Analysis of a fork/join station with inputs from a finite population. Sub-network with multi-server stations

Nico Goossens, Ananth Krishnamurthy and Nico Vandaele

DEPARTMENT OF DECISION SCIENCES AND INFORMATION MANAGEMENT (KBI)

Analysis of a Fork/Join Station with Inputs from a Finite Population Sub-network with Multi-server Stations

Nico Goossens

OM Partners n.v.,

Koralenhoeve 23 2160 Wommelgem Antwerpen, Belgium.

Email: ngoossens@ompartners.com

Fax: 32-3-650.22.90

Ananth Krishnamurthy¹

Department of Decision Sciences and Engineering Systems,

Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY-12180, USA.

Email: krisha@rpi.edu

Phone: 1-518-276-2958, *Fax:* 1-518-276-8227

Nico Vandaele²

Faculty of Applied Economics

Katholieke Universiteit Leuven, Campus Kortrijk, Belgium.

Email: Nico.Vandaele@kuleuven-kortrijk.be

Abstract

This paper presents an exact analysis of a fork/join station with inputs from stations composed of multiple exponential servers. The queue length process at the input buffers is analyzed exactly in terms of the underlying Markov process. The semi-Markov kernel characterizing the departure process is analyzed to derive expressions for the marginal and joint distributions of inter-departure times from the fork/join station. These analyses are used to study the effect of inputs from multiple servers on key performance measures at a fork/join station. Comparison studies show that replacing the multiple servers by a single server with equivalent capacity could result in significantly different estimates of throughput, synchronization delays and queue lengths at the individual buffers. These insights could lead to better representation of synchronization constraints in closed queuing network models of computer networks, fabrication/assembly systems and material control strategies for manufacturing systems.

Keywords: fork/join, synchronization, assembly systems, closed queuing networks

¹Corresponding author: Ananth Krishnamurthy (E-mail: krisha@rpi.edu, Phone: 1-518-276-2958)

²Formerly at the University of Antwerp, Department MTT.

1 Introduction

Fork/join stations model synchronization constraints in queuing systems. A fork/join station consists of two or more input buffers and a station with zero service times. Entities arrive at each input buffer according to a random process, and arriving entities are synchronized with entities in other buffers prior to departure from the station. In queuing models of fabrication/assembly systems, fork/join stations model synchronization constraints prior to assembly operations (Harrison [8], Hopp and Simon [9], Rao and Suri [21], and de Boeck [5], [7]). In computer systems analysis, queuing networks with fork/join stations have been studied in the context of parallel processing, database concurrency control, and communication protocols (Baccelli et al [2], Varki [25], Prabhakar et al. [19]).

To develop efficient methods for analyzing these networks, several researchers have analyzed different aspects of the performance of fork/join stations in isolation. The typical inputs for such an analysis are the capacity of each input buffer and a description of the inter-arrival times of the entities to each input buffer. Performance measures of interest include synchronization delays, queue length distributions at the different input buffers, and station throughput. However, a majority of the analysis of fork/join stations assume that the inputs to the individual buffers are simple Poisson processes that are independent of the contents of the buffers at the fork/join stations. While this assumption of exogenous arrivals may be reasonable when the fork/join station is part of an open queuing network, it may not be realistic when the fork/join station is part of a closed network.

When the fork/join station is part of a closed queuing network model of fabrication/assembly system, or a system operating pull control strategies (Di Mascolo et al [6], Krishnamurthy and Suri [12]), then once the queue length of an input buffer equals the size of the population from which arrivals occur to the buffer, the arrival process shuts down temporarily. Further, even when the arrival process is not shut down, the arrival process to a fork/join station in a closed queueing network may not be a simple Poisson process. The arrival process could be significantly influenced by the characteristics of the sub-networks (such as population limit, and service characteristics of the different servers) from which arrivals occur. The analysis in this paper demonstrates that these characteristics of the arrival process must be adequately captured in order to obtain reliable estimates of the performance measures.

This paper considers a fork/join station with two input buffers, each receiving inputs from finite population sub-networks composed of multi-server stations. In each sub-network, it is assumed that the individual servers at each station are identical and have exponentially distributed service times. The queue length process at the input buffers is analyzed exactly in terms of the underlying Markov process. Next, it is shown that under the assumptions made, the departure process forms a Markov renewal process. Exact analysis of the underlying semi-Markov kernel yields expressions for the marginal and joint distributions of inter-departure times from the fork/join station. Using

the results from this analysis, the effect of inputs from finite population sub-networks with multiple servers on key performance measures is studied. Comparison studies show that replacing the multiple servers by a single server with equivalent capacity could result in significantly different estimates of throughput, synchronization delays and queue lengths at the individual buffers.

The remainder of this paper is organized as follows. Section 2 provides a summary of the literature to date on the analysis of fork/join stations. Section 3 defines the model of the fork/join station and Section 4 describes the analysis approach. Section 5 describes the analysis of the queue length process, and Section 6 describes the analysis of the departure process from the fork/join station. Section 7 compares performance of different fork/join stations with inputs from multiple servers. Section 8 presents the conclusions.

2 Literature Review

Fork/join stations with two or more inputs have been studied extensively in the past. Early work by Harrison [8] and Latouche [16] consider fork/join stations with exponential inputs and derive conditions for stability. Much of the subsequent research has focused on detailed analysis for performance measures of fork/join stations with two inputs. When the inter-arrival time distributions are exponential for each input, Markov chain models have been used to analyze queue length and departure processes at the fork/join station (Som et al. [22] and Takahashi et al. [23]). However, in many applications the inter-arrival times have a distribution different from the exponential. Recent studies by Takahashi et al. [24] use matrix analytical methods to analyze fork/join stations where the distribution of inter-arrival times are following a phase type distribution.

Exact analysis of fork/join stations with more than two inputs can be prohibitive due to the phenomenon of state space explosion. Therefore, several approximations have been proposed. Baccelli and Makowski [1], Nelson and Tantawi [18], Kumar and Shorey [15] identify properties and bounds of fork/join stations with multiple inputs. Knessl [10] and Varma and Markowski [26] approximate the queue length distribution under heavy traffic limits using diffusion approximations. Prabhakar et al.[19] study the departure process of fork/join stations with exponential inputs under limiting conditions. Bonomi[4], Liu and Perros[17] analyze fork/join stations with multiple Poisson inputs. However, a majority of the above assume that the arrivals are from infinite populations, i.e. the arrivals are an uninterrupted process and an arriving entity that finds the input buffer full is lost.

When the fork/join station is a part of a closed queuing network, then once the content in the input buffer reaches a certain level, the arrival process shuts down temporarily. Krishnamurthy et al. [14] evaluate performance measures of fork/join stations with 2 inputs, assuming that the inter-arrival time distributions are phase type. Subsequent analysis by Krishnamurthy et al. [13]

proposes two-moment approximations for fork/join stations with two renewal inputs. Varki [25] assumes a finite customer population in the fork/join queuing network but restricts all stations to have single servers with exponentially distributed service times. None, of these above works consider fork/join stations where the arrivals to the buffers are from finite population sub-networks with multi-server stations.

