

# Determining Transfer Batch Sizes in Trip-Based Material Handling Systems

YAVUZ A. BOZER

*Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109-2117*

JONGHWA KIM

*Department of Industrial Engineering, Konkuk University, Seoul, Korea*

**Abstract.** Trip-based material handling systems such as AGV systems or lift trucks are often designed with a given flow matrix (or from-to chart), which typically shows the number of loaded trips that the devices must perform per unit time between the workstations. A from-to chart that would result from the parts flow in a facility actually is dictated by the transfer batch size; that is, the number of parts transferred from one workstation to the next in one trip. In this paper, we present analytical and simulation results aimed at determining optimal or near-optimal transfer batch sizes in manufacturing systems and develop an analytical relationship between the material handling capacity and the expected work in process (WIP) in a manufacturing system. Although the results apply to any discrete-parts flow, trip-based material handling system, they are particularly relevant for the electronics manufacturing industry, where parts (such as printed circuit boards or substrates for flat panel displays) typically are handled as a group (in specially designed containers such as cassettes) and the costs associated with WIP tend to be large. In such applications, the cassette size is the transfer batch size.

**Key Words:** material handling, unit load sizing, cassette sizing, handling capacity planning

## 1. Introduction

Trip-based material handling systems consist of one or more handling devices that are self-powered and operate independent of each other (Srinivasan, Bozer, and Cho 1994). Examples of trip-based handling systems include unit load automated guided vehicle (AGV) systems, lift trucks, microload automated storage/retrieval (AS/R) systems, and bridge cranes, among others. In a trip-based handling system, a device is assumed to perform a trip to move the unit loads one at a time. (This assumption does not apply to tractor-trailer systems, where a device may pull multiple unit loads at the same time.) The *transfer batch size* (TBS) is defined as the number of parts in a unit load. In this study, we show that the TBS has a significant impact on the performance of the handling devices and the total expected work in process (WIP) in the system.

## 2. Problem description and motivation

We define a *part* as the smallest unit processed individually in the system; a *job* is defined as a set of parts that are identical in processing requirements. Parts are handled in a *container*;

which is the smallest possible unit moved by a device on one trip. A container may hold one or more parts (depending on the part size and weight); we assume each container holds only one part type. A *unit load* is a collection of containers moved together by a device on one trip. (Note that the smallest possible unit load size is one container.) Hence, the TBS is the number of parts or containers in a unit load, which may vary depending on the part type.

For example, in electronic assembly plants, circuit boards are handled in cassettes; see Figure 1(a). In such an application, a cassette of circuit boards would constitute a unit load, and each slot would be treated as a container. Another application of the transfer batch sizing problem is encountered in flat panel display manufacturing, where substrates are typically handled in cassettes. Determining the appropriate cassette size (i.e., the number of substrates per cassette) is a well-recognized and significant design problem in flat panel display manufacturing facilities, especially because “clean room” material handling equipment as well as WIP in the system tend to be large contributors to cost.

The transfer batch sizing problem, however, is not limited to the electronics industry. For example, in an automotive stamping plant, parts such as doors, side panels, or hoods are handled in a transportable rack as shown in figure 1(b). The rack consists of multiple slots and, as in the preceding example, each slot holds one part. A set of slots taken together (i.e., the rack) would be treated as a unit load, since a device moves only one rack on each trip. (Note that devices can move partially filled racks if the TBS of a particular part type is less than the number of slots in the rack.)

For simplicity, we assume that a container holds only one part. That is, the number of parts in a unit load is equal to the number of containers in a unit load (as shown in figure 1). If a container holds more than one part of a particular type, this assumption can be relaxed simply by setting the TBS of that part type equal to an integer multiple of the container size.

Consider next the manufacturing system. We define two types of workstations: input/output (I/O) stations and processing stations. Each workstation has a dedicated *input queue* and *output queue* of infinite capacity. Parts arriving from outside the system wait in the output queue of an I/O station, while parts that require no further processing are delivered to the input queue of an I/O station, where they leave the system instantly. External arrivals follow a Poisson process and each arrival consists of one or more parts depending

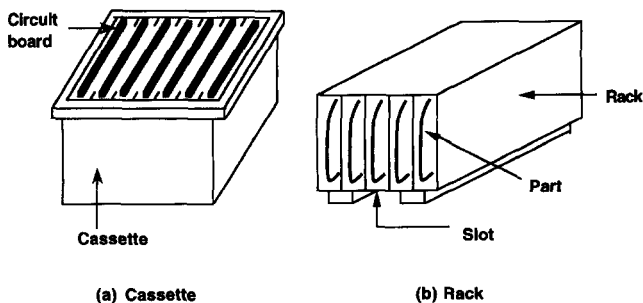


Figure 1. Examples of unit loads.

on the TBS of that part type. Flow is not necessarily conserved at an I/O station because parts may enter the system from one I/O station and exit from another.

At a *processing station*, unit loads delivered by a device are deposited at the input queue, where they wait until the processor is available. Parts are removed, one at a time, from the input queue on a first-come–first-served (FCFS) basis and processed for a given period. After a part is processed, it is staged by the processor until the desired TBS is reached, at which point the parts are placed in the output queue as a unit load. (Material handling concerns *within* the workstations are beyond the scope of our study.)

A unit load placed in an output queue (i.e., a “move request”) must wait for a handling device. When assigned to a particular move request, the device first travels empty to the output queue where the move request is located and then it delivers the load to the appropriate input queue. (Thus, each trip consists of empty travel, which may be of zero length, followed by loaded travel.) The next move request to be served by an empty device is determined by the (empty device) dispatching rule. We use the FCFS dispatching rule, where an empty device is assigned to the oldest move request in the system. Although the FCFS rule tends to increase empty device travel, it lends itself to analytical treatment more readily than other dispatching rules such as shortest travel time first (STTF). When a device becomes empty, if there are no unassigned move requests in the system, it becomes idle at its last delivery point. Also, at the time a move request occurs, if more than one idle device is available, the oldest idle device is dispatched.

In this study, we are concerned with the handling devices and the expected WIP in the system. The latter may be divided into four categories: (1) loads waiting in the input queues, (2) loads being processed and staged by the processors, (3) loads waiting in the output queues, and (4) loads being transferred. We will show that, for a fixed job arrival rate and fixed number of devices, the expected WIP in the input queues increases with the TBS (see figure 2). This is primarily due to “bulk arrivals”; that is, as the TBS increases, the expected number of parts per arrival instance at each input queue increases. In contrast, the expected WIP in the output queues generally decreases with the TBS, since the number of trips that the devices must perform per unit time decreases as the TBS is increased.

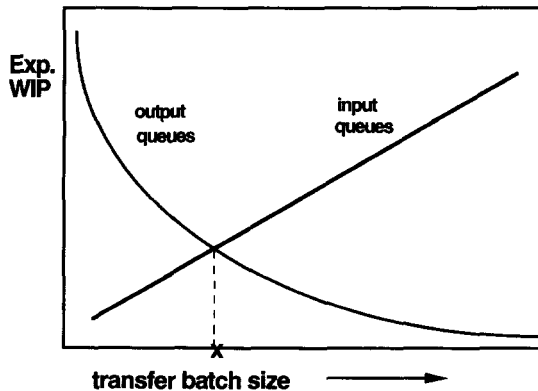


Figure 2. Transfer batch size vs. expected WIP.

Given the trade-off shown in figure 2, we are concerned with determining the optimal TBS to minimize the total expected WIP in the system (or the total cost associated with it). For this purpose, we develop separate analytical models to estimate the expected WIP in the input and output queues. For the former, we present a  $M^{(b)}/G/1$  approximation. For the latter, we present a  $M/G/c$  approximation, where we explicitly capture the empty device travel time as a function of the FCFS empty device dispatching rule.

Note that determining the optimal TBS is not the same as determining the optimal production lot size (PLS). The PLS is determined by examining the trade-offs between inventory holding costs and setup costs, while the TBS is determined by examining the trade-offs between WIP costs associated with the processors and the handling devices. In most manufacturing systems, the TBS generally is smaller than the PLS (unless, of course, PLSs of "1" become a reality). In some cases, one may have to set the TBSs relative to the PLSs. To keep our results general, however, we did not constrain the TBSs by the PLSs.

The assumptions for the study may be summarized as follows:

1. Multiple part types are processed in the system. The TBS of each part type is determined independently.
2. No setup times are considered for the processor stations. However, if the setup times are known in advance, they can be incorporated into the service time for the processor.
3. Each workstation has sufficient processing capacity and is utilized less than 100%.
4. At each input queue, unit loads are delivered at *random points in time* following a Poisson process (Srinivasan et al. 1994).
5. Move requests occur according to a Poisson process (Chow, 1986a).
6. The layout of the system, the production route and throughput requirement for each job, and the speed of the handling devices are given.
7. The first and second moments of the travel time distribution from one station to another are given. The first moment is obtained by dividing the corresponding distance by the device speed; that is, we do not consider congestion due to device interference.
8. The devices are homogeneous, and they each move one unit load at a time; a unit load consists of only one part type.

