# PROPERTIES OF FORK/JOIN NETWORKS WITH BLOCKING UNDER VARIOUS OPERATING MECHANISMS

Y. DALLERY, Z. LIU and D. TOWSLEY

**COINS Technical Report 92-39**
May 1992

# Properties of Fork/Join Queueing Networks with Blocking under Various Operating Mechanisms

Yves DALLERY* Zhen LIU† Don TOWSLEY‡

April 2, 1992

**Abstract**

Queueing networks with fork/join mechanisms and finite-capacity buffers are of interest because they are particularly suited to the modeling and performance evaluation of a large class of discrete event systems such as manufacturing systems (e.g., manufacturing flow lines, assembly lines, kanban controlled manufacturing systems). In a recent paper, Dallery, Liu and Towsley [10] considered a special class of queueing networks with fork/join mechanisms and finite-capacity buffers called Basic Fork/Join Queueing Networks with Blocking (B-FJQN/B). For this class of networks, they established a set of properties such as duality, reversibility, symmetry and concavity. However, in order to be able to accurately model the various operating mechanisms (blocking, loading and unloading mechanisms) encountered in manufacturing systems, we need to consider a larger class of networks that will be referred to as Fork/Join Queueing Networks with Blocking (FJQN/B). The purpose of this paper is to investigate the properties of FJQN/B's. Our approach is the following: first show that any FJQN/B can equivalently be represented as a B-FJQN/B; secondly, using the results derived in [10] for the underlying B-FJQN/B, establish the properties of the model under consideration. Another issue of interest in this paper is to compare the behavior of two models having different operating mechanisms.

**Keywords:** Queueing networks, fork/join mechanisms, finite buffers, manufacturing systems, blocking mechanisms, loading and unloading policies, throughput properties.

*Laboratoire MASI, Université Pierre et Marie Curie, 4, place Jussieu, 75252 Paris Cedex 05, FRANCE.

†INRIA Centre Sophia Antipolis, 2004 route des Lucioles, 06565 Valbonne, FRANCE.INRIA Centre Sophia Antipolis, 2004 route des Lucioles, 06565 Valbonne, FRANCE.

‡Department of Computer Science, University of Massachusetts, Amherst, MA 01003, U.S.A. The work by this author was supported in part by the National Science Foundation under grant ASC 88-8802764 and by the Office of Naval Research under grant ONR N00014-87-K-0796. This work was performed in part while the author was visiting Laboratoire MASI, UPMC, Paris.

## 1 Introduction

Queueing networks with fork/join mechanisms and finite-capacity buffers are of interest because they are particularly suited to the modeling and performance evaluation of a large class of discrete event systems such as manufacturing systems.

Many manufacturing systems, namely manufacturing flow lines (also referred to as production lines or transfer lines), can be modeled as tandem or closed tandem queueing networks with finite-capacity buffers [9]. Moreover, fork/join mechanisms are also of high interest for modeling manufacturing systems. First, fork and join operations correspond to physical operations in a manufacturing system. Join operations (also called assembly operations in the manufacturing literature) are encountered in the modeling of assembly systems [12, 15]. In this case, a join operation corresponds to the assembly of two or more subcomponents into a single component. Fork operations (also called disassembly operations in the manufacturing literature) are also encountered. Consider for instance a manufacturing system where, at a certain stage, parts need to be fixed onto pallets. The resulting part/pallet pair then visits a set of machines in order for different operations to be performed on the part. When these operations have been performed, the part is unloaded and the pallet is released. This unloading operation is a fork operation for which an item (the part/pallet pair) is split into two items (the part and the pallet). Fork and join mechanisms may also be encountered when modeling certain control mechanisms in manufacturing systems. For instance, fork and join operations are encountered in models of kanban controlled manufacturing systems [13, 14]. In such systems, a part is processed at a given stage of the manufacturing system only if a kanban associated with this stage is available. The part then holds the kanban while being processed throughout this stage. The kanban is released only when the part is consumed by the next stage, which occurs when a kanban of the next stage is available. Thus, kanbans act as production orders. In that case, a join operation corresponds to the assembly of a part and a kanban and a fork operation corresponds to the disassembly of the part/kanban pair.

The purpose of this paper is to introduce and study the behavior of a class of queueing networks with fork/join mechanisms and finite-capacity buffers that can be used to model such manufacturing systems. A wide variety of mechanisms that exist in manufacturing systems are provided in these networks. For example, both blocking-after-service and blocking-before-service blocking mechanisms (cf. [21]) are considered. In the case of assembly and disassembly, both independent and simultaneous loading and unloading mechanisms are modeled. Here, an independent loading mechanism allows an assembly machine to load different subcomponents independent of each other, whereas the simultaneous loading mechanism requires that all subcomponents be present and loaded at the same time. The independent and simultaneous unloading mechanisms operate in an analogous manner at disassembly. Other features accounted for in these queueing networks are described later on.

We primarily study the behavior of the throughput of these networks. Specifically, we study properties of reversibility, symmetry, monotonicity and concavity. Our approach in this regard is to show how these queueing networks can be transformed into queueing networks belonging to a smaller class of networks first introduced in [2] and studied in [10]. We shall refer to such networks as Basic Fork/Join Queueing Networks with Blocking (B-FJQN/B's). For such networks we established duality, reversibility, symmetry, and concavity properties in [10]. Except for the concavity property, these results were obtained under very weak assumptions on the sequences of service times. These assumptions include as a special case sequences of service times that are i.i.d. random variables. The concavity property is restricted to a class of distributions called PERT distributions [4].

We first establish the equivalence between the larger class of networks of interest to us in this paper and the class of B-FJQN/B's. Using this equivalence, we extend some of the previously mentioned properties exhibited by B-FJQN/B's to the larger class of queueing networks. Another issue of interest in this paper is to compare the behavior of two models having different operating mechanisms. Again, our approach is to compare the behavior of the equivalent B-FJQN/B's. Some of the results established in this paper were already obtained in some special cases (mainly tandem and closed tandem queueing networks). However, the results presented in this paper not only generalize and unify these results but also offer a simple way of transferring results obtained for a basic class of networks, namely B-FJQN/B's, to a large variety of queueing networks with fork/join mechanisms and finite buffers.

The paper is organized as follows. In Section 2, the class of B-FJQN/B's is defined and the main properties derived in [10] are reviewed. Further properties pertaining to the monotonicity of the throughput of a B-FJQN/B are also presented. In Section 3, the class of Fork/Join Queueing Networks with Blocking (FJQN/B's) considered in this paper is introduced. It is then shown how any model of this class can equivalently be represented as a B-FJQN/B. This equivalence is used in Section 4 to establish the properties of FJQN/B's. Some applications of these results to models of manufacturing systems are described in Section 5. Finally, extensions of our approach to handle a larger class of models is briefly discussed in Section 6.

# 2 Basic Fork/Join Queueing Networks with Blocking and their Properties

In this section, we define the class of B-FJQN/B's and summarize the most important results obtained by Dallery, Liu and Towsley [10] for this class of networks. We also present some further results pertaining to monotonicity properties of the throughput of B-FJQN/B's. All these results will be used later in the paper for establishing properties of the general class of networks.

## 2.1 Basic Fork/Join Queueing Networks with Blocking

A Basic Fork/Join Queueing Networks with Blocking (B-FJQN/B) is a queueing network consisting of a set of servers and a set of buffers such that each buffer has exactly one upstream server and one downstream server. On the other hand, each server may have several input buffers and/or several output buffers. We restrict our attention to the case where there is at most one buffer between any pair of servers. This is without loss of generality since any B-FJQN/B can be transformed into an equivalent B-FJQN/B satisfying this condition; see Theorem 3.2 of [10].

Under this condition, a B-FJQN/B is structurally characterized by a directed graph whose nodes are the set of servers and whose directed edges represent the buffers that connect the servers. Let $V$ denote the set of servers and $E$ the set of directed edges. An edge $(i,j) \in E$ indicates that there is a buffer connecting server $i$ to server $j$ and that material flows through this buffer from server $i$ to server $j$. Servers $i$ and $j$ will be referred to as the upstream and downstream servers of buffer $(i,j)$, respectively. Each buffer has a finite capacity. Let $B_{i,j} \in \mathbb{N}^+$ denote the finite capacity of buffer $(i,j)$ and let $\boldsymbol{B}$ denote the vector of buffer capacities. Let $\mathcal{N} = (V, E, \boldsymbol{B})$ denote the B-FJQN/B and let $G = (V, E)$ denote the underlying directed graph. This graph is assumed to be connected.

Let $u(i)$ be the set of upstream buffers (or input buffers) of server $i$ and $d(i)$ be the set of downstream servers (or output buffers) of server $i$, defined as:

$$u(i) = \{(j,i) \in E, j \in V\}, \qquad i \in V,$$

$$d(i) = \{(i,j) \in E, j \in V\}, \qquad i \in V.$$

Let $p(i)$ be the set of immediate server predecessors of server $i$ and $s(i)$ be the set of immediate server successors of server $i$, defined as:

$$p(i) = \{j \in V \ |(j,i) \in E\}, \qquad i \in V,$$

$$s(i) = \{j \in V \ |(i,j) \in E\}, \qquad i \in V.$$

The behavior of a B-FJQN/B, in general, depends on an initial marking, which is defined as the number of jobs present in each buffer at the initial instant. Let $M_{i,j}$ denote the number of jobs present at the initial instant in buffer $(i,j)$ and let $\boldsymbol{M}$ be the initial job marking vector. Also, let $H_{i,j}$ denote the number of holes (empty rooms) present at the initial instant in buffer $(i,j)$ and let $\boldsymbol{H}$ be the initial hole marking vector. Note that $H_{i,j} = B_{i,j} - M_{i,j}$ and $\boldsymbol{H} = \boldsymbol{B} - \boldsymbol{M}$. Let $\mathcal{S} = (\mathcal{N}, \boldsymbol{M}) = (V, E, \boldsymbol{B}, \boldsymbol{M})$ denote the B-FJQN/B $\mathcal{N}$ associated with the initial marking $\boldsymbol{M}$. Finally, a B-FJQN/B is further characterized by the durations of service times which are random variables (r.v.'s). The durations of the service periods at server $i$ are given by a sequence of non-negative service times, $\{\sigma_{i,n}\}_{n \geq 1}$, $i \in V$. We note that service times may take

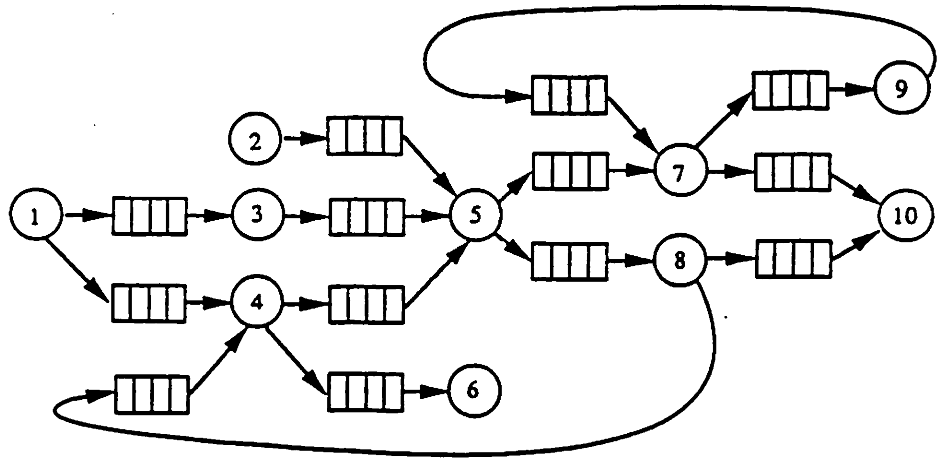


Figure 1: Example of a B-FJQN/B.

value zero. The introduction of servers whose service periods are of zero length is useful for modeling synchronization mechanisms.

The B-FJQN/B behaves in the following manner. Server $i$ initiates a *service period* (or *service activity*) whenever there resides at least one job in each of the buffers in $u(i)$ and there is space for at least one job in each of the buffers in $d(i)$. Server $i$ is said to be *starved* if at least one of its upstream buffers is empty and *blocked* if at least one of its downstream buffers is full. Note that the server can simultaneously be starved and blocked. Jobs remain in the buffers in $u(i)$ throughout the service period, i.e., there is no space associated with the servers for storing jobs. At the completion of the service period, a job is removed from each of the buffers in $u(i)$ and a job is immediately placed in each of the buffers in $d(i)$. (Note that the number of upstream buffers and the number of downstream buffers may be different.) We assume that server $i$ initiates its first service period at the first instant $t \geq 0$ where it is neither starved nor blocked. (Note that a more general initial timing condition was considered in [10]).

There may be some servers for which there are no incoming edges. Each such server is referred to as a *source*. A source is never starved. (It is sometimes of interest to interpret a source as being a server with an infinite number of jobs available at its input.) There may be other servers for which there are no outgoing edges. Each such server is referred to as a *sink*. A sink is never blocked. Note that if server $i$ is a source (resp. sink), we have $u(i) = \emptyset$ and $p(i) = \emptyset$ (resp. $d(i) = \emptyset$ and $s(i) = \emptyset$).

An example of a B-FJQN/B is given in Figure 1. This B-FJQN/B has 10 servers and 13 buffers. Servers 1 and 2 are sources and servers 6 and 10 are sinks.

**Remark.** It is important to observe that there is no buffer space associated with each server in a B-FJQN/B. As a result, jobs are always contained in the buffers. Models of FJQN/B's in which servers include buffer spaces to accommodate the jobs in service will be considered in Section 3.1.

## 2.2 Properties of B-FJQN/B's

In this section, we review the properties of B-FJQN/B's derived in [10] that will be useful in the rest of the paper. (More details, as well as the proof of these results, can be found in [10].) These properties pertain to the asymptotic throughput of B-FJQN/B's and require one or more of the following assumptions on the sequences of service times.

**Assumption A1.** The service times form jointly stationary and ergodic sequences of integrable r.v.'s.

**Assumption A2.** The sequences of service times at different servers are mutually independent.

**Assumption A3.** The sequences of service times are jointly reversible.

**Assumption A4.** The service times form mutually independent sequences of i.i.d. r.v.'s having PERT distributions [4, 10].

It is worth emphasizing that the first three assumptions include the case where the service times are i.i.d. r.v.'s as a special case. Assumption A4 is more restrictive but includes exponential and Erlang distributions as a special case.

Consider a B-FJQN/B $\mathcal{S} = (\mathcal{N}, M)$. Let $D_{i,n}$ denote the completion time of the $n$-th service period of server $i$. Denote by $\theta_i(\mathcal{S})$ the (asymptotic) throughput of server $i \in V$, given by:

$$\theta_i(\mathcal{S}) = \left[\lim_{n\to\infty} \frac{E[D_{i,n}]}{n}\right]^{-1}, \qquad i \in V, \tag{2.1}$$

provided the limit exists.

Let $C$ be a cycle in $\mathcal{N}$ and let $E(C)$ denote the set of edges of $C$. Let us define an arbitrary orientation of this cycle. $E(C)$ can be partitioned into two subsets with respect to this reference orientation. Let $E^+(C)$ be the subset of edges oriented according to the reference orientation and $E^-(C)$ be the subset of edges oriented in the reverse direction. We have $E^+(C) + E^-(C) = E(C)$. Let $I_C^+(M)$ ($I_C^-(M)$, respectively) be the total number of jobs (holes, respectively) in all buffers corresponding to the reference direction plus the total number of holes (jobs, respectively) in all buffers corresponding to the reverse direction, i.e.,

$$\begin{aligned} I_C^+(M) &= \sum_{(i,j)\in E^+(C)} M_{i,j} + \sum_{(i,j)\in E^-(C)} (B_{i,j} - M_{i,j}); \\ I_C^-(M) &= \sum_{(i,j)\in E^-(C)} M_{i,j} + \sum_{(i,j)\in E^+(C)} (B_{i,j} - M_{i,j}). \end{aligned}$$

Note that

$$I_C^+(M) + I_C^-(M) = B_C \equiv \sum_{(i,j)\in E(C)} B_{i,j}, \qquad I_C^+(M) = I_C^-(B - M).$$

A FJQN/B $\mathcal{S} = (\mathcal{N}, M)$ is said to be deadlocked if it is impossible for any server to commence a service period, i.e., every server is either starved, blocked, or both.

