ and $Q_B(z)$.

Due to symmetry between class A and B, we only present the approach to obtain $p_A(i,j)$ and $P_A(z,x)$. The expressions for class B are obtained similarly. Using Eq. 1, we obtain, $\forall k, l \geq 0$

$$\begin{aligned}
p_A(m,l) &= \lim_{k \to \infty} Pr[t_{k+1} = A, q_{k+1} = m, \hat{c}_{k+1} = l] \\
&= \sum_{i=0}^{\infty} \lim_{k \to \infty} Pr[t_k = A, q_k = i] Pr[t_{k+1} = A, q_{k+1} = m, \hat{c}_{k+1} = l | t_k = A, q_k = i] \\
&\quad + \sum_{i=0}^{\infty} \lim_{k \to \infty} Pr[t_k = B, q_k = i] Pr[t_{k+1} = A, q_{k+1} = m, \hat{c}_{k+1} = l | t_k = B, q_k = i] \\
&= \alpha \lim_{k \to \infty} q_A(0) Pr[q_{k+1} = m, \hat{c}_{k+1} = l | t_{k+1} = A, t_k = A, q_k = 0] \\
&\quad + (1-\beta) \lim_{k \to \infty} q_B(0) Pr[q_{k+1} = m, \hat{c}_{k+1} = l | t_{k+1} = A, t_k = B, q_k = 0] \\
&\quad + \sum_{i=1}^{\infty} \lim_{k \to \infty} q_B(i) Pr[q_{k+1} = m, \hat{c}_{k+1} = l | t_{k+1} = A, t_k = B, q_k = i] \ . \qquad (5)
\end{aligned}$$

Taking the 2-D Z-transform of Eq. 5 yields

$$\begin{aligned}
P_A(z,x) = &\alpha q_A(0) \lim_{k \to \infty} E[z^{q_{k+1}} x^{\hat{c}_{k+1}} | t_{k+1} = A, t_k = A, q_k = 0] \\
&+ (1-\beta) q_B(0) \lim_{k \to \infty} E[z^{q_{k+1}} x^{\hat{c}_{k+1}} | t_{k+1} = A, t_k = B, q_k = 0] \\
&+ \sum_{i=1}^{\infty} \lim_{k \to \infty} q_B(i) E[z^{q_{k+1}} x^{\hat{c}_{k+1}} | t_{k+1} = A, t_k = B, q_k = i] \ . \qquad (6)
\end{aligned}$$

This can be further reduced by using Eq. 2 for q_{k+1}. The number of customers being served in the $(k+1)$-th batch ($\hat{c}_{k+1}$) is the minimum of the theoretical capacity (c_{k+1}) and the number of

customers in the queue before service initiation. We illustrate our approach for the third term in the right-hand side of the previous expression. The other terms can be found analogously. Invoking the property that c_{k+1} is conditionally independent of $\hat{e}_k$ and q_k given t_k, and that $\hat{e}_k$ is conditionally independent of q_k and t_k, leads to

$$\begin{aligned}
&\sum_{i=1}^{\infty} \lim_{k\to\infty} q_B(i) E[z^{q_{k+1}} x^{\hat{c}_{k+1}} | t_{k+1} = A, t_k = B, q_k = i] \\
&\quad = \sum_{i=1}^{\infty} \lim_{k\to\infty} q_B(i) E[z^{(i+\hat{e}_k - c_{k+1})^+} x^{\min(i+\hat{e}_k, c_{k+1})} | t_{k+1} = A, t_k = B] \\
&\quad = \sum_{i=1}^{\infty} q_B(i) \sum_{e=0}^{\infty} Pr[\hat{e}_k = e | t_k = B] \Big(\sum_{n=1}^{i+e} Pr[c_{k+1} = n | t_{k+1} = A] z^{i+e-n} x^n \\
&\qquad + \sum_{n=i+e+1}^{\infty} Pr[c_{k+1} = n | t_{k+1} = A] x^{i+e} \Big) \ .
\end{aligned}$$

Since the theoretical capacity of a class A batch follows a geometric distribution with parameter α, working out the innermost sums results in

$$\begin{aligned}
&\sum_{i=1}^{\infty} \lim_{k\to\infty} q_B(i) E[z^{q_{k+1}} x^{\hat{c}_{k+1}} | t_{k+1} = A, t_k = B, q_k = i] \\
&\quad = \sum_{i=1}^{\infty} q_B(i) \sum_{e=0}^{\infty} Pr[\hat{e}_k = e | t_k = B] \Big(\frac{(1-\alpha)x}{\alpha x - z} ((\alpha x)^{i+e} - z^{i+e}) + (\alpha x)^{i+e} \Big) \ .
\end{aligned}$$

Using the definitions of $P_B(z,x)$, $S_B(z)$ and $E(z)$ leads to

$$\begin{aligned}
&\sum_{i=1}^{\infty} \lim_{k\to\infty} q_B(i) E[z^{q_{k+1}} x^{\hat{c}_{k+1}} | t_{k+1} = A, t_k = B, q_k = i] \\
&\quad = \frac{x-z}{\alpha x - z} S_B(E(\alpha x)) P_B(\alpha x, 1) - \frac{(1-\alpha)x}{\alpha x - z} S_B(E(z)) P_B(z, 1) \\
&\qquad - q_B(0) \Big(\frac{x-z}{\alpha x - z} S_B(E(\alpha x)) - \frac{(1-\alpha)x}{\alpha x - z} S_B(E(z)) \Big) \ .
\end{aligned}$$

Combining the three terms that result from the above derivations yields for Eq. 6, after some calculations,

$$\begin{aligned}
P_A(z,x) =& \alpha q_A(0) \Big(S_A(E(0)) \Big(\frac{x-z}{\alpha x - z} \frac{E(\alpha x) - E(0)}{1 - E(0)} - \frac{(1-\alpha)x}{\alpha x - z} \frac{E(z) - E(0)}{1 - E(0)} - 1 \Big) \\
&+ \frac{x-z}{\alpha x - z} S_A(E(\alpha x)) - \frac{(1-\alpha)x}{\alpha x - z} S_A(E(z)) \Big) \\
&+ q_B(0) \Big(S_B(E(0)) \Big(\frac{(1-\beta)(x-z)}{\alpha x - z} \frac{E(\alpha x) - E(0)}{1 - E(0)} - \frac{(1-\alpha)(1-\beta)x}{\alpha x - z} \frac{E(z) - E(0)}{1 - E(0)} \\
&- (1-\beta) \Big) - \beta \frac{x-z}{\alpha x - z} S_B(E(\alpha x)) + \beta \frac{(1-\alpha)x}{\alpha x - z} S_B(E(z)) \Big) \\
&+ \frac{x-z}{\alpha x - z} S_B(E(\alpha x)) P_B(\alpha x, 1) - \frac{(1-\alpha)x}{\alpha x - z} S_B(E(z)) P_B(z, 1) \ . \qquad (7)
\end{aligned}$$

A similar analysis leads to the symmetric equation for class B:

$$\begin{aligned}
P_B(z,x) =& \beta q_B(0)\Big(S_B(E(0))\Big(\frac{x-z}{\beta x - z}\frac{E(\beta x)-E(0)}{1-E(0)} - \frac{(1-\beta)x}{\beta x - z}\frac{E(z)-E(0)}{1-E(0)} - 1\Big) \\
&+ \frac{x-z}{\beta x - z}S_B(E(\beta x)) - \frac{(1-\beta)x}{\beta x - z}S_B(E(z))\Big) \\
&+ q_A(0)\Big(S_A(E(0))\Big(\frac{(1-\alpha)(x-z)}{\beta x - z}\frac{E(\beta x)-E(0)}{1-E(0)} - \frac{(1-\alpha)(1-\beta)x}{\beta x - z}\frac{E(z)-E(0)}{1-E(0)} \\
&- (1-\alpha)\Big) - \alpha\frac{x-z}{\beta x - z}S_A(E(\beta x)) + \alpha\frac{(1-\beta)x}{\beta x - z}S_A(E(z))\Big) \\
&+ \frac{x-z}{\beta x - z}S_A(E(\beta x))P_A(\beta x,1) - \frac{(1-\beta)x}{\beta x - z}S_A(E(z))P_A(z,1) \ . \qquad (8)
\end{aligned}$$

The unknowns $q_A(0)$ and $q_B(0)$ can be determined by studying $P_A(z,1)$ and $P_B(z,1)$, which are the pgfs of the queue occupancy after initiation of a class A and class B service respectively. These variables are calculated in the next section during the analysis of the steady-state pgf of the queue occupancy.

4.4. Probability generating function of the queue occupancy

4.4.1. After service initiation epochs

Using the results of the previous section we readily obtain the marginal steady-state pgf of the number of customers in the queue after service initiation. The partial pgf $Q_A(z)$ is found by evaluating $P_A(z,x)$ at $x=1$. This gives the following equation:

$$\begin{aligned}
Q_A(z) =& \alpha q_A(0)\Bigg(S_A(E(0))\Big[\frac{1-z}{\alpha - z}\frac{E(\alpha)-E(0)}{1-E(0)} - \frac{1-\alpha}{\alpha - z}\frac{E(z)-E(0)}{1-E(0)} - 1\Big] + \frac{1-z}{\alpha - z}S_A(E(\alpha)) \\
&- \frac{1-\alpha}{\alpha - z}S_A(E(z))\Bigg) + q_B(0)\Bigg(S_B(E(0))\Big[\frac{(1-\beta)(1-z)}{\alpha - z}\frac{E(\alpha)-E(0)}{1-E(0)} \\
&- \frac{(1-\alpha)(1-\beta)}{\alpha - z}\frac{E(z)-E(0)}{1-E(0)} - (1-\beta)\Big] - \beta\frac{1-z}{\alpha - z}S_B(E(\alpha)) + \beta\frac{1-\alpha}{\alpha - z}S_B(E(z))\Bigg) \\
&+ \frac{1-z}{\alpha - z}S_B(E(\alpha))Q_B(\alpha) - \frac{1-\alpha}{\alpha - z}S_B(E(z))Q_B(z) \ . \qquad (9)
\end{aligned}$$

Note that this expression contains the unknown constant $S_B(E(\alpha))Q_B(\alpha)$. Letting $z \to 0$ in Eq. 9 and invoking that $Q_A(0) = q_A(0)$ yields

$$\begin{aligned}
S_B(E(\alpha))Q_B(\alpha) =& \alpha q_A(0)\left(1 + S_A(E(0))\frac{1-E(\alpha)}{1-E(0)} - S_A(E(\alpha))\right) \\
&+ q_B(0)\left(\beta S_B(E(\alpha)) + S_B(E(0))(1-\beta)\frac{1-E(\alpha)}{1-E(0)}\right) \ . \qquad (10)
\end{aligned}$$