Assumptions 4 and 5 have not been validated in a theoretical sense. However, given a sufficient number of stations and the randomness induced by different production routes, as in Srinivasan et al. (1994), our empirical results indicate that the coefficient of variation for the interarrival times at the input queue of a workstation is consistently close to 1, which is a necessary but not sufficient condition for a Poisson process. Superimposing the move requests generated at many output queues, on the other hand, would generally approach a Poisson process (see Kuehn 1979 and Chow 1986a, among others).

### 3. Literature review

The expected WIP associated with the handling system depends in part on the dispatching rule. Few analytical models take the dispatching rule into account. Chow (1986a, 1986b)

presents an analytical model to approximate the device utilization and the overall expected (output queue) waiting time in a single-device system. Assuming infinite buffers and FCFS device dispatching, the author derives the first and second moments of the device service time distribution and models the system as a  $M/G/1/FCFS$  queue. The expected waiting time is obtained only for a single "conceptual" queue (as opposed to individual output queues). Chow does not extend the model to multiple device systems.

Cho (1990) also uses the  $M/G/1$  queue with FCFS dispatching to model single-device systems. He estimates the expected waiting times in (individual) output queues by "tagging" a move request. The author shows empirically that this model works well in single-device systems. To model systems with, say,  $K$  devices, he uses the single-device model and increases the speed of the device  $K$  times.

Yao and Buzacott (1985, 1986, and 1987) model the material handling system as a central server station. As in Chow's model, the expected waiting times in the output queues are estimated only for a single "conceptual" queue located ahead of the central station. Furthermore, each move request is treated in the same manner, regardless of its origin and destination. Solberg (1981) and Solot (1988) use a similar approach by modeling the material handling system as another workstation.

Bertrand (1985) extends the classical EOQ model to production shops with multiple stations. Using Solberg's model (1981) to estimate the expected time in the system for each job, he shows that ignoring the WIP carrying cost may result in considerably larger production batch sizes. Bertrand models the material handling system as another workstation as in Yao and Buzacott (1985, 1986, and 1987) and Solberg (1981).

Bozer, Cho, and Srinivasan (1994) use the "tagging" approach and develop an iterative algorithm to estimate the expected waiting times in the output queues for a single device operating under the MOD FCFS rule. The authors do not extend the model to multiple-device systems.

Egbelu (1993a) presents an optimization model to concurrently determine the container size (i.e., the TBS) and the number of handling devices. He assumes that only one container type is selected for all the part types. Hence, the weight capacity of the selected container dictates the TBS for each part type; that is,

$$\text{TBS of part type } j = \left( \frac{\text{the weight capacity of selected container}}{\text{the weight of part type } j} \right).$$

For example, if the container holds 500 pounds and a part weighs 100 pounds, then the TBS of that part type is set equal to 5 and TBSs of 1 through 4 are not evaluated. For each candidate container size, the author first uses a simulation model to estimate the expected WIP in the system as a function of the number of devices. He then uses these estimates in an optimization model to determine the optimal container size. In a subsequent model, in addition to the container size and the number of handling devices, Egbelu (1993b) uses a similar approach to determine the number of processors required.

Grasso and Tanchoco (1983) use an EOQ-type model to derive the optimal production batch sizes when the material handling cost, storage space cost, setup cost, and holding cost are considered. The material handling cost is obtained by multiplying a (user-specified)

unit cost per move with the total number of movements in a planning horizon. They show that including material handling and storage costs reduces the optimal order quantities and results in less total cost.

The TBS problem was also studied by Jackman (1991) as well as Mahadevan and Narendran (1992) in the context of unit load sizing. In Jackman (1991), material handling is modeled as an  $M/M/1$  system. In Mahadevan and Narendran (1992), the optimal unit load size is obtained through the solution of an integer programming problem, where material handling related unit costs are assumed to be supplied by the user.

The problem we address here is similar to the one studied by Egbelu (1993a). However, instead of simulation, we develop analytical models to estimate the expected WIP in the system. For each part type we allow any integer number (within the capacity limit of the container) as the TBS. Furthermore, we use a genetic algorithm to determine the "optimal" TBSs. Although the cost model we use is simple, it can be extended to include the cost elements used in Egbelu's model.

#### 4. Analytical model to estimate WIP

##### 4.1. Notation

The following notation is used throughout the paper. In the analytical model, subscripts  $h$ ,  $i$ , and  $j$  refer to a station, and  $k$  refers to a part type. Let

- $M$  = number of workstations in the system
- $JT$  = number of part types in the system
- $D_k$  = demand for part type  $k$  (parts/time unit) (= production rate)
- $Q_k$  = transfer batch size of part type  $k$
- $R_i$  = set of part types which require processing at workstation  $i$   
=  $\{k \mid \text{part type } k \text{ visits workstation } i\}$
- $\Omega_k$  = the set of workstations at which part type  $k$  requires processing  
=  $\{i \mid \text{workstation } i \text{ is visited by part type } k\}$
- $\Lambda_i$  = arrival rate at the input queue of workstation  $i$  (unit loads/time unit)
- $\Lambda_T$  = total arrival rate across all input queues ( $\sum_{i=1}^M \Lambda_i$ )
- $E(S_i)$  = expected processing time at workstation  $i$  (time units/part)
- $E(S_i^{(2)})$  = second moment of processing time at workstation  $i$
- $E(N_i)$  = expected number of parts in a unit load arriving at the input queue of workstation  $i$
- $E(J_i)$  = expected number of parts in a unit load arriving at the output queue of workstation  $i$  ( $E(N_i) = E(J_i)$  for processing stations)
- $\lambda_i$  = arrival rate at the output queue of workstation  $i$  (unit loads/time unit)
- $\lambda_T$  = total arrival rate across all output queues ( $\sum_{i=1}^M \lambda_i = \Lambda_T$ )

$ND$  = number of handling devices in the system

$p_{ij}$  = fraction of unit loads routed from workstation  $i$  to  $j$

$P$  = time required for a device to pick up or deposit a unit load (constant)

$\sigma_{ij}$  = expected empty travel time from workstation  $i$  to  $j$

$\sigma_{ij}^{(2)}$  = second moment of the empty travel time from workstation  $i$  to  $j$

$\tau_{ij}$  = expected loaded travel time from workstation  $i$  to  $j$  ( $\sigma_{ij} + 2P$ )

$\tau_{ij}^{(2)}$  = second moment of the loaded travel time from workstation  $i$  to  $j$

In the preceding list,  $Q_k$  is the primary decision variable, and  $\Lambda_i, \Lambda_T, E(N_i), E(J_i), \lambda_i, \lambda_T$ , and  $p_{ij}$  are functions of  $Q_k$ . All others are user-specified parameters. We implicitly assume that the empty and loaded travel time parameters ( $\sigma_{ij}, \sigma_{ij}^{(2)}, \tau_{ij}$ , and  $\tau_{ij}^{(2)}$ ) as well as the load pickup or deposit times ( $P$ ) are independent of the TBS. In many trip-based handling systems, the weight of the unit load typically has little or no impact on the average travel speed of the device.

