*Proceedings of the 1990 Winter Simulation Conference*
Osman Balci, Randall P. Sadowski, Richard E. Nance (eds.)

# A FAST SIMULATION APPROACH FOR TANDEM QUEUEING SYSTEMS

Liang Chen
Chien-Liang Chen

Department of Industrial Engineering
1513 University Avenue
University of Wisconsin-Madison
Madison, Wisconsin 53706

## ABSTRACT

This paper presents a new simulation approach, which is based on a recursive expression of sample path and can be applied to single-server tandem queueing systems. Numerical results show that compared with the event scheduling based simulation, the new simulation can dramatically save run time, particularly for large scale systems.

## 1. INTRODUCTION

There are two general approaches to discrete-event simulation modeling [Law and Kelton 1982; Banks and Carson 1984]. One is the event-scheduling approach used by some simulation languages, for example, GASP IV, SIMSCRIPT II.5, and SLAM. Another is process-interaction approach employed by GPSS, SIMSCRIPT II.5, and SLAM. Other approaches include transaction flow, three-phase, and activity scanning [Derrick et al. 1989]. Developing a simulator using general-purpose languages, such as C, Pascal, or FORTRAN, one is most likely to choose the event-scheduling approach [Banks and Carson 1984, pp. 52-62]. This approach results in relatively shorter run time (the CPU time needed for running simulation) [Nance 1971].

For a large scale queueing network, even with the event scheduling approach, the corresponding run time is still very long. In many practical situations, simulation speed becomes a significant factor that affects the period of engineering analysis and design. In this paper, we develop a new approach for single-server tandem queueing systems simulation, which can dramatically save run time, particularly for large scale systems. In what follows we use the term "traditional simulation" (TS) to mean simulation using an event scheduling approach and "fast simulation" (FS) to indicate the one employing the algorithm developed in this paper.

An open single-server tandem queueing system consists of a number of single servers in series, each server (say, i) is preceded by a buffer (i) of infinite or finite size. In a system with finite buffers, blocking may occur. Two types of blocking are commonly encountered in practice [Altiok and Stidham 1982; Perros and Altiok 1986; Brandwajn and Jow 1988], namely: manufacturing blocking and communication blocking. Suppose buffer i+1 has finite size, manufacturing blocking occurs if a customer sees that buffer i+1 is full as he completes service at server i. Then the customer has to wait at server i until buffer i+1 has some room available. On the other hand, if the first customer at buffer i sees that server i is empty, but buffer i+1 is full, he can not immediately enter server i to receive service and has to wait at buffer i until buffer i+1 has available space. This is referred to as communication blocking.

It is well known that an event scheduling based simulation contains an event list. When a new event is created, the event list must be searched for inserting the new event into a proper position according to the order of occurrence of events. Usually, this searching operation is quite time consuming, and the search time significantly increases as the number of servers in the system increases.

To avoid this time consuming searching operation, we notice that a set of recursive expressions of sample path have been proposed for tandem queueing systems with infinite buffer sizes [Chu and Naylor 1965; Saboo and Wilhelm 1986] and/or finite buffer sizes [Chen and Gao 1987; Chen and Suri 1989; Shanthikumar and Yao 1988]. From these recursive relationships among departure times of the customers, we find that the departure time of the jth customer from server i depends only on the service time of the jth customer at server i, the departure times of the first j-1 customers and the departure times of the jth customer from the first i-1 servers. This important observation results in a new simulation approach for single-server tandem queueing systems, where the departure times of customer 1 from server 1, server 2,..., server M is sequentially evaluated first, then the departure times of customer 2 from server 1 through server M, and so on. Figure 1 illustrates this procedure (see Section 3 for detail). We assume here that the system has M servers and N customers go through the system in one simulation run. The notation $d_{i,j}$ is used to denote the departure time of the customer j from server i.

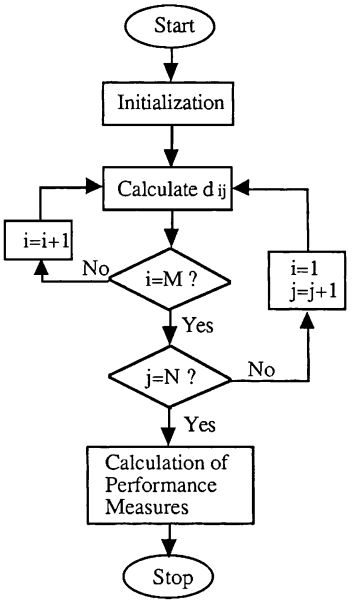


**Figure 1.** Procedure for the Fast Simulation Approach

It can be seen from Figure 1 that calculating each event time $d_{i,j}$ in the fast simulation needs at most two comparing operations of real numbers, rather than the searching operations of the event list in the traditional simulation. Moreover, in contrast to traditional simulation, the number of the comparing operations for each $d_{i,j}$ in the fast simulation is not affected by the number of servers in the system. Numerical results show that the run time required by the traditional simulator increases exponentially as the number of servers is increased, but the time needed by the fast simulator is a linear function of the number of servers in the system.

In Section 2 we derive the recursive expressions of departure times. Section 3 gives the algorithms of the fast simulations for three types of single-server tandem queueing systems, and derives formulas of various performance measures. Scenarios of the experimental design are described in Section 4. The analysis and discussion of numerical results are given in Section 5. Section 6 is a brief conclusion.

## 2. RECURSIVE EXPRESSIONS FOR DEPARTURE TIMES

In this section, we derive recursive expressions for single-server tandem queueing systems that have either infinite buffer sizes or finite buffer sizes. In the latter case, it may be subject to manufacturing blocking or communication blocking.

We first define the following notation:
$C_j$: the jth customer;
$S_i$: the ith server;
$a_j$: the arrival time of $C_j$ to the system;
b(i): the size of buffer i, including the one in service;
$s_{i,j}$: the service time of $C_j$ at $S_i$;
$e_{i,j}$: the starting time of service of $C_j$ at $S_i$;
$d_{i,j}$: the departure time of $C_j$ from $S_i$.
where all $a_j$'s equal zero for systems with infinite supply, i.e. customers are always available for the first server.

Throughout this paper, we always assume that there are M single-server workstations in the system and N single-class customers go through the system in a simulation run. First-Come-First-Served (FCFS) service discipline is employed.