**Theorem 2.1** *A B-FJQN/B with initial marking $M$ is deadlock-free iff the following relation is satisfied,*

$$I_C^+(M) > 0 \text{ and } I_C^-(M) > 0,$$

*or equivalently*

$$0 < I_C^+(M) < B_C$$

*for all cycles $C$ in $\mathcal{N}$.*

The results pertaining to the throughput of a B-FJQN/B derived in [10] are based on the following evolution equations that relate the quantities $D_{i,n}$:

$$D_{i,n} = \sigma_{i,n} + \max\left(D_{i,n-1}, \max_{j\in p(i)} D_{j,n-M_{j,i}}, \max_{k\in s(i)} D_{k,n-(B_{i,k}-M_{i,k})}\right), \forall i \in V, n \geq 1 \quad (2.2)$$

where, by convention, $D_{i,n} = 0$, $n \leq 0$.

This equation simply states that the instant at which the $n$-th service period of server $i$ is completed is equal to the instant of its beginning plus the duration of this service period, that is $\sigma_{i,n}$. The second term expresses the instant at which the $n$-th service period of server $i$ can begin. Indeed, three conditions must be fulfilled: the server must have completed its $(n-1)$-th service activity; it must not be starved; and it must not be blocked.

The following result establishes the existence of the asymptotic throughputs under very weak assumptions on the sequences of the service times. Moreover, it states that all the servers of a B-FJQN/B have the same throughput. This later result is the generalization of the conservation of flow property of tandem queues [9].

**Theorem 2.2** *Let $\mathcal{S}$ be a B-FJQN/B. Then, under Assumption A1, the asymptotic throughputs of the servers exist and moreover we have:*

$$\theta_i(\mathcal{S}) = \theta(\mathcal{S}), \quad \forall i \in V. \quad (2.3)$$

The quantity $\theta(\mathcal{S})$ will be referred to as the throughput of $\mathcal{S}$.

The next result establishes that the throughput of a B-FJQN/B is independent of the initial markings, provided these initial markings are equivalent as defined below. Consider a B-FJQN/B $\mathcal{S} = (\mathcal{N}, M)$.

**Definition 2.1** *Let $\mathcal{N}$ contain exactly $n_c$ distinct elementary cycles $C_1, \cdots, C_{n_c}$. Let*

$$I(M) = (I_1^+(M), I_2^+(M), \cdots, I_{n_c}^+(M)).$$

*Then markings $M$ and $M'$ are equivalent (written $M \sim M'$) iff $I(M) = I(M')$.*

**Theorem 2.3** *Let $M^1$ and $M^2$ be two initial markings of a B-FJQN/B $\mathcal{N}$. Under Assumptions A1 and A2, if $M^1 \sim M^2$, we have:*

$$\theta(\mathcal{N}, M^1) = \theta(\mathcal{N}, M^2).$$

The next result derived in [10] is called the reversibility property.

**Definition 2.2** *Let $\mathcal{S} = (\mathcal{N}, M)$ be a B-FJQN/B. The reverse of $\mathcal{S}$ is the B-FJQN/B $\mathcal{S}^r = (V^r, E^r, B^r, M^r)$ defined as follows:*

$$\begin{aligned} V^r &= V, \\ E^r &= \{(i,j) | (j,i) \in E\}, \\ B^r_{i,j} &= B_{j,i}, \qquad (j,i) \in E, \\ M^r_{i,j} &= M_{j,i}, \qquad (j,i) \in E. \end{aligned}$$

In words, the B-FJQN/B $\mathcal{S}^r$ is the reverse of $\mathcal{S}$ if it is obtained from $\mathcal{S}$ by reversing the flow of jobs through all the buffers while keeping the same initial marking. The following result states that the reverse of a B-FJQN/B has the same throughput as the original network.

**Theorem 2.4** *Consider a B-FJQN/B $\mathcal{S}$ and its reverse $\mathcal{S}^r$ with the same (joint distribution of the) sequences of service times. Then, under Assumptions A1 and A3, we have:*

$$\theta(\mathcal{S}^r) = \theta(\mathcal{S}). \tag{2.4}$$

Let $\mathcal{S}^s = (\mathcal{N}, B - M)$ be the B-FJQN/B with the symmetrical initial marking of $\mathcal{S} = (\mathcal{N}, M)$. The following symmetry property was established in [10]. It states that the throughput of a B-FJQN/B with a given initial marking is identical to the throughput of the same B-FJQN/B with symmetrical initial marking.

**Theorem 2.5** *Consider a B-FJQN/B $\mathcal{N}$. Then, under Assumptions A1 and A3, we have:*

$$\theta(\mathcal{N}, B - M) = \theta(\mathcal{N}, M). \tag{2.5}$$

Another result proved in [10] is the concavity of the throughput of a B-FJQN/B with respect to both the buffer capacities and the initial marking.

**Theorem 2.6** *Under Assumption A4, $\theta(\mathcal{S})$ is a concave function of $B$ and $M$.*

## 2.3 Monotonicity Properties of B-FJQN/B's

In this subsection, we present some monotonicity properties of the throughput of a B-FJQN/B. Let $\mathbf{0}$ denote the vector with all components being zero.

The following result is due to the monotonicity property of stochastic marked graphs [4] and the equivalence between B-FJQN/B's and strongly connected marked graphs [10].

**Theorem 2.7** *Consider two B-FJQN/B's $\mathcal{S}^1 = (V, E, \mathbf{B}^1, \mathbf{M})$ and $\mathcal{S}^2 = (V, E, \mathbf{B}^2, \mathbf{M})$ that differ only from one another by the buffer capacity vector. Assume that $\mathbf{B}^2 = \mathbf{B}^1 + \mathbf{k}$, with $\mathbf{k} \geq \mathbf{0}$. If the service times are the same in both systems, i.e., $\sigma^2_{i,n} = \sigma^1_{i,n}$, $i \in V$, $n \geq 1$, then*

$$D^2_{i,n} \leq D^1_{i,n}, \qquad \forall i \in V, \quad n \geq 1. \tag{2.6}$$

**Corollary 2.1** *Under Assumption A1, we have:*

$$\theta(V, E, \mathbf{B} + \mathbf{k}, \mathbf{M}) \geq \theta(V, E, \mathbf{B}, \mathbf{M}), \qquad \forall \mathbf{k} \geq \mathbf{0}.$$

The above corollary states that increasing the capacity of any buffer in a B-FJQN/B increases the throughput. On the other hand, increasing the initial marking of a B-FJQN/B does not always increase the throughput. Actually, it may decrease it. (The simplest example of B-FJQN/B's for which this is encountered is a closed tandem queueing network with all buffers being finite; see [11].) However, increasing the initial marking, as well as the capacity of any buffer by the same amount, does increase the throughput. This result is expressed in the following corollary.

**Corollary 2.2** *Under Assumption A1, we have:*

$$\theta(V, E, \mathbf{B} + \mathbf{k}, \mathbf{M} + \mathbf{k}) \geq \theta(V, E, \mathbf{B}, \mathbf{M}), \quad \forall \mathbf{k} \geq \mathbf{0}.$$

**Proof.** The proof is simply obtained by applying Corollary 2.1 to the full duals [10] of the two FJQN/B's and then transposing the result to the original B-FJQN/B's using the duality property established in [10]. ∎

Another result of interest is concerned with subnetworks.

**Definition 2.3** *Consider a B-FJQN/B $\mathcal{S} = (V, E, \mathbf{B}, \mathbf{M})$. The B-FJQN/B $\tilde{\mathcal{S}} = (\tilde{V}, \tilde{E}, \tilde{\mathbf{B}}, \tilde{\mathbf{M}})$ is a subnetwork of $\mathcal{S}$ if:*

$$\tilde{V} \subset V,$$

$$\begin{aligned}
\tilde{E} &\subset E, \\
\tilde{B}_{i,j} &= B_{i,j}, \qquad (i,j) \in \tilde{E}, \\
\tilde{M}_{i,j} &= M_{i,j}, \qquad (i,j) \in \tilde{E},
\end{aligned}$$

*and the graph $(\tilde{V}, \tilde{E})$ is connected.*

The following result is also due to the monotonicity property of stochastic marked graphs [4] and the equivalence between B-FJQN/B's and strongly connected marked graphs [10].

**Theorem 2.8** *Let $S$ be a B-FJQN/B and $\tilde{S}$ be a subnetwork of $S$. If $\tilde{\sigma}_{i,n} = \sigma_{i,n}$, $i \in \tilde{V}$, $n \geq 1$, then,*

$$\tilde{D}_{i,n} \leq D_{i,n}, \qquad \forall i \in \tilde{V}; \quad n \geq 1. \tag{2.7}$$

**Corollary 2.3** *Let $S$ be a B-FJQN/B and $\tilde{S}$ be a subnetwork of $S$. Under Assumption A1, we have:*

$$\theta(\tilde{S}) \geq \theta(S).$$

As a special case of the above result, it follows that the throughput of a B-FJQN/B is increased when deleting the buffer connecting any two servers.

Last, we establish a result that will be useful in the paper.

**Definition 2.4** *Consider a B-FJQN/B $S = (V, E, \mathbf{B}, \mathbf{M})$. The B-FJQN/B $S^e = (V^e, E^e, \mathbf{B}^e, \mathbf{M}^e)$ is an* expanded network *of $S$ with respect to server $i$ if*

$$\begin{aligned}
V^e &= V + \{i_0\}, \\
E^e &= E - \{(i,k), k \in s(i)\} + \{(i,i_0)\} + \{(j,i_0), j \in p(i)\} + \{(i_0,k), k \in s(i)\}, \\
B^e_{j,k} &= B_{j,k}, \qquad j,k \in V, \quad j \neq i, \\
B^e_{j,i} &= B^e_{j,i_0} = B_{j,i}, \qquad j \in p(i), \\
B^e_{i_0,k} &= B_{i,k}, \qquad k \in s(i), \\
B^e_{i,i_0} &\in \mathbb{N}^+, \\
M^e_{j,k} &= M_{j,k}, \qquad j,k \in V, \quad j \neq i, \\
M^e_{j,i} &= M^e_{j,i_0} = M_{j,i}, \qquad j \in p(i), \\
M^e_{i_0,k} &= M_{i,k}, \qquad k \in s(i), \\
M^e_{i,i_0} &= 0.
\end{aligned}$$

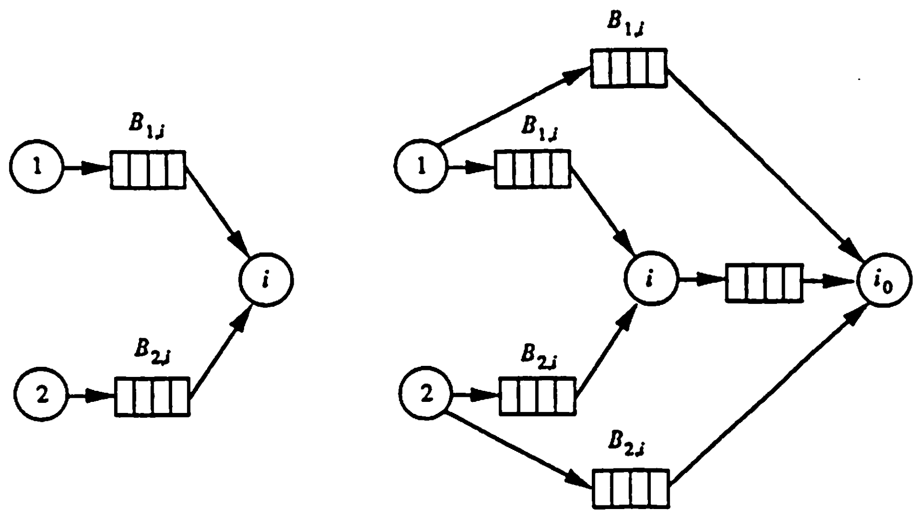


Figure 2: Illustration of the expanded B-FJQN/B.

In words, $S^e$ is constructed from $S$ as follows. Server $i$ is replaced by a pair of servers, say $i$ and $i_0$, separated by a finite capacity buffer. For each buffer $(j, i), j \in p(i)$, in $S$, there are two buffers in $S^e$, one connecting server $j$ to server $i$, the other connecting server $j$ to server $i_0$. These buffers have both the same capacity and the same initial marking as buffer $(j, i)$ in $S$. Server $i_0$ in $S^e$ is connected in the same way as server $i$ in $S$ to all servers $k \in s(i)$. Buffer $(i, i_0)$ in $S^e$ has a finite capacity $B_{i,i_0} \in \mathbb{N}^+$ and initial marking $M_{i,i_0} = 0$. This transformation is illustrated in Figure 2.

**Theorem 2.9** *Let $S$ be a B-FJQN/B and $S^e$ be an expanded network of $S$ with respect to server $i$. For any sequences of service times such that $\sigma^e_{i,n} = \sigma_{i,n}, \forall i \in V, n \geq 1$ and $\sigma^e_{i_0,n} = 0, \forall n \geq 1$, then,*

$$D^e_{j,n} \leq D_{j,n}, \qquad \forall j \in V, \quad n \geq 1. \tag{2.8}$$

**Proof.** In order to compare the behavior of $S$ and $S^e$, let us introduce a third network, denoted by $S^m$, which "short-circuits" server $i_0$ of $S^e$. This network is obtained from $S^e$ by inserting a buffer between server $i$ and each server $k \in s(i)$, with capacity $B_{i,k}$ and initial marking $M_{i,k}$. It turns out that $S$ can be obtained from $S^m$ by deleting server $i_0$ as well as all its upstream and downstream buffers. Now, in network $S^m$, servers $i$ and $i_0$ are connected exactly in the same way to the rest of the network. This, together with the fact that $\sigma^m_{i_0} = \sigma^e_{i_0} = 0$, implies that any time a completion occurs at server $i$ in $S^m$, a completion occurs immediately at server $i_0$, i.e., $D^m_{i_0,n} = D^m_{i,n}$. As a result, server $i_0$ does not at all affect the behavior of network $S^m$ and thus we get: $D^m_{j,n} = D_{j,n}, \forall j \in V, n \geq 1$. (The formal proof of this result is omitted but can easily be obtained using the evolution equations of $S$ and $S^m$.) On the other hand, by definition of

$\mathcal{S}^m$, $\mathcal{S}^e$ is a subnetwork of $\mathcal{S}^m$. Thus, from Theorem 2.8, we get: $D^e_{j,n} \leq D^m_{j,n}, \forall j \in V, n \geq 1$. By combining the above two results, we get $D^e_{j,n} \leq D_{j,n}, \forall j \in V, n \geq 1$. ∎

The above result can be interpreted as follows. When server $i$ becomes blocked in $\mathcal{S}$ because one of its downstream buffers is full, server $i_0$ in $\mathcal{S}^e$ also becomes blocked. Server $i$ in $\mathcal{S}^e$, however, is not blocked and can still perform service activities as long as buffer $(i, i_0)$ is not full. Therefore, server $i$ in $\mathcal{S}^e$ will always be ahead of server $i$ in $\mathcal{S}$.

**Remark.** The above result also holds when the service time of server $i$ in $\mathcal{S}$ is split in any manner in between server $i$ and server $i_0$ in $\mathcal{S}^e$, i.e., $\sigma^e_{i,n} + \sigma^e_{i_0,n} = \sigma_{i,n}$. However, the proof is slightly more tedious.

**Corollary 2.4** *Let $\mathcal{S}$ be a B-FJQN/B and $\mathcal{S}^e$ be an expanded network of $\mathcal{S}$ with respect to server $i$. Under Assumption A1, we have:*

$$\theta(\mathcal{S}^e) \geq \theta(\mathcal{S}).$$

# 3 Fork/Join Queueing Networks with Blocking under Various Operating Mechanisms

In this section, we introduce a general class of Fork/Join Queueing Networks with Blocking (FJQN/B). This class contains the class of B-FJQN/B's described in Section 2.1. FJQN/B's are more general than B-FJQN/B's in that: servers may have space to accommodate jobs being processed; jobs can be loaded and unloaded according to different policies; and various blocking mechanisms may be considered. In Section 3.2, we show that any FJQN/B can equivalently be modeled as a B-FJQN/B. This equivalence will be used in Section 4 to establish properties of FJQN/B's.

## 3.1 Definition

A Fork/Join Queueing Network with Blocking (FJQN/B) is a queueing network consisting of a set of servers and a set of buffers such that each buffer has exactly one upstream server and one downstream server. On the other hand, each server may have several input buffers and/or several output buffers. There is at most one buffer between any pair of servers. There may be some servers that have either no input buffers or no output buffers. A FJQN/B is structurally characterized by a directed graph whose nodes represent the servers and whose directed edges represent the buffers that connect the servers. Let $V$ denote the set of servers and $E$ the set of directed edges. An edge $(i, j) \in E$ indicates that there is a buffer connecting server $i$ to server $j$ and that jobs flow through this buffer from server $i$ to server $j$. Let $G = (V, E)$ denote the underlying directed graph, which is assumed to be connected.

