Journal of the Korean Statistical Society (2006), **35***: 3, pp 281–303*

MARKOVIAN EARLY ARRIVAL DISCRETE TIME JACKSON NETWORKS

A. ABOUL-HASSAN[1] AND S. I. RABIA[2]

ABSTRACT

In an earlier work, we investigated the problem of using linear programming to bound performance measures in a discrete time Jackson network. There it was assumed that the system evolution is controlled by the early arrival scheme. This assumption implies that the system can't be modelled by a Markov chain. This problem was resolved and performance bounds were calculated. In the present work, we use a modification of the early arrival scheme (without corrupting it) in order to make the system evolves as a Markov chain. This modification enables us to obtain explicit expressions for certain moments that could not be calculated explicitly in the pure early arrival scheme setting. Moreover, this feature implies a reduction in the linear program size as well as the computation time. In addition, we obtained tighter bounds than those appeared before due to the new setting.

AMS 2000 subject classifications. Primary 60K25; Secondary 68M20.
Keywords. Discrete time queueing systems, early arrival scheme, performance bounds.

1. INTRODUCTION

Discrete time queueing systems received great interest in recent years because they are commonly used in the design and analysis of many communication and computer systems where the time is slotted. These systems include slotted ALOHA, slotted Carrier-Sense Multiple-Access (CSMA) and Asynchronous Transfer Mode (ATM) networks (see Atencia and Moreno, 2004; Gelenbe and Pujolle, 1997; Li and Yang, 1998; Woodward, 1998). The present work is concerned

---

Received March 2005; accepted August 2006.
[1]Department of Engineering Mathematics and Physics, Alexandria University, Alexandria 21544, Egypt (e-mail: osstd@soficom.com.eg)
[2]Corresponding author. Department of Engineering Mathematics and Physics, Alexandria University, Alexandria 21544, Egypt (e-mail: shrfrabia@hotmail.com)

with the analysis of a class of discrete time queueing networks namely discrete time Jackson network with batch arrivals and single departure.

The simplest discrete time queueing network is the tandem queue. This model working with Bernoulli arrivals (at the first node) and geometric service times at all nodes was studied by Hsu and Burke (1976). It was shown that each node can be treated independently as a Bernoulli arrivals-geometric service queue. This result is very similar to that of the continuous time case (see Burke, 1956). A discrete time Jackson network with Bernoulli arrivals and geometric service times at all nodes was studied by Bharath-Kumar (1980). The generating function of the number of customers at various nodes was derived. However, it was shown that the independence property of the continuous time Jackson network (see Jackson, 1957) does not hold in the discrete setting. This is the starting point of the present work as we will explain below. A general discrete time queueing network model that allows for both arrivals and departures that are state dependent was introduced by Henderson and Taylor (1990). It was proved that the model possesses a product form solution.

As mentioned by Bharath-Kumar (1980), the discrete time Jackson network does not have the decomposition property of the continuous time case. Therefore, obtaining exact expressions for performance measures seems to be a hard problem. In this paper, we treat the problem differently. Instead of obtaining exact expressions for the performance measures, we calculate upper and lower bounds on these measures by solving a linear program. The origin of this idea appeared in the work of Kumar and Kumar (1994) to analyze a class of continuous time queueing networks. In a previous work (see Aboul-Hassan and Rabia, 2002, 2003), we extended the application of this idea to include the discrete time batch arrivals-geometric service Jackson network working under the early arrival scheme (see Gelenbe and Pujolle, 1997). The system evolution of the model described in Aboul-Hassan and Rabia (2002, 2003) does not follow a Markov chain. However, this problem was resolved and performance bounds were calculated. In the present work, we use a modification of the early arrival scheme (without corrupting it) in order to make the system evolves as a Markov chain. This modification enables us to obtain explicit expressions for certain moments that could not be calculated explicitly in Aboul-Hassan and Rabia (2002, 2003). Moreover, this feature implies a reduction in the linear program size as well as the computation time. In addition, we obtained tighter bounds than those appeared in Aboul-Hassan and Rabia (2002, 2003) due to the new setting.

This paper is organized as follows. In Section 2, we describe the mathematical model of the network and introduce the notations that will be used throughout this work. In Section 3, the model is analyzed by constructing a linear program whose solution provides bounds on the performance measures. Numerical examples are presented in Section 4. Conclusions and some open problems are given in Section 5.

2. MATHEMATICAL MODEL

We are interested here in analyzing a discrete time Jackson network. In the discrete time setting (see for more details Gelenbe and Pujolle, 1997), the time axis is divided into intervals of equal lengths called time slots. The boundaries of these slots are called time points. System events, *i.e.*, arrivals and departures, occur at these time points. More precisely, system events occur only *just after* or *just before* a time point. In order to be able to compute the system state at any time point, one must determine the order at which arrivals and departures occur. Hence, system evolution is assumed to be controlled by one of two basic schemes: late arrival scheme or early arrival scheme. In the late arrival scheme (see Figure 2.1), arrivals (external or internal) are assumed to occur at the end of

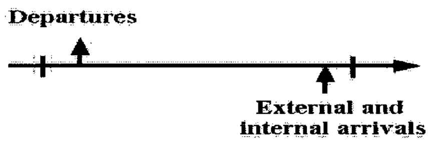

FIGURE 2.1 *Late arrival scheme.*

a time slot where departures occur at the beginning of the time slot. According to this scheme, an arriving customer to an idle server can't depart during the same time slot. He must wait until the next time slot (at least) in order to complete his service. The other possibility is to control the system with the early arrival scheme (see Figure 2.2). To do so, arrivals are assumed to occur at the beginning of a time slot where departures take place at the end of the time slot. In contrast to the late arrival scheme, an arriving customer to an idle server will get service in the same time slot and may depart before the beginning of the next time slot. In the present work, it is assumed that the system evolution is controlled by

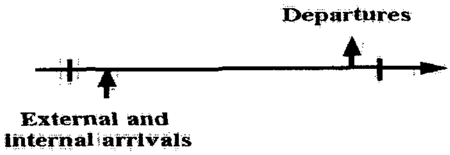

FIGURE 2.2 *Pure early arrival scheme.*

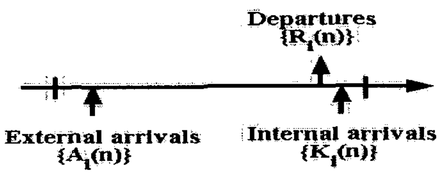

FIGURE 2.3 *Markovian early arrival scheme.*

the early arrival scheme. However, in the pure early arrival scheme as the one shown in Figure 2.2, the system state does not evolve as a Markov chain (see Aboul-Hassan and Rabia, 2003, for more details). To overcome this problem, we modify the scheme slightly by assuming that departures from a node arrive at their destination before the beginning of the next time slot. In other words (see Figure 2.3), we assume that external arrivals occur at the beginning of the time slot whereas internal arrivals occur at the end of the time slot. This modification implies that the system state evolves as a Markov chain. This variation of the early arrival scheme was used in Gelenbe and Pujolle (1997).

