Equivalence, Reversibility, Symmetry and Concavity Properties in Fork–Join Queuing Networks with Blocking

YVES DALLERY
Laboratoire MASI, Paris, France

ZHEN LIU
INRIA, Valbonne, France

AND

DON TOWSLEY
University of Massachusetts, Amherst, Massachusetts

Abstract. In this paper, we study quantitative as well as qualitative properties of Fork–Join Queuing Networks with Blocking (FJQN/Bs). Specifically, we prove results regarding the equivalence of the behavior of a FJQN/B and that of its duals and a strongly connected marked graph. In addition, we obtain general conditions that must be satisfied by the service times to guarantee the existence of a long-term throughput and its independence on the initial configuration. We also establish conditions under which the reverse of a FJQN/B has the same throughput as the original network. By combining the equivalence result for duals and the reversibility result, we establish a symmetry property for the throughput of a FJQN/B. Last, we establish that the throughput is a concave function of the buffer sizes and the initial marking, provided that the service times are mutually independent random variables belonging to the class of PERT distributions that includes the Erlang distributions. This last result coupled with the symmetry property can be used to identify the initial configuration that maximizes the long-term throughput in closed series-parallel networks.

Categories and Subject Descriptors: C.4 [Computer Systems Organization]: Performance of Systems—modeling techniques; D.4.8 [Operating Systems]: Performance—queueing theory; stochastic analysis; G.1.3 [Mathematics of Computing]: Miscellaneous—queueing theory

The work of Z. Liu was supported in part by the CEC DG XIII under the ESPRIT BRA grant QMIPS.
The work of D. Towsley was supported in part by the National Science Foundation (NSF) under grant ASC-88-8802764 and by the Office of Naval Research (ONR) under grant N00014-87-K-0796.
It was performed while D. Towsley was visiting Laboratoire MASI, UPMC, Paris, France.
Authors’ addresses: Y. Dallery, Laboratoire MASI, Université Pierre et Marie Curie, 4, place Jussieu, 75252 Paris Cedex 05, France; Z. Liu, INRIA Centre Sophia Antipolis, 2004 route des Lucioles, 06565 Valbonne, France; D. Towsley, Department of Computer Information Science, University of Massachusetts, Amherst, MA 01003.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.
© 1994 ACM 0004-5411/94/0900-0903 $03.50

1. Introduction

A Fork–Join Queuing Network with Blocking (FJQN/B) is a queuing 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. Some servers may have no input buffers or no output buffers. Such servers are referred to as sources and sinks, respectively. All of the buffers have finite capacity. A server is allowed to start service whenever there is at least one job in each of its input buffers and space for at least one job in each of its output buffers, that is, the server is neither starved nor blocked. Sources are never starved and sinks are never blocked. Upon completion of service, one job is removed from each of the input buffers and one job is added to each of the output buffers. The FJQN/B, as described above, was first introduced by Ammar and Gershwin [1989]. Note that the blocking mechanism in a FJQN/B corresponds to the so-called blocking-before-service mechanism [Perros 1989].

A FJQN/B is structurally characterized by a graph representing the links between the set of servers and the set of finite-capacity buffers. Moreover, the behavior of a 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. Finally, since we are mainly interested in the quantitative behavior, a FJQN/B is further characterized by the durations of service times which are random variables (r.v.'s).

In this paper, we focus on five aspects of FJQN/Bs. (1) their equivalence properties including duality, (2) the existence and uniqueness properties of their throughput, (3) their reversibility properties, (4) their symmetry properties, and (5) the concavity of the throughput with respect to the buffer sizes and to the initial marking.

1.1 Equivalence Properties. The first equivalence property that we study is that of duality. The duality property of FJQN/Bs is related to the concept of job/hole duality introduced by Gordon and Newell [1967] within the context of closed tandem queuing networks, and earlier observed by Sevast'yono [1962]. Here a hole corresponds to space for one job in a buffer. The idea is that the movement of jobs in one direction is equivalent to the movement of holes in the opposite direction. This duality concept was generalized to FJQN/Bs by Ammar and Gershwin [1989]. A dual FJQN/B is obtained by reversing the direction of the flow in a subset of buffers and changing the corresponding initial marking to the number of holes initially present in these buffers. The full dual is a dual FJQN/B for which all of the flows have been reversed. We show that any dual FJQN/B, in particular the full dual, has exactly the same behavior as the original FJQN/B. As a consequence, a FJQN/B and its dual have the same throughput and stationary marking (with markings replaced by holes when the flow of the corresponding edges is changed) distribution, provided that these quantities exist. These results were already obtained by Ammar and Gershwin [1989] in the special case of exponentially distributed
Properties of FJQN/BS

We also introduce a canonical form for FJQN/BSs, where there exists at most one buffer between two servers. We show that for any FJQN/BS, there exists a FJQN/BS in canonical form that exhibits an equivalent behavior. Last, we show the equivalence between the class of FJQN/BSs and the class of Strongly Connected Marked Graphs (SCMGs) [Commoner et al. 1971; Murata 1989].

1.2 Existence and Uniqueness Properties. We prove the existence of the throughput of a FJQN/BS under very general conditions on the service times. Specifically, we require that the service times of the servers be jointly stationary and ergodic, and be integrable (i.e., have finite means). This existence property was first proved by Baccelli [1992] within the context of SCMGs. Our primary new result concerning this part of the problem is that the throughputs of a FJQN/BS under two different initial markings are equal, provided that these two markings are reachable from each other and the service times of different servers are independent of each other. Because of the equivalence of FJQN/BSs and SCMGs, this result also holds for SCMGs thus proving a basic assumption made in Baccelli et al. [1991] for deriving stability conditions of marked graphs.

1.3 Reversibility Properties. For any FJQN/BS with an initial marking, we define the reverse FJQN/BS to be the network obtained by reversing the flows of jobs while keeping the same initial marking. We show that the reverse FJQN/BS has the same throughput as the original FJQN/BS provided that the service times form jointly stationary reversible ergodic sequences of integrable random variables. Transient results are also derived. These generalize the results established in Dallery and Towsley [1991] for closed tandem queuing networks. As an application of these results, one can also obtain the reversibility property for open tandem queuing systems with finite buffers and blocking-after-service mechanism, and relax the statistical assumption of independent identically distributed (i.i.d.) service times made in Melamed [1986], Muth [1979], and Yamazaki et al. [1985].

1.4 Symmetry Properties. We establish a symmetry property of the throughput of a FJQN/BS. We consider a FJQN/BS with a given initial marking and the same FJQN/BS with the symmetrical initial marking, that is, the initial marking with jobs replaced by holes at each buffer. We show that these two FJQN/BSs have the same throughput by combining the results on reversibility and duality. Such a symmetry property was first conjectured by Onvural and Perros [1987] for closed tandem queuing networks with blocking before service mechanism and exponential distributions of service times, and first proved by Dallery and Towsley [1991] under the assumption that service times have phase type distributions.

Note that the assumptions required for obtaining the throughput, reversibility, and symmetry properties allow the sequences of i.i.d. service times as a special case.

1.5 Concavity Properties. Finally, we establish that the throughput of a FJQN/BS is a concave function of the buffer sizes and the initial markings, provided that the service times are mutually independent r.v.'s coming from a subclass of log-concave distributions called PERT distributions [Baccelli and Liu 1992b], which includes Erlang distributions as a special case. The concavity
of the throughput with respect to the buffer sizes can also be obtained by a result of Baccelli and Liu [1992b], which states that the throughput is a concave function of the initial marking in a Marked Graph (MG) when the holding times have PERT distributions. Our new contribution is the concavity with respect to the initial marking. Our results generalize those of Shanthikumar and Yao [1989] on the concavity of the throughput with respect to the number of jobs in closed tandem queuing networks and the results of Meester and Shanthikumar [1990] and Anantharam and Tsoucas [1990] on the concavity of the throughput with respect to the buffer sizes in open-tandem queuing networks, all of which were established under the assumption of exponential service times.

FJQNs/Bs are of interest because they are particularly suited to the modeling and performance evaluation of manufacturing systems, computer systems and communication networks. In the case of manufacturing, forking corresponds to the process of splitting a component into two or more subcomponents, joining corresponds to the assembly of two or more subcomponents into a single component. Examples of the study of fork-join queuing networks (also called assembly/disassembly networks) in manufacturing can be found in Di Mascolo et al. [1991] and Gershwin [1988/1989]. The FJQN/B also captures the behavior of parallel programming constructs such as the fork-join primitive available in many parallel programming languages [Andrews and Schneider 1983; Hoare 1978; Pyle 1991]. Some of the many papers that have been devoted to the study of such behavior include Baccelli and Liu [1990], Baccelli and Makowski [1989], Baccelli et al. [1989] and Nelson and Tantawi [1988]. Most of these studies assume infinite buffering. Last, FJQNs/Bs have been used to model problems in error control [Bhargava et al. 1988; Caseau and Pujolle 1979] and flow control [Fdida et al. 1990] in communication networks.

Our results also have application to the optimization of FJQNs/Bs. For example, we can identify, on the basis of the symmetry and concavity properties, the initial marking that maximizes the throughput for a subclass of FJQNs/Bs that includes the closed tandem queuing network. For this last example, the throughput is maximized by setting the number of jobs in the network to one half of the sum of the buffer sizes when the service times form mutually independent sequences of i.i.d. r.v.'s with PERT type distributions.

The paper is organized as follows: The formal definition of FJQNs/Bs is given in Section 2. The equivalence properties of FJQNs/Bs are described in Section 3. Sections 4 and 5 contain the results pertaining to the qualitative and throughput behaviors of FJQNs/Bs, respectively. Reversibility, symmetry, and concavity properties are presented in Sections 6, 7, and 8, respectively. Applications of these results to modeling, performance evaluation, and optimization of computer systems, communication networks, and manufacturing systems (including those with buffers of infinite capacity and those with servers prone to failures) are provided in Section 9. Finally, conclusions are given in Section 10.

2. Fork–Join Queuing Networks with Blocking

A Fork–Join Queuing Network with Blocking (FJQN/B) is generally represented as a bipartite graph satisfying certain constraints. Let

$$\mathcal{A} = (V_r, V_s, E, B)$$
be a FJQN/B where \( V_s \) is a set of \( n_s \) servers, \( V_b \) is a set of \( n_b \) buffers, 
\( E \subseteq V_s \times V_b + V_b \times V_s \) is a set of directed edges indicating the flow of jobs 
from servers to buffers and from buffers to servers.\(^1\) Here, \( E \) is required 
to satisfy the constraint that \( \forall k \in V_b: |\{(i, k) \in E: i \in V_s\}| = 1 \) and \( \{(k, i) \in E: i \in V_s\}| = 1 \), that is, each buffer has one incoming and one outgoing edge. The 
buffers are of finite capacity with sizes given by \( B = (B_1, \ldots, B_{n_b}) \) where 
\( B_k \in \mathbb{N} \) is the capacity of buffer \( k \in V_b \). It is convenient to refer to the 
underlying graph, \( G = (V = V_s + V_b, E) \), which is assumed to be connected. 
We will abuse notation for sake of readability by labeling the servers \( i = 1, \ldots, n_s \) and the buffers as \( k = 1, \ldots, n_b \). 

Define the set of immediate buffer predecessors (or, the upstream buffers) 
of server \( i \in V_s \), \( p_b(i) \), to be the set of buffers that have a direct link to \( i \), 
\[ p_b(i) = \{k \in V_b \mid (k, i) \in E\} \]
and the set of immediate buffer successors (or, the downstream buffers) of 
servers \( i \in V_s \), \( s_b(i) \), to be the set of buffers to which \( i \) has direct links, 
\[ s_b(i) = \{k \in V_b \mid (i, k) \in E\} \]
Define the set of immediate predecessors of server \( i \in V_s \), \( p_s(i) \), to be the set 
of servers that can reach \( i \) without passing through any other server, 
\[ p_s(i) = \{j \in V_s \mid \exists k \in V_b: (j, k), (k, i) \in E\} \]
and the set of immediate server successors of server \( i \in V_s \), \( s_s(i) \), to be the set 
of servers that \( i \) can reach without passing through any other servers, 
\[ s_s(i) = \{j \in V_s \mid \exists k \in V_b: (i, k), (k, j) \in E\} \].

The FJQN/B behaves in the following manner. Server \( i \) initiates a service period 
whenever there resides at least one job in each of the buffers in \( p_b(i) \) and there is space for at least one job in each of the buffers in \( s_b(i) \). Server \( i \) is 
said to be started if at least one of the immediate upstream buffers is empty 
and blocked if at least one of the immediate downstream buffers is full. Note 
that the server can simultaneously be starved and blocked. Jobs remain in the 
buffers in \( p_b(i) \) throughout the service period, that is, 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 \( p_b(i) \) and a job is 
immediately placed in each of the buffers in \( s_b(i) \). Observe that the blocking 
mechanism described above corresponds to what is referred to as blocking before service 
in the literature [Perros 1989]. Note that a FJQN/B thus defined 
allows no routing choices.

There may be some servers for which there are no incoming edges. Each 
such server is referred to as a source, and it is assumed that there are an 
infinite number of jobs available to the source so that it is never starved. There 
may be other servers for which there are no outgoing edges. Each such server 
is referred to as a sink, and is assumed to never be blocked. Each job that 
completes at a sink leaves the system immediately.

An example of a FJQN/B is given in Figure 1 (servers are represented 
by circles and buffers by rectangles). This FJQN/B has 10 servers and 12 buffers. 
Servers 1 and 6 are sources and server 5 is a sink. Two special cases of 
FJQN/Bs that have received much attention are open tandem queuing networks 
(see Figure 2) and closed tandem queuing networks (see Figure 3). 
Another special class of FJQN/Bs that is of interest corresponds to closed

\(^1\) When \( A \) and \( B \) are sets, \( A + B = A \cup B \) and \( A - B = \{a: a \in A, a \notin B\} \).
series-parallel fork-join networks. An example of such a network is given in Figure 4.

