Performance analysis of parallel identical machines with a generalized shortest queue arrival mechanism

Performance Analysis of Parallel Identical Machines with a Generalized Shortest Queue Arrival Mechanism

G.J. van Houtum[a,*], I.J.B.F. Adan[b], J. Wessels[b], W.H.M. Zijm[a,c]

[a] Eindhoven University of Technology, Faculty of Technology Management

[b] Eindhoven University of Technology, Faculty of Mathematics and Computing Science

[c] University of Twente, Faculty of Applied Mathematics

January 2000

Abstract

In this paper we study a production system consisting of a group of parallel machines producing multiple job types. Each machine has its own queue and it can process a restricted set of job types only. On arrival a job joins the shortest queue among all queues capable of serving that job. Under the assumption of Poisson arrivals and identical exponential processing times we derive upper and lower bounds for the mean waiting time. These bounds are obtained from so-called flexible bound models, and they provide a powerful tool to efficiently determine the mean waiting time. The bounds are used to study how the mean waiting time depends on the amount of overlap (i.e. common job types) between the machines.

Keywords: Queueing system, shortest queue routing, performance analysis, flexibility, truncation model, bounds

*Corresponding author. Mailing address: Eindhoven University of Technology, Faculty of Technology Management, P.O. Box 513, 5600 MB Eindhoven, The Netherlands. Phone: +31 40 2475163. Fax: +31 40 2464596. E-mail: G.J.v.Houtum@tm.tue.nl .

1 Introduction

In this paper we consider a queueing system consisting of a group of parallel identical servers serving multiple job types. Each server has its own queue and is capable of serving a restricted set of job types only. Jobs arrive according to a Poisson process and on arrival they join the shortest feasible queue. The service times are exponentially distributed. We will refer to this queueing model as the *Generalized Shortest Queue System* (GSQS). This model is motivated by a situation encountered in the assembly of Printed Circuit Boards (PCBs). This is explained in more detail below.

Figure 1 shows a typical layout of an assembly system for PCBs. It consists of three parallel *insertion machines*, each with its own local buffer. An insertion machine mounts vertical components, such as resistors and capacitators, on a PCB by the *insertion head*. The components are mounted in a certain sequence, which is prescribed by a Numerical Control program. The insertion head is fed by the *sequencer*, which picks components from tapes and transports them in the right order to the insertion head. Each tape contains only *one* type of components. The tapes are stored in the *component magazine*, which can contain at most 80 tapes, say. Each PCB needs on average 60 different types of components. To assemble a PCB all required components have to be available in the component magazine. Hence, the set of components available in the magazine determines the set of PCB types that can be processed on that machine. The system in Figure 1 has to assemble three PCB types, labeled A, B and C. The machines are basically similar, but due to the fact that they are loaded with different types of components, the sets of PCB types that can be handled by the machines are different. Machine M_1 can handle the A and B types, machine M_2 the A and C, and machine M_3 the B and C. When the mounting times for all PCB types are approximately the same, it is reasonable to send arriving PCBs to the shortest feasible queue.

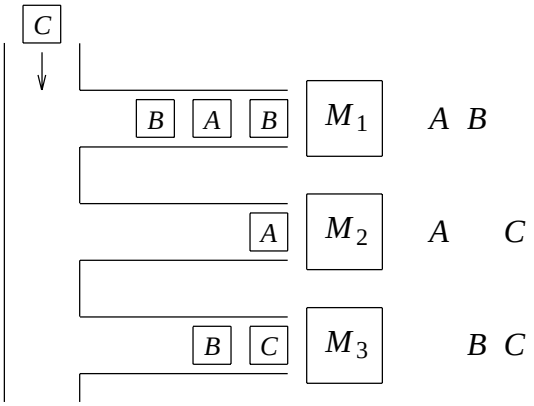

Figure 1: *A flexible assembly system consisting of three parallel insertion machines, on which three types of PCBs are made.*

Since the assembly of PCBs is often characterized by relatively few job types, large production batches and small mounting times (see Zijm [16]), the use of a queueing model seems appropriate to predict performance characteristics such as the mean waiting time. An important issue is the assignment of the required components to the machines. Ideally, each machine should get all components needed to process all PCB types. However, since

the component magazines have a finite capacity, they can contain the components needed for a (small) subset of PCB types only. In this paper we will investigate how much overlap (i.e. common components) between the machines is required such that the system nearly performs as in the ideal situation where the machines are equipped with all components.

The GSQS is also relevant for many other practical situations; e.g., for parallel machines loaded with different sets of tools, computer disks loaded with different information files, or operators in a call center handling requests from different customers. Nevertheless, the literature on the GSQS is limited. Schwartz [12] (see also Roque [11]) considered a system related to the GSQS, but with a specific server hierarchy. He derived some expressions for the mean waiting times. Adan, Wessels and Zijm [2] derived rough approximations for the mean waiting times in a GSQS. Green [7] constructed a truncation model for a related system with two types of jobs and two types of servers: servers which can serve both job types and servers which can only serve jobs of the second type.

For the present model with general (i.e. nonexponential) arrivals, Sparaggis, Cassandras and Towsley [13] showed that the generalized shortest queue routing is optimal with respect to the overall mean waiting time for symmetric cases (see Theorem 3.1 in [13]; see also Subsection 2.3). For more general systems, Foss and Chernova [6] used a fluid approximation approach to establish ergodicity conditions (see also the remarks at the end of Subsection 2.2). The issue of ergodicity has also been considered in a recent report by Foley and McDonald [5]. Their main contribution, however, consists of results on the asymptotic behavior of a GSQS with two exponential servers with different service rates. Finally, Hassin and Haviv [8] have studied a symmetric GSQS with two servers and an additional property called threshold jockeying. They focus on the difference in waiting time between jobs which can choose between both servers and jobs which can not choose.

The GSQS can be described by a continuous-time Markov process with multi-dimensional states where each component denotes the queue length at one of the servers. Only in very special cases exact analytical solutions can be found (see e.g. [3]). Therefore, to determine the mean waiting times, we will construct truncation models which: (i) are flexible (i.e. the size of their state space can be controlled by one or more truncation parameters); (ii) can be solved efficiently; (iii) provide upper and lower bounds for the mean waiting times. Such models are called solvable flexible bound models. They are derived by using the so-called the *precedence relation method*. This is a systematic approach for the construction of bound models, which has been developed in [14, 15]. In this paper we will construct a lower and upper bound model for the mean waiting times. These two models constitute the core of a powerful numerical approach: the two bound models are solved for increasing sizes of the truncated state space until the mean waiting times are determined *within a given, desired accuracy*.

This paper is organized as follows. In Section 2, we describe the GSQS and we discuss conditions under which the GSQS is ergodic and balanced. Next, in Section 3, we construct the flexible bound models and we formulate a numerical approach to determine the mean waiting times. Finally, in Section 4, we investigate how the mean waiting times for the GSQS depend on the amount of overlap (i.e. common job types) between the servers. This is done by numerically evaluating several scenarios.

