PERFORMANCE ANALYSIS OF PARALLEL MACHINES WITH A GENERALIZED SHORTEST QUEUE ARRIVAL MECHANISM

G.J. van Houtum[1,*], I.J.B.F. Adan[2], J. Wessels[2], W.H.M. Zijm[1]

[1] *University of Twente, Faculty of Mechanical Engineering, P.O. Box 217, 7500 AE Enschede, The Netherlands.*

[2] *Eindhoven University of Technology, Dept. of Math. and Comp. Sci., P.O. Box 513, 5600 MB Eindhoven, The Netherlands.*

Abstract

We consider the production/assembly of Printed Circuit Boards on parallel insertion machines, which all have their own queue. The machines serve several types of jobs. Because of technical reasons, most or all types of arriving jobs can only be served by a restricted set of machines. All jobs require (roughly) the same mean service time, and, in order to minimize its own service time, each arriving job joins (one of) the shortest queue(s) of all queue(s) where the job can be served. Hence, the resulting queueing system is a generalized shortest queue system. In order to facilitate the analysis of the main characteristic of this system, viz. the generalized shortest queue routing, we assume that the jobs arrive according to Poisson processes and that the service times are exponential. Under these assumptions, the behavior of the resulting queueing system may be described by a multi-dimensional Markov process and the performance of the system can be efficiently obtained from flexible bound models. Several numerical results are presented to show the influence of the shortest queue routing on the the waiting times for each of the job types separately and for all job types together.

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

* *Corresponding author.* Mailing Address: see above. Phone: +31 53 4893192. Fax: +31 53 4893471. E-mail: g.j.j.a.n.vanhoutum@wb.utwente.nl .

1 Introduction

In this paper, we study a queueing situation stemming from a flexible assembly system consisting of a group of parallel insertion machines, which have to mount vertical components on Printed Circuit Boards. The description of this system is started with explaining how an insertion machine operates.

An *insertion machine* mounts vertical components, such as resistors and capacitators, on a Printed Circuit Board (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 may contain 80 tapes, say. Each PCB needs, on average, 60 different types of components. If a machine has to mount components on a PCB, then all the components need to be available on that machine. That means that for all those components a tape must be placed in the magazine. So the set of components available on the machine completely determines which types of PCBs can be handled.

In general we have a group of parallel insertion machines which have to process a number of different types of PCBs at the same time. Each insertion machine has its own queue, and the PCBs are transported to the insertion machines by an Automatic Conveyor System. In Figure 1, we have depicted a system which consists of three insertion machines and which has to process three different types of PCBs. The machines are basically similar, but due to the fact that they may be loaded with different types of components, the classes of PCB-types that can be handled by the machines may be different. In the situation depicted in Figure 1, machine M_1 can handle PCBs of the types A and B, machine M_2 can handle the types A and C, and machine M_3 can handle the types B and C.

In fact, there are two decision problems: the *assignment problem* and the *routing problem*. We first describe the assignment problem, which is the major problem. The assignment problem concerns how the tapes with components have to be divided among the machines. One should try to allocate the tapes with components to the machines such that, for example, the waiting times (and/or sojourn times) of the PCBs are minimized. There would be no problem if the magazines were big enough to contain all components needed to process all types of PCBs. However, in general they can only contain the components needed for a small subset of the different types of PCBs.

In order to solve the assignment problem, we must be able to evaluate the performance characteristics of a *given assignment* of the components to the machines. These performance characteristics depend on how the second decision problem, i.e. the routing problem, is handled. This problem concerns to which machines the PCBs must be sent upon arrival. For an arriving PCB, we must select one of the machines which can handle that PCB. If for all types the mounting times are roughly the same, then it is reasonable to select the machine with the *shortest queue* (let ties be broken with equal probabilities); this at least (roughly) minimizes the waiting time

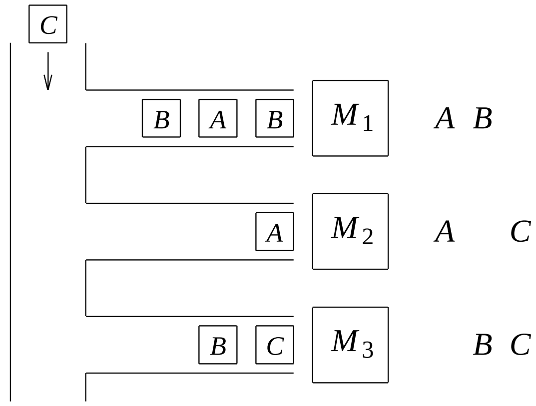

Figure 1: A flexible assembly system consisting of three parallel insertion machines, on which three types of PCBs are made.

of the arriving PCB itself, and it may be expected that this also roughly minimizes the average waiting time for all PCBs together, provided that we are in a balanced situation (i.e. a situation in which each server will have to handle the same amount of work on average). We assume that the shortest queue routing indeed is used by the Automatic Conveyer System, and that, once arrived in a queue, the PCBs are served in a First-Come-First-Served (FCFS) manner.