By substituting Eq. 10 into Eq. 9 we obtain

$$\begin{aligned}Q_A(z) =& \frac{\alpha q_A(0)}{\alpha - z}\left(1 - z + (1-\alpha)\frac{1-E(z)}{1-E(0)}S_A(E(0)) - (1-\alpha)S_A(E(z))\right)\\ &+ \frac{(1-\alpha)q_B(0)}{\alpha - z}\left((1-\beta)\frac{1-E(z)}{1-E(0)}S_B(E(0)) + \beta S_B(E(z))\right)\\ &- \frac{1-\alpha}{\alpha - z}S_B(E(z))Q_B(z) \ , \end{aligned} \tag{11}$$

which is a linear equation for $Q_A(z)$ and $Q_B(z)$ that contains the unknown probabilities $q_A(0)$ and $q_B(0)$. A similar analysis leads to a symmetric equation for class B:

$$\begin{aligned}Q_B(z) =& \frac{\beta q_B(0)}{\beta - z}\left(1 - z + (1-\beta)\frac{1-E(z)}{1-E(0)}S_B(E(0)) - (1-\beta)S_B(E(z))\right)\\ &+ \frac{(1-\beta)q_A(0)}{\beta - z}\left((1-\alpha)\frac{1-E(z)}{1-E(0)}S_A(E(0)) + \alpha S_A(E(z))\right)\\ &- \frac{1-\beta}{\beta - z}S_A(E(z))Q_A(z) \ . \end{aligned} \tag{12}$$

Expressions 11 and 12 constitute a set of 2 independent linear equations in $Q_A(z)$ and $Q_B(z)$, with the solution

$$\begin{aligned}&Q_A(z)\Big[(\alpha - z)(\beta - z) - (1-\alpha)(1-\beta)S_A(E(z))S_B(E(z))\Big]\\ &=\alpha(\beta - z)q_A(0)\left(1 - z + (1-\alpha)\frac{1-E(z)}{1-E(0)}S_A(E(0)) - (1-\alpha)S_A(E(z))\right)\\ &\quad + (1-\alpha)(\beta - z)q_B(0)\left((1-\beta)\frac{1-E(z)}{1-E(0)}S_B(E(0)) + \beta S_B(E(z))\right)\\ &\quad - (1-\alpha)S_B(E(z))\Bigg(\beta q_B(0)\left(1 - z + (1-\beta)\frac{1-E(z)}{1-E(0)}S_B(E(0)) - (1-\beta)S_B(E(z))\right)\\ &\quad + (1-\beta)q_A(0)\left((1-\alpha)\frac{1-E(z)}{1-E(0)}S_A(E(0)) + \alpha S_A(E(z))\right)\Bigg) \ , \end{aligned} \tag{13}$$

and

$$\begin{aligned}&Q_B(z)\Big[(\alpha - z)(\beta - z) - (1-\alpha)(1-\beta)S_A(E(z))S_B(E(z))\Big]\\ &=\beta(\alpha - z)q_B(0)\left(1 - z + (1-\beta)\frac{1-E(z)}{1-E(0)}S_B(E(0)) - (1-\beta)S_B(E(z))\right)\\ &\quad + (1-\beta)(\alpha - z)q_A(0)\left((1-\alpha)\frac{1-E(z)}{1-E(0)}S_A(E(0)) + \alpha S_A(E(z))\right)\\ &\quad - (1-\beta)S_A(E(z))\Bigg(\alpha q_A(0)\left(1 - z + (1-\alpha)\frac{1-E(z)}{1-E(0)}S_A(E(0)) - (1-\alpha)S_A(E(z))\right)\\ &\quad + (1-\alpha)q_B(0)\left((1-\beta)\frac{1-E(z)}{1-E(0)}S_B(E(0)) + \beta S_B(E(z))\right)\Bigg) \ . \end{aligned} \tag{14}$$

The two boundary probabilities $q_A(0)$ and $q_B(0)$ are yet to be determined. To find a solution for these unknowns, we require two independent linear equations. The first one is derived from the normalization condition of the pgf $Q(z) \triangleq Q_A(z) + Q_B(z)$, i.e. $Q(1) = 1$. Using the l'Hôpital's rule to establish this equation, we obtain

$$\begin{aligned}
&2 - \alpha - \beta - (1-\alpha)(1-\beta)\lambda(S'_A(1) + S'_B(1)) \\
&\quad = q_A(0)(1-\alpha)\Big(2\alpha + S'_A(1)\lambda\big(\alpha(\beta-\alpha) + 1 - \beta\big) - S'_B(1)\lambda\alpha(1-\beta) \\
&\qquad + \frac{\lambda S_A(E(0))}{1-E(0)}\big(\alpha(1-\alpha) + 2(1-\alpha)(1-\beta) + (1-\beta)\big)\Big) \\
&\qquad + q_B(0)(1-\beta)\Big(2\beta + S'_B(1)\lambda\big(\beta(\alpha-\beta) + 1 - \alpha\big) - S'_A(1)\lambda\beta(1-\alpha) \\
&\qquad + \frac{\lambda S_B(E(0))}{1-E(0)}\big(\beta(1-\beta) + 2(1-\alpha)(1-\beta) + (1-\alpha)\big)\Big) \ .
\end{aligned}$$

A second equation is obtained by exploiting the property that $Q(z)$ is a pgf, and thus analytic for $|z| < 1$ and bounded for $|z| \leq 1$. We first define the numerators of $Q_A(z)$ and $Q_B(z)$ respectively as $N_A(z)$ and $N_B(z)$, that is, $N_A(z)$ and $N_B(z)$ are the right-hand sides of Eqs. 13 and 14 respectively. The common denominator $Den(z)$ of $Q_A(z)$ and $Q_B(z)$ is equal to $(\alpha - z)(\beta - z) - (1-\alpha)(1-\beta)S_A(E(z))S_B(E(z))$. We start by proving that $Den(z)$ has two zeroes inside the closed complex unit disk $\{z \in \mathbb{C} : |z| \leq 1\}$. The first zero is equal to 1 and leads to the same equation as the normalization condition. To prove that there is a second zero inside the unit disk, we aim to apply Rouché's theorem, see Adan et al. [50], and therefore rewrite $Den(z)$ as $f(z) - g(z)$, where the functions $f(z)$ and $g(z)$ are defined as

$$\begin{aligned}
f(z) &\coloneqq (z-\alpha)(z-\beta) \ , \\
g(z) &\coloneqq (1-\alpha)(1-\beta)S_A(E(z))S_B(E(z)) \ .
\end{aligned}$$

First, both of these functions are analytic functions inside the unit disk, and it is clear that $f(z)$ has two zeroes, α and β, inside this disk. The next step is to prove that $|f(z)| > |g(z)|$ on the contour $|z| = 1 + \epsilon$, $\epsilon > 0$

$$\begin{aligned}
|f(z)| &= |z-\alpha||z-\beta| \geq (1+\epsilon-\alpha)(1+\epsilon-\beta) = (1-\alpha)(1-\beta) + \epsilon(2-\alpha-\beta) + O(\epsilon^2) \\
|g(z)| &= (1-\alpha)(1-\beta)|S_A(E(z))||S_B(E(z))| \\
&\leq (1-\alpha)(1-\beta)S_A(E(1+\epsilon))S_B(E(1+\epsilon)) \\
&\leq (1-\alpha)(1-\beta)(1 + \epsilon\lambda S'_A(1) + O(\epsilon^2))(1 + \epsilon\lambda S'_B(1) + O(\epsilon^2)) \\
&\leq (1-\alpha)(1-\beta) + \epsilon\left[(1-\alpha)(1-\beta)\lambda\left(S'_A(1) + S'_B(1)\right)\right] + O(\epsilon^2) \ .
\end{aligned}$$

Consequently,

$$|f(z)| - |g(z)| \geq \epsilon(2 - \alpha - \beta - (1-\alpha)(1-\beta)\lambda(S'_A(1) + S'_B(1)) + O(\epsilon^2) > 0 \ ,$$

where the last inequality follows from applying the stability condition, see Eq. 3.

Consequently, all conditions are fulfilled for applying Rouché's theorem, leading to the conclusion that $Den(z)$ has two zeroes inside and on the closed complex unit disk, including $z = 1$. The

zero $\hat{z}$, different from 1, can be calculated numerically. Since the partial pgfs $Q_A(z)$ and $Q_B(z)$ are analytical in the complex unit disk, their numerators must vanish at $\hat{z}$, that is $N_A(\hat{z}) = 0$ and $N_B(\hat{z}) = 0$. Combined with the normalisation condition, this would constitute a set of three linear equations for the two unknowns. However, a closer investigation of $N_A(\hat{z}) = 0$ and $N_B(\hat{z}) = 0$ reveals that these are not independent equations, so either one of them, together with the normalisation condition, allows us to calculate $q_A(0)$ and $q_B(0)$.

4.4.2. After departure epochs

The results for the queue occupancy after service initiation epochs can be used to determine the pgf of the queue occupancy at departure epochs. At these time instances, the batch has left the server and a new service has not yet been initiated. We define the partial pgf of the queue occupancy at departure epochs of a class-A (B) batch as $Q_{D,A}(z)$ ($Q_{D,B}(z)$). The combined pgf of all departure epochs is defined as $Q_D(z)$. As the queue occupancy at the end of a service is given by the sum of the queue occupancy after service initiation and the number of arrivals during that service, we readily obtain these pgfs as

$$\begin{aligned}
Q_{D,A}(z) =& Q_A(z)S_A(E(z)) \\
Q_{D,B}(z) =& Q_A(z)S_B(E(z)) \\
Q_D(z) =& Q_{D,A}(z) + Q_{D,B}(z) = Q_A(z)S_A(E(z)) + Q_B(z)S_B(E(z)) \ .
\end{aligned}$$

4.4.3. At customer arrival epochs

In this subsection, an expression is established for $Q_{Arr}(z)$, the steady-state pgf of the queue occupancy at customer arrival epochs. Since a customer arrives during a random slot, the server during this slot can be busy with an ongoing service, resulting in two different periods based on the class of the customers in service, or idle. In case that the server is idle in a random slot, we also need to keep track of the class of the previous service, which is equal to the class of the last customer to arrive, since the class of the next customer to arrive depends on its predecessor. This means that the system is in one of the four following phases:

- Service of a batch of class A is going on
- Service of a batch of class B is going on
- Server is idle and the previous batch was of class A
- Server is idle and the previous batch was of class B

Each service period of a class A (B) batch corresponds with a single A (B)-period, and all consecutive slots in which a slot is idle are grouped in a single (I, A)- or (I, B)-period based on the previous service. An example of how the time-axis is divided by these four types of periods is shown in Fig. 3.