The network consists of N nodes. Each node has an infinite queue and a single server. At the beginning of the time slot n, $n = 1, 2, \ldots$, external arrivals at node i, $i = 1, 2, \ldots, N$ occur in batches. The batch size is denoted by $A_i(n)$. We assume that $\{A_i(n), n = 1, 2, \ldots\}$ constitutes a time stationary stochastic process as defined in Nelson (1995). Moreover, we assume that arrivals at different nodes are independent. Throughout this paper, we will use the notation $\overline{Y}$ to denote the expectation of the random variable Y. Hence, $\overline{A_i(n)} = E\{A_i(n)\}$, $\overline{A_i{}^2(n)} = E\{A_i{}^2(n)\}$ and so on. Since the process $\{A_i(n)\}$ is time stationary, we will write $\overline{A_i}$ to denote the expected number of arrivals at node i during any time slot.

By the end of the time slot n, a single customer departs from node i with probability σ_i provided that node i is not empty. It is assumed that the service probability σ_i is independent of the number of time slots that the customer has stayed in this node. This assumption implies that the service time follows a geometric distribution. The departure process at node i is denoted by $\{R_i(n)\}$ where $R_i(n) \in \{0, 1\}$ is the number of departures from node i during the n^{th} time slot. We will use R_i to represent the steady state of the process $\{R_i(n)\}$, *i.e.*, $R_i = \lim_{n\to\infty} R_i(n)$ assuming that the system is stable.

After the customer completes his service at node i, he moves to node j with probability P_{ij} and leaves the network with probability $1 - \sum_{j=1}^{N} P_{ij}$. We assume for simplicity that $P_{ii} = 0$. The internal arrivals process at node i is denoted by $\{K_i(n)\}$ where $K_i(n) \in \{0, 1, \ldots, N-1\}$ is the number of internal arrivals at node i during the n^{th} time slot (see Figure 2.3). Moreover, $K_i = \lim_{n\to\infty} K_i(n)$ represents the steady state of the process $\{K_i(n)\}$.

The system state is given by $X(n) = (X_1(n), X_2(n), \ldots, X_N(n))$ where $X_i(n)$ is the number of customers at node i at the beginning of time slot n, $n = 1, 2, \ldots$ and $X(1)$ is the initial state. Following our notation the vector $X = (X_1, X_2, \ldots, X_N)$ represents the steady state of the network.

Finally, we assume that the policy applied at each node is non-idling, *i.e.*, the server at node i will be idle during the time slot n if and only if $X_i(n) = A_i(n) = 0$.

3. APPROXIMATE ANALYSIS

Our analysis is based on building up a linear program whose solution (for both minimization and maximization) gives upper and lower bounds on the performance measures. We are interested mainly in the expected number of customers in the network. Hence, this will represent the objective function of the linear program. The constraints (see Lemmas 3.1, 3.2) are obtained mainly by assuming that the system reaches a steady state and examining the implication of this assumption on the following moments:

$$\begin{aligned} &E\{X_i{}^2(n)\}, \quad i = 1, 2, \ldots, N, \\ &E\{X_i(n)X_j(n)\}, \quad i, j = 1, 2, \ldots, N, \ i \neq j. \end{aligned}$$

Another set of constrains (see Lemma 3.3) is added due to the non-idling nature of the applied policy. Moreover, a class of the constraints variables is shown to be bounded. These bounds are given in Lemma 3.4. The complete linear program is presented in Theorem 3.1. Before presenting our results, we need the following definitions:

DEFINITION 3.1. *The utilization stochastic processes* $\{U_i(n), n = 1, 2, \ldots\}$, $i = 1, 2, \ldots, N$ *are defined as follows:*

$$U_i(n) = 0, \text{ if node } i \text{ is idle during the } n^{th} \text{ time slot,}$$
$$= 1, \text{ otherwise.}$$

Following our notations given in Section 2, $U_i = \lim_{n\to\infty} U_i(n)$ and $\overline{U_i}$ represents the steady state utilization at node i which can be computed using the usual traffic equations

$$\sigma_i \overline{U_i} = \overline{A_i} + \sum_{j=1,\ j\neq i}^{N} \sigma_j \overline{U_j} P_{ji}. \tag{3.1}$$

DEFINITION 3.2. *Assuming that the system is stable, we define the following variables:*

$$z_{ij} = \lim_{n\to\infty} E\{U_i(n)X_j(n)\}, \quad i, j = 1, 2, \ldots, N,$$
$$v_{ij} = \lim_{n\to\infty} E\{U_i(n)U_j(n)\}, \quad i, j = 1, 2, \ldots, N.$$

Note that the non-idling assumption implies that $U_i(n)X_i(n) = X_i(n)$ for all i. Moreover, since $U_i(n) \in \{0, 1\}$, $v_{ii} = \overline{U_i}$.

The following lemma is a consequence of assuming that the system has reached a steady state and examining the implication of this assumption on the moments $E\{X_i^2(n)\}$, $i = 1, 2, \ldots, N$. In other words, we consider the consequence of the equalities $E\{X_i^2(n+1)\} = E\{X_i^2(n)\}$.

LEMMA 3.1. *For* $i = 1, 2, \ldots, N$,

$$2(\overline{A_i} - \sigma_i)z_{ii} + 2\sum_{j=1,j\neq i}^{N} \sigma_j P_{ji} z_{ji} + \sum_{j=1,j\neq i}^{N} \sum_{q=1,q\neq i,q\neq j}^{N} v_{jq}\sigma_j\sigma_q P_{ji}P_{qi} + \overline{A_i^2}$$
$$+ \overline{U_i}\sigma_i - 2\overline{A_i}\sigma_i + (1 + 2\overline{A_i})\sum_{j=1,j\neq i}^{N} \overline{U_j}\sigma_j P_{ji} - 2\sigma_i \sum_{j=1,j\neq i}^{N} v_{ij}\sigma_j P_{ji} = 0. \tag{3.2}$$

PROOF.

$$E\{X_i^2(n+1)|X(n)\} = \sum_{a=0}^{\infty}\sum_{k=0}^{N-1}\sum_{r=0}^{1}(X_i(n)+a+k-r)^2 P(A_i(n)=a, K_i(n)=k, R_i(n)=r|X(n)).$$

After some simplifications, one arrives at the following result:

$$E\{X_i^2(n+1)|X(n)\} = X_i^2 + 2X_i(n)(\overline{A_i} + \overline{K_i(n)} - \overline{R_i(n)}) + \overline{A_i^2} + \overline{K_i^2(n)} + \overline{R_i^2(n)} + 2\overline{A_iK_i(n)} - 2\overline{A_iR_i(n)} - 2\overline{K_i(n)R_i(n)}. \quad (3.3)$$