4.2. Expected waiting times in the input queues

Parts arrive in bulk at the input queue of each workstation; the number of parts per unit load varies depending on the TBS of each part type. Using the  $M^{(b)}/G/1$  results given by Ross (1985), the expected waiting time of a part in the input queue of workstation  $i$ ,  $WI_i^p$ , can be obtained as follows:

$$WI_i^p = \frac{E(S_i)[E(N_i^{(2)}) - E(N_i)]/E(N_i) + \Lambda_i E(N_i)E(S_i^{(2)})}{2[1 - \Lambda_i E(N_i)E(S_i)]} \tag{1}$$

In equation (1), the arrival rate at each input queue,  $\Lambda_i$ , is given by

$$\Lambda_i = \sum_{k \in R_i} \frac{D_k}{Q_k} \tag{2}$$

The first and the second moments of the number of parts in a unit load arriving at the input queue of workstation  $i$  are given as follows:

$$E(N_i) = \frac{\sum_{k \in R_i} \frac{D_k}{Q_k} Q_k}{\sum_{l \in R_i} \frac{D_l}{Q_l}} = \frac{\sum_{k \in R_i} D_k}{\Lambda_i} \tag{3}$$

$$E(N_i^{(2)}) = \frac{\sum_{k \in R_i} \frac{D_k}{Q_k} Q_k^2}{\sum_{l \in R_i} \frac{D_l}{Q_l}} = \frac{\sum_{k \in R_i} D_k Q_k}{\Lambda_i} \tag{4}$$

Note that  $E(N_i)$  and  $E(N_i^{(2)})$  are assumed to be 0 for an I/O station since all the unit loads arriving at the input queue of an I/O station leave the system immediately. By substituting equations (2), (3), and (4) into equation (1), we obtain

$$WI_i^p = \frac{E(S_i)[\sum_{k \in R_i} D_k(Q_k - 1) / \sum_{k \in R_i} D_k] + E(S_i^{(2)}) \sum_{k \in R_i} D_k}{2[1 - \sum_{k \in R_i} D_k E(S_i)]} \tag{5}$$

On the other hand, the expected waiting time of a unit load in the input queue of workstation  $i$ ,  $WI_i^u$ , is expressed as follows:

$$\begin{aligned} WI_i^u &= WI_i^p - \text{waiting time due to the parts in the same unit load} \\ &= \frac{E(S_i)[E(N_i^{(2)}) - E(N_i)]/E(N_i) + \Lambda_i E(N_i) E(S_i^{(2)})}{2[1 - \Lambda_i E(N_i) E(S_i)]} \\ &\quad - \frac{E(S_i)[E(N_i^{(2)}) - E(N_i)]}{2E(N_i)}. \end{aligned} \tag{6}$$

By substituting equations (2), (3), and (4) into equation (6), and simplifying, we obtain

$$WI_i^u = \frac{E(S_i)^2 [\sum_{k \in R_i} D_k(Q_k - 1)] + E(S_i^{(2)}) (\sum_{k \in R_i} D_k)}{2[\sum_{k \in R_i} D_k][1 - \sum_{k \in R_i} D_k E(S_i)]} \tag{7}$$

### 4.3. Expected waiting times in the output queues

Under the FCFS dispatching rule, the expected empty travel time to workstation  $i$ ,  $E_i$ , is estimated as follows:

$$E_i = \sum_{h=1}^M \sum_{j=1}^M p_{hj} \sigma_{ji} \frac{\lambda_h}{\lambda_T} = \sum_{j=1}^M \frac{\Lambda_j}{\lambda_T} \sigma_{ji}, \tag{8}$$

since  $\sum_{h=1}^M p_{hj} \lambda_h = \Lambda_j$ . Note that, in equation (8), the term  $\sum_{j=1}^M p_{hj} \sigma_{ji}$  reflects the expected empty travel time to output queue  $i$  given that the last unit load (moved by the device) originated at station  $h$ , and the term  $\lambda_h/\lambda_T$  is the probability that the last unit load originated at station  $h$  (due to competing exponentials and the FCFS dispatching rule). Likewise, the second moment of the expected empty travel time to workstation  $i$ ,  $E_i^{(2)}$ , is given by

$$E_i^{(2)} = \sum_{j=1}^M \frac{\Lambda_j}{\lambda_T} \sigma_{ji}^{(2)}. \tag{9}$$

The expected loaded travel time from workstation  $i$ ,  $L_i$ , and its second moment,  $L_i^{(2)}$ , are easily obtained by



$$L_i = \sum_{j=1}^M p_{ij} \tau_{ij} \tag{10}$$

$$L_i^{(2)} = \sum_{j=1}^M p_{ij} \tau_{ij}^{(2)}, \tag{11}$$

where  $\tau_{ij}$  is set equal to  $\sigma_{ij} + 2P$  for simplicity.

Based on equations (8) and (10), the expected service time for a device to serve a move request at workstation  $i$ ,  $T_i$ , is given by

$$T_i = E_i + L_i, \tag{12}$$

and the second moment of this service time is given by

$$T_i^{(2)} = E_i^{(2)} + L_i^{(2)} + 2E_i L_i. \tag{13}$$

We note that the results given by equations (8) through (13) also are derived by Cho (1990) for single-device systems.

The expected device utilization is estimated as the total workload divided by the number of devices. The total workload for the handling system,  $WL$ , is given by

$$WL = \sum_{i=1}^M \lambda_i T_i. \tag{14}$$

Therefore, the expected device utilization,  $\rho$ , is given by

$$\rho = \sum_{i=1}^M \frac{\lambda_i T_i}{ND}, \tag{15}$$

since the handling system consists of  $ND$  homogeneous devices.

The device utilization consists of two components: the expected fraction of loaded travel and the expected fraction of empty travel. The former, say,  $\alpha_f$ , is easily obtained by

$$\alpha_f = \frac{1}{ND} \sum_{i=1}^M \lambda_i \sum_{j=1}^M p_{ij} \tau_{ij}, \tag{16}$$

or

$$\alpha_f = \frac{1}{ND} \sum_{k=1}^{JT} \frac{D_k}{Q_k} \sum_{(i < j) \in \Omega_k} \tau_{ij} \tag{17}$$

where  $(i < j) \in \Omega_k$  designates all the workstation pairs on part type  $k$ 's production route. Hence, the expected fraction of empty travel,  $\alpha_e$ , is given by

$$\alpha_e = \rho - \alpha_f. \tag{18}$$

We now estimate the expected waiting time of a *unit load* in each output queue. Under the FCFS rule, the move requests form a single ‘‘conceptual queue’’ that is served on a FCFS basis. Given Poisson arrivals for the move requests, we can use a  $(M/G/c)$  model with FCFS service to estimate the above expected waiting time. Of course, the service time parameters are based on the origin and destination of the move requests. Unlike central server models, we account for empty travel explicitly.

The expected waiting time for a  $M/G/c$  queue is given as (Nozaki and Ross 1978)

$$W_q = \frac{\lambda E(S^2)[\lambda E(S)]^{c-1}}{2(c-1)![c-\lambda E(S)]^2 \left[ \sum_{n=0}^{c-1} \frac{[\lambda E(S)]^n}{n!} + \frac{[\lambda E(S)]^c}{(c-1)![c-\lambda E(S)]} \right]}, \tag{19}$$

where  $c$  is the number of servers,  $S$  is the service time, and  $\lambda$  is the arrival rate. In equation (19),  $\lambda E(S)$  and  $\lambda E(S^2)$  represent the first and second moments of the total workload, respectively. The total workload for the handling system was derived earlier in equation (14), and the second moment of the total workload is equal to  $\sum_{i=1}^M \lambda_i T_i^{(2)}$ . So the expected waiting time of a *unit load* in an output queue is given by

$$W_q = \frac{\sum_{i=1}^M \lambda_i T_i^{(2)} \left[ \sum_{i=1}^M \lambda_i T_i \right]^{ND-1}}{2(ND-1)! \left[ ND - \sum_{i=1}^M \lambda_i T_i \right]^2 \left[ \sum_{n=0}^{ND-1} \frac{\left[ \sum_{i=1}^M \lambda_i T_i \right]^n}{n!} + \frac{\left[ \sum_{i=1}^M \lambda_i T_i \right]^{ND}}{(ND-1)! \left[ ND - \sum_{i=1}^M \lambda_i T_i \right]} \right]} \tag{20}$$

In deriving equation (20), the waiting time in the output queue is defined as the time spent by a move request from the instance of arrival to the instance of receiving service. In a trip-based handling system, while the device is traveling empty, the move request physically remains in the output queue. Therefore, the actual expected waiting time of a move request in the output queue of workstation  $i$ ,  $WO_i^u$ , is given by

$$WO_i^u = W_q + E_i. \tag{21}$$

#### 4.4. Expected time in the system

The expected time in the system for a *unit load* of part type  $k$ ,  $TW_k^u$ , is given by

$$TW_k^u = \sum_{(i < j) \in \Omega_k} (WI_i^u + PR_{ik}^u + WO_i^u + \tau_{ij}), \tag{22}$$

where  $PR_{ik}^u$  is the expected “time at the processor” at workstation  $i$ . The time at the processor is defined as the time spent by the first part of a unit load from the instance of its removal from the input queue to the instance of its placement in the output queue of workstation  $i$ . Since parts are “staged” by the server until all the parts in the unit load are processed,  $PR_{ik}^u$  is given by

$$PR_{ik}^u = Q_k E(S_i). \tag{23}$$

In equation (22), the right-hand side represents the expected time spent by a unit load at workstation  $i$  plus the expected travel time from workstation  $i$  to workstation  $j$ . Since parts are always handled as a unit load, the expected time a unit load spends at workstation  $i$  is equal to the expected time a part in that unit load spends at workstation  $i$ . The same holds true for travel times. Hence, the expected time a unit load spends in the system is equal to the expected time a part in that unit load spends in the system.

