Noname manuscript No.
(will be inserted by the editor)

Performance analysis of a continuous-time two-class global first-come-first-served queue with two servers and presorting

Willem Mélange · Joris Walraevens · Herwig Bruneel

Received: date / Accepted: date

Abstract This paper considers a continuous-time queueing model with two types (classes) of customers each having their own dedicated server. The objective is to have a better grasp on the concept of a global first-come-first-served service discipline with presorting, i.e., all arriving customers are accommodated in one single FCFS queue, regardless of their type, with an exception of the first P customers. For the first P customers the FCFS rule holds only within the type, i.e., customers of different types can overtake each other in order to be served. Due to the global FCFS rule the model becomes non-workconserving and on the other hand we also have to keep track of the types of customers in the first P customers. The motivation of our work is the concept of a turn lane in road traffic, i.e., a lane reserved for vehicles making a specific turn at the next junction. This paper intends to be a step towards an analytic model to aid in the decision process of various policy makers of the optimal length of turn lanes.

Keywords Queueing Theory · Road traffic · Global FCFS · Presorting

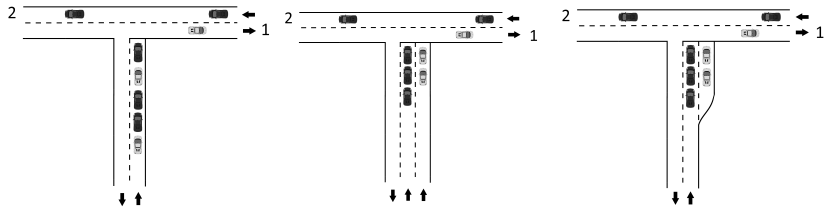
1 Introduction

The model discussed in this paper emerges from an every day situation in road traffic. Traffic jams might occur at a junction as shown in Fig. 1(a). Even though the road toward the desired destination is free (destination 1),

W. Mélange
Department of Telecommunications and Information Processing, Ghent University

J. Walraevens
Department of Telecommunications and Information Processing, Ghent University
E-mail: Joris.Walraevens@UGent.be

H. Bruneel
Department of Telecommunications and Information Processing, Ghent University
E-mail: Herwig.Bruneel@UGent.be



(a) One lane (b) Two lanes (c) One lane with a turn lane

Fig. 1 Light grey vehicles with destination 1 and dark grey vehicles with destination 2 approaching a traffic junction

it is possible to get stuck in traffic caused by vehicles for another destination (destination 2). This blocking effect is caused by the inherent First-Come-First-Served (FCFS) order on the shared road regardless the destination they have. In queueing theory terms, we have a single queue accommodating two types of customers to be served in a FCFS manner. In the rest of this paper, we will call this service discipline global FCFS (gFCFS).

In an ideal scenario, both destinations would have their own lane as shown in Fig. 1(b). In queueing theory terms, both customers then have their own queue in front of their dedicated server. Unfortunately, this is physically not always possible. A workaround for this problem is often the use of presorting or a turn lane, i.e., a short lane reserved for vehicles making a specific turn at a junction as seen in Fig. 1(c). This often counters the blocking effect of a single lane on the main road, without wasting too much road capacity. We call this new service discipline, which can be seen as a sort of relaxation of the gFCFS service discipline, gFCFS with presorting (P-gFCFS). Again in queueing theory terms, this is a service discipline where there are 2 types of customers that are accommodated in a single queue and that are served in a FCFS manner regardless of their type with an exception of the first P customers. These are served in a per-type FCFS order. This paper intends to be a step towards an analytic model and analysis to aid in the decision process of various policy makers of the optimal length of turn lanes.

As discussed in [13], traffic flows are usually modelled empirically: speed and flow data are collected for a specific road and econometrically fitted into curves, i.e., the speed-flow-density diagrams. However, several papers are available in the academic literature that model traffic flows using a queueing-theory approach (e.g. [6,12,14]). These publications have demonstrated the usability of queueing models to adequately model traffic flows by comparing queueing results with empirical data. We refer to [13] for a thorough review on modelling traffic flows with queueing models.

In [2, 3, 8], the blocking effect caused by the gFCFS service discipline is thoroughly researched in both continuous and discrete time. Here we already quantified the often devastating effect in system performance. In this paper, we add the important concept of presorting to counter this effect. The result

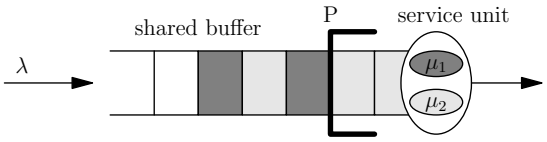


Fig. 2 Model of the system with global FCFS and presorting

of previous work can in fact be regarded as a worst case scenario (when there is no turn lane), while two separate queues can be regarded as a best case scenario.

The structure of the paper is as follows: we first describe the mathematical model in Section 2. In Section 3, we briefly turn to the problem of the stability condition. Next, in Section 4, we analyse the distribution of the number of customers in the system in two steps. The paper continues with a discussion of the results and some numerical examples in Section 5. Finally, we draw some conclusions and discuss possible future work in Section 6.

2 Mathematical model

We consider a continuous-time queueing system (as shown in Fig. 2) with infinite storage capacity. The customers enter the system according to a Poisson arrival process with mean arrival rate λ. The types of consecutive customers are independent, i.e., an arriving customer is of type 1 with probability σ or of type 2 with probability $1-\sigma$. Two types of customers (types 1 and 2) are to be served by two dedicated servers (servers 1 and 2). Customers of type 1 (2) are served by server 1 (2) and have an exponential service time with a service rate of μ_1 (μ_2). All service times are independent. The system operates under the gFCFS policy with presorting (P-gFCFS), such that the customers are served in the order of their arrival, regardless of the class they belong to, except for the P leading customers, i.e., the P oldest customers in the system (those being served included). These customers are served according to a *per-type* FCFS policy. Obviously, the most important consequence is that server 1 (2) is working if and only if there is at least one customer of type 1 (2) in the leading customers. This also means that server 1 (2) is not working if all leading customers are of type 2 (1) even though there can be a customer of type 1 (2) in the system.

Since the relation between the model and the physical junction described in the introduction might not be immediately clear, we describe it here in more detail. The main correspondence is that the events of the actual blocking in both system and model (events where one of the servers is not working although customers of that type are in the system) occur during the same time periods. The difference between model and system is that blocking is considered as blocking of the server in the model (vehicle is not able to make its turn although his destination lane is free) while in the system blocking of the turn lane (lane blockage or lane overflow) occurs. Therefore, the leading customers

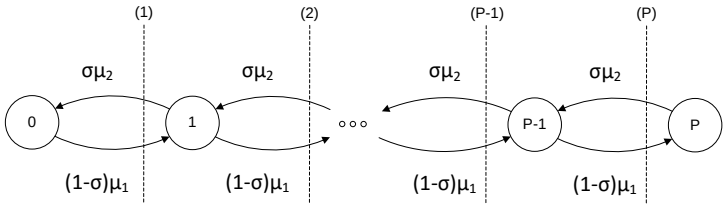


Fig. 3 $(P+1)$-state Markov chain to determine the stability condition

in the model are *not* necessarily the vehicles on the turn lanes. Likewise, when a vehicle is on the turn lane, it also does not mean that the corresponding customer is necessarily part of the leading customers in the model. However, in case the vehicle *is first in line on the turn lane*, the corresponding customer *is* necessarily part of the leading customers. Therefore in both the model and physical junction, blocking only occurs when all leading customers are of the same type, and thus the service rates and performance measures are indeed equal. More practical issues are explained briefly in Section 5.3.

