Annals of Operations Research
https://doi.org/10.1007/s10479-021-03940-5

S.I. : QUEUESTOCHMOD2019

On two product form modifications for finite overflow systems

Nico van Dijk[1] · Barteld Schilstra[1]

Accepted: 8 January 2021

Abstract
Overflow mechanisms can be found in a variety of queueing models. This paper studies a simple and generic overflow system that allows the service times to be both job type and station dependent. This system does not exhibit a product form. To justify simple product form computations, two product form modifications are given, as by a so-called call packing principle and by a stop protocol. The provided proofs are self-contained and straightforward for the exponential case and of merit by itself. Next, it is numerically studied whether and when, or under which conditions, the modifications lead to a reasonable approximation of the blocking probability, if not an ordering. The numerical results indicate that call packing provides a rather accurate approximation when the overflow station is not heavily utilized. Moreover, when overflowed jobs have an equal or faster service rate, the approximation is consistently found to be pessimistic, which can be useful for practical purposes. The stop protocol, in contrast, appears to be less accurate for most natural situations. Nevertheless, for an extreme situation the order might change. In addition, for the stop protocol the product form is proven to be insensitive (i.e. to also apply for arbitrary non-exponential service times). For call packing, this numerically appears not to be the case, as of interest by itself. However, from a practical viewpoint the sensitivity seems light. The results are intriguing for both theoretical and practical further research.

Keywords Overflow queues · Call packing · Stop protocol · Product form · Blocking probability · Insensitivity

1 Introduction

Overflow queueing is a most natural phenomenon to let service requests be handled by an auxiliary or alternate or second preference service source when the primary service facility is congested or unavailable. Examples are found from classical alternate routing in communications up to daily life logistics, such as call centers (skill based routing), health care (see e.g. Asaduzzaman and Chaussalet 2014; Litvak et al. 2008) or emergency units (e.g.

✉ Nico van Dijk
n.m.vandijk@utwente.nl

[1] Stochastic Operations Research, Department of Applied Mathematics, University of Twente, PO box 217, 7500, AE Enschede, The Netherlands

Published online: 30 March 2021

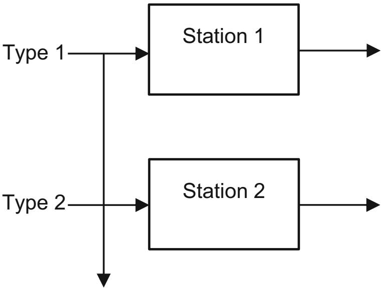

Fig. 1 Overflow model of interest in this paper

ambulances) from neighbouring areas. In this paper, we study a simple and generic overflow system consisting of two stations as depicted in Fig. 1. Arriving type 1 jobs are overflowed to station 2 if all servers at the primary station 1 are busy.

In most natural situations, type 1 and 2 jobs do not require the same amount of service, which means that their mean service times will be different (i.e. the service times are job type dependent). Moreover, type 1 jobs may also have a different mean service time if the jobs are overflowed to station 2. Therefore, the service times are also allowed to be station dependent. This can be of practical interest, such as for call centers, in which call center agents might be less or more suited to handle a specific call type. For example, the overflow servers may cause the call duration to be longer (e.g. by having to read a more general script), or just the opposite, they may represent more generalized or multiple skilled or advanced servers who will service faster, but probably at higher costs. Another example of an application area is health care (e.g. different specialized care units in which intensive care is to be provided).

1.1 Motivation and objective

For the overflow system of interest, no simple analytic expression for the joint steady-state distribution of the number of jobs in service or related performance measures (e.g. blocking or loss probabilities) seems to be available. One possible way to recover analytic solvability is to modify the system into a product form model (i.e. a model that does exhibit a product form). This paper mainly concerns two intuitively obvious modifications which turn the overflow system into a product form model. These modifications turn out to coincide with either of two concepts, the so-called call packing principle or the stop protocol. The call packing principle roughly states that jobs will be served at the station of highest preference, possibly by switching if capacity at a more preferred station becomes available. The stop protocol is a purely artificial modification which requires the service of overflowed type 1 jobs to be stopped if not all servers at station 1 are occupied.

The resulting product forms can be used to approximate specific performance measures for the original system, such as the mean number of busy servers or blocking probabilities. In this paper, the main focus will be on the blocking probability of type 1 jobs (i.e. the probability that a type 1 job finds both station 1 and 2 congested upon arrival). In particular, the primary objective is to study whether the call packing and stop protocol lead to a simple and reasonable approximation or even particularly to a secure upper bound for this blocking probability. Moreover, it is investigated whether the product forms and hence the expressions for the blocking probabilities are insensitive (i.e. only depending on the service time distribution through their means).

In short, the objective of the paper is threefold:

- to provide straightforward verifiable product form modifications for a generic overflow system with job type as well as station dependent characteristics,
- to numerically study to which extent these modifications lead to accurate approximations if not ordered bounds for the blocking probability of type 1 jobs,
- to examine the feature of insensitivity.

1.2 Outline

The structure of this paper is as follows. First, in Sect. 2 related literature is discussed. Then, in Sect. 3 the overflow model is presented more formally. The product form modifications is then proven for the exponential case in Sect. 4. Next, in Sect. 5 it is argued in which cases call packing and the stop protocol can be expected to provide a pessimistic approximation for the blocking probability of type 1 jobs. Subsequently, in Sect. 6 numerical results are given. The results are studied and discussed extensively as to accuracy and ordering. Finally, in Sect. 7 it is investigated whether the results remain valid for arbitrary (i.e. non-exponential) service distributions.

2 Related literature

In this section, literature on product forms, approximations and bounds and insensitivity related to overflow systems is discussed.

2.1 Overflow systems

Overflow systems are well known to be hard to solve. For one thing, as already detected in teletraffic engineering, overflow traffic at an overflow group will generally violate standard Poissonian arrival stream assumptions. For example, Van Doorn (1984) shows that the overflow stream from a standard Erlang loss system is hyperexponential. As a consequence, analytic results for overflow loss systems appear to be rather limited.

Nevertheless, in some cases analytic results can be obtained. For example, El-Taha and Heath (2000) derive the joint probabilities of the number of primary and secondary busy servers for an overflow system with two arrival streams. This overflow system is closely related to the system of interest in this paper, but there are some differences. One of these differences is that there is no direct arrival stream at the secondary server group, which is purely meant to serve overflowed jobs. Moreover, the service times are allowed to be station or server group dependent, but not job type dependent. Other types of overflow systems for which analytic results can be obtained include, for example, the overflow system under the assumption of call packing or the stop protocol. This is described more extensively in Sect. 2.3.

In order to determine specific performance measures for overflow systems, numerous approximation results have also been developed in early up to recent years (see e.g. Akimaru and Takahashi 1983; Borst et al. 1999; Brandt and Brandt 2001; Shortle 2004). Many of these originate from classic teletraffic engineering and are related to communications. These approximations can be based on, for example, moment approximations (e.g. Brandt and Brandt 2001) or an Equivalent Random Method (ERM) (e.g. Borst et al. 1999; Shortle 2004; Wilkinson 1956).

In this paper, another approach to approximate the performance measures, and in particular the blocking probability, is taken. This approach is to turn the overflow system into a product form model, which can then be used to approximate the performance measures for the original system. Moreover, this may even lead to bounds, as is more widely discussed in Sect. 2.4. However, before stepping into further detail for the overflow system of interest, as the area of product forms and related insensitivity results is huge, let us first place these phenomena in slightly more perspective in Sect. 2.2.

2.2 General product form approaches

Product form results have been established as based upon different concepts. A vast majority relies upon principles of partial balances for the global balance equations. Initiated by the pioneering work by Jackson (1957, 1963), this has already been made explicit in early papers as by Gordon and Newell (1967a, b) and Kingman (1969), who devotes a special section to partial balance. The well-known papers by Baskett et al. (1975) and Chandy and Martin (1983) also rely upon this approach. Partial balance roughly requires that the global balance equations can be decomposed and verified by specified subequations. These subequations generally have a natural interpretation of outrate and inrate equalities. Here, different levels or forms of partial balance might be distinguished, such as for each station, for a group of stations (e.g. Boucherie and Van Dijk 1993), for each job class or for each individual job separately (e.g. Van Dijk 2011).

A second process based approach is the well-known concept of quasi-reversibilty, as most elegantly displayed and made explicit in the famous book of Kelly (1979). It roughly preserves the Poissonian nature of arrivals and departures at specified service stages, so that service stations can be regarded as individual queues.

One common feature of partial balance and quasi-reversibility is that such results can be related to a characterization of reversibility. This can be either in direct form (as by Kelly 1979; Kingman 1969; Pittel 1979) or indirectly by a newly constructed process. For example, the constructed process can be an additional process, such as an adjoint process (e.g. Hordijk and Van Dijk 1983a; Van Dijk 2011), or an opposite or dual process by considering flows (as of holes) in reversed direction (e.g. Harrison 2004). The reversibility, in turn, has the appealing property that it can be checked in different ways, such as by path invariance or Kolmogorov's criterion (e.g. Hordijk and Ridder 1987, 1988; Kelly 1979).

Also worthwhile to mention is the viewpoint and product form result developed by Boucherie (1994) since it may find an application in the context of this paper (see Remark 1). In this reference, the equilibrium distribution for a product process of a collection of Markov chains competing over resources is given. Here, the state of a Markov chain determines which resource it is using. Two or more Markov chains are then competing over a resource when they cannot simultaneously use that resource. Therefore, as soon as one of the competing Markov chains starts using a specific resource, the other Markov chains that are competing over this resource are frozen. In Appendix B.2, this competition mechanism is further explored for its possible application to the system of interest.

A similar line of thought holds for a characterization in the work of Harrison (2004). In this reference, the time-reversed process is studied. A Markovian process algebra (MPA) description is used, and a generalization of the Reversed Compound Agent Theorem (RCAT) is applied in order to determine the equilibrium state probabilities of interacting Markov processes. The same approach is also outlined and made more practical by Balsamo et al. (2010). Several network examples of serial (hence non-reversible routing) and parallel structures are contained. In the latter case, the parallel routing probabilities are fixed, which means state independent. It does therefore not directly seem to cover overflow as by state dependent routing and different service rates.

To conclude the introduction on general product form approaches, it is important to briefly mention the aspect of finite capacity constraints. In queueing networks, these constraints generally destroy analytic feasibility in terms of product forms with two major exceptions:

- The routing from one station to another is reversible (see e.g. Kelly 1979; Kingman 1969; Pittel 1979).
- A skipping blocking mechanism is assumed. This means that a customer or job 'skips' a station or service center if there is no capacity available. For networks, this has already extensively been studied by Pittel (1979) (see also e.g. Balsamo et al. 2010).

To a certain extent, a simple "open single service network" can be seen as by a reversible routing with the exterior. From this perspective, product form results have also been expected for networks with internal multiple stages but total population constraints, as shown more detailed by, for example, Kaufman (1981) and Lam (1977).

The overflow system as studied in this paper does not fit in either of the two exceptions, but it is indirectly related. Clearly, when no access is feasible at all, the request is lost, which can be seen as skipping. However, when station 1 is congested and a server at station 2 is available, instead of skipping, overflow of type 1 jobs takes place. As discussed in Sect. 4, for these overflowed jobs a natural (and reversible) partial balance interpretation of outflow is equal to inflow is necessarily violated. An adaptation to make it product form solvable will thus be required. This, in turn, will lead to modifications as will be discussed in Sect. 2.3 and specified analytically in Sect. 4.

2.3 Overflow systems and product forms

For the overflow system of interest, the joint steady-state distribution of the number of jobs in service cannot be expected to have a product form solution (see Sect. 4). However, a product form can be obtained when either of the following protocols is assumed:

- A call packing protocol
- A stop protocol

Both these protocols have already been associated with product form results in literature. First of all, the call packing principle has long been known in the area of telecommunications and is well known to lead to a product form solution. For example, product form results for call packing networks have already been provided by Berry and Henderson (1989) and Henderson and Taylor (1988) (see also Remark 1). Besides that, it has been shown by Van Dijk and Van der Sluis (2009) that both call packing and the stop protocol lead to a product form result. The system that is studied in this reference is similar to the overflow system of interest in the present paper. As main difference, the present paper allows that the mean service times for type 1 jobs at station 1 and overflowed type 1 jobs at station 2 are not necessarily equal (i.e. the service parameters are allowed to be station dependent). In addition, the overflow station will be restricted to a so-called coordinate convex set, as explained in Sect. 3.