### 2.1 Infinite Buffer Sizes

For single-server tandem queueing systems where all buffers are of infinite sizes, no blocking occurs. Thus $d_{i,j}$, the instant of $C_j$ leaving $S_i$, is just the instant when it starts service ($e_{i,j}$) at $S_i$ plus its service time ($s_{i,j}$):

$$d_{i,j} = e_{i,j} + s_{i,j} \qquad (2\text{-}1)$$

for i=1,...,M; j=1,...,N. Furthermore, if we define $d_{0,j}=a_j$ and $d_{i,j}=0$ if i>M or j≤ 0 or j>N, then the value of $e_{i,j}$ in (2-1) is itself different under the following two situations:

**Case a:** If $S_i$ is busy when $C_j$ leaves $S_{i-1}$, then $C_j$ can not enter $S_i$ to receive service immediately and has to wait at buffer i until $C_{j-1}$ departs from $S_i$:

$$e_{i,j} = d_{i,j-1} \qquad (2\text{-}2)$$

**Case b:** If $S_i$ is idle when $C_j$ leaves $S_{i-1}$, then $C_j$ will immediately enter $S_i$ to receive service:

$$e_{i,j} = d_{i-1,j} \qquad (2\text{-}3)$$

Note that in Case b, $C_{j-1}$ has departed from $S_i$ before $C_j$ departs from $S_{i-1}$, i.e., $d_{i-1,j} > d_{i,j-1}$, and conversely, $d_{i-1,j} < d_{i,j-1}$ in Case a. Summarizing the above two situations, $e_{i,j}$ can be expressed as

$$e_{i,j} = d_{i-1,j} \text{ V } d_{i,j-1} \qquad (2\text{-}4)$$

where the notation V denotes the maximizing operation: ∀ a, b ∈ R,

$$a \text{ V } b = \max(a,b) \qquad (2\text{-}5)$$

From (2-1) and (2-4), $d_{i,j}$ can be expressed as

$$d_{i,j} = d_{i-1,j} \text{ V } d_{i,j-1} + s_{i,j} \qquad (2\text{-}6)$$

for i=1,...,M; j=1,...,N, where $d_{0,j}=a_j$ and $d_{i,0}=0$. (2-6) is a simple recursive expression for $d_{i,j}$. It is also valid for the case where $d_{i-1,j} = d_{i,j-1}$.

### 2.2 Finite Buffer Sizes

We now extend (2-6) to such tandem queueing systems where some buffers are finite, and blocking may occur. We separately consider two types of blocking; manufacturing blocking and communication blocking.

#### *2.2.1 Manufacturing blocking*

It is easy to see that (2-6) holds as buffer i+1 has finite size and is not full. However, if buffer i+1 is full as $C_j$ completes service at $S_i$, then $C_j$ will be blocked by $S_{i+1}$. When this happens, the customer in service at $S_{i+1}$ must be $C_{j-b(i+1)}$. Thus, $C_j$ cannot leave $S_i$ until the instant $d_{i+1,j-b(i+1)}$, when $C_{j-b(i+1)}$ departs from $S_{i+1}$. In this case,

$$d_{i,j} = d_{i+1,j-b(i+1)} \qquad (2\text{-}7)$$

Clearly, this situation will occur only if $(d_{i-1,j} \text{ V } d_{i,j-1} + s_{i,j}) < d_{i+1,j-b(i+1)}$. That is, $C_j$ can not leave $S_i$ at the instant expressed by right-hand side of (2-6) and is delayed by the blocking. Summarizing the analysis in 2.1 and 2.2.1, we obtain the following recursive expression of $d_{i,j}$ for open tandem queueing systems that may be subjected to manufacturing blocking:

$$d_{i,j} = (d_{i-1,j} \text{ V } d_{i,j-1} + s_{i,j}) \text{ V } d_{i+1,j-b(i+1)} \qquad (2\text{-}8)$$

for i=1,...,M; j=1,...,N, where $d_{0,j}=a_j$ and $d_{i,j}=0$ if i>M or j≤ 0 or j>N.

The formula (2-8) is valid for systems with infinite or finite buffer sizes. In the case of an infinite buffer, say, buffer i+1, we always have j-b(i+1)<0, and so by the definition, the quantity $d_{i+1,j-b(i+1)}$ equals zero. Thus the formula (2-8) degenerates into (2-6).

#### *2.2.2 Communication blocking*

Like manufacturing blocking, the formula (2-6) holds for $d_{i,j}$ as buffer i+1 is not full. But, when buffer i+1 is full, the time of $C_j$ starting service at $S_i$ is delayed until the instant $d_{i+1,j+1-b(i+1)}$, when $C_{j-b(i+1)}$ departs from $S_{i+1}$:

$$e_{i,j} = d_{i+1,j-b(i+1)} \qquad (2\text{-}9)$$

Note that (2-9) holds only if $(d_{i,j-1} \text{ V } d_{i-1,j}) < d_{i+1,j+1-b(i+1)}$. Thus, for open tandem queueing systems with finite buffer size and communication blocking, there exists a recursive expression for $d_{i,j}$:

$$d_{i,j} = d_{i-1,j} \text{ V } d_{i,j-1} \text{ V } d_{i+1,j-b(i+1)} + s_{i,j} \qquad (2\text{-}10)$$

for i=1,...,M; j=1,...,N, where $d_{0,j}=a_j$, and $d_{i,j}=0$ if i>M or j≤ 0 or j>N. We notice that (2-10) also degenerates into (2-6) as all buffers have infinite sizes.

## 3. PERFORMANCE MEASURES AND FAST SIMULATION ALGORITHMS

In the fast simulation, we only accumulate $d_{i,j}$'s and $a_{i,j}$'s, by which the following sample performance measures can be evaluated:
W: sample average system time of one customer;
$L_i$: sample average number of customers at $S_i$;
$W_i$: sample average sojourn time of one customer at $S_i$;
$W_{si}$: sample average service time of one customer at $S_i$;
$W_{qi}$: sample average queueing time of one customer at buffer i;
$W_{bi}$: sample average blocking time of one customer at $S_i$;
$I_i$: sample idle time of $S_i$;
$U_i$: sample utilization of $S_i$;
TP: sample throughput rate of the system.