3 Stability condition

The system is stable when the average amount of work per time unit that enters the system (ρ) is smaller than the average amount of work the system can execute per time unit, i.e., the average amount of work the system would execute per time unit when it would be constantly provided with new customers. Here we can define ρ as the average amount of work of type 1 and 2 per unit time

$$\rho = \rho_1 + \rho_2 \triangleq \frac{\sigma\lambda}{\mu_1} + \frac{(1-\sigma)\lambda}{\mu_2}. \tag{1}$$

To determine the average amount of work the system would execute per time unit, we first calculate the steady-state probabilities to be in states where either one or both servers are working. Some observations can help us to construct a Markov chain to calculate these probabilities. First of all, since we are looking at the stability condition, we can presume that the system is constantly provided with new customers and the system will therefore be filled with at least P customers all the time. Second, only the leading customers are of importance since customers can only be served when they are in the first P customers of the system because of the P-gFCFS service discipline. The exact queueing order of the leading customers is also of no importance; once a customer is one of the leading customers, he can be served by his server if no other customers of his type are in front of him. Therefore, we are only interested in the number of customers of type 1 and 2 in the leading customers. Notice here that when we know the number of customers of type 2 in the leading customers, we also know the number of customers of type 1. We therefore keep track of the number of customers of one type in the leading customers,

say, those of type 2. These observations lead to the $(P+1)$-state Markov chain in Fig. 3. The state m represents that m leading customers are of type 2 (and therefore, $P-m$ of type 1). The rate to go from state m to state $m-1$ is $\sigma\mu_2$; namely a rate μ_2 to end the service in state m of the customer with type 2 multiplied with the probability σ that the new P-th customer is of type 1. Similarly, the rate to go from state m to state $m+1$ is $(1-\sigma)\mu_1$. It is clear that Fig. 3 models the well-known birth-and-death process for a $M|M|1|P$ queue [7] with arrival rate $(1-\sigma)\mu_1$ and service rate $\sigma\mu_2$. The probability $p(m)$ to be in state m is known to be given by

$$p(m) = \frac{\left(\frac{(1-\sigma)\mu_1}{\sigma\mu_2}\right)^m \left(1 - \frac{(1-\sigma)\mu_1}{\sigma\mu_2}\right)}{1 - \left(\frac{(1-\sigma)\mu_1}{\sigma\mu_2}\right)^{P+1}}. \tag{2}$$

To find the stability condition, we observe that the system is able to execute 1 unit of work per unit of time when only one server is able to work, i.e., when the system is in state 0 or state P. Otherwise, when both servers are working, the system executes 2 units of work per unit of time. Therefore, the stability condition is given by

$$\rho < p(0) + 2\sum_{m=1}^{P-1} p(m) + p(P), \tag{3}$$

$$\tag{4}$$

or

$$\rho < \frac{\left(1 + \frac{(1-\sigma)\mu_1}{\sigma\mu_2}\right)\left(1 - \left(\frac{(1-\sigma)\mu_1}{\sigma\mu_2}\right)^P\right)}{1 - \left(\frac{(1-\sigma)\mu_1}{\sigma\mu_2}\right)^{P+1}}. \tag{5}$$

Equation (5) can also be rewritten as

$$\lambda < \frac{\left(\frac{\sigma}{\mu_1}\right)^P - \left(\frac{1-\sigma}{\mu_2}\right)^P}{\left(\frac{\sigma}{\mu_1}\right)^{P+1} - \left(\frac{1-\sigma}{\mu_2}\right)^{P+1}}. \tag{6}$$

which represents that on average, there are not more arrivals than service completions, i.e., the right-hand side represents the average number of service completions in a system with an infinite supply of customers.

4 Analysis of the distribution and moments of the system occupancy

We now concentrate on the system occupancy distribution. With the observations of Section 3 in mind, the system can be described by a continuous-time

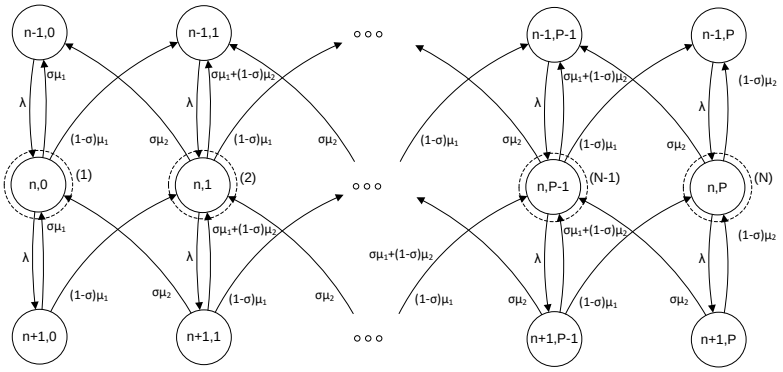


Fig. 4 Repeating part of the QBD

Markov chain where the state of the system is characterised by a pair (n, m) where $n \geq 0, 0 \leq m \leq min(n, P)$. Here n represents the total number of customers in the system (those in service included) and m represents the number of leading customers that are of type 2. The Markov chain is thus a QBD process with maximum $P + 1$ phases, and the levels are represented by the number of customers in the system. In QBD processes, the balance equations can be divided into boundary equations and repeating equations [9]. We will regard both separately.

4.1 Repeating equations

We start by looking at the repeating part of the Markov chain. QBD processes are commonly solved by using matrix-geometric techniques. Grassmann states in [5] that the problem with matrix-geometric methods is that they do not preserve the sparsity of the matrices involved. In other words, the matrix-geometric method does not exploit the fact that the matrices involved are tridiagonal which means that the computational effort can be reduced significantly. Although eigenvalues also have their problems, these seem to be manageable for the problem under investigation. In this paper we will therefore use the method of eigenvalues as described in [5]. For more details on comparisons between complexity of the matrix-geometric method and the eigenvalues method to solve similar problems, we refer to [5].