We introduce some terminology from graph theory related to the underlying graph, $G$. A path from node $i_1$ to node $i_n$, both in $V$, is a sequence of contiguous edges, $(i_1, i_2), (i_2, i_3), \ldots, (i_{n-1}, i_n)$, that are in $E$. A circuit, $C$, is a path with $i_1 = i_n$. A chain is a sequence of undirected contiguous edges, $(i_1, i_2), (i_2, i_3), \ldots, (i_{n-1}, i_n)$, such that either $(i_{k}, i_{k+1}) \in E$ or $(i_{k+1}, i_{k}) \in E$. A cycle, $C$, is a chain with $i_1 = i_n$. We refer to the set of edges associated with either a circuit or cycle $C$ as $E(C) \subseteq E$. Throughout the paper, we only consider elementary circuits and cycles, that is, those containing no subcircuits or subcycles. For simplicity, we use the terms circuits and cycles to mean elementary circuits and elementary cycles. Note that all paths are chains and all circuits are cycles. The reader should be careful to remember the definitions of these terms as they are not always consistent with the definitions sometimes introduced in the theory of queuing networks.

In the FJQN/B of Figure 1, there is a single circuit and several cycles. The circuit consists of the set of edges $((7, 8), (8, 9), (9, 10), (10, 7))$ and a cycle is, for instance, defined by the set of edges $((2, 3), (3, 5), (5, 4), (4, 2))$. An open tandem queuing network has no cycle while a closed tandem queuing network is a FJQN/B that consists of a single circuit.

A FJQN/B is circuit-free (also called acyclic in the queuing literature [Baccelli et al. 1989]) if it contains no circuit. It is cycle-free (also referred to as tree in the literature [Gershwin 1988/1989]) if it contains no cycle. A circuit-free FJQN/B and a cycle-free FJQN/B are illustrated in Figures 5 and 6, respectively. Note that an open tandem queuing network (see Figure 2) is a special case of a cycle-free FJQN/B.

Let $m(t) = (m_1(t), \ldots, m_n(t))$ be the marking of the system at time $t \geq 0$ where $m_k(t)$ denotes the number of jobs in buffer $k \in V_b$ at time $t$. The initial
marking at time $t = 0$ is assumed to be $m(0) = M = (M_1, \ldots, M_n)$, $M_k \in \{0, 1, \ldots, B_k\}$; $k \in V_b$. Note that $B_k - m_k(t)$ is the number of holes in buffer $k$ at time $t$. In general, the qualitative as well as quantitative behavior of a FJQN/B is related to its initial marking. Hence, we introduce the notation

$$F = (N, M)$$

\[ F' = (N, M) \]

to denote the FJQN/B $\mathcal{F}$ associated with the initial marking $M$.

The durations of the service periods at server $i$ are given by a sequence of nonnegative service times, $\{\sigma_i, i \geq 1, i \in V_s\}$. We note that service times may take value zero. The introduction of servers whose service periods are of zero length is useful for modeling synchronization mechanisms. In addition, there is an initial timing condition $Y = (Y_1, \ldots, Y_n)$, $Y_i \geq 0$, after which the $i$th server can begin its first service period, $i \in V_s$. In other words, server $i$ does not become available to provide service to any job until time $Y_i$. The introduction of these initial timing conditions proves useful in proving the independence of the asymptotic throughput (defined below) with respect to the initial marking of the network. In addition, there are a number of applications where it is useful to include different initial timing conditions, for example, the modeling of distributed computing systems in which the clocks are not synchronized at system startup.

Denote by

$$F = (\mathcal{F}, Y)$$

the FJQN/B $\mathcal{F}$ coupled with the initial timing condition $Y$. Note that this timing condition need not be independent of the service times.
The performance measure of greatest interest to us is the system throughput. Let \( Q_i(t, \mathcal{F}) \) denote the number of service completions by time \( t \geq 0 \) at server \( i \), \( i \in V \). Let

\[
\theta_i(\mathcal{F}) = \lim_{t \to \infty} \frac{Q_i(t, \mathcal{F})}{t}, \quad i \in V,
\]

(2.1)

provided the limits exist.

Let \( D_i(n, \mathcal{F}) \) denote the time of the \( n \)th service completion at server \( i \) in \( \mathcal{F} \) under initial timing condition \( Y \). Let \( D_i(\mathcal{F}) = \max_{n \in V} D_i(n, \mathcal{F}) \). Often (when
the FJQN/B is deadlock-free, cf. Section 5) the throughputs can equivalently be expressed as

\[
\theta_i(S) = E\left[ \lim_{n \to \infty} \frac{D_{i,r}(S)}{n} \right]^{-1}, \quad i \in V_i, \tag{2.3}
\]

\[
\theta(S) = E\left[ \lim_{n \to \infty} \frac{D_{r}(S)}{n} \right]^{-1} \tag{2.4}
\]

provided the limits exist. We discuss the relationship between these two definitions in Section 5. We will also give conditions for the existence of these limits independent of the sample paths and of the initial conditions (i.e., \(M\) and \(Y\)) in Section 5. Moreover, we will see that, under these conditions, the limits are almost surely (a.s.) constants. We use \(\theta(S)\) to denote the throughput when it is independent of \(Y\).

3. Equivalence Properties

In this section, we define the dual of a FJQN/B and prove the equivalence of a FJQN/B and its duals. We also establish the equivalence between the general class of FJQN/Bs and the class of FJQN/Bs in canonical form. Last, we establish the equivalence between the class of FJQN/Bs and the class of SCMGs.

3.1 Duality Properties of FJQN/Bs. Consider a FJQN/B \(S' = (\mathcal{A}; M)\). Let \(A\) be an arbitrary set of buffers, that is, \(A \subseteq V_b\). We define the A-dual of \(S'\) to be the FJQN/B created from \(S\) by reversing the job flow through the buffers in \(A\) and switching the initial markings with the holes at those buffers.

**Definition 3.1.** Let \(Y = (X, M)\) be a FJQN/B and let \(A \subseteq V_b\). The FJQN/B \(S'^d = (\mathcal{A}^d, M^d)\) is the A-dual of \(S\) if

\[
V_i^d = V_i, \\
V_b^d = V_b, \\
B^d = B, \\
E^d = E - \{(i, k) \in E \mid k \in \Delta\} + \{(k, i) \mid (i, k) \in E, k \in \Delta\} \\
- \{(k, j) \in E \mid k \in \Delta\} + \{(j, k) \mid (k, j) \in E, k \in \Delta\}, \\
M_k^d = \begin{cases} B_k - M_k, & k \in \Delta, \\ M_k, & \text{otherwise.} \end{cases}
\]

Note that if \(S\) is cycle-free, then any \(\Delta\)-dual of \(S\) is also cycle free. Of special interest is the \(V_b\)-dual of \(S\). We shall refer to this as the full dual of \(S\) and denote it as \(S'^f\). Thus, the full dual of a FJQN/B is the FJQN/B obtained by reversing the job flow through all the buffers and switching the initial markings with the holes. The full dual of the FJQN/B illustrated in Figure 1 is illustrated in Figure 7.

The main result of this section is the equivalence between the FJQN/B \(S\) and a \(\Delta\)-dual \(S'^d\). Let \(S'^d = ((V_i, V_b, E^d, B^d), M^d)\) be the \(\Delta\)-dual of \(S = ((V_i, V_b, E, B), M)\).
FIG. 7. Full Dual of the FJQN/B in Figure 1.

Let $m(t) = (m_1(t), \ldots, m_n(t))$, $Q_i(t)(i \in V_\mathcal{S})$, $m^d(t) = (m^d_1(t), \ldots, m^d_n(t))$, and $Q^d_i(t)(i \in V_\mathcal{S}^d)$ be the markings and number of departures of $\mathcal{S}$ and $\mathcal{S}^d$, respectively, at time $t \geq 0$.

**Theorem 3.1.** Let $\mathcal{S}$ and $\mathcal{S}^d$ be duals as defined above. If the service times and initial timing conditions are the same in both systems, that is, $\sigma_{i,n} = \sigma_{i,n}^d$, $i \in V_\mathcal{S}$, $n = 1, 2, \ldots$, and $Y_i^d = Y_i$, $i \in V_\mathcal{S}$, then for all $t \geq 0$,

$$m_i^d(t) = m_k(t), \quad k \notin \Delta,$$

$$m_i^d(t) = B_k - m_i(t), \quad k \in \Delta;$$

and

$$Q_i(t) = Q^d_i(t), \quad \forall i \in V_\mathcal{S}^d.$$

**Proof.** Let us first note that it is sufficient to prove the theorem in the case where $\Delta$ consists of a single buffer. Indeed, the equivalence for any subset of buffers can then be obtained by constructing a sequence of FJQN/Bs such that each FJQN/B in the sequence is obtained from the preceding one by reversing the job flow through a single buffer. So, let $k$ be the buffer for which the job flow is reversed, that is, $\Delta = \{k\}$. Let $i$ and $j$ be the upstream and downstream servers of buffer $k$ in $\mathcal{S}$, respectively, that is, $(i, k), (k, j) \in E$. Note that $i$ and $j$ are also the downstream and upstream servers of buffer $k$ in $\mathcal{S}^d$, respectively, that is, $(j, k), (k, i) \in E^d$. The occupancy of a buffer affects the behavior of a FJQN/B by precluding its upstream (respectively, downstream) server from performing an operation whenever the buffer is currently full (respectively, empty). Thus, server $i$ in $\mathcal{S}$ (respectively, $\mathcal{S}^d$) is precluded from performing an operation at time $t$ if buffer $k$ is full (respectively, empty), that is, $m_i(t) = B_k$ (respectively, $m_i^d(t) = 0$). In a similar way, server $j$ in $\mathcal{S}$ (respectively, $\mathcal{S}^d$) is precluded from performing an operation at time $t$ if buffer $k$ is empty (respectively, full), that is, $m_j(t) = 0$ (respectively, $m_j^d(t) = B_k$). Now, as long as blocking or starvation of servers $i$ or $j$ does not occur in either $\mathcal{S}$ or $\mathcal{S}^d$, the sample-path behavior of these two FJQN/B is identical in the sense that servers initiate and complete their operations at the same instants in both networks. (Remember that both networks are assumed to have...
Properties of FJQN/BS

Moreover, any time the occupancy of buffer \( k \) increases (respectively, decreases) by one in \( S \), that of buffer \( k \in \mathcal{S}^d \) decreases (respectively, increases) by one. During this period of time, we thus have \( m_k(t) = B_k - m_k(t) \).

As a result, a blocking of server \( i \) in \( S \) will occur exactly at the same instant and last exactly the same time as a starvation of server \( i \) in \( \mathcal{S}^d \). Similarly, a starvation of server \( i \) in \( S \) will occur exactly at the same instant and last exactly the same time as a blocking of server \( i \) in \( \mathcal{S}^d \). Thus, both networks will have identical sample-path behaviors.

In words, there is a one to one correspondence between jobs in the buffers within \( \Delta \) in the original FJQN/B \( S \) and the holes within the same buffers in the \( \Delta \)-dual \( \mathcal{S}^d \) and vice versa. Movement of jobs on those links in one FJQN/B corresponds to movement of holes in the other FJQN/B. In addition, there is a one-to-one correspondence between jobs in buffers not in \( \Delta \) in both FJQN/Bs. We note that this equivalence is an equivalence in terms of sample-path from which the following results can be obtained:

**Corollary 3.1.** Whenever the throughput exists for \( S \), the throughput exists for \( \mathcal{S}^d \) and is identical to that of \( S \), that is,

\[
\theta(S, Y) = \theta(S^d, Y)
\]

for all initial timing conditions \( Y \).

**Corollary 3.2.** Whenever the stationary distribution for the system marking exists (the distribution of \( m(t) \) as \( t \to \infty \)) for \( S \), then the stationary distribution for the system marking exists for \( \mathcal{S}^d \) and is related to that of \( S \) by (3.1).

Remark. General conditions for the existence of the stationary marking distribution of \( S \) are found in Baccelli [1992] and Baccelli and Liu [1992a]. In particular, a stationary marking distribution exists when the service times at each server form mutually independent sequences of i.i.d. r.v.'s having infinite support. Hence, the above corollary extends the results of Ammar and Gershwin [1989] developed under the assumption of exponential service times at each server.

The duality property provides some insight on the behavior of FJQN/Bs. First, it shows that there is no fundamental difference between starvation and blocking (recall that we assume blocking before service). Indeed, duality switches around starvation and blocking events. For instance, the starvation of a server due to the empty state of a buffer in the original FJQN/B corresponds to the blocking of the same server due to the full state of the same buffer in the dual FJQN/B.

Second, duality shows that there is no fundamental difference between circuits and cycles that are not circuits. Indeed, it is always possible to transform a circuit into a cycle that is not a circuit using duality. At first glance, one might think that a circuit-free FJQN/B is in some respect simpler than a general FJQN/B. This is however not true and by appropriately choosing a set of buffers, \( \Delta \subseteq V_n \), it is possible to construct a \( \Delta \)-dual of any FJQN/B that is circuit-free. For example, \( \Delta \) can be taken as the set of buffers where the index of the upstream server of a buffer in \( \Delta \) is larger than that of its downstream server.
3.2 FJQN/Bs in Canonical Form. In this subsection, we present an alternate and simpler canonical representation of a FJQN/B. Most of the paper deals with such a FJQN/B.

Definition 3.2. A FJQN/B \( \mathcal{J} = ((V, V_b, E, B), M) \) is said to be in canonical form if there is at most one buffer between every pair of servers:

\[ \forall i, j \in V; \quad |\{k \in V_b | (i, k), (k, j) \in E \text{ or } (j, k), (k, i) \in E \}| \leq 1. \]

We prove the equivalence between the classes of arbitrary FJQN/Bs and FJQN/Bs in canonical form. First, we describe a procedure for transforming an arbitrary FJQN/B \( \mathcal{J} = ((V, V_b, E, B), M) \) into a FJQN/B \( \mathcal{J}^* = ((V', V''_b, E', B'), M') \) in canonical form.

Algorithm:

1. Choose an arbitrary orientation between every pair of servers \( i, j \in V \). Let \( \Delta \subseteq V_b \) be the set of buffers oriented in the reverse direction (with respect to these arbitrary orientations). Let \( \mathcal{J}^d = ((V, V_b, E^d, B), M') \) be the \( \Delta \)-dual of \( \mathcal{J} \). Let \( S_{ij} = \{k \in V_b; (i, k), (k, j) \in E \} \).