2 Model

This section consists of three subsections. In the first subsection, we describe the GSQS. In Subsection 2.2 we present a simple condition that is necessary and sufficient for ergodicity. In the last subsection, we present a related condition under which the GSQS is said to be *balanced* and we briefly discuss *symmetric* systems.

2.1 Model description

The GSQS consists of $c \geq 2$ parallel servers serving multiple job types. Each server has its own queue and is capable of serving a restricted set of job types only. All service times are exponentially distributed with the same parameter $\mu > 0$. The arrival stream of each job type is Poisson and an arriving job joins the shortest queue among all queues capable of serving that job (ties are broken with equal probabilities). Figure 2 shows a GSQS with $c = 2$ servers and three job types: type A, B and C jobs arrive with intensity λ_A, λ_B and λ_C, respectively. The A jobs can be served by both servers, the B jobs can only be served by server 1, and the C jobs must be served by server 2.

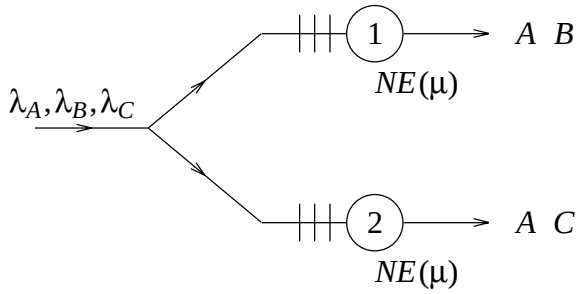

Figure 2: *A GSQS with $c = 2$ servers and three job types.*

We introduce the following notations. The servers are numbered from $1, \ldots, c$ and the set I is defined by $I = \{1, \ldots, c\}$. The set of all jobs type is denoted by J. The arrival intensity of type $j \in J$ jobs is given by $\lambda_j \geq 0$, and $\lambda = \sum_{j \in J} \lambda_j$ is the total arrival intensity. We further assume that each job type can be served by at least one server and each server can handle at least one job type; so, $I(j) \neq \emptyset$ for all $j \in J$, and $\cup_{j \in J} I(j) = I$. Without loss of generality, we set $\mu = 1$. Then the average workload per server is given by $\rho = \lambda / c$. Obviously, the requirement $\rho < 1$ is necessary for ergodicity.

The behavior of the GSQS is described by a continuous-time Markov process with states $(m_1, \ldots, m_c)$, where m_i denotes the length of the queue at server i, $i \in I$ (jobs in service are included). So, the state space is equal to

$$M = \{m \mid m = (m_1, \ldots, m_c) \text{ with } m_i \in \mathbb{N}_0 \text{ for all } i \in I\} . \tag{1}$$

We assume that $\sum_{j \in J} \lambda_j 1_{\{i \in I(j)\}} > 0$ for all servers $i \in I$ (here, $1_{\{G\}}$ is the indicator function, which is 1 if G is true and 0 otherwise), i.e., that all servers have a positive *potential* arrival rate. This guarantees that the Markov process is irreducible. The transition rates are denoted by $q_{m,n}$. Figure 3 shows the transition rates for the GSQS in Figure 2.

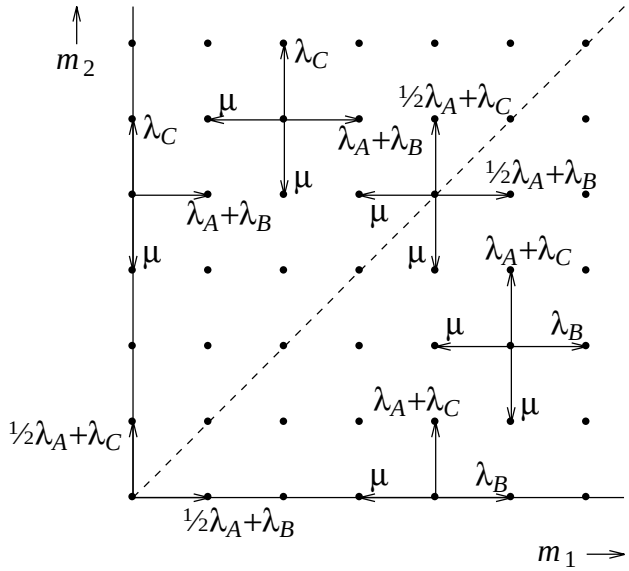

Figure 3: *The transition rate diagram for the GSQS in Figure 2.*

The relevant performance measures are the mean waiting times $W^{(j)}$ for each of job type $j \in J$, and the overall mean waiting time W, which is equal to

$$W = \sum_{j \in J} \frac{\lambda_j}{\lambda} W^{(j)} . \tag{2}$$

It is obvious that for an ergodic system,

$$W^{(j)} = \sum_{(m_1,\ldots,m_c) \in M} \left(\min_{i \in I(j)} m_i \right) \pi(m_1, \ldots, m_c) , \qquad j \in J, \tag{3}$$

where $\pi(m_1, \ldots, m_c)$ denotes the steady-state probability for state $(m_1, \ldots, m_c)$.

2.2 Ergodicity

By studying the job routing, we obtain a simple, *necessary* condition for the ergodicity of the GSQS. For each subset $J' \in J$, $J \neq \emptyset$, jobs of type $j \in J'$ arrive with an intensity equal to $\sum_{j \in J'} \lambda_j$ and they must be served by the servers $\cup_{j \in J'} I(j)$. This immediately leads to the following lemma.

Lemma 1 *The GSQS can only be ergodic if*

$$\sum_{j \in J'} \lambda_j < |\cup_{j \in J'} I(j)| \qquad \textit{for all } J' \subset J,\ J' \neq \emptyset. \tag{4}$$

Note that for $J' = J$, this inequality is equivalent to $\rho < 1$. For the GSQS in Figure 2, condition (4) states that for ergodicity it is necessary that the inequalities $\lambda_B < 1$, $\lambda_C < 1$ and $\lambda < 2$ (or, equivalently, $\rho < 1$) are satisfied. It appears that condition (4)

is also sufficient for ergodicity. To show this, we consider so-called corresponding static systems.