More recently, Ko and Serfozo [11] develop bounds and approximate expressions for evaluating the mean response time and queue length distribution at fork/join stations. Their analysis considers a system where the inputs are from multi-server stations with exponential service times. However, it assumes that the arrivals are from infinite populations. The work closest to the analysis presented here corresponds to that by Baynat and Dallery [3] and Di Mascolo et al [6]. They analyze a fork/join station with load dependent inputs and subsequently use their analysis to solve queuing models of kanban systems. While the characterization of the arrival process can be described as a generalization of the process assumed in this analysis, their work only presents an analysis of the queue length process. This paper analyzes the queue length process as well as provides a precise characterization of the departure process from the fork/join station, and provides exact expressions for marginal and joint distribution of inter-departure times from the fork/join station.

3 System Description

Figure 1 represents a fork/join station, J with inputs from multi-server stations, station 1 and station 2 respectively. Station, J has two input buffers B_1 and B_2 . If an entity arriving in buffer B_1 (B_2) finds input buffer B_2 (B_1) empty, it waits for the corresponding entity to arrive in input buffer B_2 (B_1). As soon as there is at least one entity in each buffer, one entity is removed from each buffer. The removed entities join and depart immediately from the fork/join station. As a result the content of each input buffer is reduced by one. Subsequent to departure from the fork/join station, the joined entity forks back into two entities that are routed back to station i , $i = 1, 2$ respectively, where it waits in queue for service if necessary. Station i consists of c_i parallel servers, and the service time at each server in station i has an exponential distribution with mean μ_i^{-1} . Upon completion of service at station i the entity revisits the fork/join station and waits in buffer B_i . There is a finite population of size K_i for the entity of type i , and it is assumed that $K_i \geq c_i, i = 1, 2$. Consequently, the number of entities in input buffer B_i and at the corresponding servers at station i always sum up to K_i , $i = 1, 2$, and the arrival process to buffer B_i shuts down when there are K_i units in buffer B_i .

Fork/join stations with such characteristics are found in queuing network models of closed multi-level fabrication/assembly systems, multi-stage kanban systems, and tandem lines with multi-server stations and finite buffers.

Figure 1: Fork/join Station with inputs from multiple servers

- The fork/join station described above can represent a synchronization station before an assembly operation in a fabrication/assembly system [20, 21]. In this case K_i could correspond to the fixed number of automated guided vehicles (AGVs) transporting components of type i from the fabrication sub-network (represented by the multi-server stations) to the assembly station. Entities in buffers B_1 and B_2 correspond to the fabricated parts that are to be assembled. The join operation corresponds to the kitting operation, while the fork operation corresponds to the release of free AGVs to carry the parts required for assembly.
- As a second example, the fork/join station model could represent the synchronization constraint in a kanban control system [6]. Here the fork/join station could model the synchronization constraint between an upstream stage (represented by station 1 and the downstream stage represented by station 2, in a multi-stage kanban system. Each entity in buffer B_1 would correspond to a part with an upstream kanban attached to it, while each entity in buffer B_2 would correspond to a free kanban returning from the downstream stage, and K_1 and K_2 would be the number of kanbans in the respective stages. During the join operation a part and upstream kanban are joined with a downstream kanban and during the fork operation, the upstream kanban is sent back, while the part and downstream kanban are sent to the next manufacturing stage.
- As a special case, the fork/join station model could also model the blocking phenomenon between two consecutive stations in in multi-server tandem lines with no buffers. The upstream station, station 1 is assumed to have c_1 servers and a buffer capacity of $K_1 = c_1$, while the downstream station, station 2 is assumed to have c_2 servers and a buffer capacity of $K_2 = c_2$.

Each entity in buffer B_1 would correspond to a server that is blocked after service, while each entity in buffer B_2 would correspond to an idle server in station 2. Whenever, there are c_1 entities in buffer B_1 station 1 is idle and when there are c_2 entities in buffer B_2 station 2 is idle. Clearly, one cannot have blocked servers in station 1 (i.e. entities in buffer B_1) when there are idle servers in station 2 (i.e entities in buffer B_2) [7].

4 Overview of the Analysis

The system description in Section 3 above implies that the inputs to the analysis is the two tuple $[(\mu_1, K_1, c_1), (\mu_2, K_2, c_2)]$. The objective of the analysis in this paper is to characterize the queue length process and the departure process at the fork/join station. Correspondingly, the analysis is composed of two parts:

- *Analysis of the queue length process:* To analyze the queue length process, the state space for the queue length process is defined and subsequently, the queue length process is analyzed as a continuous time Markov process. The underlying Markov chain is solved to obtain the steady state probability distributions. From these probability distributions, performance measures are estimated, namely, the throughput λ_D and mean queue lengths $E(L_1)$ and $E(L_2)$ at buffers B_1 and B_2 respectively. The detailed analysis is given in Section 5.
- *Analysis of the departure process:* First, the output process is recognized to be a Markov renewal process. In addition, it is noted that the Markov process embedded at departure instants has a special structure. This enables the determination of the transition probability matrix and stationary probabilities of the Markov chain embedded at departure instants. Using the stationary probabilities, the marginal distribution of inter-departure times from the fork/join station is obtained. Finally, the joint distribution of successive inter-departure times is written in terms of the corresponding semi-Markov kernel. The details are given in Section 6.

Table 1 summarizes the notation used in the analysis.

Symbol	Description
K_i	size of the finite population allowed to workstation i ($i = 1, 2$)
k_i	number of entities in buffer B_i ($i = 1, 2$)
c_i	number of servers available in workstation i ($i = 1, 2$)
μ_i	service rate of a server in workstation i
λ_D	throughput of the fork/join
$E(L_i)$	expected queue length at buffer B_i ($i = 1, 2$)

Table 1: Notation used in the analysis

5 Analysis of the Queue Length Process

Let $N_1(t)$ and $N_2(t)$ denote the number of units in buffers B_1 and B_2 respectively at time t . If at some time t , $N_i(t) = k_i, i = 1, 2$, then the remaining $K_i - k_i$ units are at station i . If $(K_i - k_i) < c_i$ then only $K_i - k_i$ of the c_i servers at the station are busy at time t . If $N_i(t) = K_i$, then all servers at station i are idle and the arrival process to buffer B_i temporarily shuts down. Finally, note that the operational characteristics of the fork/join station imply that buffers B_1 and B_2 cannot be both non-empty simultaneously, i.e., if $N_1(t) > 0$, then $N_2(t) = 0$, and vice versa.