The first step is to obtain the steady-state probability of the type of a randomly tagged phase,

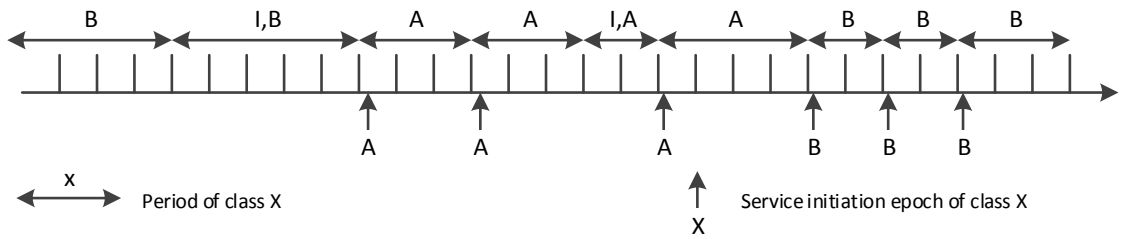


Figure 3: Sample of time-axis divided into periods.

that is compute

$$
\begin{aligned}
p_A &:= \lim_{k\to\infty} Pr[\tilde{t}_k = A] \ , \\
p_B &:= \lim_{k\to\infty} Pr[\tilde{t}_k = B] \ , \\
p_{I,A} &:= \lim_{k\to\infty} Pr[\tilde{t}_k = I, \tilde{t}_{k-1} = A] \ , \\
p_{I,B} &:= \lim_{k\to\infty} Pr[\tilde{t}_k = I, \tilde{t}_{k-1} = B] \ ,
\end{aligned}
$$

with $\tilde{t}_k$ the type of the k-th period. Crucial here is that the start of A- and B-periods are exactly the embedded points from Subsection 4.4.1, that is they correspond to the service initiation epochs. Consequently, we find the following expression for $p_{I,A}$ and $p_{I,B}$:

$$
\begin{aligned}
p_{I,A} =& \frac{q_A(0)S_A(E(0))}{1 + q_A(0)S_A(E(0)) + q_B(0)S_B(E(0))} \ , \\
p_{I,B} =& \frac{q_B(0)S_B(E(0))}{1 + q_A(0)S_A(E(0)) + q_B(0)S_B(E(0))} \ .
\end{aligned}
$$

Along the same lines we obtain

$$
\begin{aligned}
p_A =& \frac{Q_A(1)}{1 + q_A(0)S_A(E(0)) + q_B(0)S_B(E(0))} \ , \\
p_B =& \frac{Q_B(1)}{1 + q_A(0)S_A(E(0)) + q_B(0)S_B(E(0))} \ .
\end{aligned}
$$

The second step is to compute the steady-state probabilities π_A, π_B, $\pi_{I,A}$ and $\pi_{I,B}$ that a randomly tagged slot falls in an A-, B-, (I, A) or (I, B)-period respectively. These probabilities are the weighted average of the expected lengths of each period, with weights equal to the steady-state probabilities that a randomly tagged period is of the corresponding type. The expected lengths of an A- and B-period are equal to the averages of their respective service time distributions, and the length of an Idle period is equal to the expected number of consecutive slots with zero arrivals. Since the length of a sequence of slots without arrivals follows a shifted geometric distribution with parameter $E(0)$, which is the probability that there are no arrivals in a random slot, the mean

length of such a sequence is given by $\frac{1}{1-E(0)}$. Hence,

$$\begin{aligned}
\pi_A =& \frac{p_A S'_A(1)}{p_A S'_A(1) + p_B S'_B(1) + \frac{p_{I,A}+p_{I,B}}{1-E(0)}} \quad , \\
\pi_B =& \frac{p_B S'_B(1)}{p_A S'_A(1) + p_B S'_B(1) + \frac{p_{I,A}+p_{I,B}}{1-E(0)}} \quad , \\
\pi_{I,A} =& \frac{\frac{p_{I,A}}{1-E(0)}}{p_A S'_A(1) + p_B S'_B(1) + \frac{p_{I,A}+p_{I,B}}{1-E(0)}} \quad , \\
\pi_{I,B} =& \frac{\frac{p_{I,B}}{1-E(0)}}{p_A S'_A(1) + p_B S'_B(1) + \frac{p_{I,A}+p_{I,B}}{1-E(0)}} \quad .
\end{aligned} \tag{15}$$

We first define $Q_{Arr,I,A}(z)$ and $Q_{Arr,I,B}(z)$ as the partial pgfs of the queue occupancy of a random customer that arrives when the server is idle and the customer at the head of the queue is either of class A or B. Using the probabilities obtained in Eq. 15, we obtain

$$\begin{aligned}
Q_{Arr,I,A}(z) =& (\alpha\pi_{I,A} + (1-\beta)\pi_{I,B})\frac{1-E(z)}{\lambda(1-z)} \quad , \\
Q_{Arr,I,B}(z) =& ((1-\alpha)\pi_{I,A} + \beta\pi_{I,B})\frac{1-E(z)}{\lambda(1-z)} \quad .
\end{aligned} \tag{16}$$

On the other hand, if the random customer arrives in a slot during which a service was being processed, we both need the number of arrivals since initiation of the ongoing service and the remaining service time of the service period. Using the random variable $e_{i,k}$ as the number of arrivals before the random customer if the customer arrives in the i-th slot of the k-th service period, we can write the probability $r_A(i,j)$ that there are i arrivals, before the arrival of the random customer during a service of class A customers, and a remaining service time of j slots as

$$r_A(i,j) = \lim_{k\to\infty} \sum_{m=j+1}^{\infty} \frac{s_A(m) Pr[e_{m-j,k} = i]}{S'_A(1)} \quad . \tag{17}$$

Taking the z-transform of Eq. 17 leads to the joint pgf $R_A(z,x)$ of the number of arrivals during the service period before arrival of the random customer, and the remaining service time of the service period while the server was processing a class A batch, yielding

$$R_A(z,x) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} r_A(i,j) z^i x^j = \frac{1-E(z)}{\lambda(1-z)} \frac{S_A(x) - S_A(E(z))}{S'_A(1)(x-E(z))} \quad .$$

And analogously, if the random customer arrives while the server is processing a class B batch

$$R_B(z,x) = \frac{1-E(z)}{\lambda(1-z)} \frac{S_B(x) - S_B(E(z))}{S'_B(1)(x-E(z))} \quad .$$

Now we can calculate the joint pgf $Q_{Arr,A}(z,x)$ as the number of customers in the queue before arrival of the random customer, that arrives when the server is busy processing a class A batch,

and the remaining service time, for the case that the customer at the head of the queue is of class A as

$$Q_{Arr,A,A}(z,x) = \frac{\pi_A}{Q_A(1)} q_A(0)\alpha R_A(z,x) \ , \tag{18}$$

and if the customer at the head of the queue is of class B

$$Q_{Arr,A,B}(z,x) = \frac{\pi_A}{Q_A(1)} R_A(z,x)\left(Q_A(z) - \alpha q_A(0)\right) \ . \tag{19}$$

Analogously, we can also calculate the partial pgfs in case that the random customer arrives while the server was busy processing a class B batch, leading to

$$\begin{aligned} Q_{Arr,B,A}(z,x) =& \frac{\pi_B}{Q_B(1)} R_B(z,x)\left(Q_B(z) - \beta q_B(0)\right) \ , \\ Q_{Arr,B,B}(z,x) =& \frac{\pi_B}{Q_B(1)} q_B(0)\beta R_B(z,x) \ , \end{aligned} \tag{20}$$

By combining the results of Eqs. 16, 18 and 20, we can now calculate the partial pgf $Q_{Arr,A}(z,x)$ and $Q_{Arr,B}(z,x)$ of the queue occupancy and the remaining service time at customer arrival epochs, when the customer at the head of the queue is respectively a class A or B customer. We also define $Q_{Arr}(z,x)$ as the joint pgf of the queue occupancy and the remaining service time at customer arrival epochs. This results in

$$\begin{aligned} Q_{Arr,A}(z,x) =& \frac{1-E(z)}{\lambda S'_A(1)S'_B(1)(1-z)(x-E(z))}\Big(S'_A(1)S'_B(1)(x-E(z))(\alpha\pi_{I,A} + (1-\beta)\pi_{I,B}) \\ &+ \frac{\pi_A}{Q_A(1)} q_A(0)\alpha S'_B(1)(S_A(x) - S_A(E(z))) \\ &+ \frac{\pi_B}{Q_B(1)}(Q_B(z) - \beta q_B(0))S'_A(1)(S_B(x) - S_B(E(z)))\Big) \ , \\ Q_{Arr,B}(z,x) =& \frac{1-E(z)}{\lambda S'_A(1)S'_B(1)(1-z)(x-E(z))}\Big(S'_A(1)S'_B(1)(x-E(z))(\beta\pi_{I,B} + (1-\alpha)\pi_{I,A}) \\ &+ \frac{\pi_B}{Q_B(1)} q_B(0)\beta S'_A(1)(S_B(x) - S_B(E(z))) \\ &+ \frac{\pi_A}{Q_A(1)}(Q_A(z) - \alpha q_A(0))S'_B(1)(S_A(x) - S_A(E(z)))\Big) \ , \\ Q_{Arr}(z,x) =& Q_{Arr,A}(z,x) + Q_{Arr,B}(z,x) \ . \end{aligned} \tag{21}$$

4.4.4. At random slot boundaries

In the previous subsection, we calculated the probabilities that the server is idle or busy with a class A or B service in a randomly tagged slot, see Eq. 15. When a randomly tagged slot belongs to an I-period, the queue is empty at the beginning of the slot, otherwise a new service would have started. On the other hand, when the slot belongs to an A- or B-period, the queue contains the customers that are left behind at the corresponding service initiation epoch and the customers that have arrived during the elapsed service period, which is independent of the number of customers left behind in the queue, see Bruneel and Kim [45]. Combining all elements, we find that the pgf $Q_R(z)$ of the the queue occupancy at random slot boundaries is equal to

$$Q_R(z) := \pi_{I,A} + \pi_{I,B} + \pi_A \frac{Q_A(z)}{Q_A(1)} \frac{S_A(E(z)) - 1}{S'_A(1)(E(z)-1)} + \pi_B \frac{Q_B(z)}{Q_B(1)} \frac{S_B(E(z)) - 1}{S'_B(1)(E(z)-1)} \ .$$