Let $u(i)$ be the set of upstream buffers (or input buffers) of server $i$ and $d(i)$ be the set of downstream servers (or output buffers) of server $i$, and let $p(i)$ be the set of immediate server predecessors of server $i$ and $s(i)$ be the set of immediate server successors of server $i$ (see Section 2.1 for a formal definition). A server of a FJQN/B is generally referred to as an *assembly/disassembly* server. A server that has no upstream buffers, i.e., $u(i) = \emptyset$, is referred to as a *source*. A server that has no downstream buffers, i.e., $d(i) = \emptyset$, is referred to as a *sink*. A server $i$ that has at most one upstream buffer and at most one downstream buffer, i.e., $|u(i)| \leq 1$ and $|d(i)| \leq 1$, is referred to as a *simple server*. A server $i$ that has several upstream buffers and at most one downstream buffer, i.e., $|u(i)| > 1$ and $|d(i)| \leq 1$, is referred to as an *assembly server*. A server $i$ that has several downstream buffers and at most one upstream buffer, i.e., $|u(i)| \leq 1$ and $d(i)| > 1$, is referred to as a *disassembly server*.

Each buffer $(i, j)$ has a finite capacity $B_{i,j} \in \mathbb{N}^+$. Let $\boldsymbol{B}$ denote the buffer capacity vector. In addition, some servers may have buffer spaces to accommodate jobs in service. For sake of simplicity, we only consider the following two cases: either a server has no buffer space to accommodate jobs in service, or a server has space to accommodate all of the jobs in service, i.e., a job from each of its upstream buffers. The first type of server will be referred to as an *unbuffered server* or *U-server* while a server of the second type will be referred to as a *buffered server* or *B-server*.

In order to characterize the behavior of a FJQN/B, we first need to define the behavior of the servers. Consider first a U-server, say $i$. Server $i$ initiates a *service period* (or *service activity*) whenever there resides at least one job in each of the buffers in $u(i)$ and there is space for at least one job in each of the buffers in $d(i)$. Jobs remain in the buffers in $u(i)$ throughout the service period. At the completion of the service period, a job is removed from each of the buffers in $u(i)$ and a job is immediately placed in each of the buffers in $d(i)$. In other words, a U-server behaves exactly as a server of a B-FJQN/B.

We now define the behavior of B-servers. There are many different types of B-servers. They differ from each other according to the loading and unloading policies as well as the blocking mechanism. For simplicity, we will only consider several special cases. For practical applications, it is in general sufficient to deal with such cases. We note however that the methodology used in this paper could be applied to servers with other operating mechanisms.

**General behavior.** All of the B-servers considered in this paper have the following common behavior.

- Any time a B-server performs an operation, it consumes one job from each of its upstream buffers and places one job in each of its downstream buffers. (Note that this is similar to the behavior of U-servers.)
- A B-server cannot begin its service activity unless all of the jobs (one from each of its upstream buffers) have been loaded onto the server.

- No new job can be loaded onto the server before all of the jobs produced during the previous service activity have been unloaded.

**Loading policy.** We consider two loading policies referred to as the *independent loading* (IL) policy and *simultaneous loading* (SL) policy.

- In the case of the IL policy, the jobs coming from each of the upstream buffers can be loaded onto the server independently of the other jobs.
- In the case of the SL policy, jobs stay in the upstream buffers until one job in each of the upstream buffers is available. At this instant, all of the jobs are simultaneously (and instantaneously) loaded onto the server.

**Unloading policy.** We consider two unloading policies referred to as the *independent unloading* (IU) policy and *simultaneous unloading* (SU) policy.

- In the case of the IU policy, jobs can be unloaded independently of each other into the downstream buffers.
- In the case of the SU policy, jobs remain on the server until space becomes available in each of the downstream buffers. At the instant that space is available for all jobs, they are simultaneously (and instantaneously) unloaded into the downstream buffers.

**Blocking mechanism.** We consider three blocking mechanisms referred to as *blocking-after-service* (BAS) mechanism, *blocking-before-service* (BBS) mechanism and *blocking-before-service with conditional loading* (BBS-CL) mechanism.

- In the case of the BAS mechanism, a server initiates a service activity whenever a job from each of its upstream buffers has been loaded onto the server. The server becomes blocked if, at the end of the service activity, at least one job cannot be unloaded (because a downstream buffer is full). It remains blocked until all the jobs have been unloaded.
- In the case of the BBS mechanism, a server initiates a service activity whenever a job from each of its upstream buffers has been loaded onto the server and there is space for at least one job in each of its downstream buffers. A server is blocked if at least one of its downstream buffers is full.
- The BBS-CL mechanism is identical to the BBS mechanism except that no job can be loaded onto the server while the server is blocked. If server $i$ operates under the BBS-CL mechanism, we assume that $B_{i,j} > 1, \forall j \in s(i)$.

**Remark.** Note that in the case of the BBS and BBS-CL mechanisms, the unloading policy is irrelevant since a buffer space in each of the downstream buffers is, by definition, available at the end of any service activity.

By combining these different operating mechanisms and taking into account the above remark, we end up with 8 different types of B-servers:

- BAS/IL/IU server: a B-server with a blocking-after-service mechanism, independent loading policy and independent unloading policy;
- BAS/SL/SU server: a B-server with a blocking-after-service mechanism, simultaneous loading policy and simultaneous unloading policy;
- BAS/IL/SU server: a B-server with a blocking-after-service mechanism, independent loading policy and simultaneous unloading policy;
- BAS/SL/IU server: a B-server with a blocking-after-service mechanism, simultaneous loading policy and independent unloading policy;
- BBS/IL server: a B-server with a blocking-before-service mechanism and independent loading policy;
- BBS/SL server: a B-server with a blocking-before-service mechanism and simultaneous loading policy;
- BBS-CL/IL server: a B-server with a blocking-before-service with conditional loading mechanism and independent loading policy;
- BBS-CL/SL server: a B-server with a blocking-before-service with conditional loading mechanism and simultaneous loading policy.

Let $T_i$ denote the type of server $i$, either a U-server or one of the above 8 types of B-servers. Let $\boldsymbol{T}$ denote the server type vector.

**Remark.** The above definitions of operating mechanisms of B-servers apply to any general assembly/disassembly server. As a result, they apply in particular to any special case of assembly/disassembly servers, namely sources, sinks, simple servers, assembly servers and disassembly servers. Note however that in some cases, part of the characterization given for a general assembly/disassembly server may be irrelevant. For instance, in the case of a simple server, the loading and unloading policies are irrelevant. Actually, in that case, there are only three different types of operating mechanisms: BAS, BBS and BBS-CL. (Note that in the case of a simple server, BBS and BBS-CL have alternatively been referred to as blocking-before-service with place occupied (BBS-PO) and blocking-before-service with place non-occupied (BBS-PNO) in [21].) In the case of a disassembly server (and a source), the loading policy is irrelevant. In the case of an assembly server (and a sink), the unloading policy is irrelevant. In the case of a sink, the blocking mechanism is irrelevant.

The behavior of a FJQN/B, in general, depends on an initial condition. This initial condition includes the number of jobs present in each buffer at the initial instant, as well as the information pertaining to the initial condition of each B-server. Let $M_{i,j}$ denote the total number of jobs present at the initial instant in buffer $(i,j)$. Note that $0 \leq M_{i,j} \leq B_{i,j}$. In the case of a B-server, we also need to specify its initial condition. Consider a B-server $i$ with any blocking mechanism (BAS, BBS or BBS-CL). Let $L_{i,j}$ denote the quantity which indicates whether a job from buffer $(j,i)$ is currently loaded on the server ($L_{i,j} = 1$) or not ($L_{i,j} = 0$), $j \in p(i)$. In the case of a simultaneous loading policy, $L_{i,j} = L_i$ does not depend on the predecessor node $j \in p(i)$. Note that in the case of a server with a BBS-CL mechanism, this initial condition has to be consistent with the blocking mechanism. Indeed, the server space cannot be occupied if one of the downstream buffers is full. In the case of a B-server with the BAS mechanism, it may be of interest to consider an initial condition for which the server is blocked. For that, we need to introduce counterparts of the above quantities. Let $U_{i,k} = 1$ if server $i$ has completed its service activity but the job going to buffer $(i,k)$ cannot be unloaded because this buffer is full, and let $U_{i,k} = 0$ otherwise, $k \in s(i)$. For a simultaneous unloading policy, $U_{i,k} = U_i$ depends only on the server, $i$. Note that the quantities $L_i$, or $L_{i,j}, j \in p(i)$, and $U_i$, or $U_{i,k}, k \in s(i)$, have to be consistent with the condition prescribing that no new job can be loaded onto the server before all the jobs produced during the previous service activity have been unloaded. Last, we arbitrarily assign $L_{i,j} = U_{i,k} = 0$ in the case of $U$ servers, and $U_{i,k} = 0$ in the case of type BBS B-servers.

Let $\boldsymbol{M}$ be the initial marking vector of the buffers. Let $\boldsymbol{L}$ and $\boldsymbol{U}$ be the initial marking vector of the servers. A FJQN/B is further characterized by the durations of service times which are random variables (r.v.'s). The durations of the service periods at server $i$ are given by a sequence of non-negative service times, $\{\sigma_{i,n}\}_{n \geq 1}$, $i \in V$. The complete characterization of a FJQN/B, say $\mathcal{S}$, is thus given by $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T}, \boldsymbol{L}, \boldsymbol{U})$.

**Remark.** Note that a B-FJQN/B is a FJQN/B in which all of the servers are U-servers.

## 3.2 Modeling a FJQN/B as a B-FJQN/B

In this section, we show how a FJQN/B can equivalently be modeled as a B-FJQN/B. By equivalent, we mean that, given any sequences of service times, $\{\sigma_{i,n}\}_{n \geq 1}$, $i \in V$ and equivalent initial conditions, the sample path behavior of the two networks is identical. In particular, the instants of the beginning and completion of the $n$-th service activity of each server $i$ are the same in both networks, $\forall i \in V, n \geq 1$. Moreover, at any time, there is a one-to-one correspondence between the buffer contents of both networks.

Let $\mathcal{S}$ denote the original FJQN/B and $\mathcal{S}^b$ denote the *equivalent* B-FJQN/B we are looking for. $\mathcal{S}^b$ can be obtained by transforming each server in $\mathcal{S}$, one at a time, into a subnetwork consisting solely of U-servers.

Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T}, \boldsymbol{L}, \boldsymbol{U})$ be an arbitrary FJQN/B. Consider any B-server of $\mathcal{S}$, say $i$. We transform $\mathcal{S}$ into a second FJQN/B $\mathcal{S}' = (V', E', \boldsymbol{B}', \boldsymbol{M}', \boldsymbol{T}', \boldsymbol{L}', \boldsymbol{U}')$ that differs from $\mathcal{S}$ only

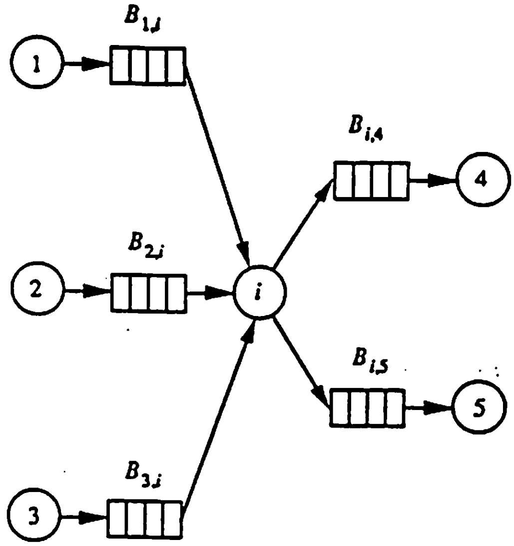


Figure 3: The original FJQN/B.

in that server $i$ is replaced by a subnetwork consisting of U-servers. We describe below how to construct $\mathcal{S}'$ for each of the 8 types of B-servers. In each case, it is easy to check that the FJQN/B $\mathcal{S}'$ obtained from $\mathcal{S}$ by transforming server $i$ behaves exactly as the original FJQN/B $\mathcal{S}$. A formal statement of equivalence and its proof will be given in the case that server $i$ is of type BAS/IL/IU. In the case of other server types, the formal statements of equivalence and their proofs are similar and are omitted.

The transformation will be illustrated by considering the example shown in Figure 3. This FJQN/B consists of 6 servers. Servers 1 to 5 are arbitrary servers and server $i$, is the B-server that will be transformed into a subnetwork of U-servers. Servers 1, 2 and 3 are sources while servers 4 and 5 are sinks. We have $p(i) = \{1,2,3\}$ and $s(i) = \{4,5\}$.

Since the transformation focuses on server $i$ and not on the remainder of the FJQN/B, we first characterize the overall transformation of $\mathcal{S}$ into $\mathcal{S}'$ without giving details regarding the transformation of server $i$. This overall characterization is given once, as it does not depend on the server type of $i$. Then we describe the detailed transformation of server $i$ for each of the B-server types.

**Overall transformation.** The equivalent FJQN/B $\mathcal{S}'$ is:

$$\begin{array}{lll} V' & = & V + V_a, \\ E' & = & E - \{(j,i), j \in p(i)\} - \{(i,k), k \in s(i)\} + E_a, \\ B'_{j,k} & = & B_{j,k}, \qquad j,k \in V, \quad j,k \neq i, \\ M'_{j,k} & = & M_{j,k}, \qquad j,k \in V, \quad j,k \neq i, \\ L'_{j,k} & = & L_{j,k}, \qquad j,k \in V, \quad k \neq i, \\ U'_{j,k} & = & U_{j,k}, \qquad j,k \in V, \quad j \neq i, \\ T'_j & = & T_j, \qquad j \in V, \quad j \neq i, \\ \sigma'_{j,n} & = & \sigma_{j,n}, \qquad j \in V, \quad j \neq i, \quad n \geq 1, \end{array}$$

where $V_a$ and $E_a$ are the servers and edges within the subnetwork that will represent server $i$ within $\mathcal{S}'$. These, as well as the buffer sizes and initial markings pertaining to the subnetwork, will now be described for each type of server $i$.

**Transformation of a BAS/IL/IU server.** The subnetwork transformation of server $i$ is defined as follows:

$$\begin{array}{lll} V_a & = & \{l_{i,j}, j \in p(i)\} + \{u_{i,k}, k \in s(i)\}, \\ E_a & = & \{(j, l_{i,j}), j \in p(i)\} + \{(l_{i,j}, i), j \in p(i)\} \\ & & +\{(u_{i,k}, k), k \in s(i)\} + \{(i, u_{i,k}), k \in s(i)\} + \{(l_{i,j}, u_{i,k}), j \in p(i), k \in s(i)\}, \end{array}$$

$$\begin{array}{llllll} B'_{j,l_{i,j}} = B_{j,i}, & M'_{j,l_{i,j}} = M_{j,i}, & j \in p(i), & \\ B'_{u_{i,k},k} = B_{i,k}, & M'_{u_{i,k},k} = M_{i,k}, & k \in s(i), & \\ B'_{l_{i,j},i} = 1, & M'_{l_{i,j},i} = L_{i,j}, & j \in p(i), & \\ B'_{i,u_{i,k}} = 1, & M'_{i,u_{i,k}} = U_{i,j}, & k \in s(i), & \\ B'_{l_{i,j},u_{i,k}} = 1, & M'_{l_{i,j},u_{i,k}} = L_{i,j} + U_{i,k}, & j \in p(i), & k \in s(i), \end{array}$$

$$\begin{array}{lll} \sigma'_{i,n} & = & \sigma_{i,n}, \quad n \geq 1, \\ \sigma'_{l_{i,j},n} & = & 0, \quad j \in p(i), \quad n \geq 1, \\ \sigma'_{u_{i,k},n} & = & 0, \quad k \in s(i), \quad n \geq 1. \end{array}$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 4 in the case where server $i$ is a BAS/IL/IU server. The shaded area is the part representing the behavior of server $i$. All the buffers within this shaded area have capacity one. The service activity of any server $l_{i,j}, j \in p(i)$, represents the loading of a job from buffer $(j,i)$ to server $i$. The service activity of any server $u_{i,k}, k \in s(i)$, represents the unloading of a job from server $i$ to buffer $(k,i)$. Buffers $(l_{i,j}, u_{i,k}), j \in p(i), k \in s(i)$, function to enforce that no new job is loaded onto the server before all the jobs produced during the previous service activity have been unloaded.