With these definitions, the state of the system is characterized by $(k_1, k_2) = [N_1(t) = k_1, N_2(t) = k_2], t \geq 0$. Clearly, (k_1, k_2) is a continuous time Markov chain defined on the state space $[(K_1, 0), (K_1 - 1, 0) \dots, (1, 0), (0, 0), (0, 1), \dots, (0, K_2 - 1), (0, K_2)]$. The state transition rates for the continuous time Markov chain representing the queue length process are illustrated in Figure 2. It can be shown that this Markov chain is positive recurrent and therefore, if $P(k_1, k_2)$ denotes the steady state probability of state (k_1, k_2) then these probabilities must satisfy the following set of balance equations:

- For $1 \leq k_1 \leq K_1 - c_1$:

$$(c_1\mu_1 + c_2\mu_2)P(k_1, 0) = c_1\mu_1P(k_1 - 1, 0) + c_2\mu_2P(k_1 + 1, 0)$$

- For $K_1 - c_1 + 1 \leq k_1 \leq K_1$:

$$((K_1 - k_1)\mu_1 + c_2\mu_2)P(k_1, 0) = (K_1 - k_1 + 1)\mu_1P(k_1 - 1, 0) + c_2\mu_2P(k_1 + 1, 0)$$

- For $k_1 = k_2 = 0$:

$$(c_1\mu_1 + c_2\mu_2)P(0, 0) = c_1\mu_1P(0, 1) + c_2\mu_2P(1, 0)$$

- For $1 \leq k_2 \leq K_2 - c_2$:

$$(c_1\mu_1 + c_2\mu_2)P(0, k_2) = c_1\mu_1P(0, k_2 + 1) + c_2\mu_2P(0, k_2 - 1)$$

- For $K_2 - c_2 + 1 \leq k_2 \leq K_2$:

$$(c_1\mu_1 + (K_2 - k_2)\mu_2)P(0, k_2) = c_1\mu_1P(0, k_2 + 1) + (K_2 - k_2 + 1)\mu_2P(0, k_2 - 1) \quad (1)$$

Substituting $c_2\mu_2P(k_1+1, 0)$ by $\min(c_1, K_1-k_1) \times \mu_1P(k_1, 0)$ and $c_1\mu_1P(0, k_2+1)$ by $\min(c_2, K_2-k_2) \times \mu_2P(0, k_2)$ in the balance equations for $P(k_1, 0)$ and $P(0, k_2)$, respectively, the above equations

Figure 2: Rate diagrams for the Markov chain

can be simplified and subsequently solved to yield:

- For $k_1 = 1$ to $K_1 - c_1$:

$$P(k_1, 0) = \left(\frac{c_1 \mu_1}{c_2 \mu_2} \right)^{k_1} P(0, 0)$$

- For $k_1 = K_1 - c_1 + 1$ to K_1 :

$$P(k_1, 0) = c_1^{K_1 - c_1} \frac{c_1!}{(K_1 - k_1)!} \left(\frac{\mu_1}{c_2 \mu_2} \right)^{k_1} P(0, 0)$$

- For $k_2 = 1$ to $K_2 - c_2$:

$$P(0, k_2) = \left(\frac{c_2 \mu_2}{c_1 \mu_1} \right)^{k_2} P(0, 0)$$

- For $k_2 = K_2 - c_2 + 1$ to K_2 :

$$P(0, k_2) = c_2^{K_2 - c_2} \frac{c_2!}{(K_2 - k_2)!} \left(\frac{\mu_2}{c_1 \mu_1} \right)^{k_2} P(0, 0)$$

(2)

Finally, the normalization equation yields:

$$P(0, 0) = \left[1 + \sum_{k_1=1}^{K_1} P(k_1, 0) + \sum_{k_2=1}^{K_2} P(0, k_2) \right]^{-1} \quad (3)$$

Therefore, throughput of the fork/join is given by:

$$\lambda_D = \left[K_1 \mu_1 \sum_{k_2=1}^{K_2} P(0, k_2) + K_2 \mu_2 \sum_{k_1=1}^{K_1} P(k_1, 0) \right] \quad (4)$$

and the mean queue lengths at buffers B_1 and B_2 are given by:

$$E(L_1) = \sum_{k_1=1}^{K_1} k_1 P(k_1, 0) \quad \text{and} \quad E(L_2) = \sum_{k_2=1}^{K_2} k_2 P(0, k_2) \quad (5)$$

6 Analysis of the Departure Process

To analyze the departure process of the fork/join station, it is first characterized as a semi Markov process. Subsequently, the semi Markov kernel corresponding to the departure process and the transition matrix of the Markov chain embedded at departure instants are derived. These are then used to derive the marginal and joint distributions of inter-departure times from the fork/join station. The details are given in the sections below.

6.1 The Departure Process as a Semi-Markov Process

Recall that, departure occurs (simultaneously from both the buffers) when at least one entity is available in each buffer, B_1 and B_2 . Consider the random process, Z , corresponding to the sequence of states of the fork/join station at the departure instants, and the random process T , corresponding to the sequence of departure times at the fork/join station. The random process $T = \{\tau_n \mid n \in \mathbb{N}\}$ is the sequence of real valued random variables τ_n , where τ_n is the time of the n^{th} departure. Associated with the random process T is the incremental process $D = \{D_n \mid n \in \mathbb{N}\}$ where D_n is the time between departure n and $n - 1$, i.e. $D_n = \tau_n - \tau_{n-1}$ with $D_0 = 0$. For each $n \in \mathbb{N}$, define the random process $Z = \{z_n \mid n \in \mathbb{N}\}$ as the states of buffers B_1 and B_2 immediately after the n^{th} departure epoch τ_n . If (k_1, k_2) is the state immediately following a departure, it follows that the states $(K_1, 0)$ and $(0, K_2)$ can never be seen immediately after a departure. Therefore, z_n takes values from the set $[(K_1 - 1, 0), (K_1 - 2, 0), \dots, (1, 0), (0, 0), (0, 1), \dots, (0, K_2 - 2), (0, K_2 - 1)]$.

The distribution of the inter-departure times is derived by analyzing the pair of random processes (Z, D) . The time to the $n + 1^{th}$ departure, D_{n+1} depends on the states z_n and z_{n+1} . However, given these states, D_{n+1} is independent of the previous states $z_0, \dots, z_{n-1}$, and $D_0, \dots, D_n$. That is, the following relation holds

$$P(Z_{n+1}, D_{n+1} | Z_0, \dots, Z_n, D_0, \dots, D_n) = P(Z_{n+1}, D_{n+1} | Z_n) \forall n \in \mathbb{N} \quad (6)$$

implying that the output process (Z, D) is a Markov renewal process. The state space of the output process (Z, D) is $\mathbb{S} \times \mathbb{R}$, and where, $\mathbb{S}$ is $[(K_1 - 1, 0), (K_1 - 2, 0), \dots, (1, 0), (0, 0), (0, 1),$