2. Construct \( \mathcal{J}^* = ((V', V''_b, E', B'), M') \) as follows:

\[ V'_i = V'_i; \quad V''_i = \emptyset; \quad E' = E \]

For every pair of servers, \( i, j \in V_i \), such that \( |S_{ij}| \geq 1 \) do

\[ V'_b = V'_b \cup \{k_{ij}\}; \]

\[ E' = E' \cup \{(i, k_{ij}), (k_{ij}, j)\}; \]

\[ M_{ij} = \min_{(i, k), (k, j) \in E} M_{ij}^{(s)}; \]

\[ B_{ij} = \min_{(i, k), (k, j) \in E} (B_{ij}^{(s)} - (M_{ij}^{(s)} - M_{ij}^{(d)})). \]

The first step of the algorithm transforms the original network into a dual network such that all buffers between any pair of servers are oriented in the same direction. The second step reduces the number of buffers between any pair of servers to one. Note that by construction of \( \mathcal{J}^d \), if \( |S_{ij}| \geq 1 \), then \( S_{ij} = \emptyset \). Therefore, there is at most one buffer between any pair of servers \( i \) and \( j \) in \( \mathcal{J}^* \). This single buffer is denoted by \( k_{ij} \). Thus, \( \mathcal{J}^* \) is in canonical form. Note that this transformation is a local transformation in the sense that the parameters of buffer \( k_{ij} \) depend only on the parameters of the buffers connecting servers \( i \) and \( j \) in \( \mathcal{J} \), and are otherwise independent of the rest of the network. Note also that in the case where there is a single buffer in between \( i \) and \( j \) in \( \mathcal{J} \) (i.e., \( \mathcal{J} \) is already in canonical form with respect to the pair of servers \( i \) and \( j \)), then by choosing the arbitrary orientation according to the direction of the flow of jobs through this buffer, the corresponding buffer in \( \mathcal{J}^* \), that is, \( k_{ij} \), has the same parameters at the original buffer.

Theorem 3.2. Let \( \mathcal{J} = ((V, V_b, E, B), M) \) be an arbitrary FJQN/B and \( \mathcal{J}^* = ((V', V''_b, E', B'), M') \) be the FJQN/B in canonical form obtained from \( \mathcal{J} \) by applying the above algorithm. If the service times and initial timing conditions are the same in both systems, that is, \( \sigma_{i,n} = \sigma_{i,n}, i \in V, n = 1, 2, \ldots \), and \( Y'_i = Y'_i, i \in V \), then for all \( t \geq 0 \),

\[ m_i(t) - M_b = \begin{cases} m_{k_{ij}}^{(s)}(t) - M_{k_{ij}}^{(s)}, & \forall k \in \Delta \text{ such that } (i, k), (k, j) \in E; \\ - (m_{k_{ij}}^{(s)}(t) - M_{k_{ij}}^{(s)}), & \forall k \in \Delta \text{ such that } (i, k), (k, j) \in E; \end{cases} \]  

and

\[ Q_i(t) = Q'_i(t), \quad \forall i \in V. \]  

Proof. See Appendix A. \( \square \)
Properties of FJQN/BS

As a result of the above equivalence, we restrict our attention to FJQN/BS in canonical form in the remainder of the paper. In this case the FJQN/B can simply be represented by \( \mathcal{F} = (V, E, B) \), where \( V \) is a set of servers and \( E \subset V \times V \) is a set of directed edges indicating the flow of jobs from servers to servers. In this formulation, the buffers are associated with the edges, that is, there is a mapping \( \pi \) from edges to buffers, \( \pi : E \rightarrow V \). The vector \( B \) retains its original meaning although it is sometimes convenient to refer to the edge, that is, \( B_{i,j} = B_{\pi(i,j)} \). Similarly, the initial marking of the buffer associated with edge \( (i, j) \in E \) is denoted by \( M_{i,j} = M_{\pi(i,j)} \). It remains convenient to refer to the underlying graph, \( G = (V, E) \). Similar definitions for the sets of immediate server predecessors and immediate server successors to server \( i \in V \) can be given. We use this simpler representation in the remainder of the paper. Note that the FJQN/B given in Figure 1 is in canonical form. Its simplified representation is given in Figure 8.

3.3 EQUIVALENCE OF FJQN/BS AND SCMGS. We show in this subsection that the class of FJQN/BS is related to the class of SCMGS. A Marked Graph (MG) [Commoner et al. 1971] is a special case of a Petri Net [Peterson 1981] in which each place has only one input and one output transition. Stochastic MG's have recently been analyzed by Bacelli [1992] and Bacelli et al. [1989]. Formally, a (canonical) MG, \( \Gamma' \) is a triple \( \Gamma = (\Theta, \Pi, \Lambda) \) where

- \( \Theta \) is the set of transitions,
- \( \Pi \) is the set of places, where a place is denoted by \( (i, j), i, j \in \Theta \), with the assumption that \( (i, i) \in \Pi \), \( \forall i \in \Theta \).
- \( \Lambda_{i,j} \in \mathbb{N} \) is the initial marking of place \( (i, j) \in \Pi \), with the assumption that \( \Lambda_{i,j} = 1, \forall i \in \Theta \).

In addition, \( \alpha_{i,n} \in \mathbb{R}^+ \) is the holding time of the \( n \)th firing of transition \( i, n = 1, 2, \ldots, \forall i \in \Theta \). Observe that underlying graph \( (\Theta, \Pi) \) is a directed graph. A MG is a strongly connected MG (SCMG), if the underlying directed graph is strongly connected.

The behavior of the MG is characterized by the flow of tokens, which stay at places and are consumed and created by transitions. A transition \( i \) is enabled to fire when there is at least one token in each of the places \( (j, i) \in \Pi \). A firing takes place as soon as it is enabled. At the end of the firing, a token is consumed at each of the places \( (j, i) \in \Pi \), and a token is created in each of the places \( (i, j) \in \Pi \). To be consistent with our discussion of FJQN/BS, we also need to introduce an initial timing condition \( \tau \in \mathbb{R}^+ \), after which the transitions can begin their first firings. Let \( \mu_{i,j}(t) \) denote the number of tokens at place \( (i, j) \in \Pi \) at time \( t \geq 0 \).

**Theorem 3.3.** For every SCMG \( \Gamma = (\Theta, \Pi, \Lambda) \), there exists a FJQN/B \( \mathcal{F} = (V, E, B, M) \) with

- \( V = \Theta \),
- \( E = \Pi - \{(j, j) \mid j \in \Theta\} \),
- \( B_{i,j} = M + \sum_{(i,j) \in \Pi} \Lambda(i,j) + 1, \ (i,j) \in E \),
- \( M_{i,j} = \Lambda(i,j), \ (i,j) \in E \),
such that, if \( Y = Z \) and \( \sigma_{i,n} = \sigma_{i,n}, \forall i \in \Theta, n = 1, 2, \ldots \), then
\[
\mu_{i,j}(t) = m_{i,j}(t), \quad t \geq 0. \quad (i, j) \in E.
\]

**Proof.** Observe that in a FJQN/B, any service consumes a job at each of the upstream buffers and creates a job at each of the downstream buffers, so that the number of jobs in a circuit is a constant at all time \( t \geq 0 \) (cf. Lemma 4.1 below). Hence, the number of jobs in any circuit of \( \mathcal{S} \) is not larger than \( M - 1 \). Since \( \Gamma \) is strongly connected, for all transitions \( i \in \Theta \), there is a nontrivial circuit \( C \neq (i, i) \) of \((\Theta, \Pi)\) such that \( i \in C \). Therefore, every server \( i \in V_i \) is in a circuit of \((V_i, E)\). As every buffer has capacity \( M \) in \( \mathcal{S} \), there is no blocking in \( \mathcal{S} \).

In comparing the evolution of \( \Gamma \) and \( \mathcal{S} \), it is easy to prove by induction on the events of transition firings that

\( - \) a transition \( i \in \Theta \) can fire if and only if the server \( i \in V_i \) can serve, and that

\( - \mu_{i,j}(t) = m_{i,j}(t) \) holds for all \( t \geq 0 \) and \( (i, j) \in E \).

The detailed proof is omitted. \( \square \)

**Theorem 3.4.** For every canonical FJQN/B \( \mathcal{S} = (V_i, E, B), M \), there exists a SCMG \( \Gamma = (\Theta, \Pi, \Lambda) \) with

\[
\Theta = V_i,
\Pi = E + \{(j, j) \mid j \in V_i\} + \{(i, j) \mid (j, i) \in E\},
\Lambda(i, j) = \begin{cases} M_{i,j}, & (i, j) \in E, \\ B_{i,j} - M_{i,j}, & (j, i) \in E, \\ 1, & i = j \in V_i, \end{cases}
\]

such that, if \( Y = Z \) and \( \sigma_{i,n} = \sigma_{i,n}, \forall i \in V_i, n = 1, 2, \ldots \), then

\[
m_{i,j}(t) = \mu_{i,j}(t), \quad t \geq 0. \quad (i, j) \in E.
\]

**Proof.** Observe that the resulting graph, \((\Theta, \Pi)\), is strongly connected as the graph \((V_i, E)\) is connected.
Again, one can prove by induction on the events of service periods that

\( - \) a server \( i \in V_i \) can serve if and only if the transition \( i \in \Theta \) can fire, and that

\( - \) for all \( t \geq 0 \) and all \( (i, j) \in E \):

\[
m_{i,j}(t) = \mu_{i,j}(t) \quad \text{and} \quad B_{i,j} - m_{i,j}(t) = \mu_{j,i}(t).
\]

The detailed proof is omitted. \( \square \)
This equivalence between FJQNBs and SCMGs is of interest for several reasons. On the one hand, it means that all the results presented in this paper can readily be applied to SCMGs. On the other hand, some results can be borrowed from the theory of MGs and adapted to FJQNBs. This is especially true for the qualitative properties presented in the next section.

4. Qualitative Behavior of FJQNBs

In this section, we present some qualitative properties of FJQNBs. Some of these properties are similar to those obtained in the framework of marked graphs [Commoner et al. 1971]. Consider a FJQNB $\mathcal{G} = (\mathcal{A}, \mathcal{M})$ that is in canonical form. Let $C$ be a cycle in $\mathcal{A}$. Let us define an arbitrary orientation of this cycle. The set of edges $E(C)$ 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 $E^+(C) + E^-(C) = E(C)$. Note that if the reverse orientation is chosen as the reference, then the two subsets are switched around. Also, if $C$ is a circuit, the natural orientation leads to $E^+(C) = E(C)$ and $E^-(C) = \emptyset$. This partition gives rise to a partition of the set of buffers of cycle $C$: $V^+_p(C) = V^-_p(C) + V^-_p(C)$.

We define the quantities $I^+_C(M)$ ($I^-_C(M)$, respectively) to 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,

$$I^+_C(M) = \sum_{k \in V^+_p(C)} M_k + \sum_{k \in V^-_p(C)} (B_k - M_k)$$

$$I^-_C(M) = \sum_{k \in V^-_p(C)} M_k + \sum_{k \in V^-_p(C)} (B_k - M_k).$$

Note that

$$I^+_C(M) + I^-_C(M) = B_C \equiv \sum_{k \in V^-_p(C)} B_k, \quad I^+_C(M) = I^-_C(B - M).$$

We have the following invariance property satisfied by every cycle in a FJQNB independently of the service times and initial conditions.

**Lemma 4.1.** Let $C$ be an arbitrary cycle in $\mathcal{A}$. The markings in this cycle satisfy the following relation,

$$\sum_{k \in V^+_p(C)} m_k(t) + \sum_{k \in V^-_p(C)} (B_k - m_k(t)) = I^+_C(M), \quad \forall t > 0, \quad (4.1)$$

or equivalently

$$\sum_{k \in V^+_p(C)} (B_k - m_k(t)) + \sum_{k \in V^-_p(C)} m_k(t) = I^-_C(M), \quad \forall t > 0, \quad (4.2)$$

independently of the service times and initial timing conditions.

**Proof.** It is easy to check that a service completion at any server belonging to this cycle does not change the above quantities and thus they are invariant and equal to those corresponding to the initial marking. □
Henceforth, we will refer to $I_c^+(M)$ and $I_c^-(M)$ as the invariants of cycle $C$. Note that in the special case where $C$ is a circuit, the above lemma states that the total number of jobs along this circuit is an invariant, as is the total number of holes. For a closed tandem queuing network, this simply means that the total number of jobs in the network is a constant, usually referred to as the population of the network.

We define the equivalence relation between markings.

**Definition 4.1.** Let $\mathcal{A}$ contain exactly $n_c$ distinct elementary cycles $C_1, \ldots, C_{n_c}$. Let

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

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

Let $M$ be a marking, then $\mathcal{R}(\mathcal{A}, M)$ denotes the set of markings equivalent to $M$, that is, $\mathcal{R}(\mathcal{A}, M) = \{M' : M \sim M\}$. Marking $M'$ is said to be reachable from marking $M$ with respect to $\mathcal{A}$ if there exists some sequence of service completions (where timing constraints are ignored) such that $\mathcal{A}$ can transit from $M$ to $M'$. It has been established for untimed marked graphs that marking reachability and marking equivalence, as defined above are the same (see Commoner et al. [1971]). Hence, $M$ and $M'$ are reachable from each other, in the absence of timing constraints, iff $M \sim M'$. It follows from Lemma 4.1 that, in the presence of timing constraints, a necessary condition for a marking $M'$ to be reachable from a marking $M$ is that it belong to $\mathcal{R}(\mathcal{A}, M)$. Note however that because of timing constraints, it is not, in general, a sufficient condition. In the special case of a cycle-free FJQN/B, since there is no cycle, all markings belong to the same class. For a closed tandem queuing network, all markings corresponding to the same population of the network belong to the same class.

**Definition 4.2.** A FJQN/B $\mathcal{A} = (\mathcal{A}, M)$ is said to be deadlocked if it is impossible for any server to commence a service period, that is, every server is either starved, blocked, or both. The FJQN/B $\mathcal{A}$ is said to be deadlock-free, that is, if $\mathcal{R}(\mathcal{A}, M)$ does not contain any marking $M'$ such that $(\mathcal{A}, M)$ is deadlocked.

The following property can easily be obtained from a similar result pertaining to SCMGS [Commoner et al. 1971] using the equivalence between FJQN/Bs and SCMGS.

**Theorem 4.1.** A FJQN/B with initial marking $M$ is deadlock-free iff the following relation is satisfied,

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

or equivalently

$$0 < I_c^-(M) < B_C$$

for all cycles $C$ in $\mathcal{A}$.