The assembly of PCBs is often characterized by relatively few job types, large production batches and small processing times (see Zijm [15]). Therefore, a queueing model approach seems natural. The flexible assembly system can be modeled as a queueing system consisting of parallel servers, each with a own queue, and serving several types of jobs, where each job upon arrival joins the shortest queue of all queues that can handle this job. We call this system a *Generalized Shortest Queue System (GSQS).*

In this paper, we shall focus on the main characteristic of the GSQS, which is the generalized shortest queue routing. We study the influence of this routing on the waiting times for each of the job types separately and for all job types together. The dynamic shortest queue routing obviously will lead to smaller waiting times than a static routing where each job chooses a queue according to given probabilities. However, the question is how much the waiting times are smaller under the shortest queue routing. Further, we are interested in how much the waiting times are affected by the fraction of jobs that can choose between all, or at least two, queues.

Since we focus on the shortest queue routing, we feel free to make the following simplifying assumptions for the arrivals and services: (i) all jobs arrive according to Poisson streams; (ii) the service times are exponentially distributed; (iii) the service times are job-independent; (iv) all insertion machines work equally fast. The assumptions (ii)-(iv) imply that all service times are exponentially distributed with the same parameter. Even under these assumptions, the GSQS constitutes a hard problem. The behavior of the GSQS is described by a continuous-time Markov process with multi-dimensional states where each component denotes the queue length at one of the servers. However, because of the shortest queue routing, the structure of the transitions is rather complicated and hence an analytical solution seems hard to obtain in general. In fact, an analytical solution is only known for the special case with two parallel servers and one type of jobs that can be handled by both servers; in this case the GSQS reduces to the two-dimensional symmetric shortest queue system, for which a generalized product-form solution has been derived by using a compensation approach (see [4]). For all other cases, even a standard numerical method is not available. Therefore, for the general case of the GSQS, we use truncation models which: (i) have a truncated state space with a flexible size (i.e. depending on one or more truncation parameters); (ii) can be solved efficiently; (iii) lead to upper/lower bounds for the waiting times. Such models are called *solvable flexible bound models*. We shall define one lower bound and one upper bound model. By solving these two models for increasing sizes of the truncated state space, we can determine the waiting times of the original GSQS as accurately as desired. In [13], it has been shown that this method works quite well. Compared to simulation, this method requires more programming work, but, once programmed, it requires relatively small computation times to compute the waiting times, even if we want them to be computed within a high accuracy and/or if we have high workloads for the different servers. It is noted that flexible bound models previously have been successfully applied to the symmetric shortest queue system (with ≥ 2 servers), the symmetric longest queue system and an $M|M|c$ system with critical jobs (see [2, 14, 1]).

Apart from the situation described above, the GSQS is also relevant for many other practical situations; for example, in a job shop with a group of identical, parallel machines which are loaded with different sets of tools, in a computer system where each information file is available on a restricted set of a number of parallel disks and requests for information files have to be handled by only one disk, and at a banking office where each clerk is able to carry out a restricted set of tasks. Nevertheless, the GSQS has hardly been studied in the literature. To our knowledge, the only contribution is made by Adan, Wessels and Zijm [3], who, for the same case as studied here, present rough approximations for the waiting times. Further, closely related systems have been studied by Schwartz [12] (see also Roque [11]), Green [6] and Hassin and Haviv [7].

This paper is organized as follows. In Section 2, we give a precise description of our model for the GSQS and we discuss conditions under which a GSQS is ergodic and balanced. Next, in Section 3, we describe the flexible bound models which we use to determine the waiting times for the GSQS. Finally, in Section 4, the influence of the shortest queue routing on the waiting times is shown on the basis of several numerical results.

2 Model

This section consists of three subsections. In the first subsection, we describe our model for the GSQS. In the Subsection 2.2, we discuss the ergodicity. We present a simple condition which easily is shown to be necessary for the ergodicity and which is

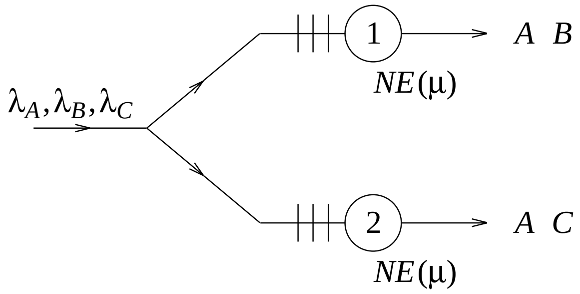

Figure 2: A GSQS with $c = 2$ servers and three job types.