$\dots, (0, -K_2 + 2), (0, -K_2 + 1)]$ and $\mathbb{R}$ is $(0, \infty)$. Since the output process (Z, D) is a Markov renewal process it is completely characterized by its semi-Markov kernel Q , where:

$$Q(Z_n, Z_{n+1}, t) = P(Z_{n+1}, D_{n+1} \leq t \mid Z_n) \quad (7)$$

For convenience the elements of the semi-Markov kernel can be expressed in the Laplace transform domain as

$$Q^*(Z_n, Z_{n+1}, ds) = \mathcal{L}\{Q(Z_n, Z_{n+1}, dt)\}. \quad (8)$$

This section describes how to compute the values of $Q(Z_n, Z_{n+1}, ds)$. Using $Q(Z_n, Z_{n+1}, ds)$ several characteristics of the output process (Z, D) can be obtained. For example, the state transition matrix P_D of the underlying Markov chain Z embedded at times τ_n , is obtained by setting $s = 0$ in $Q(z_n, z_{n+1}, ds)$, i.e.,

$$P(z_{n+1}|z_n) = Q^*(z_n, z_{n+1}, ds)_{s=0} \forall n \quad (9)$$

6.2 Construction of the Semi-Markov Kernel Q

Note that each departure instant from the fork/join station corresponds to an arrival instant at the buffer that was empty just prior to departure. The states z_n (prior to the $n + 1^{th}$ departure) and z_{n+1} (immediately following the $n + 1^{th}$ departure), determine the number of arrivals that occur in the non-empty buffer between the n^{th} and $n + 1^{th}$ departure instants. Based on this information, expressions for $P(z_{n+1}|z_n)$ and $Q^*(z_n, z_{n+1}, ds)$ are written. Consider the following example, where, $z_n = (k_1, 0) = (k_1, 0) = z_{n+1}$. This implies that the departure instant of the $n + 1^{th}$ departure corresponds to an arrival instant in buffer B_2 , and exactly one arrival occurs in buffer B_1 between the n^{th} and $n + 1^{th}$ departure instants. Therefore:

$$\begin{aligned} Q(z_n, z_{n+1}, t) &= P(\text{An arrival at buffer } B_1 \text{ precedes the arrival at buffer } B_2 \text{ prior to time } t) \\ &= \int_0^t \min(c_1, K_1 - k_1) \mu_1 e^{-\min(c_1, K_1 - k_1) \mu_1 \tau} e^{-c_2 \mu_2 \tau} d\tau \end{aligned}$$

Therefore in the Laplace domain one could write,

$$Q^*(z_n, z_{n+1}, ds) = \frac{\min(c_1, K_1 - k_1) \mu_1}{\min(c_1, K_1 - k_1) \mu_1 + c_2 \mu_2 + s}.$$

By similar reasoning, one could write the expression for $P(z_{n+1}|z_n)$ as:

$$\begin{aligned} P(z_{n+1}|z_n) &= \int_0^\infty \min(c_1, K_1 - k_1) \mu_1 e^{-\min(c_1, K_1 - k_1) \mu_1 \tau} e^{-c_2 \mu_2 \tau} d\tau \\ &= \frac{\min(c_1, K_1 - k_1) \mu_1}{\min(c_1, K_1 - k_1) \mu_1 + c_2 \mu_2} \end{aligned}$$

Note that $P(z_{n+1}|z_n)$ can also be computed as $Q^*(z_n, z_{n+1}, ds)_{s=0}$.

The elements $Q(z_n, z_{n+1}, t)$ of the semi-markov kernel, Q and $P(Z_{n+1} | Z_n)$ of the transition probability matrix P_D , are derived by recognizing that Q and P_D can be partitioned into 9 distinct regions (see Figure 3). In each region, the logic described above is applied to obtained the desired expressions. These are summarized below.

Region 1:

$Z_{n+1} = (i, 0), Z_n = (k_1, 0)$ and $(0, 0) \leq (k_1, 0) < (K_1 - 2, 0), (k_1, 0) < (i, 0) < (K_1 - 1, 0)$

$$\begin{aligned} Q^*(Z_{n+1}, Z_n, ds) &= \left[\prod_{z=k_1}^i \frac{\min(c_1, K_1 - z)\mu_1}{\min(c_1, K_1 - z)\mu_1 + c_2\mu_2 + s} \right] \left[\frac{c_2\mu_2}{\min(c_1, K_1 - i - 1)\mu_1 + c_2\mu_2 + s} \right] \\ P(Z_{n+1} | Z_n) &= \left[\prod_{z=k_1}^i \frac{\min(c_1, K_1 - z)\mu_1}{\min(c_1, K_1 - z)\mu_1 + c_2\mu_2} \right] \left[\frac{c_2\mu_2}{\min(c_1, K_1 - i - 1)\mu_1 + c_2\mu_2} \right] \end{aligned} \quad (10)$$

Region 2:

$Z_{n+1} = (i, 0), Z_n = (k_1, 0)$ and $(0, 0) \leq (k_1, 0) \leq (K_1 - 1, 0), (i, 0) = (K_1 - 1, 0)$

$$\begin{aligned} Q^*(Z_{n+1}, Z_n, ds) &= \prod_{z=k_1}^{K_1-1} \left[\frac{\min(c_1, K_1 - z)\mu_1}{\min(c_1, K_1 - z)\mu_1 + c_2\mu_2 + s} \right] \\ P(Z_{n+1} | Z_n) &= \prod_{z=k_1}^{K_1-1} \left[\frac{\min(c_1, K_1 - z)\mu_1}{\min(c_1, K_1 - z)\mu_1 + c_2\mu_2} \right] \end{aligned} \quad (11)$$

Region 3:

$Z_{n+1} = (i, 0), Z_n = (k_1, 0)$ and $(0, 0) < (k_1, 0) \leq (K_1 - 2, 0), (i, 0) = (k_1, 0)$

$$\begin{aligned} Q^*(Z_{n+1}, Z_n, ds) &= \left[\frac{\min(c_1, K_1 - k_1)\mu_1}{\min(c_1, K_1 - k_1)\mu_1 + c_2\mu_2 + s} \right] \left[\frac{c_2\mu_2}{\min(c_1, K_1 - k_1 - 1)\mu_1 + c_2\mu_2 + s} \right] \\ P(Z_{n+1} | Z_n) &= \left[\frac{\min(c_1, K_1 - k_1)\mu_1}{\min(c_1, K_1 - k_1)\mu_1 + c_2\mu_2} \right] \left[\frac{c_2\mu_2}{\min(c_1, K_1 - k_1 - 1)\mu_1 + c_2\mu_2} \right] \end{aligned} \quad (12)$$

Region 4:

$Z_{n+1} = (i, 0), Z_n = (k_1, 0)$ and $(0, 0) < (k_1, 0) \leq (K_1 - 1, 0), (i, 0) = (k_1 - 1, 0)$

$$\begin{aligned} Q^*(Z_{n+1}, Z_n, ds) &= \left[\frac{c_2\mu_2}{\min(c_1, K_1 - k_1)\mu_1 + c_2\mu_2 + s} \right] \\ P(Z_{n+1} | Z_n) &= \left[\frac{c_2\mu_2}{\min(c_1, K_1 - k_1)\mu_1 + c_2\mu_2} \right] \end{aligned} \quad (13)$$

Region 5:

$Z_{n+1} = (0, j), Z_n = (0, k_2)$ and $(0, 0) \leq (0, k_2) < (0, K_2 - 2), (0, k_2) < (0, j) < (0, K_2 - 1)$

$$\begin{aligned}
Q^*(Z_{n+1}, Z_n, ds) &= \left[\prod_{z=k_2}^j \frac{\min(c_2, K_2 - z)\mu_2}{c_1\mu_1 + \min(c_2, K_2 - z)\mu_2 + s} \right] \left[\frac{c_1\mu_1}{c_1\mu_1 + \min(c_2, K_2 - j - 1)\mu_2 + s} \right] \\
P(Z_{n+1} \mid Z_n) &= \left[\prod_{z=k_2}^j \frac{\min(c_2, K_2 - z)\mu_2}{c_1\mu_1 + \min(c_2, K_2 - z)\mu_2} \right] \left[\frac{c_1\mu_1}{c_1\mu_1 + \min(c_2, K_2 - j - 1)\mu_2} \right]
\end{aligned} \tag{14}$$

Region 6:

$Z_{n+1} = (0, j), Z_n = (0, k_2)$ and $(0, 0) \leq (0, k_2) \leq (0, K_2 - 1), (0, j) = (0, K_2 - 1)$

$$\begin{aligned}
Q^*(Z_{n+1}, Z_n, ds) &= \prod_{z=k_2}^{K_2-1} \left[\frac{\min(c_2, K_2 - z)\mu_2}{c_1\mu_1 + \min(c_2, K_2 - z)\mu_2 + s} \right] \\
P(Z_{n+1} \mid Z_n) &= \prod_{z=k_2}^{K_2-1} \left[\frac{\min(c_2, K_2 - z)\mu_2}{c_1\mu_1 + \min(c_2, K_2 - z)\mu_2} \right]
\end{aligned} \tag{15}$$

Region 7:

$Z_{n+1} = (0, j), Z_n = (0, k_2)$ and $(0, 0) < (0, k_2) \leq (0, K_1 - 2), (0, j) = (0, k_2)$

$$\begin{aligned}
Q^*(Z_{n+1}, Z_n, ds) &= \left[\frac{\min(c_2, K_2 - k_2)\mu_2}{c_1\mu_1 + \min(c_2, K_2 - k_2)\mu_2 + s} \right] \left[\frac{c_1\mu_1}{c_1\mu_1 + \min(c_2, K_2 - k_2 - 1)\mu_2 + s} \right] \\
P(Z_{n+1} \mid Z_n) &= \left[\frac{\min(c_2, K_2 - k_2)\mu_2}{c_1\mu_1 + \min(c_2, K_2 - k_2)\mu_2} \right] \left[\frac{c_1\mu_1}{c_1\mu_1 + \min(c_2, K_2 - k_2 - 1)\mu_2} \right]
\end{aligned} \tag{16}$$

Region 8:

$Z_{n+1} = (0, j), Z_n = (0, k_2)$ and $(0, 0) < (0, k_2) \leq (0, K_2 - 1), (0, j) = (0, k_2 - 1)$

$$\begin{aligned}
Q^*(Z_{n+1}, Z_n, ds) &= \left[\frac{c_1\mu_1}{c_1\mu_1 + \min(c_2, K_2 - k_2)\mu_2 + s} \right] \\
P(Z_{n+1} \mid Z_n) &= \left[\frac{c_1\mu_1}{c_1\mu_1 + \min(c_2, K_2 - k_2)\mu_2} \right]
\end{aligned} \tag{17}$$

Region 9:

$$Z_{n+1} = (0, 0), Z_n = (0, 0)$$

$$\begin{aligned} Q^*(Z_{n+1}, Z_n, ds) &= \left[\frac{c_1\mu_1 \ c_2\mu_2}{c_1\mu_1 + c_2\mu_2} \right] \left[\frac{1}{\min(c_1, K_1 - 1)\mu_1 + c_2\mu_2 + s} + \frac{1}{c_1\mu_1 + \min(c_2, K_2 - 1)\mu_2 + s} \right] \\ P(Z_{n+1} \mid Z_n) &= \left[\frac{c_1\mu_1 \ c_2\mu_2}{c_1\mu_1 + c_2\mu_2} \right] \left[\frac{1}{\min(c_1, K_1 - 1)\mu_1 + c_2\mu_2} + \frac{1}{c_1\mu_1 + \min(c_2, K_2 - 1)\mu_2} \right] \end{aligned} \quad (18)$$

Using the above equations for the transition probabilities, $P(Z_{n+1} \mid Z_n)$ transition probabilities, the transition matrix P_D is constructed. Similarly, the semi-Markov kernel, Q can be constructed in terms of $Q^*(Z_{n+1}, Z_n, ds)$ defined above.

		Z_{n+1}											
		5,0	4,0	3,0	2,0	1,0	0,0	0,1	0,2	0,3	0,4	0,5	
Z_n	5,0	2	4										
	4,0	2	3	4									
	3,0	2	1	3	4								
	2,0	2	1	1	3	4							
	1,0	2	1	1	1	3	4						
	0,0	2	1	1	1	1	9	5	5	5	5	6	
	0,1						8	7	5	5	5	6	
	0,2							8	7	5	5	6	
	0,3								8	7	5	6	
	0,5									8	7	6	
	0,6										8	6	

Figure 3: Regions in the transition matrix P_D and Semi Markov kernel Q for $K_i = 6, c_i = 2, i = 1, 2$

6.3 Marginal and Joint Distributions of Inter-departure Times

Since the output process, $\{Z, D\}$ is an irreducible, nonnull, recurrent and persistent Markov renewal process for finite K_1 and K_2 , it possesses a stationary distribution. Let $\Pi_D = \{\pi_D(z)\}$, the stationary probability distribution, with $\pi_D(z)$ denoting the probability that the fork/join station is in state $z \in \mathbb{S}$ at a departure instant. Then, the stationary probability vector $\{\Pi_D\}$ of the underlying Markov chain is obtained by recognizing that:

$$\Pi_D = \Pi_D P_D \quad (19)$$

Furthermore,

$$\sum_{k_1=0}^{K_1-1} \pi_D(k_1, 0) + \sum_{k_2=1}^{K_2-1} \pi_D(0, k_2) = 1 \quad (20)$$

Note that since $P(Z_{n+1} | Z_n)$ is independent of n , the inter-departure times are identically distributed. Let $D_D(t)$ be the distribution function of the inter-departure times i.e. $D_D(t) = P\{D_m \leq t\} \forall m$. Then, $D_D(t)$ can be written in terms of Π_D and the distribution function of the inter-arrival times at buffer B_1 and B_2 . Define:

$$\begin{aligned}\Pi_1 &= \sum_{k_1=1}^{K_1-1} \pi(k_1, 0) \\ \Pi_0 &= \pi(0, 0) \\ \Pi_2 &= \sum_{k_2=1}^{K_2-1} \pi(0, k_2)\end{aligned}$$

The marginal distribution function of the inter-departure times is given by:

$$D_D(t) = \Pi_1(1 - e^{-c_2\mu_2 t}) + \Pi_2(1 - e^{-c_1\mu_1 t}) + \Pi_0(1 - e^{-c_1\mu_1 \tau})(1 - e^{-c_2\mu_2 t}) \quad (21)$$