The system time of $C_j$ is $(d_{M,j}-a_j)$ for systems with finite supply and the sojourn time of $C_j$ at $S_i$ is $(d_{i,j}-d_{i-1,j})$. Thus, we have

$$W = \frac{1}{N}\sum_{j=1}^{N}(d_{M,j} - A_j) \qquad (3\text{-}1)$$

where $A_j = a_j$ if the system has finite supply, and $A_j = d_{1,j-1}$ if the system has infinite supply.

$$W_i = \frac{1}{N}\sum_{j=1}^{N}(d_{i,j} - d_{i-1,j}) \qquad (3\text{-}2)$$

The idle time of $S_i$ that is seen by $C_j$ is $\max(0, d_{i-1,j}-d_{i,j-1})$ for the systems with infinite buffer sizes or manufacturing blocking.

$$I_i = \sum_{j=1}^{N}\max[0, d_{i-1,j} - d_{i,j-1}] \qquad (3\text{-}3)$$

In (3-2) and (3-3), if a system has infinite supply, then $I_1 = 0$ and $W_1$ is the average service time of customers at server 1.

For systems with communication blocking, the idle time of $S_i$ that is seen by $C_j$ is

$$I_i' = I_i + \sum_{j=1}^{N}\max[0, c_{i,j}] \qquad (3\text{-}4)$$

where $c_{i,j} = d_{i+1,j-b(i+1)} - (d_{i-1,j} \vee d_{i,j-1})$. Furthermore, we have

$$U_i = \frac{d_{M,N} - I_i}{d_{M,N}} \qquad (3\text{-}5)$$

$$U_i' = \frac{d_{M,N} - I_i'}{d_{M,N}} \qquad (3\text{-}6)$$

$$TP = \frac{N}{d_{M,N}} \qquad (3\text{-}7)$$

where $U_i$ and $U'_i$ are the sample utilizations of $S_i$ for systems with manufacturing blocking and communication blocking respectively.

$W_i$ can be divided into two parts: $W_{si}$ and $W_{qi}'$ for systems having infinite buffer sizes or communication blocking (for the latter, we do not distinguish blocking time from queueing time), or into three part: $W_{si}$, $W_{qi}''$ and $W_{bi}$ for systems with manufacturing blocking. In any situation, we always have

$$W_{si} = \frac{1}{N}\sum_{j=1}^{N} s_{i,j} \qquad (3\text{-}8)$$

But, $W_{qi}'$ and $W_{qi}''$ are different. $W_{qi}'$ can be directly obtained by

$$W_{qi}' = W_i - W_{si} \qquad (3\text{-}9)$$

While $W_{qi}''$ should be evaluated by

$$W_{qi}'' = \frac{1}{N}\sum_{j=1}^{N}\max[0, d_{i,j-1} - d_{i-1,j}] \qquad (3\text{-}10)$$

For the system with infinite supply, $W_{q1}'' = 0$. Finally, we have

$$W_{bi} = \frac{1}{N}\sum_{j=1}^{N}\max[0, b_{i,j}] \qquad (3\text{-}11)$$

where $b_{i,j} = d_{i+1,j-b(i+1)} - (d_{i-1,j} \vee d_{i,j-1} + s_{i,j})$.

The average number of customers at one server can be derived by the above performance measures and Little's Law;

$$E[L_i] = E[TP]\,E[W_i] \qquad (3\text{-}12)$$

and the variance of $L_i$ can be obtained by the generalization of Little's Law (Gross and Harris 1974, p. 245).

$$Var[L_i] = E[TP]^2\{E[W_i^2] - E[W_i]^2\} + E[TP]\,E[W_i] \qquad (3\text{-}13)$$

Based on the recursive expressions (2-6), (2-8), (2-10) and the formulas (3-1) through (3-7) of performance measures, we can now formulate the following algorithms of the fast simulations for single-server tandem queueing systems. Algorithm 1 (Figure 2) is used for systems with infinite buffer sizes. Algorithms 2 and 3 (Figures 3 and 4) are for systems with manufacturing blocking and communication blocking respectively.

**Algorithm 1**
(1) Initialize:
Set $W_i$=0, $I_i$=0, W=0, $d_{i,j}$=0 and $d_{now}(i) = 0$,
i=1,...,M; j=1,...N.
Set i=1, j=1.
(2) Simulate:
If i=1, then $d_{i,j} = \max[a_j, d_{now}(i)] + s_{i,j}$,
otherwise, $d_{i,j} = \max[d_{now}(i-1), d_{now}(i)] + s_{i,j}$.
$W_i = W_i + d_{i,j} - d_{now}(i)$;
$I_i = I_i + \max[0, d_{now}(i-1) - d_{now}(i)]$;
$d_{now}(i) = d_{i,j}$.
(3) Update 1:
If i ≠ M, then i = i + 1, and return to (2),
otherwise, i = 1, W = W + $d_{now}(i)$ - $a_j$, and go to (4).
(4) Update 2 and Stopping Criterion:
If j ≠ N, then j = j + 1, and return to (2).
Otherwise, let
$W_i = W_i/N$ and $U_i = (d_{now}(i)-I_i)/d_{now}(i)$, i=1,..., M;
W = W/N;
TP = $N/d_{now}(i)$.
Stop.

**Figure 2.** The Algorithm 1 (Infinite Buffer Sizes)

**Algorithm 2**
(1) Initialize:
Set $W_i$=0, $I_i$=0, W=0, $d_{i,j}$=0, $d_{now}(i) = 0$ and $d_{block}(i,k) = 0$,
k=1,..., N-b(i+1); i=1,..., M; j=1,..., N.
Set i=1, j=1.
(2) Simulate:
If i=1, then
$d_{i,j} = \max\{ \max[a_j, d_{now}(i)] + s_{i,j}, d_{block}(i+1,j-b(i+1)) \}$*,
otherwise,
$d_{i,j} = \max\{\max[d_{now}(i-1),d_{now}(i)]+s_{i,j}, d_{block}(i+1,j-b(i+1)) \}$*.
$W_i = W_i + d_{i,j} - d_{now}(i)$;
$I_i = I_i + \max[0, d_{now}(i-1) - d_{now}(i)]$;
$d_{now}(i) = d_{i,j}$;
$d_{block}(i,j) = d_{i,j}$;
delete $d_{block}(i+1,j-b(i+1))$.
(* When j-b(i+1) ≤ 0, $d_{block}(i+1,j-b(i+1)) = 0$.)
(3) Update 1:
If i ≠ M, then i = i + 1, and return to (2),
otherwise, i = 1, W = W + $d_{now}(i)$ - $a_j$, and go to (4).
(4) Update 2 and Stopping Criterion:
If j ≠ N, then j = j + 1, and return to (2).
Otherwise, let
$W_i = W_i/N$ and $U_i = (d_{now}(i)-I_i)/d_{now}(i)$, i=1,..., M;
W = W/N;
TP = $N/d_{now}(i)$.
Stop.