4.5. *Expected WIP in the system*

In this section, the expected WIP (in terms of the total number of parts) in the system is estimated by Little’s formula. First, the expected total WIP in the input queues,  $WIP_T^{in}$ , is given by

$$\begin{aligned} WIP_T^{in} &= \sum_{i=1}^M \Lambda_i E(N_i) W I_i^p \\ &= \sum_{i=1}^M \sum_{k \in R_i} D_k W I_i^p. \end{aligned} \tag{24}$$

Second, to estimate the expected total WIP in the output queues, we need the expected number of parts in a move request. The expected number of parts in a move request at workstation  $i$ ,  $E(J_i)$ , is obtained as

$$E(J_i) = \frac{\sum_{k \in R_i} \frac{D_k}{Q_k} Q_k}{\lambda_i} = \frac{\sum_{k \in R_i} D_k}{\lambda_i}. \tag{25}$$

Therefore, the total expected WIP in the output queues,  $WIP_T^{out}$ , is obtained as

$$\begin{aligned} WIP_T^{out} &= \sum_{i=1}^M \lambda_i E(J_i) W I_i^p \\ &= \sum_{i=1}^M \sum_{k \in R_i} D_k W O_i^u. \end{aligned} \tag{26}$$

Recall that  $E(J_i)$  is equal to  $E(N_i)$  for processing stations since flow is conserved (i.e.,  $\Lambda_i = \lambda_i$ ). However, at I/O stations,  $E(N_i)$  is assumed to be 0, and  $E(J_i)$  may be nonzero depending on the job routing.

Finally, the total expected WIP in the system,  $WIP_T^{sys}$ , is obtained as

$$WIP_T^{sys} = \sum_{k=1}^{JT} Q_k \frac{D_k}{Q_k} T W_k^u = \sum_{k=1}^{JT} D_k T W_k^u. \tag{27}$$

In the next section, the performance of this model, which we term the *WIP model*, is evaluated via simulation. Note that the WIP model is an approximate queuing model strictly due to assumptions 4 and 5 we presented in section 2. That is, for each station we treat the loaded device arrivals (at the input queue) and the move request arrivals (at the output queue) as two independent Poisson processes.

### 5. Evaluation of the WIP model

Three layouts were used to test the WIP model: layout 1 (Srinivasan et al. 1994), layout 2 (which we generated), and layout 3 (Egbelu 1987). Layout 1 is shown in figure 3. The distance matrix and additional data are shown in tables 1 and 2. (Data for layouts 2 and 3 are presented in the appendix.) For evaluation purposes, the processing time at each station is assumed to be exponentially distributed with the same mean regardless of part type; the mean values are selected such that the utilization of each processing station is equal to 0.70 (or 0.75). Also, the travel times are assumed to be exponentially distributed.

We compare the simulation results with those obtained from the WIP model. The simulation model simulates the “actual system”; that is, we do not force Poisson arrivals except

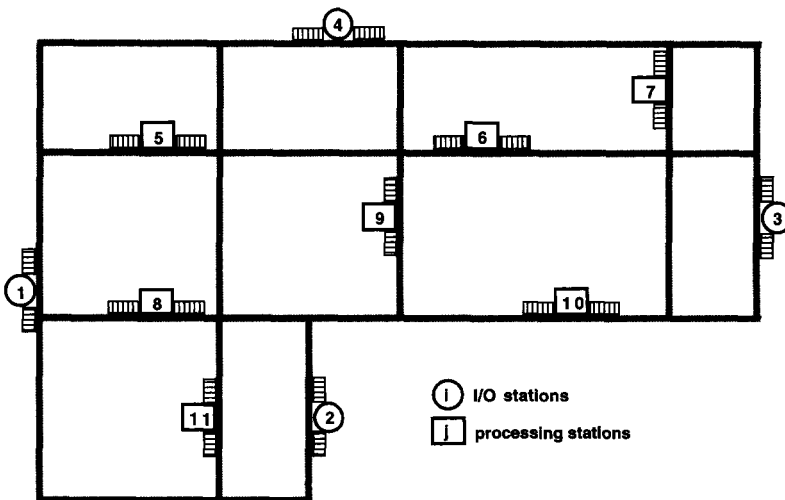


Figure 3. Layout 1.

Table 1. Distance matrix for layout 1.

Station number	1	2	3	4	5	6	7	8	9	10	11
1	0	22	47	30	14	32	46	8	27	31	17
2	22	0	36	29	23	24	38	14	16	20	14
3	47	36	0	33	37	19	12	39	27	16	41
4	30	29	33	0	16	14	21	25	13	28	27
5	14	23	37	16	0	18	32	16	17	32	18
6	32	24	19	14	18	0	14	27	8	23	29
7	46	38	12	21	32	14	0	41	22	18	43
8	8	14	39	25	16	27	41	0	19	23	9
9	27	16	27	13	17	8	22	19	0	15	21
10	31	20	16	28	32	23	18	23	15	0	25
11	17	14	41	27	18	29	43	9	21	25	0

Table 2. Job routes and throughput requirements 1 of layout 1.

Part type	Parts/min	Part Route 1.L1												
1	0.1	1	-	5	-	6	-	7	-	9	-	10	-	3
2	0.2	1	-	8	-	11	-	10	-	6	-	9	-	3
3	0.1	1	-	11	-	5	-	8	-	7	-	4	-	
4	0.1	2	-	10	-	9	-	7	-	6	-	4	-	
5	0.1	2	-	8	-	5	-	10	-	7	-	6	-	3
6	0.2	2	-	11	-	8	-	9	-	5	-	4	-	

Device speed: from 200 (with 1 device) to 50 (with 4 devices) distance units/min.  
 Pickup/deposit time is negligible.

for parts that arrive from outside the system. Simulation results are obtained from 10 replications, where at least 1,000 unit loads of each part type are processed through the system per replication. For simplicity, we varied only the TBS of part type 6 in table 2. Also, as we increase the number of devices, we proportionally reduce the device speed and increase the pickup/deposit times to maintain a comparable device utilization. In evaluating the WIP model, we observed similar results for the three layouts. Therefore, in this section we present only the results obtained with layout 1; results obtained with the other layouts are presented in the appendix.

5.1. Expected waiting times in the input queues

The results for layout 1 are presented in figure 4, where *WIP* represents the results obtained from the WIP model, while *sim 1*, 2, and 4 represent simulation results obtained with one, two, and four devices, respectively. (The numbers in parentheses show the station numbers that exhibited similar behavior.) As the TBS increases, the WIP model shows two types of trends. For “affected” workstations (i.e., those stations on the route of part type 6 such as stations 11 and 8), the WIP model overestimates the expected waiting times in the input queues. For “unaffected” stations, the simulation results suggest that the expected input

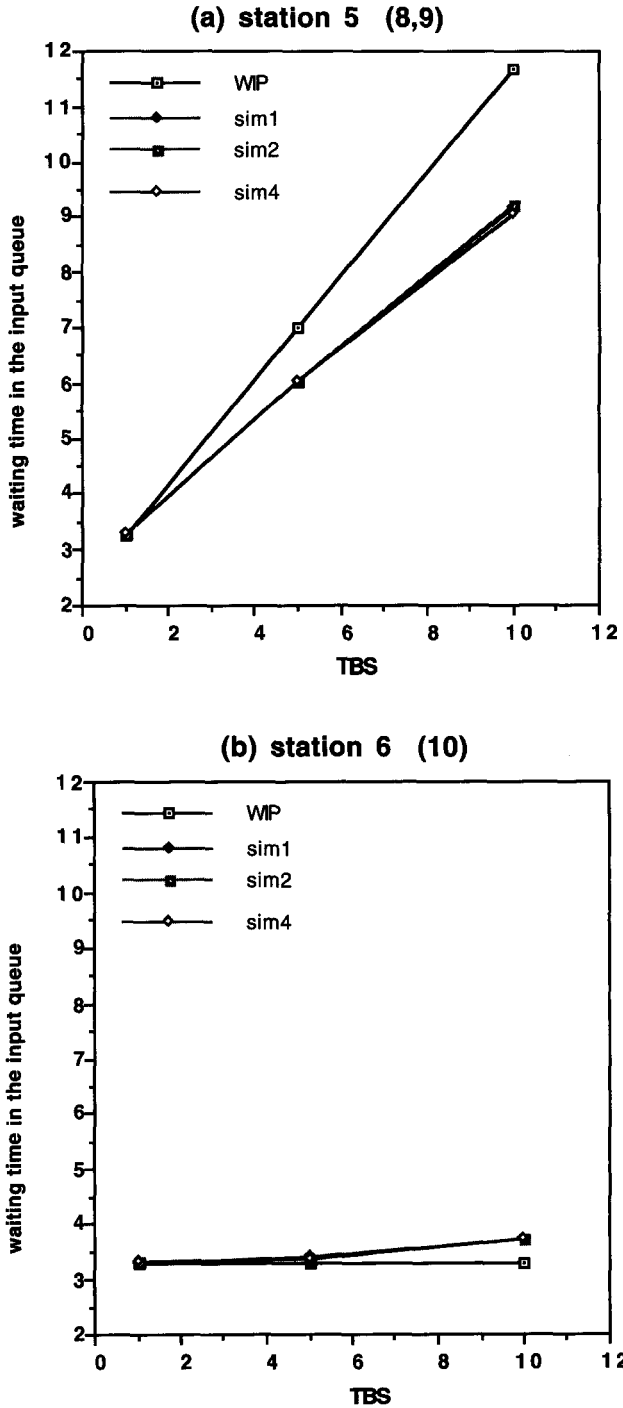


Figure 4. Expected waiting times in input queues of layout 1.