From $D_D(t)$, the density function and the first and second moment of the inter-departure times can be obtained. The density function is given by:

$$f(t) = (\Pi_1 + \Pi_0) c_2 \mu_2 e^{-c_2 \mu_2 t} + (\Pi_2 + \Pi_0) c_1 \mu_1 e^{-c_1 \mu_1 t} - \Pi_0 (c_1 \mu_1 + c_2 \mu_2) e^{-(c_1 \mu_1 + c_2 \mu_2) t} \quad (22)$$

It is evident from the above equations that the inter-departure times have a phase-type distribution. The first moment or the expected inter-departure time:

$$E(D) = \frac{(\Pi_1 + \Pi_0)}{c_2 \mu_2} + \frac{(\Pi_2 + \Pi_0)}{c_1 \mu_1} - \frac{\Pi_0}{(c_1 \mu_1 + c_2 \mu_2)} \quad (23)$$

Note that $E(D)$ equals $\frac{1}{\lambda_D}$. The second moment of the inter-departure times:

$$E(D^2) = \frac{2(\Pi_1 + \Pi_0)}{(c_2 \mu_2)^2} + \frac{2(\Pi_2 + \Pi_0)}{(c_1 \mu_1)^2} - \frac{2\Pi_0}{(c_1 \mu_1 + c_2 \mu_2)^2} \quad (24)$$

Note that the expression for the marginal distribution $D_D(t)$ is also given by $D_D(t) = \Pi_D Q(t) \mathbf{U}$, where $\mathbf{U}$ is a column vector with all elements equal to 1. The joint distribution of k successive inter-departure times from the fork/join station, $P(D_m \leq t_m, D_{m+1} \leq t_{m+2}, \dots, D_{m+k} \leq t_{m+k})$ is equal to $\Pi_D Q(t_m) Q(t_{m+1}) \dots Q(t_{m+k}) \mathbf{U}$ for any (m, k) .

7 Impact of Inputs from Multi-Server Stations

This section summarizes the performance insights obtained from the exact analysis of the queue length and departure processes. Table 2 below shows a sample of the results illustrating the different performance measures of a fork/join station with inputs from multi-server stations. The results correspond to two sets of experiments. In the first set, the number of servers, $c_i, i = 1, 2$ are varied

from 5 to 20, but server rates are adjusted to ensure that the maximum station rate (or station capacity) is equal to 1 at each station (i.e. $c_1\mu_1 = c_2\mu_2 = 1.0$). In the second set, $c_i, i = 1, 2$ are varied from 5 to 20, but server rates are adjusted to ensure that the maximum station rate (or station capacity) at station 1 is equal to 1 (i.e. $c_1\mu_1 = 1.0$), and station 2 is equal to 1.25 (i.e. $c_2\mu_2 = 1.25$). In all the experiments, the population size $K_1 = K_2 = 25$.

The table indicates that although, station capacities may be equal, the throughput of the fork/join station varies with the number of servers at each station. In fact, throughput appears to decrease with the increase in number of servers. Correspondingly, it is also observed that the mean queue length at the synchronization buffers B_1 and B_2 decrease with increase in servers at the input stations. When station capacities are imbalanced, a similar trend is observed with respect to throughput and mean queue lengths, although the trend is less significant. In the following subsections, the above phenomenon is investigated in more detail. The main objective is to determine in what situations do the performance of fork/join station with inputs from multi-server stations differ significantly from that of fork/join stations with inputs from single server stations with the same capacity.

μ_1	c_1	$c_1\mu_1$	K_1	μ_2	c_2	$c_2\mu_2$	K_2	λ_D	$E(L_1)$	$E(L_2)$
0.2	5	1	25	0.2	5	1	25	0.9783	5.7628	5.7628
0.1	10	1	25	0.1	10	1	25	0.9739	4.8252	4.8252
0.05	20	1	25	0.05	20	1	25	0.9537	2.8474	2.8474
0.2	5	1	25	0.25	5	1.25	25	1	0.0246	18.8098
0.1	10	1	25	0.125	10	1.25	25	1	0.0532	15.4238
0.05	20	1	25	0.0625	20	1.25	25	0.998	0.2274	8.3771

Table 2: Sample results showing effect of multiple servers

7.1 Effect of Multiple Servers on Queue Lengths

This section compares the effect of multiple servers on the queue lengths at the synchronization buffers B_1 and B_2 of the fork/join station. Two kinds of multi-server fork/join stations are considered. In the first case (labeled as the MM case), both station 1 and station 2 are multi-server stations, and it is assumed that $c_1 = K_1$ and $c_2 = K_2$. The population size is varied from $K_1 = K_2 = 5$ to 50, and in each case the server rates are adjusted to ensure that $c_1\mu_1 = c_2\mu_2 = 1$, i.e., station capacities are balanced. In the second case (labeled as the MS case), station 1 is a multi-server station with $c_1 = K_1$, while station 2 is a single server station with $c_2 = 1$, and $K_2 \geq 1$. Again, the population size is varied from $K_1 = K_2 = 5$ to 50, and the server rate at station 1 is adjusted to ensure that $c_1\mu_1 = \mu_2 = 1$. Finally, the performance is compared against that of a system where both stations 1 and 2 are single server stations (labeled as the SS case). Here $c_1 = c_2 = 1$ while $K_1, K_2 \geq 1$. Therefore, although the population size is varied from $K_1 = K_2 = 5$ to 50, $\mu_1 = \mu_2 = 1$. Figure 4 plots the mean queue lengths $E(L_1)$ and $E(L_2)$ as a function of the

population size for each of these three systems. The labels SS, MS and MM are used to distinguish the mean queue lengths in the corresponding cases. Based on the graphs, several observations are made regarding effect of multiple servers.

Figure 4: Mean queue lengths at fork/join stations for MM, MS and SS systems

- The graphs indicate that the mean queue lengths at buffers B_1 and B_2 vary roughly linearly with the population constraints. Further, even if the station capacities are the same in all three systems, SS, MS and MM, the mean queue lengths can be significantly different.
- $E(L_i^{SS}) \geq E(L_i^{MM}), i = 1, 2$: When station capacities are the same, the mean queue length at buffers B_1 and B_2 is always has higher for SS system than the MM system. Although station capacities at each station is equal to one in both MM and SS systems, the service rates at each station in the SS system is equal to 1 when the station is not idle. However, in the case of the MM system, the service rate of a station is at most equal to 1 (being equal to one only when all the servers at a station are busy). Since in the MM system, the service rates at a station depend on the number of servers busy at any given time (or equivalently on the contents in buffers B_1 and B_2), the average service rate at each station is less than that in an SS system, and this phenomenon leads less built up of queues at buffers B_1 and B_2 .
- $E(L_1^{SS}) = E(L_2^{SS})$ and $E(L_1^{MM}) = E(L_2^{MM})$ but $E(L_1^{MS}) \leq E(L_2^{MS})$: For SS and MM systems, when station capacities and population limits are the same, the mean queue length at buffer B_1 is equal to the mean queue length at buffer B_2 . For MS systems, even when station capacities are the same, the mean queue length at the buffer following multi-server station in an MS system (i.e. buffer B_1) is always less than the mean queue length at the buffer following single server station (i.e. buffer B_2). In the MS system, station 2 operates at a rate of one when it is not idle. However, station 1 operates at a rate of one only when all servers at the station are busy. This imbalance in station rates, despite balanced station capacities results in queues at buffer B_2 .