It should be mentioned here that the expectations given in the right hand side of the above equation are in fact conditional expectations based on knowing the system state at time n: $X(n)$. To avoid complications in notations, we didn't introduce a sperate notation for conditional expectations. The expectations $\overline{A_i(n)}$ and $\overline{A_i^2(n)}$ are system parameters and it is easy to show that

$$\overline{R_i(n)} = \overline{R_i^2(n)} = U_i(n)\sigma_i. \quad (3.4)$$

The other expectations that appear in Equation 3.3 need further computation. To compute $\overline{K_i(n)}$, we define $K_{ij}(n) \in \{0,1\}$ to be the number of customers that depart from node j during the n^{th} time slot and arrive at node i before the beginning of the $(n+1)^{th}$ time slot. It can be easily proved that $\overline{K_{ij}(n)} = \overline{R_j(n)}P_{ji}$. Using Equation 3.4, then $\overline{K_{ij}(n)} = U_j(n)\sigma_j P_{ji}$. Since $K_i(n) = \sum_{j=1,j\neq i}^{N} K_{ij}(n)$,

$$\overline{K_i(n)} = \sum_{j=1,j\neq i}^{N} U_j(n)\sigma_j P_{ji}. \quad (3.5)$$

To compute $\overline{A_iK_i(n)}$, one notes that arrivals to the network during any time slot are independent of the network state. Moreover, the number of internal arrivals $K_i(n)$ at node i during the n^{th} time slot does not depend on the number of arrivals A_i to the same node during the same time slot due to the assumption that $P_{ii} = 0$. Therefore, A_i and $K_i(n)$ are independent. Hence,

$$\overline{A_iK_i(n)} = \overline{A_i}\,\overline{K_i(n)} = \overline{A_i}\sum_{j=1,j\neq i}^{N} U_j(n)\sigma_j P_{ji}. \quad (3.6)$$

To compute $\overline{K_i^2(n)}$, we apply Equation 3.5 and use the independence among $K_{ij}(n)$s for the same i to obtain

$$\overline{K_i^2(n)} = \sum_{j=1,j\neq i}^{N} Var\{K_{ij}(n)\} + (\sum_{j=1,j\neq i}^{N} U_j(n)\sigma_j P_{ji})^2.$$

Because $K_{ij}(n) \in \{0,1\}$, one can check that

$$Var\{K_{ij}(n)\} = U_j(n)\sigma_j P_{ji} - (U_j(n)\sigma_j P_{ji})^2.$$

Hence,

$$\overline{K_i{}^2(n)} = \sum_{j=1, j\neq i}^{N} U_j(n)\sigma_j P_{ji} + \sum_{j=1, j\neq i}^{N} \sum_{q=1, q\neq i, q\neq j}^{N} (U_j(n)\sigma_j P_{ji})(U_q(n)\sigma_q P_{qi}). \quad (3.7)$$

To compute $\overline{A_i R_i(n)}$, we note that the single service assumption implies that

$$\begin{aligned}\overline{A_i R_i(n)} &= \sum_{a_i=0}^{\infty} \sum_{r_i=0}^{1} a_i r_i P(A_i = a_i,\ R_i(n) = r_i) \\ &= \sum_{a_i=1}^{\infty} a_i P(A_i = a_i) P(R_i(n) = 1 | A_i = a_i). \end{aligned} \quad (3.8)$$

The early arrival assumption implies that

$$P(R_i(n) = 1 | A_i(n) = a_i > 0) = \sigma_i. \quad (3.9)$$

Substituting from Equation 3.9 into Equation 3.8,

$$\overline{A_i(n) R_i(n)} = \overline{A_i}\sigma_i. \quad (3.10)$$

To derive an expression for $\overline{K_i(n) R_i(n)}$, one notes that for the present setting the two random variables $K_i(n)$ and $R_i(n)$ are independent for all n. The independence follows essentially from the assumption that internal arrivals occur before the beginning of the next time slot. This implies that departures from node i during a time slot are not affected by internal arrivals at the same node during the same time slot. Moreover, internal arrivals at node i during a time slot are not affected by departures from the same node during the same time slot because $P_{ii} = 0$. Hence,

$$\overline{K_i(n) R_i(n)} = \overline{K_i(n)}\ \overline{R_i(n)}.$$

Substituting from Equations 3.4 and 3.5,

$$\overline{K_i(n) R_i(n)} = U_i(n)\sigma_i \sum_{j=1, j\neq i}^{N} U_j(n)\sigma_j P_{ji}. \quad (3.11)$$

Now, substituting from Equations 3.4, 3.5, 3.6, 3.7, 3.10 and 3.11 into Equation 3.3, taking the expectation of both sides and applying the steady state assumption,

$$2(\overline{A_i}\,\overline{X_i} + \sum_{j=1,j\neq i}^{N} \overline{X_i U_j}\sigma_j P_{ji} - \sigma_i \overline{X_i}) + \overline{A_i{}^2} + \sum_{j=1,j\neq i}^{N} \overline{U_j}\sigma_j P_{ji}$$
$$+ \sum_{j=1,j\neq i}^{N} \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_j U_q}\sigma_j\sigma_q P_{ji}P_{qi} + \overline{U_i}\sigma_i + 2\overline{A_i} \sum_{j=1,j\neq i}^{N} \overline{U_j}\sigma_j P_{ji}$$
$$- 2\overline{A_i}\sigma_i - 2\sigma_i \sum_{j=1,j\neq i}^{N} \overline{U_i U_j}\sigma_j P_{ji} = 0.$$

Applying Definition 3.2 to the above equation gives Equation 3.2 which completes the proof. □

The next lemma gives the consequence of the steady state assumption on the moments $E\{X_i(n)X_j(n)\},\ i,j = 1,2,\ldots,N,\ i \neq j$.

LEMMA 3.2. *For* $i = 1,2,\ldots,N-1,\quad j = i+1, i+2,\ldots,N,$

$$\overline{A_j}z_{ii} + \sum_{q=1,q\neq j}^{N} \sigma_q P_{qj} z_{qi} - \sigma_j z_{ji} + \overline{A_i}z_{jj} + \sum_{q=1,q\neq i}^{N} \sigma_q P_{qi} z_{qj} - \sigma_i z_{ij} + \overline{A_i}\,\overline{A_j}$$
$$+ \overline{A_i} \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_q}\sigma_q P_{qj} + \overline{A_i}\sigma_i P_{ij} - \overline{A_i}\,\overline{U_j}\sigma_j + \overline{A_j} \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_q}\sigma_q P_{qi} + \overline{A_j}\sigma_j P_{ji}$$
$$+ \sum_{q_1=1,q_1\neq i}^{N} \sum_{q_2=1,q_2\neq j,q_1\neq q_1}^{N} v_{q_1q_2}\sigma_{q_1}\sigma_{q_2} P_{q_1 i} P_{q_2 j} - \sigma_j \sum_{q=1,q\neq i,q\neq j}^{N} v_{jq}\sigma_q P_{qi} - \overline{U_j}\sigma_j P_{ji}$$
$$- \overline{A_j}\,\overline{U_i}\sigma_i - \sigma_i \sum_{q=1,q\neq i,q\neq j}^{N} v_{iq}\sigma_q P_{qj} - \overline{U_i}\sigma_i P_{ij} + v_{ij}\sigma_i\sigma_j = 0. \qquad (3.12)$$

PROOF.