conjectured to be also sufficient. Support for the latter part is obtained by comparing the GSQS to an identical system but with a static routing instead of the dynamic shortest queue routing. 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 Description of the model itself

The GSQS consists of $c \geq 2$ parallel servers, which all have their own queue, and there are several types of jobs which must be served by the GSQS. Because of technical reasons, for example, each server can only serve a restricted set of job types. It is assumed that all service times are exponentially distributed with the same parameter $\mu > 0$. Furthermore, it is assumed that for each job type, the jobs arrive according to a Poisson stream, and that each arriving job joins the shortest queue of all queues where the job can be served (ties are broken with equal probabilities). In Figure 2, we have depicted a GSQS with $c = 2$ servers and three types of jobs: jobs of type A, which arrive with intensity λ_A and which can be served by both servers, jobs of type B, which arrive with intensity λ_B and which can only be served by server 1, and jobs of type C, which arrive with intensity λ_C and which must be served by server 2.

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 J contains the job types. For each $j \in J$, we let $I(j)$ denote the set of servers which can serve the jobs of type j. Assume that each job type can be served by at least one server and that each server can handle at least one job type; so, $I(j) \neq \emptyset$ for all $j \in J$, and $\cup_{j \in J} I(j) = I$. Further, for each $j \in J$, the arrival intensity of the Poisson stream of the jobs of type j is given by $\lambda_j \geq 0$, and $\lambda := \sum_{j \in J} \lambda_j$ denotes the total arrival intensity.

Without loss of generality, we may assume that $\mu = 1$. Then the average workload per server is given by $\rho = \lambda / c$. It is obvious that we at least must satisfy the condition that $\rho < 1$ in order to have an ergodic system.

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

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

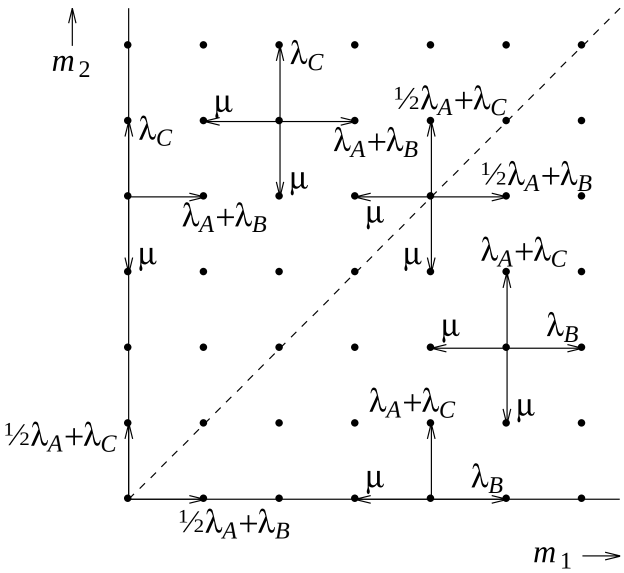

Figure 3: The transition rate diagram for the GSQS depicted in Figure 2.

In order to obtain an irreducible Markov process, 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 an indicator funtion, which is equal to 1 if G is true and equal to 0 otherwise), i.e. that all servers have a positive potential arrival rate. The transition rates are denoted by $q_{m,n}$. In Figure 3, we have depicted the transition rates for the GSQS of Figure 2.

The performance measures we are interested in are the mean waiting times $W^{(j)}$ for all job types $j \in J$, and the mean waiting time W for all job types together, which is equal to

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

It is obvious that for each $j \in J$,

$$W^{(j)} = L_{sq}^{(j)} , \tag{3}$$

where $L_{sq}^{(j)}$ denotes the mean of the shortest queue length of the queues at the servers $i \in I(j)$.

2.2 Discussion of the ergodicity

By studying the routing of the jobs, we obtain a simple, *necessary* condition for the ergodicity of the GSQS. For each subset $J' \in J$, $J \neq \emptyset$, jobs of the types $j \in J'$ 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 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 of Figure 2, condition (4) states that for the ergodicity at least the inequalities $\lambda_B < 1$, $\lambda_C < 1$ and $\lambda < 2$ (or $\rho < 1$) must be satisfied.

The condition stated in (4) is expected to be also sufficient for the ergodicity of the GSQS. This is argued by considering so-called corresponding static systems.

A *corresponding static system* is a system that is identical to the GSQS, but which uses a *static routing* instead of the dynamic shortest queue routing to route the arriving jobs to the servers. Such a 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 with which an arriving job of type j is sent to server i. Under a static routing, for each $j \in J$ the Poisson stream of arriving jobs of type j may be divided into Poisson streams with intensities $x_{j,i} = \lambda_j x_i^{(j)}$, $i \in I(j)$, for arrivals of jobs of type j which join server i, and hence the queues at the servers $i \in I$ constitute independent $M|M|1$ queues with identical mean service times equal to $\mu = 1$ and arrival intensities $\sum_{j \in J, (j,i) \in A} x_{j,i}$, where

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

As a result, we obtain a simple necessary and sufficient condition for the ergodicity of a corresponding static system, viz.