4.5. Probability generating function of the real service capacity

A second important characteristic of this system is the real capacity of (i.e., the number of customers in) each batch service, which differs from the theoretical service capacity introduced in Section 4.1 in that it also depends on the number of customers in the queue and not only on the class of the batch. More precisely, it follows a geometric distribution, with parameter α or β if the customers are of class A or B, that is truncated by the number of customers in the queue. We define the steady-state probability that the server processes i customers during a class-A (or B) service and a random service as

$$\hat{c}_A(i) := \lim_{k\to+\infty} Pr[t_k = A, \hat{c}_k = i],\ \ \hat{c}_B(i) := \lim_{k\to+\infty} Pr[t_k = B, \hat{c}_k = i],\ \ \hat{c}(i) := \hat{c}_A(i) + \hat{c}_B(i)\ \ ,$$

for all $i \geq 0$ with corresponding partial pgfs $\hat{C}_A(z)$, $\hat{C}_B(z)$ and $\hat{C}(z) := \hat{C}_A(z) + \hat{C}_B(z)$. These pgfs can be calculated by evaluating the joint pgfs given in Eqs. 7 and 8 at $z = 1$. We find the following equation for the real service capacity of a class-A service:

$$\begin{aligned}
\hat{C}_A(x) =& P_A(1,x)\\
=& \alpha q_A(0)\Big(S_A(E(0))\Big(\frac{x-1}{\alpha x-1}\frac{E(\alpha x)-E(0)}{1-E(0)} - \frac{(1-\alpha)x}{\alpha x-1} - 1\Big) + \frac{x-1}{\alpha x-1}S_A(E(\alpha x)) - \frac{(1-\alpha)x}{\alpha x-1}\Big)\\
&+ q_B(0)\Big(S_B(E(0))\Big(\frac{(1-\beta)(x-1)}{\alpha x-1}\frac{E(\alpha x)-E(0)}{1-E(0)} - \frac{(1-\alpha)(1-\beta)x}{\alpha x-1} - (1-\beta)\Big)\\
&- \beta\frac{x-1}{\alpha x-1}S_B(E(\alpha x)) + \beta\frac{(1-\alpha)x}{\alpha x-1}\Big) + \frac{x-1}{\alpha x-1}S_B(E(\alpha x))P_B(\alpha x,1) - \frac{(1-\alpha)x}{\alpha x-1}P_B(1,1)\\
=& \frac{1}{1-\alpha x}\Big(\alpha q_A(0)\Big((1-\alpha)x + (1-x)S_A(E(0))(\frac{E(\alpha x)-E(0)}{1-E(0)} - 1) + (1-x)S_A(E(\alpha x))\Big)\\
&+ q_B(0)\Big((1-\beta)(1-x)S_B(E(0))(\frac{E(\alpha x)-E(0)}{1-E(0)} - 1) - \beta(1-x)S_B(E(\alpha x)) - \beta(1-\alpha)\Big)\\
&+ (1-x)S_B(E(\alpha x))P_B(\alpha x,1) + (1-\alpha)xP_B(1,1)\Big)\ \ .
\end{aligned}$$

In case of a class-B service, this becomes:

$$\begin{aligned}
\hat{C}_B(x) =& P_B(1,x)\\
=& \frac{1}{1-\beta x}\Big(\beta q_B(0)\Big((1-\beta)x + (1-x)S_B(E(0))(\frac{E(\beta x)-E(0)}{1-E(0)} - 1) + (1-x)S_B(E(\beta x))\Big)\\
&+ q_A(0)\Big((1-\alpha)(1-x)S_A(E(0))(\frac{E(\beta x)-E(0)}{1-E(0)} - 1) - \alpha(1-x)S_A(E(\beta x)) - \alpha(1-\beta)\Big)\\
&+ (1-x)S_A(E(\beta x))P_A(\beta x,1) + (1-\beta)xP_A(1,1)\Big)\ \ .
\end{aligned}$$

4.6. Probability generating function of the system occupancy

Taking into account the results of Section 4.3, we can also calculate the steady-state pgfs of the total number of customers in the system after service initiation of a class-A, B or random service. The number of customers in the system is the sum of the number of customers left in the queue

after service initiation and the real capacity of the server; note that these are not independent random variables.

We define the steady-state partial pgf of the total number of customers in the system after service initiation of a class $A(B)$ service as $U_A(z)$ ($U_B(z)$). These generating functions are found by evaluating Eqs. 7 and 8 for $x = z$. This gives the following formula for $U_A(z)$:

$$\begin{aligned} U_A(z) =& P_A(z,z) \\ =& \alpha q_A(0)\Big(S_A(E(z)) - S_A(E(0))\frac{1-E(z)}{1-E(0)}\Big) \\ &+ q_B(0)\Big(-\beta S_B(E(z)) - (1-\beta)S_B(E(0))\frac{1-E(z)}{1-E(0)}\Big) + S_B(E(z))P_B(z,1) \ , \end{aligned}$$

and the symmetric equation for a class-B service:

$$\begin{aligned} U_B(z) =& P_B(z,z) \\ =& \beta q_B(0)\Big(S_B(E(z)) - S_B(E(0))\frac{1-E(z)}{1-E(0)}\Big) \\ &+ q_A(0)\Big(-\alpha S_A(E(z)) - (1-\alpha)S_A(E(0))\frac{1-E(z)}{1-E(0)}\Big) + S_A(E(z))P_A(z,1) \ . \end{aligned}$$

The sum of these two formulas produces the steady-state pgf of the system occupancy after initiation of a random service. This leads to

$$\begin{aligned} U(z) =& S_B(E(z))P_B(z,1) - q_A(0)S_A(E(0))\frac{1-E(z)}{1-E(0)} \\ &+ S_B(E(z))P_A(z,1) - q_B(0)S_B(E(0))\frac{1-E(z)}{1-E(0)} \\ =& Q_D(z) - Q_D(0)\frac{1-E(z)}{1-E(0)} \ . \end{aligned}$$

This result can be understood probabilistically, since the system occupancy after a service initiation is the same as the queue occupancy after the departure of the previous service if the queue is not empty at a departure instant. If the queue was empty, the system occupancy is equal to the number of customers that arrive during the period while the server is idle.

The steady-state pgf of the system occupancy at departure epochs is identical to the pgf of the queue occupancy since the server is empty. The calculation for the steady-state pgf of the system occupancy at random slot boundaries is similar to the calculation described in section 4.4.4. The pgfs for the system occupancy at these two time instants are therefore given by

$$\begin{aligned} U_D(z) =& Q_D(z) \\ U_R(z) =& \pi_I + \pi_A \frac{U_A(z)}{U_A(1)}\frac{S_A(E(z))-1}{S'_A(1)(E(z)-1)} + \pi_B \frac{U_B(z)}{U_B(1)}\frac{S_B(E(z))-1}{S'_B(1)(E(z)-1)} \ . \end{aligned}$$

4.7. Delay analysis

We start the delay analysis by calculating the steady-state pgf of the delay (without residual service time) of a random customer that finds n customers in the queue on arrival and the first

customer is of class A and B, denoted by $D_{A,n}(z)$ and $D_{B,n}(z)$. Because the server is capable of grouping all consecutive same-class customers at the head of the queue, if the class of the random customer is equal to the class of the customer before it, then both customers will have the same delay. If this is not the case, then the delay of a random customer will be equal to the sum of the delay of its predecessor and the duration of a single service period. We also note that when the random customer finds an empty queue on arrival, the customer will be served first since we do not include the residual service time in these calculations. Summarising, we obtain

$$\begin{aligned}
D_{A,0}(z) =& S_A(z) \\
D_{B,0}(z) =& S_B(z) \\
\begin{bmatrix} D_{A,n}(z) \\ D_{B,n}(z) \end{bmatrix} =& \begin{bmatrix} \alpha & (1-\alpha)S_A(z) \\ (1-\beta)S_B(z) & \beta \end{bmatrix} \begin{bmatrix} D_{A,n-1}(z) \\ D_{B,n-1}(z) \end{bmatrix} \\
=& \begin{bmatrix} \alpha & (1-\alpha)S_A(z) \\ (1-\beta)S_B(z) & \beta \end{bmatrix}^n \begin{bmatrix} S_A(z) \\ S_B(z) \end{bmatrix} = \boldsymbol{M}(z)^n \begin{bmatrix} S_A(z) \\ S_B(z) \end{bmatrix} \quad .
\end{aligned} \tag{22}$$

The eigenvalues $\lambda_1(z)$ and $\lambda_2(z)$ of $\boldsymbol{M}(z)$ are given by the expression:

$$\lambda_{1,2}(z) = \frac{\alpha+\beta}{2} \pm \frac{1}{2}\sqrt{(\alpha-\beta)^2 + 4(1-\alpha)(1-\beta)S_A(z)S_B(z)} \ , \tag{23}$$

and the matrices of the right and left eigenvectors of $\boldsymbol{M}(z)$, denoted respectively by $\boldsymbol{R}(z)$ and $\boldsymbol{L}(z)$, are

$$\begin{aligned}
\boldsymbol{R}(z) =& \begin{bmatrix} \frac{(1-\alpha)S_A(z)}{\lambda_1(z)-\alpha} & \frac{(1-\alpha)S_A(z)}{\lambda_2(z)-\alpha} \\ 1 & 1 \end{bmatrix} =: \begin{bmatrix} r_1(z) & r_2(z) \\ 1 & 1 \end{bmatrix} \ , \\
\boldsymbol{L}(z) =& \boldsymbol{R}^{-1}(z) = \begin{bmatrix} \frac{(1-\beta)S_B(z)}{2\lambda_1(z)-\alpha-\beta} & \frac{\lambda_1(z)-\alpha}{2\lambda_1(z)-\alpha-\beta} \\ \frac{(1-\beta)S_B(z)}{2\lambda_2(z)-\alpha-\beta} & \frac{\lambda_2(z)-\alpha}{2\lambda_2(z)-\alpha-\beta} \end{bmatrix} \quad .
\end{aligned} \tag{24}$$