A *corresponding static system* is a system that is identical to the GSQS, but with *static (random) routing* instead of dynamic shortest queue routing. The static routing is described by discrete distributions $\{x_i^{(j)}\}_{i\in I(j)}$, $j \in J$, where for each $j \in J$ and $i \in I(j)$, the variable $x_i^{(j)}$ denotes the probability that an arriving job of type j is sent to server i. Under static routing, it holds for each $j \in J$ that the Poisson stream of arriving type j jobs is split up into Poisson streams with intensities $x_{j,i} = \lambda_j x_i^{(j)}$, $i \in I(j)$, for type j arrivals joining server i. Hence the queues $i \in I$ constitute independent $M/M/1$ queues with identical mean service times equal to $\mu = 1$ and arrival intensities $\sum_{j\in A(i)} x_{j,i}$, where $A(i) = \{j \in J | i \in I(j)\}$. As a result, we obtain a simple necessary and sufficient condition for the ergodicity of a corresponding static system, viz.

$$\sum_{j\in A(i)} x_{j,i} \;<\; \mu \qquad \text{for all } i \in I.$$

Lemma 2 *For a GSQS, there exists a corresponding static system that is ergodic, if and only if condition (4) is satisfied.*

Proof. There exists a corresponding static system that is ergodic if and only if there exists a nonnegative solution $\{x_{j,i}\}_{(j,i)\in A}$ of the following equations and inequalities:

$$\sum_{i\in I(j)} x_{j,i} \;=\; \lambda_j \;\text{ for all } j \in J, \qquad \sum_{j\in A(i)} x_{j,i} \;<\; 1 \;\text{ for all } i \in I; \tag{5}$$

the equalities in (5) guarantee that the solution $\{x_{j,i}\}_{(j,i)\in A}$ corresponds to discrete distributions $\{x_i^{(j)}\}_{i\in I(j)}$ which describe a static routing, and the inequalities in (5) must be satisfied for ergodicity. It is easily seen that (5) has no solution if condition (4) is not satisfied.

Now, assume that condition (4) is satisfied. To prove that there exists a nonnegative solution $\{x_{j,i}\}_{(j,i)\in A}$ of (5), we consider a *transportation problem* with supply nodes $\hat{V}_1 = J \cup \{0\}$, demand nodes $\hat{V}_2 = I$, and arcs $\hat{A} = A \cup \{(0,i) | i \in I\}$, with

$$A \;=\; \{(j,i) \mid j \in J,\; i \in I \text{ and } i \in I(j)\}\,.$$

(supply node 0 denotes an extra type of jobs, which can be served by all servers). Define the supplies $\hat{a}_j$ by $\hat{a}_j = \lambda_j$ for all $j \in \hat{V}_1 \setminus \{0\}$ and $\hat{a}_0 = c - \lambda - c\epsilon$, where

$$\epsilon \;:=\; \min_{\substack{J'\subset J\\ J'\neq\emptyset}} \frac{|\cup_{j\in J'} I(j)| \;-\; \sum_{j\in J'} \lambda_j}{|\cup_{j\in J'} I(j)|}$$

(from (4), it follows that $\epsilon > 0$, and $\hat{a}_0 \geq 0$ since by taking $J' = J$ we obtain the inequality $\epsilon \leq (c-\lambda)/c$). Further, we define the demands $\hat{b}_i$ by $\hat{b}_i = 1 - \epsilon$ for all $i \in \hat{V}_2$; note that $\sum_{j\in\hat{V}_1} \hat{a}_j = \sum_{i\in\hat{V}_2} \hat{b}_i$. It may be verified that this transportation problem satisfies a

necessary and sufficient condition for the existence of a feasible flow; see Lemma 5.4 of [14] and its proof is based on a transformation to a maximum-flow problem followed by the application of the max-flow min-cut theorem (see e.g. [4]). So, there exists a feasible flow for the transportation problem, i.e., there exists a nonnegative solution $\{\hat{x}_{j,i}\}_{(j,i)\in\hat{A}}$ of the equations

$$\sum_{\substack{i\in\hat{V}_2\\(j,i)\in\hat{A}}} \hat{x}_{j,i} \;=\; \hat{a}_j \;\text{ for all } j\in\hat{V}_1, \qquad \sum_{\substack{j\in\hat{V}_1\\(j,i)\in\hat{A}}} \hat{x}_{j,i} \;=\; \hat{b}_i \;\text{ for all } i\in\hat{V}_2.$$

It is easily seen that then the solution $\{x_{j,i}\}_{(j,i)\in A}$ defined by $x_{j,i}=\hat{x}_{j,i}$ for all $(j,i)\in A$, is a nonnegative solution of (5), which completes the proof. □

In situations with many job types shortest queue routing will balance the queue lengths more than any static routing. So if there is a corresponding static system that is ergodic, then the GSQS will also be ergodic. Together with Lemma 2, this informally shows that the following theorem holds.

Theorem 1 *The GSQS is ergodic if and only if condition (4) is satisfied.*

For a formal proof of this theorem, the reader is referred to Foss and Chernova [6] or Foley and McDonald [5]. In the latter paper, a generalization of condition (4) is proved to be necessary and sufficient for the (more general) model with different service rates. Their proof also exploits the connection with a corresponding static system. Foss and Chernova [6] use a fluid approximation approach to derive necessary conditions for a model with general arrivals and general service times.

2.3 Balanced and symmetric systems

It is desirable that the shortest queue routing, as reflected by the sets $I(j)$, balances the workload among the servers. Formally, we say that a GSQS is *balanced* if there exists a corresponding static system for which all queues have the same workload. This means that there must exist discrete distributions $\{x_i^{(j)}\}_{i\in I(j)}$ such that for each server $i\in I$, the arrival intensity $\sum_{j\in J,\ (j,i)\in A} x_{j,i}$ is equal to $\lambda/c=\rho$, where the $x_{j,i}$ and the set A are defined as before. Such discrete distributions exist if and only if there exists a nonnegative solution $\{x_{j,i}\}_{(j,i)\in A}$ of the equations

$$\sum_{\substack{i\in I\\(j,i)\in A}} x_{j,i} \;=\; \lambda_j \;\text{ for all } j\in J, \qquad \sum_{\substack{j\in J\\(j,i)\in A}} x_{j,i} \;=\; \frac{\lambda}{c} \;\text{ for all } i\in I. \tag{6}$$

These equations are precisely the equations which must be satisfied by a feasible flow for the transportation problem with supply nodes $V_1=J$, demand nodes $V_2=I$, arcs A, supplies $a_j=\lambda_j$ for all $j\in V_1$ and demands $b_i=\lambda/c$ for all $i\in V_2$. Applying the necessary and sufficient condition for the existence of such a feasible flow (see [14]) leads to the following lemma.

Lemma 3 *A GSQS is balanced if and only if*

$$\sum_{j \in J'} \lambda_j \;\leq\; |\cup_{j \in J'} I(j)|\, \frac{\lambda}{c} \qquad \textit{for all } J' \subset J. \tag{7}$$

Note that for $J' = \emptyset$ and $J' = J$, condition (7) holds by definition. Further, it follows that a balanced GSQS satisfies condition (4) if and only if $\rho < 1$. So, for a balanced GSQS, the simple condition $\rho < 1$ is necessary and sufficient for the ergodicity.