**Figure 3.** The Algorithm 2 (Manufacturing Blocking)

**Algorithm 3**
(1) Initialize:
Set $W_i$=0, $I_i$=0, W=0, $d_{i,j}$=0, $d_{now}(i)$=0, and $d_{block}(i,k)$=0, k=1,..., N-b(i+1); i=1,..., M; j=1,..., N.
Set i=1, j=1.
(2) Simulate:
If i=1, then
$d_{i,j} = \max\{a_j, d_{now}(i), d_{block}[i+1,j-b(i+1)]\} + s_{i,j}$,*
otherwise,
$d_{i,j} = \max\{d_{now}(i-1), d_{now}(i), d_{block}[i+1,j-b(i+1)]\} + s_{i,j}$.*
$W_i = W_i + d_{i,j} - d_{now}(i)$;
$I_i = I_i + \max[0, d_{now}(i-1) - d_{now}(i)]$;
$d_{now}(i) = d_{i,j}$;
$d_{block}(i,j) = d_{i,j}$;
delete $d_{block}[i+1,j-b(i+1)]$.
(* When $j-b(i+1) \le 0$, $d_{block}[i+1,j+1-b(i+1)] = 0$.)
(3) Update 1:
If $i \neq M$, then $i = i + 1$, and return to (2),
otherwise, $i = 1$, $W = W + d_{now}(i) - a_j$, and go to (4).
(4) Update 2 and Stopping Criterion:
If $j \neq N$, then $j = j + 1$, and return to (2).
Otherwise, let
$W_i = W_i/N$ and $U_i = (d_{now}(i)-I_i)/d_{now}(i)$, i=1,..., M;
W = W/N;
$TP = N/d_{now}(i)$.
Stop.

**Figure 4.** The Algorithm 3 (Communication Blocking)

In Algorithm 1 through Algorithm 3, $d_{i,j}$ is stored in $d_{now}(i)$ temporarily for calculating $d_{i+1,j}$ and $d_{i,j+1}$. While $d_{block}[i+1,j-b(i+1)]$ in Algorithms 2 and 3 is used to record $d_{i+1,j-b(i+1)}$.

It can be seen that all three algorithms have identical step (3) and (4). Although step (1) and (2) in Algorithms 2 and 3 are not same, they have very similar structures. In fact, if a simulation program for Algorithm 2 has been developed, then it can be easily modified as a program for Algorithm 1 or Algorithm 3. Although only open tandem queueing systems are discussed here, these algorithms can be easily extended to closed tandem queueing systems.

## 4. EXPERIMENTAL DESIGN

Two different simulators, employing the event scheduling approach and the recursive approach separately, are developed for three types of single-server tandem queueing systems described in previous sections. For all these systems, we assume that there is only one type of customers going through the systems. The programs are written in C. All of the simulations in this study are run on an IBM/PS2 model 80 with 80386 processor and 80387 co-processor.

In order to compare the fast simulation with the traditional one, we apply the Common Random Number (CRN) technique to generate interarrival times and service times of customers. One generator is used to generate interarrival times. Moreover, each server has its own generator for generating service times. Under these considerations, the numerical results of performance measures of systems obtained by the fast and traditional simulators respectively are entirely identical. The only difference between the two sets of simulation runs are their run times. The performance measures evaluated in each simulation include all those discussed in the last section.

In the case of infinite buffer sizes, we consider $r_i$ (we use r to mean all $r_i$'s), i=1,...,M, the ratio of mean service time at server i to mean interarrival time, to be homogeneous (r=0.5 and r=0.9) and heterogeneous ($r_i \neq r_j$ when $i \neq j$). Experiments with various number of customers and number of servers are undertaken (see Table 1). In different simulation runs, we change the number of customers from 5,000 to 50,000 and the number of servers from 10 to 100. In the case of finite buffer sizes with manufacturing blocking, we run simulations with system configurations listed in Table 2, where each scenario consists of 10 experiments, i.e. 10, 20,..., 100 servers in the system respectively and the number of customers is set to 10,000 for each experiment. The experimental design for the case of finite buffer with communication blocking is exactly the same as that of the system with manufacturing blocking.

**Table 1.** Scenarios for Studying Systems With Infinite Buffer Sizes

| Customer # or WS # | r=0.5 | r=0.9 | Heterogeneous |
|---|---|---|---|
| 30 stations, change customer # | A | B | C |
| 30000 customers, change station # | D | E | F |

**Table 2.** Scenarios for Studying Systems With Finite Buffer Sizes (The Number of Customers = 10,000, Change The Number of Servers)

| Buffer Size | r=0.5 | r=0.9 | r=1.0 | Heterogeneous |
|---|---|---|---|---|
| 2 | G | H | I | J |
| 6 | K | L | M | N |
| 11 | O | P | Q | R |
| Unequal | S | T | U | V |

## 5. NUMERICAL RESULTS AND DISCUSSION

The numerical results of run times are shown in Appendix (Table A to V corresponding to scenarios A to V). Limited by space requirements, we do not present the results of communication blocking cases, which are very similar to the results of manufacturing blocking cases. Figure 5 (infinite buffer sizes, Table A to C) pictorially depicts the relationship between the run time and the number of customers. We observe that the run time is a linear function of the number of customers for both fast and traditional simulations. For example, the run time for 50,000 customers is approximately 10 times of that for 5,000 customers. Thus, the number of customers is not a significant factor that affects FS/TS, the ratio of the time required by the fast simulation to that needed by the traditional simulation.

Figure 6 (infinite buffer sizes, Table D to F) illustrates the relationship between the run time and the system size, i.e. the number of servers in the system. Figure 7, 8, 9 and 10 show the relationships between the run time and the system size when buffer sizes are 2 (Table G to J), 6 (Table K to N), 11 (Table O to R), and variant (Table S to V) respectively. The buffer size we mention here includes the one in server.

From simulation results, it is found that the run time needed by the fast simulation is always less than that required by the traditional simulation. When we fix the number customers and increase the system size, the run time of the traditional simulation increases exponentially. In contrast to this, by using the fast simulation developed in this paper, the run time has a linear relation with the system size. That means, the larger the system size becomes, the more significant the effect of the fast simulation on FS/TS is.

Another factor that may affect the run time is $r_i$'s. When $r_i$'s increase, the run time needed by the traditional simulation increases too. However, the run time required by the fast simulation keeps constant and is not influenced by the change of $r_i$'s. We can see this situation from Figure 7 to 10 where the lines indicating fast simulation with different r's pile together. Thus, in a large scale system where utilizations of some machines are relatively high (reflected by high $r_i$'s), the fast simulation can save much more run time.