$$E\{X_i(n+1)X_j(n+1)|X(n)\} =$$
$$\sum_{a_i=0}^{\infty}\sum_{k_i=0}^{N-1}\sum_{r_i=0}^{1}\sum_{a_j=0}^{\infty}\sum_{k_j=0}^{N-1}\sum_{r_j=0}^{1}(X_i(n) + a_i + k_i - r_i) \times (X_j(n) + a_j + k_j - r_j) \times$$
$$P(A_i(n) = a_i, K_i(n) = k_i, R_i(n) = r_i, A_j(n) = a_j, K_j(n) = k_j, R_j(n) = r_j|X(n))$$

After some simplification, we obtain

$$\begin{aligned}E\{X_i(n+1)X_j(n+1)|X(n)\} &= X_i(n)X_j(n) + X_i(n)(\overline{A_j} + \overline{K_j(n)} - \overline{R_j(n)})\\ &+ X_j(n)(\overline{A_i} + \overline{K_i(n)} - \overline{R_i(n)}) + \overline{A_iA_j} + \overline{A_iK_j(n)} - \overline{A_iR_j(n)} + \overline{A_jK_i(n)}\\ &+ \overline{K_i(n)K_j(n)} - \overline{K_i(n)R_j(n)} - \overline{A_jR_i(n)} - \overline{R_i(n)K_j(n)} + \overline{R_i(n)R_j(n)}. \quad (3.13)\end{aligned}$$

As mentioned in the proof of Lemma 3.1, the expectations given in the right hand side of the above equation are conditionals ones. The main task now is to compute the expectations in the right hand side of Equation 3.13. Since arrivals at different nodes are independent,

$$\overline{A_iA_j} = \overline{A_i}\,\overline{A_j}. \quad (3.14)$$

Next, we consider $\overline{A_iK_j(n)}$. Note that in Aboul-Hassan and Rabia (2002, 2003), A_i and $K_j(n)$ were independent. In the present setting they are not independent as it will be shown in the following derivation. Recall that $K_j(n) = \sum_{q=1,q\neq j}^{N} K_{jq}(n)$. Hence,

$$\overline{A_iK_j(n)} = \sum_{q=1,q\neq j}^{N} \overline{A_iK_{jq}(n)} = \sum_{q=1,q\neq j,q\neq i}^{N} \overline{A_iK_{jq}(n)} + \overline{A_iK_{ji}(n)}. \quad (3.15)$$

During any time slot n, the number of internal arrivals $K_{jq}(n)$ at node j that are coming from node q are independent of the number of external arrivals A_i at node i for $q \neq i$. Hence,

$$\sum_{q=1,q\neq j,q\neq i}^{N} \overline{A_iK_{jq}(n)} = \sum_{q=1,q\neq j,q\neq i}^{N} \overline{A_i}\,\overline{K_{jq}(n)}.$$

Recall that $\overline{K_{ij}(n)} = U_j(n)\sigma_j P_{ji}$. Hence,

$$\sum_{q=1,q\neq j,q\neq i}^{N} \overline{A_iK_{jq}(n)} = \sum_{q=1,q\neq j,q\neq i}^{N} \overline{A_i}U_q(n)\sigma_q P_{qj}. \quad (3.16)$$

On the other hand, the number of internal arrivals $K_{ji}(n)$ at node j that are coming from node i are clearly dependent on A_i. To compute $\overline{A_iK_{ji}(n)}$, we proceed as follows:

$$\begin{aligned}\overline{A_iK_{ji}(n)} &= \sum_{a_i=0}^{\infty}\sum_{k_{ji}=0}^{1} a_ik_{ji}P(A_i = a_i,\ K_{ji}(n) = k_{ji})\\ &= \sum_{a_i=1}^{\infty} a_iP(A_i = a_i,\ K_{ji}(n) = 1). \quad (3.17)\end{aligned}$$

Note that $\{K_{ji}(n) = 1\}$ implies that $\{R_i(n) = 1\}$ (the reverse implication is not true). Hence,

$$\begin{aligned} &P(A_i = a_i,\ K_{ji}(n) = 1) \\ &= P(A_i = a_i,\ R_i(n) = 1,\ K_{ji}(n) = 1) \\ &= P(A_i = a_i) \times P(R_i(n) = 1|A_i = a_i) \times P(K_{ji}(n) = 1|A_i = a_i,\ R_i(n) = 1). \end{aligned}$$

Applying Equation 3.9, then

$$P(A_i = a_i,\ K_{ji}(n) = 1) = P(A_i = a_i)\sigma_i P_{ij}.$$

Substituting from the above equation into Equation 3.17 yields,

$$\overline{A_i K_{ji}(n)} = \overline{A_i}\sigma_i P_{ij}. \tag{3.18}$$

Substituting from Equations 3.16 and 3.18 into Equation 3.15 gives the required result

$$\overline{A_i K_j(n)} = \overline{A_i} \sum_{q=1, q\neq j, q\neq i}^{N} U_q(n)\sigma_q P_{qj} + \overline{A_i}\sigma_i P_{ij}. \tag{3.19}$$

We consider now $\overline{A_i R_j(n)}$. Number of departures $R_j(n)$ from node j during the n^{th} time slot depends only on the state $X_j(n)$ of the node at the beginning of the time slot, the number of arrivals A_j to this node during the same time slot and the service probability σ_j. This implies that A_i and $R_j(n)$, $i \neq j$, are independent. Using Equation 3.4, then

$$\overline{A_i R_j(n)} = \overline{A_i} U_j(n)\sigma_j. \tag{3.20}$$

To compute $\overline{K_i(n)K_j(n)}$, recall that $K_i(n) = \sum_{q_1=1, q_1\neq i}^{N} K_{iq_1}(n)$ and $K_j(n) = \sum_{q_2=1, q_2\neq j}^{N} K_{jq_2}(n)$. Hence,

$$\overline{K_i(n)K_j(n)} = \sum_{q_1=1, q_1\neq i}^{N} \sum_{q_2=1, q_2\neq j}^{N} \overline{K_{iq_1}(n)K_{jq_2}(n)}. \tag{3.21}$$

Since $K_{iq_1}(n),\ K_{jq_2}(n) \in \{0, 1\}$,

$$\overline{K_{iq_1}(n)K_{jq_2}(n)} = P(K_{iq_1}(n) = 1,\ K_{jq_2}(n) = 1).$$

If $q_1 = q_2$, then the two events $\{K_{iq_1}(n) = 1\}$ and $\{K_{jq_1}(n) = 1\}$ are disjoint because $\{K_{iq_1}(n) = 1\}$ implies that $\{K_{jq_1}(n) = 0\}$. On the other hand, the events $\{K_{iq_1}(n) = 1\}$ and $\{K_{jq_2}(n) = 1\}$ are independent if $q_1 \neq q_2$ because the behavior of departing customers at different nodes are independent. Hence,

$$\begin{aligned} \overline{K_{iq_1}(n)K_{jq_2}(n)} &= 0, & q_1 = q_2 \\ &= P(K_{iq_1}(n) = 1) \times P(K_{jq_2}(n) = 1), & q_1 \neq q_2. \end{aligned} \tag{3.22}$$