For a balanced GSQS the workloads under the shortest queue routing are not necessarily balanced. This can be seen by considering the GSQS in Figure 2. According to condition (7), this GSQS is balanced if and only if $\lambda_B \leq \lambda/2$ and $\lambda_C \leq \lambda/2$, i.e. if and only if $\lambda_B \leq \lambda_A + \lambda_C$ and $\lambda_C \leq \lambda_A + \lambda_B$. This condition is obviously satisfied if we take $\lambda_C = \lambda_A + \lambda_B$. In this case, equal workloads for both servers can only be obtained if all jobs of type A are sent to server 1. But, under the shortest queue routing, it will still occur that jobs of type A are sent to server 2, and therefore server 2 will have a higher workload than server 1. Nevertheless, one may expect that for a balanced GSQS, the shortest queue routing at least ensures that the workloads will not differ too much.

A subclass of balanced systems are the symmetric systems. A GSQS is said to be *symmetric*, if

$$\lambda(I_1) \;=\; \lambda(I_2) \qquad \text{for all } I_1, I_2 \subset I \text{ with } |I_1| = |I_2|, \tag{8}$$

where

$$\lambda(I') \;:=\; \sum_{\substack{j \in J \\ I(j) = I'}} \lambda_j \;, \qquad I' \subset I.$$

So, a GSQS is symmetric, if for all subsets $I' \subset I$ with the same number of servers $|I'|$, the arrival intensity $\lambda(I')$ for the jobs which can be served by precisely the servers of I', is the same. The GSQS in Figure 2 is symmetric if $\lambda_B = \lambda_C$.

For a symmetric GSQS, all queue lengths have the same distribution, which implies that all servers have equal workloads. For such a system, it follows from Sparaggis *et al.* [13], that the shortest queue routing minimizes the total number of jobs in the system and hence the overall mean waiting time W. In particular, this implies that the overall mean waiting time in a symmetric GSQS is less than in the corresponding system consisting of N independent $M/M/1$ queues with workload ρ.

3 Flexible bound models

In this section we construct two truncation models which are much easier to solve than the original model. One truncation model produces lower bounds for the mean waiting times, and the other one upper bounds. At the end of this section we describe a numerical method for the computation of the mean waiting times within a given, desired accuracy.

The truncation models exploit the property that the shortest queue routing causes a drift towards states with equal queue lengths. The state space M' of the two models is obtained by truncating the original state space M around the diagonal, i.e.,

$$M' = \{m \in M \mid m = (m_1, \ldots, m_c) \text{ and } m_i \leq \min(m) + T_i \text{ for all } i \in I\}, \qquad (9)$$

where $\min(m) := \min_{i \in I} m_i$ and $T_1, \ldots, T_c \in \mathbb{N}$ are so-called threshold parameters; the corresponding vector $\hat{T} := (T_1, \ldots, T_c)$ is called the threshold vector. So state $m \in M$ also lies in M' if and only if for each $i \in I$ the length of queue i is at most T_i greater than the length of any other queue. Later on in this section we discuss how appropriate values for $\hat{T}$ can be selected. There are two types of transitions pointing from states inside M' to states outside M':

(i) in state $m = (m_1, \ldots, m_c) \in M'$ with $\min(m) > 0$ and $I' = \{i \in I | m_i = \min(m) + T_i\} \neq \emptyset$, at a server $k \in I$ with $m_k = \min(m)$ a service completion occurs with rate μ and leads to a transition from m to state $n = m - e_k \notin M'$;

(ii) in state $m = (m_1, \ldots, m_c) \in M'$ with $I' = \{i \in I | m_i = \min(m) + T_i\} \neq \emptyset$, at a server $i \in I'$ an arrival of a new job leads to a transition from m to the state $n = m + e_i \notin M'$; this transition occurs with rate $\sum_{j \in J} |I(j; m)|^{-1} \lambda_j 1_{\{i \in I(j;m)\}}$, where the set $I(j; m)$ is defined by $I(j; m) = \{i \in I(j) \mid m_i = \min_{k \in I(j)} m_k\}$ (note that this rate may be equal to 0).

In the lower (upper) bound model, the transitions to states n outside M' are redirected to states n' with less (more) jobs inside M'.

In the lower bound model, the transition in (i) is redirected to $n' = m - e_k - \sum_{i \in I'} e_i \in M'$. This means that the departure of a job at a non-empty shortest queue is accompanied by killing one job at each of the queues $i \in I'$, which are already T_i greater than the shortest queue. The transition in (ii) is redirected to m itself, i.e., a new job arriving at one of the servers $i \in I'$ is rejected. The lower bound model is therefore called the *Threshold Killing and Rejection (TKR) model*.

In the upper bound model, the transition in (i) is redirected to m itself. This means that if at least one queue is already T_i greater than the shortest queue, the finished job in the shortest queue is not allowed to depart, but is served once more; this is equivalent to saying that the servers at the shortest queues are blocked. Transition (ii) is redirected to $n' = m + e_i + \sum_{k \in I_{sq}} e_k \in M'$, with $I_{sq} = \{k \in I | m_k = \min(m)\}$. This means that an arrival of a new job at one of the queues which is already T_i greater than the shortest queue, is accompanied by the addition of one extra job at each of the shortest queues. The upper bound model is therefore called the *Threshold Blocking and Addition (TBA) model*. Note that this model may be non-ergodic while the original model is ergodic. However, the larger the values of the thresholds T_i the more unlikely this situation. In Figure 4, we show the redirected transitions in the lower and upper bound model for the GSQS of Figure 3.

It is intuitively clear that the queues in the TKR model are stochastically smaller than the queues in the original model. Hence, for each $j \in J$, the TKR model yields a lower

