mathematics

MDPI

Article

Evaluation of the Waiting Time in a Finite Capacity Queue with Bursty Input and a Generalized Push-Out Strategy

Chris Blondia

IDLab-Department of Computer Science, University of Antwerp-imec, 2000 Antwerp, Belgium; chris.blondia@uantwerpen.be

Abstract: In this paper, we study a finite capacity queue where the arrival process is a special case of the discrete time Markov modulated Poisson process, the service times are generally distributed, and the server takes repeated vacations when the system is empty. The buffer acceptance strategy is based on a generalized push-out scheme: when the buffer is full, an arriving customer pushes out the Nth customer in the queue, where N takes values between 2 and the capacity of the system, and the arriving customer joins the end of the queue. Such a strategy is important when, as well as short waiting times for served customers, the time a pushed-out customer occupies a buffer space is also an important performance measure. The Laplace transform of the waiting time of a served customer is determined. Numerical examples show the influence of the bustiness of the input process and also the trade-off between the average waiting time of served customers and the occupancy of the buffer space of pushed-out customers.

Keywords: push-out strategy; bursty input; finite capacity queue; server vacations; Markov chain

MSC: 60J10; 60K25; 68M20; 68M12

Citation: Blondia, C. Evaluation of the Waiting Time in a Finite Capacity Queue with Bursty Input and a Generalized Push-Out Strategy. *Mathematics* **2022**, *10*, 4771. https://doi.org/10.3390/math10244771

Academic Editors: Ivan Atencia and José Luis Galán-García

Received: 25 November 2022
Accepted: 12 December 2022
Published: 15 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

This paper considers a queue with a finite buffer, where the server takes repeated vacations. Customers arrive at the system according to a bursty arrival process. They are served according to a first-come-first-served scheduling scheme. Arriving customers are admitted to the system using a general push-out strategy. When a customer arrives at a full buffer, then the customer in the queue on the Nth place (N being a value between 2 and the capacity of the queue) is pushed out and the arriving customer joins the end of the queue. This extends the results of [1] valid for $N = 2$, when the server is active but differs from the one presented in this paper when the server is on vacation. Indeed, in [1] when a customer arrives at a full system while the server takes a vacation, the customer that is ready to be served first will be pushed out. In our system, it is not the customer ready for service that is pushed out, but the next customer ($N = 2$). The analysis is based on the approach in [1,2].

Pushing out the oldest customer as in [1] is the appropriate strategy if the aim is to minimize the waiting time of a served customer. However, in this case, we do not take into account the time a pushed-out customer occupies a buffer space, an important resource of the system. In the system considered in this paper, this occupation time is taken into account by not necessarily pushing out the oldest customer, but a customer who has spent less time in the queue. Still, the resulting waiting time is an important performance measure, and therefore the trade-off between the waiting time of a served customer and the time a pushed-out customer occupies a position in the queue is considered.

Push-out strategies have received a lot of attention in the context of multi-class queueing systems, where customers of a higher priority class may push out a customer of a lower class when a threshold is reached or the queue is full (e.g., [3–5]). More complex systems have been studied too. In [6], a queueing system M1, M2/G1, G2/1/N with different

scheduling and push-out schemes is evaluated. Ref. [7] considers two traffic flows, namely regular and negative customers. Regular customers are stored in a finite buffer and a negative customer pushes regular customers out of the queue and moves them to another queue. A push-out scheme with differentiated dropping that uses a weight function to estimate the weight of active flows based on their traffic intensity is described in [8]. Ref. [9] considers a push-out scheme for a queueing system, where a packet requires several rounds of service before it can be transmitted. The push-out scheme is based on the amount of work that is needed by a new arrival. In [10], an arriving customer that finds the queue full joins a retrial waiting group in order to seek service again after a period of time that is exponentially distributed. Ref. [11] considers a randomized push-out mechanism for secondary customers to free up space that can be taken by primary customers. In [12], a finite capacity M/G/1 priority queue is studied, where non-preemptive time priority is given to delay-sensitive traffic and push-out space priority to loss-sensitive traffic. Also, combination with other strategies has been studied [13]. A push-out strategy as in [1] was used in [14] in the context of battery-less low power sensor node communication.

In this paper, we do not consider different traffic classes but aim at a generalization of the results obtained in [1]. We are interested in the waiting time of a customer that is served when a push-out mechanism is used that does not necessarily push out the oldest customer, but a customer that has spent less time in the queue than the oldest customer and hence has occupied fewer resources during its stay in the queue.

Hence, the aim of this paper is threefold:

- To extend the results of [1] for a bursty input stream.
- To evaluate the waiting time of a served customer in a system where an arriving customer at a full buffer pushes out the customer on position N, where N can take all possible values between 2 and the capacity of the queue; this extends the result of [1] for other values than $N = 2$.
- To determine the trade-off between minimizing the waiting time of a served customer and the time a pushed-out customer occupies a position in the queue.

Whereas the first two goals are more of scientific interest as the corresponding results extend earlier work [1], the third goal leads to results that can be applied in a system where occupying a buffer place comes with a cost, and therefore there may be a trade-off between the time a pushed-out customer occupies a buffer place and the waiting time of a served customer. To obtain these results, we first computed the number of customers in the system, both at embedded time instants and at an arbitrary time instant using the appropriate Markov chain. The approach is similar to Lee [2], extended in [8] to allow a two-dimensional chain. The waiting time distribution of a tagged served customer is obtained by considering both the time that a customer spends in the queue between its arrival instant and the end of the vacation or the service upon arrival, together with the time needed to serve all customers that were in front of the tagged customer in the queue upon its arrival. The mean time a pushed-out customer spends in the queue is also derived.

2. System Model

Consider a single server queue with following characteristics. The service time of a customer has distribution $S(t)$, with mean $E[S]$ and Laplace Transform $S^*(\theta)$. When at the end of a service of a customer, the system is empty, then the server takes repeated vacations (i.e., vacations are repeated as long as the system is empty upon return of the server), the length of which has probability distribution $V(t)$, with mean $E[V]$ and Laplace Transform $V^*(\theta)$. The arrival process is a special case of the discrete time Markov modulated Poisson process (MMPP) and is characterized by a state transition matrix $\boldsymbol{Q} = (q_{i,j})_{1 \le i,j \le M}$ and Poisson arrival rates $\boldsymbol{\Lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_M)$. Transitions between states only take place at the end of a service or a vacation, using the transition matrix $\boldsymbol{Q}$. While in state i, customers arrive according to a Poisson process with rate λ_i, $1 \le i \le M$.