Note that if $C$ is a circuit in a deadlock-free FJQN/B, the above condition states that there is at least one job and at least one hole along the circuit.
THEOREM 4.2. Consider a FJQN/BS $\mathcal{N} = (V, E, B)$ and an initial marking $M$ such that $\mathcal{N} = (\mathcal{N}', M)$ is not deadlock-free. Then, $\mathcal{N}$ reaches a marking for which it is deadlocked after each server has performed a finite number of service activities.

PROOF. Since $\mathcal{N}$ is not deadlock-free, there exists at least one cycle, say $C$, for which the relation given in Theorem 4.1 is not satisfied. This implies that every server that is part of this cycle is either starved, blocked, or both. Moreover, this condition will not change, that is, none of these servers will ever be able to start a service activity. Consider now any server $i$ that is not part of cycle $C$. Since the graph $(V', E)$ is connected, there exists a chain between server $i$ and one of the servers, say $j$, that is part of $C$. It is easy to check that there is an upper bound on the number of service activities that server $i$ can perform before becoming either starved or blocked as the (indirect) consequence of the fact that server $j$ cannot perform any service activity. This upper bound is a function of the initial marking of the buffers that belong to the chain from $i$ to $j$. Since this is true for every server $i$, $i \in V'$, the result follows. □

The following result was obtained in the context of MGs (Theorem 6 in [Commoner et al. 1971]) and thus also holds for FJQN/BSs.

LEMMA 4.2. For any sequence of service completions such that the final marking is the same as the initial marking, the number of service completions is the same for all servers.

Finally, we establish the following result that will be useful in the next section for proving the independence of the throughput with respect to the initial marking. The proof of this result can be found in Appendix B.

THEOREM 4.3. Let $\mathcal{N}^1 = (\mathcal{N}', M^1)$ and $\mathcal{N}^2 = (\mathcal{N}', M^2)$ be two deadlock-free FJQN/BSs that differ only in their initial markings, that is, $M^1 \neq M^2$. If $M^1 \sim M^2$, then by forbidding an arbitrary server $i$ in $\mathcal{N}^1$ from ever initiating a service period, that is, $\mathcal{N} = \mathcal{N}^1 \setminus \{i\}$ and $\mathcal{N}^2$, reach the same marking $M$ where every server other than $i$ is either blocked, starved, or both.

5. Throughput

In this section, we focus on the existence of the asymptotic throughput of an arbitrary FJQN/BS and the equivalences, in term of throughput, between the FJQN/BSs with different initial timing conditions and markings. Let us first consider the case of a non-deadlock-free network.

THEOREM 5.1. Consider a FJQN/BS $\mathcal{N} = (V, E, B)$ and an initial marking $M$ such that $\mathcal{N} = (\mathcal{N}', M)$ is not deadlock-free. Then $\theta_1(\mathcal{N}) = 0$, $\forall i \in V'$.

PROOF. Owing to Theorem 4.2, there is an upper bound, say $N_t$, on the number of service activities any server $i$ can perform before the network becomes deadlocked. As a result, $Q(t, \mathcal{N}) \leq N_t < \infty$, $\forall t \geq 0$. Therefore, the result follows from the expression of the asymptotic throughput given by (2.1). □

As a result, in the rest of this section, we restrict our attention to deadlock-free networks. Our results are based on using the definitions of the throughputs expressed in (2.3) and (2.4). Let us first show that, as long as the network
is deadlock-free, the definitions given in (2.3) and (2.4) are equivalent to those given in (2.1) and (2.2), respectively. It is readily checked that

\[ Q_i(t, \mathcal{C}) = \inf\{n \mid D_{i,n+1}(\mathcal{C}) > t\}, \quad i \in V, \]

This implies that

\[ \frac{n}{n+1} Q_i(t, \mathcal{C}) < \frac{n}{n+1} D_{i,n+1}(\mathcal{C}) \leq \frac{n}{D_{i,n}(\mathcal{C})}, \quad D_{i,n}(\mathcal{C}) \leq t < D_{i,n+1}(\mathcal{C}). \]

It follows that if the limit in (2.3) exists, the limit in (2.1) exists as well and they are identical. We also have

\[ Q(t, \mathcal{C}) = \inf\{n \mid D_{n+1}(\mathcal{C}) > t\}. \]

In a similar way, it follows that if the limit in (2.4) exists a.s., the limit in (2.2) exists a.s. as well and they are identical.

Before establishing the main results of this section, we associate a precedence graph with the canonical FJQN/B. Let \( \mathcal{C} = (V, E, B, M) \) be an arbitrary deadlock-free FJQN/B. Consider a relation \( \prec_{\mathcal{C}} \) between the pairs \((i, n)\), where \( i \in V, n \geq 1 \).

**Definition 5.1.** The pairs \((i, n)\) and \((j, m)\) have the relation \((i, n) \prec_{\mathcal{C}} (j, m)\) iff one of the following relations is satisfied,

1. \( n = m - M_{i,j}, \quad i \in p_i(j) \),
2. \( n = m - 1, \quad i = j \),
3. \( n = m - (B_{j,i} - M_{i,j}), \quad i \in s_i(j) \).

**Definition 5.2.** The graph \( \mathcal{G}_{\mathcal{C}} = (\mathcal{C}, \mathcal{E}) \) is the precedence graph associated with \( \mathcal{C} \) if

\[ \mathcal{C} = \{(i, n) \mid n \geq 1, i \in V\} \]
\[ \mathcal{E} = \{(i, n) \rightarrow (j, m) \mid (i, n), (j, m) \in \mathcal{C}, \quad (i, n) \prec_{\mathcal{C}} (j, m)\}. \]

As we shall see later on, the edges in this graph indicate the ordering in which service periods may begin in \( \mathcal{C} \). The edge corresponding to relation (5.1) carries the meaning that the \( m \)th service period at server \( j \) cannot begin before the completion of the \((m - M_{i,j})\)-th service period at server \( i \in p_i(j) \), that is, buffer \((i, j)\) must be nonempty (because there must be a job in each of the upstream buffers before the server can begin a service period). The edge corresponding to relation (5.2) carries the meaning that the \( m \)th service period at server \( j \) cannot begin before the completion of the previous service period at that server (because only one job can be served at once). The edge corresponding to relation (5.3) carries the meaning that the \( m \)th service period at server \( j \) cannot begin before the completion of the \((m - (B_{j,i} - M_{i,j}))\)-th service period at server \( i \in s_i(j) \), that is, buffer \((i, j)\) must not be full (because there must be space in each of the downstream buffers before the server can begin service).

**Lemma 5.1.** If \( \mathcal{C} \) is deadlock-free, then the precedence graph \( \mathcal{G}_{\mathcal{C}} \) is circuit-free.

**Proof.** Suppose there is a circuit

\[ C = (i_1, n_1) \rightarrow (i_2, n_2) \rightarrow \cdots \rightarrow (i_k, n_k) \rightarrow (i_1, n_1) \]
Properties of FJQN / BS in $S^*$, $k \geq 1$. According to relations (5.1)–(5.3),
\[ n_1 \leq n_2 \leq \ldots \leq n_k \leq n_1, \]
which implies that
\[ n_1 = n_2 = \ldots = n_k, \]
which further implies that $i_1, i_2, \ldots, i_k$ are distinct and form a circuit in $S$. Let $i_{k+1} = i_1$. Using now the definition of $S^*$, we obtain that for $j = 1, \ldots, k$,
either
\[ i_j \in p(i_{j+1}), \quad \text{and} \quad M_{i_j,i_{j+1}} = 0, \]
or
\[ i_j \in s(i_{j+1}), \quad \text{and} \quad M_{i_{j+1},i_j} = B_{i_{j+1},i_j}. \]
Theorem 4.1 can now be applied to show that $S^*$ is not deadlock-free. Therefore, $S^*$ is circuit-free. \(\square\)

We now study the behavior of the throughput of $S$. The existence of the limit in equation (2.4) will be shown in Theorem 5.2 below. We establish first the evolution equations of $S$ that capture the synchronization mechanisms (e.g., starvation and blocking) of $S^*$.

**Lemma 5.2.** If $S$ is deadlock-free, then the evolution equations are:
\[
D_{i,n}(S, Y) = \sigma_{i,n} + \max \left\{ Y, \max \left\{ 0, \max_{j \in p(i)} D_{j,n-1}(S, Y) + M_{i,j}, \right. \right. \right. \\
space \left. \left. \left. \max_{k \in s(i)} D_{k,n-1}(S, Y) + M_{i,k}, \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \right. \r...
obtained in a similar way except that the availability of the server is given by the initial timing condition $Y_t$. 

Based on these evolution equations, we can express the departure times in $\mathcal{S}$ by the lengths of paths in $\mathcal{S}$. Let $\mathcal{P}(i, n)$ be the set of all paths that end at node $(i, n) \in \mathcal{S}$.

**Lemma 5.3.** If $\mathcal{S}$ is deadlock-free, then

$$
D_{i,n}(\mathcal{S}, Y) = \max_{P = (i_1, n_1) \rightarrow \cdots \rightarrow (i_k, n_k) \in \mathcal{P}(i, n)} Y_{i_1} + \sum_{h=1}^{k} \sigma_{i_h, n_h},
$$

$\forall i \in V_s, n \geq 1$. (5.6)

**Proof.** The proof can be carried out by simple induction on the vertices of $\mathcal{S}$, using Lemma 5.2. 

We are now in a position to show the existence of the throughput.

**Theorem 5.2.** Let $\mathcal{S}$ be an arbitrary deadlock-free FJQN/B. Assume that the service times form jointly stationary and ergodic sequences of integrable r.v.s. Then there exists a constant $\mu$ that is independent of the initial timing condition $Y$ such that

$$
\theta^{-1}(\mathcal{S}) = \mu = \lim_{n \to \infty} \frac{D_n(\mathcal{S}, Y)}{n} = \lim_{n \to \infty} \frac{E[D_n(\mathcal{S}, Y)]}{n}
$$

a.s. $\forall Y \in \mathbb{R}^{n_i}$. (5.7)

**Proof.** Let $\mathcal{P}$ denote the set of all paths in $\mathcal{S}$. For all $1 \leq m \leq n$, define

$$
X_{m,n} = \max_{P = (i_1, n_1) \rightarrow \ldots \rightarrow (i_k, n_k) \in \mathcal{P}} \sum_{h=1}^{k} \sigma_{i_h, n_h}
$$

It is easy to check that for all $1 \leq m \leq l < n$,

$$
X_{m,n} \leq X_{m,l} + X_{l+1,n}.
$$

Therefore, $X_{m,n}$ is a subadditive process. Since the service times are jointly stationary, the process $X_{m,n}$ is stationary. Due to the fact that

$$
X_{m,n} \leq \sum_{h=m}^{n} \sum_{i \in V_s} \sigma_{i,h}
$$

the integrability of the service times entails that $X_{m,n}$ is integrable. Using Kingman’s theorem on stationary ergodic subadditive processes (c.f. [Kingman 1968]), we obtain

$$
\mu = \lim_{n \to \infty} \frac{X_{1,n}}{n} = \lim_{n \to \infty} \frac{E[X_{1,n}]}{n} \quad a.s.
$$

(5.8)

Observe that

$$
X_{1,n} \leq D_n(\mathcal{S}, Y) \leq Z + X_{1,n},
$$

where $Z = \max_{i \in V_s} Y_i$, from which we immediately get (5.7).
Remark. Observe that (5.7) implies that the proof of the above theorem is similar to the proof of the existence of the throughput in SCMGs due to Baccelli [1992]. In fact, Theorem 5.2 can also be obtained by using the result of Baccelli [1992] and the equivalence between the class of FJQN/Bs and that of the SCMGs. However, the notion of precedence graph and the evolution equations in Lemmas 5.2 and 5.3 are useful in the remainder of the paper.

The above theorem shows that, under the stationary and ergodic assumptions, the throughput exists and is independent of the initial timing constraints. The existence of the throughput does not require any independence of the service times, and trivially holds for i.i.d. r.v.’s.

The computation of the throughput is quite difficult in general. Exact results are available only in the case of deterministic service times [Ramamoorthy and Ho 1980]. Various bounds based on stochastic ordering techniques are obtained by Baccelli and Liu [Baccelli and Liu 1992b]. If the service times can be represented by phase type distributions [Neuts 1981; Walrand 1988], then the system can be modeled by a Markov chain that is amenable to numerical solution (provided that the state space is not too large). Approximations have been proposed for the case of tandem and closed networks (see Dallery and Gershwin [1992] for references), and cycle-free FJQN/Bs [Mascolo et al. 1991; Gershwin 1989/1990].

As a consequence of Lemma 4.2 and Theorem 5.2, we obtain

COROLLARY 5.1. Let $\mathcal{S}$ be an arbitrary deadlock free FJQN/B. Under the assumptions of Theorem 5.2, the throughputs associated with each server are identical, $\theta(\mathcal{S}, Y) = \theta(\mathcal{S}, Y) = \theta(\mathcal{S}), i = 1, \ldots, n$.

We now establish that the throughput of a FJQN/B is independent of the initial markings provided these initial markings are equivalent (cf. Definition 4.1).

THEOREM 5.3. Let $\mathcal{N} = (V, E, B)$ and let $\mathcal{S}^1 = (\mathcal{N}, M^1)$ and $\mathcal{S}^2 = (\mathcal{N}, M^2)$ be two FJQN/Bs with the same joint distribution of the sequences of service times. Assume that the service times form jointly stationary and ergodic sequences of integrable r.v.’s, and that the sequences of service times at different servers are mutually independent. If $M^1 \sim M^2$, then the throughputs of these two networks are identical:

$$\theta(\mathcal{S}^1) = \theta(\mathcal{S}^2).$$

PROOF. See Appendix C.

Remark. The above theorem shows that one of the basic assumptions made in Baccelli et al. [1991] for deriving stability conditions for marked graphs holds when the sequences of the service times are mutually independent. However, the service times at each server need not be independent of each other.

The following simple counterexample illustrates the necessity that service times at the different servers be independent. Consider a closed tandem network with two servers, two jobs, and buffer capacities greater than two (the actual values are not important in this example). Assume that $\sigma_{1,n} = \sigma_{2,n} = \sigma_n, n = 1, 2, \ldots$ and that $\{\sigma_n\}_{n \geq 1}$ is a sequence of i.i.d. exponential r.v.’s with parameter $\lambda$. Consider two initial markings, one with both jobs in the same buffer and the other with the two jobs in different buffers. In the first case, it is not difficult to show that the throughput is $2\lambda/3$ whereas in the second case,
the jobs commence and complete service simultaneously and the throughput is $\lambda$.