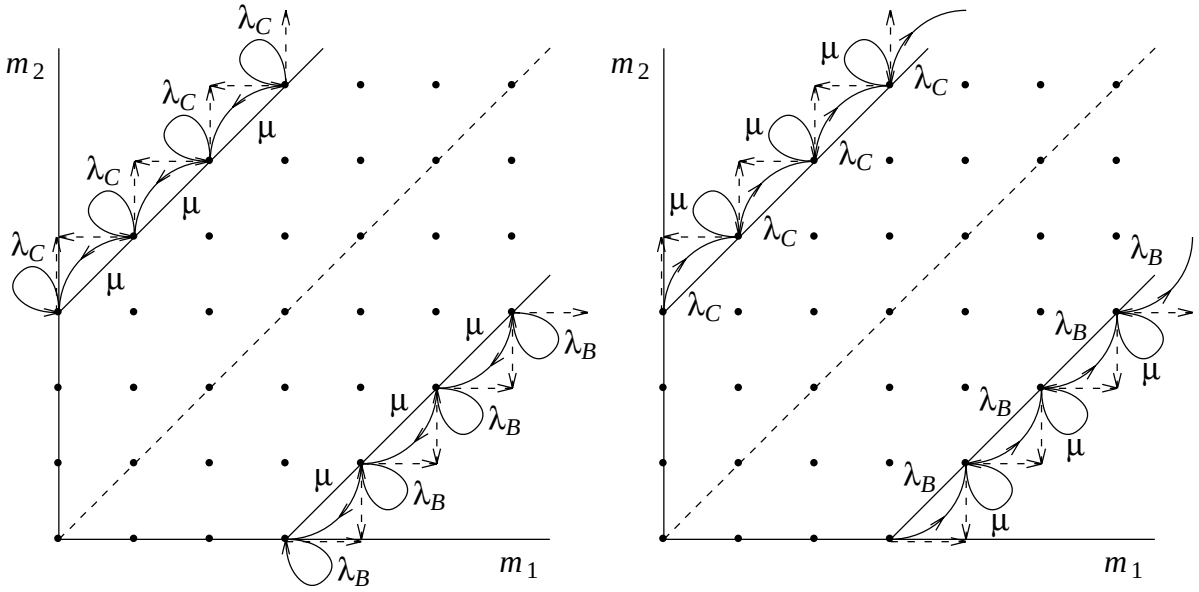

Figure 4: *The redirected transitions in the TKR and TBA model for the GSQS depicted in Figure 2. For both models,* $\hat{T} = (T_1, T_2) = (3, 3)$.

bound for the mean length of the shortest queue among the queues $i \in I(j)$, and thus also for the mean waiting time of type j jobs (cf. (3)). Denote the steady-state probabilities in the TKR model by $\pi_{TKR}(m_1, \ldots, m_c)$ and let

$$W_{TKR}^{(j)}(\hat{T}) = \sum_{(m_1,\ldots,m_c)\in M'} \left(\min_{i\in I(j)} m_i \right) \pi_{TKR}(m_1, \ldots, m_c) \,, \qquad j \in J.$$

Then we have for each $j \in J$ that $W_{TKR}^{(j)}(\hat{T}) \leq W^{(j)}$, and thus (cf. (2))

$$W_{TKR}(\hat{T}) = \sum_{j\in J} \frac{\lambda_j}{\lambda} \, W_{TKR}^{(j)}(\hat{T})$$

yields a lower bound for the overall mean waiting time W. The lower bounds $W_{TKR}^{(j)}(\hat{T})$ monotonically increase as the thresholds $T_1, \ldots, T_c$ increase. Similarly the TBA model produces monotonically decreasing upper bounds $W_{TBA}^{(j)}(\hat{T})$, $j \in J$, and $W_{TBA}(\hat{T})$. The bounds and the monotonicity properties can be rigorously proved by using the *precedence relation method*, see [14]. This method is based on Markov reward theory and it has been developed in [14, 15].

The truncation models can be solved efficiently by using the *matrix-geometric approach* described in [10]. Since the truncation models exploit the property that shortest queue routing tries to balance the queues, one may expect that the bounds are tight for already moderate values of the thresholds $T_1, \ldots, T_c$.

We will now formulate a numerical method to determine the mean waiting times with an absolute accuracy ϵ_{abs}. The method repeatedly solves the TKR and TBA model for

increasing threshold vectors $\hat{T} = (T_1, \ldots, T_c)$. For each vector $\hat{T}$ we use $(W^{(j)}_{TKR}(\hat{T}) + W^{(j)}_{TBA}(\hat{T}))/2$ as an approximation for $W^{(j)}$ and and $\Delta^{(j)}(\hat{T}) = (W^{(j)}_{TBA}(\hat{T}) - W^{(j)}_{TKR}(\hat{T}))/2$ as an upper bound for the error; we similarly approximate W by $(W_{TKR}(\hat{T}) + W_{TBA}(\hat{T}))/2$ where the error is at most $\Delta(\hat{T}) = (W_{TBA}(\hat{T}) - W_{TKR}(\hat{T}))/2$. The approximations and error bounds are set equal to ∞ if the TBA model is not ergodic (which may be the case for small thresholds). The computation procedure stops when all error bounds are less than or equal to ϵ_{abs}; otherwise at least one of the thresholds is increased by 1 and new approximations are computed. The decision to increase a threshold T_i is based on the rate of redirections $r_{rd}(i)$. This is explained in the next paragraph.

The variable $r_{rd}(i)$, $i \in I$, denotes the rate at which redirections occur in the boundary states $m = (m_1, \ldots, m_c)$ with $m_i = \min(m) + T_i$ of the truncated state space. If for given $\hat{T}$ only the TKR model is ergodic, then $r_{rd}(i)$ denotes the rate for the TKR model, otherwise $r_{rd}(i)$ denotes the sum of the rate for the TKR and TBA model. The rates $r_{rd}(i)$ can be computed directly from the steady-state distributions of the bound models. The higher the rate $r_{rd}(i)$, the higher the expected impact of increasing T_i. The computation procedure increases all thresholds T_i for which $r_{rd}(i) = \max_{k \in I} r_{rd}(k)$. The numerical method is summarized below.

Algorithm *(to determine the mean waiting times for the GSQS)*

Input:	The data of an ergodic instance of the GSQS, i.e., c, J, $I(j)$ for all $j \in J$, and λ_j for all $j \in J$; the absolute accuracy ϵ_{abs}; the initial threshold vector $\hat{T} = (T_1, \ldots, T_c)$.
Step 1.	Determine $W^{(j)}_{TKR}(\hat{T})$, $W^{(j)}_{TBA}(\hat{T})$ and $\Delta^{(j)}(\hat{T})$ for all $j \in J$, and $W_{TKR}(\hat{T})$, $W_{TBA}(\hat{T})$ and $\Delta^{(j)}(\hat{T})$, and $r_{rd}(i)$ for all $i \in I$.
Step 2.	If $\Delta^{(j)}(\hat{T}) > \epsilon_{abs}$ for some $j \in J$ or $\Delta(\hat{T}) > \epsilon_{abs}$, then $T_i := T_i + 1$ for all $i \in I$ with $r_{rd}(i) = \max_{k \in I} r_{rd}(k)$, and return to Step 1.
Step 3.	$W^{(j)} = (W^{(j)}_{TKR}(\hat{T}) + W^{(j)}_{TBA}(\hat{T}))/2$ for all $j \in J$, and $W = (W_{TKR}(\hat{T}) + W_{TBA}(\hat{T}))/2$.

Note that for a symmetric GSQS it is natural to start with a threshold vector $\hat{T}$ with equal components. Then in each iteration all rates $r_{rd}(i)$ will be equal, and hence each T_i will be increased by 1. So the components of $\hat{T}$ will remain equal.