The system has capacity K, consisting of $K-1$ buffer places and a place for the customer in service. The buffer acceptance strategy is a generalized push-out strategy with parameter N, $2 \leq N \leq K$. When a customer arrives at a full system (i.e., K customers are present), the Nth customer is pushed out and the arriving customer joins the end of the queue. We will compare the performance of the system using a Push-Out strategy (PO_N), for different values of N, with the system using the Drop-Tail (DT) strategy (i.e., a customer arriving at a full system is dropped).

Let r be the steady state vector of the MMPP, i.e.,

$$\boldsymbol{r}\,\boldsymbol{Q} = \boldsymbol{r} \text{ and } \boldsymbol{r}\,\boldsymbol{e} = 1, \tag{1}$$

where e is the unit column vector of size M. The fundamental arrival rate is then given by

$$\lambda = \boldsymbol{r}\,\boldsymbol{\Lambda}'. \tag{2}$$

3. Number of Customers in the System

Notice that the number of customers in the system using the Drop-Tail strategy is the same as in the system using the Push-Out strategy with parameter N. Clearly this will not be the case for the waiting time of a served customer. The derivation of the number of customers in the system follows a similar method to that in [15]. However, in that paper, the additional variable models the available energy in the energy-harvesting system, whereas in the current system the additional variable models the state of the arrival process.

3.1. Number of Customers in the System at the End of a Vacation or a Service

Let t_n be the time instants at which a served customer leaves the system or a vacation ends. Furthermore, let L_n be the number of customers at t_n and a_n be the state of the arrival process at t_n (after a transition has been made). Then define

$$p_{k,i} = \lim_{n\to\infty} Prob\{L_n = k \wedge a_n = i\},\ 0 \leq k \leq K,\ 1 \leq i \leq M, \tag{3}$$

and

$$\boldsymbol{p_k} = (p_{k,1}, p_{k,2}, \cdots, p_{k,M}) \text{ and } \boldsymbol{p} = (\boldsymbol{p}_0, \cdots, \boldsymbol{p_K}). \tag{4}$$

Let $g_{k,i}$, resp. $h_{k,i}$, be the probability that k customers arrive during a service time, resp. a vacation, while the state of the arrival process is i. Then

$$g_{k,i} = \int_0^\infty \frac{(\lambda_i t)^k}{k!} e^{-\lambda_i t} dS(t), \tag{5}$$

$$h_{k,i} = \int_0^\infty \frac{(\lambda_i t)^k}{k!} e^{-\lambda_i t} dV(t). \tag{6}$$

Let $(\boldsymbol{A_k})_{i,j}$, resp. $(\boldsymbol{B_k})_{i,j}$, be the probability that during a service time, resp. a vacation, k customers arrive, the state of the arrival process at the start of the service, resp. vacation, was i, and at the end the state is j. Then

$$(\boldsymbol{A_k})_{i,j} = g_{k,i}\ q_{i,j}, \tag{7}$$

$$(\boldsymbol{B_k})_{i,j} = h_{k,i}\ q_{i,j}. \tag{8}$$

The stochastic process $(L_n, a_n)_{n\in N}$ is a Markov chain with transition matrix

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{B_0} & \boldsymbol{B_1} & \boldsymbol{B_2} & \cdots & \boldsymbol{B_{K-2}} & \boldsymbol{B_{K-1}} & \sum_{k=K}^{\infty} \boldsymbol{B_k} \\ \boldsymbol{A_0} & \boldsymbol{A_1} & \boldsymbol{A_2} & \cdots & \boldsymbol{A_{K-2}} & \sum_{k=K-1}^{\infty} \boldsymbol{A_k} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A_0} & \boldsymbol{A_1} & \cdots & \boldsymbol{A_{K-3}} & \sum_{k=K-2}^{\infty} \boldsymbol{A_k} & \boldsymbol{0} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{A_0} & \sum_{k=1}^{\infty} \boldsymbol{A_k} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \sum_{k=0}^{\infty} \boldsymbol{A_k} & \boldsymbol{0} \end{bmatrix} \tag{9}$$

The distribution vector $\boldsymbol{p}$ satisfies the following equations

$$\boldsymbol{p} = \boldsymbol{p}\,\boldsymbol{P}, \tag{10}$$

$$\boldsymbol{p}\,\boldsymbol{e} = 1, \tag{11}$$

where $\boldsymbol{e}$ is the unit column vector of size $(K+1)M$.

Now it is possible to derive an expression for the joint distribution of the system occupancy and the state of the arrival process at the end of a service, $\boldsymbol{\pi_k} = (\pi_{k,1}, \pi_{k,2}, \cdots, \pi_{k,M})$, resp. at the end of a vacation $\boldsymbol{\omega_k} = (\omega_{k,1}, \omega_{k,2}, \cdots, \omega_{k,M})$.

Indeed, at the end of a service, there are $k = 0, \cdots, K-2$ customers present, if at the end of the previous time instant there were $l = 1, \cdots, k+1$ customers present and $k-l+1$ customers arrived during that service time. Taking into account the state of the arrival process leads to the following expression for $\boldsymbol{\pi_k}$, $k = 0, \cdots, K-2$:

$$\boldsymbol{\pi_k} = \sum_{l=1}^{k+1} \boldsymbol{p_l}\,\boldsymbol{A_{k-l+1}},\ k = 0, \cdots, K-2. \tag{12}$$

A similar reasoning leads to the following expressions:

$$\boldsymbol{\pi_{K-1}} = \sum_{l=1}^{K} \boldsymbol{p_l}\ \sum_{n=K-l}^{\infty} \boldsymbol{A_n}, \tag{13}$$

$$\boldsymbol{\omega_k} = \boldsymbol{p_0}\,\boldsymbol{B_k},\ k = 0, \cdots, K-1, \tag{14}$$

$$\boldsymbol{\omega_K} = \boldsymbol{p_0}\ \sum_{n=K}^{\infty} \boldsymbol{B_n}. \tag{15}$$