For single-server tandem queueing systems with finite buffer sizes and manufacturing blocking, the ratio FS/TS may reach 25% if the system contains 100 servers and $r_i$'s are around 0.9. In another case where systems have infinite buffers and 100 servers, the ratio FS/TS may achieve 20%.

It should be mentioned that appropriate simulation analysis of a queueing system requires replicated runs for obtaining accurate performance measures of the system and derivation of confidence intervals. For a simple but large tandem queueing system with finite buffer sizes and high $r_i$'s (say, 100 servers and $r_i$'s are around 0.9), the traditional simulation needs more than 4 hours for one single run with 100,000 customers on an IBM/PS2 model 80. For the same

system, on the other hand, the fast simulation uses only about one hour.

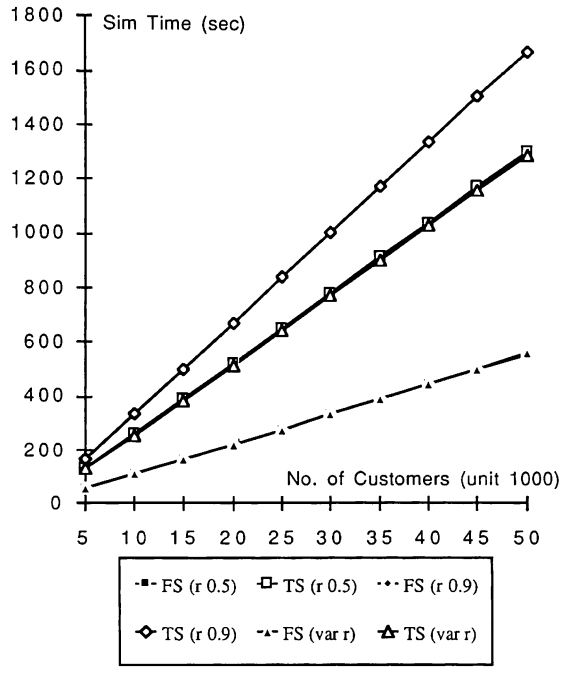


**Figure 5.** Infinite Buffer, 30 stations

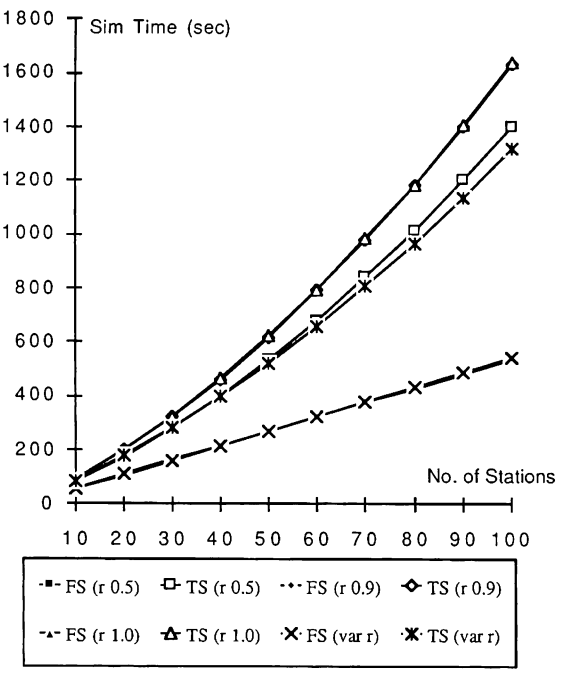


**Figure 7.** Finite Buffer (=2), 10,000 Customers

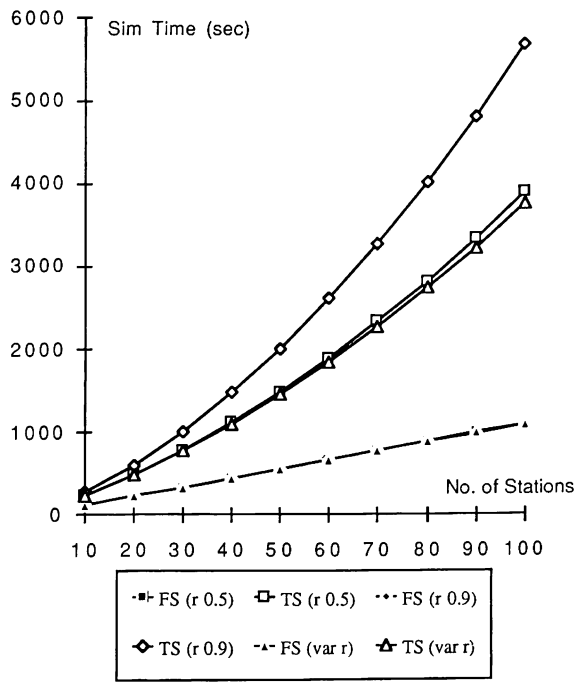


**Figure 6.** Infinite Buffer, 30,000 Customers

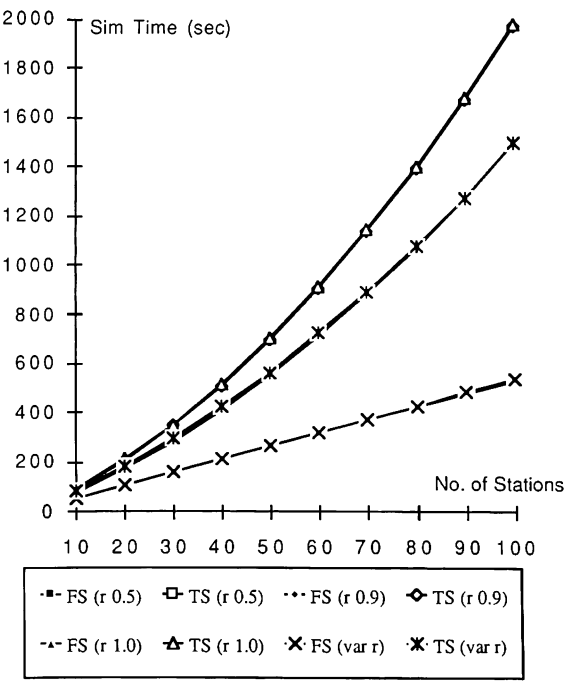


**Figure 8.** Finite Buffer (=6), 10,000 Customers

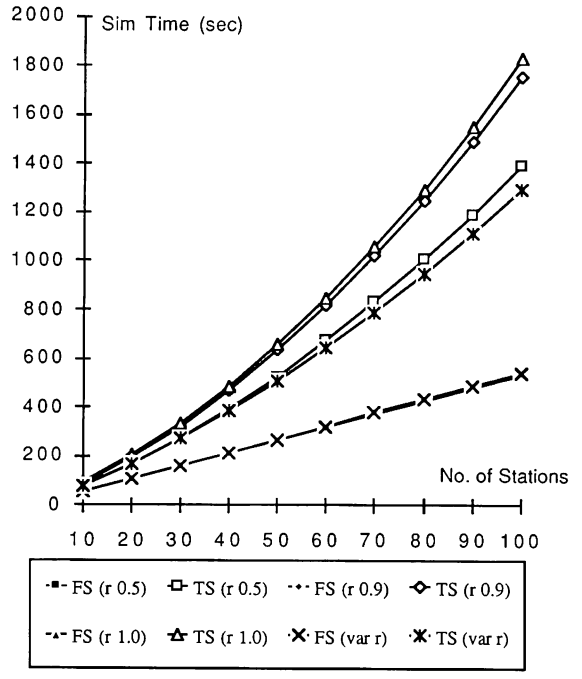


**Figure 9.** Finite Buffer (= variable), 10,000 Customers

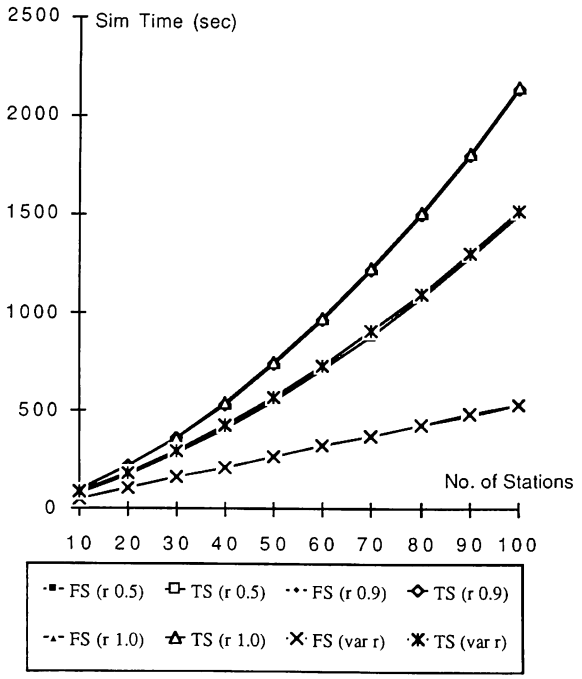


**Figure 10.** Finite Buffer (= 11), 10,000 Customers

## 6. CONCLUSIONS

The time needed for employing simulation to solve problems consists of simulator development and simulation implementation. For modeling the single-server tandem queueing systems, the program size of the fast simulator, such as the ones we developed, is less than half of that of the traditional simulator. Because the fast simulator does not use an event list, reduction in its program size is predictable. Besides, it is easier to develop the fast simulator since we do not have to manage the tedious event-related operations for the fast simulator.

As we have shown, the run time needed by the fast simulator is always less than the time required by the traditional simulator. In most of the cases we studied, the run time ratio FS/TS falls between 20% to 50%. That is, compared to traditional simulator, fast simulator may save 50% to 80% of run time. The ratio FS/TS decreases when the system size increases.