4 Numerical study of the GSQS

In this section we consider three scenarios. In Subsection 4.1 we distinguish two types of jobs: *common jobs* and *specialist jobs*. The common jobs can be served by all servers and the other ones can be served by only one specific server. We focus on the behavior of the overall mean waiting time W as a function of the fraction of work due to common jobs. The higher this fraction, the more balanced the queues and the better the performance. So

W will be decreasing as the number of common jobs increases. In one extreme case, viz. when all jobs are specialist jobs, the GSQS reduces to independent $M/M/1$ queues, and W is maximal. In the other extreme case, viz. when all jobs are common jobs, the GSQS is identical to a pure Symmetric Shortest Queue System (SSQS), and W is minimal. In Subsection 4.1 we investigate how W behaves in between these two extremes.

In Subsection 4.2 we consider a symmetric GSQS with $c = 3$ servers, and, besides common and specialist jobs, we also have *semi-common jobs*. These jobs can be served by two servers. We compare two situations: (i) a GSQS with a given fraction of common jobs (and no semi-common jobs); (ii) a GSQS with twice this fraction of semi-common jobs (and no common jobs). In both cases the average number of servers capable of serving an arbitrary job is the same. In Subsection 4.3 we evaluate a series of balanced, asymmetric systems. We investigate how the mean waiting times deteriorate due to the asymmetry. Finally, in Subsection 4.4, the main conclusions are summarized.

4.1 The impact of common jobs

We distinguish $c + 1$ job types, numbered $1, \ldots, c, c + 1$. Type j jobs are specialist jobs, which can only be served by server j, $j = 1, \ldots, c$. The type $c + 1$ jobs are common jobs, which can be served by all servers. The total arrival intensity is equal to $\lambda = c\rho$, with $\rho \in (0, 1)$. The common jobs constitute a fraction p, $p \in [0, 1]$, of the total arrival stream, while each of the streams of specialist jobs constitutes an equal part of the remaining stream. So $\lambda_{c+1} = p\lambda$ and $\lambda_j = (1 - p)\lambda/c$ for $j = 1, \ldots, c$.

Table 1 lists the mean waiting times for specialist jobs ($= W^{(1)} = \ldots = W^{(c)}$), common jobs ($= W^{(c+1)}$), and an arbitrary job ($= W$) as a function of p for a system with $c = 2$ and $c = 3$ servers, respectively. For $p = 0$ there are no common jobs; then $W^{(c+1)}$ is defined as the limiting value of the waiting time of common jobs as $p \downarrow 0$. For $p = 1$ a similar remark holds for the mean waiting times $W^{(1)} = \cdots = W^{(c)}$. Table 1 also lists the *realized reduction* $rr(p)$. This is defined as

$$rr(p) \;=\; \frac{W_{M/M/1} - W}{W_{M/M/1} - W_{SSQS}}\,, \tag{10}$$

where $W_{M/M/1}$ and W_{SSQS} denote the mean waiting time in an $M/M/1$ system and SSQS, respectively, both with the same workload $\rho = 0.9$ and mean service time $\mu = 1$ as for the GSQS. The mean waiting time $W_{M/M/1}$ is realized when $p = 0$, and W_{SSQS} is realized when $p = 1$. Clearly, $rr(0) = 0$ and $rr(1) = 1$ by definition. For all cases in Table 4.1, $W_{M/M/1} = 9$ and $W_{SSQS} = 4.475$ for $c = 2$ and $W_{SSQS} = 2.982$ for $c = 3$. The mean waiting times in the SSQS have been determined with an absolute accuracy of 0.0001 by using the bound models in [1]. The mean waiting times in Table 1 have been determined by using the algorithm described in Section 3 with an absolute accuracy $\epsilon_{abs} = 0.005$.

	$c=2$				$c=3$			
p	$W^{(1)}$	$W^{(c+1)}$	W	$rr(p)$	$W^{(1)}$	$W^{(c+1)}$	W	$rr(p)$
0.0	9.00	4.26	9.00	0.0 %	9.00	2.69	9.00	0.0 %
0.1	6.80	4.36	6.56	54.0 %	6.07	2.82	5.75	54.1 %
0.2	6.04	4.40	5.72	72.6 %	5.06	2.88	4.63	72.7 %
0.3	5.66	4.43	5.29	82.0 %	4.56	2.91	4.06	82.0 %
0.4	5.43	4.44	5.04	87.6 %	4.25	2.93	3.72	87.7 %
0.5	5.28	4.45	4.86	91.4 %	4.05	2.95	3.50	91.4 %
0.6	5.17	4.46	4.74	94.1 %	3.90	2.96	3.34	94.1 %
0.7	5.09	4.46	4.65	96.1 %	3.79	2.97	3.21	96.1 %
0.8	5.02	4.47	4.58	97.7 %	3.71	2.97	3.12	97.7 %
0.9	4.97	4.47	4.52	99.0 %	3.64	2.98	3.04	99.0 %
1.0	4.93	4.48	4.48	100.0 %	3.58	2.98	2.98	100.0 %

Table 1: *Mean waiting times as a function of p and c.*

In Table 1 we see that the overall mean waiting time $W = pW^{(c+1)} + (1-p)W^{(1)}$ sharply decreases for small values of p; see Figure 5. Already 73% of the maximal reduction is realized when 20% of the jobs is common and 91% of the maximal reduction is realized when 50% of the jobs is common. A surprising result is that the realized reduction $rr(p)$ is almost the same for $c = 2$ and $c = 3$ servers. Further note that for large p the mean waiting time $W^{(1)}$ for specialist jobs is only a little bit larger than the mean waiting time $W^{(c+1)}$ for common jobs. This is due to the balancing effect of the common jobs.

			$c=2$			$c=3$		
p	ρ	$W_{M/M/1}$	W	W_{SSQS}	$rr(p)$	W	W_{SSQS}	$rr(p)$
0.25	0.2	0.25	0.19	0.07	32.1 %	0.18	0.02	30.8 %
	0.4	0.67	0.51	0.26	39.6 %	0.46	0.13	38.6 %
	0.6	1.50	1.10	0.68	49.2 %	0.97	0.42	48.9 %
	0.8	4.00	2.67	1.96	64.8 %	2.24	1.29	64.8 %
	0.9	9.00	5.47	4.47	77.9 %	4.31	2.98	78.0 %
	0.95	19.00	10.69	9.49	87.3 %	7.94	6.33	87.4 %
	0.98	49.00	25.86	24.49	94.4 %	18.17	16.35	94.4 %
0.50	0.2	0.25	0.14	0.07	58.7 %	0.12	0.02	57.2 %
	0.4	0.67	0.40	0.26	66.4 %	0.32	0.13	65.5 %
	0.6	1.50	0.89	0.68	74.5 %	0.70	0.42	74.3 %
	0.8	4.00	2.27	1.96	84.7 %	1.70	1.29	84.8 %
	0.9	9.00	4.86	4.47	91.4 %	3.50	2.98	91.4 %
	0.95	19.00	9.93	9.49	95.4 %	6.92	6.33	95.4 %
	0.98	49.00	24.97	24.49	98.1 %	16.98	16.35	98.1 %