Using these results, we can diagonalize the matrix $\boldsymbol{M}(z)$ in Eq. 22, leading to

$$\begin{bmatrix} D_{A,n}(z) \\ D_{B,n}(z) \end{bmatrix} = \boldsymbol{R}(z) \begin{bmatrix} \lambda_1(z)^n & 0 \\ 0 & \lambda_2(z)^n \end{bmatrix} \boldsymbol{L}(z) \begin{bmatrix} S_A(z) \\ S_B(z) \end{bmatrix} \quad . \tag{25}$$

We also define $L_1(z)$ and $L_2(z)$ as

$$L_{1,2}(z) = \frac{(1-\beta)S_A(z)S_B(z) + (\lambda_{1,2}(z)-\alpha)S_B(z)}{2\lambda_{1,2}(z)-\alpha-\beta} \quad . \tag{26}$$

The delay we calculated in Eq. 25 did not take into account the phase of the server. Lets start by defining the delay of a random customer that arrives during an idle slot of the server and the first customer is either of class A or B, denoted by $D_{I,A}(z)$ and $D_{I,B}(z)$. In this case, a new service is initiated at the start of the next slot which means that there is no residual service time which could lead to an increased delay. Using $q_{Arr,I,A}$ as the random variable of the customers before the random customer on arrival and the customer at the head of the queue is of class A, these partial

pgfs of the delay are equal to

$$\begin{aligned}
D_{I,A}(z) =& \sum_{n=0}^{\infty} \begin{bmatrix} Pr[q_{Arr,I,A} = n] & 0 \end{bmatrix} \boldsymbol{R}(z) \begin{bmatrix} \lambda_1(z)^n & 0 \\ 0 & \lambda_2(z)^n \end{bmatrix} \begin{bmatrix} L_1(z) \\ L_2(z) \end{bmatrix} \\
=& r_1(z) Q_{Arr,I,A}(\lambda_1(z)) L_1(z) + r_2(z) Q_{Arr,I,A}(\lambda_2(z)) L_2(z) \ , \\
D_{I,B}(z) =& Q_{Arr,I,B}(\lambda_1(z)) L_1(z) + Q_{Arr,I,B}(\lambda_2(z)) L_2(z) \ . \qquad (27)
\end{aligned}$$

On the other hand, if the server is processing a class A batch on arrival of the random customer, we also need to incorporate the remaining service time. The partial pgfs in case that the server is busy processing a class A batch and the customer at the head of the queue is of class A or B, denoted by $D_{A,A}(z)$ and $D_{A,B}(z)$, are

$$\begin{aligned}
D_{A,A}(z) =& r_1(z) Q_{Arr,A,A}(\lambda_1(z), z) L_1(z) + r_2(z) Q_{Arr,A,A}(\lambda_2(z), z) L_2(z) \ , \\
D_{A,B}(z) =& Q_{Arr,A,B}(\lambda_1(z), z) L_1(z) + Q_{Arr,A,B}(\lambda_2(z), z) L_2(z) \ . \qquad (28)
\end{aligned}$$

and analogously in case that the customer arrives while the server is busy with a class B batch

$$\begin{aligned}
D_{B,A}(z) =& r_1(z) Q_{Arr,B,A}(\lambda_1(z), z) L_1(z) + r_2(z) Q_{Arr,B,A}(\lambda_2(z), z) L_2(z) \ , \\
D_{B,B}(z) =& Q_{Arr,B,B}(\lambda_1(z), z) L_1(z) + Q_{Arr,B,B}(\lambda_2(z), z) L_2(z) \ . \qquad (29)
\end{aligned}$$

Combining all partial pgfs of Eqs. 27, 28 and 29 results in the generating function of the delay of a completely random customer. This pgf, denoted by $D(z)$, can be written as

$$\begin{aligned}
D(z) =& \Big(r_1(z) Q_{Arr,A}(\lambda_1(z), z) + Q_{Arr,B}(\lambda_1(z), z) \Big) L_1(z) \\
&+ \Big(r_2(z) Q_{Arr,A}(\lambda_2(z), z) + Q_{Arr,B}(\lambda_2(z), z) \Big) L_2(z) \ .
\end{aligned}$$

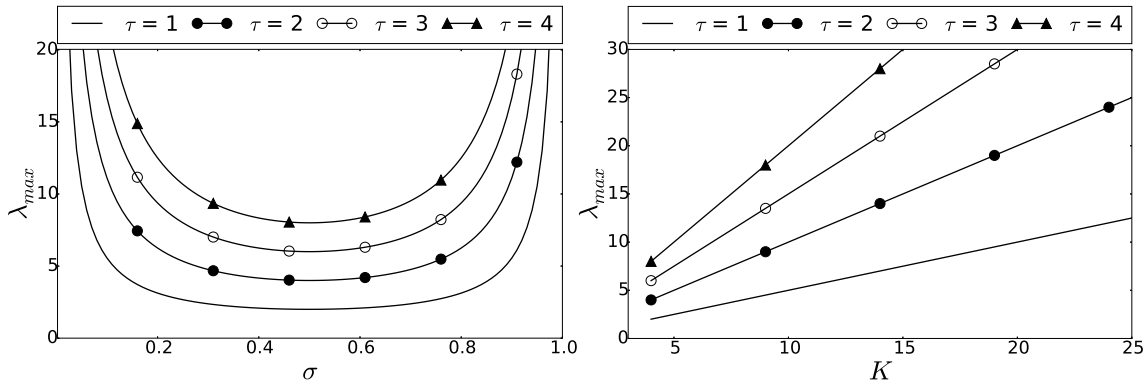
5. Discussion of results and numerical examples

In this section, we illustrate the obtained results throughout the previous sections by looking at some numerical examples. The considered performance measures are the maximum allowed arrival rate, the mean queue occupancy at different time instants, the average number of customers in a served batch, and the mean delay of a random customer. These are readily obtained by taking the first-order derivative of the respective steady-state pgfs with respect to z (or x) at $z = 1$ (or $x = 1$). Although the calculations are straightforward, they can become quite tedious, and lead to extensive formulas that are not repeated here. Obviously, higher-order moments such as the variance, can be obtained in a similar manner by considering higher-order derivatives. These higher-order moments will not be shown here. In all these examples we consider a geometric arrival process with $E(z) = \frac{1}{1+\lambda(1-z)}$, and use either deterministic or geometrically distributed service times.

We first consider the maximum allowed arrival rate which is defined by

$$\lambda_{max} := \frac{\tau}{\sigma(1-\sigma)} \frac{1}{(S'_A(1) + S'_B(1))} = \frac{\tau K}{S'_A(1) + S'_B(1)} \ , \qquad (30)$$

and in view of Eq. 4, the equilibrium condition requires that $\lambda < \lambda_{max}$. In this example, we set the service time of a class-A or B batch always to be equal to 1 slot, implying that $S_A(z) = S_B(z) = z$



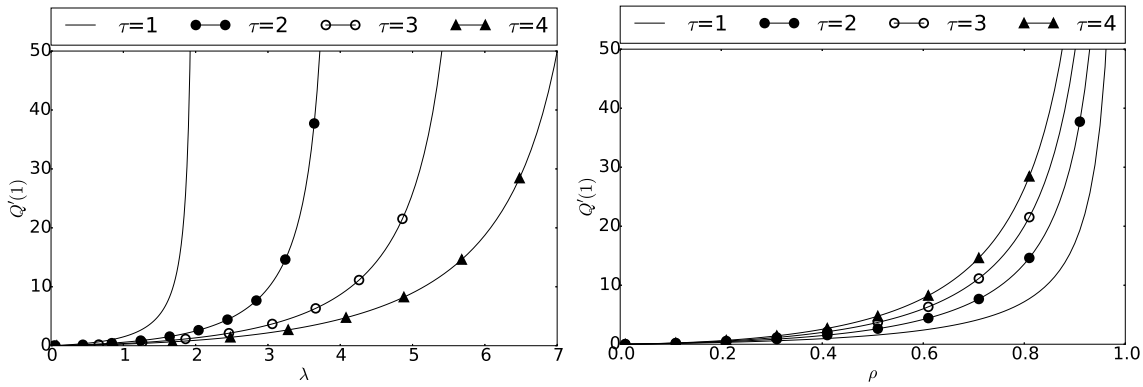
(a) Versus σ (b) Versus K

Figure 4: Maximum allowed arrival rate, for fixed 1-slot service times for class A and B, $\tau = 1, 2, 3, 4$, as a function of σ(a) and K(b).

and $S'_A(1) = S'_B(1) = 1$. Fig. 4 shows λ_{max} as a function of σ and K, for values of τ as indicated. In previous studies (see Bruneel et al. [33, 34], Mélange et al. [35], Maertens et al. [36], Reveil et al. [37]) it was observed that higher levels of correlation lead to a lower performance of the system. However, in this system, the average length of the sequences of same-class customers increases with a higher level of correlation in the arrival process (i.e., a higher value of τ), which leads to a higher λ_{max}. This indicates that more correlation between same-class customers will lead to a better performance. This is clearly visible on the figure, since larger values of τ lead to a system that can handle a higher average arrival rate.

We also observe that there is a global minimum in $\sigma = 0.5$ (or $K = 4$) for all values of τ, meaning that a symmetric system is a worst-case situation. This can also easily be derived from Eq. 4. An intuitive explanation is that the more asymmetric the system (larger K or σ further from 0.5), the larger the batches of one class in comparison to the other, or the larger the probability of "back-to-back" arrivals for one class. Since the average length of same-class sequences is inversely proportional to α and β, more asymmetry in a system will lead to a better performance. In the following results we will use K instead of σ, since K gives a global indication of the asymmetry, while σ is a parameter of a single class. Note that the expression for λ_{max} shows that this quantity is also a linearly increasing function of τ for fixed values of K, which is why we have not plotted this dependency.

In Fig. 5a, we have plotted the mean queue occupancy after service initiation epochs versus the mean arrival rate for a symmetric system ($K = 4$) and multiple values of $\tau = 1, 2, 3, 4$. The influence of the correlation variable τ is clearly linearly increasing in the mean number of customers that must arrive to obtain the same average queue occupancy. Let us also define the load ρ of the system as $\rho := \frac{\lambda}{\lambda_{max}}$; obviously, $\rho < 1$ represents the equilibrium condition as well. The influence of τ on the average queue occupancy as a function of the load is shown in Fig. 5b. For small loads, the queue will often be empty after service initiation, which means that the typically small number of customers in the queue is the limiting factor of the performance of the system. Consequently, we observe in Fig. 5b that the influence of the correlation in the arrival process is negligible for small loads. For increasing loads, it occurs more frequently that the performance is governed by the real service capacity of the server, which does depend on the correlation in the arrival process.