The recursive expression based fast simulation is a promising and exciting concept. It derives its strength in two aspects: simplicity and time saving. However, there are some limitations on the algorithms introduced in this paper. For example, these algorithms can not trace the distribution of the number of customers in a system unless a more time consuming procedure is added to the current algorithms. In addition, we have to make some assumptions, such as single server in each station, First-Come-First-Served (FCFS) service discipline, single type customers, reliable servers, and tandem systems.

The authors are currently working on the relaxation of the assumptions of single server station and FCFS discipline. Some progress has been made.

## APPENDIX A. TABULATED SIMULATION RESULTS

**Table A.** Simulation Results of Tandem System With 30 Stations,Infinite Buffers, r=0.5

| Cstm No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 5000 | 56 | 131 | 42.75% |
| 10000 | 112 | 260 | 43.08% |
| 15000 | 167 | 390 | 42.82% |
| 20000 | 224 | 521 | 42.99% |
| 25000 | 279 | 651 | 42.86% |
| 30000 | 335 | 781 | 42.89% |
| 35000 | 391 | 913 | 42.83% |
| 40000 | 447 | 1042 | 42.90% |
| 45000 | 502 | 1173 | 42.80% |
| 50000 | 558 | 1303 | 42.82% |

**Table B.** Simulation Results of Tandem System With 30 Stations, Infinite Buffers, r=0.9

| Cstm No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 5000 | 56 | 170 | 32.94% |
| 10000 | 112 | 335 | 33.43% |
| 15000 | 167 | 503 | 33.20% |
| 20000 | 224 | 669 | 33.48% |
| 25000 | 279 | 841 | 33.17% |
| 30000 | 335 | 1005 | 33.33% |
| 35000 | 390 | 1175 | 33.19% |
| 40000 | 447 | 1341 | 33.33% |
| 45000 | 502 | 1510 | 33.25% |
| 50000 | 558 | 1676 | 33.29% |

**Table C.** Simulation Results of Tandem System With 30 Stations,Infinite Buffers, r=0.1~0.9

| Cstm No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 5000 | 56 | 130 | 43.08% |
| 10000 | 112 | 258 | 43.41% |
| 15000 | 168 | 386 | 43.52% |
| 20000 | 223 | 515 | 43.30% |
| 25000 | 279 | 645 | 43.26% |
| 30000 | 335 | 775 | 43.23% |
| 35000 | 391 | 904 | 43.25% |
| 40000 | 447 | 1034 | 43.23% |
| 45000 | 503 | 1163 | 43.25% |
| 50000 | 559 | 1292 | 43.27% |

**Table D.** Simulation Results of Tandem System With 30,000 Customers, Infinite Buffers, r=0.5

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 117 | 228 | 51.32% |
| 20 | 225 | 486 | 46.30% |
| 30 | 335 | 781 | 42.89% |
| 40 | 445 | 1115 | 39.91% |
| 50 | 554 | 1485 | 37.31% |
| 60 | 664 | 1892 | 35.10% |
| 70 | 774 | 2338 | 33.11% |
| 80 | 885 | 2819 | 31.39% |
| 90 | 995 | 3339 | 29.80% |
| 100 | 1104 | 3897 | 28.33% |