Table 2: *Mean waiting times as a function of p, ρ and c.*

The behavior of the overall mean waiting time W is further investigated in Table 2

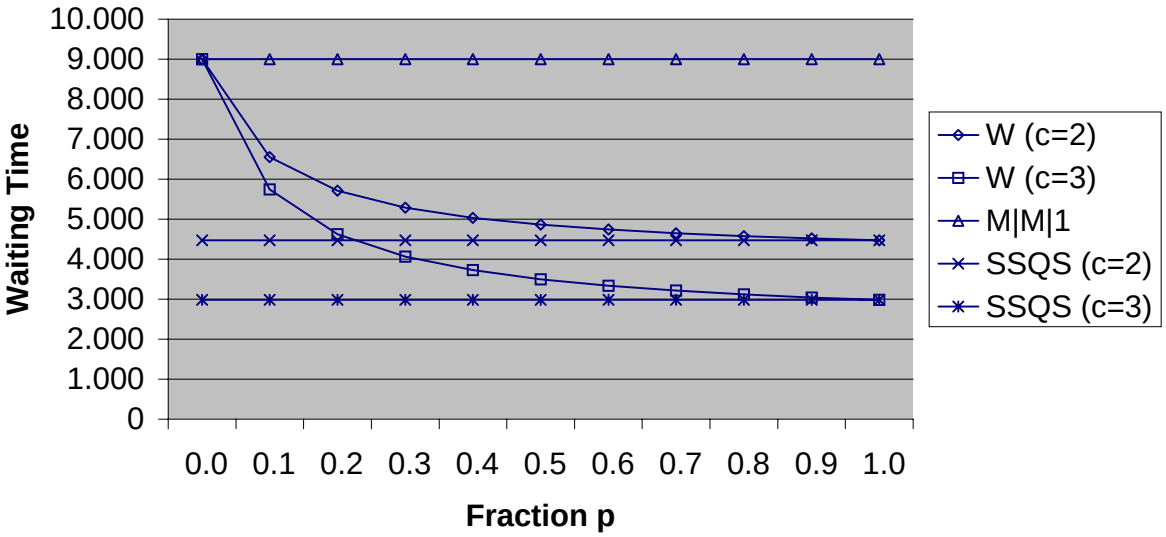

Figure 5: *Graphical representation of the mean waiting times W listed in Table 1.*

for different values of p, ρ and c. The mean waiting times are again determined with an absolute accuracy $\epsilon_{abs} = 0.005$ (and 0.0001 for W_{SSQS}). Only for low workloads (i.e., $\rho \leq 0.4$), the mean waiting time has been determined even more accurately in order to obtain sufficiently accurate estimates for $rr(p)$. The results in Table 2 show that for each combination of p and c the mean waiting time W is close to W_{SSQS} for all ρ. The results also suggest that the $rr(p)$ is insensitive to the number of servers c. However, $rr(p)$ strongly depends on ρ; it is rather small for low workloads and large for high workloads (it seems that $rr(p) \uparrow 1$ as $\rho \uparrow 1$). The small values for $rr(p)$ for low workloads are due to the fact that $W_{M/M/1}$ is also close to W_{SSQS} in these cases.

4.2 Common versus semi-common jobs

In Subsection 4.1 we distinguished two job types only, specialist and common jobs. For GSQSs more than two servers, one may also have jobs in between, i.e., jobs that can be served by two or more, but not all servers. In this subsection we investigate which job types lead to the largest reduction of W: common or semi-common jobs?

We consider a GSQS with $c = 3$ servers and a total arrival rate $\lambda = 3\rho$ with $\rho \in (0, 1)$. The following two cases are distinguished for the detailed arrival streams. For *case I*, we copy the situation in Subsection 4.1. In this case there are 4 job types. The type 4 jobs are common jobs; they arrive with intensity $\lambda_4 = p\lambda$ with $p \in [0, 0.5]$ (the reason why p may not exceed 0.5 follows below). Type j jobs, $j = 1, 2, 3$ are specialist jobs which only can be served by server j; they arrive with intensity $\lambda_j = (1 - p)\lambda/3$. So the mean number of servers capable of serving an arbitrary job is equal to $1 + 2p$. In *case II* we have 6 job types. The type j jobs, $j = 1, 2, 3$, are again specialist jobs which can only be served by server j. The type 4, 5 and 6 jobs are semi-common jobs; the type 4 jobs can be served by the servers 1 and 2, the type 5 jobs by 1 and 3, and the type 6 jobs by 2 and 3. To guarantee that the mean number of servers capable of serving an arbitrary job remains the same (i.e., equal to $1 + 2p$), the arrival intensity λ_j is set equal to $\lambda_j = 2p\lambda/3$ for

$j = 4, 5, 6$ and $\lambda_j = (1 - 2p)\lambda/3$ for $j = 1, 2, 3$ (to avoid negative intensities, p must be less than or equal to 0.5).

		W		Diff. (I−II)	
p	ρ	Case I	Case II	Abs.	Rel.
0.25	0.2	0.18	0.14	0.04	23.4 %
	0.4	0.46	0.38	0.08	18.1 %
	0.6	0.97	0.83	0.15	14.9 %
	0.8	2.24	1.97	0.27	11.9 %
	0.9	4.31	3.92	0.38	8.9 %
	0.95	7.94	7.46	0.48	6.0 %
	0.98	18.17	17.62	0.55	3.0 %
0.50	0.2	0.12	0.05	0.07	55.1 %
	0.4	0.32	0.22	0.10	32.5 %
	0.6	0.70	0.56	0.14	20.1 %
	0.8	1.70	1.51	0.19	11.2 %
	0.9	3.50	3.27	0.22	6.4 %
	0.95	6.92	6.67	0.25	3.6 %
	0.98	16.98	16.72	0.26	1.5 %

Table 3: *Mean waiting times as a function of p and ρ.*

Table 3 lists the overall mean waiting time W for different values of p and ρ. The results for case I are copied from Table 2. We can conclude that the absolute difference between the mean waiting time W in case I and II is rather small in each situation. This suggests that W is mainly determined by the mean number of servers capable of serving an arbitrary job; it does not matter whether this mean number is realized by common or by (twice as many) semi-common jobs. Nevertheless, the results in Table 3 also show that in each situation case II yields a smaller W than case I. This may be explained as follows. Let us consider the situation with $p = 0.5$. In case I, $\lambda_1 = \lambda_2 = \lambda_3 = \lambda/6$ and $\lambda_4 = \lambda/2$. Hence, for each group of 6 arriving jobs, on average 4 jobs join the shortest queue, 1 job joins the shortest but one queue, and 1 job joins the longest queue. In case II, however, $\lambda_1 = \lambda_2 = \lambda_3 = 0$ and $\lambda_4 = \lambda_5 = \lambda_6 = \lambda/3$. Thus for each group of 6 arriving jobs, on average 4 jobs join the shortest queue and 2 jobs joins the shortest but one queue. So in case II the balancing of queues will be slightly stronger, and thus W will be slightly smaller.