As mentioned above, a product form solution for the joint steady-state distribution of the number of jobs in the overflow system of interest can be derived if either call packing or the stop protocol is assumed. To the best of the authors' knowledge, it seems that these product forms have not been proven directly or reported explicitly, although they could implicitly be concluded from literature (e.g. see Remark 1). In this paper, easily verifiable and purely self-contained proofs of the product forms are given. These proofs are implicitly based on partial balance. For call packing, partial balance is shown at station and class level, which means that the mean service rates can be allowed to differ at these levels. For the stop protocol, it is possible to show partial balance for each individual job, which is also referred to as job-local-balance (see e.g. Hordijk and Van Dijk 1983a). This principle is known to be directly related to the concept of insensitivity, as is further discussed in Sect. 2.5.

Finally, it is noted that call packing could be applicable in two ways, either as natural part of the system (e.g. Berry and Henderson 1989; Henderson and Taylor 1988) or as modification to obtain an approximation or bound (e.g. Van Dijk and Van der Sluis 2009). The stop protocol, in contrast, is merely meant to be seen as a purely artificial modification. For the stop protocol, the resulting performance measures should therefore only be considered as an approximation or bound (e.g. Hordijk and Ridder 1987, 1988). In this paper, both protocols are specifically applied as modifications in order to obtain approximations or even ordered estimates or bounds for the blocking probability of type 1 jobs for the original overflow system. This is further discussed in Sect. 2.4.

2.4 Product form approximations and bounds

In Sect. 2.3, it is discussed that the overflow system of interest can be turned into a product form model by either of two modifications, which are referred to as call packing and stop protocol. These modifications might be useful when the original overflow system (i.e. without call packing or stop protocol) is analyzed. For example, as mentioned in Sect. 1.1, several performance measures for the original system can be approximated by using the resulting product forms. In fact, in some cases this may lead to a bound for a specific performance measure of interest.

More specifically, it has already been shown by Van Dijk and Van der Sluis (2009) that call packing leads to an upper bound for the blocking probability of type 1 jobs if the service times are only job type dependent (i.e. $\gamma = \mu_1$, using the notation that is introduced in Sect. 3). Furthermore, it is proven by Van Dijk (1989) that the stop protocol leads to an insensitive upper bound for the blocking probability of type 1 jobs for a pure overflow system, that is, without arrival stream of type 2 jobs (i.e. $\lambda_2 = 0$). Also worthwhile to mention is the work of Hordijk and Ridder (1987, 1988). In this reference, the overflow system with only station dependent parameters (i.e. $\gamma = \mu_2$) is considered. The authors suggest a different modification, which leads to the same expression for the blocking probability as the stop protocol. It is shown that this expression provides an insensitive upper bound for the blocking probability.

In this paper, the more general case of both job type and station dependent service times is considered. For this more general case, it seems to be an open question under which conditions call packing and the stop protocol lead to a bound for the blocking probability of type 1 jobs. In this paper, this will be studied numerically. Hence, as opposed to the references mentioned above, no formal proof will be provided (see also Remark 5). Instead, intuitive arguments are given to determine in which cases the modifications can be expected to lead to a pessimistic approximation (in the sense that the blocking probability for the original system will be smaller). Moreover, numerical experiments are performed to provide support of the intuitive arguments. This may indicate under which conditions on the parameters the modifications could be expected to provide an upper bound for the blocking probability of type 1 jobs. The ordering results that are obtained are discussed in Sect. 6.

2.5 Insensitivity

Another aspect, which is only briefly mentioned yet, concerns the feature of insensitivity. This feature is well known to be highly related to product form results. It implies that the product form is independent of the service distributional forms other than determined by their means (e.g. Barbour 1976; Hordijk and Van Dijk 1983b; Taylor 2011). This appealing property is known to be if and only if related to detailed notions of partial balance as per fixed position or individual job (e.g. Hordijk and Van Dijk 1983a; Schassberger 1977, 1978). In the first reference, such balance is specifically referred to as job-local-balance.

For the overflow system of interest, the insensitivity feature is also studied as an intriguing phenemenon. For the stop protocol, as mentioned in Sect. 2.3, a balance per position implicitly appears to apply. Therefore, an insensitivity result can be expected. A more technical, but self-contained proof of this result is included in Sect. 7.2. It is shown that the product form remains valid for the general class of Erlang mixtures, which can be argued to represent all non-negative distributions.

For call packing, in contrast, strict insensitivity appears not to hold, as illustrated by numerical counterexamples in Sect. 7.1. From a practical point of view, however, at least for the simulated situations, the effect appears to be light. Nevertheless, such a sensitive product form result seems uncommon, if not unreported. It is in line, though, with earlier queueing results for classical ideal gradings in telephony. It also does not conflict with the insensitivity results in the work of Burman et al. (1984) and Conway (1989), who both study communication networks and exclude alternative or overflow routing. The multi-server product form sensitivity, even though light, can be regarded as of interest in itself. Particularly, from a theoretical point of view, sensitivity error bounds would be appealing for future research.

3 Model

The overflow model that is considered in this paper is depicted in Fig. 1. Two types of jobs are distinguished. Type 1 jobs arrive according to a Poisson process with arrival rate λ_1 at station 1, which has a finite number of N_1 servers. When station 1 is congested, arriving type 1 jobs are rerouted to station 2. Moreover, type 2 jobs arrive at station 2 according to a Poisson process with arrival rate λ_2. Upon arrival at station 2, overflowed type 1 jobs and arriving type 2 jobs are assigned to a server, provided they are accepted. Here, as an extension of just a common constraint of a finite number of servers, a more general admission interaction between the two job types is also allowed. More precisely, let $(\mathbf{n}, m) = (n_1, n_2, m)$ denote the state of the system, where

n_1 is the number of type 1 jobs at station 1,
n_2 is the number of type 2 jobs at station 2,
m is the number of overflowed type 1 jobs at station 2.

Then, station 2 is restricted to a coordinate convex set $\boldsymbol{C}$ as characterized by:

$$(n_2, m) \in \boldsymbol{C} \Rightarrow \begin{cases} (n_2 - 1, m) \in \boldsymbol{C} & (n_2 > 0) \\ (n_2, m - 1) \in \boldsymbol{C} & (m > 0) \end{cases} \tag{1}$$

An overflowed type 1 job is then accepted at station 2 if $(n_2, m + 1) \in \boldsymbol{C}$ and a type 2 job is accepted at station 2 if $(n_2 + 1, m) \in \boldsymbol{C}$. If this is not the case, the overflowed type 1 job or arriving type 2 job is rejected and lost.

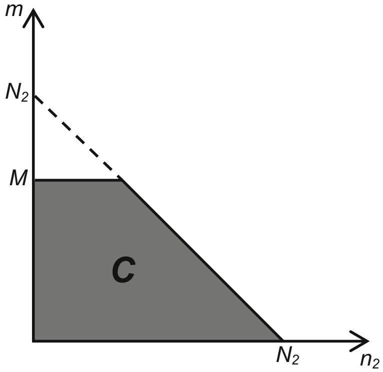

Fig. 2 Coordinate convex example for station 2

An example of a coordinate convex structure is the natural situation in which station 2 has just a fixed number of N_2 servers that can be used by either type 1 or 2 jobs. In this case, we have that $\boldsymbol{C} = \{(n_2, m)|n_2 + m \leq N_2\}$. But special 'reservation' schemes are conceivable as well. For example, it is also possible to choose $\boldsymbol{C}$ as illustrated in Fig. 2. In that case, type 1 jobs can use at most M servers at station 2, so that $N_2 - M$ servers are exclusively kept for type 2 jobs.

Finally, the service times at a server are assumed to be exponential with parameters μ_1 for type 1 jobs at station 1, μ_2 for type 2 jobs at station 2 and γ for overflowed type 1 jobs at station 2.

4 Analytic solution

For the system as described in Sect. 3, a notion of class balance for overflowed type 1 jobs is necessarily violated, as illustrated more detailed by Van Dijk and Van der Sluis (2009) for the special case $\gamma = \mu_1$. In short: when there is an available server at station 1, the services of overflowed type 1 jobs still continue (positive outrate), while they could not enter (zero inrate). Therefore, no product form solution for the joint steady-state distribution of the number of jobs in the system can be expected (cf. Chandy and Martin 1983; Van Dijk 2011). Sects. 4.1 and 4.2 discuss two product form modifications which can be used to repair this rate inconsistency and recover analytic solvability.

4.1 Call packing

The first modification that can be suggested to recover analytic solvability is the following call packing principle.

Definition 1 *(Call packing)* An overflowed type 1 job at station 2 will be switched to station 1 when a server at station 1 becomes available.

Theorem 1 *Under the assumption of call packing and with c a normalizing constant, for all states* $(\mathbf{n}, m)$ *with* $n_1 \in \{0, ..., N_1\}$ *and* $(n_2, m) \in \boldsymbol{C}$ *the steady-state distribution* π_{cp} *is given*

Table 1 Proof balances

For $n_1 < N_1$	For $n_1 = N_1,\ m = 0$	For $n_1 = N_1,\ m > 0$
(4.1) = (4.7)	(4.1) = (4.7)	(4.1) + (4.2) = (4.8)
(4.3) = (4.9)	(4.3) = (4.9)	(4.3) = (4.9)
(4.4) = (4.10)	(4.5) = (4.11) + (4.12)	(4.5) = (4.11) + (4.12)
(4.6) = (4.13)	(4.6) = (4.13)	(4.6) = (4.13)

by:

$$\begin{aligned}
&\pi_{cp}(n_1, n_2, m) = cF(m) \prod_{i=1,2} \frac{1}{n_i!} \left(\frac{\lambda_i}{\mu_i}\right)^{n_i}, \quad \textit{with} \\
&F(m) = \begin{cases} \lambda_1^m / \prod_{k=1}^m (N_1\mu_1 + k\gamma) & m > 0 \\ 1 & m = 0 \end{cases}
\end{aligned} \tag{2}$$

Proof First, it is noted that the set of admissible states $\boldsymbol{S}$ is restricted to:

$$\boldsymbol{S} = \{(n_1, n_2, m) | 0 \le n_1 < N_1, m = 0, (n_2, 0) \in \boldsymbol{C} \text{ or } n_1 = N_1, (n_2, m) \in \boldsymbol{C}\} \tag{3}$$

For each $(\mathbf{n}, m) \in \boldsymbol{S}$, we need to verify the global balance equations. Here, beforehand, it is mentioned that terms are organized for the different interpretations and the verification by 'detailed' balances below. The global balance equations can then be written as follows:

$$\begin{array}{ll}
\left\{\begin{array}{l}
\pi_{cp}(n_1, n_2, m) n_1 \mu_1 1_{\{n_1>0\}} + \\
\pi_{cp}(n_1, n_2, m) m\gamma 1_{\{m>0\}} + \\
\pi_{cp}(n_1, n_2, m) n_2 \mu_2 1_{\{n_2>0\}} + \\
\pi_{cp}(n_1, n_2, m) \lambda_1 1_{\{n_1<N_1\}} + \\
\pi_{cp}(n_1, n_2, m) \lambda_1 1_{\{n_1=N_1\}} 1_{\{(n_2,m+1)\in \boldsymbol{C}\}} + \\
\pi_{cp}(n_1, n_2, m) \lambda_2 1_{\{(n_2+1,m)\in \boldsymbol{C}\}}
\end{array}\right\} &
\begin{array}{l} (4.1) \\ (4.2) \\ (4.3) \\ (4.4) \\ (4.5) \\ (4.6) \end{array} \\
= & \qquad (4) \\
\left\{\begin{array}{l}
\pi_{cp}(n_1 - 1, n_2, m) \lambda_1 1_{\{n_1>0\}} 1_{\{m=0\}} + \\
\pi_{cp}(n_1, n_2, m - 1) \lambda_1 1_{\{n_1=N_1\}} 1_{\{m>0\}} + \\
\pi_{cp}(n_1, n_2 - 1, m) \lambda_2 1_{\{n_2>0\}} + \\
\pi_{cp}(n_1 + 1, n_2, m)(n_1 + 1) \mu_1 1_{\{n_1<N_1\}} + \\
\pi_{cp}(n_1, n_2, m + 1) N_1 \mu_1 1_{\{n_1=N_1\}} 1_{\{(n_2,m+1)\in \boldsymbol{C}\}} + \\
\pi_{cp}(n_1, n_2, m + 1)(m + 1) \gamma 1_{\{n_1=N_1\}} 1_{\{(n_2,m+1)\in \boldsymbol{C}\}} + \\
\pi_{cp}(n_1, n_2 + 1, m)(n_2 + 1) \mu_2 1_{\{(n_2+1,m)\in \boldsymbol{C}\}}
\end{array}\right\} &
\begin{array}{l} (4.7) \\ (4.8) \\ (4.9) \\ (4.10) \\ (4.11) \\ (4.12) \\ (4.13) \end{array}
\end{array}$$

By substituting (2), the global balance equations (4) are satisfied by more detailed equalities as specified in Table 1. Here, the special role of the indicators in the left and right hand sides of these balances is noted. Also note that all unmentioned equations are equal to 0.