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

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:

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

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 the ergodicity. If is easily seen that a solution of (5) can not be found if condition (4) is not satisfied.

Now, assume that condition (4) is satisfied. To prove that then 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 $\hat{A} = A \cup \{(0,i) | i \in I\}$ (supply node 0 denotes an extra type of jobs, which may 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 $\hat{a}_0 = N - \lambda - N\epsilon$, 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 $\epsilon \leq (N - \lambda)/N$). 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; this condition is stated in Lemma 5.4 of [13] and its proof may be given by a transformation to a maximum-flow problem succeeded by the application of the max-flow min-cut theorem (see e.g. [5]). 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 jobs in the system the shortest queue routing will balance the queue lengths better (or at least equally good) than any static routing (more specifically, the shortest queue routing will better control the maximum queue length). So, if there is a corresponding static system which is ergodic, then the GSQS itself is expected to be also ergodic. Together with Lemma 2, this leads to the following conjecture.

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

2.3 Balanced and symmetric systems

From a practical point of view, for a GSQS it is desirable that the structure of the shortest queue routing, as given by the sets $I(j)$, is such that the workloads for the different servers are balanced. Formally, we say that a GSQS is *balanced* if there exists a corresponding static system for which the queues at all servers constitute $M|M|1$ queues with equal workloads. 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 $\lambda/N = \rho$, 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

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

These equations are 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 $b_i = \lambda/N$ for all $i \in V_2$. Applying the necessary and sufficient condition for the existence of such a feasible flow, as given in [13], leads to the following lemma.

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

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

Note that for $J' = \emptyset$ and $J' = J$, the condition stated in (7) is satisfied 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 not only necessary for the ergodicity, but it may be expected to be also sufficient.

If a GSQS is balanced, i.e. if there exists a static routing under which the workloads for the servers of the system are balanced, then this does not necessarily mean that these workloads are also balanced under the shortest queue routing. This can be seen by considering the GSQS depicted 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 $\lambda_C = \lambda_A + \lambda_B$. For 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, there always will occur situations in which jobs of type A are sent to server 2, and therefore server 2 will have a higher workload than server 1. Nevertheless, it is expected that for a balanced GSQS, the shortest queue routing at least ensures that the workloads for all servers will not differ too much.

A subclass of balanced systems is constituted by 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 of Figure 2 is symmetric if $\lambda_B = \lambda_C$.

For a symmetric GSQS, all queue lengths are equally distributed which implies that all servers have equal workloads. For such a system it may be shown that the shortest queue routing used by the different types of jobs minimizes the total number of jobs in the system and hence also the mean waiting time W for all jobs together (this may be done by the technique used in [8]). In particular, this means that a symmetric GSQS has a smaller mean waiting time for all job types together than the corresponding system consisting of N independent $M|M|1$ queues with workload ρ.

3 Solution by flexible bound models

In this section, we define two truncation models: one leading to lower bounds for the waiting times $W^{(A)}$, $W^{(B)}$, $W^{(C)}$ and W, and another one leading to upper bounds.

Since the shortest queue routing in general will cause a drift to the states with equal queue lengths, for both the lower and the upper bound model the truncated state space is defined by

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

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. The definition of M' shows that a state $m \in M$ is contained in the truncated state space M' if and only if for each $i \in I$ the length m_i of the queue at server i is at most T_i larger than the length of any other queue. For this choice of the truncated state space, there are two types of transitions pointing from states inside M' to states outside M':

(i) for a 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 each of the servers $k \in I$ 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) for a state $m = (m_1, \ldots, m_c) \in M'$ with $I' = \{i \in I | m_i = \min(m) + T_i\} \neq \emptyset$, at each of the servers $i \in I'$ 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 bound model, these transitions are redirected from the states n to states n' which correspond to situations with a smaller number of jobs at at least one of c servers. With respect to waiting times and queue lengths these states are more attractive. In the upper bound model, redirections are made to less attractive states corresponding to situations with a larger number of jobs at at least one of the c servers.