Clearly

$$\boldsymbol{p_k} = \boldsymbol{\pi_k} + \boldsymbol{\omega_k},\ k = 0, \cdots, K-1, \tag{16}$$

$$\boldsymbol{p_K} = \boldsymbol{\omega_K}. \tag{17}$$

3.2. Number of Customers in the System at an Arbitrary Time Instant

Let t be an arbitrary time instant. Let $\boldsymbol{\Omega}_{k,i}(t)dt$ the probability that t falls in a vacation that starts with the arrival process state equal to i, that at time t there are k customers in the queue and the remaining time of the vacation $\widetilde{V}$ satisfies $t < \widetilde{V} \leq t + dt$. Similarly, let $\boldsymbol{\Pi}_{k,i}(t)dt$ be the probability that t falls in a service, that at the start of this service the state of the arrival process is i, that at time t there are k customers in the queue and the remaining time of the service $\widetilde{S}$ satisfies $t < \widetilde{S} \leq t + dt$. We denote the corresponding Laplace Stieltjes Transforms (LSTs) by

$$\boldsymbol{\Omega}^*_{k,i}(\theta) = \int_0^{\infty} e^{-\theta t}\,\boldsymbol{\Omega}_{k,i}(t)dt, \tag{18}$$

$$\boldsymbol{\Pi}^*_{k,i}(\theta) = \int_0^{\infty} e^{-\theta t}\,\boldsymbol{\Pi}_{k,i}(t)dt. \tag{19}$$

We follow a similar method of reasoning to Lee ([2]) to compute these LSTs. The first terms in Equations (20)–(23) and the first terms of (5) in Lee [2] clearly coincide. However, the simplifications that can be made in Equations (5a), (5b) and (5d) of Lee [2] to turn the two sums in the second terms into a single sum are no longer possible in our case. E.g., in

Equation (5a) of [2], the second term $\sum_{k=1}^{j+1} g_{j-k+1}(p_k + q_k)$ can be replaced by p_j. This is not possible in our model, since $q_{i,j}$ is missing to apply this simplification (see Equations (7) and (8)), and hence, instead of having a single sum as in [2], the double sum remains.

$$\mathbf{\Omega}^*_{k,i}(\theta) = \frac{1}{\lambda_i \cdot D}\left\{ V^*(\theta)\ (\pi_{0,i} + \omega_{0,i})\left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{k+1} - \sum_{l=0}^{k}(\ \pi_{0,i} + \omega_{0,i})\ h_{l,i} \cdot \left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{k-l+1}\right\}, \tag{20}$$

$$\mathbf{\Omega}^*_{K,i}(\theta) = -\frac{1}{\lambda_i\ D}\left\{ V^*(\theta)\ (\pi_{0,i} + \omega_{0,i})\left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{k+1} - \sum_{l=0}^{K}\sum_{j=0}^{K-1-l}(\ \pi_{l,i} + \omega_{l,i})\ h_{j,i}\ \left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{K-l-j}\right\}, \tag{21}$$

$$\mathbf{\Pi}^*_{k,i}(\theta) = \frac{1}{\lambda_i\ D}\left\{ S^*(\theta)\ \sum_{l=1}^{k}(\ \pi_{l,i} + \omega_{l,i})\ \left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{k-l+1} - \sum_{l=1}^{k}\sum_{j=0}^{k-l}(\pi_{l,i} + \omega_{l,i})\ g_{j,i}\ \left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{k-l-j+1}\right\}, \tag{22}$$

$$\mathbf{\Pi}^*_{K,i}(\theta) = -\frac{1}{\lambda_i\ D}\left\{ S^*(\theta)\ [\sum_{l=1}^{K-1}(\pi_{l,i} + \omega_{l,i})\ \left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{K-l} + \omega_{K,i}] - \sum_{l=0}^{K-1}\sum_{j=0}^{K-1-l}(\ \pi_{l,i} + \omega_{l,i})\ g_{j,i}\ \left(\frac{\lambda_i}{\lambda_i - \theta}\right)^{K-l-j}\right\}, \tag{23}$$

with

$$D = \sum_{j=1}^{M}\left\{(\ \omega_{0,j} + \pi_{0,j})\ E[V] + \left(\sum_{k=1}^{K}\omega_{k,j} + \sum_{k=1}^{K-1}\pi_{k,j}\right)\ E[S]\right\}. \tag{24}$$

The distribution of the number of packets in the system at an arbitrary time instant is then given by

$$\eta_0 = \sum_{j=1}^{M}\mathbf{\Omega}^*_{0,j}(0), \tag{25}$$

$$\eta_k = \sum_{j=1}^{M}[\mathbf{\Omega}^*_{k,j}(0) + \mathbf{\Pi}^*_{k,j}(0)],\ k = 1, \ldots, K. \tag{26}$$

4. Waiting Time Distribution of a Served Customer

We follow a reasoning similar to [1], but where the input is a special case of the discrete-time MMPP and the customer on the Nth place, $2 \leq N \leq K$, is pushed out in case of a full system. The waiting time of a tagged served customer consists of two parts. The first part consists of the time the tagged customer spends in the queue between its arrival instant and the end of the vacation or the service upon its arrival. The second part is the time needed to serve all customers that were in front of the tagged customer in the queue upon its arrival. Notice that these two parts are not independent. In what follows, the LST of these two parts are determined.

Let t be an arbitrary time instant. Let $\mathbf{\Omega}_{k:n,i}(t)dt$ be the probability that t falls in a vacation that starts with the arrival process state equal to i, that at time t there are k customers in the queue, the remaining time of the vacation $\widetilde{V}$ satisfies $t < \widetilde{V} \leq t + dt$, and that during this remaining time n new customers arrive.

Similarly, let $\mathbf{\Pi}_{k:n,i}(t)dt$ be the probability that t falls in a service, that at the start of this service the state of the arrival process is i, that at time t there are k customers in the queue, that the remaining time of the service $\widetilde{S}$ satisfies $t < \widetilde{S} \leq t + dt$, and that during this remaining time n new customers arrive.