The proof is hereby completed. □

Remark 1 *(Proof)* The overflow system of interest allows that $\gamma \neq \mu_1$, which means that overflowed type 1 jobs can have a different mean service time (e.g. an accelerated service speed, say at higher costs). It can be noted that this differs from the call packing systems that are discussed by Henderson and Taylor (1988) and Van Dijk and Van der Sluis (2009), in which the mean service rate of overflowed jobs is kept identical. Here, it is mentioned that in Example 2 in the former reference type 1 calls are sped up if the number of type 1 calls present exceeds a predetermined limit N^*, which could be chosen equal to N_1. However, this concerns all type 1 calls instead of only the overflowed type 1 calls. Nevertheless, the product form (2) could implicitly be concluded from this reference, as illustrated in Appendix B.1. Moreover, another approach could be to model the overflow system of interest in the framework of Boucherie (1994), that is, by seeing it as competing Markov chains. For its insight, this is illustrated in Appendix B.2 for the case without type 2 jobs. The present proof is easily verifiable and purely self-contained. In simplicity, it relies upon balance for each job class separately. As such, it can be seen as of merit in its own right.

Remark 2 *(Coordinate convex structure)* The inclusion of a coordinate convex structure at one station is also worthwhile mentioning. Such product form structures have generally been reported as by Burman et al. (1984), Kaufman (1981), Lam (1977) and Van Dijk (2011) on total population constraints for the entire network. These exclude multiple stations or routing with call packing (e.g. Burman et al. (1984) state: "this precludes alternate or hierarchical routing"). Condition (g) of Henderson and Taylor (1988) can be seen as a directly related condition with its proof relying upon the paper by Chandy and Martin (1983).

Remark 3 *(Call packing: practical?)* The term call packing (or repacking) originates from the area of telecommunications (see e.g. Berry and Henderson 1989; Henderson and Taylor 1988), which may illustrate its practical and appealing interest for computation. Clearly, also in the aforementioned applications of skill based routing in call centers or specialized care units within health care a practical applicability seems well conceivable depending on the actual availabilities and protocols.

4.2 Stop protocol

The second intuitive product form modification could be referred to as conservative (e.g. Hordijk and Van Dijk 1983a; Van Dijk 1993) or stop (e.g. Van Dijk 1993) protocol and is stated as follows.

Definition 2 *(Stop protocol)* When a server at station 1 (i.e. a primary preferred server for type 1 jobs) becomes available, preemptively stop the servicing of overflowed type 1 jobs at station 2. These services are only resumed whenever station 1 becomes and stays saturated.

Theorem 2 *Under the stop protocol and with c a normalizing constant, for all states* $(\mathbf{n}, m)$ *with* $n_1 \in \{0, ..., N_1\}$ *and* $(n_2, m) \in \boldsymbol{C}$*, the steady-state distribution* π_s *is given by:*

$$\pi_s(n_1, n_2, m) = c \prod_{i=1,2} \frac{1}{n_i!}\left(\frac{\lambda_i}{\mu_i}\right)^{n_i} \frac{1}{m!}\left(\frac{\lambda_1}{\gamma}\right)^m \tag{5}$$

Proof First, it is noted that the set of admissible states $\boldsymbol{S}$ is now restricted to:

$$\boldsymbol{S} = \{(n_1, n_2, m) | 0 \leq n_1 \leq N_1,\ (n_2, m) \in \boldsymbol{C}\} \tag{6}$$

As in the proof of Theorem 1, the global balance equations need to be verified for each $(\mathbf{n}, m) \in \boldsymbol{S}$. Again, beforehand, it is mentioned that terms are organized for the different interpretations and the verification by 'detailed' balances as specified below. The global balance equations can then be written as follows:

$$
\left\{\begin{array}{l}
\pi_s(n_1, n_2, m) n_1 \mu_1 1_{\{n_1>0\}} + \\
\pi_s(n_1, n_2, m) m \gamma 1_{\{n_1=N_1\}} 1_{\{m>0\}} + \\
\pi_s(n_1, n_2, m) n_2 \mu_2 1_{\{n_2>0\}} + \\
\pi_s(n_1, n_2, m) \lambda_1 1_{\{n_1<N_1\}} + \\
\pi_s(n_1, n_2, m) \lambda_1 1_{\{n_1=N_1\}} 1_{\{(n_2, m+1)\in \boldsymbol{C}\}} + \\
\pi_s(n_1, n_2, m) \lambda_2 1_{\{(n_2+1, m)\in \boldsymbol{C}\}}
\end{array}\right\}
\qquad
\begin{array}{r}
(7.1)\\ (7.2)\\ (7.3)\\ (7.4)\\ (7.5)\\ (7.6)
\end{array}
$$

$$
= \qquad (7)
$$

$$
\left\{\begin{array}{l}
\pi_s(n_1 - 1, n_2, m) \lambda_1 1_{\{n_1>0\}} + \\
\pi_s(n_1, n_2, m - 1) \lambda_1 1_{\{n_1=N_1\}} 1_{\{m>0\}} + \\
\pi_s(n_1, n_2 - 1, m) \lambda_2 1_{\{n_2>0\}} + \\
\pi_s(n_1 + 1, n_2, m)(n_1 + 1) \mu_1 1_{\{n_1<N_1\}} + \\
\pi_s(n_1, n_2, m + 1)(m + 1) \gamma 1_{\{n_1=N_1\}} 1_{\{(n_2, m+1)\in \boldsymbol{C}\}} + \\
\pi_s(n_1, n_2 + 1, m)(n_2 + 1) \mu_2 1_{\{(n_2+1, m)\in \boldsymbol{C}\}}
\end{array}\right\}
\qquad
\begin{array}{r}
(7.1)'\\ (7.2)'\\ (7.3)'\\ (7.4)'\\ (7.5)'\\ (7.6)'
\end{array}
$$

It is noted that the indicator $1_{\{n_1=N_1\}}$ is included in (7.2) and (7.5)′ because of the stop protocol. As a consequence, we have that the indicator functions in (7.i) are equal to those in (7.i)′ for each $i = 1, ..., 6$. Next, by substituting (5), it is verified that the global balance equations (7) are satisfied by the more detailed equalities (7.i)=(7.i)′, $i = 1, ..., 6$.

The proof is hereby completed. □

Remark 4 *(Proof)* Indirect proofs can be concluded from general results as in the work of Hordijk and Van Dijk (1983a) as based on specific notions of station balance or job-local-balance. Moreover, in Sect. 7.2 a detailed proof is provided for the more general case of non-exponential service times. Nevertheless, for illustrating the distinction with call packing and self-containedness, a direct proof is provided for the case of exponential service times in Theorem 2 as well.

5 Blocking probability

Several performance measures can be derived from the steady-state distributions (2) and (5), such as the mean number of busy servers, the throughput and loss or blocking probabilities. For example, with the coordinate convex set $\boldsymbol{C}$ as illustrated in Fig. 2, the blocking probability of type 1 and 2 jobs (denoted by B_1 and B_2, respectively) for the call packing system can be found as follows:

$$
B_1 = \sum_{n_2=0}^{N_2} \pi_{cp}(N_1, n_2, \min\{M, N_2 - n_2\}) \qquad (8)
$$

$$
B_2 = \sum_{n_1=0}^{N_1} \pi_{cp}(n_1, N_2, 0) + \sum_{m=1}^{M} \pi_{cp}(N_1, N_2 - m, m) \qquad (9)
$$

Similarly, B_1 and B_2 for the system with stop protocol are given by:

$$
B_1 = \sum_{n_2=0}^{N_2} \pi_s(N_1, n_2, \min\{M, N_2 - n_2\}) \qquad (10)
$$

$$B_2 = \sum_{n_1=0}^{N_1} \sum_{m=0}^{M} \pi_s(n_1, N_2 - m, m) \tag{11}$$

In what follows, the blocking probability of type 1 jobs is taken as our performance measure of primary interest. In particular, it is studied to what extent the blocking probability of type 1 jobs for the original system (i.e. without call packing or the stop protocol) can be well approximated by the expressions for B_1 in (8) and (10). Here, it would be of particular interest if the modifications always provide a pessimistic approximation (in the sense that the blocking probability for the original system will be smaller) and thus an upper bound.

For call packing, this is true if $\gamma = \mu_1$, as formally proven by Van Dijk and Van der Sluis (2009). This is also intuitively supported since without call packing overflowed jobs use (one may speak of 'steal') capacity from type 2 jobs, even while own capacity at station 1 is available. In this way, this capacity is kept exclusively available for newly arriving type 1 jobs. The same intuition also holds for any $\gamma \neq \mu_1$. In this case, however, the change of service speed after switching from station 2 to station 1 also affects the blocking probability of type 1 jobs.

$\gamma > \mu_1$: In the call packing system, an overflowed type 1 job switches from a faster server at station 2 to a slower server at station 1 if a server at station 1 becomes available. It is thus intuitively obvious that the slower servers will be used more often than in the original system (i.e. without call packing). This leads to a longer mean sojourn time of type 1 jobs in the call packing system. Consequently, call packing can be expected to give a pessimistic approximation of the blocking probability of type 1 jobs if $\gamma \geq \mu_1$.

$\gamma < \mu_1$: Similarly, in this case the faster servers will be used more often in the call packing system than in the original system. This means that the mean sojourn time of type 1 jobs will be shorter and consequently the blocking probability might be smaller than that for the original system. Therefore, no secure upper or lower bound for the blocking probability of type 1 jobs can be expected if $\gamma < \mu_1$, as also visible in Fig. 3.

Finally, in most cases it can be expected that the blocking probability of type 1 jobs for the system with stop protocol will be larger than for the original system. This is intuitively clear because under the stop protocol overflowed type 1 jobs will occupy the servers at station 2 for a longer time. However, if the service of overflowed jobs is substantially faster than the service of type 2 jobs ($\gamma \gg \mu_2$), the stop protocol might even lead to a smaller blocking probability.

The intuitive reasoning for this is as follows. In the system with stop protocol, overflowed type 1 jobs at station 2 will be stopped when station 1 is not saturated. This can also be seen as if the servers at station 2 that are occupied by overflowed type 1 jobs are 'reserved' as long as type 1 jobs can be served at station 1 (i.e. there are servers at station 1 available). Obviously, these 'reserved' servers do not become available immediately after station 1 becomes congested, but after a relatively short service time of on average $1/\gamma$. However, this may still be beneficial since it may prevent these servers to be taken by relatively much slower type 2 jobs during the time that station 1 is not fully occupied.

As a consequence, in such special cases the blocking probability of type 1 jobs for the system with stop protocol might be smaller than for the original system, as illustrated in Fig. 7. On the other hand, since such a situation can only occur if γ is larger than μ_2, it can be expected that the stop protocol leads to a pessimistic approximation of the blocking probability of type 1 jobs if $\gamma \leq \mu_2$.

Remark 5 *(Formal proof bounds)* For special cases of the overflow system as depicted in Fig. 1, both call packing and the stop protocol have already been suggested and shown to provide an upper bound for the blocking probability of type 1 jobs, as more extensively discussed in Sect. 2.4. However, for the more general case with job type as well as station dependent service times, as dealt with in this paper, it seems unknown under which conditions on the parameters the modifications provide a bound for the blocking probability of type 1 jobs. From the intuitive arguments in this section, it is conjectured that call packing leads to an upper bound for the blocking probability of type 1 jobs if $\gamma \geq \mu_1$ and the stop protocol if $\gamma \leq \mu_2$. This is supported by all numerical experiments that are performed (see Sect. 6). Nevertheless, a formal proof that the modifications provide an upper bound for the blocking probability of type 1 jobs (under the aforementioned or even less strict conditions) would still be of interest. This remains a challenging point for future research.