(a) Versus the mean arrival rate λ

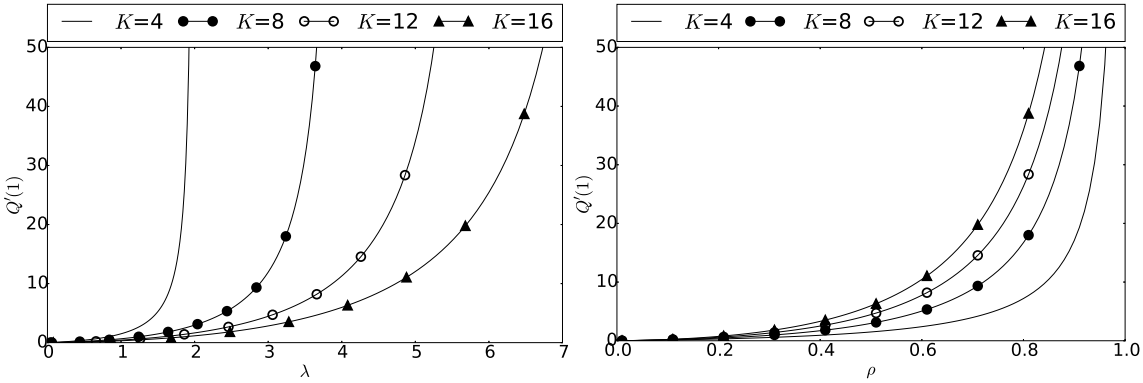
(b) Versus the load ρ

Figure 5: Mean system occupancy $Q'(1)$ for fixed 1-slot service times for class A and B services, with $K = 4$ and $\tau = 1, 2, 3, 4$, as a function of λ(a) and ρ(b).

As opposed to Fig. 5a, we now observe that increasing values of τ have a negative impact on the system performance. This is caused by the higher mean number of customers that must arrive to obtain the same value of ρ for increasing τ. The result of this effect is that for higher values of τ, more customers will be present in the queue (on average) after service initiation epochs.

The impact of asymmetry in the arrival process (captured by the parameter K) can be seen in Fig. 6. These results were obtained for $\tau = 1$. By comparing the Figs. 5 and 6, we clearly observe that asymmetry in the arrival process has a similar impact as correlation in the arrival process, and the above qualitative conclusions therefore remain valid. From Figs. 5b and 6b we can deduce that the impact of K is somewhat larger than that of τ. This is because, in asymmetric systems, a larger batch and a smaller batch alternate. After service initiation of the smaller batch, a typically large group of customers will be in the system in order to be able to create the larger batch, which leads to a higher average number of customers in the queue after service initiation.

In Fig. 7, we show the average queue occupancy after service initiation epochs, at departure epochs, and at random slot boundaries, as a function of the mean arrival rate λ and load ρ. It is important to note that the results that are discussed here are only valid for fixed deterministic service times of a single slot for both classes. As expected, the mean queue occupancy is largest at departure epochs, since the queue has been building up further during the preceding batch service time. Therefore, the difference between the mean queue occupancy at service initiation and departure epochs is equal to the average number of arrivals during a single slot, since we consider fixed single-slot service times for both classes. We expect that this difference becomes larger for larger average batch service times. We can also conclude that the average queue occupancy at random slot boundaries is typically smaller than the queue occupancy after service initiation epochs. This is another consequence of the single slot service times. A random slot boundary is either the boundary of an idle slot, or a service initiation epoch, which leads to a small difference between the average queue occupancy at the two time instances, determined by the probability that a random slot is an idle slot. At low arrival rates, the average queue occupancy in both cases is almost zero, but the relative difference is highest, because there is a high probability that a random slot is an idle slot. This probability decreases with increasing arrival rates, which implies that the average queue occupancy at random slot boundaries will approach the queue occupancy after service initiation.



(a) Versus the mean arrival rate λ

(b) Versus the load ρ

Figure 6: Mean system occupancy $Q'(1)$ for fixed 1-slot service times for class A and B services, with $\tau = 1$ and $K = 4, 8, 12, 16$, as a function of λ(a) and ρ(b).

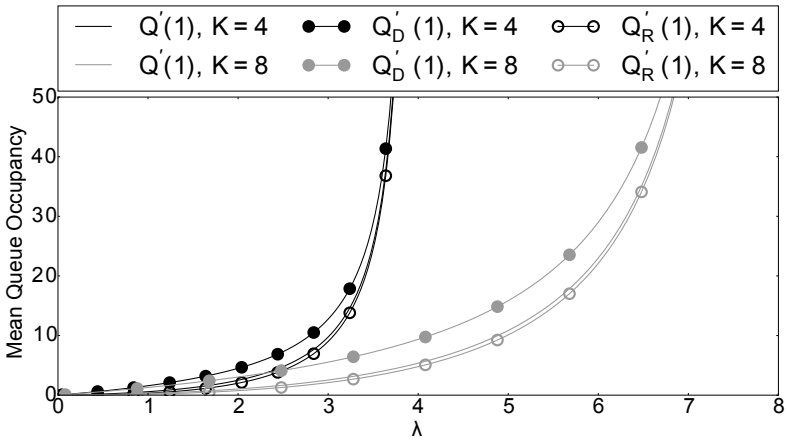
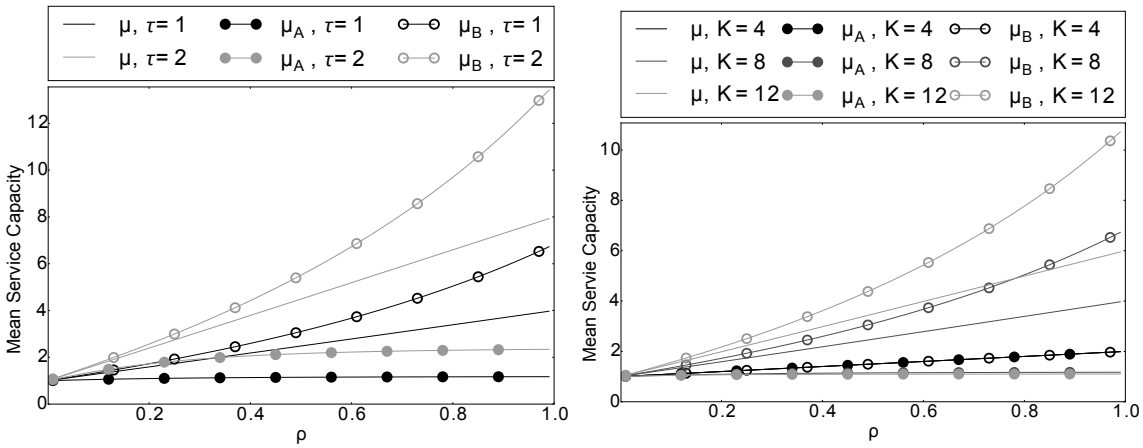


Figure 7: Mean queue occupancy after service initiation epochs, departure epochs and random slot boundaries for fixed 1-slot service times for class A and B, with $\tau = 2$ and $K = 4, 8$ as a function of the arrival rate λ.



(a) $K = 8$ and $\tau = 1, 2$

(b) $\tau = 1$ and $K = 4, 8, 12$

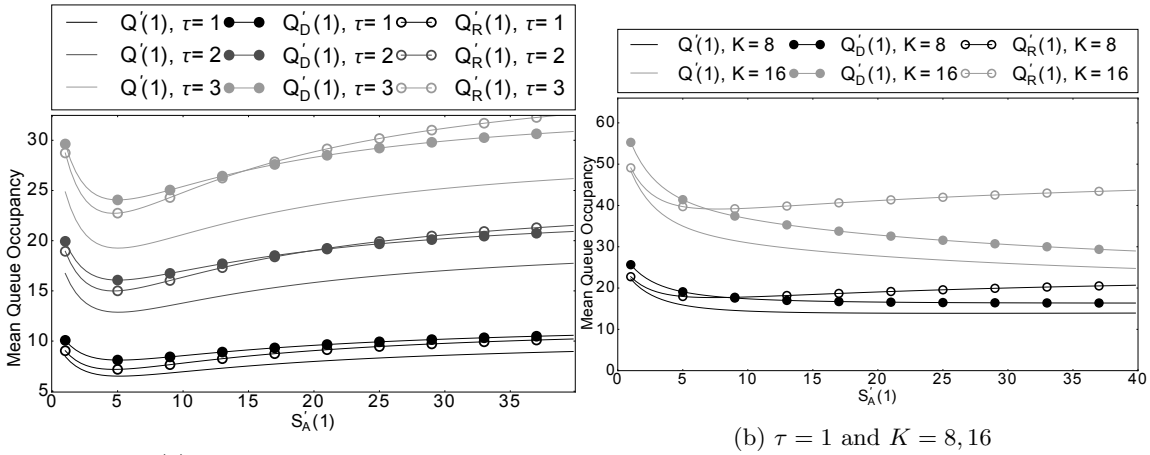
Figure 8: Mean real service capacity of a class A (μ_A), B (μ_B) or random service (μ), for fixed 1-slot service times of class A and B services, as a function of the load ρ for $\tau = 1, 2$ and $K = 8$(a) or $\tau = 1$ and $K = 4, 8, 12$(b).

Another important characteristic of the system, which is influenced by both τ and K, is the average number of customers that are being processed in a single service, also called the average real service capacity of the server. Fig. 8 shows the average real service capacity of a class A service, a class B service and a random service given respectively by μ_A, μ_B and μ. First, we observe that μ is nearly linearly proportional to the load. We also see that the real service capacity is close to 1 for all combinations of τ and K, and small loads. This is what we expected since the probability that more than one customer arrives during a single slot is negligible under these circumstances. On the other hand, for values of ρ close to 1, the queue becomes saturated, and the average real service capacity of a random service becomes equal to the average theoretical service capacity, which is

$$\mu = \frac{1}{2}\left(\frac{1}{1-\alpha} + \frac{1}{1-\beta}\right) = \frac{\tau K}{2} \quad .$$

These are indeed the values that we observe in these figures in case of a saturated queue, which, amongst others, confirms the calculations of the preceding sections. Fig. 8a and Fig. 8b show the influence of the parameters τ and K on the average real service capacities. We clearly see that the average real service capacities of both class A and B services are linearly proportional to τ. The influence of K can be split in two cases. If the arrival process is symmetric ($K = 4$), the average real service capacities of class A and B services are equal to each other which leads to $\mu = \mu_A = \mu_B$. In case of an asymmetric arrival process ($K > 4$), we see that the parameter K only has a significant impact on the mean real service capacity of a class B customer. This is caused by our implementation of an asymmetric system which leads to $\beta > \alpha$.