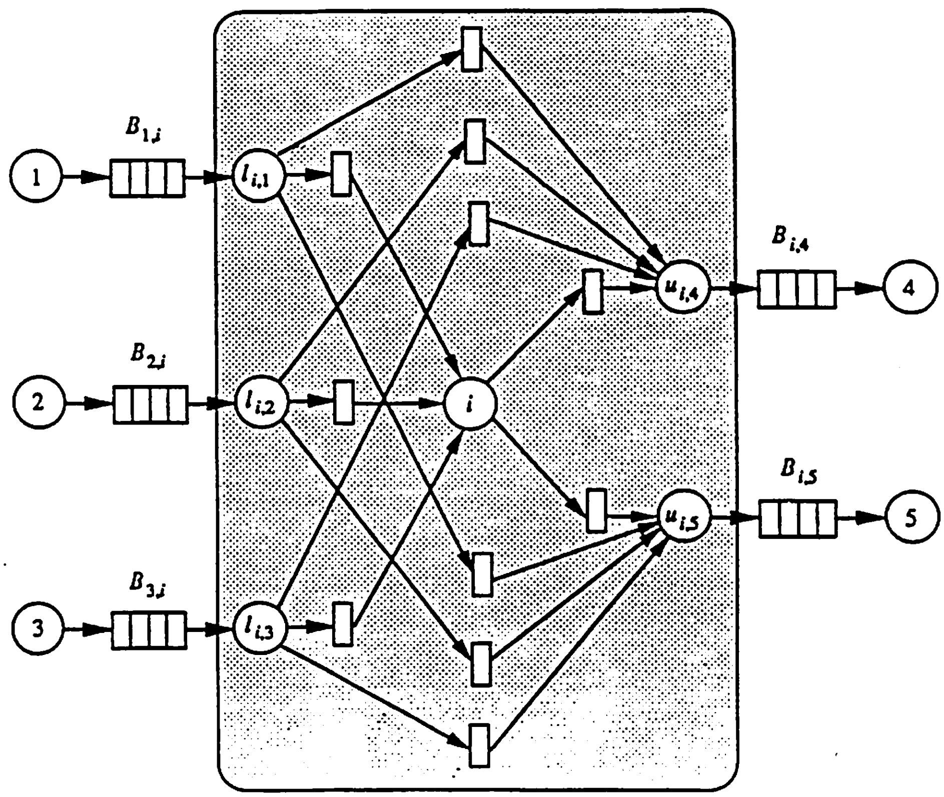


Figure 4: Illustration of the transformation of a BAS/IL/IU server.

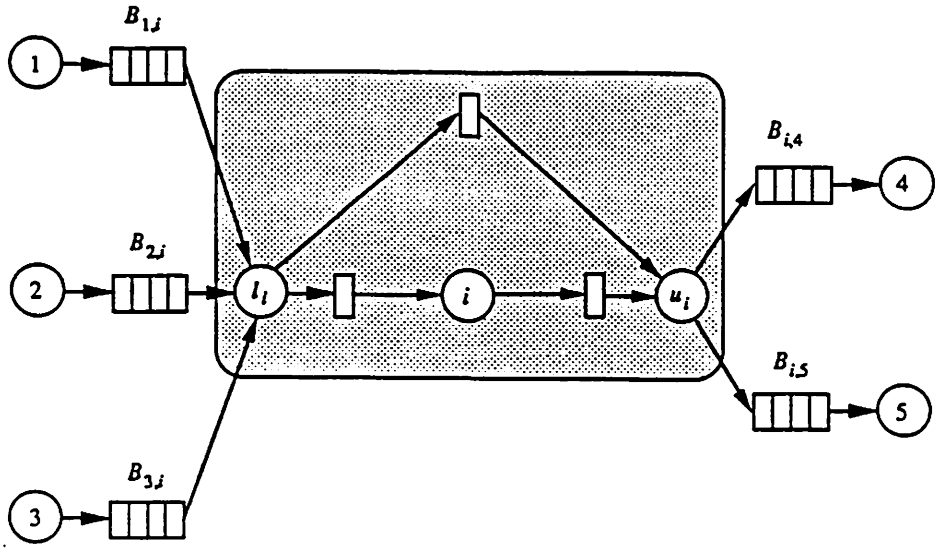


Figure 5: Illustration of the transformation of a BAS/SL/SU server.

**Transformation of a BAS/SL/SU server.** The subnetwork transformation of server $i$ is defined as follows:

$$\begin{aligned} V_a &= \{l_i\} + \{u_i\}, \\ E_a &= \{(j,l_i), j \in p(i)\} + \{(l_i,i)\} \\ &\quad + \{(u_i,k), k \in s(i)\} + \{(i,u_i)\} + \{(l_i,u_i)\}, \end{aligned}$$

$$\begin{array}{lllllll} B'_{j,l_i} & = & B_{j,i}, & M'_{j,l_i} & = & M_{j,i}, & j \in p(i), \\ B'_{u_i,k} & = & B_{i,k}, & M'_{u_i,k} & = & M_{i,k}, & k \in s(i), \\ B'_{l_i,i} & = & 1, & M'_{l_i,i} & = & L_i, & \\ B'_{i,u_i} & = & 1, & M'_{i,u_i} & = & U_i, & \\ B'_{l_i,u_i} & = & 1, & M'_{l_i,u_i} & = & L_i + U_i, & \end{array}$$

$$\begin{array}{llll} \sigma'_{i,n} & = & \sigma_{i,n}, & n \geq 1, \\ \sigma'_{l_i,n} & = & 0, & n \geq 1, \\ \sigma'_{u_i,n} & = & 0, & n \geq 1. \end{array}$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 5 in the case where server $i$ is a BAS/SL/SU server. The shaded area is the part representing the behavior of server $i$. All the buffers within this shaded area are of capacity one. The service activity of server $l_i$ represents the simultaneous loading of the jobs. The service activity of server $u_i$ represents the simultaneous unloading of the jobs. Buffer $(l_i, u_i)$, is placed to enforce that no new job is loaded onto the server before all of the jobs produced during the previous service activity have been unloaded.

**Transformation of a BAS/IL/SU server.** The subnetwork transformation of server $i$ is defined as follows:

$$\begin{aligned} V_a &= \{l_{i,j}, j \in p(i)\} + \{u_i\}, \\ E_a &= \{(j, l_{i,j}), j \in p(i)\} + \{(l_{i,j}, i), j \in p(i)\} \\ &\quad +\{(u_i, k), k \in s(i)\} + \{(i, u_i)\} + \{(l_{i,j}, u_i), j \in p(i)\}, \end{aligned}$$

$$\begin{array}{lllll} B'_{j,l_{i,j}} = B_{j,i}, & M'_{j,l_{i,j}} = M_{j,i}, & j \in p(i), \\ B'_{u_i,k} = B_{i,k}, & M'_{u_i,k} = M_{i,k}, & k \in s(i), \\ B'_{l_{i,j},i} = 1, & M'_{l_{i,j},i} = L_{i,j}, & j \in p(i), \\ B'_{i,u_i} = 1, & M'_{i,u_i} = U_i, & \\ B'_{l_{i,j},u_i} = 1, & M'_{l_{i,j},u_i} = L_{i,j} + U_i, & j \in p(i), \end{array}$$

$$\begin{aligned} \sigma'_{i,n} &= \sigma_{i,n}, \quad n \geq 1, \\ \sigma'_{l_{i,j},n} &= 0, \quad j \in p(i), \quad n \geq 1, \\ \sigma'_{u_i,n} &= 0, \quad n \geq 1. \end{aligned}$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 6 in the case where server $i$ is a BAS/IL/SU server. The shaded area is the part representing the behavior of server $i$. All of the buffers within this shaded area are of capacity one. The service activity of any server $l_{i,j}, j \in p(i)$, represents the loading of a job from buffer $(j,i)$. The service activity of any server $u_i$ represents the simultaneous unloading of the jobs. Buffers $(l_{i,j}, u_i), j \in p(i)$, are placed to enforce that no new job is loaded onto the server before all of the jobs produced during the previous service activity have been unloaded.

**Transformation of a BAS/SL/IU server.** The subnetwork transformation of server $i$ is defined as follows:

$$\begin{aligned} V_a &= \{l_i\} + \{u_{i,k}, k \in s(i)\}, \\ E_a &= \{(j, l_i), j \in p(i)\} + \{(l_i, i)\} \\ &\quad +\{(u_{i,k}, k), k \in s(i)\} + \{(i, u_{i,k}), k \in s(i)\} + \{(l_i, u_{i,k}), k \in s(i)\}, \end{aligned}$$

$$\begin{array}{lllll} B'_{j,l_i} = B_{j,i}, & M'_{j,l_i} = M_{j,i}, & j \in p(i), \\ B'_{u_{i,k},k} = B_{i,k}, & M'_{u_{i,k},k} = M_{i,k}, & k \in s(i), \\ B'_{l_i,i} = 1, & M'_{l_i,i} = L_i, & \\ B'_{i,u_{i,k}} = 1, & M'_{i,u_{i,k}} = U_{i,j}, & k \in s(i), \\ B'_{l_i,u_{i,k}} = 1, & M'_{l_i,u_{i,k}} = L_i + U_{i,k}, & k \in s(i), \end{array}$$

$$\begin{aligned} \sigma'_{i,n} &= \sigma_{i,n}, \quad n \geq 1, \\ \sigma'_{l_i,n} &= 0, \quad n \geq 1, \\ \sigma'_{u_{i,k},n} &= 0, \quad k \in s(i), \quad n \geq 1. \end{aligned}$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 7 in the case where server $i$ is a BAS/SL/IU server. The shaded area is the part representing the behavior

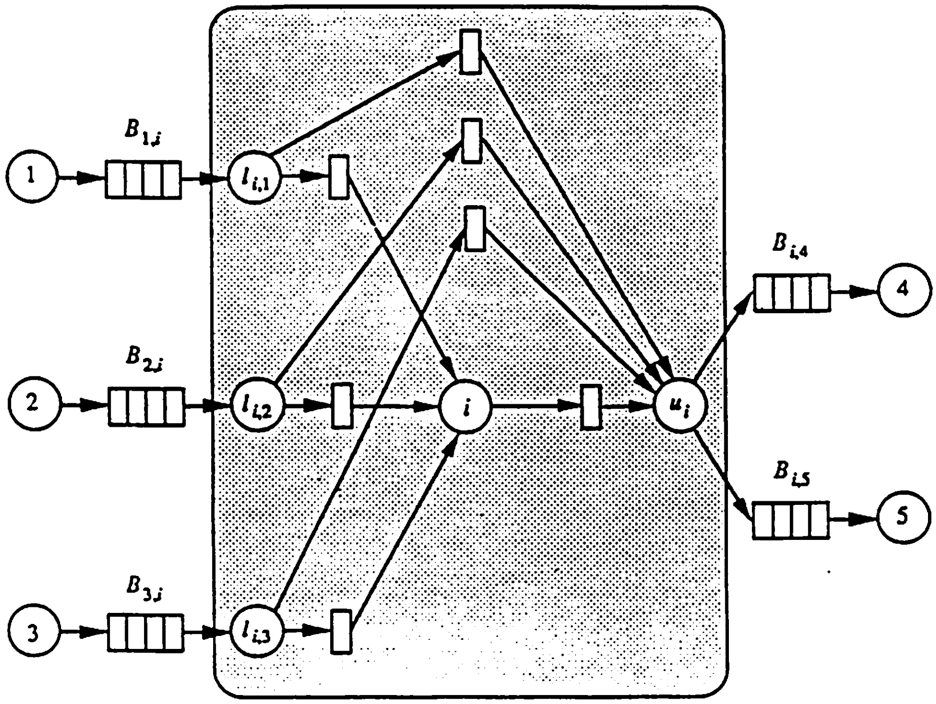


Figure 6: Illustration of the transformation of a BAS/IL/SU server.

of server $i$. All of the buffers within this shaded area are of capacity one. The service activity of server $l_i$ represents the simultaneous loading of the jobs. The service activity of any server $u_{i,k}, k \in s(i)$, represents the unloading of a job into buffer $(k, i)$. Buffers $(l_i, u_{i,k}), k \in s(i)$, are placed to enforce that no new job is loaded onto the server before all of the jobs produced during the previous service activity have been unloaded.

**Transformation of a BBS/IL server.** The subnetwork transformation of server $i$ is defined as follows:

$$V_a = \{l_{i,j}, j \in p(i)\},$$
$$E_a = \{(j, l_{i,j}), j \in p(i)\} + \{(l_{i,j}, i), j \in p(i)\}$$

$$B'_{j,l_{i,j}} = B_{j,i}, \quad M'_{j,l_{i,j}} = M_{j,i}, \quad j \in p(i),$$
$$B'_{l_{i,j},i} = 1, \quad M'_{l_{i,j},i} = L_{i,j}, \quad j \in p(i),$$

$$\sigma'_{i,n} = \sigma_{i,n}, \quad n \geq 1,$$
$$\sigma'_{l_{i,j},n} = 0, \quad j \in p(i), \quad n \geq 1.$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 8 in the case where server $i$ is a BBS/IL server. The shaded area is the part representing the behavior of

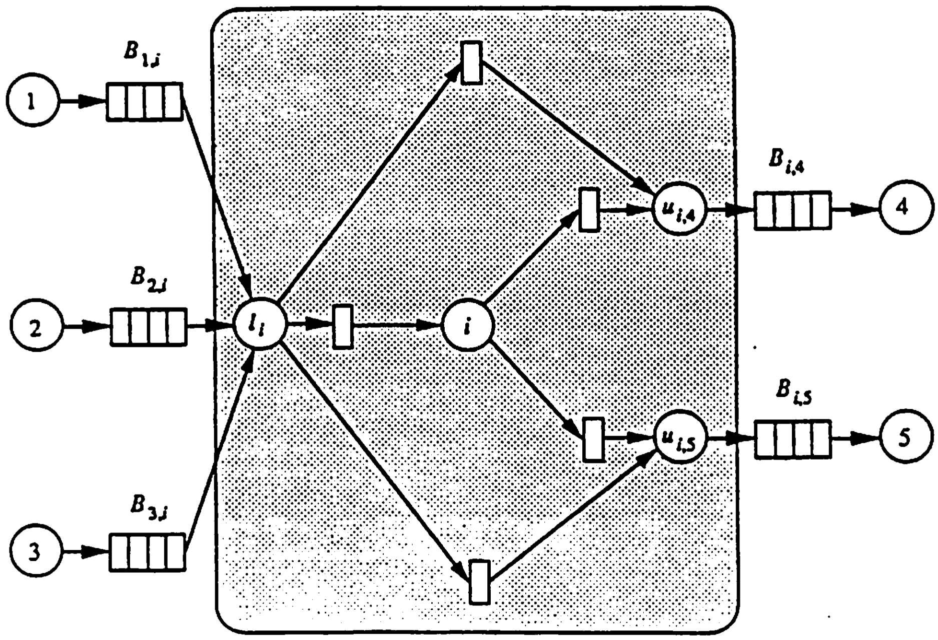


Figure 7: Illustration of the transformation of a BAS/SL/IU server.

server $i$. All of the buffers within this shaded area are of capacity one. The service activity of any server $l_{i,j}, j \in p(i)$, represents the loading of a job from buffer $(j, i)$.

**Remark.** Note that since every server $l_{i,j}, j \in p(i)$ has zero service time, the two buffers $(j, l_{i,j})$ and buffer $(l_{i,j}, i)$ can be equivalently replaced by a single buffer of capacity $B_{i,j} + 1$. In other words, the transformation of a BBS-IL server simply consists in increasing the capacity of all of its upstream buffers by one.

**Transformation of a BBS/SL server.** The subnetwork transformation of server $i$ is defined as follows:

$$\begin{aligned} V_a &= \{l_i\}, \\ E_a &= \{(j, l_i), j \in p(i)\} + \{(l_i, i)\} \end{aligned}$$

$$\begin{aligned} B'_{j,l_i} &= B_{j,i}, & M'_{j,l_i} &= M_{j,i}, \quad j \in p(i), \\ B'_{l_i,i} &= 1, & M'_{l_i,i} &= L_i, \end{aligned}$$

$$\begin{aligned} \sigma'_{i,n} &= \sigma_{i,n}, \quad n \geq 1, \\ \sigma'_{l_i,n} &= 0, \quad n \geq 1. \end{aligned}$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 9 in the case where server $i$ is a BBS/SL server. The shaded area is the part representing the behavior of server $i$. The buffer within this shaded area is of capacity one. The service activity of server $l_i$ represents the simultaneous loading of the jobs.