We denote the corresponding LSTs by $\mathbf{\Omega}^*_{k:n,i}(\theta)$ and $\mathbf{\Pi}^*_{k:n,i}(\theta)$. From their definition, it is clear that

$$\mathbf{\Omega}^*_{k:n,i}(\theta) = \int_0^\infty \frac{(\lambda_i t)^n}{n!} e^{-(\lambda_i + \theta)\ t}\ \mathbf{\Omega}_{k,i}(t)dt, \tag{27}$$

$$\mathbf{\Pi}^*_{k:n,i}(\theta) = \int_0^\infty \frac{(\lambda_i t)^n}{n!} e^{-(\lambda_i + \theta)\ t}\ \mathbf{\Pi}_{k,i}(t)dt. \tag{28}$$

Now consider the second part of the waiting time. Let $\boldsymbol{W}_{k:n,j}(t)$ be the waiting time distribution of a tagged customer that has k customers ahead and n customers behind it at the end of a service or a vacation that started with the arrival process in state j. The corresponding LST is denoted by $\boldsymbol{W}^*_{k:n,j}(\theta)$. Let

$$S^*_{n,i}(\theta) = \int_0^\infty \frac{(\lambda_i t)^n}{n!} e^{-(\theta+\lambda_i)t} dS(t). \tag{29}$$

The LSTs $\boldsymbol{W}^*_{k:n,j}(\theta)$ then satisfy the following recursive formulas. These formulas are obtained by considering at the end of a service $W^*_{k:n,j}(\theta)$ and seeing how at the end of the next service the tagged customer sees k-1 customers ahead of it, in case no customer of the k customers has been pushed out, or the correct number of customers ahead of it, in case one or more of the k customers ahead of it has been pushed out (as is the case in the second term of Equation (34)), together with the number of customers behind it. Notice that the tagged customer will be served eventually, as we are interested in the waiting time distribution of a served customer.

For $k = 0$:

$$\boldsymbol{W}^*_{0:n,j}(\theta) = 1 \tag{30}$$

For $1 \le k \le N-2$, $0 \le n \le K-k-2$ and $1 \le i \le M$:

$$\begin{aligned} \boldsymbol{W}^*_{k:n,i}(\theta) \;=\; & \sum_{j=1}^{M} \Big\{ \sum_{l=0}^{K-k-n-1} S^*_{l,i}(\theta)\; W^*_{k-1:n+l,j}(\theta) \\ & + \left[\sum_{l=K-k-n}^{\infty} S^*_{l,i}(\theta) \right] \boldsymbol{W}^*_{k-1:K-k-1,j}(\theta) \Big\}\; q_{i,j}\,. \end{aligned} \tag{31}$$

It is easy to verify that in the previous equations, the infinite sum can be replaced by

$$\sum_{l=K-k-n}^{\infty} S^*_{l,i}(\theta) = S^*(\theta) - \sum_{l=0}^{K-k-n-1} S^*_{l,i}(\theta). \tag{32}$$

For $k = N-1$, $0 \le n \le K-N$ and $1 \le i \le M$:

$$\boldsymbol{W}^*_{N-1:n,i}(\theta) = \sum_{j=1}^{M} \sum_{l=0}^{K-N-n} S^*_{l,i}(\theta)\; \boldsymbol{W}^*_{N-2:n+l,j}(\theta)\; q_{i,j} \tag{33}$$

Finally, for $N \le k \le K-1$, $0 \le n \le K-k-1$ and $1 \le i \le M$:

$$\begin{aligned} \boldsymbol{W}^*_{k:n,i}(\theta) = \sum_{j=1}^{M} \Big\{ & \sum_{l=0}^{K-k-n-1} S^*_{l,i}(\theta)\; \boldsymbol{W}^*_{k-1:n+l,j}(\theta) + \\ & \sum_{l=K-k-n}^{K-N-n} S^*_{l,i}(\theta)\; \boldsymbol{W}^*_{K-l-n-2:n+l,j}(\theta) \Big\}\; q_{i,j}. \end{aligned} \tag{34}$$

Using the expressions for $\boldsymbol{\Omega}^*_{k:n,i}(\theta)$, $\boldsymbol{\Pi}^*_{k:n,i}(\theta)$ and $\boldsymbol{W}^*_{k:n,i}(\theta)$, it is possible to derive a formula for the LST of the waiting time of a tagged customer that is served.

Let $W(t)$ be the distribution of the waiting time of a served customer and $W^*(\vartheta)$ the corresponding LST. Then, according to Theorem 5 in [16],

$$W^*(\vartheta) = \frac{1}{(1-P_{loss})\; r\; \boldsymbol{\Lambda}'} \sum_{i=1}^{M} (r_i\; \lambda_i)\; \boldsymbol{W}^*_i(\theta), \tag{35}$$

with $\boldsymbol{W}^*_i(\theta)$ consisting of a sum of all possible values of k and n of two factors, where the first factors $\boldsymbol{\Omega}^*_{k:n,i}(\theta)$ and $\boldsymbol{\Pi}^*_{k:n,i}(\theta)$ stand for the time the tagged customer spends in the queue between its arrival instant and the end of the vacation or the service upon its arrival,

while the second factor $\boldsymbol{W}^*_{k:n,i}(\theta)$ takes into account the time needed to serve all customers that were in front of the tagged customer in the queue upon its arrival:

$$\begin{aligned}\boldsymbol{W}^*_i(\theta) = & \sum_{k=0}^{N-2}\{\sum_{n=0}^{K-k-1}\boldsymbol{\Omega}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{k:n,i}(\theta) + [\sum_{n=K-k}^{\infty}\boldsymbol{\Omega}^*_{k:n,i}(\theta)]\ \boldsymbol{W}^*_{k:K-k-1,i}(\theta)\}\\ & + \sum_{n=0}^{K-N}\boldsymbol{\Omega}^*_{N-1:n,i}(\theta)\ \boldsymbol{W}^*_{N-1:n,i}(\theta) + \sum_{k=N}^{K-1}[\sum_{n=0}^{K-k-1}\boldsymbol{\Omega}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{k:n,i}(\theta)\\ & + \sum_{n=K-k}^{K-N}\boldsymbol{\Omega}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{K-n-1:n,i}(\theta)] + \sum_{n=0}^{K-N}\boldsymbol{\Omega}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{K-n-1:n,i}(\theta) +\\ & \sum_{k=1}^{N-2}\{\sum_{n=0}^{K-k-1}\boldsymbol{\Pi}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{k-1:n,i}(\theta) + [\sum_{n=K-k}^{\infty}\boldsymbol{\Pi}^*_{k:n,i}(\theta)]\ \boldsymbol{W}^*_{k-1:K-k-1,i}(\theta)\}\\ & + \sum_{n=0}^{K-N}\boldsymbol{\Pi}^*_{N-1:n,i}(\theta)\ \boldsymbol{W}^*_{N-2:n,i}(\theta) + \sum_{k=N}^{K-1}[\sum_{n=0}^{K-k-1}\boldsymbol{\Pi}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{k-1:n,i}(\theta)\\ & + \sum_{n=K-k}^{K-N}\boldsymbol{\Pi}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{K-n-2:n,i}(\theta)] + \sum_{n=0}^{K-N}\boldsymbol{\Pi}^*_{k:n,i}(\theta)\ \boldsymbol{W}^*_{K-n-2:n,i}(\theta).\end{aligned} \tag{36}$$