6 Numerical results

This section contains some numerical results for the blocking probability of type 1 jobs. The blocking probabilities for the original system are calculated from the steady-state probabilities which are determined using the Grassmann-Taksar-Heyman (GTH) algorithm (see e.g. Stewart 2009, chapter 10). Besides that, the blocking probabilities for the call packing system and for the system with stop protocol are computed using the expressions for B_1 as given in (8) and (10), respectively.

Finally, for reference, the results of an approximation using Erlang loss expressions are also given. First of all, let L_1 be the blocking probability for an Erlang loss system with arrival rate λ_1 and mean service time $1/\mu_1$. Furthermore, let L_2 be the blocking probability for an Erlang loss system with two types of arrivals, type 2 jobs with arrival rate λ_2 and mean service time $1/\mu_2$ and type 1 jobs with arrival rate $L_1\lambda_1$ (the loss rate of the first station) and mean service time $1/\gamma$ (or, equivalently, the blocking probability for an Erlang loss system with arrival rate $L_1\lambda_1 + \lambda_2$ and mean service time $\frac{L_1\lambda_1}{L_1\lambda_1+\lambda_2}\frac{1}{\gamma} + \frac{\lambda_2}{L_1\lambda_1+\lambda_2}\frac{1}{\mu_2}$). Combining these blocking probabilities leads to the approximation $L_1 \cdot L_2$, which will be referred to as the Erlang loss approximation. The approximation is more likely to be optimistic (in the sense that the blocking probability for the original system will be larger) since it samples overflow at random times instead of during busier periods.

6.1 Low utilization for station 2

In Table 2, the parameter values for numerical experiment 1 are given. It is noted that this example corresponds to a quite natural situation. The servers at station 1 are heavily utilized (utilization of 100%), while station 2 has a relatively low utilization of 40%, so that overflow is reasonable. Fig. 3 shows the results of the experiment. A noteworthy observation in this example is that the blocking probability of type 1 jobs for the call packing system can both be smaller ($\gamma = 1$) and larger ($\gamma = 2$) than for the original system if $\gamma < \mu_1$. On the other hand, call packing is found to provide a pessimistic approximation if $\gamma \geq \mu_1$. It can also be noted that call packing leads to a far more accurate approximation than the stop protocol. However, this is not generally the case, as illustrated by numerical experiments 3 and 4. Finally, it can be observed that the Erlang loss approximation leads to an underestimation of the blocking probability.

The parameter values and results of numerical experiment 2 are given in Table 3 and Fig. 4. This experiment shows how the blocking probability is affected if not all servers at station 2 can be used to serve overflowed type 1 jobs (i.e. $M < N_2$). In this example,

Fig. 3 Results of numerical experiment 1

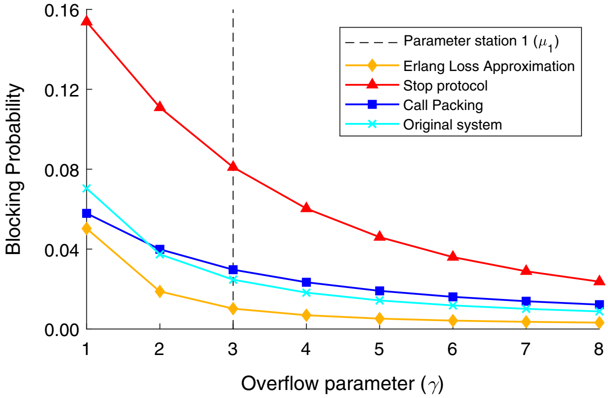

Table 2 Parameter values of numerical experiment 1

λ_1	λ_2	μ_1	μ_2	γ	N_1	N_2	M
30	20	3	5	–	10	10	10

Fig. 4 Results of numerical experiment 2

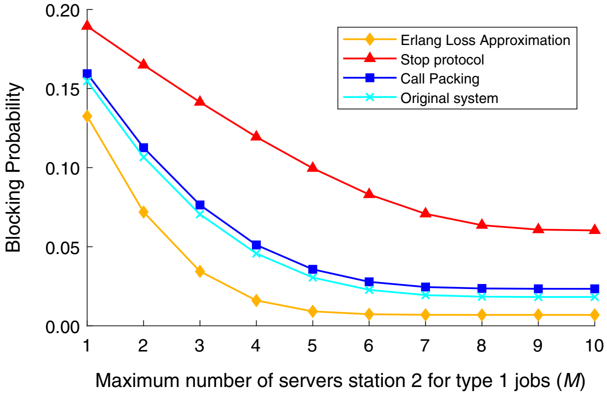

Table 3 Parameter values of numerical experiment 2

λ_1	λ_2	μ_1	μ_2	γ	N_1	N_2	M
30	20	3	5	4	10	10	–

call packing appears to provide an accurate approximation, which lies slightly above the blocking probability for the original system. The stop protocol also leads to a pessimistic approximation, although it is not nearly as accurate as the call packing approximation. Furthermore, the decreasing marginal effect of making extra servers at station 2 available for serving overflowed type 1 jobs is interesting to note.

6.2 High utilization for station 2

The numerical experiments in Sect. 6.1 considered situations with a low utilization for station 2. In order to study the effect of the workload at station 2 on the performance of the approx-

Fig. 5 Results of numerical experiment 3

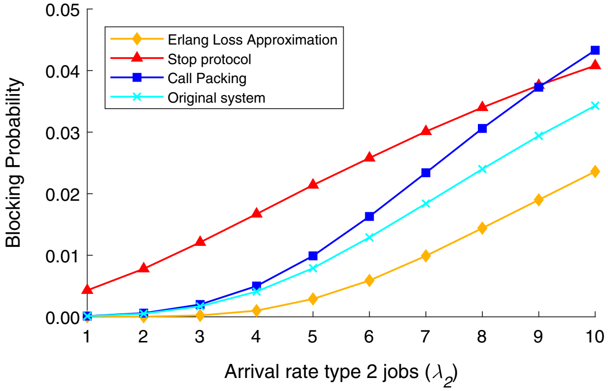

Table 4 Parameter values of numerical experiment 3

λ_1	λ_2	μ_1	μ_2	γ	N_1	N_2	M
15	–	2	1	3	10	10	10

Fig. 6 Results of numerical experiment 4

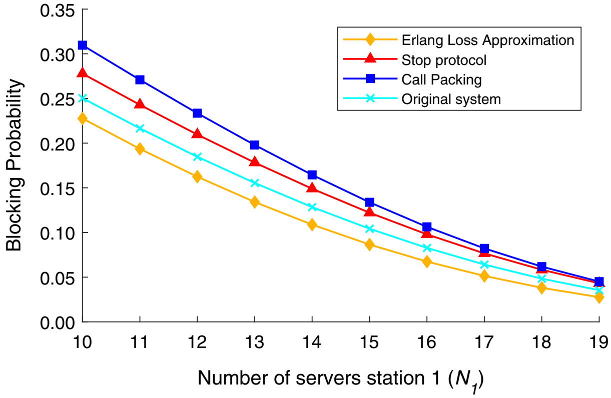

Table 5 Parameter values of numerical experiment 4

λ_1	λ_2	μ_1	μ_2	γ	N_1	N_2	M
15	20	1	3	2	–	5	5

imations, the arrival intensity of type 2 jobs (λ_2) is varied in numerical experiment 3 (see Table 4). Fig. 5 shows that call packing again provides a pessimistic approximation, which is quite accurate for small λ_2. However, the call packing approximation becomes less accurate when the arrival intensity of type 2 jobs gets larger. As a consequence, the stop protocol provides a better approximation of the blocking probability if λ_2 is equal to 10. Finally, it can be noted that the Erlang loss approximation again leads to an optimistic approximation of the blocking probability.

In numerical experiment 4, we consider a scenario with a high workload for both station 1 and 2 (utilization up to 150% for station 1 and utilization of 133% for station 2), as could occur for example during peak hours in call centers. Table 5 and Fig. 6 contain the parameter values and results for this experiment. It can be seen that the call packing approximation is

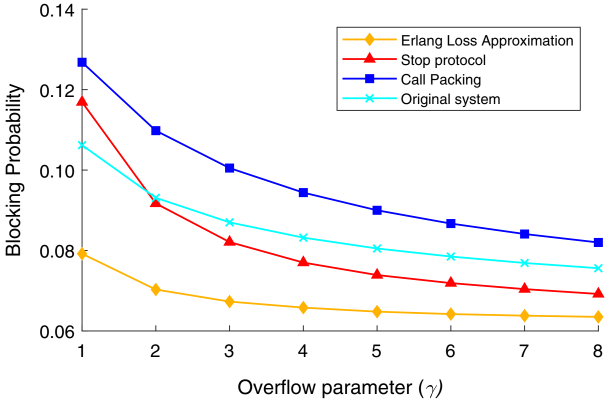

Fig. 7 Results of numerical experiment 5

Table 6 Parameter values of numerical experiment 5

λ_1	λ_2	μ_1	μ_2	γ	N_1	N_2	M
5	1	1	0.1	–	5	10	10

not as accurate as in numerical experiments 1 and 2, although it still performs reasonably well. Interestingly though, in this experiment the stop protocol performs superior over call packing. The reason for this is that stopping overflowed type 1 jobs occurs less frequently if the stations are highly congested.

In the previous experiments, the blocking probability for the system with stop protocol turned out to be larger than for the original system. However, this does not hold true in general since the stop protocol might lead to a smaller blocking probability for special cases. This is illustrated by numerical experiment 5. As can be seen in Table 6, this experiment considers a situation with a mean service time for type 2 jobs that is a relatively long compared to the mean service time for overflowed type 1 jobs (i.e. μ_2 is small compared to γ). Fig. 7 shows that the stop protocol leads to an underestimation of the blocking probability if $\gamma \geq 2$. On the other hand, the blocking probability for call packing again lies above the blocking probability for the original system. This is as expected, since $\gamma \geq \mu_1$ in this experiment.

6.3 Conclusions of numerical experiments

In conclusion, call packing appears to provide a rather accurate approximation of the blocking probability of type 1 jobs for natural situations with a low utilization for station 2. The blocking probability for the stop protocol numerically appears to be far more off than for call packing for these situations. However, this does not hold true for more extreme situations with highly congested stations. In such cases, the negative impact of the stop protocol on the blocking probability is smaller. Consequently, the stop protocol might provide a more accurate approximation than call packing for such situations.

Besides that, the numerical results support the intuitive arguments in Sect. 5. More specifically, in all numerical experiments call packing appears to provide a pessimistic approximation if $\gamma \geq \mu_1$. The stop protocol approximation, in turn, is always found to be pessimistic if $\gamma \leq \mu_2$. Moreover, it is shown that call packing does not provide a secure upper or lower bound if $\gamma < \mu_1$ and the stop protocol if $\gamma > \mu_2$ (see the counterexamples in numerical experiments 1 and 5, respectively). Hence, this supports the idea that call packing

Table 7 Insensitivity experiment: parameter values

	λ_1	λ_2	μ_1	μ_2	γ	N_1	N_2	M
Scenario 1	30	20	3	5	4	10	10	10
Scenario 2	15	5	2	1	3	10	10	10

provides an upper bound for the blocking probability of type 1 jobs if $\gamma \geq \mu_1$ and the stop protocol if $\gamma \leq \mu_2$. Formal proofs for such ordering results, even for limited cases, remain of interest.

Finally, in all numerical experiments the blocking probability of type 1 jobs according to the Erlang loss approximation is consistently found to be smaller than that for the original system. In general, however, an upper bound for a blocking probability seems of more practical value, such as for dimensioning capacities.

7 Insensitivity

In this section, the effect of non-exponentiality of service times is addressed. Until now, the service times have been assumed to be exponentially distributed. However, in reality service times might not be accurately described by the exponential distribution. It would thus be of interest if the results remain valid when the service times follow a different distribution. As discussed in Sect. 2.5, such a notion is well known under the term of insensitivity. The overflow system of interest is already known to be sensitive (e.g. Hordijk and Ridder 1987; Van Marion 1968), that is, the stationary distribution is affected if the exponentiality of service times is no longer valid. In this section, it is studied whether the product forms (2) and (5) are insensitive. In that case, the expressions for the blocking probabilities in (8)–(11) would also remain valid if the service times follow a different (non-exponential) distribution.

7.1 Simulation

In this section, it is investigated whether the product forms (2) and (5) can be expected to be insensitive. In that case, the blocking probability of type 1 jobs for the call packing system and the system with stop protocol should not be affected by the service time distribution. In order to investigate whether this holds true, two scenarios are considered (see Table 7 for the parameter values). For both scenarios, it is studied whether or not the same blocking probability of type 1 jobs results if the service times are assumed to be lognormally distributed instead of exponentially distributed. Therefore, for comparison, in Table 8 the blocking probabilities of type 1 jobs in case of exponentially distributed service times are given. These blocking probabilities are either determined numerically by using the GTH algorithm (original system) or analytically by using equation (8) or (10) (call packing and stop protocol, respectively).