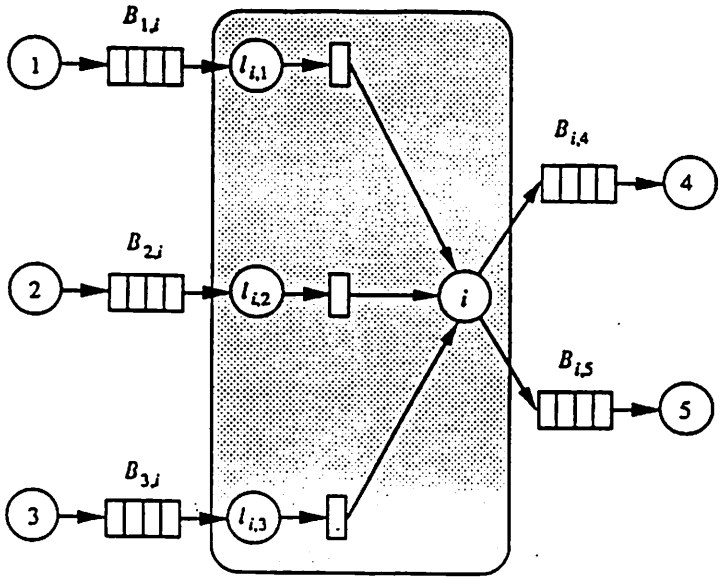


Figure 8: Illustration of the transformation of a BBS/IL server.

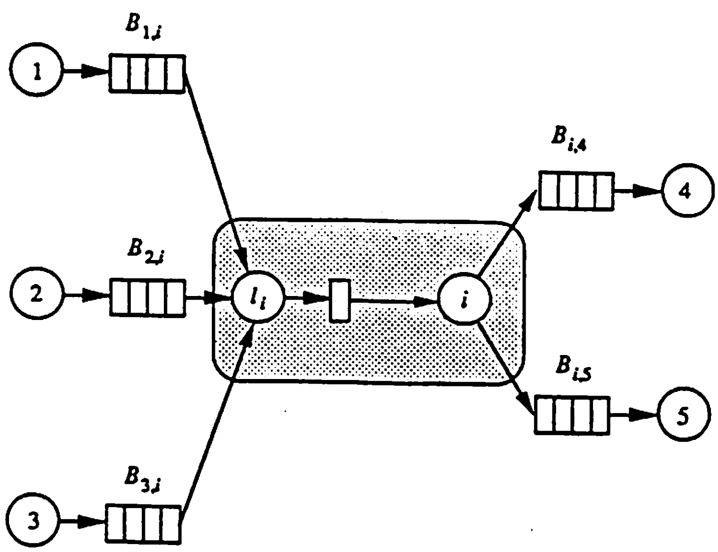


Figure 9: Illustration of the transformation of a BBS/SL server.

**Transformation of a BBS-CL/IL server.** The subnetwork transformation of server $i$ is defined as follows:

$$
\begin{aligned}
V_a &= \{l_{i,j}, j \in p(i)\} + \{c_{i,k}, k \in s(i)\}, \\
E_a &= \{(j, l_{i,j}), j \in p(i)\} + \{(l_{i,j}, i), j \in p(i)\} \\
&\quad + \{(c_{i,k}, k), k \in s(i)\} + \{(i, c_{i,k}), k \in s(i)\} + \{(l_{i,j}, c_{i,k}), j \in p(i), k \in s(i)\},
\end{aligned}
$$

$$
\begin{array}{llllll}
B'_{j,l_{i,j}} &= B_{j,i}, & M'_{j,l_{i,j}} &= M_{j,i}, & j \in p(i), \\
B'_{c_{i,k},k} &= B_{i,k} - 1, & M'_{c_{i,k},k} &= \min(M_{i,k}, B_{i,k} - 1), & k \in s(i), \\
B'_{l_{i,j},i} &= 1, & M'_{l_{i,j},i} &= L_{i,j}, & j \in p(i), \\
B'_{i,c_{i,k}} &= 1, & M'_{i,c_{i,k}} &= (M_{i,k} - (B_{i,k} - 1))^+, & k \in s(i), \\
B'_{l_{i,j},c_{i,k}} &= 1, & M'_{l_{i,j},c_{i,k}} &= L_{i,j} + (M_{i,k} - (B_{i,k} - 1))^+, & j \in p(i), k \in s(i),
\end{array}
$$

$$
\begin{aligned}
\sigma'_{i,n} &= \sigma_{i,n}, \quad n \geq 1, \\
\sigma'_{l_{i,j},n} &= 0, \quad j \in p(i), \quad n \geq 1, \\
\sigma'_{c_{i,k},n} &= 0, \quad k \in s(i), \quad n \geq 1.
\end{aligned}
$$

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 10 in the case where server $i$ is a BBS-CL/IL server. The shaded area is the part representing the behavior of server $i$. All of the buffers within this shaded area are of capacity one. The service activity of any server $l_{i,j}, j \in p(i)$, represents the loading of a job from buffer $(j, i)$. Any buffer $(i, k), k \in s(i)$ is split into two buffers, $(i, c_{i,k})$ and $(c_{i,k}, k)$, separated by a server, $c_{i,k}$. These buffers are of capacity $B_{i,k} - 1$ and 1, respectively. Server $c_{i,k}$ has zero service time. The service activity of any server $c_{i,k}, k \in s(i)$, represents the transfer of a job from the last position of buffer $(i, k)$ to the last but one position. Buffers $(l_{i,j}, c_{i,k}), j \in p(i), k \in s(i)$, each having capacity one, are placed to enforce that no new job is loaded onto the server until the last position of each buffer $(i, k), k \in s(i)$ is unoccupied.

**Transformation of a BBS-CL/SL server.** The subnetwork transformation of server $i$ is defined as follows:

$$
\begin{aligned}
V_a &= \{l_i\} + \{c_{i,k}, k \in s(i)\}, \\
E_a &= \{(j, l_i), j \in p(i)\} + \{(l_i, i)\} \\
&\quad + \{(c_{i,k}, k), k \in s(i)\} + \{(i, c_{i,k}), k \in s(i)\} + \{(l_i, c_{i,k}), k \in s(i)\},
\end{aligned}
$$

$$
\begin{array}{llllll}
B'_{j,l_i} &= B_{j,i}, & M'_{j,l_i} &= M_{j,i}, & j \in p(i), \\
B'_{c_{i,k},k} &= B_{i,k} - 1, & M'_{c_{i,k},k} &= \min(M_{i,k}, B_{i,k} - 1), & k \in s(i), \\
B'_{l_i,i} &= 1, & M'_{l_i,i} &= L_i, & \\
B'_{i,c_{i,k}} &= 1, & M'_{i,c_{i,k}} &= (M_{i,k} - (B_{i,k} - 1))^+, & k \in s(i), \\
B'_{l_i,c_{i,k}} &= 1, & M'_{l_i,c_{i,k}} &= L_i + (M_{i,k} - (B_{i,k} - 1))^+, & k \in s(i),
\end{array}
$$

$$
\begin{aligned}
\sigma'_{i,n} &= \sigma_{i,n}, \quad n \geq 1, \\
\sigma'_{l_i,n} &= 0, \quad n \geq 1, \\
\sigma'_{c_{i,k},n} &= 0, \quad k \in s(i), \quad n \geq 1.
\end{aligned}
$$

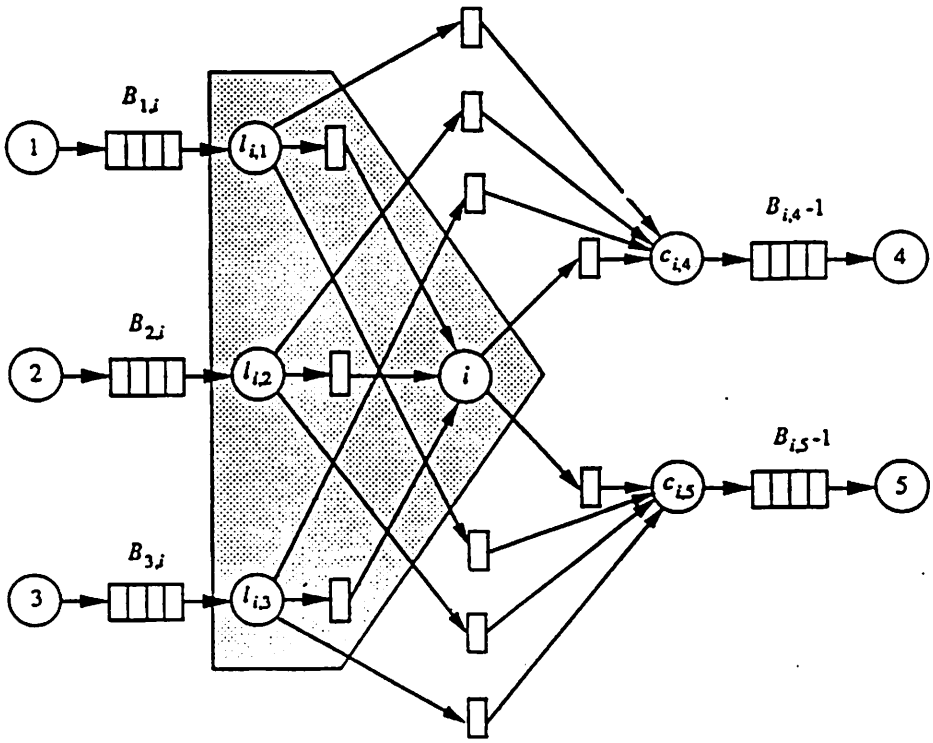


Figure 10: Illustration of the transformation of a BBS-CL/IL server.

The B-FJQN/B equivalent to the FJQN/B shown in Figure 3 is given in Figure 11 in the case where server $i$ is a BBS-CL/SL server. The shaded area is the part representing the behavior of server $i$. The buffer within this shaded area is of capacity one. The service activity of server $l_i$ represents the simultaneous loading of the jobs. Any buffer $(i, k), k \in s(i)$ is split into two buffers, $(i, c_{i,k})$ and $(c_{i,k}, k)$, separated by a server, $c_{i,k}$. These buffers are of capacity $B_{i,k} - 1$ and 1, respectively. Server $c_{i,k}$ has zero service time. The service activity of any server $c_{i,k}, k \in s(i)$, represents the transfer of a job from the last position of buffer $(i, k)$ to the last but one position. Buffers $(l_i, c_{i,k}), k \in s(i)$, each having capacity one, are placed to enforce that no new job is loaded onto the server until the last position of each buffer $(i, k), k \in s(i)$ is unoccupied.

In the remainder of this section, we make explicit what is meant by the equivalence of $\mathcal{S}$ and $\mathcal{S}'$ for the case that server $i$ is of type BAS/IL/IU. We introduce the following r.v.'s for each server $k$ in an arbitrary FJQN/B $\mathcal{S}$.

- $D_{k,n}$ - the $n$-th completion time at server $k$, $n \geq 1$.

In the case that server $k$ is a B-server, we have the following additional r.v.'s,

- $X_{k,j,n}$ - the time at which the $n$-th job from buffer $(j, k)$ is loaded into server $k$, $j \in p(k)$, $n \geq 1$,

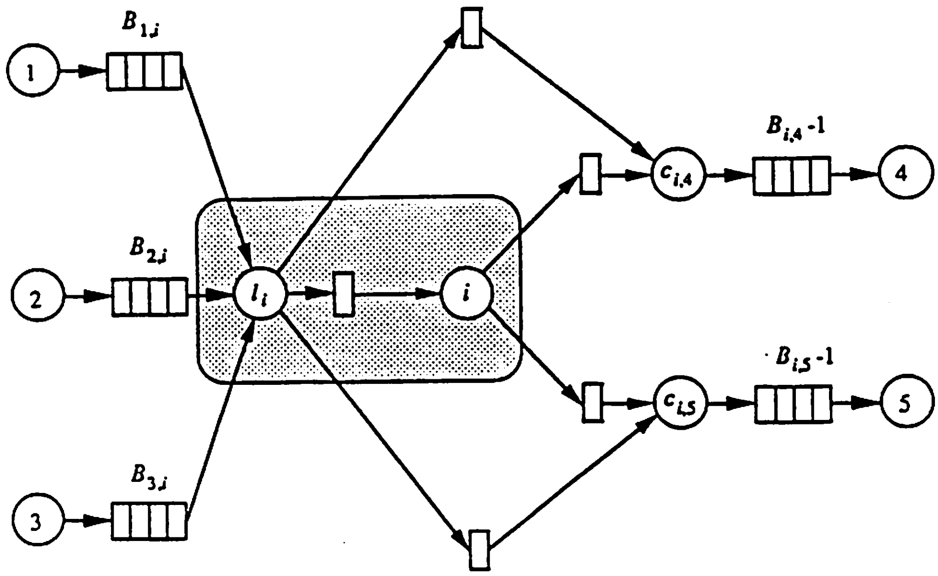


Figure 11: Illustration of the transformation of a BBS-CL/SL server.

- $Y_{k,j,n}$ - the time at which the $n$-th job is unloaded from server $k$ into buffer $(k, j)$ $j \in s(k)$, $n \geq 1$,

Note that for a B-server $k$ with an SL policy, $X_{k,j,n}$ does not depend on $j$. Similarly, $Y_{k,j,n}$ does not depend on $j$ in the case of a B-server $k$ with a SU policy. This is also true if server $k$ is of type BBS and, in addition, $Y_{k,j,n} = D_{k,n}$.

**Lemma 3.1** *Let $S$ be an arbitrary FJQN/B with $i$ being a type BAS/IL/IU server. Let $S'$ be the FJQN/B obtained from $S$ by applying the above transformation to server $i$. The behavior of $S'$ is equivalent to that of $S$ in the following sense:*

$$\begin{aligned}
X'_{k,j,n} &= X_{k,j,n}, \quad \forall j \in p(k); \quad k \in V; \quad k \neq i, \\
D'_{k,n} &= D_{k,n}, \quad \forall k \in V; \quad k \neq i, \\
Y'_{k,j,n} &= Y_{k,j,n}, \quad \forall j \in s(k); \quad k \in V; \quad k \neq i, \\
X'_{l_i,j,n} &= D_{i,j,n}, \quad \forall j \in p(i), \\
D'_{i,n} &= D_{i,n}, \\
Y'_{u_i,k,n} &= D_{i,k,n}, \quad \forall k \in s(i),
\end{aligned}$$

*for all $n \geq 1$.*

**Proof.** See the Appendix. ∎

We have stated the equivalence in terms of the times of loadings, completions and unloadings as it will be useful to us later in the paper. It should be clear that there is also an equivalence between markings in both networks as well.

An arbitrary FJQN/B $\mathcal{S}$ can be transformed into a B-FJQN/B through a series of independent transformations of the B-servers using the above procedure. Furthermore, the resulting network $\mathcal{S}'$ will not depend on the order the transformations are performed on the B-servers.

Due to Lemma 3.1 and the equivalences that hold for the transformations of the other B-server types, we get equivalence between a general FJQN/B $\mathcal{S}$ and its corresponding B-FJQN/B $\mathcal{S}^b$:

**Theorem 3.1** *Let $\mathcal{S}$ be an arbitrary FJQN/B. Let $\mathcal{S}^b$ be the B-FJQN/B obtained by transforming each B-server in $\mathcal{S}$ into a subnetwork consisting of U-servers according to the above described construction. Then*

$$D^b_{i,n} = D_{i,n}, \quad i \in V; n \geq 1.$$

**Remark.** Additional results can be made regarding the mapping of loading and unloading times in $\mathcal{S}$ and service completion times within $\mathcal{S}'$. However, only the results regarding the completion times of servers in $\mathcal{S}$ are required in the remainder of the paper.

## 4 Properties of FJQN/B's

In this section, we establish properties of FJQN/B's. These results are obtained by converting an arbitrary FJQN/B into an equivalent B-FJQN/B as described in Section 3.2, together with the properties of B-FJQN/B's presented in Sections 2.2 and 2.3. Most often, the properties of FJQN/B's follow in a straightforward manner from those of B-FJQN/B's. Special cases of some of the results presented here were obtained in earlier papers [1, 3, 5, 17, 18, 19, 20, 22, 25]. Most of these results pertain to tandem and closed tandem queueing networks, which are special cases of FJQN/B's.