The average waiting time of an arriving served customer is then given by

$$E[W] = -\frac{d}{d\theta}W^*(\theta)|_{\theta=0} \tag{37}$$

According to [16], Theorem 5, the average waiting time of an arriving served customer in the system using the Drop-Tail strategy W_{DT} is then given by

$$E[W_{DT}] = \frac{\sum_{n=1}^{N} n\ \eta_k}{(1-P_{loss})\ r\ \boldsymbol{\Lambda}'} - E[S]. \tag{38}$$

5. Mean Time That Pushed-out Customers Occupy Buffer Space

In order to study the trade-off between the mean waiting time of a served customer and the time a pushed-out customer occupies a buffer space, we need to know the mean waiting of a pushed-out customer. Using the same reasoning as in [1], it is possible to show that

$$E[W_{PO}|customer\ pushed-out] = \frac{(1-P_{loss})}{P_{loss}}(E[W_{DT}] - E[W_{PO}|customer\ served]), \tag{39}$$

where W_{PO} is the waiting time of a customer when the push-out strategy is used, and W_{DT} is the waiting time in the Drop-Tail system. Since both $E[W_{DT}]$ (formula (38)) and $E[W_{PO}|customer\ served]$ (formula (37)) are known, it is possible to obtain $E[W_{PO}|customer\ pushed\ out]$. This gives the time a customer that will eventually be pushed out occupies a buffer space in the system and hence is a measure for the waist of buffer capacity due to the push-out buffer acceptance strategy.

6. Numerical Results

In the following numerical examples, we let both the service time length and the vacation length have a constant value equal to 1. The capacity of the system is $K = 16$. This allows the explicit computation of (7) and (8), and therefore also of (12), . . . , (15). Once the results for $\boldsymbol{\pi_k}$ and $\boldsymbol{\omega_k}$ are known, it is possible to compute the inverse LST of $\boldsymbol{\Omega}^*_{k,i}(\theta)$ and $\boldsymbol{\Pi}^*_{k,i}(\theta)$ using formulas (20), . . . , (23). Both parts of the waiting time of a served customer $\boldsymbol{\Omega}^*_{k:n,i}(\theta)$, $\boldsymbol{\Pi}^*_{k:n,i}(\theta)$ and $\boldsymbol{W}^*_{k:n,i}(\theta)$ can be computed and, taking the derivative in Formula (37), this leads to the mean waiting time of a served customer. In the examples, we let the parameter N vary between 2 and 16. The value $N = 1$ was not used as this would have led to the customer in service being pushed out.

The four examples address the goals of this paper, as stated in the introduction. Examples 1 and 2 show the average waiting time as a function of the load when the parameter N varies between 2 and 16. Example 2 not only showed that our analysis extends

the results of [1], but also aimed at giving an insight in the convergence of the waiting time as the load increases for different values of N. Since one of the goals of this paper was to extend the results of [1] to a bursty input process, the impact of burstiness on the average waiting time was investigated in Example 3. In Example 4, the trade-off between minimizing the waiting time of a served customer and the time a pushed-out customer occupies a position in the queue was investigated.

6.1. Example 1: Poisson Input

First, we considered the push-out scheme with Poisson input. The arrival rate λ varied between 0 and 20. Figure 1 shows the average waiting time for served customers for variable push-out parameter N, $2 \leq N < 16$, together with the corresponding result for the Drop-Tail strategy (i.e., when a customer arriving at a full buffer was dropped). The average waiting time increased clearly with the increasing value of N, with average waiting time for the Drop-Tail strategy as an upper bound. These results correspond to the results obtained in [1] for $N = 2$. The difference between the two models in terms of $E[W]$ for $N = 2$ is very small indeed. If the load was high, then the server only exceptionally took a vacation and hence the difference between the models was negligible in this case. In cases where the load was low, the server took vacations when, at the end of a service, the system was empty. However, unless the vacation was very long, the probability that the queue was full, and hence a push-out occurred, was extremely low. Hence, only if the vacation was long enough and the load was such that vacations occurred with non-negligible probability, and such that the queue was full at the end of a vacation, was the difference between the two models noticeable. However, even in that case, the difference was small as the models only differ in one buffer place, which was the candidate to be pushed out.

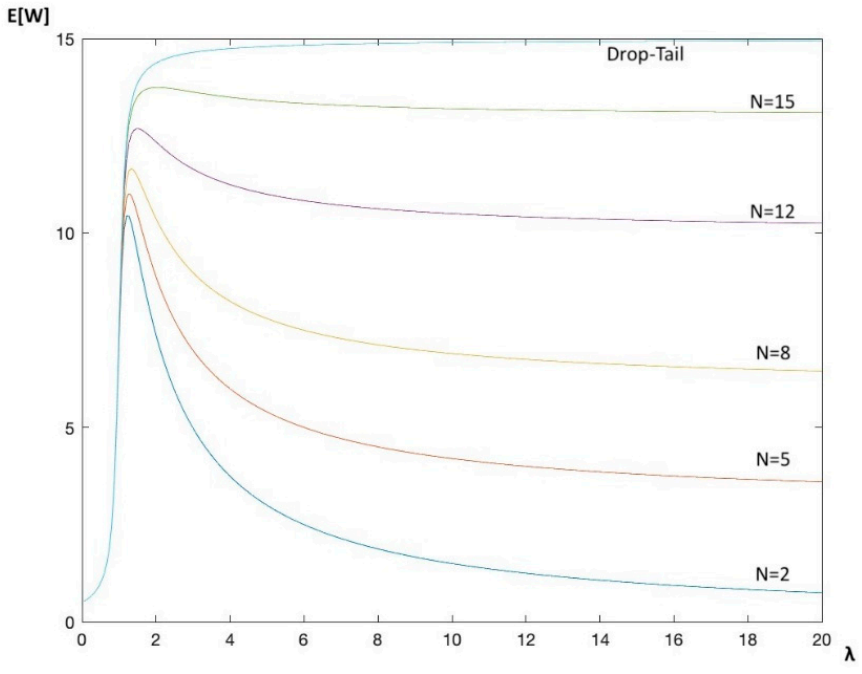

Figure 1. Mean waiting time for variable Poisson input rate for different values of N.