Let us also consider a model where the distributions of class A and B service times are geometric. In the following experiments, we let the average service time of a class A service vary from 1 slot to 40 slots while we keep the average service time of a class B service at 5 slots. Fig. 9a shows the influence of τ on a symmetric system with a load of 0.8 in terms of the average service time of a class A service. We note that there is a minimum for all values of τ and for the three chosen time instances at $S'_A(1) = 5$, which is as expected since the arrival process is symmetric. We also note



(a) $K = 4$ and $\tau = 1, 2, 3$

(b) $\tau = 1$ and $K = 8, 16$

Figure 9: The mean queue occupancy at random slot boundaries ($Q'_R(1)$), service initiation ($Q'(1)$), and departure epochs ($Q'_D(1)$) in function of the mean class A service time $S'_A(1)$, which is geometrically distributed, for the load $\rho = 0.8$, mean class B service time $S'_B(1) = 5$ and $\tau = 1, 2, 3$ and $K = 4$(a) or $\tau = 1$ and $K = 8, 16$(b).

that, for $\tau = 1$, the average queue occupancy at random slot boundaries is always smaller than at departure epochs, while this is not the case for larger values of τ. The ratio $\frac{S'_A(1)}{S'_B(1)}$, for which the average queue occupancy at random slot boundaries and at departure epochs are equal, is influenced by τ. Larger values of τ shift this ratio closer to 1. As in the results of the average real service capacities, symmetric and asymmetric arrival processes lead to a significantly different behaviour. A symmetric system, as shown in Fig. 9a, has a minimum at the three observed time instances while an asymmetric system only has a minimum for the average queue occupancy at random slot boundaries, see Fig. 9b. We clearly see that the average queue occupancy after service initiation epochs and departure epochs are monotone decreasing functions of $S'_A(1)$. We note that this change in behaviour has a significant impact on the point where the average queue occupancy at random slot boundaries and at service departure epochs are equal. Larger values of the parameter K cause this point to shift to $S'_B(1)$ and the minimum occurs at a larger value of $S'_A(1)$.

In the previous plots we used a load of 0.8. In Fig. 10 we observe the average queue occupancy after service initiation epochs, departure epochs and random slot boundaries with $\tau = 2$ and $K = 4$ for $\rho = 0.4$ and $\rho = 0.6$ as a function of the average service time of a class A service. We see that, while the behaviour of these two loads is similar to the results with $\rho = 0.8$, there is a difference in the point where the average queue occupancy at random slot boundaries and at departure epochs are equal. Smaller loads mean this point occurs at a significantly larger $S'_A(1)$.

To conclude the section on the numerical experiments, we take a look at the impact of correlation in the arrival process on the mean queue occupancy at service initiation epochs and the mean delay of a random customer for a number of different values of the mean arrival rate λ, see Fig. 11, and of the mean load ρ of the system, see Fig. 12. In these figures, we used a symmetric arrival process with $K = 4$. Similar figures for $Q'(1)$ and $D'(1)$ in function of the degree of asymmetry K in the arrival process were also considered. They show similar behaviour as Figs. 11 and 12, and are therefore not shown here. In these figures, the service times for batches of both classes follow a geometric distribution with a mean of 3 slots. In Fig. 11, we clearly see that, for a constant

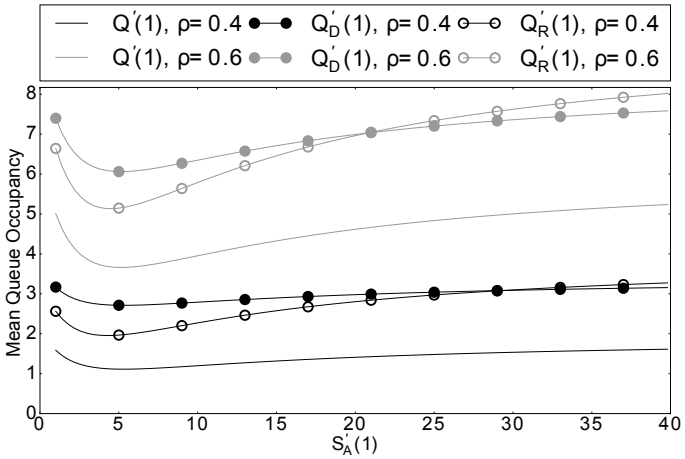
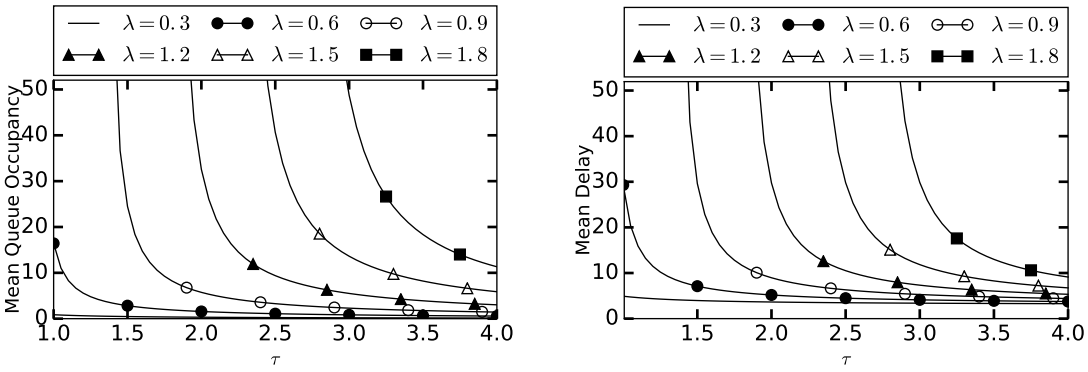


Figure 10: The mean queue occupancy at service initiation, service departure epochs and random slot boundaries in function of the mean class A service time $S'_A(1)$, which is geometrically distributed, for $\tau = 2$, $K = 4$, $S'_B(1) = 5$ and $\rho = 0.4, 0.6$.

arrival rate λ, both the mean queue occupancy and mean delay are decreased significantly when τ, the degree of correlation between the classes of consecutive customers, increases. We also observe that the mean queue occupancy $Q'(1)$ is smaller than the mean delay $D'(1)$ when τ is small but the opposite, namely $D'(1) > Q'(1)$, is true when there is a high degree of correlation in the arrival process. The reason for this is that an increasing correlation leads to a higher mean real service capacity resulting in a decrease of the delay for a fixed number of customers in the queue on arrival, which means that even if the delay decreases, the delay will decrease faster resulting in a higher mean queue occupancy for high values of τ. The behaviour of this inequality can also be observed in the same figures but for a number of different loads in the system, see Fig. 12. As shown in Fig. 5b, we see that increasing the degree of correlation also increases the mean queue occupancy for a system under a constant load but that the mean delay of a random customer remains almost constant for increasing degrees of correlation. This occurs because, while the mean number of customers in the queue on arrival of a customer will also be larger, a higher value of τ will mean that the server will be able to form larger groups of customers. These two effects of increasing mean queue occupancy and mean real service capacity, almost cancel each other out.

6. Conclusions

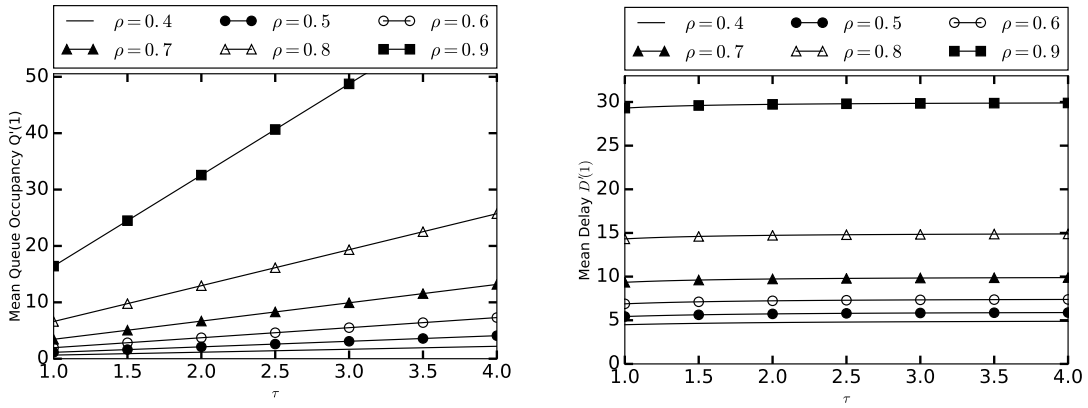
In this paper, we have deduced an expression for the joint pgf of the queue occupancy and the size of the batch in service at service initiation epochs, for the discrete-time two-class single-server queueing system with variable capacity batch service, correlated customer types and generally distributed class-dependent service times. From this joint pgf, we have deduced the pgfs of the queue and system occupancy at various time instances and the pgf of the variable service capacity. In the last part of the analysis, we focused on the delay of a random customer. Using these results, we have demonstrated the impact of asymmetry and correlation between the classes of consecutive customers in the arrival process on the performance of the system. Also, we investigated the impact of differences between the service processes of both classes of batches.



(a) Mean Queue Occupancy

(b) Mean Delay

Figure 11: Mean queue occupancy(a) at service initiation epochs and mean delay(b) of a random customer in function of the degree of correlation τ in the arrival process for a number of different arrival rates and geometrically distributed service times with mean of 3 slots.



(a) Mean Queue Occupancy

(b) Mean Delay

Figure 12: Mean queue occupancy(a) at service initiation epochs and mean delay(b) of a random customer in function of the degree of correlation τ in the arrival process for a number of different loads and geometrically distributed service times with a mean of 3 slots.