Next, Discrete Event Simulation is used to determine the blocking probability of type 1 jobs in case of lognormally distributed service times. Moreover, for completeness, the blocking probability in case of exponentially distributed service times is also determined by simulation. The simulated blocking probabilities are shown in Table 9. Here, 95% confidence intervals as based on the t-distribution are given between brackets.

Original system: For both scenarios, the simulated blocking probability for the original system is not significantly different from the numerical blocking probability if the service times are exponentially distributed. However, apart from when the coefficient of variation

Table 8 Insensitivity experiment: blocking probability B_1 in case of exponential service times, determined numerically by using the GTH algorithm (original system) or analytically by using expression (8) or (10) (call packing and stop protocol, respectively)

	Original system	Call packing	Stop protocol
Scenario 1	0.0182 (1.82%)	0.0234 (2.34%)	0.0603 (6.03%)
Scenario 2	0.0079 (0.79%)	0.0099 (0.99%)	0.0214 (2.14%)

Table 9 Insensitivity experiment: blocking probability B_1 for different service time distributions, determined by simulation

	Service dist.	CV[a]	Scenario 1	Scenario 2
Original system	Exponential	1	0.0183 (0.0181–0.0185)[b]	0.0080 (0.0079–0.0080)
	Lognormal	0.1	0.0194 (0.0193–0.0194)	0.0089 (0.0088–0.0090)
	Lognormal	0.5	0.0184 (0.0183–0.0184)	0.0083 (0.0083–0.0084)
	Lognormal	1	0.0182 (0.0181–0.0183)	0.0079 (0.0079–0.0080)
	Lognormal	2	0.0180 (0.0180–0.0181)	0.0076 (0.0075–0.0077)
	Lognormal	10	0.0162 (0.0158–0.0165)	0.0070 (0.0067–0.0072)
Call packing (resume)	Exponential	1	0.0234 (0.0233–0.0234)	0.0100 (0.0099–0.0100)
	Lognormal	0.1	0.0264 (0.0263–0.0265)	0.0114 (0.0113–0.0114)
	Lognormal	0.5	0.0244 (0.0244–0.0245)	0.0105 (0.0104–0.0105)
	Lognormal	1	0.0234 (0.0234–0.0235)	0.0100 (0.0099–0.0101)
	Lognormal	2	0.0226 (0.0225–0.0228)	0.0095 (0.0095–0.0096)
	Lognormal	10	0.0220 (0.0217–0.0223)	0.0092 (0.0088–0.0097)
Call packing (resample)	Exponential	1	0.0235 (0.0231–0.0239)	0.0099 (0.0098–0.0099)
	Lognormal	0.1	0.0651 (0.0649–0.0652)	0.0148 (0.0147–0.0149)
	Lognormal	0.5	0.0467 (0.0466–0.0468)	0.0139 (0.0138–0.0140)
	Lognormal	1	0.0300 (0.0299–0.0302)	0.0114 (0.0114–0.0115)
	Lognormal	2	0.0176 (0.0175–0.0177)	0.0086 (0.0085–0.0087)
	Lognormal	10	0.0075 (0.0073–0.0076)	0.0052 (0.0049–0.0054)
Stop protocol	Exponential	1	0.0604 (0.0603–0.0606)	0.0214 (0.0213–0.0215)
	Lognormal	0.1	0.0603 (0.0602–0.0604)	0.0214 (0.0213–0.0215)
	Lognormal	0.5	0.0603 (0.0602–0.0604)	0.0214 (0.0213–0.0215)
	Lognormal	1	0.0604 (0.0602–0.0605)	0.0214 (0.0213–0.0215)
	Lognormal	2	0.0604 (0.0602–0.0605)	0.0213 (0.0212–0.0215)
	Lognormal	10	0.0601 (0.0594–0.0608)	0.0212 (0.0205–0.0220)

[a] Coefficient of variation

[b] 95% confidence interval between brackets

(CV) is 1, this is not the case if the service times are lognormally distributed. Hence, this implies the aforementioned sensitivity of the original system.

Call packing: For the call packing system, two aspects need to be taken into consideration when the service times are assumed to be non-exponential:

- When an overflowed type 1 jobs goes from station 2 to station 1 by call packing, the (residual) service time at station 1 can be determined in either of two ways. The service can be continued (resume) or completely started over (resample). In the resuming case, the service time at station 1 is computed as the residual service time multiplied by γ/μ_1 to account for the difference in service speed. In the resampling case, the service time at station 1 is determined by sampling from the service time distribution with mean equal to $1/\mu_1$. Because of the memoryless property of the exponential distribution, the results for resuming and resampling should be similar for exponential service times. On the other hand, this does not hold true for non-exponential service times. In Table 9, the results for resuming as well as resampling are given.

- When multiple overflowed type 1 jobs are present at station 2 and a type 1 job departs from station 1, it must be decided which overflowed type 1 job goes from station 2 to station 1. In this case, there are different methods to make this decision, such as selecting the job that is present for the longest time (first in, first out, FIFO), picking an arbitrary job (random) or choosing the job that arrived last (last in, first out, LIFO). Again, with exponential service times these methods should lead to similar results because of the memoryless property. In Table 9, the results for FIFO are given, but it is mentioned that for random and LIFO similar conclusions are drawn.

From Table 9, it can be seen that for both scenarios the simulated blocking probability is indeed similar for resuming and resampling if the service times are exponentially distributed. Moreover, they are not significantly different from the corresponding analytic blocking probabilities as given in Table 8. The numerical experiment also shows that the blocking probability differs across the coefficients of variation for both resuming and resampling if the service times follow a lognormal distribution. Here, almost all computed results are well outside each other's 95% confidence intervals. At the same time, in case of resuming, the amount of sensitivity with the CV range of variability appears to remain rather limited within these as well as more scenarios. Furthermore, the simulated blocking probabilities for the call packing (resume) system were found to be similar for the different coefficients of variation if the mean service time of overflowed type 1 jobs at station 2 is set equal to the mean service time at station 1 (i.e. $\gamma = \mu_1$).

Stop protocol: As opposed to the results for call packing, the scenarios show no statistically significant difference between the analytic and simulated blocking probabilities for the system with stop protocol. This holds for exponentially as well as lognormally distributed service times.

Conclusion: Simulation is used to investigate whether the product forms can be expected to remain valid for non-exponential service times. For call packing, in contrast to the special case for which $\gamma = \mu_1$, for the case dealt with in this paper, which allows $\gamma \neq \mu_1$, a strict insensitivity result appears not to be valid. This observation itself is of considerable theoretical interest since both stations could be regarded as standard Erlang loss systems. These, in turn, are well known to be strictly (i.e. 100%) insensitive.

In this respect, the stop protocol also becomes of interest. Admittedly, as illustrated in Sect. 6, the stop protocol numerically appears to provide a less accurate approximation than call packing if the stations are not highly congested. Nevertheless, the stop protocol might still be of interest since the product form can be concluded for arbitrary (i.e. non-exponential) service time distributions. This statement is formally shown for the general class of Erlang mixtures in Sect. 7.2.

7.2 Insensitivity of the product form for the system with stop protocol

As the stop protocol is of particular interest for the non-exponential case, in this section the insensitivity result is studied more detailed. More specifically, it is proven that the product form (5) is insensitive. As discussed in Remark 8, such a result could be concluded indirectly based on more abstract and combined literature. Below a proof is provided for a so-called class of mixtures of Erlang distributions. These are known to be dense and can be argued to represent all non-negative distributions (see Remark 6). In this section, we will mostly follow the notation of Van Dijk (2011).

As we will only focus on the overflow station 2, while overflowed type 1 jobs will remain there until completion of the service, without loss of generality only the service times of jobs at station 2 are assumed to be non-exponential. More specifically, it is assumed that type t jobs at station 2 require an amount of service with distribution function:

$$G_t = \sum_{k=1}^{\infty} q_t(k) E(k, \nu_t), \qquad t = 1, 2 \tag{12}$$

Here, $q_t(k)$ represents the probability that the distribution is an Erlang $E(k, \nu_t)$ distribution of k exponential phases with parameter ν_t.

Furthermore, let

$$\tau_t = \sum_{k=1}^{\infty} q_t(k)[k/\nu_t], \qquad t = 1, 2 \tag{13}$$

$$\gamma = [\tau_1]^{-1} \quad \text{and} \quad \mu_2 = [\tau_2]^{-1} \tag{14}$$

$$H_t(r) = [\tau_t \nu_t]^{-1} \sum_{k=r}^{\infty} q_t(k), \qquad t = 1, 2 \tag{15}$$

Here, τ_t is the mean service requirement and γ and μ_2 are similar to the parameters for the exponential case as introduced in Sect 3. Finally, $H_t(r)$ can be seen as steady-state probability that there are r residual exponential phases until a next renewal in a discrete renewal process with (inter) renewal distribution function G_t. By (15) the following discrete renewal relation is verified directly:

$$H_t(r) = H_t(r+1) + H_t(1) q_t(r), \qquad t = 1, 2 \tag{16}$$

In order to prove the insensitivity result, as rather standard (e.g. Barbour 1976; Schassberger 1977), we first aim to establish a detailed product form result that also includes residual service times. This, in turn, requires that individual jobs are kept track of. Therefore, we introduce the notion of positions, say $p = 1, \ldots, n$, when n jobs are present at station 2. Moreover, as discussed in Remark 7, a randomized allocation in combination with a simple shift protocol is used. More specifically, when $n-1$ jobs are present and a (type 1 or 2) job arrives at station 2, it will be assigned one of the positions $p = 1, ..., n$ each with probability $1/n$. The jobs previously at positions $p, ..., n-1$ then move to positions $p+1, ..., n$. When a job at position p completes its service, the jobs previously at positions $p+1, ..., n$ shift to positions $p, ..., n-1$.

The system can now be represented by a continuous-time Markov chain (CTMC) with state description:

$$(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \quad \text{where} \quad [\boldsymbol{T}, \boldsymbol{R}] = [(t_1, r_1), (t_2, r_2), ..., (t_n, r_n)] \quad (n = n_2 + m) \tag{17}$$

Here, the pth element of $[\boldsymbol{T}, \boldsymbol{R}]$ denotes that the job at position p at station 2 is of type t_p and has r_p residual exponential phases for servicing each with parameter ν_t, where $t = t_p$.

Furthermore, the following shorthand notation will be useful when the job at position p for a fixed p is considered.

$$[\boldsymbol{T}, \boldsymbol{R}] - (t_p, r_p)_p = ((t_1, r_1), ..., (t_{p-1}, r_{p-1}), (t_{p+1}, r_{p+1}), ..., (t_n, r_n)) \tag{18}$$

$$\begin{aligned}&[\boldsymbol{T}, \boldsymbol{R}] - (t_p, r_p)_p + (t_p, r_p + 1)_p\\ &\quad = ((t_1, r_1), ..., (t_{p-1}, r_{p-1}), (t_p, r_p + 1), (t_{p+1}, r_{p+1}), ..., (t_n, r_n))\end{aligned} \tag{19}$$

$$\begin{aligned}&[\boldsymbol{T}, \boldsymbol{R}] + (t, r)_p\\ &\quad = ((t_1, r_1), ..., (t_{p-1}, r_{p-1}), (t, r), (t_p, r_p), ..., (t_n, r_n)) \quad t = 1, 2 \quad r = 1, 2, ...\end{aligned} \tag{20}$$

Here, in (18) the job at position p is left out, and the jobs that were previously at positions $p + 1, ..., n$ are moved to positions $p, ..., n - 1$. Besides that, in (19) the job at position p has its number of residual phases changed from r_p to $r_p + 1$ phases. Finally, in (20) a type t job with r exponential phases is added at position p, and the jobs that were previously at positions $p, ..., n$ are moved to positions $p + 1, ..., n + 1$.

The following detailed product form result can now be obtained for the system with stop protocol. This detailed product form, in turn, will lead to the insensitivity result as aimed for (see Corollary 1).