**Table E.** Simulation Results of Tandem System With 30,000 Customers, Infinite Buffers, r=0.9

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 116 | 273 | 42.49% |
| 20 | 225 | 606 | 37.13% |
| 30 | 334 | 1006 | 33.20% |
| 40 | 445 | 1473 | 30.21% |
| 50 | 554 | 2005 | 27.63% |
| 60 | 664 | 2605 | 25.49% |
| 70 | 773 | 3274 | 23.61% |
| 80 | 884 | 4015 | 22.02% |
| 90 | 994 | 4818 | 20.63% |
| 100 | 1104 | 5700 | 19.37% |

**Table F.** Simulation Results of Tandem System With 30,000 Customers, Infinite Buffers, r=0.1~0.9

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 117 | 230 | 50.87% |
| 20 | 226 | 485 | 46.60% |
| 30 | 336 | 774 | 43.41% |
| 40 | 446 | 1099 | 40.58% |
| 50 | 555 | 1458 | 38.07% |
| 60 | 664 | 1852 | 35.85% |
| 70 | 774 | 2279 | 33.96% |
| 80 | 886 | 2739 | 32.35% |
| 90 | 995 | 3234 | 30.77% |
| 100 | 1106 | 3763 | 29.39% |

**Table G.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2, r=0.5

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 82 | 65.85% |
| 20 | 108 | 174 | 62.07% |
| 30 | 160 | 280 | 57.14% |
| 40 | 214 | 400 | 53.50% |
| 50 | 267 | 534 | 50.00% |
| 60 | 321 | 681 | 47.14% |
| 70 | 375 | 840 | 44.64% |
| 80 | 428 | 1014 | 42.21% |
| 90 | 482 | 1200 | 40.17% |
| 100 | 535 | 1400 | 38.21% |

**Table H.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2, r=0.9

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 55 | 92 | 59.78% |
| 20 | 110 | 198 | 55.56% |
| 30 | 163 | 321 | 50.78% |
| 40 | 217 | 460 | 47.17% |
| 50 | 270 | 617 | 43.76% |
| 60 | 325 | 791 | 41.09% |
| 70 | 378 | 979 | 38.61% |
| 80 | 432 | 1180 | 36.61% |
| 90 | 486 | 1400 | 34.71% |
| 100 | 540 | 1632 | 33.09% |

**Table I.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2, r=1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 57 | 94 | 60.64% |
| 20 | 110 | 201 | 54.73% |
| 30 | 164 | 324 | 50.62% |
| 40 | 217 | 463 | 46.87% |
| 50 | 271 | 622 | 43.57% |
| 60 | 325 | 795 | 40.88% |
| 70 | 378 | 982 | 38.49% |
| 80 | 433 | 1185 | 36.54% |
| 90 | 486 | 1404 | 34.62% |
| 100 | 540 | 1638 | 32.97% |

**Table J.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2, r=0.1~1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 56 | 83 | 67.47% |
| 20 | 109 | 177 | 61.58% |
| 30 | 162 | 282 | 57.45% |
| 40 | 216 | 397 | 54.41% |
| 50 | 269 | 522 | 51.53% |
| 60 | 323 | 660 | 48.94% |
| 70 | 376 | 809 | 46.48% |
| 80 | 431 | 967 | 44.57% |
| 90 | 485 | 1137 | 42.66% |
| 100 | 539 | 1319 | 40.86% |

**Table K.** Simulation Results of Tandem System With 10,000 Customers, Buffer=6, r=0.5

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 82 | 65.85% |
| 20 | 107 | 176 | 60.80% |
| 30 | 160 | 285 | 56.14% |
| 40 | 213 | 411 | 51.82% |
| 50 | 265 | 553 | 47.92% |
| 60 | 318 | 710 | 44.79% |
| 70 | 372 | 883 | 42.13% |
| 80 | 425 | 1072 | 39.65% |
| 90 | 478 | 1277 | 37.43% |
| 100 | 531 | 1499 | 35.42% |

**Table L.** Simulation Results of Tandem System With 10,000 Customers, Buffer=6, r=0.9

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 94 | 57.45% |
| 20 | 107 | 209 | 51.20% |
| 30 | 160 | 348 | 45.98% |
| 40 | 213 | 510 | 41.76% |
| 50 | 267 | 697 | 38.31% |
| 60 | 320 | 907 | 35.28% |
| 70 | 374 | 1141 | 32.78% |
| 80 | 427 | 1398 | 30.54% |
| 90 | 480 | 1677 | 28.62% |
| 100 | 534 | 1978 | 27.00% |

**Table M.** Simulation Results of Tandem System With 10,000 Customers, Buffer=6, r=1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 96 | 56.25% |
| 20 | 107 | 212 | 50.47% |
| 30 | 161 | 350 | 46.00% |
| 40 | 214 | 512 | 41.80% |
| 50 | 267 | 700 | 38.14% |
| 60 | 321 | 911 | 35.24% |
| 70 | 374 | 1145 | 32.66% |
| 80 | 427 | 1402 | 30.46% |
| 90 | 480 | 1682 | 28.54% |
| 100 | 534 | 1983 | 26.93% |

**Table N.** Simulation Results of Tandem System With 10,000 Customers, Buffer=6, r=0.1~1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 55 | 86 | 63.95% |
| 20 | 107 | 186 | 57.53% |
| 30 | 161 | 297 | 54.21% |
| 40 | 214 | 423 | 50.59% |
| 50 | 268 | 563 | 47.60% |
| 60 | 321 | 722 | 44.46% |
| 70 | 374 | 892 | 41.93% |
| 80 | 428 | 1081 | 39.59% |
| 90 | 481 | 1278 | 37.64% |
| 100 | 536 | 1502 | 35.69% |

**Table O.** Simulation Results of Tandem System With 10,000 Customers, Buffer=11, r=0.5

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 53 | 81 | 65.43% |
| 20 | 106 | 176 | 60.23% |
| 30 | 160 | 286 | 55.94% |
| 40 | 212 | 411 | 51.58% |
| 50 | 265 | 553 | 47.92% |
| 60 | 319 | 710 | 44.93% |
| 70 | 371 | 883 | 42.02% |
| 80 | 425 | 1074 | 39.57% |
| 90 | 478 | 1280 | 37.34% |
| 100 | 531 | 1501 | 35.38% |