6.2. Example 2: Busty Input

In this example, we let the input process be defined by the state transition matrix $\boldsymbol{Q} = \begin{pmatrix} 0.3 & 0.7 \\ 0.1 & 0.9 \end{pmatrix}$ and Poisson arrival rates $\boldsymbol{\Lambda} = (k \times 6.66\ k \times 0.134)$, $k = 1, \cdots, 50$. Figure 2 shows the average waiting time for served customers for variable push-out parameter N, $2 \leq N < 16$ as a function of the fundamental arrival rate λ.

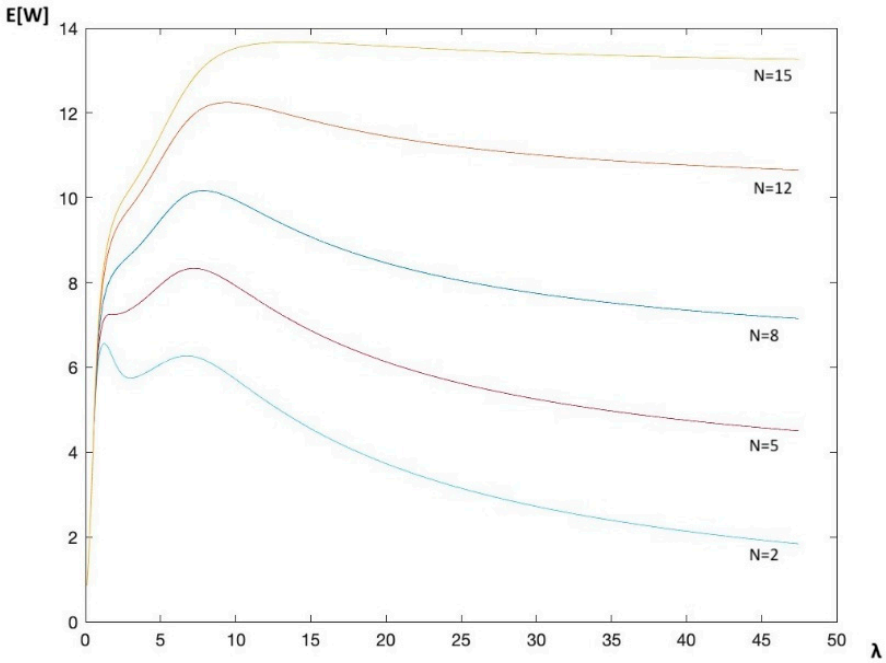

Figure 2. Mean waiting time for variable input rate of a bursty source for different values of N.

From these results, we see that for a very high load, the average waiting time tends to a value close to $N-1$. Indeed, when the load is high, a tagged customer will reach position N very rapidly after its arrival.

In most cases, this tagged customer will be pushed out, but if not (and this happens if between the arrival instant of the tagged customer at position N and the time instant the customer in service leaves the system, no new customer arrives at the system and hence the tagged customer is not pushed-out and moves to position $N-1$), the tagged customer needs to wait $N-1$ service times before it is ready to be served. A similar observation can be made if the input is a Poisson process (Example 1).

In the above example, both states gave rise to arrival rates higher than 1 (for $k > 7$). If we consider an input process where some states of the arrival process lead to low arrival rates, then the above observation no longer holds.

Let

$$Q = \begin{bmatrix} 0.1 & 0.9 & 0.0 & 0.0 \\ 0.0 & 0.1 & 0.9 & 0.0 \\ 0.0 & 0.0 & 0.2 & 0.8 \\ 0.1 & 0.0 & 0.0 & 0.9 \end{bmatrix}$$

and let $\boldsymbol{\Lambda} = (k \times 0.5,\ k \times 0.7,\ k \times 0.01,\ k \times 0.001),\ k = 1, \cdots, 10$. Figure 3 shows the mean waiting time of a served customer for different values of N. The observation that the average waiting time clearly tends towards a value close to $N-1$ no longer holds. This is due to the fact that with probability $r_4 = 0.7423$ the input process is in the fourth state, and the corresponding rate is $k \times 0.001,\ k = 1, \cdots, 10$, leading to low buffer occupancy, and hence, contrary to the previous example, a tagged customer will not reach position N very rapidly after its arrival.

6.3. Example 3: Impact of the Burstiness

In this example, we investigated the impact of the burstiness of the input stream on the mean waiting time for different values of the parameter N. We used a simple definition for the burstiness b, namely the maximum arrival rate divided by the fundamental arrival rate:

$$b = \frac{\max\limits_{i=1\cdots M} \lambda_i}{\boldsymbol{r}\,\boldsymbol{\Lambda}'} \tag{40}$$