Theorem 3 *Let $n = n_2 + m$ be the number of jobs at station 2 and c a normalizing constant. Under the stop protocol, the following detailed product form then applies for all states $(n_1, [\boldsymbol{T}, \boldsymbol{R}])$ with $n_1 \in \{0, ..., N_1\}$ and $(n_2, m) \in \boldsymbol{C}$:*

$$\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) = c \frac{1}{n_1!} \left(\frac{\lambda_1}{\mu_1}\right)^{n_1} \frac{1}{n!} \prod_{p:t_p=1} \left\{\frac{\lambda_1}{\gamma} H_1(r_p)\right\} \prod_{p:t_p=2} \left\{\frac{\lambda_2}{\mu_2} H_2(r_p)\right\} \tag{21}$$

Proof As before, the proof will be based on the global balance equations. Herein, it will be shown that a notion of balance is satisfied for each position p (i.e. each individual job), separately. This principle could also be referred to as job-local-balance (e.g. Hordijk and Van Dijk 1983a).

Before stating the balance equations, it is first recalled that $n = n_2 + m$ represents the positions at station 2 and noted that the set of admissible states $\boldsymbol{S}$ is restricted to:

$$\begin{aligned}\boldsymbol{S} =&\{(n_1, [\boldsymbol{T}, \boldsymbol{R}])|\ 0 \le n_1 \le N_1,\ (n_2, m) \in \boldsymbol{C},\\ &t_p = 1, 2 \quad (p = 1, ..., n),\ r_p = 1, 2, ... \quad (p = 1, ..., n)\}\end{aligned} \tag{22}$$

Now, the rate out of state $(n_1, [\boldsymbol{T}, \boldsymbol{R}])$ is given by (23.1) + ... + (23.6).

$$\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) n_1 \mu_1 1_{\{n_1>0\}} \tag{23.1}$$

$$\sum_{p=1}^{n} \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \nu_1 1_{\{n_1=N_1\}} 1_{\{t_p=1\}} \tag{23.2}$$

$$\sum_{p=1}^{n} \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \nu_2 1_{\{t_p=2\}} \tag{23.3}$$

$$\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \lambda_1 1_{\{n_1<N_1\}} \tag{23.4}$$

$$\sum_{p=1}^{n+1} \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \frac{1}{n+1} \lambda_1 1_{\{n_1=N_1\}} 1_{\{(n_2,m+1)\in \boldsymbol{C}\}} \tag{23.5}$$

$$\sum_{p=1}^{n+1} \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \frac{1}{n+1} \lambda_2 1_{\{(n_2+1,m)\in \boldsymbol{C}\}} \tag{23.6}$$

The rate into state $(n_1, [\boldsymbol{T}, \boldsymbol{R}])$, in contrast, is given by (23.1)$' + ... +$(23.6)$'$.

$$\pi(n_1 - 1, [\boldsymbol{T}, \boldsymbol{R}])\lambda_1 1_{\{n_1>0\}} \tag*{(23.1)$'$}$$

$$\sum_{p=1}^{n} 1_{\{t_p=1\}} \Big\{ \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] - (1, r_p)_p) \frac{1}{n} q_1(r_p) \lambda_1 1_{\{n_1=N_1\}} + \\ \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] - (1, r_p)_p + (1, r_p + 1)_p) \nu_1 1_{\{n_1=N_1\}} \Big\} \tag*{(23.2)$'$}$$

$$\sum_{p=1}^{n} 1_{\{t_p=2\}} \Big\{ \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] - (2, r_p)_p) \frac{1}{n} q_2(r_p) \lambda_2 + \\ \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] - (2, r_p)_p + (2, r_p + 1)_p) \nu_2 \Big\} \tag*{(23.3)$'$}$$

$$\pi(n_1 + 1, [\boldsymbol{T}, \boldsymbol{R}])(n_1 + 1)\mu_1 1_{\{n_1<N_1\}} \tag*{(23.4)$'$}$$

$$\sum_{p=1}^{n+1} \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] + (1, 1)_p) \nu_1 1_{\{n_1=N_1\}} 1_{\{(n_2,m+1)\in \boldsymbol{C}\}} \tag*{(23.5)$'$}$$

$$\sum_{p=1}^{n+1} \pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] + (2, 1)_p) \nu_2 1_{\{(n_2+1,m)\in \boldsymbol{C}\}} \tag*{(23.6)$'$}$$

Here, it is noted that the indicator $1_{\{n_1=N_1\}}$ in (23.2), in the second line of (23.2)$'$ and in (23.5) is to be included because of the stop protocol. The proof can now be completed by showing that (23.i)= (23.i)$'$, $i = 1, ..., 6$, for each $(n_1, [\boldsymbol{T}, \boldsymbol{R}]) \in \boldsymbol{S}$. Here, beforehand, it is mentioned that we will write (23.i.p) and (23.i.p)$'$ for $i = 2, 3, 5, 6$ and all positions p when referring to the pth element of the sum in these equations.

First of all, it is noted that the indicators in (23.i) are equal to those in (23.i)$'$ for $i = 1, ..., 6$. Therefore, it suffices to only consider the non-zero cases. These, in turn, are verified as follows.

(23.1)=(23.1)$'$: This can be verified directly for $n_1 > 0$ by substituting the detailed product form (21).

(23.2)=(23.2)$'$: First, it can be noted that by assuming (21) we have:

$$\frac{\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] - (1, r_p)_p)}{\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}])} = n \frac{\gamma}{\lambda_1} \frac{1}{H_1(r_p)} \qquad p = 1, ..., n \tag{24}$$

$$\frac{\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] - (1, r_p)_p + (1, r_p + 1)_p)}{\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}])} = \frac{H_1(r_p + 1)}{H_1(r_p)} \qquad p = 1, ..., n \tag{25}$$

By using these ratios and the discrete renewal relation (16) (and noting that $H_1(1) = \gamma/\nu_1$), it can be verified that (23.2.p) = (23.2.p)$'$ for all $p = 1, ..., n$ with $t_p = 1$ if $n_1 = N_1$. Hence, by summing over p it follows that (23.2) = (23.2)$'$.

(23.3)=(23.3)$'$: In a similar way as for the previous case, it is possible to show that (23.3.p) = (23.3.p)$'$ for all $p = 1, ..., n$ with $t_p = 2$, after which summing over p leads to the desired result.

(23.4)=(23.4)′: This can be verified directly for $n_1 < N_1$ by substituting the detailed product form (21).

(23.5)=(23.5)′: It can be noted that by assuming (21) we have:

$$\frac{\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}] + (1,1)_p)}{\pi(n_1, [\boldsymbol{T}, \boldsymbol{R}])} = \frac{1}{n+1}\frac{\lambda_1}{\gamma}H_1(1) = \frac{1}{n+1}\frac{\lambda_1}{\nu_1} \qquad p = 1, ..., n+1 \tag{26}$$

Using this ratio, it is easily verified that (23.5.p)=(23.5.p)′ for $p = 1, ..., n+1$ with $t_p = 1$ if $n_1 = N_1$ and $(n_2, m+1) \in \boldsymbol{C}$. Summing over p then yields (23.5) = (23.5)′.

(23.6)=(23.6)′ : In a similar way as for the previous case, it is possible to show that (23.6.p) = (23.6.p)′ for all $p = 1, ..., n+1$ with $t_p = 2$ if $(n_2 + 1, m) \in \boldsymbol{C}$, after which summing over p again results in the desired result.

Hence, it is shown that (23.1) +...+ (23.6) = (23.1)′ +...+ (23.6)′ by substituting the detailed product form (21). This completes the proof. □

From the result in Theorem 3, it can now be shown that the product form (5) also remains valid for mixtures of Erlang distributions as service time distribution.

Corollary 1 *Under the stop protocol and with c a normalizing constant, the following product form applies for all states* (n_1, n_2, m) *with* $n_1 \in \{0, ..., N_1\}$ *and* $(n_2, m) \in \boldsymbol{C}$*:*

$$\pi(n_1, n_2, m) = c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1}\frac{1}{m!}\left(\frac{\lambda_1}{\gamma}\right)^{m}\frac{1}{n_2!}\left(\frac{\lambda_2}{\mu_2}\right)^{n_2} \tag{27}$$

Proof The result follows by summing the detailed product form (21) over all possible configurations with m type 1 and n_2 type 2 jobs at station 2, and, for each configuration, over all possible phases r_p for each position $p = 1, ..., n$. See Appendix A for the technical details. □

Remark 6 *(General service time distributions)* General non-negative distributions can be approximated arbitrarily closely in the sense of weak convergence by mixtures of Erlang distributions, that is, distributions as in (12). By using general weak convergence limit theorems on D-sample path spaces (e.g. Barbour 1976; Hordijk and Schassberger 1982; Schassberger 1977, 1978), the insensitivity result as expressed by Corollary 1 can therefore also be concluded for general non-negative service time distributions.

Remark 7 *(Service discipline of station 2)* As mentioned before, the positions at station 2 are introduced in order to keep track of the individual jobs and their number of residual phases. However, the positioning itself is not essential, as each job at station 2 gets fully served by one of the N_2 servers. Therefore, it is justified to use a randomized allocation in combination with a shift protocol, so that the positions remain successive. In fact, the described service discipline coincides with a processor sharing protocol (see e.g. Baskett et al. 1975; Kelly 1979), which is a special case of a symmetric service discipline. With additional notation but by the same proof steps a similar product form expression can also be concluded for other service disciplines, such as any "symmetric" discipline (see Kelly 1979). This contains, for example, preemptive-resume last come, first served (LCFS) servicing.

Remark 8 *(Literature)* For the stop protocol, the insensitivity could indirectly be concluded by combining a specific feature or concept of detailed balance, such as job-local-balance (e.g. Hordijk and Van Dijk 1983a), for the exponential case with more abstract insensitivity results as given by Barbour (1976), Hordijk and Van Dijk (1983b) or Schassberger (1977, 1978). The present proof along with its extension to the general restriction by $\boldsymbol{C}$ is kept self-contained.

8 Evaluation

Overflow within service systems is a most common feature in practice, such as in telecommunications, health care and daily life logistics. It is thus of interest to obtain insights in its possible solvability, either exact or approximate, and effects of underlying service distributional assumptions, even for simplified situations. In this paper, we study a most simplistic two-station overflow system, which essentially allows overflowed jobs to have a different mean service rate. From a product form point of view, this system already appears to be hard to analyze, if not unsolvable.

A threefold direction is taken from a product form perspective:

- It studied two product form modifications, which are both already known from available literature. These are the call packing principle and stop protocol.
- It numerically explored to which extent these lead to a useful approximation of the blocking probability and it pays attention to possible ordering.
- It studied the possibility of insensitivity as to non-exponentiality assumptions.

Here, the call packing mechanism might also be regarded as natural. As such, it is also studied in literature and already shown to exhibit a product form in slightly different setting. The primary purpose in this paper is to regard it as modification for computation. The stop protocol, which is purely artificial, is meant for that purpose as well.

First, for the exponential case and both protocols, straightforward and self-contained proofs of the product forms are given as by forms of partial balance. Despite the vast literature on product forms and practical interest for the system that is studied, the product form expressions do not seem to be proven directly or reported explicitly. For the overflow system with call packing, one closely related product form reference and one which has an entirely different viewpoint are also studied for possible application and further insight. It shows that even such a ‘simple’ system can still be intriguing from different perspectives and be of considerable interest in their own right.

As next step, an extensive numerical evaluation of the simple analytic (product form) expressions is performed. For both protocols, the quality of the approximations of the blocking probability of type 1 jobs is studied. It appears that call packing leads to a far more accurate approximation than the stop protocol for natural situations with a low utilization at station 2. For more extreme situations with heavily loaded stations, in contrast, the stop protocol might outperform call packing. Besides that, call packing is consistently found to provide a pessimistic approximation when the overflowed type 1 jobs are served faster (on average) than the non-overflowed type 1 jobs. The stop protocol, in turn, appears to lead to a pessimistic approximation for all experiments except those in which it is assumed that the service of overflowed type 1 jobs is extremely fast relative to the service of type 2 jobs.

Finally, as third aspect, for both protocols the (in)sensitivity is also studied. As could be expected based on literature, the stop protocol is shown to lead to insensitivity as based on a sufficiently detailed notion of partial balance. From a practical point of view, this property could make the stop protocol appealing since it allows not to bother about exponentiality assumptions. For call packing, in contrast, it appears that strict insensitivity cannot be concluded, as supported by simulation. From a theoretical perspective, this might be seen as noteworthy given a product form result reflecting Erlang loss stations. For practical purposes, the sensitivity seems light.