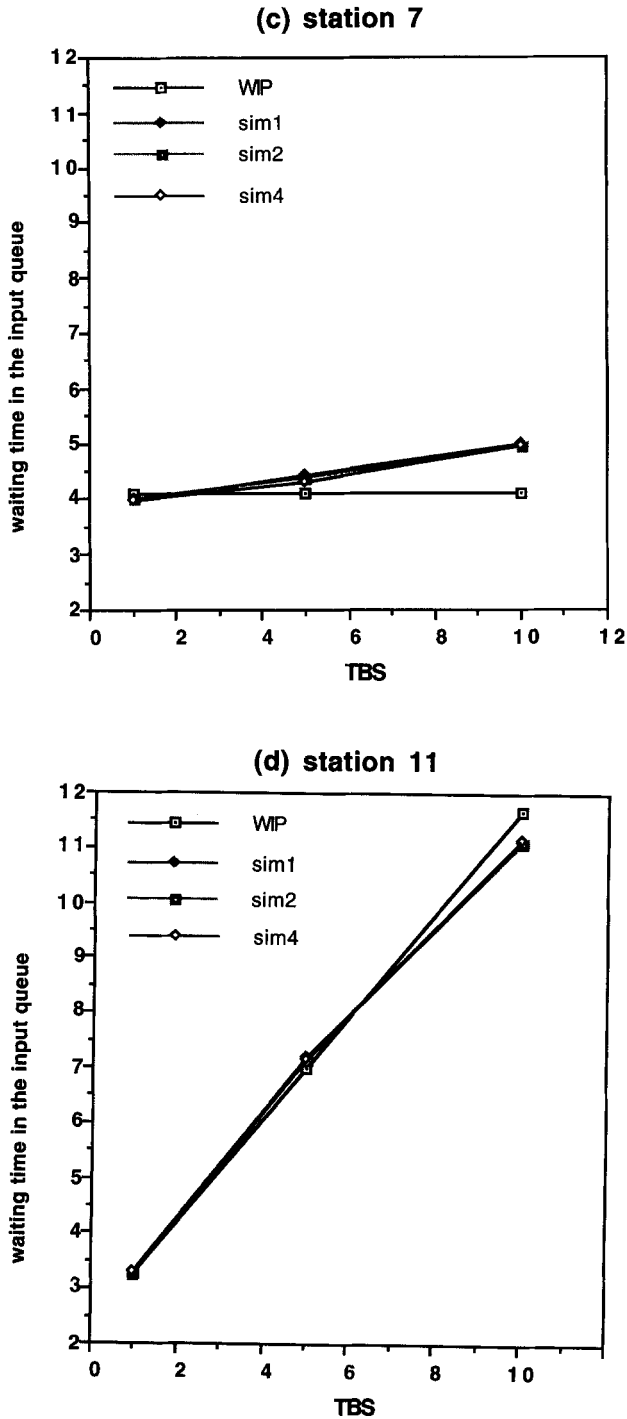


Figure 4. Continued.

queue waiting times increase slightly with the TBS. In the WIP model, however, the expected input queue waiting times at such stations are independent of the TBS and therefore remain constant. We also observe from both the WIP model and simulation results that the number of devices has little or no effect on the expected input queue waiting times.

In spite of these absolute differences, the *trends* shown by the WIP model generally agree with those seen from the simulation results. That is, as the TBS of part type 6 increases, the expected input queue waiting times increase or remain approximately the same depending on the type of station.

### 5.2. *Expected waiting times in the output queues*

The expected output queue waiting times obtained with one, two, and four devices are shown in table 3. (Due to limited space, we will not show these results graphically.) In general, the WIP model estimates the expected output queue waiting times reasonably well, regardless of the TBS or the number of devices. The maximum relative error is 13%, and in many cases it is less than 5%. Also, the absolute errors are fairly small. For a fixed job arrival rate and fixed number of devices, the expected waiting time in each output queue decreases as the TBS increases, since the number of unit loads that must be moved per time unit decreases as the TBS increases.

### 5.3. *Expected device utilization*

In table 4 we present the expected device utilizations obtained from the WIP model and by simulation for TBSs of 1, 5, and 10 for part type 6. The results obtained from the WIP model do not vary with the number of devices, since we adjust the device speed and the pickup/deposit times as the number of devices varies. The results in table 4 indicate that the WIP model slightly overestimates the expected device utilization. However, the error is less than 2% in all cases.

### 5.4. *Total expected WIP in the system*

The original data for layout 1 (see table 2) was presented in Srinivasan et al. (1994). In the remainder of the paper, however, for layout 1 we use the alternate data shown in table 5, which is designed to emphasize the throughput differences between the part types and also to control the stations visited by each part type. Using six devices, and increasing the TBS from 1 through 10 for each part type, one at a time, we obtained the expected WIP results shown in figure 5. (The results for part types 4 and 5 are not shown since they are almost identical to those obtained for part type 1.)

Depending on the TBS, the WIP model underestimates the expected output queue WIP and overestimates the expected input queue WIP. However, these errors are not significant for our purposes, and as shown in figure 5, the WIP model tracks the expected total



Table 3. Expected waiting times in output queues of Layout 1.

TBS	St. no	1	2	3	4	5	6	7	8	9	10	11	Avg.	Max	
One device	1	WIP	1.106	1.086	0.000	0.000	1.072	1.057	1.091	1.076	1.048	1.072	1.084		
		sim	1.121	1.105	0.000	0.000	1.092	1.092	1.135	1.095	1.066	1.106	1.107		
		95% C.I. % error	0.050 1.40%	0.053 1.72%	0.000 0.00%	0.000 0.00%	0.049 1.85%	0.048 3.14%	0.047 3.83%	0.048 3.14%	0.052 1.74%	0.053 1.63%	0.051 3.05%	0.051 2.02%	2.27%
	5	WIP	0.572	0.547	0.000	0.000	0.537	0.513	0.540	0.541	0.507	0.524	0.550		
		sim	0.613	0.588	0.000	0.000	0.555	0.562	0.594	0.546	0.523	0.578	0.553		
		95% C.I. % error	0.012 6.74%	0.016 6.97%	0.000 0.00%	0.000 0.00%	0.010 3.26%	0.011 8.73%	0.012 9.14%	0.011 8.73%	0.011 9.95%	0.010 3.10%	0.012 9.34%	0.010 0.58%	5.42%
	10	WIP	0.539	0.514	0.000	0.000	0.504	0.478	0.505	0.508	0.474	0.490	0.518		
		sim	0.590	0.596	0.000	0.000	0.524	0.540	0.579	0.523	0.505	0.563	0.518		
		95% C.I. % error	0.009 8.68%	0.016 9.71%	0.000 0.00%	0.000 0.00%	0.010 3.69%	0.008 11.47%	0.012 12.91%	0.009 2.87%	0.007 6.31%	0.010 12.99%	0.009 12.99%	0.009 0.10%	7.64%
Two devices	1	WIP	1.162	1.122	0.000	0.000	1.095	1.066	1.133	1.102	1.047	1.095	1.119		
		sim	1.174	1.144	0.000	0.000	1.114	1.093	1.171	1.121	1.062	1.128	1.139		
		95% C.I. % error	0.043 1.01%	0.047 1.89%	0.000 0.00%	0.000 0.00%	0.044 1.64%	0.041 2.53%	0.043 3.24%	0.041 2.53%	0.048 1.64%	0.045 1.35%	0.042 2.95%	0.049 1.76%	2.00%
	5	WIP	0.642	0.593	0.000	0.000	0.573	0.524	0.578	0.580	0.513	0.547	0.599		
		sim	0.672	0.624	0.000	0.000	0.577	0.560	0.617	0.576	0.516	0.586	0.590		
		95% C.I. % error	0.010 4.51%	0.014 5.01%	0.000 0.00%	0.000 0.00%	0.011 0.76%	0.010 6.52%	0.011 6.33%	0.007 7.5%	0.007 6.62%	0.012 6.0%	0.010 6.62%	0.010 1.47%	3.62%
	10	WIP	0.611	0.561	0.000	0.000	0.542	0.490	0.542	0.550	0.480	0.513	0.568		
		sim	0.671	0.622	0.000	0.000	0.564	0.552	0.620	0.565	0.513	0.588	0.570		
		95% C.I. % error	0.007 8.87%	0.013 9.78%	0.000 0.00%	0.000 0.00%	0.007 3.85%	0.009 11.25%	0.013 12.48%	0.008 2.62%	0.008 6.41%	0.010 6.41%	0.010 12.82%	0.007 0.33%	7.60%
Four devices	1	WIP	1.326	1.246	0.000	0.000	1.192	1.133	1.268	1.206	1.096	1.191	1.240		
		sim	1.343	1.270	0.000	0.000	1.209	1.162	1.299	1.223	1.111	1.223	1.258		
		95% C.I. % error	0.044 1.29%	0.046 1.83%	0.000 0.00%	0.000 0.00%	0.041 1.37%	0.039 2.48%	0.042 2.9%	0.042 2.48%	0.045 1.40%	0.045 1.31%	0.041 2.60%	0.041 1.41%	1.79%
	5	WIP	0.838	0.741	0.000	0.000	0.700	0.602	0.711	0.716	0.580	0.649	0.752		
		sim	0.870	0.780	0.000	0.000	0.689	0.631	0.744	0.700	0.575	0.684	0.729		
		95% C.I. % error	3.75% 0.812	5.09% 0.712	0.000 0.00%	0.000 0.00%	1.66% 0.674	4.49% 0.570	4.46% 0.675	4.46% 0.675	2.24% 0.690	0.96% 0.550	5.03% 0.615	3.18% 0.726	3.43%
	10	WIP	0.863	0.778	0.000	0.000	0.666	0.621	0.729	0.675	0.570	0.675	0.694		
		sim	0.907	0.818	0.000	0.000	0.674	0.607	0.708	0.675	0.570	0.675	0.694		
		95% C.I. % error	5.94% 8.51%	8.51% 8.51%	0.000 0.00%	0.000 0.00%	1.16% 8.19%	8.19% 8.19%	7.41% 7.41%	2.24% 3.52%	0.008 0.008	0.008 3.52%	0.006 8.90%	0.006 4.59%	5.61%