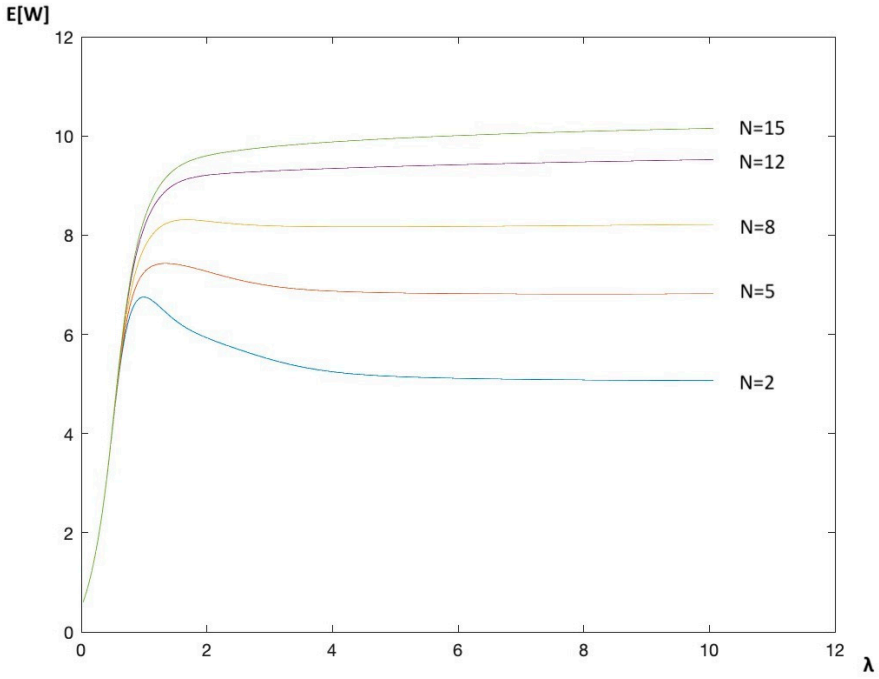

Figure 3. Mean waiting time for variable input rate of a bursty source for different values of N.

In this example, we let $M = 3$ and kept $\boldsymbol{Q}$ fixed, but changed the arrival rates $\boldsymbol{\Lambda} = (\lambda_1, \lambda_2, \lambda_3)$ in order to have increasing values of the burstiness b under constant fundamental arrival rate λ.

We let

$$\boldsymbol{Q} = \begin{bmatrix} 0.3 & 0.7 & 0.0 \\ 0.0 & 0.2 & 0.8 \\ 0.2 & 0.0 & 0.8 \end{bmatrix}$$

and considered three sets of arrival rates, $\boldsymbol{\Lambda}_1 = (8.9,\ 12.0,\ 0.6)$, $\boldsymbol{\Lambda}_2 = (4.45,\ 18.28,\ 0.3)$, $\boldsymbol{\Lambda}_3 = (1.112,\ 23.0,\ 0.075)$, chosen such that the fundamental arrival rate for all three sets is the same, namely $\lambda = 4.0$, and the respective burstiness values are given by $b_1 = 3.0$, $b_2 = 4.57$ and $b_3 = 5.75$.

In Figure 4, we see that higher burstiness leads to shorter waiting times, in particular for larger values of the parameter N. Indeed, to increase the burstiness, the values of λ_2 were increased, but to keep the fundamental arrival rate λ constant, the values of λ_1 and λ_3 were decreased. Since we did not change the transition matrix $\boldsymbol{Q}$, the lower values of λ_1 and $\lambda_3$led to shorter mean waiting times.

6.4. Example 4: Waiting Time of Pushed-Out Customers

Let $\boldsymbol{Q} = \begin{pmatrix} 0.3 & 0.7 \\ 0.1 & 0.9 \end{pmatrix}$ and arrival rates $\boldsymbol{\Lambda} = (k \times 0.6,\ k \times 0.12)$, $k = 1, \cdots, 50$. Figure 5 shows the average waiting time of served customers for variable push-out parameter N (in Figure 5 denoted by served—N = 2, 5, 8) as a function of the fundamental arrival rate λ, together with the average waiting time of pushed-out customers for these values of N (in Figure 5 denoted by PO—N = 2, 5, 8).

Figure 5 shows that, while the average waiting time of served customers for fixed fundamental arrival rate increases with increasing values of N, the average time a pushed-out customer occupies a buffer space decreases for increasing N.

Assume that the fundamental arrival rate λ is 2 and that the allowed average waiting time is 10, while the allowed average time a pushed-out customer stays in the queue is 5. N = 2 then clearly does not satisfy the latter requirement, while N = 5 does. This illustrates

the possible trade-off between the average waiting time of a served customer and the average time a pushed-out customer is allowed to occupy a buffer place.

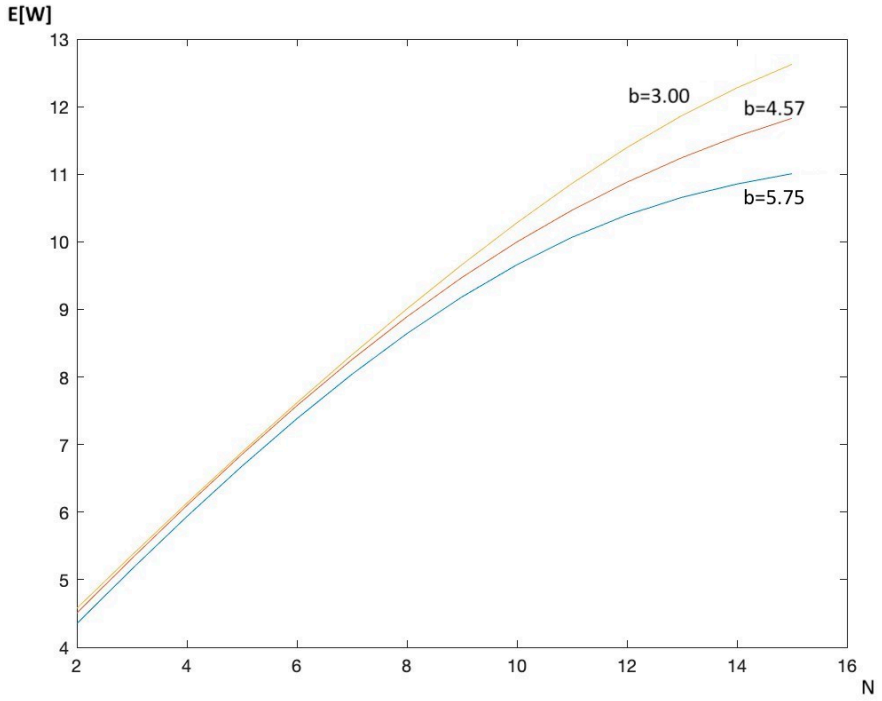

Figure 4. Mean waiting time as a function of N for different values of the burstiness of the input.