The repeating part of the QBD is shown in Fig. 4. The repeating equation can be written as

$$0 = \underline{\pi}_{n-1} Q_1 + \underline{\pi}_n Q_0 + \underline{\pi}_{n+1} Q_{-1}, \text{ for } n \geq P + 1 \tag{7}$$

with $\underline{\pi}_n = [\pi_{n,0}, \cdots, \pi_{n,P}]$, $n \geq P$, and where $\pi_{n,m}$ represents the steady-state probability to be in state (n, m), for $m = 0, \cdots, P$ and $n \geq P$. From Fig. 4

these matrices are deduced as

$$Q_1 = \begin{bmatrix} \lambda & & \\ & \ddots & \\ & & \lambda \end{bmatrix}, \tag{8}$$

$$Q_0 = \begin{bmatrix} -\lambda-\mu_1 & & & & & \\ & -\lambda-\mu_1-\mu_2 & & & & \\ & & \ddots & & & \\ & & & -\lambda-\mu_1-\mu_2 & & \\ & & & & -\lambda-\mu_2 \end{bmatrix}, \tag{9}$$

$$Q_{-1} = \begin{bmatrix} \sigma\mu_1 & (1-\sigma)\mu_1 & & & \\ \sigma\mu_2 & \sigma\mu_1+(1-\sigma)\mu_2 & (1-\sigma)\mu_1 & & \\ \ddots & \ddots & \ddots & \ddots & \\ & \sigma\mu_2 & \sigma\mu_1+(1-\sigma)\mu_2 & (1-\sigma)\mu_1 & \\ & & \sigma\mu_2 & (1-\sigma)\mu_2 \end{bmatrix}. \tag{10}$$

QBD processes have a well known geometric relation, which means that equation (7) has solutions of the form $\underline{d}x^n$ with $\underline{d}$ a vector of size $P+1$ (see also [9]). Replacing $\underline{\pi}_n$ by $\underline{d}x^n$ in (7), yields

$$0 = \underline{d}x^{n-1}Q_1 + \underline{d}x^nQ_0 + \underline{d}x^{n+1}Q_{-1}, \tag{11}$$

or by dividing by x^{n-1}

$$0 = \underline{d}Q(x), \tag{12}$$

where

$$Q(x) = Q_1 + Q_0x + Q_{-1}x^2. \tag{13}$$

The row vector $\underline{d}$ is referred to as the eigenvector and the scalar x is called the eigenvalue, because they can be found by solving a so-called generalized eigenvalue problem (sometimes referred to as matrix pencil) given by (12) [15]. It was shown in [4] that if the process is recurrent, and all eigenvalues are distinct, there are $P+1$ distinct solutions of the form $\underline{d}x^n$. We denote the k-th couple (eigenvector, eigenvalue) by $(\underline{d}^{(k)}, x_k)$, $k = 0, \cdots, P$.

The problem at hand reduces to finding these eigenvalues and eigenvectors. Expansion of (12) yields

$$0 = d_0\left[\lambda - (\lambda+\mu_1)x + \sigma\mu_1x^2\right] + d_1\left[\sigma\mu_2x^2\right], \tag{14}$$

$$0 = d_{i-1}\left[(1-\sigma)\mu_1x^2\right] + d_i\left[\lambda - (\lambda+\mu_1+\mu_2)x + (\sigma\mu_1+(1-\sigma)\mu_2)x^2\right] + d_{i+1}\left[\sigma\mu_2x^2\right], \text{ for } i = 1, \cdots, P-1 \tag{15}$$

$$0 = d_{P-1}\left[(1-\sigma)\mu_1x^2\right] + d_P\left[\lambda - (\lambda+\mu_2)x + (1-\sigma)\mu_2x^2\right]. \tag{16}$$

We now introduce functions $d_i(x)$ satisfying $d_i(x) = d_i$ whenever x is an eigenvalue or $det(Q(x)) = 0$. We can set $d_0(x) = d_0 = 1$ and replace $d_i = d_i(x)$

in (14) to (16) and solve for $d_i(x)$. This yields

$$d_1(x) = -\frac{\lambda - (\lambda + \mu_1)x + \sigma\mu_1 x^2}{\sigma\mu_2 x^2}, \tag{17}$$

$$\begin{aligned} d_{i+1}(x) = -\frac{1}{\sigma\mu_2 x^2} \big(& d_{i-1}(x) \left[(1-\sigma)\mu_1 x^2\right] \\ & + d_i(x) \left[\lambda - (\lambda + \mu_1 + \mu_2)x + (\sigma\mu_1 + (1-\sigma)\mu_2)x^2\right] \big), \\ & \text{for } i = 1, \cdots, P-1, \end{aligned} \tag{18}$$

$$\begin{aligned} d_{P+1}(x) = -\frac{1}{\sigma\mu_2 x^2} \big(& d_{P-1}(x) \left[(1-\sigma)\mu_1 x^2\right] \\ & + d_P(x) \left[\lambda - (\lambda + \mu_2)x + (1-\sigma)\mu_2 x^2\right] \big). \end{aligned} \tag{19}$$

Notice here that we have introduced the function $d_{P+1}(x)$ which satisfies $det(Q(x)) = d_{P+1}(x)\prod_{i=0}^{P}(-\sigma\mu_2 x^2)$, as shown by Wilkinson in [15]. The problem then transforms in finding an x such that $d_{P+1} = d_{P+1}(x) = 0$ (and thus $det(Q(x)) = 0$). Essentially, to find the eigenvalues, we use the fact that $\{d_i(x), i = 0, \cdots, P+1\}$ is a Sturm sequence. A Sturm sequence is any sequence with (i) $d_0(x)$ has no real roots (does not change its sign), (ii) $d_i(\epsilon) = 0$ implies $d_{i-1}(\epsilon)d_{i+1}(\epsilon) < 0$ $(sign(d_{i-1}(\epsilon)) = -sign(d_{i+1}(\epsilon)))$, (iii) all real roots of $d_{P+1}(x)$ are simple [11]. Fundamental to Sturm sequences are sign variations. The number of sign variations in the Sturm sequence $\{d_i(x), i = 0, 1, \cdots, P+1\}$ $(n(x))$ is given by

$$n(x) = \#\{d_i(x)d_{i+1}(x) < 0, 0 \leq i < P\} + \#\{d_i(x) = 0, 0 \leq i < P\}. \tag{20}$$

In [11], it is proved that there are at least $\mid n(x_1) - n(x_2) \mid$ eigenvalues between x_1 and x_2. In this specific case, $n(0+) = P+1$ and $n(1-) = 0$ which means there at least $P+1$ eigenvalues between $0+$ and $1-$. Therefore, all $P+1$ eigenvalues within the unit circle are accounted for and we can use the divide-and-conquer algorithm described in [5], which is an extension of the binary search algorithm, to find these eigenvalues. We recursively divide the search interval into two parts and discard any interval not containing an eigenvalue. We do this until we have found $P+1$ intervals containing at least one eigenvalue. Since there are only $P+1$ eigenvalues within the unit circle, all $P+1$ intervals will hold exactly one eigenvalue. Once we determined the intervals with only one eigenvalue, we can use the false position method to determine the exact value. After determining all the eigenvalues, equations (17), (18) and (19) can be used to determine the corresponding eigenvectors recursively.

Any linear combination of these solutions also forms a solution:

$$\underline{\pi}_n = \sum_{k=0}^{P} c_k \underline{d}^{(k)} x_k^n = \underline{c}\Lambda^n D, n \geqslant P \tag{21}$$

where $\underline{c} = [c_0, \cdots, c_P]$, $\Lambda = diag(x_k)$ and $D = \left[\underline{d}^{(0)}, \cdots, \underline{d}^{(P)}\right]^T$. The next step is determining $\underline{c}$ by solving the boundary equations.