For simplicity, and unless otherwise stated, we assume throughout this section that the initial condition of the FJQN/B's we consider is such that all of the B-servers are empty so that all of the jobs reside in the buffers, i.e., all of the components of $\boldsymbol{L}$ and $\boldsymbol{U}$ are zero. In this case, a FJQN/B $\mathcal{S}$ is simply characterized by $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$. This will allow us to keep the notation as simple as possible. However, all of the results that we present can be generalized to other initial conditions.

Before going further, we prove a sufficient condition under which a FJQN/B, $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$, is deadlock-free. It does not seem easy to state (in a simple form) a general necessary and sufficient condition under which a FJQN/B is deadlock free. However, it can always be checked whether or not a given FJQN/B is deadlock-free by verifying the conditions given in Theorem 2.1 on the equivalent B-FJQN/B. For an arbitrary FJQN/B, the following sufficient condition can be used as a "quick check" without transforming $\mathcal{S}$ into a B-FJQN/B. Let $I(M)$ be defined as in Section 2.2.

**Theorem 4.1** *A FJQN/B $\mathcal{S} = (V, E, B, M, T)$ with initial marking $M$ is deadlock-free if the following relation is satisfied,*

$$I_C^+(M) > 0 \text{ and } I_C^-(M) > 0,$$

*or equivalently*

$$0 < I_C^+(M) < B_C$$

*for all cycles $C$ in $\mathcal{S}$.*

**Proof.** Let $\mathcal{S}^b = (V^b, E^b, B^b, M^b, T^b)$ be the corresponding B-FJQN/B performing the appropriate transformation on each of the B-servers of $\mathcal{S}$. Let $C$ be an elementary cycle of $\mathcal{S}^b$, $I_C^{+b}$ and $I_C^{-b}$ be the corresponding quantities defined on $\mathcal{S}^b$. Let $V_C$ be the set of servers in cycle $C$ and also in $V$.

If $V_C = \{i\} \subset V$, then $C$ is composed of servers $i$, $l_{i,j}$ for some $j \in p(i)$, and $u_{i,k}$ (or $c_{i,k}$) for some $k \in s(i)$. It is readily checked (cf. Figures 4—11) that in such a case, we have

$$I_C^{+b}(M^b) > 0 \text{ and } I_C^{-b}(M^b) > 0.$$

Assume now that $V_C$ is not a singleton. It then follows that for all $i \in V_C$, the servers $i$, $l_{i,j}$ (with $j \in p(i)$), and $u_{i,k}$ or $c_{i,k}$ (with $k \in s(i)$) appear successively in the chain $C$. Let $C_i$ be the subchain of $C$ composed of these servers (i.e., $i$, $l_{i,j}$ and $u_{i,k}$ or $c_{i,k}$) and of the buffers connecting these servers. Note that if $i$ is a U-server in $\mathcal{S}$, then $C_i$ is a singleton $C_i = \{i\}$. Let $C'$ be the chain obtained from $C$ by replacing subchains $C_i$ with $i$ for all $i \in V_C$. Replace further the edges $(i, j)$ of $C'$ by the the edges of $E$ in $\mathcal{S}$ with buffer size $B_{i,j}$ and initial marking $M_{i,j}$. Thus, the resulting chain, denoted by $C^\circ$, is a chain of the original network $\mathcal{S}$.

Let $\underline{C}_i$ be the collection of subchains obtained from $C_i$ by deleting servers $c_{i,k}$ (with $k \in s(i)$) and their neighboring buffers in $C_i$. It is clear that

$$I_{\underline{C}_i}^{+b}(M^b) \geq 0, \qquad I_{\underline{C}_i}^{-b}(M^b) \geq 0, \qquad i \in V_C.$$

It then follows that

$$\begin{aligned} I_C^{+b}(M^b) &= I_{C^\circ}^+(M) + \sum_{i \in V_C} I_{\underline{C}_i}^{+b}(M^b) > 0 \\ I_C^{-b}(M^b) &= I_{C^\circ}^-(M) + \sum_{i \in V_C} I_{\underline{C}_i}^{-b}(M^b) > 0. \end{aligned}$$

Therefore, according to Theorem 2.1, under the condition that for all cycles $C$ in $\mathcal{S}$.

$$I_C^+(M) > 0 \text{ and } I_C^-(M) > 0,$$

the B-FJQN/B $\mathcal{S}^b$ is deadlock-free, so is the equivalent FJQN/B $\mathcal{S}$. ∎

## 4.1 Throughput

Consider a FJQN/B $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$. Let $D_{i,n}$ and $\theta_i(\mathcal{S})$ be defined as in sections 3.2 and 2.2 respectively. The following results follow from Theorems 2.2 and 2.3.

**Corollary 4.1** *Let $\mathcal{S}$ be a FJQN/B. Then, under Assumption A1, the asymptotic throughputs exist and moreover we have:*

$$\theta_i(\mathcal{S}) = \theta(\mathcal{S}), \quad \forall i \in V.$$

Again, $\theta(\mathcal{S})$ will be referred to as the throughput of $\mathcal{S}$.

**Corollary 4.2** *Let $\boldsymbol{M}^1$ and $\boldsymbol{M}^2$ be two initial markings of a FJQN/B $\mathcal{S}$. Under Assumptions A1 and A2, if $\boldsymbol{M}^1 \sim \boldsymbol{M}^2$, we have:*

$$\theta(\boldsymbol{M}^1) = \theta(\boldsymbol{M}^2).$$

## 4.2 Monotonicity and Concavity

All of the results presented in Section 2.3 for B-FJQN/B's obviously hold for any FJQN/B. The proofs are obtained by a straightforward application of the results of Section 2.3 on the equivalent B-FJQN/B's. Therefore, we get the following monotonicity properties of the throughput of FJQN/B's.

**Corollary 4.3** *Consider a FJQN/B $\mathcal{S}$ with two different buffer capacity vectors $\boldsymbol{B}$ and $\boldsymbol{B} + \boldsymbol{k}$. Then, under Assumption A1, we have:*

$$\theta(\boldsymbol{B} + \boldsymbol{k}) \geq \theta(\boldsymbol{B}), \ \forall \boldsymbol{k} \geq 0.$$

Special cases of the above result were previously obtained by Shanthikumar and Yao [22] in the case of a closed tandem queueing network and by Adan and Van Der Waal [1] in the case of an assembly network.

**Corollary 4.4** *Consider a FJQN/B $\mathcal{S}$ with two different buffer capacity vectors and initial marking vectors $(\boldsymbol{B},\boldsymbol{M})$ and $(\boldsymbol{B}+\boldsymbol{k},\boldsymbol{M}+\boldsymbol{k})$. Then, under Assumption A1, we have:*

$$\theta(\boldsymbol{B}+\boldsymbol{k},\boldsymbol{M}+\boldsymbol{k}) \geq \theta(\boldsymbol{B},\boldsymbol{M}),\ \forall \boldsymbol{k} \geq \boldsymbol{0}.$$

**Corollary 4.5** *Let $\mathcal{S}$ be a FJQN/B and $\tilde{\mathcal{S}}$ be a subnetwork of $\mathcal{S}$. Then, under Assumption A1, we have:*

$$\theta(\tilde{\mathcal{S}}) \geq \theta(\mathcal{S}),$$

**Corollary 4.6** *Let $\mathcal{S}$ be a FJQN/B and $\mathcal{S}^e$ be an expanded network of $\mathcal{S}$ with respect to server $i$. Then, under Assumption A1, we have:*

$$\theta(\mathcal{S}^e) \geq \theta(\mathcal{S}),$$

The following corollary follows by applying Theorem 2.6 to the equivalent B-FJQN/B.

**Corollary 4.7** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B. Then, under Assumption A4, $\theta(\mathcal{S})$ is a concave function of $\boldsymbol{B}$ and $\boldsymbol{M}$.*

Special cases of the above result were previously obtained by Anantharam and Tsoucas [3] and Meester and Shanthikumar [18] in the case of tandem queueing networks and by Shanthikumar and Yao [22] in the case of a closed tandem queueing networks, all assuming exponential service times.

## 4.3 Comparison of Various Operating Mechanisms

Consider a particular B-server, say $i$, of a FJQN/B $\mathcal{S}$. We are interested in comparing the behavior of $\mathcal{S}$ for different operating mechanisms of server $i$. We first establish an equivalence between the BBS-CL and BAS blocking mechanisms. Let $\boldsymbol{B}$ be a buffer capacity vector and let $\boldsymbol{B}^+$ be the buffer capacity vector obtained from $\boldsymbol{B}$ as:

$$\begin{aligned} B^+_{i,k} &= B_{i,k}+1, \quad \forall k \in s(i), \\ B^+_{j,k} &= B_{j,k}, \qquad \forall j,k \in V, \quad j \neq i. \end{aligned}$$

**Theorem 4.2** *Let $\mathcal{S}^1$ be a FJQN/B with buffer capacity vector $\boldsymbol{B}^1 = \boldsymbol{B}$ in which server $i$ is a BAS/IL/IU (resp. BAS/SL/IU) server. Let $\mathcal{S}^2$ be a FJQN/B identical to $\mathcal{S}^1$ except that it has buffer capacity vector $\boldsymbol{B}^2 = \boldsymbol{B}^+$ and server $i$ is a BBS-CL/IL (resp. BBS-CL/SL) server. Then, $\mathcal{S}^2$ has exactly the same behavior as $\mathcal{S}^1$ in the sense that they have the same equivalent B-FJQN/B.*

**Proof.** Consider, for instance, the case where server $i$ is a BAS/IL/IU server in $\mathcal{S}^1$ and a BBS-CL/IL server in $\mathcal{S}^2$. (The proof for the case where server $i$ is a BAS/SL/IU server in $\mathcal{S}^1$ and a BBS-CL/SL server in $\mathcal{S}^2$ is similar and therefore omitted.) Consider the two B-FJQN/B's, $\mathcal{S}^{1b}$ and $\mathcal{S}^{2b}$, equivalent to $\mathcal{S}^1$ and $\mathcal{S}^2$, respectively. Let us concentrate on the transformation of server $i$ since the rest of the networks is identical. Let us apply the transformations described in Section 3.2 to server $i$ in both networks. It turns out that the two B-FJQN/B's obtained are identical. Indeed, server $l_{i,j}$ in $\mathcal{S}^{2b}$ plays the same role as server $l_{i,j}$ in $\mathcal{S}^{1b}$, $\forall j \in p(i)$, and server $c_{i,k}$ in $\mathcal{S}^{2b}$ plays the same role as server $u_{i,k}$ in $\mathcal{S}^{1b}$, $\forall k \in s(i)$. Moreover, the capacity of buffer $(c_{i,k}, k)$ in $\mathcal{S}^{2b}$ is $B^2_{i,k} - 1$ and the capacity of buffer $(u_{i,k}, k)$ in $\mathcal{S}^{1b}$ is $B^1_{i,k}$ and are both equal to $B_{i,k}$, $\forall k \in s(i)$. It is easy to check that the rest of the two B-FJQN/B's are identical (see Figures 4 and 10). ∎

Thus, the BBS-CL blocking mechanism can be viewed as equivalent to the BAS blocking mechanism with a reduction in the buffer capacity of one in each of the downstream buffers. As a result of this equivalence, we no longer deal with servers having BBS-CL blocking mechanisms. This equivalence between BAS and BBS-CL mechanisms was first reported by Onvural and Perros [20] in the special case of tandem queueing networks.

The following result compares the loading and unloading policies of a server with a BAS blocking mechanism.

**Theorem 4.3** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B. Let $\theta(BAS/IL/IU)$, $\theta(BAS/SL/SU)$, $\theta(BAS/IL/SU)$ and $\theta(BAS/SL/IU)$, denote the throughputs of this network when server $i$ is a BAS/IL/IU server, a BAS/SL/SU server, a BAS/IL/SU server and a BAS/SL/IU server, respectively. Then, under Assumption A1, we have:*

$$\theta(BAS/IL/IU) \geq \theta(BAS/IL/SU) \geq \theta(BAS/SL/SU),$$

*and*

$$\theta(BAS/IL/IU) \geq \theta(BAS/SL/IU) \geq \theta(BAS/SL/SU).$$

**Proof.** Let us prove that $\theta(BAS/IL/SU) \geq \theta(BAS/SL/SU)$. (The proofs of the other results are easily obtained using similar arguments and are therefore omitted.) Let $\mathcal{S}^1$ (resp. $\mathcal{S}^2$) denote the FJQN/B $\mathcal{S}$ in the case where server $i$ is a BAS/IL/SU (resp. BAS/SL/SU) server. Let $\mathcal{S}^{1b}$ (resp. $\mathcal{S}^{2b}$) be the B-FJQN/B's equivalent to $\mathcal{S}^1$ and $\mathcal{S}^2$, respectively (see Figures 6 and 5). Consider the B-FJQN/B, say $\mathcal{S}^m$, obtained from $\mathcal{S}^{1b}$ as follows. For every $j \in p(i)$, add a buffer in between every server $k \in p(i), k \neq j$ and server $l_{i,j}$ with the same buffer capacity and initial marking as the buffer already existing in between server $j$ and server $l_{i,j}$, that is: $B_{j,i}$ and $M_{j,i}$. It then appears that all of the servers $l_{i,j}, j \in p(i)$, in $\mathcal{S}^m$ are connected exactly in the same way to the rest of the network and, as a result, behave identically. Moreover, they behave exactly in the same way as server $l_i$ in $\mathcal{S}^{2b}$. They can actually be viewed as $|p(i)|$ replications of server $l_i$ in $\mathcal{S}^{2b}$. In other words, the addition of the buffers to the B-FJQN/B $\mathcal{S}^{1b}$ has transformed the independent loading policy into a simultaneous loading policy. As a result, we have $\theta(\mathcal{S}^m) = \theta(\mathcal{S}^{2b})$. Now, $\mathcal{S}^{1b}$ is a subnetwork of $\mathcal{S}^m$ and thus, from Corollary 2.3, we get $\theta(\mathcal{S}^{1b}) \geq \theta(\mathcal{S}^m)$. As a result, we have $\theta(\mathcal{S}^{1b}) \geq \theta(\mathcal{S}^{2b})$ and thus $\theta(BAS/IL/SU) \geq \theta(BAS/SL/SU)$. ∎

This theorem states that an independent loading (resp. unloading) policy achieves a higher throughput than a simultaneous loading (resp. unloading) policy. The result pertaining to the ordering of the loading policies was stated (however not proved) in [5] in the case of an assembly network. (In that paper, the independent and simultaneous loading policies were called push and pull modes).

The following result compares the performance of a server using the BBS blocking mechanism under different loading policies.

**Theorem 4.4** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B. Let $\theta(BBS/IL)$ and $\theta(BBS/SL)$ denote the throughputs of this network when server $i$ is a BBS/IL server and a BBS/SL server, respectively. Then, under Assumption A1, we have:*

$$\theta(BBS/IL) \geq \theta(BBS/SL).$$

**Proof.** The proof is similar to that of Theorem 4.3 and is therefore omitted.

The last result is concerned with the comparison of BAS and BBS blocking mechanisms.