From the definition of the stochastic process $\{K_{iq}(n),\ n \geq 1\}$, then

$$P(K_{iq_1}(n) = 1) = P(R_{q_1}(n) = 1) \times P_{q_1 i} = \overline{R_{q_1}(n)} \times P_{q_1 i}.$$

Substituting for $\overline{R_{q_1}(n)}$ from Equation 3.4, then

$$P(K_{iq_1}(n) = 1) = U_{q_1}(n)\sigma_{q_1}P_{q_1 i}. \tag{3.23}$$

Similarly,

$$P(K_{jq_2}(n) = 1) = U_{q_2}(n)\sigma_{q_2}P_{q_2 j}. \tag{3.24}$$

Substituting from Equations 3.23 and 3.24 into Equation 3.22,

$$\begin{aligned} \overline{K_{iq_1}(n)K_{jq_2}(n)} &= 0, & q_1 = q_2 \\ &= U_{q_1}(n)U_{q_2}(n)\sigma_{q_1}\sigma_{q_2}P_{q_1 i}P_{q_2 j}, & q_1 \neq q_2. \end{aligned} \tag{3.25}$$

Substituting from Equation 3.25 into Equation 3.21, we obtain the required expression

$$\overline{K_i(n)K_j(n)} = \sum_{q_1=1, q_1\neq i}^{N} \sum_{q_2=1, q_2\neq j, q_1\neq q_2}^{N} U_{q_1}(n)U_{q_2}(n)\sigma_{q_1}\sigma_{q_2}P_{q_1 i}P_{q_2 j}. \tag{3.26}$$

The remaining expectations are $\overline{R_i(n)R_j(n)}$ and $\overline{R_i(n)K_j(n)}$. In Aboul-Hassan and Rabia (2002, 2003), we were not able to obtain explicit expressions for these expectations. For the present setting, these expectations are calculated as follows. For $\overline{R_i(n)R_j(n)}$, the result is based on the independence between $R_i(n)$ and $R_j(n)$. As said before, the number of departures $R_i(n)$ from node i during time slot n depends only on three factors: $X_i(n)$, A_i and σ_i. These factors are not related to $R_j(n)$. Hence,

$$\overline{R_i(n)R_j(n)} = \overline{R_i(n)}\;\overline{R_j(n)}.$$

Substituting from Equation 3.4 gives the required result

$$\overline{R_i(n)R_j(n)} = U_i(n)U_j(n)\sigma_i\sigma_j. \tag{3.27}$$

The derivation of $\overline{R_i(n)K_j(n)}$ is similar to $\overline{A_iK_j(n)}$. Recall that

$$K_j(n) = \sum_{q=1,q\neq j}^{N} K_{jq}(n).$$

Hence,

$$\overline{R_i(n)K_j(n)} = \sum_{q=1,q\neq j}^{N} \overline{R_i(n)K_{jq}(n)} = \sum_{q=1,q\neq j,q\neq i}^{N} \overline{R_i(n)K_{jq}(n)} + \overline{R_i(n)K_{ji}(n)}. \tag{3.28}$$

The number of internal arrivals $K_{jq}(n)$ at node j that are coming from node q are independent of the number of departures $R_i(n)$ from node i for $q \neq i$. Hence,

$$\sum_{q=1,q\neq j,q\neq i}^{N} \overline{R_i(n)K_{jq}(n)} = \sum_{q=1,q\neq j,q\neq i}^{N} \overline{R_i(n)}\,\overline{K_{jq}(n)}.$$

Recall that $\overline{K_{ij}(n)} = U_j(n)\sigma_j P_{ji}$ and $\overline{R_i(n)} = U_i(n)\sigma_i$. Hence,

$$\sum_{q=1,q\neq j,q\neq i}^{N} \overline{R_i(n)K_{jq}(n)} = U_i(n)\sigma_i \sum_{q=1,q\neq j,q\neq i}^{N} U_q(n)\sigma_q P_{qj}. \tag{3.29}$$

On the other hand, the number of internal arrivals $K_{ji}(n)$ at node j that are coming from node i are clearly dependent on $R_i(n)$. To compute $\overline{R_i(n)K_{ji}(n)}$, we note that since a single service is assumed,

$$\overline{R_i(n)K_{ji}(n)} = P(R_i(n) = 1,\ K_{ji}(n) = 1). \tag{3.30}$$

However,

$$\begin{aligned} P(R_i(n) = 1,\ K_{ji}(n) = 1) &= P(R_i(n) = 1)P(K_{ji}(n) = 1|R_i(n) = 1) \\ &= \overline{R_i(n)}P_{ij}. \end{aligned}$$

Substituting for $\overline{R_i(n)}$ from Equation 3.4, then

$$P(R_i(n) = 1,\ K_{ji}(n) = 1) = U_i(n)\sigma_i P_{ij}.$$

Now, substituting from the above equation into Equation 3.30 yields,

$$\overline{R_i(n)K_{ji}(n)} = U_i(n)\sigma_i P_{ij}. \tag{3.31}$$

Substituting from Equations 3.29 and 3.31 into Equation 3.28 gives the required result

$$\overline{R_i(n)K_j(n)} = U_i(n)\sigma_i \sum_{q=1,q\neq j,q\neq i}^{N} U_q(n)\sigma_q P_{qj} + U_i(n)\sigma_i P_{ij}. \tag{3.32}$$

Now, substituting from Equations 3.4, 3.5, 3.14, 3.19, 3.20, 3.26, 3.27 and 3.32 into Equation 3.13, taking the expectation of both sides and applying the steady state assumption,

$$\begin{aligned}
&\overline{A_j}\,\overline{X_i} + \sum_{q=1,q\neq j}^{N} \overline{U_q X_i}\sigma_q P_{qj} - \overline{U_j X_i}\sigma_j + \overline{A_i}\,\overline{X_j} + \sum_{q=1,q\neq i}^{N} \overline{U_q X_j}\sigma_q P_{qi} - \overline{U_i X_j}\sigma_i \\
&+ \overline{A_i}\,\overline{A_j} + \overline{A_i} \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_q}\sigma_q P_{qj} + \overline{A_i}\sigma_i P_{ij} - \overline{A_i}\,\overline{U_j}\sigma_j + \overline{A_j} \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_q}\sigma_q P_{qi} \\
&+ \overline{A_j}\sigma_j P_{ji} + \sum_{q_1=1,q_1\neq i}^{N} \sum_{q_2=1,q_2\neq j,q_1\neq q_2}^{N} \overline{U_{q_1} U_{q_2}}\sigma_{q_1}\sigma_{q_2} P_{q_1 i} P_{q_2 j} \\
&- \sigma_j \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_j U_q}\sigma_q P_{qi} - \overline{U_j}\sigma_j P_{ji} - \overline{A_j U_i}\sigma_i - \sigma_i \sum_{q=1,q\neq i,q\neq j}^{N} \overline{U_i U_q}\sigma_q P_{qj} \\
&- \overline{U_i}\sigma_i P_{ij} + \overline{U_i U_j}\sigma_i\sigma_j = 0.
\end{aligned}$$

Applying Definition 3.2 to the above equation gives Equation 3.12 which completes the proof. □

The following lemma is a consequence of the non-idling nature of the service policy.