Both the ordering and (in)sensitivity results do not seem to conflict with existing literature, yet they illustrate intriguing aspects for comparisons and extensions. Different challenging points remain, such as:

- Formal proofs for bounds and its specific conditions, such as for station and/or job type depending characteristics.
- Numerical support, if not even formal error bounds, for the effect of service distributional forms (sensitivity).
- More complex overflow structures, such as with multiple phases, parallel as well as serial overflow and hierarchical structures, as arising in different practices (e.g. flexible manufacturing, skill based routing or flexible specialized intensive care units).

The results illustrate that even simple network structures and corresponding product form findings might still be of considerable interest for future research from both a theoretical and practical point of view.

Acknowledgements The authors are grateful to the referees for the comments that helped to improve the presentation.

Appendix A Proof of Corollary 1

In this appendix, a proof of Corollary 1 is given.

Proof In order to prove the result, consider an arbitrary state (n_1, n_2, m) with $n_1 \in \{0, ..., N_1\}$ and $(n_2, m) \in \boldsymbol{C}$. Moreover, let $n = n_2 + m$ again denote the total number of jobs at station 2. Then, it is first noted that there are several possible configurations with m type 1 and n_2 type 2 jobs at station 2, positioned at positions 1, ..., n. In total, the number of such configurations is equal to:

$$\binom{n_2+m}{m} = \frac{(n_2+m)!}{m!(n_2+m-m)!} = \frac{(n_2+m)!}{m!n_2!} \tag{28}$$

Next, arbitrarily choose one of these configurations, specified by $t_1, ..., t_n$ (remember that $t_p \in \{1, 2\}$ denotes that the job at position p is of type t_p, $p = 1, ..., n$). Then, the probability to observe this configuration (and n_1 type 1 jobs at station 1), denoted by P, needs to be determined. This can be done by summing the detailed product form (21) over all possible phases r_p for each position $p = 1, ..., n$. Hence, using the factorizing form of the detailed product form, it follows that:

$$\begin{aligned} P &= \sum_{r_1=1}^{\infty} \cdots \sum_{r_n=1}^{\infty} c \frac{1}{n_1!} \left(\frac{\lambda_1}{\mu_1}\right)^{n_1} \frac{1}{n!} \prod_{p:t_p=1} \left\{\frac{\lambda_1}{\gamma} H_1(r_p)\right\} \prod_{p:t_p=2} \left\{\frac{\lambda_2}{\mu_2} H_2(r_p)\right\} \\ &= c \frac{1}{n_1!} \left(\frac{\lambda_1}{\mu_1}\right)^{n_1} \frac{1}{n!} \left(\frac{\lambda_1}{\gamma}\right)^{m} \left(\frac{\lambda_2}{\mu_2}\right)^{n_2} \sum_{r_1=1}^{\infty} \cdots \sum_{r_n=1}^{\infty} \prod_{\substack{p: \\ t_p=1}} \{H_1(r_p)\} \prod_{\substack{p: \\ t_p=2}} \{H_2(r_p)\} \end{aligned}$$

$$
\begin{aligned}
&= c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1}\frac{1}{n!}\left(\frac{\lambda_1}{\gamma}\right)^{m}\left(\frac{\lambda_2}{\mu_2}\right)^{n_2}\prod_{\substack{p:\\ t_p=1}}\left\{\sum_{r_p=1}^{\infty}H_1(r_p)\right\}\prod_{\substack{p:\\ t_p=2}}\left\{\sum_{r_p=1}^{\infty}H_2(r_p)\right\}\\
&= c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1}\frac{1}{n!}\left(\frac{\lambda_1}{\gamma}\right)^{m}\left(\frac{\lambda_2}{\mu_2}\right)^{n_2} \qquad (29)
\end{aligned}
$$

Here, the last step follows by recalling the probability interpretation of $H_t(\cdot)$ as by (15), so that $\sum_{r=1}^{\infty} H_t(r) = 1, t = 1, 2$.

Hence, the expression for P as specified by (29) is independent of t_1, ..., t_n and thus equal for each configuration with m type 1 and n_2 type 2 jobs at station 2. Therefore, using (28) and (29), it follows that:

$$
\begin{aligned}
\pi(n_1, n_2, m) &= \left(\frac{(n_2+m)!}{m!n_2!}\right)\cdot P\\
&= \left(\frac{n!}{m!n_2!}\right)\cdot\left(c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1}\frac{1}{n!}\left(\frac{\lambda_1}{\gamma}\right)^{m}\left(\frac{\lambda_2}{\mu_2}\right)^{n_2}\right)\\
&= c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1}\frac{1}{m!}\left(\frac{\lambda_1}{\gamma}\right)^{m}\frac{1}{n_2!}\left(\frac{\lambda_2}{\mu_2}\right)^{n_2} \qquad (30)
\end{aligned}
$$

This completes the proof of Corollary 1. □

Appendix B Alternative proofs of Theorem 1

In Sect. 4.1, a proof of Theorem 1 based on the global balance equations is given. As discussed in Remark 1, the product form (2) could also be concluded from product form results in literature. For illustrative purposes, two alternative proofs of Theorem 1 are therefore given in this appendix.

B.1 Alternative routing network satisfying certain conditions

The product form (2) can also be proven by showing that the overflow system with call packing satisfies conditions (a) to (g) stated by Henderson and Taylor (1988). Then, the product form follows directly from the result in this reference. This is made explicit below. Here, it is assumed that overflowed type 1 jobs preemptively resume service if they go to station 1 by call packing.

Proof In order to prove the product form (2), it needs to be shown that conditions (a) to (g) of Henderson and Taylor (1988) are satisfied. For this purpose, the overflow system under the assumption of call packing needs to be described using the notation in the paper by Henderson and Taylor (1988). To this end, the following notation is introduced.

First of all, let $T = \{1, 2\}$ and $\mathbf{n} = \{n(t), t \in T\} = \{n(1), n(2)\}$, where

$n(1)$ is the number of type 1 jobs present in the system ($n(1) = n_1 + m$),

$n(2)$ is the number of type 2 jobs present in the system ($n(2) = n_2$).

Moreover, let $\lambda(t) = \lambda_t$, $t = 1, 2$, and $\mathcal{F} = \boldsymbol{S}$, where the state space $\boldsymbol{S}$ is as specified by (3). Besides that, the following non-negative function is defined:

$$\phi(\mathbf{n}) = \begin{cases} \frac{1}{n(1)!}\left(\frac{1}{\mu_1}\right)^{n(1)} \frac{1}{n(2)!}\left(\frac{1}{\mu_2}\right)^{n(2)} & \text{for } n(1) \leq N_1 \\ \frac{1}{N_1!}\left(\frac{1}{\mu_1}\right)^{N_1} \frac{1}{\prod_{k=1}^{n(1)-N_1}(N_1\mu_1 + k\gamma)} \frac{1}{n(2)!}\left(\frac{1}{\mu_2}\right)^{n(2)} & \text{for } n(1) > N_1 \end{cases} \tag{31}$$

Finally, in order to describe the overflow system under the assumption of call packing, $r(t)$, $p(t, s)$, $\delta_t(l, \mathbf{n})$ and $\gamma_t(l, \mathbf{n})$ can be chosen as follows (see Henderson and Taylor (1988) for the interpretation of these functions):

$$r(t) = 1 \qquad t = 1, 2 \tag{32}$$

$$p(t, s) = 0 \qquad t, s = 1, 2 \tag{33}$$

$$\delta_1(l, \mathbf{n}) = 1/n(1) \qquad \text{for } l = 1, ..., n(1) \qquad (\text{if } n(1) \leq N_1) \tag{34}$$

$$\delta_1(l, \mathbf{n}) = \begin{cases} 0 & \text{for } l = 1, ..., N_1 \\ \frac{1}{n(1)-N_1} & \text{for } l = N_1 + 1, ..., n(1) \end{cases} \qquad (\text{if } n(1) > N_1) \tag{35}$$

$$\delta_2(l, \mathbf{n}) = 1/n(2) \qquad \text{for } l = 1, ..., n(2) \tag{36}$$

$$\gamma_1(l, \mathbf{n}) = 1/n(1) \qquad \text{for } l = 1, ..., n(1) \qquad (\text{if } n(1) \leq N_1) \tag{37}$$

$$\gamma_1(l, \mathbf{n}) = \begin{cases} \frac{\mu_1}{N_1\mu_1 + (n(1)-N_1)\gamma} & \text{for } l = 1, ..., N_1 \\ \frac{\gamma}{N_1\mu_1 + (n(1)-N_1)\gamma} & \text{for } l = N_1 + 1, ..., n(1) \end{cases} \qquad (\text{if } n(1) > N_1) \tag{38}$$

$$\gamma_2(l, \mathbf{n}) = 1/n(2) \qquad \text{for } l = 1, ..., n(2) \tag{39}$$

Then, it can be verified that conditions (a) to (g) are satisfied. As a consequence, it follows from Theorem 2 of Henderson and Taylor (1988) that the steady-state distribution $\pi(\mathbf{n})$ is given by:

$$\pi(\mathbf{n}) = c\phi(\mathbf{n}) \prod_{t \in T} [y(t)]^{n(t)} \tag{40}$$

where c is the normalizing constant and the $y(t)$ satisfy:

$$y(t) = \lambda(t) + \sum_{s \in T} y(s)p(s, t) \qquad \text{for } t \in T \tag{41}$$

This expression is equivalent to the expression for π_{cp} that is given in (2). Hence, this completes the proof of Theorem 1. □

Remark 9 ($\gamma = \mu_1$) It is noted that it does not matter whether a type 1 job receives service at station 1 or 2 if $\gamma = \mu_1$. Therefore, $\delta_1(l, \mathbf{n})$ (i.e. the probability that an arriving type 1 job is allocated label l amongst the type 1 jobs when $n(1) - 1$ type 1 and $n(2)$ type 2 jobs are present) can then also be given by $1/n(1)$ for $l = 1, ..., n(1)$ (even if $n(1) > N_1$). Moreover, the expression for $\gamma_1(l, \mathbf{n})$ (i.e. the proportion of service effort that is given to the lth type 1 job when the state is $\mathbf{n}$) then simplifies to $1/n(1)$ for $l = 1, ..., n(1)$. This means that a symmetric service discipline results (i.e. $\delta_t(l, \mathbf{n}) = \gamma_t(l, \mathbf{n})$). As a consequence, condition (f) ("If type t calls have a non-exponential holding time distribution $\delta_t(l, \mathbf{n}) = \gamma_t(l, \mathbf{n})$") is also satisfied for non-exponential service times if $\gamma = \mu_1$. On the other hand, this is not the case if $\gamma \neq \mu_1$. This means that insensitivity can be concluded if $\gamma = \mu_1$, while the product form may be sensitive if $\gamma \neq \mu_1$. Hence, this is in line with the simulation results in Sect. 7.1.

B.2 Competing Markov chains

Another way to prove the product form (2) is to model the overflow system with call packing as competing Markov chains. Then, the product form can be concluded from the result of Boucherie (1994). This is illustrated below for the case without type 2 jobs and coordinate convex structure at station 2. As discussed at the end of this appendix, a proof along similar lines can be expected when type 2 jobs and a coordinate convex structure at station 2 are also included.

Proof First of all, the following two continuous-time Markov chains are defined.

Markov chain 1: First of all, Markov chain 1 describes the transitions of type 1 jobs at station 1. Here, it is noted that station 1 can be considered as a standard Erlang loss system ($M|M|N_1|N_1$ queue) with arrival rate λ_1 and mean service rate μ_1.

Therefore, it follows that the steady-state distribution π_1 is as follows:

$$\pi_1(n_1) = c_1 \frac{1}{n_1!} \left(\frac{\lambda_1}{\mu_1}\right)^{n_1} \qquad n_1 \in S_1 \tag{42}$$

Here, c_1 is a normalizing constant and S_1 the state space, which is given by:
$S_1 = \{n_1 | 0 \leq n_1 \leq N_1\}$.

Markov chain 2: Secondly, Markov chain 2 describes the transitions of overflowed type 1 jobs at station 2 when station 1 is congested (i.e. $n_1 = N_1$). This means that arrivals occur with rate λ_1. Moreover, when m jobs are present, the total service rate is equal to $N_1\mu_1 + m\gamma$ if $m > 0$ and 0 if $m = 0$. It is noted that the term $N_1\mu_1$ is included since there is also a departure from station 2 if a job at station 1 completes its service (because of call packing).