**Theorem 4.5** *Consider a FJQN/B $\mathcal{S}$ in which server $i$ may operate under different operating mechanisms. Then, under Assumption A1, we have:*

$$\theta(BBS/IL, \boldsymbol{B}) \leq \theta(BAS/IL/SU, \boldsymbol{B}) \leq \theta(BAS/IL/IU, \boldsymbol{B}) \leq \theta(BBS/IL, \boldsymbol{B}^+),$$

*and*

$$\theta(BBS/SL, \boldsymbol{B}) \leq \theta(BAS/SL/SU, \boldsymbol{B}) \leq \theta(BAS/SL/IU, \boldsymbol{B}) \leq \theta(BBS/SL, \boldsymbol{B}^+).$$

**Proof.** Let us prove the first assertion. (The proof of the second assertion is similar and is thus omitted.) From Theorem 4.3, we already know that $\theta(BAS/IL/SU, \boldsymbol{B}) \leq \theta(BAS/IL/IU, \boldsymbol{B})$. Let us first prove that $\theta(BBS/IL, \boldsymbol{B}) \leq \theta(BAS/IL/SU, \boldsymbol{B})$. Let $\mathcal{S}^1$ (resp. $\mathcal{S}^2$) denote the FJQN/B with buffer capacity vector $\boldsymbol{B}$ and in which server $i$ is a BBS/IL (resp. BAS/IL/SU) server and let $\mathcal{S}^{1b}$ (resp. $\mathcal{S}^{2b}$) denote its equivalent B-FJQN/B. Then, it follows from the transformations defined in Section 3.2 (see Figures 6 and 8) that $\mathcal{S}^{2b}$ is an expanded network of $\mathcal{S}^{1b}$ (see Definition 2.4). Indeed, server $u_i$ plays the role of server $i_0$ and the buffer in between servers $i$ and $i_0$ has capacity one. As a result, from Corollary 2.4, we get that $\theta(\mathcal{S}^{1b}) \leq \theta(\mathcal{S}^{2b})$ and thus $\theta(BBS/IL, \boldsymbol{B}) \leq \theta(BAS/IL/SU, \boldsymbol{B})$. Let us now prove that $\theta(BAS/IL/IU, \boldsymbol{B}) \leq \theta(BBS/IL, \boldsymbol{B}^+)$. Let $\mathcal{S}^3$ (resp. $\mathcal{S}^4$) denote the FJQN/B with buffer capacity vector $\boldsymbol{B}$ (resp. $\boldsymbol{B}^+$) and in which server $i$ is a BAS/IL/IU (resp. BBS/IL) server and let $\mathcal{S}^{3b}$ (resp. $\mathcal{S}^{4b}$) denote its equivalent B-FJQN/B (cf. Figures 4 and 8). Let $\mathcal{S}^m$ be the B-FJQN/B obtained from $\mathcal{S}^{3b}$ by deleting all of the buffers $(l_{i,j}, u_{i,k}), j \in p(i), k \in s(i)$. It follows from Corollary 2.3 that $\theta(\mathcal{S}^m) \geq \theta(\mathcal{S}^{3b})$. Now, for every $k \in s(i)$ in $\mathcal{S}^m$, it appears that servers $i$ and $k$ are connected by a series of two buffers, buffer $(i, u_{i,k})$ with capacity one and buffer $(u_{i,k}, k)$ with capacity $B_{i,k}$, separated by a server, $u_{i,k}$, with zero service time. As a result, this series of two buffers can equivalently be replaced by a single buffer of capacity $B_{i,k} + 1$. Then, it is easy to see that, after this transformation, $\mathcal{S}^m$ is exactly identical to $\mathcal{S}^{4b}$ and therefore $\theta(\mathcal{S}^m) = \theta(\mathcal{S}^{4b})$. By combining the above two statements, we get $\theta(\mathcal{S}^{3b}) \leq \theta(\mathcal{S}^{4b})$ and as a result $\theta(BAS/IL/IU, \boldsymbol{B}) \leq \theta(BBS/IL, \boldsymbol{B}^+)$. ∎

The above result implies that the throughput of a FJQN/B with BAS servers and buffer capacity vector $\boldsymbol{B}$ can be bounded from below and from above by the throughputs of two FJQN/B's with BBS servers and buffer capacity vectors $\boldsymbol{B}$ and $\boldsymbol{B} + \boldsymbol{1}$, respectively, where $\boldsymbol{1}$ denote the unity vector. Correspondingly, the result also implies that the throughput of a FJQN/B with BBS servers and buffer capacity vector $\boldsymbol{B}$ can be bounded from below and from above by the throughputs of two FJQN/B's with BAS servers and buffer capacity vectors $\boldsymbol{B} - \boldsymbol{1}$ and $\boldsymbol{B}$, respectively.

## 4.4 Reversibility Properties

In this section, we establish reversibility properties of FJQN/B's.

**Definition 4.1** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B. The FJQN/B $\mathcal{S}^r = (V^r, E^r, \boldsymbol{B}^r, \boldsymbol{M}^r, \boldsymbol{T}^r)$ is the reverse of $\mathcal{S}$ if:*

$$\begin{aligned}
V^r &= V, \\
E^r &= \{(i,j) | (j,i) \in E\}, \\
B^r_{i,j} &= B_{j,i}, \qquad (j,i) \in E, \\
M^r_{i,j} &= M_{j,i}, \qquad (j,i) \in E, \\
\boldsymbol{T}^r &= \boldsymbol{T}.
\end{aligned}$$

Observe that $\mathcal{S}^r$ differs from $\mathcal{S}$ in that all of the buffers are reversed. The nodes remain the same type as in the original FJQN/B.

**Theorem 4.6** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B in which all servers are either BAS/IL/IU or BAS/SL/SU B-servers, or U servers. Let $\mathcal{S}^r = (V^r, E^r, \boldsymbol{B}^r, \boldsymbol{M}^r, \boldsymbol{T}^r)$ be the reverse of $\mathcal{S}$. If $\mathcal{S}$ and $\mathcal{S}^r$ have the same (joint distribution of the) sequences of service times, then under Assumptions A1 and A3, we have:*

$$\theta(\mathcal{S}^r) = \theta(\mathcal{S}). \tag{4.1}$$

**Proof.** It is easy to check from the transformations described in Section 3.2 that the equivalent B-FJQN/B of $\mathcal{S}^r$ is the reverse of the equivalent B-FJQN/B of $\mathcal{S}$. The proof then follows from Theorem 2.4. ∎

Special cases of the above result were previously obtained by Yamazaki and Sakasegawa [24] and Muth [19] in the case of tandem queueing networks and by Liu [17] in the case of closed tandem queueing networks.

**Remark:** There exists an alternate definition for the reverse FJQN/B which only applies to FJQN/B's consisting of U-servers and/or BAS type B-servers. Within the reverse network, a BAS type B-server is given a loading mechanism and an unloading mechanism corresponding respectively to the unloading mechanism and loading mechanism for that server in the original network (e.g., a BAS/IL/SU server in the original network is transformed into a BAS/SL/IU server in the reverse network). Theorem 4.6 can be proven for this alternate form of a reverse FJQN/B as well, without any restrictions on the loading and unloading mechanisms associated with BAS type B-servers.

Next, we establish a reversibility property of FJQN/B's with BBS/IL servers. On the other hand, there is no reversibility property for FJQN/B's with BBS/SL servers. The following lemma will be useful.

**Lemma 4.1** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B in which all servers are BBS/IL servers. Then, the B-FJQN/B $\mathcal{S}^b = (V, E, \boldsymbol{B} + \boldsymbol{1}, \boldsymbol{M})$ is equivalent to $\mathcal{S}$.*

**Proof.** Consider the transformation of any server $i \in V$ of $\mathcal{S}$ as defined in Section 3.2. Each buffer $(j, i), j \in p(i)$, is replaced by a series of two buffers: buffer $(j, l_{i,j})$ with capacity $B_{j,i}$ and buffer $(l_{i,j}, i)$ with capacity one, separated by server $l_{i,j}$. Since $l_{i,j}$ has zero service time, this series of two buffers can be replaced by a single buffer $(j, i)$ with capacity equal to the sum of the capacity of the two buffers, that is $B_{j,i} + 1$. Thus, the transformation of a BBS-IL server can simply be obtained by increasing the capacity of each of its downstream buffers by one, and replacing it by a U-server. ∎

**Theorem 4.7** *Let $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ be a FJQN/B in which all servers are BBS/IL servers. Let $\mathcal{S}^r = (V^r, E^r, \boldsymbol{B}^r, \boldsymbol{M}^r, \boldsymbol{T}^r)$ be the reverse of $\mathcal{S}$. If $\mathcal{S}$ and $\mathcal{S}^r$ have the same (joint distribution* *of the) sequences of service times, then under Assumptions A1 and A3, we have:*

$$\theta(\mathcal{S}^r) = \theta(\mathcal{S}). \tag{4.2}$$

**Proof.** The proof follows simply from Lemma 4.1 and Theorem 2.4. ∎

## 4.5 Symmetry Properties

In this section, we investigate whether or not a symmetry property similar to that of B-FJQN/B's (see Theorem 2.5 in Section 2.2) can be established for FJQN/B's. We first establish the following result pertaining to FJQN/B's with BBS servers.

**Theorem 4.8** *Consider a FJQN/B $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T}, \boldsymbol{L})$ in which all servers are BBS servers (either BBS/IL or BBS/SL servers). Consider the FJQN/B $\mathcal{S}^s(V, E, \boldsymbol{B}, \boldsymbol{M}^s, \boldsymbol{T}, \boldsymbol{L}^s)$ which is identical to $\mathcal{S}$ except that: $\boldsymbol{M}^s = \boldsymbol{B} - \boldsymbol{M}$ and $\boldsymbol{L}^s = 1 - \boldsymbol{L}$. Then, under Assumptions A1 and A3, we have:*

$$\theta(\mathcal{S}^s) = \theta(\mathcal{S}) \tag{4.3}$$

**Proof.** The proof simply follows by applying Theorem 2.5 to the equivalent B-FJQN/B's of $\mathcal{S}$ and $\mathcal{S}^s$. ∎

The following corollary follows from Theorem 4.8 and Corollary 4.2.

**Corollary 4.8** *Consider a FJQN/B $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M}, \boldsymbol{T})$ in which all servers are BBS servers (either BBS/IL or BBS/SL servers) with empty initial loading. Assume that $\boldsymbol{M} \geq \boldsymbol{1}$. Let $\boldsymbol{M}^s$ be the initial marking defined as: $\boldsymbol{M}^s = \boldsymbol{B} - \boldsymbol{M} + 1$. Then, under Assumptions A1 and A3, we have that for $\mathcal{S}^s = (V, E, \boldsymbol{B}, \boldsymbol{M}^s, \boldsymbol{T})$,*

$$\theta(\mathcal{S}^s) = \theta(\mathcal{S}) \tag{4.4}$$

On the other hand, there is no symmetry property for FJQN/B's in which servers have BAS blocking mechanisms. Let us briefly explain why. Consider a simple case of a FJQN/B $\mathcal{S} = (V, E, \boldsymbol{B}, \boldsymbol{M})$ in which server $i$ is, for instance, a BAS/SL/SU server. Again, assume that the initial condition of server $i$ is given by $L_i = U_i = 0$. Let $\mathcal{S}^b$ be the B-FJQN/B equivalent to $\mathcal{S}$. Its initial marking is such that: $M^b_{l_i,i} = M^b_{i,u_i} = M^b_{l_i,u_i} = 0$. Theorem 2.5 can be applied to $\mathcal{S}^b$. Let $\boldsymbol{M}^s$ denote the symmetrical marking of $\boldsymbol{M}^b$, i.e., $\boldsymbol{M}^s = \boldsymbol{B}^b - \boldsymbol{M}^b$, and let $\mathcal{S}^s$ denote the resulting B-FJQN/B. In particular, we have: $M^s_{l_i,i} = M^s_{i,u_i} = M^s_{l_i,u_i} = 1$. Unfortunately, the $\mathcal{S}^s$ cannot be interpreted as the equivalent B-FJQN/B of a FJQN/B where server $i$ has BAS/SL/SU operating mechanism. Actually, it is the equivalent B-FJQN/B of a FJQN/B where server $i$ operates as follows: after completion of a service activity, the jobs cannot be unloaded before new jobs have been loaded onto the server.

# 5 Applications to Manufacturing Systems

The purpose of this section is to illustrate the usefulness of the results derived in this paper for various problems encountered in the design and operation of manufacturing systems.

## 5.1 Closed-Loop Production Lines

Consider a production line consisting of $L$ machines in series, $M_1$, $M_2$,..., $M_L$, separated by buffers of finite capacity. In order to be processed by the different machines of the line, parts need to be fixed onto pallets. Each part is first loaded onto a pallet and is then carried by the pallet during its sojourn in the production line. When the last operation has been performed, the part is unloaded and the pallet is available to carry a new part. The (feedback) buffer that contains pallets available also has a finite capacity so that the last machine may be blocked if this buffer is full. The production capacity of such a closed-loop production line is defined as the throughput of this system, assuming that raw parts are always available at the input of the production line. An issue of high interest is to determine the optimal number of pallets that maximize this production capacity.

This closed-loop production line can be modeled as a closed tandem queueing network having $L$ servers. Servers 1 up to $L$ model machines $M_1$ up to $M_L$. The capacity of buffer $(i, i+1)$, $B_{i,i+1}$, is equal to the maximum number of parts that can be stored in between machines $M_i$ and $M_{i+1}$, $i = 1, ..., L-1$. The capacity of buffer $(L, 1)$, $B_{L,1}$, is equal to the maximum number of pallets that can be stored in the feedback buffer. Let $C$ denote the total storage capacity of the system including the buffer space on each machine, i.e.:

$$C = \sum_{i=1}^{L-1} B_{i,i+1} + B_{L,1} + L \tag{5.1}$$

Assume the production line operates under the BBS blocking mechanism, i.e., a machine is not allowed to start processing a part unless a space is available in the next buffer. Also, assume that the processing times at each machine are i.i.d. random variables. The next result follows from Theorem 4.8 and Corollaries 4.1 4.2:

**Theorem 5.1** *The production capacity of a closed-loop production line with BBS blocking mechanism is a symmetrical function of the number of pallets $N$, i.e., $\theta(C-N) = \theta(N)$, $\forall N = 1, ..., C-1$.*

To answer the pallet optimization problem, the following argument is further needed: the throughput of a closed tandem queueing networks with finite buffers is such that there exists a value $N^*$ such that $\theta(N)$ is a non-decreasing function of $N$ for $N = 1, ..., N^*$, and $\theta(N)$ is a non-increasing function of $N$ for $N = N^*, ..., C-1$. If this is true, then Theorem 5.1

implies that the optimal number of pallets is equal to half of the total storage capacity of the closed-loop production lines. Although it is easy to convince oneself that this result is true, it does not seem easy to prove. There is however at least one case for which this result can be established. This is when the throughput is a concave function of $N$, which is the case when the processing time distributions are of PERT-type (Assumption A4). Thus, the next result follows from Theorem 5.1 and Corollary 4.1.

**Theorem 5.2** *The production capacity of a closed-loop production line with BBS blocking mechanism and PERT-type processing time distributions is maximized when the number of pallets $N$ is such that $N = \lfloor C/2 \rfloor$ or $N = \lceil C/2 \rceil$.*

## 5.2 Kanban Controlled Production Lines

Consider a production line operating using a kanban control mechanism; see, e.g., [6]. The production line consists of a series of $L$ machines, $M_1$, $M_2$,..., $M_L$, decomposed into $N$ stages. Stage 1 consists of machines $M_1$ up to $M_{l_1}$, stage 2 consists of machines $M_{l_1+1}$ up to $M_{l_2}$,..., stage $N$ consists of machines $M_{l_{N-1}+1}$ up to $M_L$. With each stage $j$ is associated a number of kanbans, say $K_j, j = 1, ..., N$. Each stage $j$ has an output buffer which contains parts that have already been processed on all of the machines $M_1$ up to $M_{l_j}$. A part located in the output buffer of stage $j$ can be transferred to stage $j+1$ only if a kanban of stage $j+1$ is available (production order). The kanban is then associated with the part during its stay in stage $j+1$. The kanban is released only when the part is transferred to stage $j+2$.

The production capacity of such a kanban controlled production line is defined as the throughput of this system, assuming that raw parts are always available at the input of the first stage and that external demands are always present at the output of the last stage. This is a performance measure of high interest since it corresponds to the maximum (average) rate of external demands that can be satisfied [6, 14]. In other words, it provides the stability condition of the kanban controlled production line.

This kanban controlled production line can be modeled as a FJQN/B [14]. The set of servers are the $L$ machines and the $N-1$ synchronization stations. The $j$-th synchronization station represents the synchronization of a part completed by stage $j$ and a kanban of stage $j+1$. All buffers pertaining to stage $j$ have capacity $K_j$. We assume that the processing times at each machine are i.i.d. random variables. All synchronization stations have zero processing times. The FJQN/B model of a kanban controlled production line consisting of $L = 6$ machines, $N = 3$ stages, each stage consisting of 2 machines, is shown in Figure 12.

The following property follows from the results of Section 4.2.

**Theorem 5.3** *The production capacity of a kanban controlled production line is a monotone and (under Assumption A4) concave function of the number of kanbans of the different stages.*

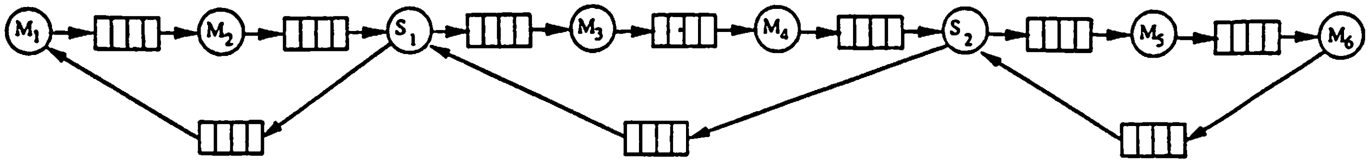


Figure 12: The FJQN/B model of a kanban controlled production line.