LEMMA 3.3. *For* $i, j = 1, 2, \ldots, N$, $i \neq j$, *we have*

$$z_{ji} \leq z_{ii}. \tag{3.33}$$

PROOF. Recall from Definition 3.2 that $z_{ij} = \lim_{n\to\infty} E\{U_i(n)X_j(n)\}$. The non-idling nature of the policy applied at each node implies that $X_i(n) = 0$ if $U_i(n) = 0$. However, for $i \neq j$, $U_j(n)$ may be zero when $X_i(n) > 0$. Hence, $U_j(n)X_i(n) \leq U_i(n)X_i(n)$ for all $n \geq 1$. Taking the expectation of both sides and assuming a steady state exists, give Equation 3.33. □

The following lemma shows that the variables v_{ij} are bounded (below and above).

LEMMA 3.4. *For* $i = 1, 2, \ldots, N-1,\ j = i+1,\ i+2, \ldots, N,$

$$\left(\overline{U_i} + \overline{U_j} - 1\right)^+ \leq v_{ij} \leq \min(\overline{U_i}, \overline{U_j}). \tag{3.34}$$

PROOF. Recall from Definition 3.2 that $v_{ij} = \lim_{n\to\infty} E\{U_i(n)U_j(n)\}$. Since $U_i(n), U_j(n) \in \{0,1\}$,

$$E\{U_i(n)U_j(n)\} = P(U_i(n) = 1,\ U_j(n) = 1). \tag{3.35}$$

However,

$$\begin{aligned}\{P(U_i(n) = 1) + P(U_j(n) = 1) - 1\}^+ &\leq P(U_i(n) = 1, U_j(n) = 1)\\ &\leq \min(P(U_i(n) = 1), P(U_j(n) = 1)).\end{aligned} \tag{3.36}$$

Since $U_i(n), U_j(n) \in \{0,1\}$,

$$\overline{U_i(n)} = P(U_i(n) = 1), \quad \overline{U_j(n)} = P(U_j(n) = 1). \tag{3.37}$$

Substituting from Equation 3.37 into Equation 3.36,

$$\left(\overline{U_i(n)} + \overline{U_j(n)} - 1\right)^+ \leq P\big(U_i(n) = 1,\ U_j(n) = 1\big) \leq \min\left(\overline{U_i(n)}, \overline{U_j(n)}\right). \tag{3.38}$$

Combining Equations 3.38 and 3.35 and taking the limit as n tends to infinity give Equation 3.34. □

The previous lemmas are the building blocks of the following theorem which is the main result of this work. It enables us to calculate a lower bound on the expected number of customers in the system by solving a linear program.

THEOREM 3.1. *In the steady state, the expected number of customers in the above described discrete time Jackson network is bounded below by the solution of the following linear program*

$$\min \sum_{i=1}^{N} z_{ii}$$

subject to the constraints given in Lemmas 3.1, 3.2, 3.3 and 3.4 by the Equations 3.2, 3.12, 3.33 and 3.34 in addition to:

$$v_{ij} = v_{ji},\ i = 1, 2, \ldots, N-1,\ j = i+1, i+2, \ldots, N, \tag{3.39}$$

$$v_{ii} = \overline{U_i},\ i = 1, 2, \ldots, N, \tag{3.40}$$

$$z_{ij} \geq 0, \quad i, j = 1, 2, \ldots, N. \tag{3.41}$$

PROOF. Recall from Definition 3.2 that $z_{ij} = \lim_{n\to\infty} E\{U_i(n)X_j(n)\}$. Since we assume a non-idling policy, $E\{U_i(n)X_i(n)\} = E\{X_i(n)\}$. Hence, $\sum_{i=1}^{N} z_{ii} = \lim_{n\to\infty} \sum_{i=1}^{N} E\{X_i(n)\}$ which proves that the objective function represents the expected number of customer in the network. The constraints in Equations 3.39 and 3.40 follow obviously from the definition of the variables v_{ij}. Finally, Equation 3.41 represents the non-negativity constraints. □

REMARK 3.1. To obtain an upper bound on the expected number of customers in the network, the minimization problem in Theorem 3.1 is replaced by a maximization one. Moreover, bounds on the expected delay time can be obtained using Little's theorem.

REMARK 3.2. The number of constraints in the linear program of Theorem 3.1 equals $4N^2-N$. The corresponding linear program presented in Aboul-Hassan and Rabia (2002, 2003) had $6N^2 - 2N$ constraints. Hence, the present setting leads to a reduction of at least 30% of the linear program size. This implies a corresponding reduction in the computation time especially for large values of N.

4. NUMERICAL RESULTS

In this section, we examine numerically the result given in Theorem 3.1. The computation process is fully automated. A Mathematica (see Wolfram, 1996) program is written to handle this task. Only the network specifications are given and the program generates the corresponding linear program and solves it for both minimization and maximization. Several network configurations are treated:

- For the arrival process, we examine Bernoulli and Poisson arrivals with equal and non-equal arrival rates. Bernoulli (res. Poisson) arrivals mean that $A_i(n)$ follows a Bernoulli (res. Poisson) distribution with a certain parameter p (res. λ) for all n. The parameters p and λ are adjusted automatically form inside the program to generate the required load factor. In the equal arrival rate case, we assume that $\overline{A_i}$ are the same for all i. In the non-equal arrival rates case, we assume that $\overline{A_i} = i\xi$ where the value of the parameter ξ will be chosen by the program to produce the required load factor.

- Service probabilities σ_i are assumed to be equal.

- For the internal routing, both uniform and non-uniform routings are considered. By uniform routing, we mean that the departing customer from node i has equal chances of joining any other node or departing from the network, *i.e.*, $P_{ij} = 1/N,\ i,j = 1,2,\ldots,N,\ i \neq j$. For the non-uniform traffic case, we consider what we call neighborhood routing. In this type of routing, we assume that the departing customer from node i either joins node $i+1$ or departs from the network. Departing customer from node N either joins node 1 or departs from the network. In other words, the routing matrix $[P_{ij}]$ takes the form:

$$\begin{pmatrix} 0 & P_{12} & 0 & \ldots & 0 \\ 0 & 0 & P_{23} & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ P_{N1} & 0 & 0 & \ldots & 0 \end{pmatrix}$$

We assume that $P_{12} = P_{23} = \cdots = P_{N1}$. For this choice, the two types of routing are identical for $N = 2$.

- The results are calculated for different number of nodes and different load factors.

The results based on Theorem 3.1 are compared with simulation results. A sample path of 30,000 time slots is generated for each network configuration. The first 1,000 time slots are deleted to remove the effect of the transient period. From a computational time point of view, it is noted that calculating performance bounds using our approach is at least 10 times faster than generating a simulation estimate for all the cases we considered. The detailed results for Bernoulli arrivals case are shown in Tables 4.1–4.4. The obtained bounds bracket the simulation results with different degrees of tightness. Moreover, in some heavy traffic cases, the simulation results underestimate or overestimate the required value which is bracketed by our bounds. In such cases, a larger sample path is needed to obtain better simulation estimates.