In the lower bound model, the transition described under (i) is redirected to the state $n' = m - e_k - \sum_{i \in I'} e_i \in M'$. The physical interpretation of this redirection is that a departure of a job at a non-empty shortest queue is accompanied by a destruction or killing of one job at each of the queues $i \in I'$, for which the difference with respect to the shortest queues has already reached its maximum value T_i. Further, the transition described under (ii) is redirected to the states $n' = m \in M'$ itself. The physical interpretation of this redirection is that a new job arriving at one of the servers $i \in I'$ is rejected. Because of the physical interpretations, the lower bound model is called the *Threshold Killing and Rejection (TKR) model.*

In the upper bound model, the transition described under (i) is redirected to $n' = m \in M'$ itself. The meaning behind this redirection is that if for at least

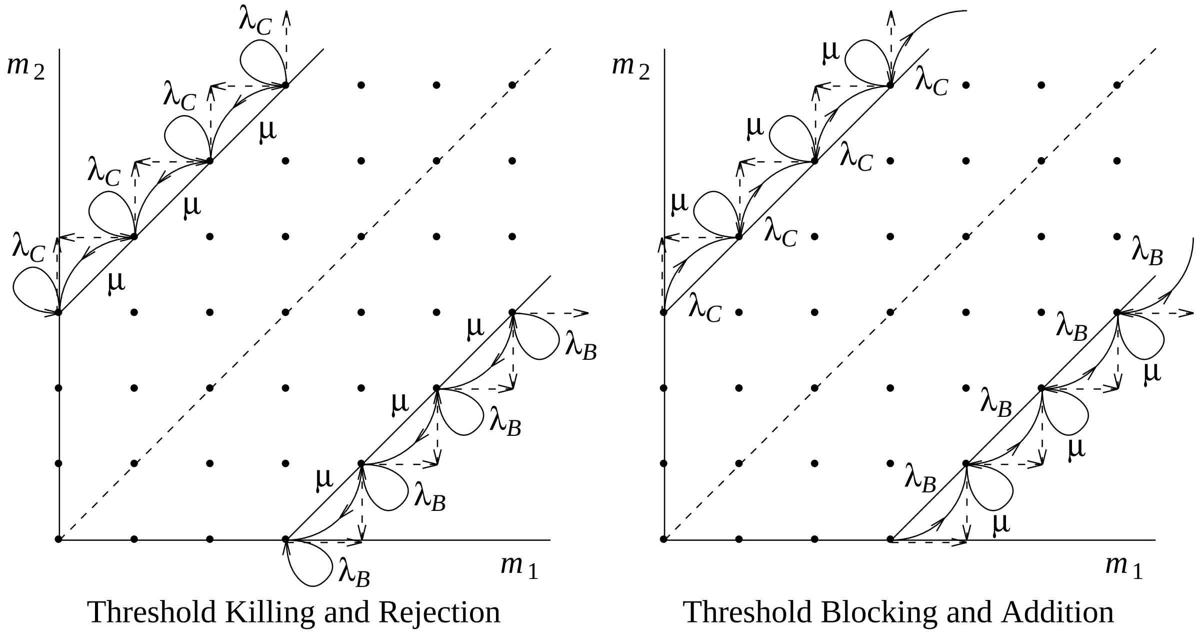

Figure 4: The redirections for the TKR and TBA model. For both models, $\hat{T} = (T_1, T_2) = (3, 3)$.

one queue the difference with respect to the shortest queue has already reached its maximum value, then a service completion at a non-empty shortest queue is not accompanied by a departure, and the job in service has to be served once more; this is equivalent to saying that then the servers at the shortest queues are blocked. Further, the transition described under (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 for which the difference with respect to the shortest queues has already reached its maximum value, is accompanied by the addition of one extra job at each of the shortest queues. The upper bound model is called the *Threshold Blocking and Addition (TBA) model.*

In Figure 4, we have depicted the redirections for the lower and upper bound model which are obtained for the GSQS of Figure 2.

In comparison with the original model, the TKR model leads to stochastically smaller lengths for the shortest queue among the queues $i \in I'$, where I' may be any non-empty subset of I. Hence the TKR model also leads to smaller means $L_{sq}^{(j)}$ for the shortest queue length at the servers $i \in I(j)$, $j \in J$, by which jobs of type j can be served. Further, it may be shown that the larger the values of the threshold parameters $T_1, \ldots, T_c$ the smaller the difference between the shortest queue lengths in the TKR model and the shortest queue lengths in the original model. The lower bounds for the mean shortest queue lengths immediately lead to lower bounds $W_{TKR}^{(j)}(\hat{T})$, $j \in J$, for the waiting times of each of the job types separately and to a lower bound $W_{TKR}(\hat{T})$ for the mean waiting time of all job types together (cf. (3) and (2)). Similarly, the TBA model leads to larger queue lengths and waiting times. The upper bounds for the mean waiting times are denoted by $W_{TBA}^{(j)}(\hat{T})$, $j \in J$, and

$W_{TBA}(\hat{T})$. A formal proof of all these monotonicity results may be given by using the *precedence relation method*. This method is based on Markov reward theory and has been developed in [13].

For both the TKR and TBA model, the steady-state distribution can be determined by the *matrix-geometric approach*, as described in [10]. This enables an efficient computation of the corresponding lower and upper bounds for the waiting times; see [13] for appropriate matrix formulae that can be used for this computation.