6. Reversibility

In this section, we formally define the reverse FJQN/B, $\mathcal{S}'$, of a FJQN/B $\mathcal{S}$ and show that, given the same initial marking, they exhibit identical throughput. We begin with the definition of the reverse of a FJQN/B.

**Definition 6.1.** Let $\mathcal{S} = (V, E, B, M)$ be a (canonical) FJQN/B. The reverse of $\mathcal{S}$, $\mathcal{S}' = (V, E', B', M')$, contains the same servers, buffers, buffer sizes, and the initial marking as $\mathcal{S}$ but the reverse of all the edges in $\mathcal{S}$: $E' = \{(i, j) | (j, i) \in E\}$, $B'_{ij} = B_{ji}$ and $M'_{ij} = M_{ji}$, $(j, i) \in E$.

Note that $B'$ and $B$ are really the same as are $M'$ and $M$. However, we introduce different symbols to account for the difference in the direction of the edges in $\mathcal{S}$ and $\mathcal{S}'$. Note that the reverse, $\mathcal{S}'$, of $\mathcal{S}$ has the same structure as its full dual, $\mathcal{S}^{\text{full}}$, that is, $\mathcal{S}' = \mathcal{S}^{\text{full}}$. They only differ by their initial marking: $M' = M$ while $M' = B - M$. Note also that there is a one-to-one correspondence between cycles/circuits in $\mathcal{S}$ and $\mathcal{S}'$. This allows us to establish the following result regarding the deadlock-freeness of $\mathcal{S}$ and $\mathcal{S}'$.

**Lemma 6.1.** A FJQN/B $\mathcal{S}$ is deadlock-free iff its reverse $\mathcal{S}'$ is deadlock-free.

**Proof.** Let $C$ be a cycle in $\mathcal{S}$ and let $C'$ be the corresponding cycle in $\mathcal{S}'$. Let the arbitrary orientation of $C'$ be the reverse of that of $C$. Then, it is immediate that $I_{C}(M') = I_{C}(M)$ and $I_{C'}(M') = I_{C}(M)$. Since there is a one to one correspondence between cycles in $\mathcal{S}$ and $\mathcal{S}'$, each satisfying this relationship, it follows from Theorem 4.1 that $\mathcal{S}$ is deadlock-free iff $\mathcal{S}'$ is deadlock-free. \(\square\)

The main results of this section are:

**Theorem 6.1.** Consider a deadlock-free FJQN/B $\mathcal{S}$ and its reverse $\mathcal{S}'$ with the same (joint distribution of the) sequences of service times. If these sequences are jointly reversible, then for all $n \geq 1$, the service completion times $D_n(\mathcal{S}, 0)$ and $D_n(\mathcal{S}', 0)$ have the same distribution.

**Proof.** See Appendix D. \(\square\)

**Remark.** If $\mathcal{S}$ is nondeadlock-free, then it is clear that $Q(t, \mathcal{S}, 0) = Q(t, \mathcal{S}', 0) = 0$, $t \geq 0$.

As a consequence of Theorems 6.1 and 5.2, we obtain

**Theorem 6.2.** Let $\mathcal{S}$ be an arbitrary FJQN/B, and $\mathcal{S}'$ be its reverse with the same (joint distribution of the) sequences of service times. If the service times form jointly reversible, stationary and ergodic sequences of integrable r.v.'s, then

$$\theta(\mathcal{S}) = \theta(\mathcal{S}').$$

Joint reversibility of the sequences $\{\sigma_{i,n}\}_{n \geq 1}$, $i \in V$, means that, for all $n \geq 1$, the joint distribution of $(\sigma_1, \sigma_2, \ldots, \sigma_n)$, where $\sigma_n = (\sigma_{i,n}, i \in V)$, is identical to that of $(\sigma_{i,n}, \sigma_{i,n-1}, \ldots, \sigma_{i})$. Joint reversibility of the sequences implies their joint stationarity, that is, for all $m, n \geq 1$, the joint distribution of $(\sigma_{i,m+1}, \sigma_{i,m+2}, \ldots, \sigma_{i,m+n})$ is identical to that of $(\sigma_{i,1}, \sigma_{i,2}, \ldots, \sigma_{i})$ [Kelly 1979, p.
Properties of FJQN/BS

5]. Note that all of the assumptions made in Theorem 6.2 are fulfilled when the service times are mutually independent and are i.i.d. at each server.

7. Symmetry

In this section, we show that the throughput of a FJQN/B with a given initial marking is identical to the throughput of the same FJQN/B with symmetrical initial marking. This result is obtained by combining the reversibility and duality properties of FJQN/Bs.

Definition 7.1. Let $(\mathcal{S}, (V, E, B, M))$ be a (canonical) FJQN/B. The symmetrical FJQN/B of $\mathcal{S}$, $\mathcal{S}' = (V, E, B, M')$, is the same as $\mathcal{S}$ except that the initial marking is symmetrical to that of $\mathcal{S}$: $M' = (B - M)$.

Note that the initial marking of jobs in $\mathcal{S}'$ corresponds to the initial marking of holes in $\mathcal{S}$ and vice-versa. Thus, $\mathcal{S}'$ can be obtained from $\mathcal{S}$ by applying the reverse and the full-dual transformations in any order, that is, $\mathcal{S}'$ is the reverse of the full dual of $\mathcal{S}$, or equivalently, $\mathcal{S}'$ is the full dual of the reverse of $\mathcal{S}$. This observation allows us to combine the results of Sections 3.1 and 6 and to obtain the following theorems. Actually, Theorem 7.1 follows from Theorems 3.1 and 6.1, whereas Theorem 7.2 follows from Theorem 3.1, Corollary 3.1, and Theorem 6.2.

Theorem 7.1. Consider a deadlock-free FJQN/B $\mathcal{S}$ and its symmetrical FJQN/B $\mathcal{S}'$ with the same (joint distribution of the) sequences of service times. If these sequences are jointly reversible, then for all $n \geq 1$, the service completion times $D_n(\mathcal{S}, 0)$ and $D_n(\mathcal{S}', 0)$ have the same distribution.

Remark. As before, if $\mathcal{S}$ is nondeadlock-free, then $Q(t, \mathcal{S}, 0) = Q(t, \mathcal{S}', 0) = 0$, $t > 0$.

Theorem 7.2. If the service times form jointly reversible, stationary, and ergodic sequences of integrable r.v.'s, then the throughput of a FJQN/B with initial marking $B - M$ is the same as with initial marking $M$, that is:

$$\theta(\mathcal{S}', B - M) = \theta(\mathcal{S}', B - M).$$

When Theorem 7.2 is applied to some particular classes of FJQN/B, various results of practical interest can be obtained. For example, consider the class of closed tandem queuing network (cf. Figure 3). According to Theorem 5.3, when the service times at different servers are independent, the throughput is a function of the population $N$ of the network, that is, the total number of jobs present in the network, and is independent of the precise initial marking. The throughput can thus be denoted by $\theta(\mathcal{N}, N)$.

Corollary 7.1. Assume that the service times form jointly reversible, stationary, and ergodic sequences of integrable r.v.'s, and that the sequences of service times at different servers are mutually independent. Then a closed tandem network FJQN/B with total buffer size $B$ and population $N$ has the same throughput as with total buffer size $B - N$:

$$\theta(\mathcal{N}, B - N) = \theta(\mathcal{N}, N), \quad 1 \leq N \leq B - 1.$$

This result was conjectured by Onvural and Perros [1987] for exponentially distributed service times, and first proved by Dallery and Towsley [1991] under the assumption that service times are characterized by phase-type distributions.
This last result can be generalized to closed networks with series parallel fork-join mechanisms (see Figure 4). However, some restrictions must be made so that the symmetry property of Theorem 7.2 can be interpreted in terms of a symmetry property with respect to the population of the network. Let us consider a subclass of closed series-parallel fork–join network, referred to as uniform closed series-parallel FJQN/Bs, where the sum of the buffer sizes in all circuits are identical, and the total number of jobs in all circuits are also identical. In such networks, the sum of the buffer sizes in a circuit is called the buffer size of the network, and the total number of jobs in a circuit is called the population of the network. It is not difficult to verify that all the initial markings with the same population in a uniform closed series-parallel FJQN/B are equivalent (i.e., reachable). Owing to Theorem 5.3, if the service times at different servers are independent, the throughput is a function of the population N of the network, and is independent of the precise initial marking. Thus, as in the case of closed tandem network, the throughput can be denoted by \( \theta(\mathcal{A}, N) \).

**Corollary 7.2.** Assume that the service times form jointly reversible, stationary, and ergodic sequences of integrable r.v.’s, and that the sequences of service times at different servers are mutually independent. Then a uniform closed series-parallel FJQN/B with buffer size B and population N has the same throughput as one with buffer size B and population B – N:

\[
\theta(\mathcal{A}, B - N) = \theta(\mathcal{A}, N), \quad 1 \leq N \leq B - 1.
\]

It is easy to check that the closed series-parallel FJQN/B of Figure 4 is uniform if

\[
B_1 + B_3 = B_4 + B_6, \quad B_0 + B_10 = B_{11} + B_{12} = B_{13} + B_{14}, \quad M_2 + M_3 = M_4 + M_5 + M_6, \quad M_9 + M_{10} = M_{11} + M_{12} = M_{13} + M_{14}.
\]

There are other FJQN/Bs for which Theorem 7.2 can be interpreted as a symmetry property with respect to population of jobs, by using Theorem 5.3. For example, Corollary 7.2 can be extended to the class of generalized uniform closed series-parallel FJQN/Bs:

**Definition 7.2.** A FJQN/B \( \mathcal{A}' = (V, E, B, M) \) is a generalized uniform closed series-parallel FJQN/B if the network obtained from \( \mathcal{A}' \) by removing all the servers and all the buffers that do not belong to a cycle is a uniform closed series-parallel FJQN/B.

Again owing to Theorem 5.3, we know that the throughput of such a generalized uniform closed series-parallel FJQN/B is independent of the initial marking of the buffers that do not belong to a cycle. Thus, under the assumptions of Corollary 7.2, the throughput of \( \mathcal{A}' \) depends only on the population, N, and the buffer size, B, of the underlying uniform closed series-parallel subnetwork of \( \mathcal{A}' \). Moreover, we have:

\[
\theta(\mathcal{A}', N) = \theta(\mathcal{A}, B - N).
\]

8. **Concavity**

In this section, we establish that the throughput of a FJQN/B is a concave function of the buffer sizes, B, and the initial marking M, provided that the service times form mutually independent sequences of i.i.d. r.v.’s having
Properties of FJQN/BS

PERT-type distributions. The concavity of the throughput with respect to the buffer sizes can also be derived by Theorem 3.4 and a result by Baccelli and Liu [1992] which states that the throughput is a concave function of the initial marking in a MG when the holding times have PERT distributions. Our new contribution is the concavity with respect to the initial marking.

Similar to Baccelli and Liu [1992], we will begin by proving the result for the case of exponential service times, and then extend the concavity result to the class of PERT-type distributions that was first introduced in Baccelli and Liu [1992]. We use the following definition of concave functions. Equivalent definitions are provided in Anantharam and Tsoucas [1990].

**Definition 8.1.** A function \( f: \mathbb{N}^k \rightarrow \mathbb{R} \) is concave iff

\[
2f(x) \geq f(x + b) + f(x - b), \quad \forall x, b \in \mathbb{N}^k,
\]

such that \( x - b \geq 0 \).

**Lemma 8.1.** Let \( \mathcal{S} = (V, E, B, M) \) be a FJQN/B. If the service times \( \{\sigma_{t,i}\}_{t=1}^{\infty} \) are mutually independent sequences of i.i.d. exponential r.v.'s with parameter \( \lambda_i, \ i \in V \), then \( \theta(\mathcal{S}) \) is a concave function of the buffer sizes \( B \) and the initial marking \( M \).

**Proof.** Owing to the memorylessness of the exponential r.v.'s, we can consider a FJQN/B \( \mathcal{S}' = (V, E, B, M) \) where all the servers are continually serving jobs. The initial timing of \( \mathcal{S}' \) is 0. Server \( i \in V \) serves (fictitious) jobs at rate \( \lambda_i \). When a (virtual) service completion occurs at server \( i \in V \), if the server is neither blocked nor starved, then this completion corresponds to a real service completion so that one job is removed from each of the upstream buffers and one job is created at each of the downstream buffers. Otherwise, no job movement takes place at this virtual service completion. It is clear that the evolution of this network is identical in law to the original FJQN/B \( \mathcal{S} \) with initial timing 0.

Observe that the behavior of \( \mathcal{S}' \) can be described as a continuous time Markov chain. We can uniformize this chain with a Poisson process having parameter \( \nu = \sum_{i=1}^{\infty} - \lambda_i \). Let \( N(t) \) denote the number of transitions in this Poisson process by time \( t > 0 \) and \( 0 = T_0 < T_1 < T_2 < \cdots < T_m < \cdots \) be the transition times. Define a r.v. \( U_{i,m} \), \( m = 1, 2, \ldots \), that is uniformly distributed in the interval \((0, 1]\) and that is independent of \( N(t) \) and let \( U_{i,m} \) be the indicator function

\[
U_{i,m} = 1_{\{\sum_{i=1}^{\infty} \lambda_i < \nu U_{i,m} \leq \sum_{i=1}^{\infty} \lambda_i\}}, \quad i \in V, \quad m = 1, 2, \ldots.
\]

Here \( U_{i,m} \) takes on value one if the \( m \)th transition corresponds to a virtual service completion at server \( i \) and zero, otherwise. A real service completion takes place only if the server is neither blocked nor starved at \( T_m \).