To examine the tightness of the bounds, we plot bounds difference for different network configurations in Figures 4.1 and 4.2. Uniform routing results are shown in Figure 4.1 whereas neighborhood routing results are shown in Figure 4.2. The first column of both figures represents the results of Bernoulli arrivals and the second column represents those of Poisson arrivals. The first row of both figures represents the results for the equal arrival rates case whereas the second row gives those of the non-equal arrival rates case. In each graph of Figures 4.1 and 4.2, we plot the difference between the upper and lower bounds on the expected number of customers in the network against load factor for different number of nodes.

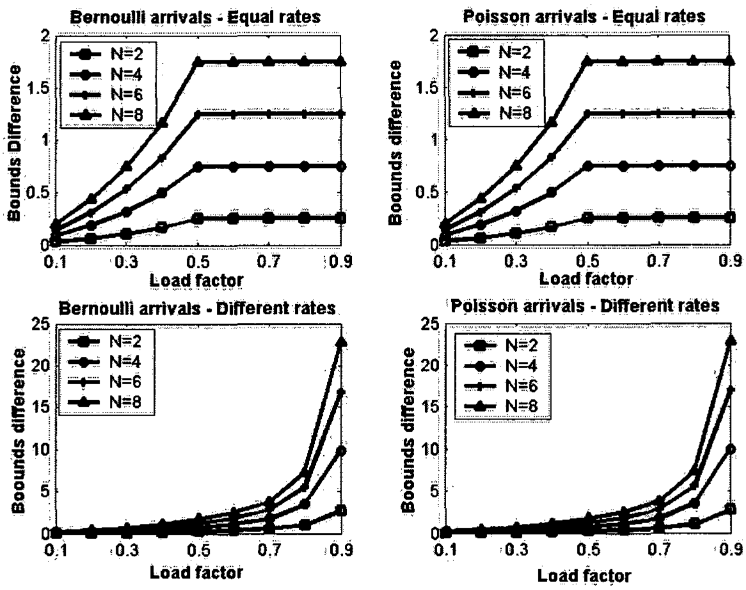

FIGURE 4.1 *Performance bounds for the uniform traffic case.*

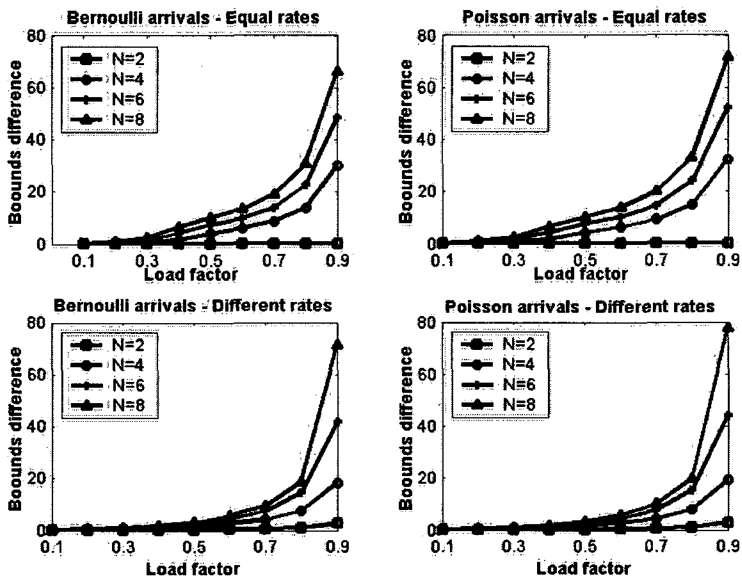

FIGURE 4.2 *Performance bounds for the neighborhood traffic case.*

TABLE 4.1 *Uniform traffic with equal arrival rates.*

	N=2			*N=4*		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.140	0.168	0.159	0.307	0.391	0.360
0.2	0.317	0.379	0.344	0.695	0.882	0.793
0.3	0.546	0.654	0.582	1.200	1.520	1.424
0.4	0.856	1.020	0.876	1.870	2.370	2.045
0.5	1.290	1.540	1.288	2.820	3.580	3.192
0.6	1.950	2.200	2.051	4.260	5.010	4.680
0.7	3.050	3.300	3.182	6.660	7.410	6.857
0.8	5.270	5.520	5.980	11.500	12.200	10.860
0.9	11.900	12.200	10.890	26.000	26.700	28.070
	N=6			*N=8*		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.474	0.613	0.536	0.641	0.835	0.757
0.2	1.070	1.380	1.298	1.450	1.880	1.769
0.3	1.840	2.380	2.170	2.490	3.240	2.872
0.4	2.880	3.710	3.011	3.890	5.050	4.477
0.5	4.340	5.590	5.007	5.850	7.600	6.101
0.6	6.540	7.790	7.457	8.800	10.600	9.171
0.7	10.200	11.500	10.590	13.700	15.500	14.990
0.8	17.600	18.800	17.940	23.600	25.400	23.370
0.9	39.700	40.900	36.410	53.300	55.100	53.740

From the results shown in Figures 4.1 and 4.2, we have the following remarks. The obtained bounds are tight in the light loading case. Increasing the load factor decreases this tightness especially when the number of nodes in the network is also increased. However, for uniform routing with equal arrival rates, the bounds difference approaches a certain limit as a function of the load factor. In all cases, the behavior of the Bernoulli and Poisson arrivals are nearly the same. To bring the obtained bounds tighter, we need to add another set of constraints. We suggest examining the implication of a steady state assumption on higher moments such as $E\{X_i^3(n)\}$, $i = 1, 2, \ldots, N$. However, this seems to lead to a non-linear programming problem. This point needs more investigation.

TABLE 4.2 *Uniform traffic with different arrival rates.*

	N=2			N=4		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.127	0.149	0.142	0.275	0.345	0.317
0.2	0.282	0.333	0.301	0.612	0.769	0.718
0.3	0.477	0.567	0.490	1.030	1.310	1.183
0.4	0.725	0.875	0.771	1.560	2.010	1.730
0.5	1.050	1.300	1.140	2.250	2.990	2.487
0.6	1.500	1.880	1.572	3.170	4.300	3.420
0.7	2.220	2.780	2.407	4.490	6.240	5.249
0.8	3.540	4.560	3.903	6.620	10.100	7.725
0.9	7.190	9.880	6.517	11.700	21.500	14.160
	N=6			N=8		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.431	0.551	0.523	0.591	0.763	0.755
0.2	0.959	1.230	1.163	1.320	1.710	1.629
0.3	1.620	2.090	1.908	2.220	2.900	2.523
0.4	2.450	3.230	2.914	3.380	4.480	4.070
0.5	3.550	4.800	4.088	4.910	6.670	5.333
0.6	5.030	6.860	5.839	7.000	9.490	8.081
0.7	7.160	9.930	8.605	10.000	13.700	11.130
0.8	10.500	16.000	11.930	14.900	22.100	18.110
0.9	17.400	34.200	22.110	24.600	47.300	31.560

5. CONCLUSIONS

This work was devoted to the analysis of discrete time Jackson networks. Instead of obtaining exact expressions for performance measures, we calculated upper and lower bounds on these measures. The bounds were obtained by constructing a linear program whose objective function is the required performance measure (mainly the expected number of customers in the network) and solving it for both minimization and maximization. The system was assumed to be controlled by a modified early arrival scheme in order to make the system evolves as a Markov chain. This modification enabled us to obtain explicit expressions for certain moments that could not be calculated explicitly in Aboul-Hassan and Rabia (2002, 2003). Moreover, this feature implied a reduction in the linear program size as well as the computation time.