There are a number of possible extensions that could be considered for this system. A first extension would be to include a switch-over time to account for certain changes that must be made to the system if the server switches between classes, such as increasing or decreasing the temperature of a furnace or changing the colour of the used paint. Secondly, we can include an upper bound on the variable capacity of the server, which depends on the class of the service. While this will lead to a more realistic model, we expect that the model in this paper will be a good approximation in systems where the load is not too high and the probability that a random customer is of either class is non-negligible. If this is the case, the variable service capacity is already limited respectively by the number of customers in the queue or the length of the sequence of same-class customers at the head of the queue. Lastly, a system with a general number of customer classes can also be taken into consideration.

References

[1] A. Gupta, A. Sivakumar, Optimization of due-date objectives in scheduling semiconductor batch manufacturing, International Journal of Machine Tools and Manufacture 46 (12) (2006) 1671–1679.

[2] A. Gupta, A. Sivakumar, Controlling delivery performance in semiconductor manufacturing using look ahead batching, International Journal of Production Research 45 (3) (2007) 591–613.

[3] L. Tadj, G. Choudhury, C. Tadj, A quorum queueing system with a random setup time under N-policy and with bernoulli vacation schedule, Quality Technology & Quantitative Management 3 (2) (2006) 145–160.

[4] A. Janssen, J. van Leeuwaarden, Analytic computation schemes for the discrete-time bulk service queue, Queueing Systems 50 (2–3) (2005) 141–163.

[5] R. Arumuganathan, S. Jeyakumar, Steady state analysis of a bulk queue with multiple vacations, setup times with N-policy and closedown times, Applied Mathematical Modelling 29 (10) (2005) 972–986.

[6] A. Banerjee, U. Gupta, Reducing congestion in bulk-service finite-buffer queueing system using batch-size-dependent service, Performance Evaluation 69 (1) (2012) 53–70.

[7] A. Banerjee, U. Gupta, S. Chakravarthy, Analysis of a finite-buffer bulk-service queue under markovian arrival process with batch-size-dependent service, Computers and Operations Research 60 (2015) 138–149.

[8] A. Banerjee, U. Gupta, V. Goswami, Analysis of finite-buffer discrete-time batch-service queue with batch-size-dependent service, Computers and Industrial Engineering 75 (2014) 121–128.

[9] S. Chang, T. Takine, Factorization and stochastic decomposition properties in bulk queues with generalized vacations, Queueing Systems 50 (2–3) (2005) 165–183.

[10] M. Chaudhry, J. Templeton, A first course in bulk queues, Wiley New York, 1983.

[11] D. Claeys, B. Steyaert, J. Walraevens, K. Laevens, H. Bruneel, Tail distribution of the delay in a general batch-service queueing model, Computers and Operations Research 39 (11) (2012) 2733–2741.

[12] D. Claeys, B. Steyaert, J. Walraevens, K. Laevens, H. Bruneel, Analysis of a versatile batch-service queueing model with correlation in the arrival process, Performance Evaluation 70 (4) (2013) 300–316.

[13] D. Claeys, B. Steyaert, J. Walraevens, K. Laevens, H. Bruneel, Tail probabilities of the delay in a batch-service queueing model with batch-size dependent service times and a timer mechanism, Computers and Operations Research 40 (5) (2013) 1497–1505.

[14] D. Claeys, J. Walraevens, K. Laevens, H. Bruneel, Analysis of threshold-based batch-service queueing systems with batch arrivals and general service times, Performance Evaluation 68 (6) (2011) 528–549.

[15] V. Goswami, J. Mohanty, S. Samanta, Discrete-time bulk-service queues with accessible and non-accessible batches, Applied Mathematics and Computation 182 (1) (2006) 898–906.

[16] H. Olbert, M. Protopappa-Sieke, U. Thonemann, Analyzing the effect of express orders on supply chain costs and delivery times, Production and Operations Management 25 (12) (2016) 2035–2050.

[17] S. Pradhan, U. Gupta, Modeling and analysis of an infinite-buffer batch-arrival queue with batch-size-dependent service: $M^X/G_n^{(a,b)}/1$, Performance Evaluation 108 (2017) 16–31.

[18] H. Wang, A. Odoni, Approximating the performance of a "Last Mile" transportation system, Transportation Science 50 (2) (2014) 659–675.

[19] M. Chaudhry, S. Chang, Analysis of the discrete-time bulk-service queue $Geo/G^Y/1/N+B$, Operations Research Letters 32 (4) (2004) 355–363.

[20] S. Chang, D. Choi, Performance analysis of a finite-buffer discrete-time queue with bulk arrival, bulk service and vacations, Computers and Operations Research 32 (9) (2005) 2213–2234.

[21] K. Sikdar, U. Gupta, On the batch arrival batch service queue with finite buffer under servers vacation: $M^X/G^Y/1/N$ queue, Computers & Mathematics with Applications 56 (11) (2008) 2861–2873.

[22] K. Sikdar, S. Samanta, Analysis of a finite buffer variable batch service queue with batch Markovian arrival process and servers vacation, Journal of the Operational Research Society of India 53 (3) (2016) 553–583.

[23] X. Yi, N. Kim, B. Yoon, K. Chae, Analysis of the queue-length distribution for the discrete-time batch-service $Geo/G^{a,Y}/1/K$ queue, European Journal of Operational Research 181 (2) (2007) 787–792.

[24] S. Pradhan, U. Gupta, S. Samanta, Queue-length distribution of a batch service queue with random capacity and batch size dependent service: $M/G_r^Y/1$, Journal of the Operational Research Society of India 53 (2) (2015) 1–15.

[25] R. Germs, N. V. Foreest, Loss probabilities for the $M^X/G^Y/1/K+B$ queue, Probability in the Engineering and Informational Sciences 24 (4) (2010) 457–471.

[26] R. Germs, N. V. Foreest, Analysis of finite-buffer state-dependent bulk queues, OR Spectrum 35 (3) (2013) 563–583.

[27] G. Reddy, R. Nadarajan, P. Kandasamy, A nonpreemptive priority multiserver queueing system with general bulk service and heterogeneous arrivals, Computers and operations research 20 (4) (1993) 447–453.

[28] Z. Zhao, C. Yi, J. Cai, H. Cao, Queueing analysis for medical data transmissions with delay-dependent packet priorities in WBANs, in: Wireless Communications & Signal Processing (WCSP), 2016 8th International Conference on, IEEE, 2016, pp. 1–5.

[29] J. Walraevens, H. Bruneel, D. Fiems, S. Wittevrongel, Delay analysis of multiclass queues with correlated train arrivals and a hybrid priority/FIFO scheduling discipline, Applied Mathematical Modelling 45 (2017) 823–839.

[30] O. Boxma, J. van der Wal, U. Yechiali, Polling with batch service, Stochastic Models 24 (4) (2008) 604–625.

[31] J. Dorsman, R. Van der Mei, E. Winands, Polling systems with batch service, OR spectrum 34 (3) (2012) 743–761.

[32] J. Fowler, N. Phojanamongkolkij, J. Cochran, D. Montgomery, Optimal batching in a wafer fabrication facility using a multiproduct G/G/C model with batch processing, International Journal of Production Research 40 (2) (2002) 275–292.

[33] H. Bruneel, W. Mélange, B. Steyaert, D. Claeys, J. Walraevens, A two-class discrete-time queueing model with two dedicated servers and global FCFS service discipline, European Journal of Operational Research 223 (1) (2012) 123–132.

[34] H. Bruneel, W. Mélange, B. Steyaert, D. Claeys, J. Walraevens, Effect of global FCFS and relative load distribution in two-class queues with dedicated servers, Quartely Journal of Operations Research 11 (4) (2013) 375–391.

[35] W. Mélange, H. Bruneel, B. Steyaert, D. Claeys, J. Walraevens, A continuous-time queueing model with class clustering and global FCFS service discipline, Journal of Industrial Management and Optimization 10 (1) (2014) 193–206.

[36] T. Maertens, H. Bruneel, J. Walraevens, Effect of class clustering on delay differentiation in priority scheduling, Electronic Letters 48 (10) (2012) 568–569.

[37] B. Réveil, D. Claeys, T. Maertens, J. Walraevens, H. Bruneel, Impact of class clustering in a multiclass FCFS queue with order-dependent service times, Computers & Operations Research 51 (2014) 90–98.

[38] J. Baetens, B. Steyaert, D. Claeys, H. Bruneel, System occupancy of a two-class batch-service queue with class-dependent variable server capacity, in: International Conference on Analytical and Stochastic Modeling Techniques and Applications, Springer, 2016, pp. 32–44.

[39] J. Baetens, B. Steyaert, D. Claeys, H. Bruneel, Delay analysis of a two-class batch-service queue with class-dependent variable server capacity, Mathematical Methods of Operations Research 88 (1) (2018) 37–57.

[40] J. Baetens, D. Claeys, B. Steyaert, H. Bruneel, System performance of a variable-capacity batch-service queue with geometric service times and customer-based correlation, in: 31st European Conference on Modelling and Simulation, ECMS 2017, Vol. 31, 2017, pp. 649–655.

[41] J. Baetens, B. Steyaert, D. Claeys, H. Bruneel, Delay analysis of a variable-capacity batch-server queue with general class-dependent service times, in: AIP Conference Proceedings, Vol. 1978, AIP Publishing, 2018, p. 190003.

[42] K. J. Duggan, Creating mixed model value streams: practical lean techniques for building to demand, Productivity Press, 2012.

[43] M. Rother, J. Shook, Learning to see: value stream mapping to add value and eliminate muda, Lean Enterprise Institute, 2003.

[44] S. M. Ross, Introduction to probability models, Academic press, 2014.

[45] H. Bruneel, B. Kim, Discrete-time models for communication systems including ATM, Kluwer Academic, Boston, USA, 1993.

[46] F. Baccelli, S. Foss, On the saturation rule for the stability of queues, Journal of Applied Probability 32 (2) (1995) 494–507.

[47] P. Kuehn, Multiqueue systems with nonexhaustive cyclic service, Bell System Technical Journal 58 (3) (1979) 671–698.

[48] S. Foss, N. Chernova, A. Kovalevskii, Stability of polling systems with state-independent routing, in: PROCEEDINGS OF THE ANNUAL ALLERTON CONFERENCE ON COMMUNICATION CONTROL AND COMPUTING, Vol. 34, Citeseer, 1996, pp. 220–227.

[49] H. Kim, G. De Veciana, Losing opportunism: Evaluating service integration in an opportunistic wireless system, in: IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications, IEEE, 2007, pp. 982–990.

[50] I. Adan, J. van Leeuwaarden, E. Winands, On the application of Rouché's theorem in queueing theory, Operations Research Letters 34 (3) (2006) 355–360.