Let \( Q_{i,m} = Q(T_m) \) be the number (real) service completions at server \( i \) in \( \mathcal{S}' \) by time \( T_m \). For any buffer \( (i, j) \in E \) and at any time, due to the starvation constraint, the number of jobs removed by server \( j \) cannot be strictly greater than that created by server \( i \) plus \( M_{i,j} \), and due to the blocking constraint,
number of jobs created by server \( j \) cannot be strictly greater than that removed by server \( i \) plus \( B_{i,j} - M_{i,j} \). Therefore, for all \( i \in V \),
\[
Q_{i,m} \leq \min \left\{ \min_{j \in p(i)} \left( Q_{j,m} + M_{j,i} \right), \min_{j \in s(i)} \left( Q_{j,m} + B_{i,j} - M_{i,j} \right) \right\},
\]
\[m = 0, 1, 2, \ldots \] (8.2)

When the equality holds in (8.2), server \( i \) is either blocked or starved at time \( T_m \), which implies that
\[
 Q'_{i,m+1} = Q_{i,m} = \min \left\{ \min_{j \in p(i)} \left( Q'_{j,m} + M_{j,i} \right), \min_{j \in s(i)} \left( Q'_{j,m} + B_{i,j} - M_{i,j} \right) \right\}.
\]

When the inequality of (8.2) is strict, the server \( i \) is serving real jobs, so that
\[
 Q'_{i,m+1} = Q'_{i,m} + U_{i,m+1} \leq \min \left\{ \min_{j \in p(i)} \left( Q'_{j,m} + M_{j,i} \right), \min_{j \in s(i)} \left( Q'_{j,m} + B_{i,j} - M_{i,j} \right) \right\}.
\]

Therefore, for all \( i \in V \),
\[
Q'_{i,m+1} = \min \left\{ Q'_{i,m} + U_{i,m+1}, \min_{j \in p(i)} \left( Q'_{j,m} + M_{j,i} \right), \min_{j \in s(i)} \left( Q'_{j,m} + B_{i,j} - M_{i,j} \right) \right\},
\]
\[m = 0, 1, 2, \ldots \] (8.3)

It follows from (8.3) and Definition 8.1 that \( Q'_{i,m+1} \) is a concave function of \( B \), \( M \) and \( Q'_{i,m}, i \in V \). An induction argument leads us to conclude that for all \( i \in V \) and all \( m \geq 0 \), \( Q'_{i,m} \) is a concave function of \( B \) and \( M \) (noting that \( Q'_{i,0} = 0 \)).

Since \( Q'(t) = Q'_{i,m} \) for \( T_m \leq t < T_{m+1} \), we deduce that the variables \( Q'(t) \) are also concave functions of \( B \) and \( M \) as \( T_m \) does not depend on either of these vectors. Since concavity is preserved in the limit and \( Q'(t) \) is equal to \( Q(t) \) in distribution for all \( t \geq 0 \), we have that \( \lim_{t \to \infty} Q(t) \) is a concave function of \( B \) and \( M \) for all \( i \in V \). We conclude that \( \theta(\mathcal{F}) \) is also concave in \( B \) and \( M \).

**Remark.** Note that the preceding result holds for values of \( B \) and \( M \) such that \( \mathcal{F} \) is nondeadlock-free as well as when \( \mathcal{F} \) is deadlock-free. This is because, although the recurrence relations for \( D_{i,a} \), (5.4) and (5.5), are valid only when \( \mathcal{F} \) is deadlock-free, (8.3) is valid for all values of \( B \) and \( M \) such that \( M \leq B \). In the case that \( \mathcal{F} \) is nondeadlock-free, there exists an integer \( m_0 > 0 \) such that \( Q'_{i,m} = Q'_{i,m_0}, \forall m > m_0 \). This is equivalent to the property that the precedence graph associated with \( \mathcal{F}, \mathcal{F}_j \), contains a cycle (cf. Lemma 5.1).

**Definition 8.2.** A stochastic PERT graph is a weighted directed acyclic graph where the weights are r.v.’s associated with the vertices. The weight of the critical path of the stochastic PERT graph is the maximum of the weights of all the paths in the graph.
Definition 8.3. Random variable $X$ has a PERT-type distribution if $X$ can be expressed as the weight of a critical path of a stochastic PERT graph $G$ where the weights of the vertices are mutually independent exponential r.v.'s. The distribution will be denoted $F(G, \lambda_1, \ldots, \lambda_{|G|})$, where $\lambda_1, \ldots, \lambda_{|G|}$ are the parameters of the exponential distributions of the vertices of $G$.

Clearly, the Erlang distribution belongs to the class of PERT-type distributions. Baccelli and Liu [1992b] have shown that all PERT-type distributions are log concave.

Theorem 8.1. If the service times of $\text{FJQN}/B$ are mutually independent sequences of i.i.d. r.v.'s having PERT-type distributions, then $\theta(\mathcal{F})$ is a concave function of $B$ and $M$.

Proof. The proof is similar to that of Baccelli and Liu [1992b]. It consists in showing that it is possible to simulate such a $\text{FJQN}/B$ by another $\text{FJQN}/B$, $\mathcal{F}'$, in which all service times are exponential r.v.'s and applying the previous result. The detailed proof is found in Appendix E. □

9. Applications and Extensions

In this section, we discuss several applications of the results to the modeling, performance evaluation and optimization of computer, communication, and manufacturing systems. We also illustrate how our results on $\text{FJQN}/B$s can be applied to other blocking mechanisms and to some other queuing models with buffers of infinite capacity or with unreliable servers.

9.1 Throughput Maximization. The symmetry and concavity properties derived earlier can be used to determine the initial markings that maximize the throughput for the class of generalized uniform closed series-parallel $\text{FJQN}/B$s. In particular, using the terminology introduced in the previous section, we have the following result:

Theorem 9.1. If the service times are mutually independent i.i.d. sequences of r.v.'s having PERT distributions, then the throughput of a generalized uniform series-parallel $\text{FJQN}/B$ is maximized by any initial marking that satisfies $N' = \lfloor B'/2 \rfloor$ or $N' = \lceil B'/2 \rceil$, where $N'$ and $B'$ are the population and the buffer size, respectively, of the largest uniform closed series-parallel subnetwork of $\mathcal{F}$.

Proof. This follows from the symmetry and concavity property of this network. □

This result has both network and manufacturing applications. For example, in networks, window flow control is often modeled by a closed tandem queuing network where the number of jobs corresponds to the window size. Hence, the above theorem can be interpreted to mean that in the case of finite buffer capacities, the window size should be one half of the total buffer capacity in order to maximize throughput. Automated manufacturing lines where parts are fixed onto pallets can also be modeled by a closed tandem queuing network [Dallery and Gershwin 1992]. The above result implies that the number of pallets should be chosen to be one half of the total storage capacity of the manufacturing system. This result also applies to certain assembly and disassembly lines that can be modeled by generalized uniform closed series-parallel $\text{FJQN}/B$s.
9.2 Parallel Processing Systems. Consider the parallel processing system analyzed in Baccelli and Liu [1990]. In such a system, there are \( M \geq 1 \) processors and a sequence of parallel programs represented by the same directed circuit-free graph, referred to as the task graph, \( G = (V, E) \). The vertices of \( V \) denote the tasks in a parallel program and the edges represent the precedence constraints between the tasks: \((i, j) \in E\) indicates that task \( j \) cannot be started before \( i \) is finished. The task running times are random variables. The \( n \)th program in the sequence is represented by the \( n \)th instantiation of the task graph.

It is assumed that the processors and/or the multiprocessor system have limited multiprogramming degrees specified by \( d_1, \ldots, d_M \) and \( d \), respectively. viz., at most \( d_i \) (respectively, \( d \)) programs are allowed to have unfinished tasks on processor \( i \) (respectively, in the system). Furthermore, it is assumed that there is an infinite supply of parallel programs and that whenever there is space for a new one in the system, it is always available.

There is a static assignment identical for all parallel programs that, subject to the multiprogramming degrees, assigns tasks to the processors. Associated with the static assignment is a static local scheduling that, subject to the precedence relations, orders the executions of the tasks belonging to the same program that are assigned to the same processor. Let \( G_0 = (V, E_0) \) be the resulting task graph after the application of the static assignment and local scheduling (i.e., \( E_0 \) is composed of \( E \) and the edges between the tasks on the same processors due to the local scheduling). For all \( 1 \leq i \leq M \), let \( b_i \) (respectively, \( e_i \)) be the first (respectively, last) task to be executed on processor \( i \) among those assigned to the same processor.

The above parallel processing system can readily be modeled by a FJQN/B \( \mathcal{F} = (V, E', B, M) \). Assume that only the processors have the limited multiprogramming degree and not the system (\( d = \infty \)). The other cases can be analyzed similarly. Then, \( E' = E_0 + \{(e_i, b_i) \mid 1 \leq i \leq M\} \), the edges of \( E \) have buffers of size \( \tilde{d}_i = \max_{1 \leq j \leq M} d_j \), and initial marking 0, the edges \( (e_i, b_i) \), \( 1 \leq i \leq M \), have buffers of size \( d_i \), and initial marking \( d_i \).

Applying the reversibility property to such a parallel processing model implies that when the task graph is reversed, the program completion times are the same in law as those of the original task graph, so that the throughput is also the same. Moreover, if there is an optimal static scheduler that minimizes the completion times or maximizes the throughput of a task graph, then the same assignment and the reversed ordering is optimal for the reversed task graph (when the task graph is a tree, such a result can be considered as the stochastic version of the equivalence between the optimal scheduling of in-tree and out-tree for the minimization of makespan). According to the concavity property, the throughput in terms of the programs is a concave function of the multiprogramming degrees.

9.3 Systems with Population Constraints. The multiprogramming degrees in the above example is a population constraint. In some communication networks and manufacturing systems, there are also such constraints. In general, consider a FJQN/B \( \mathcal{F} \). Suppose one would like to add a restriction on the total number of jobs that can be present between a pair of servers, say \( i \) and \( j \). For the sake of simplicity, we assume that there is a single path between these two servers. Let \( B_0 \) be the total buffer capacity among all of the buffers
on the path between servers $i$ and $j$. Server $i$ is prevented from working if the total number of jobs between $i$ and $j$ is equal to a given value, say $P$, with $P < B_{ij}$. This population constraint can easily be modeled by adding a buffer between server $i$ and server $j$ having a capacity $B_{ij} = P$. The initial marking of the buffer is equal to the sum of the initial markings of the buffers between server $i$ and server $j$. Let $\mathcal{P}''$ denote the resulting FJQN/BS.

Consider the reversibility property. Let $\mathcal{S}'$ be the reverse of $\mathcal{S}$. Let $\mathcal{S}''$ be the network obtained from $\mathcal{S}'$ by adding a population constraint of size $P$ between server $j$ and server $i$ in the same way as for $\mathcal{S}'$. Now, it is easy to check that $\mathcal{S}''$ is the reverse of $\mathcal{S}'$ and as a result they have the same throughput. So, $\mathcal{S}'$ with a population constraint in between servers $j$ and $i$ has the same throughput as $\mathcal{S}''$ with the same population constraint between servers $i$ and $j$. Thus, reversibility holds for FJQN/BSs with population constraints.

Consider now the symmetry property. Let $\mathcal{S}'''$ denote the symmetrical network of $\mathcal{S}''$. Let $\mathcal{S}'''$ be the network obtained from $\mathcal{S}''$ by adding the population constraint of size $P$ between server $i$ and server $j$ in the same way as for $\mathcal{S}'$. Now, one can check that $\mathcal{S}'''$ is not the symmetrical network of $\mathcal{S}'''$ and as a result they do not have the same throughput. So, $\mathcal{S}''$ with a population constraint of size $P$ between servers $i$ and $j$ does not have the same throughput as $\mathcal{S}'''$ with the same population constraint. Thus, symmetry does not hold for FJQN/BSs with population constraints.

As in the previous subsection, the concavity property holds in $\mathcal{S}'''$. In particular, one obtains that the throughput is a concave function of the population constraints.

9.4 Modeling of Other Blocking Mechanisms. Throughout this paper, we considered FJQN/BSs operating under the so-called blocking-before-service (BBS) mechanism [Perros 1989]. That is, a server is allowed to initiate a service period only if there is a space available in each of its downstream buffers. Although this type of blocking is of interest in many applications, other blocking mechanisms are also useful, in particular blocking-after-service (BAS) and repetitive blocking (RB) [Perros 1989]. We now briefly discuss how some of the results presented in this paper can be applied to such blocking mechanisms. We restrict our attention to open and closed tandem queuing networks. In other words, we only consider networks that do not involve fork–join operations. In the case of fork–join operations, careful attention must be paid to the definition of these blocking mechanisms that is beyond the scope of this paper.

Consider first the case of repetitive blocking. In networks operating under RB, a server can also operate when the downstream buffer is full. However, if the downstream buffer is full at the instant of service completion, the server has to repeat its service. Assume that the service times at the different servers are i.i.d. exponential r.v.'s. Then, due to the memoryless property of the exponential distribution, the behavior of the network operating under repetitive blocking is identical to that of the same network operating under blocking-before-service. As a result, all of the properties derived in this paper, especially reversibility, symmetry, and concavity, also hold for networks with exponentially distributed service times and operating under repetitive blocking. Moreover, the optimal initial marking obtained in Theorem 9.1 also holds.

Consider now the case of blocking-after-service. In networks operating under BAS, a server can operate even though the downstream buffer is full. The
blocking of the server will only occur at the instant of service completion if the buffer is full at that time. Most of the properties derived in this paper can also be established for networks with BAS, especially the reversibility and concavity (however, there are no duality of symmetry properties for such networks). Although these results (i.e., reversibility and concavity) could be obtained in a similar way as we did in this paper, a simpler approach is to use an equivalent representation (in terms of sample-path behavior) of a network with BAS by a network operating under BBS. Details are reported in Dallery et al. [1992]. The basic idea is given below. Consider a particular server, say $i$. Let $i-1$ and $i+1$ denote the indexes of its upstream and downstream buffers respectively. Let $B_{i-1,d}$ and $B_{i+1}$ be the capacities of the corresponding intermediate buffers. The equivalent network with BBS is then obtained by adding two dummy servers, say $u$ and $d$, one in between $i-1$ and $i$ and the other in between $i$ and $i+1$. These two servers have zero service times. The buffer capacities are defined as follows: $B_{i-1,u} = B_{i-1,d} - 1$; $B_{u,d} = 1$; $B_{i,d} = 1$; $B_{d,i+1} = B_{i+1}$. In addition, there is a buffer between $u$ and $d$ with capacity $B_{u,d} = 1$. This buffer ensures that there is at most one job in the buffers $(u,i)$ or $(i,d)$. A job in buffer $(u,i)$ is currently receiving service while a job in buffer $(i,d)$ has already completed its service but cannot be transferred into the downstream buffer because it is full.