4 Numerical results

In this final section, we present numerical results for three series of instances of the GSQS depicted in Figure 2. The main objective is to show the influence of the shortest queue routing on the waiting times.

The instance with

$$\rho = 0.9, \quad \lambda = 2\rho, \quad \lambda_A = p\lambda \text{ with } p = \frac{1}{2}, \quad \lambda_B = \lambda_C = \frac{1}{2}(1-p)\lambda$$

is chosen as a basic instance. In the first series, we vary the value of the workload ρ. In the second series, we vary the value of the fraction p of jobs that can be handled by both servers. Both these series concern symmetric instances. In the third series, we vary the values of λ_B and λ_C, while keeping the sum of these two rates equal to $\lambda_B + \lambda_C = \frac{1}{2}\lambda$. We take $\lambda_B = \frac{1}{2}\hat{p}\lambda$ and $\lambda_C = \frac{1}{2}(1-\hat{p})\lambda$, where $\hat{p}$ is varied from 0.0 up to 0.5; cases with $\hat{p} > 0.5$ are not considered, since they lead to the same results as the with $\hat{p} < 0.5$, but with the roles of the types B and C interchanged.

For each instance, the waiting times $W^{(A)}$, $W^{(B)}$, $W^{(C)}$ and W are determined by solving the TKR and TBA model for vectors $\hat{T} = (T_1, T_2)$ with increasing values of T_1 and T_2. For each vector $\hat{T}$, the mean $(W^{(j)}_{TKR}(\hat{T}) + W^{(j)}_{TBA}(\hat{T}))/2$ and half of the difference, i.e. $\Delta^{(j)}(\hat{T}) = (W^{(j)}_{TBA}(\hat{T}) - W^{(j)}_{TKR}(\hat{T}))/2$, are used as an approximation for $W^{(j)}$ and an upper bound for the corresponding absolute inaccuracy, where $j \in J$, and similarly $(W_{TKR}(\hat{T}) + W_{TBA}(\hat{T}))/2$ and $\Delta(\hat{T}) = (W_{TBA}(\hat{T}) - W_{TKR}(\hat{T}))/2$ are used as an approximation and an upper bound for the corresponding absolute inaccuracy for W. Here, all means and upper bounds $\Delta^{(j)}(\hat{T})$ and $\Delta(\hat{T})$ are taken equal to ∞, if the TBA model is not ergodic. If for some $\hat{T}$ each of the upper bounds $\Delta^{(j)}(\hat{T})$ and $\Delta(\hat{T})$ is smaller than or equal to ϵ_{abs}, then the computation process is stopped, otherwise the procedure continues with the computation of improved bounds and approximations for a next threshold vector $\hat{T}$ with increased values for at least one of the threshold parameters T_1 and T_2. In the latter case, the decision on how to increase the threshold parameters is based on the rates of redirections $r_{rd}(1)$ and $r_{rd}(2)$; this decision is described in the next paragraph.

The variable $r_{rd}(1)$ denotes the rate with which redirections occur on the boundary $m_1 = m_2 + T_1$ of the truncated state space. If for the present threshold vector $\hat{T}$ only the TKR model is ergodic, then we let $r_{rd}(1)$ denote the rate for this TKR

ρ	$\hat{T}$	GSQS				Related Systems		
		$W^{(A)}$	$W^{(B)}$	$W^{(C)}$	W	$W_{M\|M\|1}$	W_{SSQS}	$W_{M\|M\|2}$
0.1	$(2,2)$	0.01	0.11	0.11	0.06	0.11	0.02	0.01
0.2	$(3,3)$	0.06	0.23	0.23	0.14	0.25	0.07	0.04
0.3	$(3,3)$	0.13	0.37	0.37	0.25	0.43	0.14	0.10
0.4	$(4,4)$	0.24	0.56	0.56	0.40	0.67	0.26	0.19
0.5	$(5,5)$	0.40	0.80	0.80	0.60	1.00	0.43	0.33
0.6	$(7,7)$	0.65	1.13	1.13	0.89	1.50	0.68	0.56
0.7	$(8,8)$	1.07	1.65	1.65	1.36	2.33	1.11	0.96
0.8	$(11,11)$	1.92	2.61	2.61	2.27	4.00	1.96	1.78
0.9	$(15,15)$	4.45	5.28	5.28	4.86	9.00	4.48	4.26
0.95	$(18,18)$	9.47	10.38	10.38	9.93	19.00	9.49	9.26
0.98	$(23,23)$	24.49	25.45	25.45	24.97	49.00	24.50	24.25
0.99	$(26,26)$	49.49	50.47	50.47	49.98	99.00	49.50	49.25