Table 4. Expected device utilization of layout 1.

TBS of part type 6	One device		Two devices		Four devices	
	WIP	simulation (95% C.I.)	WIP	simulation (95% C.I.)	WIP	simulation (95% C.I.)
1	0.8536	0.8507 (0.0051)	0.8536	0.8494 (0.0050)	0.8536	0.8494 (0.0053)
5	0.7150	0.7130 (0.0037)	0.7150	0.7112 (0.0038)	0.7150	0.7101 (0.0046)
10	0.6972	0.6901 (0.0036)	0.6972	0.6891 (0.0032)	0.6972	0.6885 (0.0038)

Table 5. Job routes and throughput requirements 2 of layout 1.

Part type	Parts/min	Part route 2.L1												
1	0.01	1	-	5	-	6	-	9	-	11	-	10	-	3
2	0.03	1	-	7	-	5	-	8	-	10	-	11	-	3
3	0.08	1	-	6	-	7	-	10	-	9	-	4	-	
4	0.01	2	-	5	-	7	-	8	-	11	-	6	-	3
5	0.01	2	-	7	-	9	-	8	-	6	-	11	-	4
6	0.02	2	-	5	-	8	-	9	-	10	-	4	-	

Device speed: from 100 (with 1 device) to 20 (with 4 devices) distance units/min. Negligible pickup/deposit time.

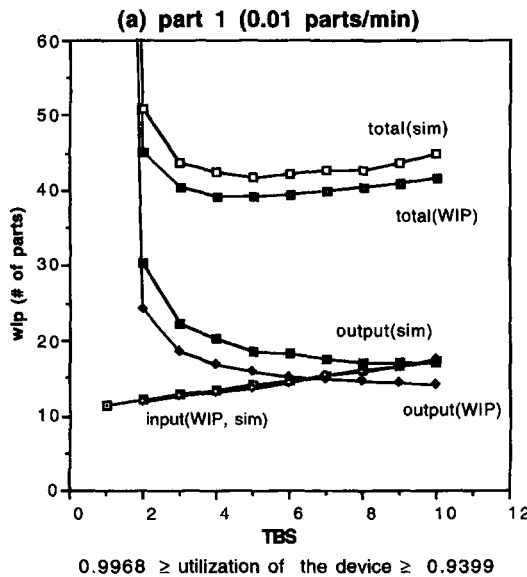


Figure 5. Expected total WIP in the system.

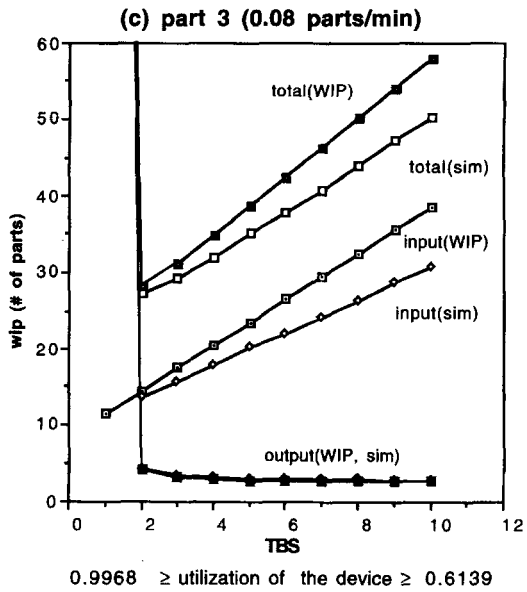
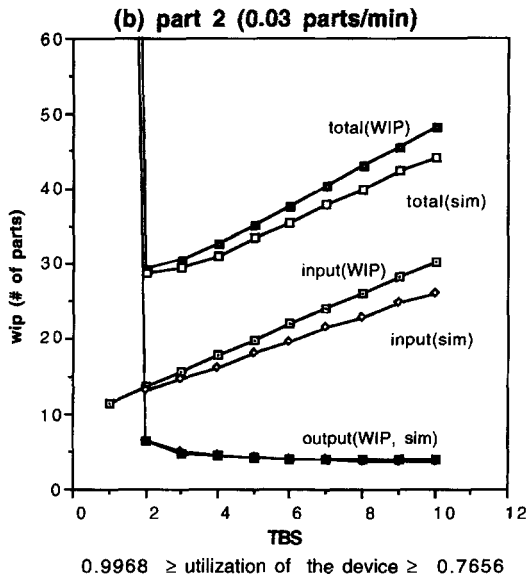


Figure 5. Continued.

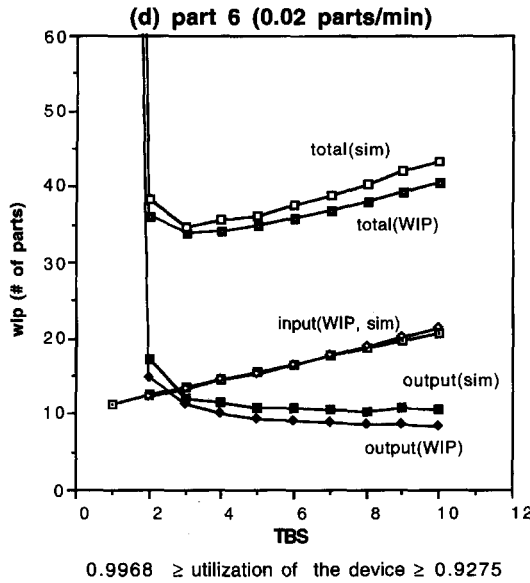


Figure 5. Continued

WIP curve (obtained from simulation) rather closely over a wide range of TBSs and device utilizations. Hence, for optimization purposes, the WIP model appears satisfactory. (For a manufacturing cell with setup times, which are analogous to empty travel time in material handling, Karmarkar et al. (1985) report results similar to those shown in figure 5.) In the next section, we formally present the objective function and evaluate the robustness of the TBSs obtained via the WIP model.

**6. Determining the “optimal” transfer batch size**

In this section, within the context of transfer batch sizing, we present alternative formulations to minimize the WIP or material handling related costs in a manufacturing system and show the “optimal” TBSs obtained via exhaustive enumeration and simulation. We also present a heuristic optimization scheme based on a genetic algorithm (GA).