9.5 SYSTEMS WITH UNRELIABLE SERVERS. Our results can also be applied to performability analysis. In such a framework, the systems are composed of servers prone to failures. For the sake of simplicity, we restrict our attention to the case where for each server, the service times, the times to failures (up time) and the times to repair (down time) are i.i.d. random variables. Also, we assume that failures are operation dependent, that is a failure can occur only when the server is working. Whenever the server is idle (starved or blocked), the failure process is stopped. In this case, the service period of a server corresponds to the so-called completion time [Gaver 1962; Nicola 1986], that is, the total time between the beginning and the completion of the operation. This time is the sum of the service time and the repair times corresponding to the failures that occur during the processing, if any. Suppose now that the time to failure is exponentially distributed. In this case, it can be shown (see, e.g., [Nicola 1986]) that the resulting service period are i.i.d. random variables. Thus, all the results presented in the paper can still be applied.

9.6 STRONGLY CONNECTED MARKED GRAPHS. There are various other applications of our results that can be obtained from, for instance, the equivalence between FJQN/Bs and SCMGs (cf. Theorems 3.3 and 3.4). In fact, SCMGs are widely used to model concurrent activities in communication networks, computer systems and manufacturing systems (see the survey of Murata [1989]).

Our results on the reversibility of the throughput of FJQN/Bs apply directly to these models. However, the duality results for FJQN/Bs appear to not have any intuitive interpretation in the context of SCMGs. This is because duality only makes sense for models having explicit capacity constraints, that is, finite buffers. Consequently, the symmetry results also do not have any intuitive interpretation. Owing to Theorem 3.3, the concavity of the throughput with respect to the initial marking of a FJQN/B implies that the throughput is a
concave function of the initial marking in a SCMG when the holding times have PERT distributions, which was first obtained in Baccelli and Liu [1992b].

9.7 SYSTEMS WITH INFINITE BUFFERS. Throughout the paper, we assumed that all buffers have finite capacity. It is however possible to handle certain FJQN/BSs with infinite buffers provided that they can be transformed into FJQN/BSs with all buffers having finite capacity. Let us first establish the following result:

Lemma 9.1. Consider a FJQN/B having some infinite buffers. Consider any infinite buffer \( k \) that is part of a cycle \( C \). Define an orientation of the cycle according to the direction of the flow of jobs through this buffer. Then, if all of the buffers oriented in the reverse direction have finite capacity, this infinite buffer can equivalently be replaced by a finite buffer with capacity set equal to the invariant of the cycle, that is, \( B_k = I^*_C(M) \).

Proof. Since all buffers in the reverse direction have finite buffers, it is easy to check that Eq. (4.1) still holds and that the quantity \( I^*_C(M) \) is finite. Now, Eq. (4.1) implies that the number of jobs in buffer \( k \) at any time, \( m_k(t) \), is bounded by \( I^*_C(M) \). Thus, the buffer \( k \) can be replaced by a buffer of finite capacity \( I^*_C(M) \) without modifying the behavior of the network. \( \square \)

Consider a FJQN/B with a subset of buffers having infinite capacity. The transformation described in Lemma 9.1 can be applied repeatedly to any infinite buffer that is part of a cycle such that all buffers in the reverse direction have finite buffers. We note that an infinite buffer that did not originally satisfy the condition in Lemma 9.1 may satisfy it after a certain number of transformations on other infinite buffers have been performed. So, this procedure must be applied as long as the resulting FJQN/B has at least one infinite buffer that can be transformed into a finite buffer.

We note that any infinite buffer that is part of a circuit can always be replaced by a finite buffer since, in this case, the condition in Lemma 9.1 is obviously satisfied (there is no buffer in the reverse direction). On the other hand, any infinite buffer that is not part of a cycle can never be replaced by a finite buffer.

A typical example for which this transformation can be performed is a closed tandem queuing network with infinite buffer capacity. Let \( N \) be the total number of jobs in the network. As all of the buffers are in the single circuit, according to Lemma 9.1, all of the buffers of infinite capacity can be replaced by buffers having a finite capacity \( N \). More generally, any closed series-parallel fork-join network with infinite buffers can be transformed into a FJQN/B with all finite buffers.

Now, if the final FJQN/B is such that all buffers have finite capacity, all of the results presented in the paper can be applied to it. Some of these results can then be reinterpreted on the original network. This is the case for the results pertaining to the throughput, the reversibility, and the concavity (with respect to the initial marking) properties presented in Sections 5, 6, and 8. On the other hand, the symmetry property has no interpretation for the original network since the buffer capacity vector \( B \) of the final FJQN/B is a function of the initial marking \( M \).
Finally, we note that the behavior of a FJQN/B having infinite buffers that cannot be transformed into finite buffers is more complex. In fact, such FJQN/Bs are equivalent to (nonstrongly connected) Marked Graphs, so that we are faced with the problem of stability. Results obtained by Baccelli [1992], Baccelli and Liu [1990], and Baccelli et al. [1989] can be applied to these FJQN/Bs.

9.8 Systems with Multiple Server Queues. All of these results presented in the paper hold for single servers. In the case of multiple servers, it is easy to check that the results pertaining to equivalence and qualitative properties (Sections 3 and 4) hold. In particular, a FJQN/B with multiple servers has the same behavior as that of any $\Delta$-dual. This again follows from the job/hole duality. It should, however, be emphasized that duality holds under the convention that there is no space on the servers; that is, the capacity of the buffers includes the server spaces. On the other hand, the results pertaining to the existence, the reversibility and the symmetry properties of the throughput cannot easily be extended to multiple servers, as Kingman’s theorem on subadditive processes does not apply. The problem we are faced with is that overtaking may occur, which was already pointed out in the case of open tandem queuing networks with blocking after service in Yamazaki et al. [1985]. However, Eqs. (5.4) and (5.5) are easily modified in the case that all multiple servers have deterministic service times. The blocking mechanism is understood not to allow a job to begin service if the number of jobs either in a downstream buffer or already in service at the station in question equals the capacity of that downstream buffer. If $c_i$ denotes the number of servers associated with $i \in V$, then the resulting equations are

$$D_{i,n}(\mathcal{S}, Y) = \sigma_{i,n} + \max \left( Y_i, \max_{j \in p_i} D_{j, 1-M_j(\mathcal{S}, Y)}, \max_{k \in s_i} D_{k, 1-(B_{i,k} - M_{i,k})(\mathcal{S}, Y)} \right), \quad n = 1, \ldots, c_i,$$

$$D_{i,n}(\mathcal{S}, Y) = \sigma_{i,n} + \max \left( D_{i,n-c_i}(\mathcal{S}, Y), \max_{j \in p_i} D_{j,n-M_j}(\mathcal{S}, Y), \max_{k \in s_i} D_{k,n-(B_{i,k} - M_{i,k})(\mathcal{S}, Y)} \right), \quad n > c_i,$$

where, by convention, $D_{i,0}(\mathcal{S}, Y) = 0$, $n \leq 0$.

In this case, Kingman’s theorem still applies, and all of the results presented in this paper, in particular reversibility and symmetry, still hold.

10. Conclusions

In this paper, we have analyzed quantitative as well as qualitative properties of Fork–Join Queueing Networks with Blocking. We have established results regarding the equivalence of the behavior of a FJQN/B and that of its duals and a strongly connected marked graph. We have established that the asymptotic throughput exists when the sequences of service times are stationary and ergodic, and that this throughput is independent of the initial marking when the sequences of service times are mutually independent. We have shown that
when the sequences of service times are further reversible, the reverse of a FJQN/B has the same throughput as the original network, which, in turn, implies that a FJQN/B has the same throughput as that with the symmetrical initial marking. Finally, we have proved that when the service times are PERT type, the throughput is a concave function of the buffer sizes and the initial marking. Our results can be applied to the modeling, performance evaluation and optimization of computer, communication, and manufacturing systems.

Extensions of some of the results presented in this paper to fork–join networks with different operating mechanisms are reported in Dallery et al. [1992]. Another extension, which is currently under investigation, is the introduction of random routing mechanisms in FJQN/Bs.

Algorithm A. A Proof of Theorem 3.2

Let us first consider the relationship between the markings in \( \mathcal{S}^d \) and \( \mathcal{S}^c \), and let us focus our attention on a particular pair of servers \( i, j \in V \) such that \( |S_{i,j}| \geq 1 \). Observe that, since by construction of \( \mathcal{S}^d \), all the buffers between any pair of servers are oriented in the same direction, the number of jobs (and the number of holes as well) are increased or decreased simultaneously in all the buffers between servers \( i \) and \( j \). Thus, we have:

\[
m^d(t) - m^d(t) = m^d(t) - m^d(t), \quad \forall k, k' \in S_{i,j}. \tag{A.1}
\]

Let \( q_{i,j}, q'_{i,j} \in S_{i,j} \) be any two buffers such that:

\[
M^d_{q_{i,j}} = \min_{l \in S_{i,j}} M^d_l, \quad B^d_{q_{i,j}} - M^d_{q_{i,j}} = \min_{l \in S_{i,j}} (B^d_l - M^d_l). \tag{A.2}
\]

Note that \( q_{i,j} \) (respectively, \( q'_{i,j} \)) is one of the buffers in \( S_{i,j} \) that has the smallest number of jobs (respectively, holes) in the initial marking. It then follows from (A.1) and (A.2) that for all \( t \geq 0 \):

\[
m^d_{q_{i,j}}(t) = \min_{l \in S_{i,j}} m^d_l(t), \quad B^d_{q_{i,j}} - m^d_{q_{i,j}}(t) = \min_{l \in S_{i,j}} (B^d_l - m^d_l(t)). \tag{A.3}
\]

Consequently, server \( j \) is starved at some time \( t \geq 0 \) by one of the buffers of \( S_{i,j} \), if and only if \( m^d_{q_{i,j}}(t) = 0 \), and server \( i \) is blocked at some time \( t \geq 0 \) by one of the buffers of \( S_{i,j} \), if and only if \( B^d_{q_{i,j}} + m^d_{q_{i,j}}(t) = 0 \).

Now, the parameters of the corresponding buffer in \( \mathcal{S}^c \), that is, \( k_{i,j} \), as defined in the algorithm given in Section 3.2, are such that:

\[
M^c_{k_{i,j}} = M^d_{q_{i,j}} \quad B^c_{k_{i,j}} - M^c_{k_{i,j}} = B^d_{q_{i,j}} - M^d_{q_{i,j}}.
\]

Thus, the parameters of buffer \( k_{i,j} \) (capacity and initial marking) are such that a starvation of server \( j \) (respectively, a blocking of server \( i \)) in \( \mathcal{S}^d \) because buffer \( q_{i,j} \) is empty (respectively, \( q'_{i,j} \) is full) occurs exactly at the same time as a starvation of server \( j \) (respectively, a blocking of server \( i \)) in \( \mathcal{S}^c \) because buffer \( k_{i,j} \) is empty (respectively, full). Since this is true for every pair of servers \( i, j \in V \), it is readily shown by induction on the service periods that
DALLERY ET AL.

the following equivalence holds between the sample paths of $\mathcal{S}^d$ and $\mathcal{S}^c$; for all $t \geq 0$, we have

$$m^d_i(t) - M^d_i = m^c_i(t) - M^c_i,$$

\quad $\forall k \in V_b$ such that $(i, k), (k, j) \in E^d$;  \quad (A.4)

and

$$Q^d_i(t) = Q^c_i(t), \quad \forall i \in V_i.$$

The equivalence between the behavior of $\mathcal{S}$ and that of $\mathcal{S}^{-1}$, expressed in Theorem 3.2, then follows from Theorem 3.1. \(\square\)

Appendix B. A Proof of Theorem 4.3

Without any loss of generality, we assume that the FJQN/B under consideration is in its canonical form. It is clear that, when a server $i$ is prohibited from serving any job, all other servers will be blocked, starved, or both, after a bounded number of service periods. Let $m^1$ and $m^2$ denote the markings of $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively, when such a situation occurs. We will prove $m^1 = m^2$ by contradiction.

We shall refer to the components of $m^1$ as $m^1_{i,j}$, $j = 1, 2$ and $(i_1, i_2) \in E$. Assume that $m^1 \neq m^2$. Then there exists at least one buffer $(j, k) \in E$ such that $m^1_{j,k} \neq m^2_{j,k}$ and there is either a chain $i_1, i_2, \ldots, i_{n-1} = j, i_n = k$ or $i_1 = i, i_2, \ldots, i_{n-1} = k, i_n = j$, where all of the buffers on the chain, except $(j, k)$ have the same markings in both $m^1$ and $m^2$. We assume that the chain is $i_1 = i, i_2, \ldots, i_{n-1} = j, i_n = k$. The other chain can be treated in a similar way. There are two cases depending on whether $(j, k)$ is part of a cycle or not:

(i) $(j, k)$ is not part of a cycle. In this case, it is not difficult to show that $m^1_{j,k} = m^2_{j,k} = 0$, which contradicts the assumption that $m^1_{j,k}$ differs from $m^2_{j,k}$.

(ii) $(j, k)$ is part of a cycle. Without loss of generality, we assume that $m^1_{j,k} > m^2_{j,k}$ \(\geq 0\). Beginning with the chain $i_1, i_2, \ldots, i_n$, the following algorithm constructs a chain $P$ that contains a cycle $C$ in $\mathcal{S}$ which will either be deadlocked or have a different invariant measure in $\mathcal{S}_1$ than $\mathcal{S}_2$. Both of these contradict the assumptions made in the theorem.