Then, it can be verified that the steady-state distribution π_2 is given by:

$$\pi_2(m) = \begin{cases} c_2 & \text{if } m = 0 \\ c_2 \frac{\lambda_1^m}{\prod_{k=1}^{m}(N_1\mu_1 + k\gamma)} & \text{if } m > 0 \end{cases} \qquad m \in S_2 \tag{43}$$

Here, c_2 is a normalizing constant and S_2 the state space, which is given by:
$S_2 = \{m | 0 \leq m \leq N_2\}$.

It can be noted that these Markov chains do not accurately describe the overflow system yet. More specifically, if $n_1 < N_1$ (and hence $m = 0$), no arrivals at station 2 should occur because arriving type 1 jobs would go to station 1. Moreover, if $m > 0$ (and hence $n_1 = N_1$), no departures from station 1 should occur since the place of a departing job at station 1 would immediately be taken by an overflowed type 1 job from station 2 (because of call packing). Hence, this is where the competition mechanism comes in.

To this end, define the index set $I = \{1, 2\}$ and let A_{ki} and C_{ki}, $k = 1, 2, i \in I$, be defined as follows (see Boucherie 1994, for the precise interpretations):

$$A_{11} = \{n_1 | n_1 = N_1\} \qquad C_{11} = \emptyset \tag{44}$$

$$A_{12} = \{n_1 | 0 \leq n_1 < N_1\} \qquad C_{12} = \{2\} \tag{45}$$

$$A_{21} = \{m | m = 0\} \qquad C_{21} = \emptyset \tag{46}$$

$$A_{22} = \{m | 0 < m \leq N_2\} \qquad C_{22} = \{1\} \tag{47}$$

This means that Markov chains 1 and 2 compete over resource 2, while they do not compete over resource 1. More specifically, both Markov chains are allowed to make a transition if both $n_1 \in A_{11}$ and $m \in A_{21}$ hold. On the other hand, only Markov chain 1 can make a transition if $n_1 \in A_{12}$ and $m \in A_{21}$ (in this case, Markov chain 2 is frozen). Similarly, only

Markov chain 2 can make a transition if $m \in A_{22}$ and $n_1 \in A_{11}$ (then, Markov chain 1 is frozen). Finally, a state (n_1, m) with both $n_1 \in A_{12}$ and $m \in A_{22}$ cannot occur.

As a consequence, the state space S is as follows:

$$\begin{aligned} S &= S_1 \times S_2 \backslash A_{12} \times A_{22} \\ &= \{(n_1, m) | 0 \leq n_1 < N_1, m = 0, \text{ or } n_1 = N_1, 0 \leq m \leq N_2\} \end{aligned} \tag{48}$$

Subsequently, the coefficients $c_1(n_1)$ and $c_2(m)$ can be chosen equal to 1 for all $n_1 \in S_1$ and $m \in S_2$, respectively. The transition rates $(q(\bar{n}, \bar{n}'), \bar{n}, \bar{n}' \in S)$, where $\bar{n} = (n_1, m)$ and $\bar{n}' = (n_1', m')$, are then given by:

$$q(\bar{n}, \bar{n}') = q_1(n_1, n_1')1_{\{m=m' \in A_{21}\}} + q_2(m, m')1_{\{n_1 = n_1' \in A_{11}\}} \tag{49}$$

Here, $(q_1(n_1, n_1'), n_1, n_1' \in S_1)$ and $(q_2(m, m'), m, m' \in S_2)$ are the transition rates of Markov chain 1 and 2, respectively.

The Markov chain at state space S as in (48) and with transition rates q as in (49) is then called the product process of the collection of Markov chains 1,2 competing over resources I. Moreover, the described product process provides an accurate description of the behaviour of the overflow system with call packing and without the presence of type 2 jobs.

As a consequence, it can be concluded from Theorem 1 of Boucherie (1994) that the steady-state distribution π is equal to:

$$\pi(n_1, m) = B\pi_1(n_1)\pi_2(n_2, m) \qquad (n_1, m) \in S \tag{50}$$

Here, B is the normalizing constant, determined by the form of S. After substitution of (42) and (43), the following product form results:

$$\pi(n_1, m) = \begin{cases} c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1} & \text{if } m = 0 \\ c\frac{1}{n_1!}\left(\frac{\lambda_1}{\mu_1}\right)^{n_1} \frac{\lambda_1^m}{\prod_{k=1}^{m}(N_1\mu_1 + k\gamma)} & \text{if } m > 0 \end{cases} \qquad (n_1, m) \in S \tag{51}$$

Here, c is the normalizing constant. □

Remark 10 *(Inclusion of type 2 jobs)* If type 2 jobs at station 2 are also included, Markov chain 2 would have to describe the transitions of both overflowed type 1 jobs and type 2 jobs at station 2. Since the arrivals and departures of type 2 jobs should not be stopped if $n_1 < N_1$, Theorem 1 of Boucherie (1994) is therefore no longer applicable. In this case, however, Theorem 2 of Boucherie (1994) could be applied. To this end, it is noted that the transition rates for Markov chain 2 can be separated into a part that describes the behaviour of overflowed type 1 jobs and a part that describes the behaviour of type 2 jobs (note that Markov chain 2 is locally balanced with respect to this separation). The processes with these transition rates are then Markov chains on their own, which are referred to as Markov chain (2,1) for overflowed type 1 jobs and Markov chain (2,2) for type 2 jobs. Hence, we obtain two Markov chains which operate on the same state space. Moreover, it is possible that Markov chain (2,1) is frozen if $n_1 < N_1$, while Markov chain (2,2) can still undergo transitions. In this way, a proof of the product form can also be expected when type 2 jobs and a coordinate convex structure at station 2 are included. As the technical description is more detailed, it is not included here (but merely conjectured).

References

Akimaru, H., & Takahashi, H. (1983). An approximate formula for individual call losses in overflow systems. *IEEE Transactions on communications*, *31*(6), 808–811.

Asaduzzaman, M., & Chaussalet, T. J. (2014). Capacity planning of a perinatal network with generalised loss network model with overflow. *European Journal of Operational Research*, *232*(1), 178–185.

Balsamo, S., Harrison, P. G., & Marin, A. (2010). A unifying approach to product-forms in networks with finite capacity constraints. *ACM SIGMETRICS Performance Evaluation Review*, *38*(1), 25–36.

Barbour, A. (1976). Networks of queues and the method of stages. *Advances in Applied Probability*, *8*(3), 584–591.

Baskett, F., Chandy, K. M., Muntz, R. R., & Palacios, F. G. (1975). Open, closed, and mixed networks of queues with different classes of customers. *Journal of the ACM*, *22*(2), 248–260.

Berry, L. T. M., & Henderson, W. (1989). Some exact results in performance analysis of alternative routing communications networks. *Australian Telecommunication Research*, *23*(1), 35–42.

Borst, S., Boucherie, R. J., & Boxma, O. J. (1999). ERMR: A generalised equivalent random method for overflow systems with repacking. In P. Key & D. Smith (Eds.), *ITC-16* (pp. 313–323). Amsterdam: Elsevier.

Boucherie, R. J., & Van Dijk, N. M. (1993). A generalization of Norton's theorem for queueing networks. *Queueing Systems*, *13*(1–3), 251–289.

Boucherie, R. J. (1994). A characterization of independence for competing Markov chains with applications to stochastic Petri nets. *IEEE Transactions on Software Engineering*, *20*(7), 536–544.

Brandt, A., & Brandt, M. (2001). Approximation for overflow moments of a multiservice link with trunk reservation. *Performance Evaluation*, *43*(4), 259–268.

Burman, D. Y., Lehoczky, J. P., & Lim, Y. (1984). Insensitivity of blocking probabilities in a circuit-switching network. *Journal of Applied Probability*, *21*(4), 850–859.

Chandy, K. M., & Martin, A. J. (1983). A characterization of product-form queuing networks. *Journal of the Association for Computing Machinery*, *30*(2), 286–299.

Conway, A. E. (1989). Product-form and insensitivity in circuit-switched networks with failing links. *Performance Evaluation*, *9*(3), 209–215.

El-Taha, M., & Heath, J. R. (2000). Traffic overflow in loss systems with selective trunk reservation. *Performance Evaluation*, *41*(4), 295–306.

Gordon, W. J., & Newell, G. F. (1967a). Closed queuing systems with exponential servers. *Operations Research*, *15*(2), 254–265.

Gordon, W. J., & Newell, G. F. (1967b). Cyclic queuing systems with restricted length queues. *Operations Research*, *15*(2), 266–277.

Harrison, P. G. (2004). Reversed processes, product forms and a non-product form. *Linear Algebra and its Applications*, *386*, 359–381.

Henderson, W. & Taylor, P.G. (1988). Alternative routing networks and interruptions. In *Proceedings of the 12th International Teletraffic Conference* (pp. 5.1B.2.1-5.1B.2.7). Torino.

Hordijk, A., & Schassberger, R. (1982). Weak convergence for generalized semi-Markov processes. *Stochastic Processes and their Applications*, *12*(3), 271–291.

Hordijk, A., & Van Dijk, N. M. (1983a). Networks of queues. Part I: Jobal-local-balance and the adjoint process. Part II: General routing and service characteristics. *Lecture Notes in Control and Information Sciences*, *60*, 158–205.

Hordijk, A., & Van Dijk, N. M. (1983b). Adjoint processes, job-local-balance and insensitivity for stochastic networks. *Bulletin 44th of the International Statistical Institute*, *50*, 776–788.

Hordijk, A., & Ridder, A. (1987). Stochastic inequalities for an overflow model. *Journal of Applied Probability*, *24*(3), 696–708.

Hordijk, A., & Ridder, A. (1988). Insensitive bounds for the stationary distribution of non-reversible Markov chains. *Journal of Applied Probability*, *25*(1), 9–20.

Jackson, J. R. (1957). Networks of waiting lines. *Operations Research*, *5*(4), 518–521.

Jackson, J. R. (1963). Jobshop-like queueing systems. *Management Science*, *10*(1), 131–142.

Kaufman, J. S. (1981). Blocking in a shared resource environment. *IEEE Transactions on Communications*, *29*(10), 1474–1481.

Kelly, F. P. (1979). *Reversibility and Stochastic Networks*. New York: Wiley.

Kingman, J. F. C. (1969). Markov population processes. *Journal of Applied Probability*, *6*(1), 1–18.

Lam, S. S. (1977). Queueing networks with population size constraints. *IBM Journal of Research and Development*, *21*(4), 370–378.

Litvak, N., Van Rijsbergen, M., Boucherie, R. J., & Van Houdenhoven, M. (2008). Managing the overflow of intensive care patients. *European Journal of Operational Research*, *185*(3), 998–1010.

Pittel, B. (1979). Closed exponential networks of queues with saturation: The Jackson-type stationary distribution and its asymptotic analysis. *Mathematics of Operations Research*, *4*(4), 357–378.

Schassberger, R. (1977). Insensitivity of steady-state distributions of generalized semi-Markov processes. *Part I. The Annals of Probability*, *5*(1), 87–99.

Schassberger, R. (1978). Insensitivity of steady-state distributions of generalized semi-Markov processes. *Part II. The Annals of Probability*, *6*(1), 85–93.

Shortle, J. F. (2004). An Equivalent Random Method with hyper-exponential service. *Performance Evaluation*, *57*(3), 409–422.

Stewart, W. J. (2009). *Probability, Markov Chains, queues, and simulation: The mathematical basis of performance modeling*. Princeton: Princeton University Press.

Taylor, P. G. (2011). Insensitivity in stochastic models. In R. J. Boucherie & N. M. van Dijk (Eds.), *Queueing networks: A fundamental approach* (pp. 121–140). New York: Springer.

Van Dijk, N. M. (1989). A proof of simple insensitive bounds for a pure overflow system. *Journal of Applied Probability*, *26*(1), 113–120.

Van Dijk, N. M. (1993). *Queueing networks and product forms: A system's approach*. Chichester: Wiley.

Van Dijk, N. M., & Van der Sluis, E. (2009). Call packing bound for overflow loss systems. *Performance Evaluation*, *66*(1), 1–20.

Van Dijk, N. M. (2011). On practical product form characterizations. In R. J. Boucherie & N. M. van Dijk (Eds.), *Queueing networks: A fundamental approach* (pp. 1–83). New York: Springer.

Van Doorn, E. A. (1984). On the overflow process from a finite Markovian queue. *Performance Evaluation*, *4*(4), 233–240.

Van Marion, E. W. B. (1968). Influence of holding time distributions on blocking probabilities of a grading. *TELE*, *20*, 17–20.

Wilkinson, R. I. (1956). Theories for toll traffic engineering in the USA. *Bell System Technical Journal*, *35*(2), 421–514.