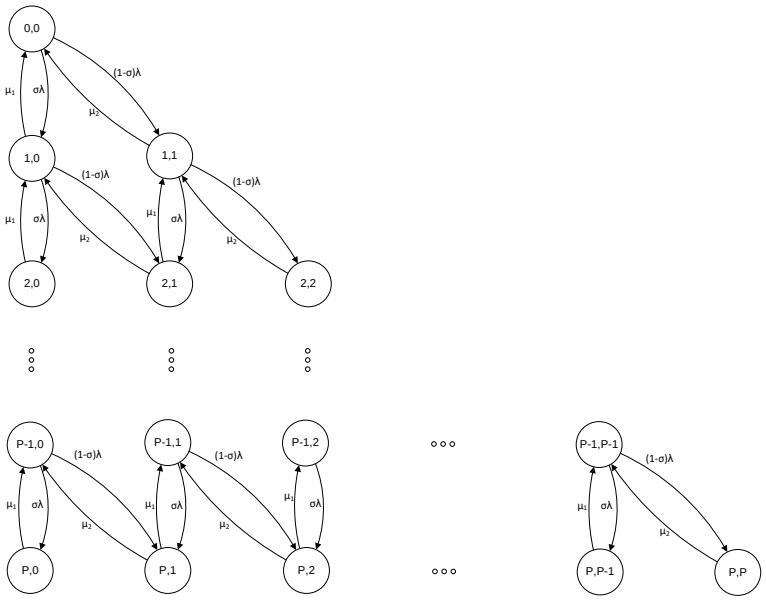


Fig. 5 Boundary part of the QBD

4.2 Boundary equations

The boundary states describe a QBD with maximum P phases. Therefore, we have $\frac{P(P+1)}{2}$ boundary states. For example, when $P = 10$, we have 55 number of states in the boundary conditions, but when $P = 100$, we have already 5050 number of states. An efficient method for solving the boundary conditions is needed. The boundary states themselves form a level-dependent QBD. Only a few approaches found in literature try to exploit the specific structure in the level-dependent case. We will follow the algorithm described in [1] and [10]. The algorithm is based on matrix continued fractions (MCF).

First we determine the generator matrix Q for $\underline{\tilde{\pi}} = \left[\underline{\pi}_0, \cdots, \underline{\pi}_{P-1}, \underline{c}\right]$ of the Markov chain of the boundary conditions where $\underline{\pi}_n = [\pi_{n,0}, \cdots, \pi_{n,m}]$ and $\pi_{n,m}$ represents the steady-state probability to be in state (n, m), for $m = 0, \cdots, n$ and $n < P$. Notice here that we have replaced $\underline{\pi}_P$ by $\underline{c}$ because we are interested in the exact values of $\underline{c}$ (which are the only unknowns left in (21) at the end of Section 4.1). This generator matrix is given by

$$Q = \begin{bmatrix} Q_{0,0} & Q_{0,1} & & & & \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & & & \\ & Q_{2,1} & Q_{2,2} & Q_{2,3} & & \\ & \ddots & \ddots & \ddots & & \\ & & Q_{P-1,P-2} & Q_{P-1,P-1} & Q_{P-1,P} & \\ & & & \Lambda^P D Q_{P,P-1} & \Lambda^P D Q_{P,P} + \Lambda^{P+1} D Q_{-1} \end{bmatrix} \tag{22}$$

where

$$Q_{i,i+1} = \begin{bmatrix} \sigma\lambda & (1-\sigma)\lambda & & \\ & \ddots & \ddots & \\ & & & \sigma\lambda \ (1-\sigma)\lambda \end{bmatrix}_{(i+1)\times(i+2)}, \text{ for } i = 0, ..., P-1, \quad (23)$$

$$Q_{0,0} = \begin{bmatrix} -\lambda \end{bmatrix}, \quad (24)$$

$$Q_{i,i} = \begin{bmatrix} -\lambda-\mu_1 & & & & & \\ & -\lambda-\mu_1-\mu_2 & & & & \\ & & \ddots & & & \\ & & & -\lambda-\mu_1-\mu_2 & & \\ & & & & -\lambda-\mu_2 \end{bmatrix}_{(i+1)\times(i+1)}, \quad (25)$$

for $i = 1, ..., P$, and,

$$Q_{i,i-1} = \begin{bmatrix} \mu_1 & & & \\ \mu_2 & \mu_1 & & \\ & \ddots & \ddots & \\ & & \mu_2 & \mu_1 \\ & & & \mu_2 \end{bmatrix}_{(i+1)\times i}, \text{ for } i = 1, ..., P. \quad (26)$$

Notice here that $Q_{P,P-1}$ and $Q_{P,P}$ in (22) are replaced by $\Lambda^P DQ_{P,P-1}$ and $\Lambda^P DQ_{P,P} + \Lambda^{P+1} DQ_{-1}$ to make the connection with the repeating equations in Section 4.1 (and introducing $\underline{c}$). This is found as follows: the last two boundary equations read

$$0 = \underline{\pi}_{P-2} Q_{P-2,P-1} + \underline{\pi}_{P-1} Q_{P-1,P-1} + \underline{\pi}_P Q_{P,P-1}, \quad (27)$$
$$0 = \underline{\pi}_{P-1} Q_{P-1,P} + \underline{\pi}_P Q_{P,P} + \underline{\pi}_{P+1} Q_{-1}, \quad (28)$$

with already a part of the repeating equations in the last term. After using (21) this becomes

$$0 = \underline{\pi}_{P-2} Q_{P-2,P-1} + \underline{\pi}_{P-1} Q_{P-1,P-1} + \underline{c}\Lambda^P DQ_{P,P-1}, \quad (29)$$
$$0 = \underline{\pi}_{P-1} Q_{P-1,P} + \underline{c}\Lambda^P DQ_{P,P} + \underline{c}\Lambda^{P+1} DQ_{-1}, \quad (30)$$

or after some rewriting

$$0 = \underline{\pi}_{P-2} Q_{P-2,P-1} + \underline{\pi}_{P-1} Q_{P-1,P-1} + \underline{c}(\Lambda^P DQ_{P,P-1}), \quad (31)$$
$$0 = \underline{\pi}_{P-1} Q_{P-1,P} + \underline{c}(\Lambda^P DQ_{P,P} + \Lambda^{P+1} DQ_{-1}). \quad (32)$$

The system of equations for solving the steady-state boundary probabilities $\tilde{\underline{\pi}}Q = 0$ with $\tilde{\underline{\pi}} = \left[\underline{\pi}_0, \cdots, \underline{\pi}_{P-1}, \underline{c}\right]$, is given by

$$0 = \underline{\pi}_0 Q_{0,0} + \underline{\pi}_1 Q_{1,0}, \tag{33}$$

$$0 = \underline{\pi}_{n-1} Q_{n-1,n} + \underline{\pi}_n Q_{n,n} + \underline{\pi}_{n+1} Q_{n+1,n}), \text{ for } n = 1, \cdots, P-2 \tag{34}$$

$$0 = \underline{\pi}_{P-2} Q_{P-2,P-1} + \underline{\pi}_{P-1} Q_{P-1,P-1} + \underline{c}\Lambda^P D Q_{P,P-1}), \tag{35}$$

$$0 = \underline{\pi}_{P-1} Q_{P-1,P} + \underline{c}(\Lambda^P D Q_{P,P} + \Lambda^{P+1} D Q_{-1}). \tag{36}$$