The process of building up and solving the linear program was fully automated using Mathematica. From the presented numerical examples, it appeared that the obtained bounds are tight in the case of light loads. Increasing the load factor

TABLE 4.3 *Neighborhood traffic with equal arrival rates.*

	N=2			*N=4*		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.140	0.168	0.159	0.216	0.352	0.322
0.2	0.317	0.379	0.344	0.467	0.852	0.710
0.3	0.546	0.654	0.582	0.779	1.620	1.182
0.4	0.856	1.020	0.876	1.210	2.970	1.784
0.5	1.290	1.540	1.288	1.810	5.920	2.563
0.6	1.950	2.200	2.051	2.700	8.900	4.051
0.7	3.050	3.300	3.182	4.190	13.000	5.856
0.8	5.270	5.520	5.980	7.160	21.200	10.180
0.9	11.900	12.200	10.890	16.000	46.100	22.450
	N=6			*N=8*		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.324	0.542	0.470	0.432	0.732	0.663
0.2	0.700	1.370	1.028	0.933	1.890	1.409
0.3	1.140	2.780	1.756	1.520	4.000	2.353
0.4	1.650	5.830	2.742	2.200	8.530	3.623
0.5	2.370	9.750	3.987	3.000	13.000	5.381
0.6	3.540	13.400	6.031	4.440	17.800	7.954
0.7	5.460	19.400	8.904	6.840	25.900	12.540
0.8	9.290	31.800	15.910	11.600	42.400	19.920
0.9	20.700	69.200	31.850	25.900	92.200	42.970

decreases this tightness especially when the number of nodes in the network is also increased. Moreover, if the arrival rates at different nodes are equal and customers are uniformly routed through the network, the bounds difference approaches a certain limit as a function of the load factor. As we mentioned before in Aboul-Hassan and Rabia (2002, 2003), obtaining performance bounds using the present technique is very fast compared with simulation. Hence, whenever the tightness of bounds is acceptable, the performance bounds technique is preferable to obtain quick estimates of performance measures.

The present work can be extended in many directions. We mention here some of them. First, we considered here open Jackson networks. A similar analysis can be carried out for closed networks. This problem is under investigation by the authors. Second, we restricted the model here (and also in Aboul-Hassan and Rabia, 2002, 2003) to the single departure case. An extension to the batch departures case is required. Finally, adding a set of constraints to bring the obtained bounds tight for all working loads is still an open problem.

TABLE 4.4 *Neighborhood traffic with different arrival rates.*

	N=2			N=4		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.127	0.149	0.142	0.172	0.259	0.234
0.2	0.282	0.333	0.301	0.365	0.596	0.503
0.3	0.477	0.567	0.490	0.587	1.050	0.799
0.4	0.725	0.875	0.771	0.868	1.720	1.261
0.5	1.050	1.300	1.140	1.230	2.760	1.702
0.6	1.500	1.880	1.572	1.750	4.440	2.399
0.7	2.220	2.780	2.407	2.540	6.770	3.252
0.8	3.540	4.560	3.903	3.970	11.400	4.802
0.9	7.190	9.880	6.517	7.740	26.000	7.757
	N=6			N=8		
Load factor	*Lower*	*Upper*	*Simulation*	*Lower*	*Upper*	*Simulation*
0.1	0.231	0.351	0.326	0.284	0.436	0.420
0.2	0.488	0.813	0.696	0.600	1.010	0.817
0.3	0.775	1.460	1.111	0.951	1.820	1.384
0.4	1.100	2.420	1.601	1.340	3.050	1.964
0.5	1.510	3.960	2.213	1.820	4.950	2.778
0.6	2.080	6.400	3.250	2.470	8.040	3.626
0.7	2.930	10.400	4.138	3.420	12.900	4.993
0.8	4.410	19.100	5.897	5.050	23.600	7.261
0.9	8.250	50.000	10.460	9.110	80.600	12.300

ACKNOWLEDGEMENTS

The authors are grateful to an anonymous referee for his valuable comments and suggestions that led to presentation of the paper in its current form.

REFERENCES

ABOUL-HASSAN, A. AND RABIA, S. I. (2002). "Approximate analysis of discrete time queueing systems with early arrival scheme", *Presented in the Madrid Conference on Queueing Theory*, Madrid, June 2002.

ABOUL-HASSAN, A. AND RABIA, S. I. (2003). "Performance bounds for early arrival discrete time queueing systems", *Alexandria Engineering Journal*, **42**, 755–766.

ATENCIA, I. AND MORENO, P. (2004). "A discrete-time $Geo/G/1$ retrial queue with general retrial times", *Queueing Systems*, **48**, 5–21.

BHARATH-KUMAR, K. (1980). "Discrete time queueing systems and their networks", *IEEE Transactions on communications*, **COM-28**, 260–263.

BURKE, P. J. (1956). "The output of a queuing system", *Operations Research*, **4**, 699–704.

GELENBE, E. AND PUJOLLE, G. (1997). *Introduction to Queueing Networks*, 2nd ed., John Wiley & Sons, New York.

HENDERSON, W. AND TAYLOR, P. G. (1990). "Product form in networks of queues with batch arrivals and batch services", *Queueing Systems-Theory and Applications*, **6**, 71–87.

HSU, J. AND BURKE, P. J. (1976). "Behavior of tandem buffers with geometric input and Markovian output", *IEEE Transactions on communications*, **COM-24**, 358–361.

JACKSON, J. R. (1957). "Networks of waiting lines", *Operations Research*, **5**, 518–521.

KUMAR, S. AND KUMAR, P. R. (1994). "Performance bounds for queueing networks and scheduling policies", *IEEE Transactions on Automatic Control*, **39**, 1600–1611.

LI, H. AND YANG, T. (1998). "$Geo/G/1$ discrete time retrial queue with Bernoulli schedule", *European Journal of Operational Research*, **111**, 629–649.

NELSON, B. L. (1995). *Stochastic Modeling: Analysis and Simulation*, McGraw-Hill, New York.

WOLFRAM, S. (1996). *The Mathematica Book*, 3rd ed., Wolfram Media, Champaign.

WOODWARD, M. E. (1998). "Towards the accurate modelling of high speed communication networks with product form discrete-time networks of queues", *Computer Communications*, **21**, 1530–1543.