*6.1. Alternative formulations*

Assuming a fixed number of devices, a simple formulation of the problem may be presented as follows:

$$\begin{aligned}
 \text{(P1)} \quad & \text{Min} && C_W \text{WIP}_T^{\text{sys}} \\
 & \text{s.t.} && \rho < 1 \\
 & && Q_k \leq \text{UB}_k, \quad k = 1, \dots, JT \\
 & && Q_k : \text{positive integer.}
 \end{aligned}$$

where  $C_W$  is the WIP cost per part per time unit and  $UB_k$  is the upper bound on the TBS of part type  $k$ . (Note that  $UB_k$  depends on the container size and the container or device weight capacity.) Of course, we must ensure that the device utilization is less than 1.0.

In the preceding formulation (P1), we use a single estimate for the WIP carrying cost. However, in most manufacturing systems, since more “value” is added as the parts are processed, the WIP carrying cost at a particular station may depend on the part type. The WIP model we present can accommodate such a case because the expected input queue waiting time at a station is independent of the part type (due to FCFS service) and the arrival rate (in parts) of a part type at a workstation is given.

Also, if necessary, two different cost coefficients can be used for the expected WIP due to the processors and to the handling system, since the WIP model estimates them separately. The expected WIP due to the handling system consists of  $WIP_T^{out}$  and the expected number of parts being transferred. The latter is obtained simply from the expression  $PU \cdot \alpha_f \cdot ND$ , where  $PU$ , the expected number of parts in a unit load, is equal to  $\sum_{i=1}^M \lambda_i E(J_i) / \lambda_T$ . The expected WIP due to the processors is obtained by subtracting the expected WIP due to the handling system from the expected total WIP in the system.

For new systems, the number of devices (i.e., the fleet size) is an important design variable since it affects the overall cost. The transfer batch sizing problem with a variable fleet size can be formulated as follows:

$$\begin{aligned}
 \text{(P2)} \quad & \text{Min} && C_W WIP_T^{sys} + C_D ND \\
 & \text{s.t.} && \rho < 1 \\
 & && Q_k \leq UB_k, \quad k = 1, \dots, JT \\
 & && C_D ND \leq B \\
 & && Q_k: \text{positive integer.}
 \end{aligned}$$

where  $B$  is the budget for the material handling system, and  $C_D$  is the equivalent cost per time unit per device. (Device maintenance costs can be included in  $C_D$ .)

Space cost for each unit space in the input or output queues may be included in the objective function as in Grasso and Tanchoco (1983). Since we assume infinite queue capacities, however, we will not address the space cost. Although we can also add cost elements such as those shown by Egbelu (1993a), for simplicity we will use only the WIP carrying cost and the material handling cost in our numerical experiments.

### 6.2. Computational results

Setting  $C_W = 1$  and  $UB_k = 10$  for all  $k$  in P1, we use layout 1 with the data shown earlier in table 5. The number of devices is varied from one to six (without adjusting the device speed and P/D times). The “optimal” TBSs are obtained through exhaustive enumeration; that is, we consider all possible feasible TBSs and use the WIP model to estimate the resulting expected WIP.

In figure 6, we present the 10 best solutions determined by the preceding procedure. (Results obtained with three and four devices are omitted since they are virtually identical to those obtained with five devices). The solutions are shown as vectors, where the  $k$ th component is the TBS of part type  $k$ . Simulation results obtained for each vector also are

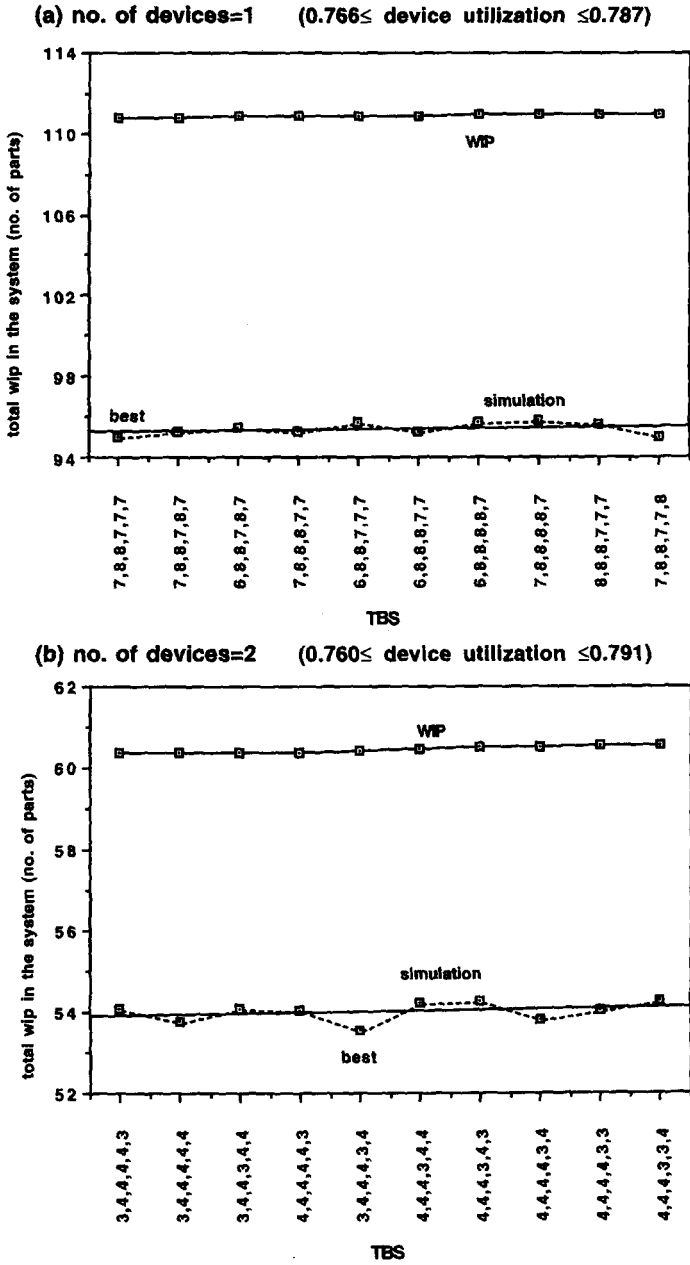
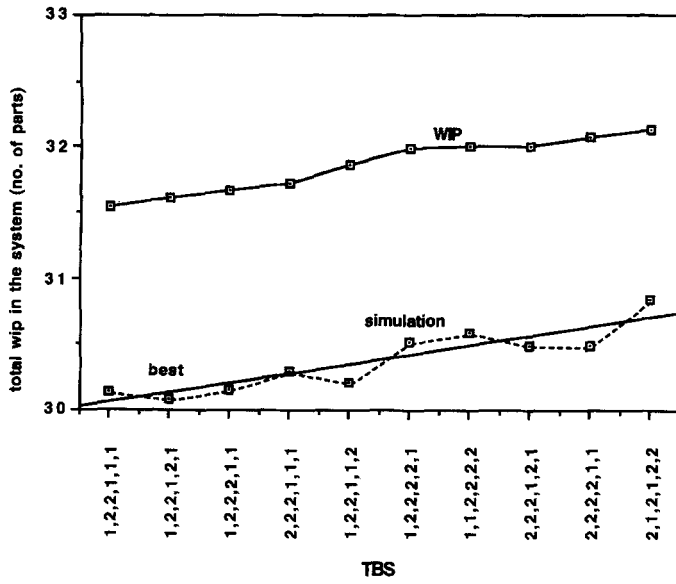


Figure 6. Best 10 solutions for layout 1.

(c) no. of devices=5 (0.695 ≤ device utilization ≤ 0.789)



(d) no. of devices=6 (0.726 ≤ device utilization ≤ 0.824)

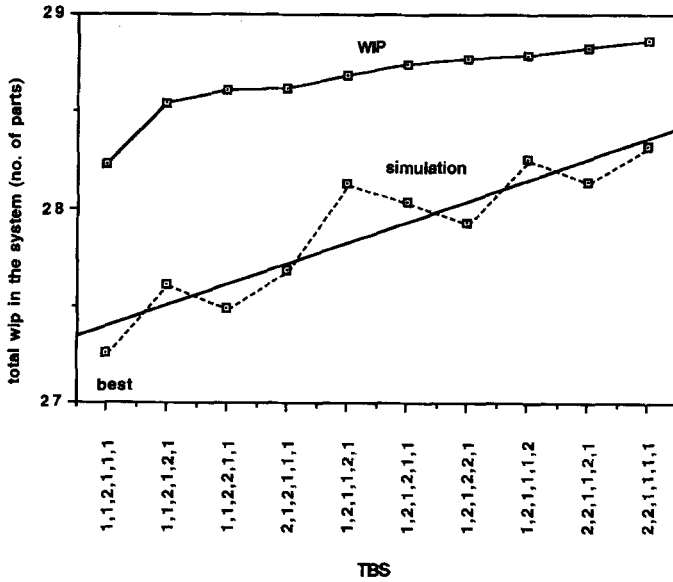


Figure 6. Continued.

presented in figure 6, along with linear regression lines based on the *method of least squares*. We cannot claim that the 10 best solutions are globally optimal, since the WIP model is an approximate model. However, in the next section, we show empirically that (within the solution space we could search) they are optimal or near-optimal solutions.