The MCF algorithm transforms this second-order vector-matrix difference equation into a first-order recurrence scheme [1]. In our case, this first-order recurrence scheme is

$$\underline{\pi}_{n+1} = \underline{\pi}_n R_n, \text{ for } n = 0, \ldots, P-2, \tag{37}$$

$$\underline{c} = \underline{\pi}_{P-1} R_{P-1}. \tag{38}$$

Substituting the recursions into (33) to (36) yields

$$0 = \underline{\pi}_0 (Q_{0,0} + R_0 Q_{1,0}), \tag{39}$$

$$0 = \underline{\pi}_{n-1} (Q_{n-1,n} + R_{n-1} Q_{n,n} + R_{n-1} R_n Q_{n+1,n}), \text{ for } n = 1, \cdots, P-2, \tag{40}$$

$$0 = \underline{\pi}_{P-2} (Q_{P-2,P-1} + R_{P-2} Q_{P-1,P-1} + R_{P-2} R_{P-1} \Lambda^P D Q_{P,P-1}), \tag{41}$$

$$0 = \underline{\pi}_{P-1} (Q_{P-1,P} + R_{P-1} (\Lambda^P D Q_{P,P} + \Lambda^{P+1} D Q_{-1})). \tag{42}$$

These equations can be used to compute R_n recursively

$$R_{P-1} = -Q_{P-1,P} (\Lambda^P D Q_{P,P} + \Lambda^{P+1} D Q_{-1})^{-1}, \tag{43}$$

$$R_{P-2} = -Q_{P-2,P-1} (Q_{P-1,P-1} + R_{P-1} \Lambda^P D Q_{P,P-1})^{-1}, \tag{44}$$

$$R_{n-1} = -Q_{n-1,n} (Q_{n,n} + R_n Q_{n+1,n})^{-1}, \text{ for } n = 1, \cdots, P-2. \tag{45}$$

The algorithm consists of first computing R_{P-1} and then calculating $R_n, n = P-2, \cdots, 0$, recursively. Using recursions (37) and (38), $\underline{\pi}_n, n = 0, \cdots, P-1$, and $\underline{c}$ can then be computed recursively in terms of $\underline{\pi}_0$. In practical experiments, we will first set $\underline{\pi}_0 = [1]$ as is often done in literature. Afterwards we normalize the results. Notice that we will use both the results from Sections 4.1 and 4.2 to normalize the final result. The normalization condition is given by

$$\begin{aligned} \sum_{n=0}^{\infty} \underline{\pi}_n \underline{e} &= \sum_{n=0}^{P-1} \underline{\pi}_n \underline{e} + \sum_{n=P}^{\infty} \underline{\pi}_n \underline{e} \\ &= \sum_{n=0}^{P-1} \underline{\pi}_n \underline{e} + \underline{c} \Lambda^P \left(I - \Lambda\right)^{-1} D \underline{e} = 1, \end{aligned} \tag{46}$$

where $\underline{e}$ is the column vector $\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$ and I is the identity matrix.

4.3 Moments

With these results we can calculate the mean system occupancy

$$\begin{aligned}\overline{N} &= \sum_{i=0}^{\infty} i\underline{\pi}_i\underline{e} \\ &= \sum_{i=1}^{P-1} i\underline{\pi}_i\underline{e} + \sum_{i=P}^{\infty} i\underline{\pi}_i\underline{e} \\ &= \sum_{i=1}^{P-1} i\underline{\pi}_i\underline{e} + \sum_{i=P}^{\infty} ic\Lambda^i D\underline{e} \\ &= \sum_{i=1}^{P-1} i\underline{\pi}_i\underline{e} + c\Lambda^P \left(I-\Lambda\right)^{-2} D\underline{e} + (P-1)c\Lambda^P \left(I-\Lambda\right)^{-1} D\underline{e} \\ &= \sum_{i=1}^{P-1} i\underline{\pi}_i\underline{e} + c\Lambda^P \left(I-\Lambda\right)^{-2} D\underline{e} + (P-1)(1-\sum_{i=0}^{P-1}\pi_i\underline{e}). \end{aligned} \tag{47}$$

Higher moments can be calculated as well. Using Little's law we can also calculate the mean delay of a customer

$$T = \frac{\overline{N}}{\lambda}. \tag{48}$$

4.4 Summary numerical procedure

We finally summarize the numerical procedure to calculate the stationary distribution $\underline{\pi}_n$, $n \geq 0$:

1. Calculate the $P+1$ eigenvalues x_i, $i = 0, \ldots, P$ in $]0,1[$ as zeroes of (19) using the divide-and-conquer algorithm. Summarize them in diagonal matrix Λ.
2. Calculate the corresponding eigenvectors $\underline{d}^{(i)}$ from equations (17)-(19) and summarize them in matrix $D = \left[\underline{d}^{(0)}, \cdots, \underline{d}^{(P)}\right]^T$.
3. Calculate matrices R_n, $n = P-1, \ldots, 0$ recursively through equations (43)-(45).
4. Set $\underline{\pi}_0 = 1$.
5. Calculate $\underline{\pi}_n$, $n = 1, \ldots, P-1$ recursively through equation (37).
6. Calculate $\underline{c}$ through equation (38).
7. Renormalize $\underline{\pi}_n$, $n = 0, \ldots, P-1$ and $\underline{c}$ by using equation (46).
8. Calculate $\underline{\pi}_n$, $n \geq P$ from equation (21).

5 Discussion of results and numerical examples

In the remainder of this section, we discuss the results obtained in the previous sections, from a quantitative and a qualitative perspective, by means of some

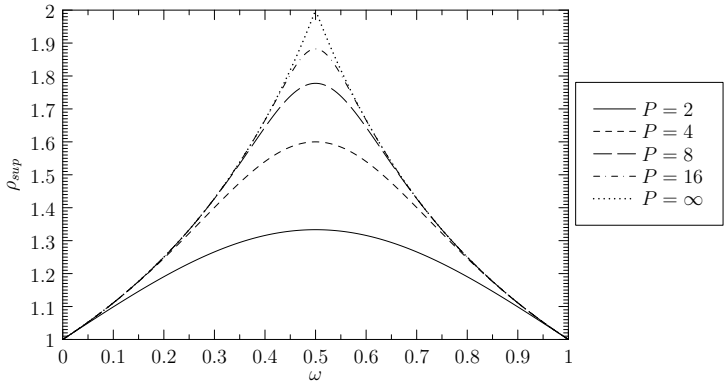


Fig. 6 Least upper bound of the set of ρ_{sup} values where the system is stable, versus ω

numerical examples. Before discussing the results, we recall the definition

$$\omega \triangleq \frac{\frac{\sigma}{\mu_1}}{\frac{\sigma}{\mu_1} + \frac{1-\sigma}{\mu_2}} = \frac{\rho_1}{\rho_1 + \rho_2}. \tag{49}$$