Consider now the following reverse system. It consists of the series of $L$ machines in reverse order, $M_L$, $M_{L-1}$,..., $M_1$, again decomposed into $N$ stages. Stage 1 consists of machines $M_L$ up to $M_{l_{N-1}+1}$, stage 2 consists of machines $M_{l_{N-1}}$ up to $M_{l_{N-2}+1}$,..., stage $N$ consists of machines $M_{l_1}$ up to $M_1$. The number of kanbans associated with stage $j$ is $K_{N-j+1}, j = 1, ..., N$. In words, the $j$-th stage of the reverse system is identical to the $K_{N-j+1}$-th stage of the original system.

Now, it is easy to check that the FJQN/B model of the reverse kanban controlled production line is the reverse of the FJQN/B model of the original kanban controlled production line. The following property then follows from the results derived in Section 4.4.

**Theorem 5.4** *The production capacity of a kanban controlled production line is the same as that of the reverse kanban controlled production line.*

The above results (monotonicity and reversibility) were derived in [7, 8, 16, 23] in the special case where each stage consists of a single machine. Our results show that they hold regardless of the number of machines in each stage. Moreover, it is easy to check that they also hold if the buffers in between consecutive machines within a stage have finite capacity.

Variations around this kanban control mechanism have been described in the literature see, e.g., [6, 8]. For each one, it is possible to model the resulting system as a FJQN/B and apply the results derived in this paper. For instance, the general blocking mechanism introduced in [8] gives rise to a FJQN/B similar to the one described above with one machine per stage, except that the input and output buffers of each machine have finite capacity.

# 6 Extensions

For sake of simplicity, we have restricted our attention to a specific class of queueing networks with finite capacity buffers and fork/join mechanisms. Although this class is likely to include many models encountered in practical applications, it may be of interest to deal with models that do not belong to this class. For each such model, the approach presented in this paper can be used to analyze its properties. The purpose of this section is to briefly discuss typical extensions of the class of FJQN/B's considered in this paper.

We have assumed that loading and unloading operations are instantaneous, i.e., they take zero time. Taking into account non-zero loading and/or unloading times is, however, very easy. In the case of a SL policy, the time required to load all the jobs onto the server can be represented by the service time of server $l_i$, i.e., the sequences of service times $\sigma_{l_i,n}, n \geq 1$. In the case of an IL policy, the time required to load the job coming from buffer $(j,i), j \in p(i)$, can be represented by the service time of server $l_{i,j}$, i.e., the sequences of service times $\sigma_{l_{i,j},n}, n \geq 1$. In the case of a SU policy, the time required to unload all the jobs from the server can be represented by the service time of server $u_i$, i.e., the sequences of service times $\sigma_{u_i,n}, n \geq 1$. In the case of an IU policy, the time required to unload the job going to buffer $(i,k), k \in s(i)$, can be represented by the service time of server $u_{i,k}$, i.e., the sequences of service times $\sigma_{u_{i,k},n}, n \geq 1$.

We have restricted our attention to the eight types of B-servers defined in Section 3.1. They are likely to be appropriate to model most of the operating mechanisms encountered in practical applications and, in particular, in manufacturing systems. However, it is worth emphasizing that it is possible to handle more different operating mechanisms. In what follows, we briefly describe such operating mechanisms.

It is possible to define more general loading and unloading policies. Consider, for instance, the loading policy. First, the set of upstream buffers of a server $i$, $u(i)$, can be partitioned into two subsets, say $u(i, IL)$ and $u(i, SL)$, such that: the jobs coming from each of the buffers in $u(i, IL)$ can be loaded independently of the other jobs, while the jobs coming from each of the buffers in $u(i, SL)$ have to be loaded simultaneously. More generally, the loading policy can be defined as a directed acyclic graph describing the *precedence constraints* among the loading of the jobs coming from the different upstream buffers. The unloading policy can be defined in a similar way.

The blocking mechanism of B-servers was either BAS, BBS or BBS-CL mechanism. However, it is also possible to mix these blocking mechanisms. Consider any server, say $i$. In the most general case, the set of downstream buffers, $d(i)$ is partitioned into three subsets, say $d(i, BAS)$, $d(i, BBS)$ and $d(i, BBS - CL)$. The blocking mechanism of server $i$ is then BAS, BBS and BBS-CL, with respect to the buffers belonging to $d(i, BAS)$, $d(i, BBS)$ and $d(i, BBS - CL)$, respectively.

We assumed that the servers were either U-servers or B-servers. However, it would also be possible to consider servers having spaces to accommodate jobs in service, but only those belonging to a subset of the upstream buffers.

Finally, it is also possible to consider assembly operations that can be decomposed into several tasks that have to be performed according to a given precedence graph. Consider, for instance, the case where the assembly operation is decomposed into two tasks, task 1 and task 2, that have to be performed successively. Performing task 1 may require one job of each buffer of only a subset of the upstream buffers. The jobs of the remaining buffers are only required in order for task 2 to be performed.

For all of the operating mechanisms described above, and for any combination of them, it is possible to transform the resulting FJQN/B into a B-FJQN/B in a similar way, although possibly more complicated, as was done in Section 3.2. It is then possible to investigate the properties of any FJQN/B operating under these more general mechanisms based on the properties of B-FJQN/B's in a similar way as was done in Section 4 for the class of FJQN/B's considered in this paper.

# A Proof of Lemma 3.1

In this appendix we prove the equivalence theorem stated in Section 3.2.

**Lemma 3.1** *Let $\mathcal{S}$ be an arbitrary FJQN/B with $i$ being a type BAS/IL/IU server. Let $\mathcal{S}'$ be the FJQN/B obtained from $\mathcal{S}$ by applying the transformation described in Section 3.2 to server $i$. The behavior of $\mathcal{S}'$ is equivalent to that of $\mathcal{S}$ in the following sense:*

$$\begin{aligned}
X'_{k,j,n} &= X_{k,j,n}, \quad \forall j \in p(k); \quad k \in V; \quad k \neq i, \\
D'_{k,n} &= D_{k,n}, \quad \forall k \in V; \quad k \neq i, \\
Y'_{k,j,n} &= Y_{k,j,n}, \quad \forall j \in s(k); \quad k \in V; \quad k \neq i, \\
X'_{l_{i,j},n} &= D_{i,j,n}, \quad \forall j \in p(i), \\
D'_{i,n} &= D_{i,n}, \\
Y'_{u_{i,k},n} &= D_{i,k,n}, \quad \forall k \in s(i),
\end{aligned}$$

*for all $n \geq 1$.*

**Proof.** We focus on the event times associated with server $i$. Let $\{A_{j,n}, j \in p(i)\}_{n=1}^{\infty}$ denote the times at which jobs enter buffer $(j,i)$ in $\mathcal{S}$ and let $\{Z_{k,n}, k \in s(i)\}_{n=1}^{\infty}$ denote the times that jobs are removed from buffer $(k,i)$ in $\mathcal{S}$. In an analogous manner, let $\{A'_{j,n}, j \in p(i)\}_{n=1}^{\infty}$ denote the times at which jobs enter buffer $(j, l_{i,j})$ in $\mathcal{S}'$ and let $\{Z'_{k,n}, k \in s(i)\}_{n=1}^{\infty}$ denote the times that jobs are removed from buffer $(u_{i,k}, k)$ in $\mathcal{S}'$. These times depend on the service times at all servers and the different mechanisms associated with the servers. The evolution equations for server $i$ within $\mathcal{S}$ are, for $n = 1, 2, \ldots$,

$$\begin{aligned}
X_{i,j,n} &= \max\left\{A_{j,n-M_{j,i}}, \max_{k \in s(i)}\left\{Y_{i,k,n-(1-(L_{i,j}+U_{i,k}))}\right\}\right\}, \qquad \forall j \in p(i) \\
D_{i,n} &= \sigma_{i,n} + \max_{j \in p(i)}\left\{X_{i,j,n-L_{i,j}}\right\}, \\
Y_{i,k,n} &= \max\left\{D_{i,n-U_{i,k}}, Z_{k,n-(B_{i,k}-M_{i,k})}\right\}, \quad \forall k \in s(i).
\end{aligned}$$

Here $A_{i,j,n}$, $C_{i,n}$, and $Y_{i,k,n}$ are taken to be 0 for $n \leq 0$.

The evolution equations for the servers in the subnetwork in $\mathcal{S}'$ that represents server $i$ are, for

$n = 1, 2, \ldots,$

$$D'_{l_{i,j},n} = \max\left\{A'_{j,n-M_{j,i}}, D'_{l_{i,j},n-1}, D'_{i,n-(1-L_{i,j})}, \max_{k\in s(i)}\left\{D'_{u_{i,k},n-(1-(U_{i,k}+L_{i,j}))}\right\}\right\}, \quad \forall j \in p(i), \qquad (A.1)$$

$$D'_{i,n} = \sigma_{i,n} + \max\left\{\max_{j\in p(i)}\left\{D'_{l_{i,j},n-L_{i,j}}\right\}, D'_{i,n-1}, \max_{k\in s(i)}\left\{D'_{u_{i,k},n-(1-U_{i,k})}\right\}\right\}, \qquad (A.2)$$

$$D'_{u_{i,k},n} = \max\left\{D'_{i,n-U_{i,k}}, D'_{u_{i,k},n-1}, \max_{j\in p(i)}\left\{D'_{l_{i,j},n-(L_{i,j}+U_{i,k})}\right\}, Z'_{k,n-(B_{i,k}-M_{i,k})}\right\}, \quad \forall k \in s(i). \qquad (A.3)$$

These quantities take value 0 whenever $n \leq 0$.

The following inequalities can be established,

| | | | |
|---|---|---|---|
| $D'_{l_{i,j},n-L_{i,j}}$ | $\geq$ | $D'_{i,n-1}$, | from (A.1), |
| $D'_{u_{i,k},n-(1-(U_{i,k}+L_{i,j}))}$ | $\geq$ | $D'_{l_{i,j},n-1}$, | from (A.3), |
| $D'_{l_{i,j},n-(L_{i,j}+U_{i,k})}$ | $\geq$ | $D'_{u_{i,k},n-1}$, | from (A.1), |
| $D'_{u_{i,k},n-(1-(U_{i,k}+L_{i,j}))}$ | $\geq$ | $D'_{i,n-(1-L_{i,j})}$, | from (A.3), |
| $D'_{i,n-U_{i,k}}$ | $\geq$ | $D'_{l_{i,j},n-(L_{i,j}+U_{i,k})}$, | from (A.2), |
| $D'_{l_{i,j},n-L_{i,j}}$ | $\geq$ | $D'_{u_{i,k},n-(1-U_{i,k})}$, | from (A.1). |

An application of these inequalities to (A.1) – (A.3) yields

$$D'_{l_{i,j},n} = \max\left\{A'_{j,n-M_{j,i}}, \max_{k\in s(i)}\left\{D'_{u_{i,k},n-(1-(U_{i,k}+L_{i,j}))}\right\}\right\},$$

$$D'_{i,n} = \sigma_{i,n} + \max_{j\in p(i)}\left\{D'_{l_{i,j},n-L_{i,j}}\right\},$$

$$D'_{u_{i,k},n} = \max\left\{D'_{i,n-U_{i,k}}, Z'_{k,n-(B_{i,k}-M_{i,k})}\right\}$$

which are identical to the evolution equations for server $i$ in $S$ except that one set is expressed in terms of $A_{j,n}$ and $Z_{k,n}$ whereas the other is expressed in terms of $A'_{j,n}$ and $Z'_{k,n}$. However, since the remainder of the networks excluding server $i$ and its transformation are identical, it follows that $A'_{j,n} = A_{j,n}$, $\forall j \in p(i); n \geq 1$, and $Z'_{k,n} = Z_{k,n}$, $k \in s(i); n \geq 1$. ∎

## References


[1] I. Adan, J. Van Der Wal, "Monotonicity of the Throughput in Single Server Production and Assembly Networks with Respect to the Buffer Sizes," in *Queueing Networks with Blocking*, H.G. Perros, T. Altiok (Ed.s), North-Holland, 325-344, 1989.

[2] M. H. Ammar, S. B. Gershwin, "Equivalence Relations in Queuing Models of Fork/Join Queueing Networks with Blocking," *Performance Evaluation* **10**, pp. 233-245, 1989.

[3] V. Anantharam, P. Tsoucas, "Stochastic Concavity of Throughput in Series of Queues with Finite Buffers," *Adv. Appl. Prob.*, **22**, 761-763, 1990.

[4] F. Baccelli, Z. Liu, "Comparison Properties of Stochastic Decision Free Petri Nets," Rapport de Recherche INRIA, No. 1433, 1991, also to appear in *IEEE Trans. on Automatic Control.*

[5] K. R. Baker, S. G. Powell, D. F. Pyke, "Buffered and Unbuffered Assembly Systems with Variable Processing Times," *Journal of Manufacturing and Operations Management.* , **3**, 200-223, 1990.

[6] J. A. Buzacott, J. G. Shanthikumar, "Stochastic Models of Manufacturing Systems," Prentice Hall, Englewood Cliffs, N. J., 1992.

[7] D. W. Cheng, "Line Reversibility of Tandem Queues with general Blocking," Working Paper, August 1991.

[8] D. W. Cheng, D. D. Yao, "Tandem Queues with General Blocking: A Unified Model and Comparison Results," Working Paper, January 1991.

[9] Y. Dallery, S. B. Gershwin, "Manufacturing Flow Line Systems: a Review of Models and Analytical Results," Tech. Report MASI No. 91-18, March 1991., to appear in *Queueing Systems Theory and Applications.*

[10] Y. Dallery, Z. Liu, D. Towsley, "Equivalence, Reversibility, Symmetry and Concavity in Fork/Join Queueing Networks with Blocking," Tech. Report COINS, TR91-88, 1991.

[11] Y. Dallery, D. Towsley, "Symmetry Property of the Throughput in Closed Tandem Queueing Networks with Finite Buffers," Tech. Report MASI, No. 90-23, May 1990, to appear in *Operations Research Letters.*

[12] M. Di Mascolo, R. David, Y. Dallery, "Modeling and Analysis of Assembly Systems with Unreliable Machines and Finite Buffers," *IIE Transactions* **23**, 4, 315-330, 1991.

[13] M. Di Mascolo, Y. Frein, Y. Dallery, R. David, "A Unified Modeling of Kanban Systems using Petri Nets," *Int. Jour. of Flexible Manufacturing Systems* **3**, 275-307, 1991.

[14] M. Di Mascolo, Y. Frein, Y. Dallery, "An Analytical Method for Performance Evaluation of Kanban Controlled Production Systems," Tech. Report LAG, September1991.

[15] S. B. Gershwin, "Assembly/Disassembly Systems: an Efficient Decomposition Algorithm for Tree-Structured Network," *IIE Transactions* **23**, 4, 1991.

[16] P. Glasserman, D. D. Yao, "Structured Buffer-Allocation Problems in Production Lines," Working Paper, June 1991.

[17] X.-G. Liu, "Toward Modeling Assembly Systems: Applications of Queueing Networks with Blocking," Ph. D. Thesis, Department of Management Sciences, University of Waterloo, Waterloo, 1990.

[18] L.E. Meester, J.G. Shanthikumar, "Concavity of the throughput of tandem queueing systems with finite buffer storage space," *Adv. Appl. Prob.*, **22**, 764-767, 1990.

[19] E. G. Muth. "The reversibility property of production lines," *Management Science*, **25**, 2, 152-158, 1979.

[20] R. O. Onvural, H. G. Perros. "On Equivalencies of Blocking Mechanism in Queueing Networks with Blocking," *Operations Research Letters*, **5-6**, 293-298, 1986.

[21] H. G. Perros, "Approximation Algorithms for Open Queueing Networks with Blocking," *Stochastic Analysis of Computer and Communication Systems*, (ed. H. Takagi), North-Holland, Amsterdam, Netherlands, 1989.

[22] J.G. Shanthikumar, D.D. Yao, "Monotonicity and Concavity Properties in Cyclic Queueing Networks with Finite Buffers," in *Queueing Networks with Blocking*, H.G. Perros, T. Altiok (Ed.s), North-Holland, 325-344, 1989.

[23] S. Tayur, "Structural Results and a Heuristic for Kanban Controlled Serial Lines," Tech. Report No. 934, Cornell University, December 1991.

[24] G. Yamazaki, H. Sakasegawa, "Properties of Duality in Tandem Queueing Systems," *Annals of the Institute of Statistical Mathematics*, **27**, 201-212, 1975.

[25] G. Yamazaki, T. Kawashiha, H. Sakasegawa, "Reversibility of Tandem Blocking Queueing Systems," *Management Science*, **31**, 1, 78-83, 1985.