4.3 Balanced asymmetric systems

In this subsection we study the GSQS with $c = 2$ servers and three job types as depicted in Figure 2. The parameters are chosen as follows: $\rho = 0.9$, $\lambda = 2\rho = 1.8$, $\lambda_A = \lambda/2 = 0.9$, $\lambda_B = \hat{p}\lambda/2 = 0.9\hat{p}$, $\lambda_C = (1 - \hat{p})\lambda/2 = 0.9(1 - \hat{p})$ where $\hat{p} \in [0, 0.5]$. So one half of the jobs are common (type A) jobs and the other half are specialist (type B and C) jobs. But the specialist jobs are not equally divided over the servers. The fraction $\hat{p}$ of specialist jobs which must be served by server 1 (i.e., the type B jobs) is less than or equal

to the fraction $1 - \hat{p}$ of specialist jobs which must be served by server 2 (i.e. the type C jobs). Only for $\hat{p} = 0.5$ we have a symmetric system. For all $\hat{p} \in [0, 0.5)$ we have an asymmetric, but balanced system; a static system with equal workloads for both servers is obtained when a fraction $1 - \hat{p}$ of the type A jobs is sent to server 1 and a fraction $\hat{p}$ to server 2.

$\hat{p}$	$W^{(A)}$	$W^{(B)}$	$W^{(C)}$	W	$rr(\hat{p})$
0.0	4.28	4.34	13.05	8.66	7.5 %
0.1	4.37	4.52	8.52	6.25	60.8 %
0.2	4.42	4.68	6.93	5.45	78.5 %
0.3	4.44	4.84	6.12	5.09	86.5 %
0.4	4.45	5.03	5.62	4.92	90.3 %
0.5	4.45	5.28	5.28	4.86	91.4 %

Table 4: *Mean waiting times as a function of* $\hat{p}$.

Table 4 shows the mean waiting times $W^{(A)}$, $W^{(B)}$, $W^{(C)}$ for each job type and the overall mean waiting time W for $\hat{p} = 0, 0.1, \ldots, 0.5$. These waiting times have again been computed with an absolute accuracy $\epsilon_{abs} = 0.005$. In the last column of Table 4 we list the realized reduction $rr(\hat{p})$ defined by (10), where $W_{M/M/1} = 9$ and $W_{SSQS} = 4.475$ for $\rho = 0.9$. The results in Table 4 show that $W^{(A)}$ is fairly constant for all values of $\hat{p}$. As expected, $W^{(B)}$ decreases and $W^{(C)}$ increases as $\hat{p}$ decreases. A striking observation is that $W^{(C)}$ sharply increases for $\hat{p}$ close to 0; and thus also $W = (W^{(A)} + \hat{p}W^{(B)} + (1 - \hat{p})W^{(C)})/2$. For $\hat{p} = 0$ we have $\lambda_A = \lambda_C = 0.9$ and $\lambda_B = 0$, and the overall mean waiting time W is equal to 8.66. This is close to $W_{M/M/1} = 9$, which is realized when all type A jobs would be sent to server 1.

4.4 Conclusion

The main conclusion from the numerical experiments is that the overall mean waiting time may already be reduced significantly by creating a little bit of (semi-)common work. Furthermore, this reduction is mainly determined by the amount of overlap, i.e., the mean number of servers capable of handling an arbitrary job. Finally, the beneficial effect of (semi-)common jobs may vanish for highly asymmetric situations.

References

[1] ADAN, I.J.B.F., VAN HOUTUM, G.J., AND VAN DER WAL, J. Upper and lower bounds for the waiting time in the symmetric shortest queue system. *Annals of Operations Research 48* (1994), pp. 197–217.

[2] ADAN, I.J.B.F., WESSELS, J., AND ZIJM, W.H.M. Queueing analysis in a flexible assembly system with a job-dependent parallel structure. In *Operations Research Proceedings 1988*, Springer-Verlag, Berlin, 1989, pp. 551–558.

[3] ADAN, I.J.B.F., WESSELS, J., AND ZIJM, W.H.M. Analysis of the symmetric shortest queue problem. *Stochastic Models 6* (1990), pp. 691–713.

[4] AHUJA, R.K., MAGNANTI, T.L., AND ORLIN, J.B. *Network Flows: Theory, Algorithms, and Applications.* Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

[5] FOLEY, R.D., AND MCDONALD, D.R. Join the shortest queue: Stability and exact asymptotics. Research report, Georgia Institute of Technology, Atlanta, 1998.

[6] FOSS, S., AND CHERNOVA, N. On the stability of a partially accessible multi-station queue with state-dependent routing. *Queueing Systems 29* (1998), pp. 55–73.

[7] GREEN, L. A queueing system with general-use and limited-use servers. *Operations Research 33* (1985), pp. 168–182.

[8] HASSIN, R., AND HAVIV, M. Equilibrium strategies and the value of information in a two line queueing system with threshold jockeying. *Stochastic Models 10* (1994), pp. 415–435.

[9] LATOUCHE, G., AND RAMASWAMI, V. A logarithmic reduction algorithm for quasi-birth-death processes. *Journal of Applied Probability 30* (1993), pp. 650–674.

[10] NEUTS, M.F. *Matrix-Geometric Solutions in Stochastic Models.* Johns Hopkins University Press, Baltimore, 1981.

[11] ROQUE, D.R. A note on "Queueing models with lane selection". *Operations Research 28* (1980), pp. 419–420.

[12] SCHWARTZ, B.L. Queueing models with lane selection: a new class of problems. *Operations Research 22* (1974), pp. 331–339.

[13] SPARAGGIS, P.D., CASSANDRAS, C.G., AND TOWSLEY, D. Optimal control of multiclass parallel service systems with and without state information. In *Proceedings of the 32nd Conference on Decision and Control*, San Antonio, 1993, pp. 1686–1691.

[14] VAN HOUTUM, G.J. *New Approaches for Multi-Dimensional Queueing Systems.* Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, 1995.

[15] VAN HOUTUM, G.J., ZIJM, W.H.M., ADAN, I.J.B.F., AND WESSELS, J. Bounds for performance characteristics: A systematic approach via cost structures. *Stochastic Models 14* (1998), pp. 205–224. (Special issue in honor of M.F. Neuts.)

[16] ZIJM, W.H.M. Operational control of automated PCB assembly lines. In *Modern Production Concepts: Theory and Applications*, G. Fandel and G. Zaepfel, Eds. Springer-Verlag, Berlin, 1991, pp. 146–164.