This parameter will allow us to interpret the results more intuitively; it represents the relative load of customers of type 1. This parameter can, for instance, be introduced in the stability condition (5), yielding

$$\rho < \frac{\left(1 + \frac{1-\omega}{\omega}\right)\left(1 - \left(\frac{1-\omega}{\omega}\right)^P\right)}{1 - \left(\frac{1-\omega}{\omega}\right)^{P+1}}. \tag{50}$$

We want to point out here that the model with a global FCFS service discipline, discussed in detail in our previous paper [8], is the case where $P = 2$ (only at the very end of the lane two vehicles can queue next to eachother). The model without global FCFS service discipline (two separate queues) is the case where $P = \infty$. In that respect, the difference with the $P = 2$ model can be seen as the performance gain of that P-scenario compared to the scenario with almost no presorting lane, while the difference with the $P = \infty$ model can be regarded as the performance loss compared to the two lanes scenario.

In the remainder, we will first show the influence of the load (ρ) and the load balance (ω) in the system. Secondly, we will zoom in on the impact of customers of one type on customers of the other type. Finally, we will demonstrate how the result in this paper could be used for dimensioning a turn lane.

5.1 Impact of load and load balance on the total system performance

Figure 6 shows the influence of the load balance on the stability condition. We have plotted the least upper bound or supremum ρ_{sup} of the set of ρ values

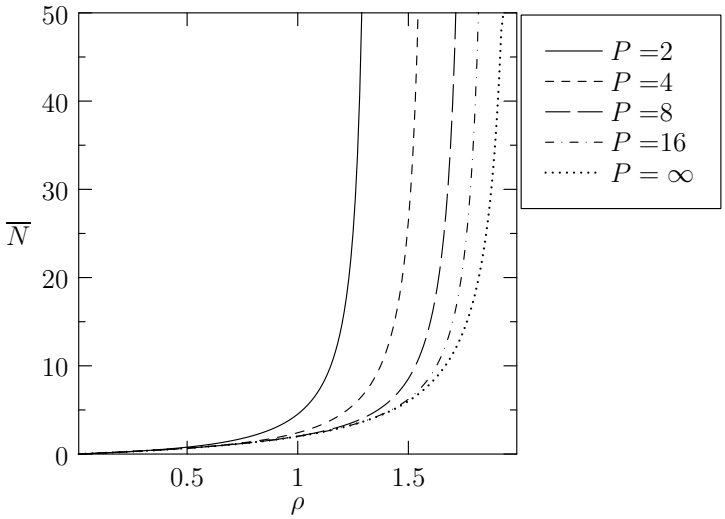


Fig. 7 Mean system occupancy versus ρ with $\omega = 0.5$, $\mu_1 = 1$ and $\mu_2 = 4$

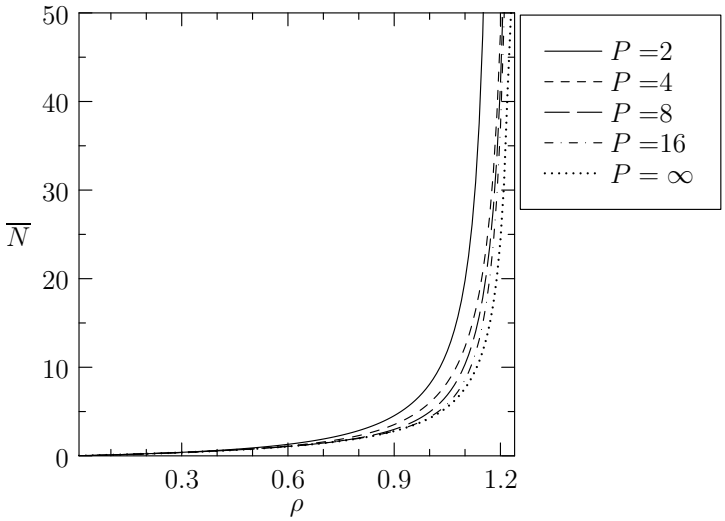


Fig. 8 Mean system occupancy versus ρ with $\omega = 0.8$, $\mu_1 = 1$ and $\mu_2 = 4$

where the system is stable versus the load balance ω as defined in equation (49). The impact of parameter P is the largest when we reach the maximum for ρ_{sup} at $\omega = \frac{1}{2}$ or when the system is well balanced. A well balanced system is a system where both customers introduce the same average amount of work in the system. If the system is completely out of balance the impact of P-gFCFS becomes negligible, which is also intuitively clear since we then approach a system with almost only one type of customers and thus a single server system.

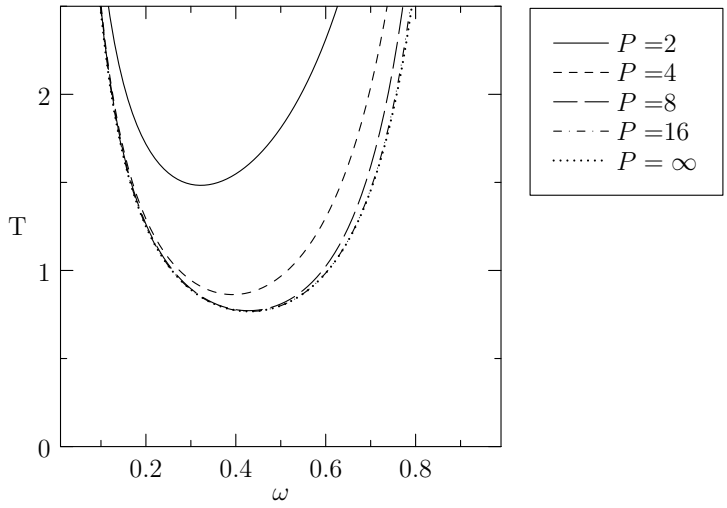


Fig. 9 Mean customer delay versus ω with $\rho = 1$, $\mu_1 = 1$ and $\mu_2 = 4$

Figures 7 and 8 show the impact of load (ρ) on the mean system occupancy for $\omega = 0.5$ and $\omega = 0.8$ respectively. Both figures show that for a small total load (ρ) the impact of P-gFCFS on the mean system occupancy becomes negligible. The impact of P-gFCFS becomes more and more noticeable when the total load increases. This is intuitively clear. In cases that the demand of the arrival stream is considerably less than what can be handled by one server, the question whether the second server is also active or not, is not very relevant. However, in cases that the demand of the arrival stream is close to or more than what can be handled by one server, the question whether the second server is also active or not, is very relevant. In these cases the impact of P-gFCFS becomes more noticeable. Comparison of both figures also confirms the impact of load balancing (ω) on the impact of P-gFCFS. The impact of P-gFCFS is considerably larger when the system is well balanced and when we consider large loads (ρ).