Table 1: The mean waiting times for increasing values of ρ. For the GSQS, $\lambda = 2\rho$, $\lambda_A = \frac{1}{2}\lambda$, $\lambda_B = \lambda_C = \frac{1}{4}\lambda$, and the mean waiting times $W^{(\cdot)}$ and W have been determined within an absolute accuracy of $\epsilon_{abs} = 0.005$.

model, else we let $r_{rd}(1)$ denote the mean of the rate found for the TKR model and the rate found for the TBA model. Similarly, $r_{rd}(2)$ denotes the rate with which redirections occur on the boundary $m_2 = m_1 + T_2$ of the truncated state space. Both fractions can be computed in a similar way as the bounds for the mean waiting times. The higher the values of $r_{rd}(1)$ and $r_{rd}(2)$, the more important it is to increase the values of T_1 and T_2, respectively. If it is established that for the present threshold vector $\hat{T}$ the desired accuracy for the approximations for the mean normalized waiting times $W^{(j)}$ and W is not reached yet, then the value of T_1 is increased by 1 if $r_{rd}(1) > r_{rd}(2)$, the value of T_2 is increased by 1 if $r_{rd}(2) > r_{rd}(1)$, and both T_1 and T_2 are increased by 1 unit if $r_{rd}(1) = r_{rd}(2)$. For symmetric instances, the rates $r_{rd}(1)$ and $r_{rd}(2)$ will always be equal, and hence the vector $\hat{T}$ will assume the values $\hat{T} = (1,1), (2,2), \ldots$.

For each of the instances, the waiting times have been determined with absolute accuracy $\epsilon_{abs} = 0.005$. The numerical results have been gathered in the Tables 1-3. The first column of these tables denotes the chosen values for ρ, p and $\hat{p}$, respectively. Further, in each of these tables, the second column depicts the vector $\hat{T}$ for which the desired absolute accuracy was reached. Next, the waiting times $W^{(A)}$, $W^{(B)}$, $W^{C)}$ and W are listed. In the last three columns, we have depicted the waiting times which under the same workload are obtained for the $M|M|1$ queue, the pure symmetric shortest queue system and the $M|M|2$ queue, respectively. These waiting times are denoted by $W_{M|M|1}$, W_{SSQS} and $W_{M|M|2}$. Note that $W_{M|M|1} = \rho/(1-\rho)$, $W_{M|M|2} = \rho^2/(1-\rho^2)$ and W_{SSQS} can be determined by the compensation approach as described in [4]. Since for all instances, the GSQS is balanced, $W_{M|M|1}$ is equal to the smallest waiting time that can be obtained by a static routing. In general, W is smaller than $W_{M|M|1}$ and the difference $W_{M|M|1} - W$ may be used as a measure for

		GSQS				Related Systems		
p	$\hat{T}$	$W^{(A)}$	$W^{(B)}$	$W^{(C)}$	W	$W_{M\|M\|1}$	W_{SSQS}	$W_{M\|M\|2}$
0.0	$(85, 85)$	4.26	9.00	9.00	9.00	9.00	4.48	4.26
0.1	$(43, 43)$	4.36	6.80	6.80	6.56	9.00	4.48	4.26
0.2	$(29, 29)$	4.40	6.04	6.04	5.72	9.00	4.48	4.26
0.3	$(22, 22)$	4.43	5.66	5.66	5.29	9.00	4.48	4.26
0.4	$(18, 18)$	4.44	5.43	5.43	5.04	9.00	4.48	4.26
0.5	$(15, 15)$	4.45	5.28	5.28	4.86	9.00	4.48	4.26
0.6	$(13, 13)$	4.46	5.17	5.17	4.74	9.00	4.48	4.26
0.7	$(11, 11)$	4.46	5.09	5.09	4.65	9.00	4.48	4.26
0.8	$(10, 10)$	4.47	5.02	5.02	4.58	9.00	4.48	4.26
0.9	$(9, 9)$	4.47	4.97	4.97	4.52	9.00	4.48	4.26
1.0	$(8, 8)$	4.48	4.93	4.93	4.48	9.00	4.48	4.26

Table 2: The mean waiting times for a series of instances with of $\rho = 0.9$. For the GSQS, $\lambda = 2\rho$, $\lambda_A = p\lambda$, $\lambda_B = \lambda_C = \frac{1}{2}(1-p)\lambda$, and the value of p has been varied. The mean waiting times $W^{(\cdot)}$ and W have been determined within an absolute accuracy of $\epsilon_{abs} = 0.005$.

the positive impact of the shortest queue routing on the waiting time. In the pure shortest queue system, all jobs can be served by all servers and hence in general W_{SSQS} constitutes a lower bound for W. Finally, $W_{M|M|2}$ denotes the waiting time if all jobs could be served by all servers and if there would be one central queue instead of local queues for each of the servers separately. In general, also $W_{M|M|2}$ constitutes a lower bound for W; in fact, $W_{M|M|2}$ appears to constitute a tight lower bound for W_{SSQS} (see [2]).