- $E(L_1^{MS}) \leq E(L_1^{MM})$ and $E(L_2^{MS}) \geq E(L_2^{SS})$: The mean queue length at the buffer following multi-server station in an MS system (i.e. buffer B_1) is always less than the mean queue length at the corresponding buffer in an MM system having identical station capacities and population limits. Similarly, the mean queue length at the buffer following single server station in an MS system (i.e. buffer B_2) is always higher than the mean queue length at the corresponding buffer in an SS system having identical station capacities and population limits. In the MS system defined above, station 2 operates at a rate of one when it is not idle. However, station 1 operates at a rate of one only when all servers at the station are busy. At all other times, the station rate is strictly less than one.

The next section discusses insights related to the departure process.

7.2 Effect of Multiple Servers on Inter-departure Times

This section compares the inter-departure times from SS, MS, and MM systems in terms of their mean and SCV. The comparison is conducted for balanced as well as unbalanced systems.

7.2.1 Insights with respect to throughput

This section discusses the insights with respect to throughput from the fork/join station under different settings. For the balanced case, the parameter settings for the SS, MS and MM cases are similar to that considered in the previous section, i.e., station capacities are set equal to 1 (i.e. $c_1\mu_1 = c_2\mu_2 = 1$), and the population limits K_1 and K_2 are varied from 5 to 50. For the MM system, $c_1 = K_1$ and $c_2 = K_2$, while for the MS system, $c_1 = K_1$ and $c_2 = 1$, and for the SS system, $c_1 = c_2 = 1$ and $K_1 = K_2 \geq 1$.

Figure 5: Throughput from fork/join station for MM, MS and SS configurations

Figure 5 plots the throughput λ_D a function of the population size for each of these three systems. The labels SS, MS and MM are used to denote the throughput corresponding to each system. The following insights are obtained:

- The graph indicates that the throughput for each system increases monotonically and is a concave function of population size.
- $\lambda_D^{MM} \leq \lambda_D^{MS} \leq \lambda_D^{SS}$: When station capacities are the same, for a given population size, the throughput of the SS system is the highest while that of the MM system is the lowest. This is because, the service rates at each station in the SS system is equal to 1 when the station is not idle. In contrast, for the MM system, the service rate of each station is at most equal to 1 (the rate being equal to one only when all the servers at a station are busy).

7.2.2 Insights with respect to distribution of inter-departure times

This section discusses insights with respect to the distribution of inter-departure times in systems with imbalance in station capacities. Again, comparisons are made between MM, MS and SS systems. In the MM case, it is assumed that $c_1 = K_1$ and $c_2 = K_2$ and the population size is varied from $K_1 = K_2 = 5$ to 50. When station capacities are balanced, for each value of population size, the server rates are adjusted to ensure that $c_1\mu_1 = c_2\mu_2 = 1$. When station capacities are imbalanced, server rates are adjusted such that $c_1\mu_1 = 1.25$ and $c_2\mu_2 = 1$. For the MS case, station 1 is a multi-server station with $c_1 = K_1$, while station 2 is a single server station with $c_2 = 1$, and $K_2 \geq 1$. Again, the population size is varied from $K_1 = K_2 = 5$ to 50. When station capacities are balanced, for each value of population size, the server rates are adjusted to ensure that $c_1\mu_1 = \mu_2 = 1$. When station capacities are imbalanced, server rates are adjusted such that either $c_1\mu_1 = 1.25$ and $c_2\mu_2 = 1$ or $c_1\mu_1 = 1$ and $c_2\mu_2 = 1.25$ respectively. Finally, for the SS case $c_1 = c_2 = 1$ and $K_1, K_2 \geq 1$. Therefore, station capacities are balanced, for each value of population size, $\mu_1 = \mu_2 = 1$. When station capacities are imbalanced, server rates are adjusted such that either $\mu_1 = 1.25$ and $\mu_2 = 1$ or $\mu_1 = 1$ and $\mu_2 = 1.25$ respectively.

Table 3 summarizes the expected inter-departure time, $E(D)$, the throughput, λ_D , and the SCV of the inter-departure time, c_D^2 , for SS, MS and MM systems. Figures 6 and 7 plot the marginal distribution, $f(t)$ of the inter-departure times as well as the exponential density function with mean equal to $E(D)$ for several cases. These graphs indicate the difference between $f(t)$ and an exponential distribution with the same mean. Since Som et al. [22] provide similar comparisons for the SS system, the related graphs is not presented here. In the figures, $R = \frac{c_1\mu_1}{c_2\mu_2}$. From Table 3, and Figures 6 and 7, the following observations are made:

- The results re-iterate the observation made earlier, that, when station capacities are the same, for a given population size, the throughput of the SS system is the highest while that of the MM system is the lowest.

			SS System			MS System			MM System		
Max. Rate: Station 1	Max. Rate: Station 2	$K_1 = K_2$	$E(D)^{SS}$	λ_D^{SS}	$c_D^{2,SS}$	$E(D)^{MS}$	λ_D^{MS}	$c_D^{2,MS}$	$E(D)^{MM}$	λ_D^{MM}	$c_D^{2,MM}$
1	1	5	1.10	0.91	0.90	1.13	0.88	0.86	1.20	0.83	0.83
1	1	50	1.01	0.99	1.00	1.02	0.98	0.98	1.06	0.94	0.95
1.25	1	5	1.02	0.98	0.93	1.04	0.96	0.87	1.10	0.91	0.83
1.25	1	50	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	0.98
1	1.25	5	1.02	0.98	0.93	1.05	0.95	0.91	1.10	0.91	0.83
1	1.25	50	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.98
SS System: Maximum Rate at Station 1= μ_1 , Maximum Rate at Station 2= μ_2 MS System: Maximum Rate at Station 1= $c_1\mu_1 = K_1\mu_1$, Maximum Rate at Station 2= μ_2 MM System: Maximum Rate at Station 1= $c_1\mu_1 = K_1\mu_1$, Maximum Rate at Station 2= $c_2\mu_2 = K_2\mu_2$											

Table 3: Comparison of the departure processes

- For all three systems, SS, MS and MM, the SCV of the inter-departure times is less than one, the SCV for the MM system being the smallest. Further, the SCV of the inter-departure times for an SS system is closer to that of the exponential distribution (c_D^2 is closer to 1) than for an MS or MM system with the same station capacity and population size.
- For all three systems, MM, MS and SS systems, the SCV of the inter-departure times increases with population size. Som et al. [22] make this observation for the SS system. This analysis shows that a similar behavior holds for MS and MM systems as well.
- When compared to systems with balanced station capacities, the SCV of the inter-departure times from imbalanced systems, tends faster towards 1 at smaller values of population sizes.