Algorithm:

1. initialize
   
   $direction := downstream; P = (i_1, \ldots, i_n); r = i_0$;

2. main body of algorithm.
   
   if $direction = downstream$ then
     
     if there exists $n \in s_i(i_t)$ such that $m^1_{i_t,n} = B_{i_t,n}$
     
     then choose $n$ to be part of the chain $P$;
     
   else $direction := upstream; go to 2$;

   else
     
     if there exists $n \in p_i(i_t)$ such that $m^1_{i_t,n} = 0$
     
     then choose $n$ to be part of the chain $P$;

   else $direction := downstream; go to 2$;

   if $n = i_t \in P$ then return $C = (i_t, i_{t+1}, \ldots, i_r)$ and stop;

   else $r = r + 1; i_r = n; P := P + (i_r);$ go to 2.
First, we show that this algorithm will terminate. Consider any server \( a \in V_s \), \( a \neq i \), that is contained within one or more cycles. According to the assumption of the theorem it is either starved, blocked, or both. Hence, either \( g(a) \) contains a buffer that is empty, or \( s(a) \) contains at least one full buffer, or both. Hence, whenever we examine the last server on chain \( P \), \( i_r \), we will always find either a server \( n \in s(i_r) \) such that buffer \( (i_r, n) \) is full or a server \( n \in g(i_r) \) such that buffer \( (n, i_r) \) is empty. Consequently, the algorithm always identifies a new server during each execution of 2. Clearly, the algorithm will terminate as \( V \) contains a finite number of servers.

It should be clear from the behavior of the algorithm that it never returns \( C = (i_{r-1}, i_r) \). This is because whenever \( (i_{r-1}, i_r) \in E \), then \( m^1_{i_{r-1}, i_r} > 0 \), and whenever \( (i_r, i_{r-1}) \in E \), then \( m^1_{i_r, i_{r-1}} = 0 \). Consequently \( i_{r-1} \) can never be a candidate for the next server on the chain after \( i_r \).

This algorithm produces a cycle that is in one of two forms depending on the position of \( i_l \) with respect to \( i_r \). If \( l \geq l_0 \), then we claim that the algorithm produces a cycle in which all of the buffers oriented in one direction are empty and all of the buffers oriented in the opposite direction are full. Consequently, \( \mathcal{L} \) in conjunction with marking \( m \) is deadlocked. This contradicts the assumption that \( \mathcal{L} \) was deadlock free.

Consider the case that \( l < l_0 \). Orient the cycle so that \( (i_{l-1}, i_l) \) is pointed upstream within the cycle. The buffers in the cycle divide into five groups, \( V_0 \) contains \( (i_{l-1}, i_l) \), \( V_1 \) contains all of the buffers in the cycle between server \( i_l \) and server \( i_{l-1} \) with the same orientation as \( (i_{l-1}, i_l) \) within the cycle, \( V_2 \) contains all of the buffers in the cycle between server \( i_l \) and server \( i_{l-1} \) with orientations opposite to \( (i_{l-1}, i_l) \) within the cycle, \( V_3 \) contains all of the remaining buffers with the same orientation as \( (i_{l-1}, i_l) \) not in \( V_0 + V_1 \), and \( V_4 \) contains all of the buffers not in \( V_2 \) that have an orientation opposite to that of \( (i_{l-1}, i_l) \). Now the invariant for this cycle is

\[
I^1_c(m_1) = m^1_{i_l, i_{l-1}} + \sum_{b \in V_1} m^1_b + \sum_{b \in V_2} (B_b - m^1_b) + \sum_{b \in V_3} B_b + \sum_{b \in V_4} B_b,
\]

\[
\geq m^1_{i_l, i_{l-1}} + \sum_{b \in V_1} m^1_b + \sum_{b \in V_2} (B_b - m^1_b) + \sum_{b \in V_3} m^2_b + \sum_{b \in V_4} (B_b - m^2_b),
\]

\[
> m^2_{i_l, i_{l-1}} + \sum_{b \in V_1} m^1_b + \sum_{b \in V_2} (B_b - m^1_b) + \sum_{b \in V_3} m^2_b + \sum_{b \in V_4} (B_b - m^2_b),
\]

\[
= I^2_c(m_2).
\]

Consequently \( m_1 \not\sim m_2 \), which can only occur if \( M_1 \not\sim M_2 \), which contradicts the statement of the theorem. □

Appendix C. A Proof of Theorem 5.3

PROOF. Consider an initial timing condition \( Y \) where \( Y_i = \infty \) for some server \( i \in \mathcal{L} \) and \( Y_j = 0 \) for \( j \in V_v \setminus \{i\} \). According to Theorem 4.3, \( \mathcal{L}^{-1} \) and \( \mathcal{L}^{-2} \) will reach the same state \( M \) under the initial timing condition. Let \( \mathcal{L} = (\mathcal{L}, M) \) be another FJQN/BS, which differs from \( \mathcal{L}^{-1} \) and \( \mathcal{L}^{-2} \) only in its initial marking. We will show that

\[
\theta(\mathcal{L}^{-1}) = \theta(\mathcal{L}) = \theta(\mathcal{L}^{-2}).
\]
For this, denote by $k_j, j \in V_i$, the number of times the server $j$ has served in $\mathcal{M}_i$ prior to reaching the final state $M$ when the initial marking is $Y (k_j = 0)$. Define a new initial timing vector $Y'$ by $Y'_j = 0$ for $j \in V_i - \{i\}$, and

$$Y'_j = \sum_{j \in V_i} \sum_{t=1}^{k_j} \sigma_{j,t},$$

where $\sum_{t=1}^{k_j} \sigma_{j,t} = 0$ when $k_j = 0$.

Let $\mathcal{M}_i = (\mathcal{M}_i, Y')$ and $\mathcal{M} = (\mathcal{M}, \mathbf{0})$. The sequences of the service times associated to $\mathcal{M}_i$ and $\mathcal{M}$ are referred to as $\{\sigma_{j,n}\}_{n=1}^{\infty}$, and $\{\sigma_{j,n}\}_{n=1}^{\infty}$, $j \in V_i$, respectively. Denote by $m_i(t)$ and $m(t)$ the markings of $\mathcal{M}_i$ and $\mathcal{M}$ at time $t \geq 0$, respectively. It follows from Theorem 4.3 that $m_i(Y'_i) = m(0)$.

Since the sequences of the service times are independent and stationary, we can couple the service times in $\mathcal{M}_i$ and $\mathcal{M}$ in the following way:

$$\sigma_{j,n} = \sigma_{j,k_j+n}, \quad n \geq 1, \quad j \in V_i.$$

Clearly, the sequences $\{\sigma_{j,n}\}_{n=1}^{\infty}, j \in V_i$ defined as above are also independent and stationary. Under such a coupling, one sees that

$$m(t) = m_i(Y'_i + t), \quad t \geq 0,$$

and

$$D_{j,n} (\mathcal{M}_i, \mathbf{0}) = D_{j,k_j+n} (\mathcal{M}_i, Y'), \quad n \geq 1, \quad j \in V_i.$$

Therefore, an application of Theorem 5.2 yields

$$\theta(\mathcal{M}_i) = \theta(\mathcal{M}).$$

Analogously, one can prove that $\theta(\mathcal{M}_i^r) = \theta(\mathcal{M}^r)$, which entails that $\theta(\mathcal{M}_i) = \theta(\mathcal{M}_i^r)$. 

**Appendix D. A Proof of Theorem 6.1**

Before proceeding with the proof of Theorem 6.1, we establish some preliminary results.

Let $\mathcal{M}' = (V, E', B', M')$ be the reverse FJQN/B of $\mathcal{M} = ((V, E, B), M)$. Let $p'_i(i) = s_i(i)$ and $s'_i(i) = p_i(i)$ denote the sets of predecessor and successor servers of server $i$ in $\mathcal{M}'$, respectively. Denote by $\sigma'_{j,n}$ the $n$th service time of server $i$ in $\mathcal{M}'$.

It follows from Lemma 5.2 that for all $i \in V_i, n \geq 1$,

$$D_{i,n} (\mathcal{M}', \mathbf{0}) = \sigma'_{j,n} + \max \left\{ D_{i,n-1} (\mathcal{M}', \mathbf{0}), \right.$$

$$\max_{l \in p'_i(i)} D_{l,n-M'_i} (\mathcal{M}', \mathbf{0}),$$

$$\max_{k \in s'_i(i)} D_{k,n-(B'_i+M'_i, \mathcal{M}', \mathbf{0})} \left( \mathcal{M}', \mathbf{0} \right) \right\} \quad (D.1)$$

where, by convention, $D_{i,n} (\mathcal{M}', \mathbf{0}) = 0$, $n \leq 0$. 

$\square$
Denote by \(<_s, \mathcal{R}_s = (\mathcal{R}_s', \mathcal{R}_s')\) the precedence relation and the precedence graph associated with \(\mathcal{R}_s', \mathcal{R}_s'\), respectively, viz. \((i, n) <_s (j, m)\) iff
\[
\begin{align*}
n &= m - M_{i,j}, \quad i \in p_i'(j), \\
n &= m - 1, \quad i = j, \\
n &= m - (B_{j,i} - M_{j,i}), \quad i \in s_i'(j),
\end{align*}
\]
and
\[
\mathcal{R}_s' = \{ (i, n) \mid n \geq 1, i \in V_s \},
\]
\[
\mathcal{R}_s'' = \{ (i, n) \rightarrow (m, j) \mid (i, n), (j, m) \in \mathcal{R}_s', (i, n) <_s (j, m) \}.
\]
It follows from the fact that \(\mathcal{R}_s\) is deadlock-free and Lemmas 5.1 and 6.1 that \(\mathcal{R}_s'\) is acyclic. Let \(\mathcal{P}'\) be the set of paths of \(\mathcal{R}_s'\). Similar to Lemma 5.3, we have
\[
\text{LEMMA D.1.} \quad \text{For all } i, j \in V_s, \text{ and all } n, m, 1 \leq l \leq n \geq 1, (j, l) <_s (i, m) \Leftrightarrow (i, n - m) <_s (j, n - l).
\]
\text{PROOF.} Suppose first \((j, l) <_s (i, m)\). If \(j \in s_i(i)\), then
\[
m - l = B_{j,i} - M_{i,j} = B_{j,i}' - M_{i,j}',
\]
and \(j \in p_i'(i)\) so that \(i \in s_i'(j)\). Using (D.4) together with (D.6), we obtain that
\[(i, n - m) <_s (j, n - l)\).
The remainder of the proof can be carried out in an analogous manner. \(\square\)

\text{PROOF OF THEOREM 6.1.} For any \(n \geq 1\), set \(\sigma_{i,m} = \sigma_{i,n+1-m}, m = 1, 2, \ldots, n\). We rewrite relation (D.5) as follows:
\[
D_n(\mathcal{R}'_s, 0) = \max_{P = ((t_1,n_1) \rightarrow \cdots \rightarrow (t_k,n_k)) \in \mathcal{P}'_s, n_1 + n_k - n = n} \sum_{h=1}^{k} \sigma_{t_h,n_1+1-n_h} \quad \forall n \geq 1. \quad (D.5)
\]
An application of Lemma D.1 yields
\[
D_n(\mathcal{R}'_s, 0) = \max_{P = ((i, n+1-n_k) \rightarrow \cdots \rightarrow (i, n+1-n_1)) \in \mathcal{R}_s'} \sum_{h=1}^{k} \sigma_{i,n+1-n_h} = D_n(\mathcal{R}'_s, 0).
\]
According to the reversibility of the service times, the sequences \(\{\sigma_{i,m}^{(n)}\}_{m=1}^{n}, i \in V_s\) have the same joint distribution as \(\{\sigma_{i,m}^{(n)}\}_{m=1}^{n}, i \in V_s\). Thus
\[
D_n(\mathcal{R}_s', 0) = \rho D_n(\mathcal{R}_s, 0), \quad n \geq 1. \quad (D.8)
\]
Appendix E. A Proof of Theorem 8.1

Let the distributions at server $i$ be $F(G_{i}, \lambda_{1,i}, \ldots, \lambda_{n_{i},i})$ where $n_{i} = |G_{i}|$, $i \in V_i$ and let $E_{i}$ denote the set of edges within $G_{i}$, $i \in V$. For simplicity, we assume that there is only one source and one sink in $G_{i}$ labeled 1 and $n_{i}$, respectively. Multiple source/sink PERT-type distributions can be handled in a similar manner.

Our new FJQN/B will have a node representing each node in each stochastic PERT graph $G_{i}$. Each node will be represented by $\langle i, j \rangle$, where $i$ denotes the PERT graph $G_{i}$ and $j$ the node within that graph. The FJQN $\mathcal{F}'$ is constructed as follows:

\[ V' = \{ \langle i, j \rangle | 1 \leq j \leq n_{i}, i \in V_{i} \}, \]
\[ E' = \{ \langle \langle i, j \rangle, \langle i, l \rangle \rangle | (j, l) \in E_{i}, i \in V_{i} \} + \{ \langle \langle i, n_{i} \rangle, \langle k, 1 \rangle \rangle | (i, k) \in E \} + \{ \langle \langle i, 1 \rangle, \langle k, n_{i} \rangle \rangle | (i, k) \in E \} + \{ \langle \langle i, l \rangle, \langle i, n_{i} \rangle \rangle | i \in V_{i} \}. \]

In words, each server is replaced by a subgraph corresponding to the stochastic PERT graph associated with the service time distribution. Each edge within the subgraph contains a buffer of capacity one and initial marking zero. In addition, a buffer of capacity one is placed between the source and sink nodes of each subgraph with a buffer of capacity one and an initial marking zero to restrict service to at most one job at a time at each server. An edge containing a buffer of capacity $B_{i,k}$ and initial marking $M_{i,k}$ connects the sink node of the $i$th subgraph with the source node of the $k$th subgraph if such a buffer existed between the $i$th and $k$th servers in $\mathcal{Y}$. This guarantees that server $\langle k, 1 \rangle$ is starved in $\mathcal{F}'$ iff server $k$ is starved in $\mathcal{Y}$. Last, a buffer of capacity $B_{i,k}$ and initial marking $M_{i,k}$ is placed between the source node of subgraph $i$ and the sink node of subgraph $k$ to ensure that server $\langle i, 1 \rangle$ is blocked in $\mathcal{F}'$ iff server $i$ is blocked in $\mathcal{Y}$.

By coupling the service time $\sigma_{i,n}$ with the $n$th realization of a r.v. with distribution $F(G_{i}, \lambda_{1,i}, \ldots, \lambda_{n_{i},i})$, one can readily show by induction that

\[ m_{i,k}(t) = m_{\langle i, n_{i}, \langle k, 1 \rangle \rangle}(t), \quad t \geq 0, \quad (i, k) \in E. \]

An application of Lemma 8.1 completes the proof. □
ACKNOWLEDGMENTS. The authors are grateful to F. Bonhoure for useful discussions at the beginning of this work, and to the anonymous referees for their careful and useful comments.

REFERENCES