The results in Table 1 show that only a small difference between the waiting times for the types B and C and the waiting time for type A is obtained, even for high workloads. Further, the mean waiting time W for all jobs together is only a little bit larger than the mean waiting time in W_{SSQS} in a pure shortest queue system. At the same time the difference with $W_{M|M|1}$ is very large. This means that the shortest queue routing, for only half of the jobs, has a very strong decreasing impact on the waiting time. From the results in Table 2, it follows that the mean waiting time W for all job types together is more than proportionally decreasing as a function of the fraction p of jobs which can be served by both servers. In fact, a small fraction p of jobs that can be handled by both servers, already leads to a considerable reduction for W, compared to the situation with $p = 0$. The results in Table 3 show that the mean waiting time W strongly depends on $\hat{p}$. In this table, small values of $\hat{p}$ correspond to situations in which, under the shortest queue routing, the workloads for the servers 1 and 2 are badly balanced; for small $\hat{p}$, there will be several jobs of type A which join server 2 while they better could join server 1 in order to repair the asymmetry in the arrival mechanism. Due to this behavior, for server 2 a significantly higher workload is obtained than for server 1 and the mean normalized waiting time W becomes relatively large.

$\hat{p}$	$\hat{T}$	GSQS $W^{(A)}$	$W^{(B)}$	$W^{(C)}$	W	Related Systems $W_{M\|M\|1}$	W_{SSQS}	$W_{M\|M\|2}$
0.0	(7, 90)	4.28	4.34	13.05	8.66	9.00	4.48	4.26
0.1	(8, 44)	4.37	4.52	8.52	6.25	9.00	4.48	4.26
0.2	(9, 30)	4.42	4.68	6.93	5.45	9.00	4.48	4.26
0.3	(11, 22)	4.44	4.84	6.12	5.09	9.00	4.48	4.26
0.4	(12, 18)	4.45	5.03	5.62	4.92	9.00	4.48	4.26
0.5	(15, 15)	4.45	5.28	5.28	4.86	9.00	4.48	4.26

Table 3: The mean waiting times for a series of instances with of $\rho = 0.9$. For the GSQS, $\lambda = 2\rho$, $\lambda_A = \frac{1}{2}\lambda$, $\lambda_B = \frac{1}{2}\hat{p}\lambda$, $\lambda_C = \frac{1}{2}(1-\hat{p})\lambda$ and the value of $\hat{p}$ has been varied. The mean waiting times $W^{(\cdot)}$ and W have been determined within an absolute accuracy of $\epsilon_{abs} = 0.005$.

The following important conclusion can be drawn for the production of Printed Circuit Boards by the flexible assembly system, as described in Section 1: *In order to obtain small mean waiting times for the given total workload, the assignment of the components to the insertion machines should be such that for the resulting GSQS a strong drift to the states with equal queue lengths is obtained.* Note that these assignments are precisely the ones for which our bound models work well. This follows from the values depicted for $\hat{T}$ in the Tables 1-3. Hence, after having selected several assignments which are expected to have the strong drift to the states with equal queue lengths, the bound models can be well used to compute the performance for each of the assignments, and subsequently to select a restricted set of assignments leading to the lowest waiting times. We recall that these assignments represent the best assignments for the simplified model of the GSQS as studied in this paper. For a more general model, simulation could be used to determine the real quality for each of the these assignments.

References

[1] Adan, I.J.B.F., and Hooghiemstra, G., The $M|M|c$ with critical jobs. Memorandum COSOR 96-20, Eindhoven University of Technology, Dept. of Math. and Comp. Sci., 1996.

[2] 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), 197–217.

[3] 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.

[4] Adan, I.J.B.F., Wessels, J., and Zijm, W.H.M. Analysis of the symmetric shortest queue problem. *Stochastic Models 6* (1990), 691–713.

[5] Ahuja, R.K., Magnanti, T.L., and Orlin, J.B. *Network Flows: Theory, Algorithms, and Applications.* Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

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

[7] 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), 415–435.

[8] Hordijk, A., and Koole, G. On the assignment of customers to parallel queues. *Probability in the Engineering and Informational Sciences 6* (1992), 495–511.

[9] Latouche, G., and Ramaswami, V. A logarithmic reduction algorithm for quasi-birth-death processes. *Journal of Applied Probability 30* (1993), 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), 419–420.

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

[13] Van Houtum, G.J. *New Approaches for Multi-Dimensional Queueing Systems.* Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, 1995.

[14] Van Houtum, G.J., Adan, I.J.B.F., and Van der Wal, J. The symmetric longest queue system. *Stochastic Models 13* (1997), 105–120.

[15] 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.