**Table P.** Simulation Results of Tandem System With 10,000 Customers, Buffer=11, r=0.9

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 96 | 56.25% |
| 20 | 107 | 217 | 49.31% |
| 30 | 159 | 362 | 43.92% |
| 40 | 212 | 534 | 39.70% |
| 50 | 266 | 737 | 36.09% |
| 60 | 319 | 963 | 33.13% |
| 70 | 372 | 1220 | 30.49% |
| 80 | 425 | 1500 | 28.33% |
| 90 | 478 | 1804 | 26.50% |
| 100 | 531 | 2138 | 24.84% |

**Table Q.** Simulation Results of Tandem System With 10,000 Customers, Buffer=11, r=1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 98 | 55.10% |
| 20 | 107 | 218 | 49.08% |
| 30 | 160 | 365 | 43.84% |
| 40 | 213 | 538 | 39.59% |
| 50 | 266 | 743 | 35.80% |
| 60 | 319 | 970 | 32.89% |
| 70 | 372 | 1226 | 30.34% |
| 80 | 426 | 1507 | 28.27% |
| 90 | 478 | 1810 | 26.41% |
| 100 | 531 | 2145 | 24.76% |

**Table R.** Simulation Results of Tandem System With 10,000 Customers, Buffer=11, r=0.1~1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 84 | 64.29% |
| 20 | 107 | 186 | 57.53% |
| 30 | 160 | 299 | 53.51% |
| 40 | 213 | 429 | 49.65% |
| 50 | 267 | 572 | 46.68% |
| 60 | 320 | 731 | 43.78% |
| 70 | 373 | 902 | 41.35% |
| 80 | 427 | 1094 | 39.03% |
| 90 | 480 | 1302 | 36.87% |
| 100 | 534 | 1521 | 35.11% |

**Table S.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2~10, r=0.5

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 54 | 80 | 67.50% |
| 20 | 107 | 170 | 62.94% |
| 30 | 160 | 274 | 58.39% |
| 40 | 213 | 392 | 54.34% |
| 50 | 267 | 523 | 51.05% |
| 60 | 320 | 669 | 47.83% |
| 70 | 373 | 829 | 44.99% |
| 80 | 426 | 1001 | 42.56% |
| 90 | 479 | 1188 | 40.32% |
| 100 | 533 | 1388 | 38.40% |

**Table T.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2~10, r=0.9

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 55 | 89 | 61.80% |
| 20 | 108 | 196 | 55.10% |
| 30 | 161 | 322 | 50.00% |
| 40 | 214 | 468 | 45.73% |
| 50 | 267 | 633 | 42.18% |
| 60 | 321 | 818 | 39.24% |
| 70 | 374 | 1021 | 36.63% |
| 80 | 428 | 1246 | 34.35% |
| 90 | 481 | 1489 | 32.30% |
| 100 | 535 | 1751 | 30.55% |

**Table U.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2~10, r=1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 55 | 92 | 59.78% |
| 20 | 108 | 204 | 52.94% |
| 30 | 162 | 333 | 48.65% |
| 40 | 215 | 484 | 44.42% |
| 50 | 268 | 654 | 40.98% |
| 60 | 321 | 847 | 37.90% |
| 70 | 376 | 1059 | 35.51% |
| 80 | 429 | 1293 | 33.18% |
| 90 | 483 | 1545 | 31.26% |
| 100 | 536 | 1821 | 29.43% |

**Table V.** Simulation Results of Tandem System With 10,000 Customers, Buffer=2~10, r=0.1~1.0

| WS No | SimT(sec) FS | SimT(sec) TS | Ratio FS/TS |
|---|---|---|---|
| 10 | 55 | 80 | 68.75% |
| 20 | 109 | 172 | 63.37% |
| 30 | 161 | 272 | 59.19% |
| 40 | 215 | 385 | 55.84% |
| 50 | 268 | 508 | 52.76% |
| 60 | 322 | 643 | 50.08% |
| 70 | 376 | 787 | 47.78% |
| 80 | 429 | 944 | 45.44% |
| 90 | 483 | 1110 | 43.51% |
| 100 | 536 | 1287 | 41.65% |

## ACKNOWLEDGEMENTS

We wish to express our deep gratitude to Professor Conrad A. Fung and Professor Arne Thesen for their encouragement and support. We are grateful to Professor Rajan Suri for his valuable comments and suggestion. Our thanks also extend to G. J. Sheen, B. R. Fu, P. C. Rao, and R. Desiraju for their helpful remarks.

## REFERENCES

Altiok, T.M. and S.Jr. Stidham (1982), "A Note On Transfer Lines With Unreliable Machines, Random Processing Times, and Finite Buffers," *IIE Transactions 14*, 125-127.

Banks, J. and J.S. II Carson (1984), *Discrete-Event System Simulation*, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Brandwajn, A. and Y.L. Jow (1988), "An Approximation method for Tandem Queues with Blocking," *Operations Research 36*, 73-83.

Chen, L. and J.Q. Gao (1987), "A Time Series Model for General Queueing Networks," Research Report, Institute of Automation, Academia Sinica, Beijing, China.

Chen, L. and R. Suri (1989), "Convergence of Infinitesimal Perturbation Analysis for Tandem Queueing Systems with Blocking and General Service Times," Submitted to *IEEE Transactions on Automation Control.*

Chu, K. and T.H. Naylor (1965), "Two Alternative Methods for Simulating Waiting Line Models," *Journal of Industrial Engineering XVI*, 6, 390-394.

Derrick, E.J., O. Balci, and R.E. Nance (1989), "A Comparison of Selected Conceptual Frameworks for Simulation Modeling," In *Proceedings of the 1989 Winter Simulation Conference*, E.A. MacNair, K.J. Musselman, and P. Heidelberger, Eds. IEEE, Piscataway, NJ, 711-718.

Gross, D. and C.M. Harris (1974), *Fundamentals of Queueing Theory*, John Wiley & Sons, Inc., New York, NY.

Law, A.M. and W.D. Kelton (1982), *Simulation Modeling and Analysis*, McGraw-Hill Book Company, New York, NY.

Nance, R.E. (1971), "On Time Flow Mechanisms for Discrete System Simulation," *Management Science 18*, 1, 59-73.

Perros, H.G. and T.M. Altiok (1986), "Approximate Analysis of Open Networks of Queues with Blocking: Tandem Configurations," *IEEE Transactions on Software Engineering SE-12*, 450-461.

Saboo, S. and W.E. Wilhelm (1986), "An Approach for Modeling Small-Lot Assemble Networks," *IEE Transactions 18*, 322-334.

Shanthikumar, J.G. and D.D. Yao (1988), "Strong Stochastic Convexity: Closure Properties and Applications," Submitted for Publication.