In Fig. 9 the mean customer delay versus ω with $\rho = 1$, $\mu_1 = 1$ and $\mu_2 = 4$ is shown. We see that a well-balanced system ($\omega = 0.5$) no longer gives the best result when we deal with a total load (ρ) smaller than the maximum throughput. A system where the fastest server gets a higher relative load performs better than the well balanced system. When P increases, the best performing system is a more balanced system. This is again intuitively clear. When P increases, the system approaches the system without a gFCFS service discipline and the system without a gFCFS service discipline performs best when the load is well-balanced [8].

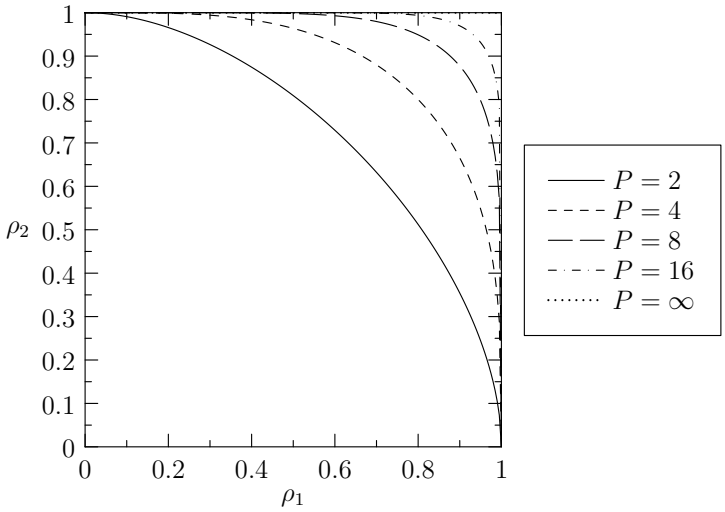


Fig. 10 Least upper bound of the set of ρ_2 values where the system is stable, for a given ρ_1 value

5.2 Impact of customers of one type on customers of the other type

Fig. 10 shows the influence of the (given) load of one type of customers on the sustainable load of the other type of customers. More precisely, we have plotted the least upper bound of the set of ρ_2 values where the system is stable, for a given ρ_1 value. Here, we see that for $P = 2$, ρ_1 has a huge impact on the maximum load ρ_2. This impact decreases when P becomes larger. In a road traffic context, this is the purpose of the turn lanes. We want to decrease the impact of the vehicles going to destination 1 on vehicles with destination 2 and vice versa.

5.3 Use for dimensioning purposes

In this subsection we focus on the practical application of the model. We consider some dimensioning possibilities of our results concerning the length of a turn lane.

Fig. 11 represents the probability that at least one customer is blocked at a random time instant by customers of the other type while his own server is idle (blockage probability) versus P with $\mu_1 = 1$, $\mu_2 = 2$ and $\sigma = 0.4$. In those cases, road capacity is wasted which should be avoided as much as possible. The blockage probability is given by

$$\text{Prob}\left[\text{Blockage}\right] = \sum_{n=P+1}^{\infty} \left(1 - \sigma^{n-P}\right) p(n,0) + \left(1 - (1-\sigma)^{n-P}\right) p(n,P) \quad (51)$$

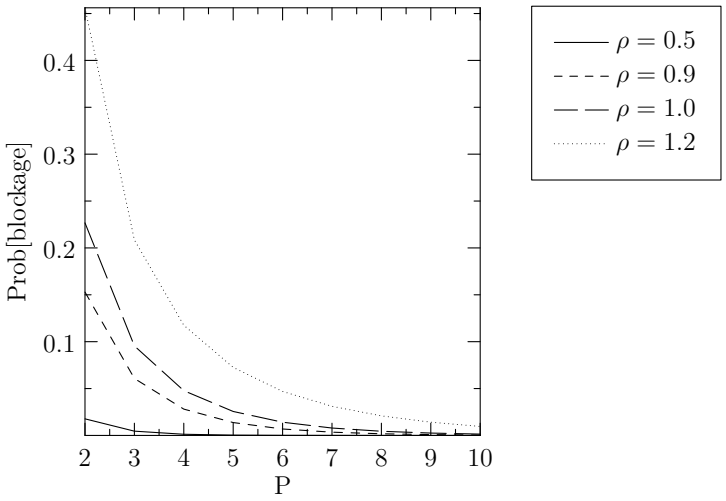


Fig. 11 Probability that at least one customer is blocked at a random time instant while his own server is idle versus P with $\mu_1 = 1$, $\mu_2 = 2$ and $\sigma = 0.4$

or in words, all the probabilities where all the leading customers are of the same type multiplied with the probability that not all customers in the system are of the same type. This blockage probability represents the impact vehicles have on vehicles with the other destination. This is an impact we want to reduce. As seen in Fig. 11, this blockage probability decreases with increasing P (increasing length of the turn lane). One possibility to dimension the turn lane is to determine a suitable threshold for the value of the blocking probability. If we choose, for example, the threshold value to be 0.05 then we see in Fig. 11 that for values of $\rho = 0.5, 0.9, 1.0, 1.2$, P should be $2, 4, 5$, or 7 respectively.

Another possibility to dimension the turn lane is to satisfy a condition for the queue length, for instance, the probability that the length of the queue is longer than a certain value is at most equal to a threshold value. This is important when a traffic jam caused by the blocking effect can spread to other junctions. For this purpose, we consider the adjusted system content, i.e., the system content keeping in mind that vehicles already using the turn lane do not add to the total queue length. For example, in Fig. 13, the queue length is 7, but the adjusted queue length is only 5. In Fig. 12, the tail probability of the adjusted system content with $\rho = 1$, $\mu_1 = 1$, μ_2 and $\sigma = 0.5$ is shown. In the example shown in Fig. 12, a turn lane with length 3 is required to meet the condition that the probability that the queue length is longer than 40 is not more than 10^{-5} where the pmf of the adjusted system content is calculated as

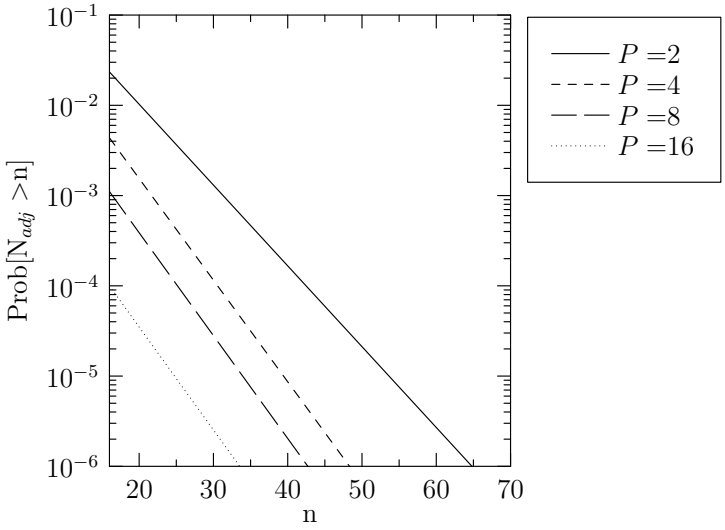


Fig. 12 The adjusted tail probability of the system contents with $\rho = 1$, $\mu_1 = 1$, $\mu_2 = 2$ and $\sigma = 0.4$