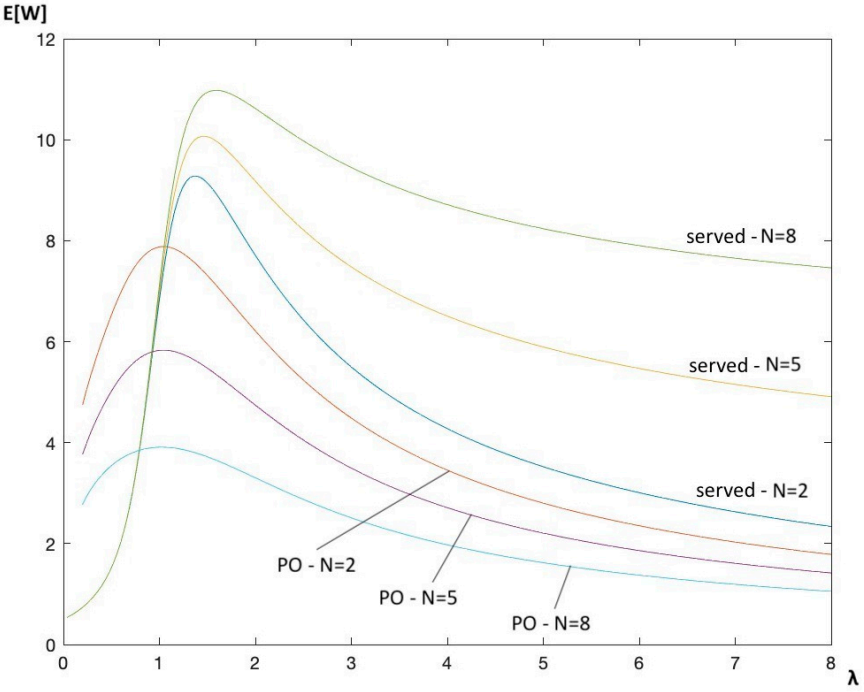

Figure 5. Mean waiting time of served and pushed-out customers for different values of N.

7. Conclusions

In this paper, we considered a queue with a finite buffer, where the server takes repeated vacations and customers arrive at the system according to a bursty arrival process. Arriving customers are admitted to the system using the following push-out strategy: when a customer arrives at a full buffer, the customer in the queue on the Nth place (N taking a

value between 2 and the capacity of the queue) is pushed out and the arriving customer joins the end of the queue. This model is an immediate extension of the one considered in [1]. The LST of the waiting time distribution of a served customer is obtained. Also, the mean time a pushed-out customer occupies a buffer is derived. The numerical results coincide with the results obtained in [1] for $N = 2$. For a deterministic distribution of both the vacation and service time, we see that the mean waiting time of served customers converges when the fundamental arrival rate increases, and that the mean waiting time is longer for increasing values of the parameter N. This means that, even for bursty arrivals, applying a push-out scheme with $N = 2$ always leads to minimal average waiting time. Furthermore, higher burstiness of the input process implies a shorter mean waiting time. Finally, we show how the model can be used to determine the possible trade-off between the average waiting time of a served customer and the average time a pushed-out customer is allowed to occupy a buffer place, a relevant application when the duration of the occupancy of a buffer comes with a cost.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kasahara, S.; Takagi, H.; Takahashi, Y.; Hasegawa, T. M/G/1/K system with push-out scheme under vacation policy. *J. Appl. Math. Stoch. Anal.* **1996**, *9*, 143–157. [CrossRef]
2. Lee, T.T. M/G/1/N queue with vacation time and exhaustive service discipline. *Oper. Res.* **1984**, *32*, 774–784. [CrossRef]
3. Lee, Y.; Choi, B.D.; Kim, B.; Sung, D.K. Delay analysis of an M/G/1/K priority queueing system with push-out scheme. *Math. Probl. Eng.* **2007**, *2007*, 14504. [CrossRef]
4. Avrachenkov, K.E.; Vilchevsky, N.O.; Shevlyakov, G.L. Priority queueing with finite buffer size and randomized push-out mechanism. *Perform. Eval.* **2005**, *61*, 1–16. [CrossRef]
5. Ilyashenko, A.; Zayats, O.; Muliukha, V.; Laboshin, L. Further Investigations of the Priority Queueing System with Preemptive Priority and Randomized Push-Out Mechanism. *Lect. Notes Comput. Sci.* **2014**, *8636*, 433–443.
6. Akyildiz, I.F.; Cheng, X. Analysis of a finite buffer queue with different scheduling and push-out schemes. *Perform. Eval.* **1994**, *19*, 317–340. [CrossRef]
7. Razumchik, R. Analysis of finite capacity queue with negative customers and bunker for ousted customers using chebyshev and gegenbauer polynomials. *Asia-Pac. J. Oper. Res.* **2014**, *31*, 1450029. [CrossRef]
8. Yang, J.P. Pushout with Differentiated Dropping Queue Management for High-Speed Networks. *Appl. Math. Inf. Sci.* **2015**, *9*, 1961–1969.
9. Kogan, K.; López-Ortiz, A.; Nikolenko, S.I. Online Scheduling FIFO Policies with Admission and Push-Out. *Theory Comput. Syst.* **2016**, *58*, 322–344. [CrossRef]
10. Korenevskaya, M.; Zayats, O.; Ilyashenko, A.; Muliukha, V. Retrial Queuing System with Randomized Push-Out Mechanism and Non-Preemptive Priority. *Procedia Comput. Sci.* **2019**, *150*, 716–725. [CrossRef]
11. Shorenko, P.; Zayats, O.; Ilyashenko, A.; Muliukha, V. Preemptive queueing system with randomized push-out mechanism and negative customers. *Lect. Notes Comput. Sci.* **2019**, *11660*, 305–317.
12. Kim, K. Finite-Buffer M/G/1 Queues with Time and Space Priorities. *Math. Probl. Eng.* **2022**, *2022*, 4539940. [CrossRef]
13. Carballo-Lozano, C.; Ayesta, U.; Fiems, D. Performance analysis of space–time priority queues. *Perform. Eval.* **2019**, *133*, 25–42. [CrossRef]
14. Sultania, A.K.; Delgado, C.; Blondia, C.; Famaey, J. Downlink Performance Modeling and Evaluation of Batteryless Low Power BLE Node. *Sensors* **2022**, *22*, 2841. [CrossRef] [PubMed]
15. Blondia, C. A queueing model for a wireless sensor node using energy harvesting. *Telecommun. Syst.* **2021**, *77*, 335–349. [CrossRef]
16. Blondia, C. Finite capacity vacation models with non-renewal input. *J. Appl. Probab.* **1991**, *28*, 174–197. [CrossRef]