The above results indicate that in fork/join stations with inputs from multi-server stations with exponential service times, the inter-departure times have a phase type distribution with SCV less than 1. However, the SCVs tend towards 1 with a decrease in the number of servers at each station, and an increase in the population limit or imbalance in station capacities.

8 Conclusions and Extensions

This paper presents an exact analysis of a fork/join station in a closed queuing network with inputs from stations composed of multiple exponential servers. The analysis is composed of two parts: queue length process analysis and departure process analysis. The queue length process is analyzed as a continuous time Markov process and steady state probability distributions are obtained. From these probability distributions, performance measures such as throughput and mean queue lengths are obtained. Subsequently, an exact analysis of the semi-Markov kernel characterizing the departure process is conducted and exact expressions for the marginal and joint distributions of inter-departure times are derived. These analysis are used to study the effect of inputs from finite

population sub-networks with multiple servers on key performance measures. Comparison studies show that replacing the multiple serves by a single server with equivalent capacity could result in significantly different estimates of throughput, synchronization delays and queue lengths at the individual buffers. These insights lead to better representation of synchronization constraints in closed queuing network models of computer networks, fabrication/assembly systems and material control strategies for manufacturing systems.

Acknowledgements: The research of the first author was supported by a grant provided by the Research Foundation - Flanders (FWO) and was conducted while he was a graduate student at University of Antwerp, in the Faculty of Applied Economics, Environment, Technology and Technology Management Department.

Bibliography

- [1] F. Baccelli and A.M. Makowski, *Queueing models for systems with synchronization constraints*, Proceedings of the IEEE **77** (1989), no. 1, 138–161.
- [2] F. Baccelli, W.A. Massey, and D. Towsley, *Acyclic fork/join queuing networks*, Journal of the ACM **36** (1989), 615–642.
- [3] B. Baynat and Y. Dallery, *Approximate techniques for general closed queueing networks with subnetworks having population constraints*, European Journal of Operational Research **69** (1993), 250–264.
- [4] F. Bonomi, *An approximate analysis for a class of assembly-like queues*, Queueing Systems **1** (1987), 289–309.
- [5] Liesje de Boeck, *Performance analysis of assemblies in open queueing systems*, Ph. D Thesis, Faculty of Applied Economics, University of Antwerp (2003).
- [6] M. Di Mascolo, Y. Frein, and Y. Dallery, *An analytical method for performance evaluation of kanban controlled production systems*, Operations Research **44** (1996), no. 1, 50–64.
- [7] Nico Goossens, *Blocking systems with zero buffer: A kanban-like approach*, Ph. D Thesis, Faculty of Applied Economics, University of Antwerp (2006).
- [8] J.H. Harrison, *Assembly-like queues*, Journal of Applied Probability **10** (1973), no. 2, 354–367.
- [9] W.J. Hopp and J.T. Simon, *Bounds and heuristics for assembly-like queues*, Queueing systems **4** (1989), no. 2, 137–156.
- [10] C. Knessl, *On the diffusion approximation to a fork and join queueing model*, SIAM Journal of Applied Mathematics **51** (1997), no. 1, 160–171.

- [11] S.-S. Ko and R.F. Serfozo, *Response times in $m/m/s$ fork-join networks*, Advances in Applied Probability **36** (2004), 854–871.
- [12] A. Krishnamurthy and R. Suri, *Performance analysis of single stage kanban controlled production systems using parametric decomposition*, Queueing Systems- Theory and Applications (To appear) (2006).
- [13] A. Krishnamurthy, R. Suri, and M. Vernon, *Two-moment approximations for throughput and mean queue length of a fork/join station with general input processes*, Stochastic modeling and optimization of manufacturing systems and supply chains, Shanthikumar, J.G., Yao D.D., and Zijm, W.H.M. (Editors), International Series in Operations Research and Management Science, Kluwer Academic Publishers (2003), 87–126.
- [14] ———, *Analysis of a fork/join synchronization station with inputs from coxian servers in a closed queuing network*, Annals of Operations Research **125** (2004), 69–94.
- [15] A. Kumar and R. Shorey, *Performance analysis and scheduling of stochastic fork-join jobs in a multicomputer system*, IEEE Transactions on Parallel and Distributed Systems **4** (1993), no. 10, 1147–1164.
- [16] G. Latouche, *Queues with paired customers*, Journal of Applied Probability **18** (1981), no. 3, 684–696.
- [17] Y.C. Liu and H.G. Perros, *Approximate analysis of a closed fork/join model*, European Journal of Operational Research **53** (1991), no. 3, 382–392.
- [18] R. Nelson and A.N. Tantawi, *Approximate analysis of fork/join synchronization in parallel queues*, IEEE Transactions on Computers **37** (1988), no. 6, 739–743.
- [19] B. Prabhakar, N. Bambos, and T.S. Mountford, *The synchronizaton of poisson processes and queueing networks with service and synchronization nodes*, Advances in Applied Probability **32** (2000), 824–843.
- [20] P.C. Rao and R. Suri, *Approximate queueing models of fabrication / assembly systems: Part i-single level systems*, Production an Operations Management **3** (1994), 244–275.
- [21] ———, *Performance analysis of an assembly station with input from multiple fabrication lines*, Production and Operations Management **9** (2000), no. 3, 283–302.
- [22] P. Som, W.E. Wilhelm, and R.L. Disney, *Kitting process in a stochastic assembly system*, Queueing Systems **17** (1994), 471–490.
- [23] M. Takahashi, H. Osawa, and T. Fujisawa, *A stochastic assembly system with resume levels*, Asia-Pacific Journal of Operational Research **15** (1998), 127–146.

- [24] ———, *On a synchronizaton queue with two finite buffers*, Queueing Systems **36** (2000), 107–123.
- [25] E. Varki, *Mean value technique for closed fork-join networks*, Proceedings of ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems (1999).
- [26] S. Varma and A. Makowski, *Interpolation approximations for symmetric fork-join queues*, Performance Evaluation **20** (1994), 245–265.

(a) $R=0.5$ and $K_1=K_2=5$

(b) $R=0.5$ and $K_1=K_2=50$

(c) $R=0.8$ and $K_1=K_2=5$

(d) $R=0.8$ and $K_1=K_2=50$

(e) $R=1.0$ and $K_1=K_2=5$

(f) $R=1.0$ and $K_1=K_2=50$

Figure 6: Comparison of the distribution of inter-departure times for an MM system

(a) $R=0.8$ and $K_1=K_2=5$

(b) $R=0.8$ and $K_1=K_2=50$

(c) $R=1.0$ and $K_1=K_2=5$

(d) $R=1.0$ and $K_1=K_2=50$

(e) $R=1.25$ and $K_1=K_2=5$

(f) $R=1.25$ and $K_1=K_2=50$

Figure 7: Comparison of the distribution of inter-departure times for an MS system