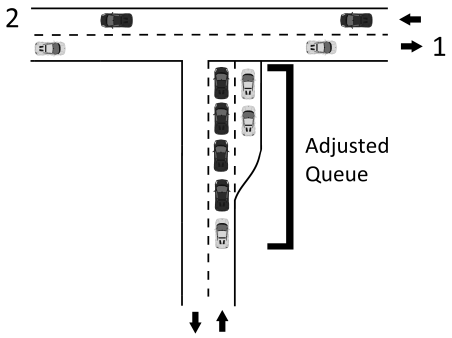


Fig. 13 Light grey vehicles with destination 1 and dark grey vehicles with destination 2 approaching a traffic junction

follows

$$
\begin{aligned}
p_{adj}(n) = {} & p(n,0) \\
& + \sum_{m=1}^{P-1} \left[\sum_{i=0}^{P-m-1} \left(\binom{m+i-1}{m-1} (1-\sigma)^i \sigma^m p(n+m+i,m) \right) \right. \\
& \left. + \sum_{i=0}^{m-1} \left(\binom{P-m+i-1}{P-m-1} \sigma^i (1-\sigma)^{P-m} p(n+P-m+i,m) \right) \right] \\
& + p(n,P), \qquad (52)
\end{aligned}
$$

for $n \geq P$ and where $\binom{m+i-1}{m-1}$ is the binomial coefficient. This formula can be understood as follows (terminology as in Fig. 13). The two separate lanes are blocked for further customers when the number of customers of one of the two types equals P. So if the number of customers of type 2 in the P leading customers is equal to m (and the number of customers of type 1 is $P-m$), the separate lanes are blocked from the m-th customer of type 1 or from the $P-m$-th customer of type 2 of the customers behind the leading customers, whichever comes first. The second line in (52) equals the probability corresponding with a blockage by a customer of type 1. This occurs, resulting in an adjusted length of n, if (i) the total number of customers equals $n+m+i$, (ii) $m-1$ customers of the first $m+i-1$ customers behind the leading customers are of type 1 (and $i \leq P-m-1$ are of type 2) and (iii) the next is of type 1. (ii) and (iii) lead to a negative binomial distribution and finally line 2 of formula (52). The third line is due to a blockage by a customer of type 2 and can be found similarly. The probabilities $p_{adj}(n)$ for $0 < n < P$ can be found analogously but are of less interest for dimensioning purposes.

The adjusted tail probability for $n \geq P$ is then given by

$$\begin{aligned} &\text{Prob}[N_{adj} > n] = \\ &\quad \text{Prob}[N > n] \\ &\quad - \sum_{i=1}^{P-1} \sum_{m=1}^{P-1} \sum_{y=max(0,i-m)}^{P-m-1} \left(\binom{y+m-1}{y} (1-\sigma)^y \sigma^m p(n+i,m) \right) \\ &\quad - \sum_{i=1}^{P-1} \sum_{m=1}^{P-1} \sum_{y=max(0,i-m)}^{m-1} \left(\binom{y+P-m-1}{y} (1-\sigma)^{P-m} \sigma^y p(n+i,m) \right) \end{aligned} \tag{53}$$

where $\binom{y+m-1}{y}$ and $\binom{y+P-m-1}{y}$ are binomial coefficients. We calculated the adjusted tail probability a little bit different. When considering $\text{Prob}[N > n]$, we have to deduct some probabilities (those cases that lead to $N_{adj} \leq n$) to get $\text{Prob}[N_{adj} > n]$.

6 Conclusions

In this paper, we introduced the concept of gFCFS service discipline with presorting (P-gFCFS), i.e., arriving customers are accommodated in one single FCFS queue, regardless of their type, with the exception of the first P customers for which the FCFS rule holds only within the type, i.e., customers of different types can overtake each other in order to be served as long as the overtaker is part of the first P customers. We have derived an expression for the steady-state distribution of the system occupancy. We have shown that for a small total load (ρ) the impact of P-gFCFS on the mean system occupancy becomes negligible. The impact of P-gFCFS becomes more and more

noticeable when the total load increases. When P increases, the best performing system is a more balanced system. Also, when P increases, the impact of one type of customer on the other type of customers decreases (which is the aim in a traffic context). We have also presented some interesting dimensioning possibilities concerning the length of a turn lane at an unsignalised intersection.

References

1. Baumann, H., Sandmann, W.: Numerical solution of level dependent quasi-birth-and-death processes. Procedia Computer Science **1**(1), 1561–1569 (2010)
2. Bruneel, H., Mélange, W., Steyaert, B., Claeys, D., Walraevens, J.: Influence of relative traffic distribution in nodes with blocking: an analytical model. In: 26th European Simulation and Modelling conference, Proceedings, pp. 136–143 (2012)
3. Bruneel, H., Mélange, W., Steyaert, B., Claeys, D., Walraevens, J.: Effect of global FCFS and relative load distribution in two-class queues with dedicated servers. 4OR **11**(4), 375–391 (2013)
4. Gail, H.R., Hantler, S.L., Taylor, B.A.: Use of characteristic roots for solving infinite state Markov chains. In: Computational Probability, pp. 205–255. Springer (2000)
5. Grassmann, W.: The use of eigenvalues for finding equilibrium probabilities of certain Markovian two-dimensional queueing problems. INFORMS Journal on Computing **15**(4), 412–421 (2003)
6. Heidemann, D.: A queueing theory approach to speed-flow-density relationships. In: Proceedings of the 13th International Symposium on Transportation and Traffic Theory, pp. 103–118 (1996)
7. Kleinrock, L.: Queueing Systems, Volume 1: Theory. Wiley-Interscience (1975)
8. Mélange, W., Bruneel, H., Steyaert, B., Walraevens, J.: A two-class continuous-time queueing model with dedicated servers and global FCFS service discipline. In: Analytical and Stochastic Modeling Techniques and Applications, *Lecture Notes in Computer Science*, vol. 6751, pp. 14–27 (2011)
9. Neuts, M.: Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach. The John Hopkins University Press (1981)
10. Phung-Duc, T., Masuyama, H., Kasahara, S., Takahashi, Y.: A simple algorithm for the rate matrices of level-dependent QBD processes. In: Proceedings of the 5th International Conference on Queueing Theory and Network Applications, pp. 46–52 (2010)
11. Sturm, J.: Mémoire sur la résolution des équations numériques, mém savans etrang edn. (1985)
12. Van Woensel, T., Vandaele, N.: Empirical validation of a queueing approach to uninterrupted traffic flows. 4OR **4**, 59–72 (2006)
13. Van Woensel, T., Vandaele, N.: Modeling traffic flows with queueing models: A review. Asia-Pacific Journal of Operational Research **24**, 435–461 (2007)
14. Vandaele, N., Van Woensel, T., Verbruggen, A.: A queueing based traffic flow model. Transportation Research D **5**(2), 121–135 (2000)
15. Wilkinson, J.H.: The Algebraic Eigenvalue Problem, vol. 87 (1965)

Conflict of interest

The authors declare that they have no conflict of interest.