According to figure 6, the best solution obtained by the WIP model (combined with exhaustive enumeration) is the best or second-best solution according to the simulation results. (Although we incur some errors in absolute values, the WIP model fairly accurately captures the relative changes in total expected WIP.) Also, any of the 10 best solutions may be acceptable for practical purposes. This does not necessarily imply that the objective function is not sensitive to the TBSs, however.

To check the sensitivity of the objective function, we randomly generated five new solutions, which meet the throughput requirement (for the given number of devices) and are in the interval of the best solution  $\pm 2$  parts. Simulation results are shown in figure 7, where the first five TBSs are the top five TBSs shown in figure 6 and the last five TBSs are those generated randomly and sorted by their objective values. The objective function value varies significantly, especially when the number of devices is large. For example, with five devices, the expected WIP in the system with TBS (1,1,2,1,1,1) is about 40% greater than the expected WIP with TBS (1,2,2,1,1,1), although the two TBSs differ by only one part (part type 2).

### 6.3. *The quality of the TBSs determined by the WIP model*

To test the quality of the solutions obtained from the WIP model, we generated all the feasible TBSs within the  $\pm 2$  range of the “optimal” TBS (determined by exhaustive enumeration using the WIP model.) Subsequently, we simulated all such feasible TBSs and ranked them in ascending order of their objective function values. We then compared the rank obtained from simulation with the rank obtained from the WIP model for layouts 1, 2, and 3 (with three devices). The results (including scatter diagrams) indicate that these two rankings are highly correlated (with correlation coefficients of 0.89 or larger).

We also compared the five best TBSs determined by the WIP model (and their objective function values obtained by simulation) against the five best solutions (and the objective function values) obtained by simulation alone (within the  $\pm 2$  range); the results are presented in table 6. The “optimal” solution obtained from the WIP model is not always the best solution obtained via simulation. However, taking into account that the error in the objective function value (see column 9 in table 6) is quite small and that simulation results contain random variation, we conclude that TBSs obtained from the WIP model are reasonably good. Although we were unable to prove convexity, the overall structure of the total expected WIP (shown in figure 5) strongly suggests that one is highly unlikely to find a much better solution outside the  $\pm 2$  range that we investigated.

### 6.4. *Genetic algorithm*

To avoid exhaustive enumeration, we developed a heuristic based on a genetic algorithm (Holland, 1975). One advantage of a GA is that it can handle complex objective functions such as ours. A simple GA is composed of three operations: reproduction, crossover, and mutation; see Goldberg (1989), among others.



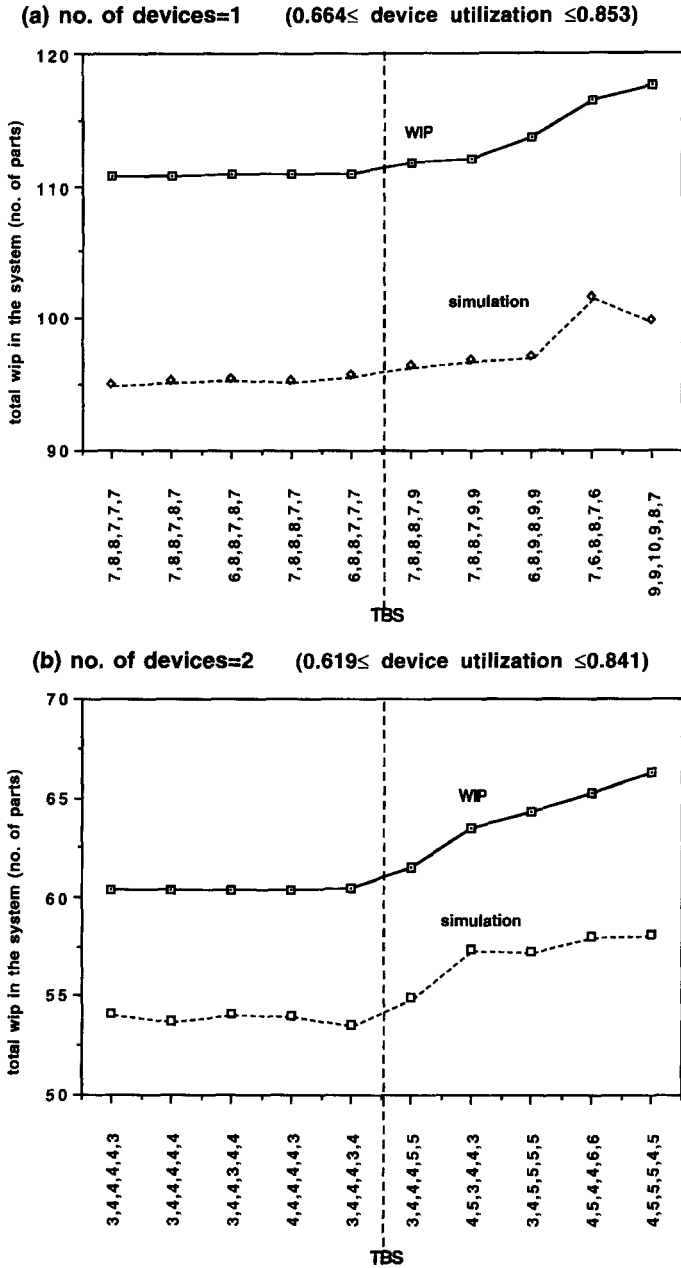


Figure 7. Best five solutions and random solutions for layout 1.

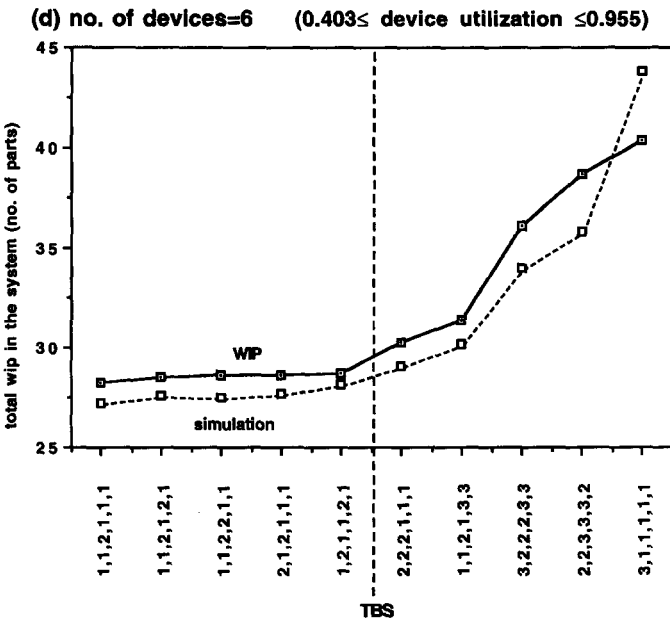
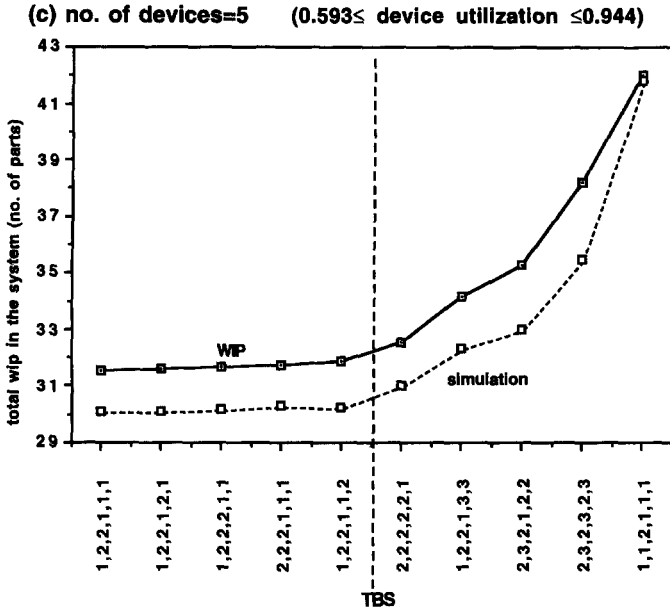


Figure 7. Continued.

Table 6. Solutions obtained by the WIP model and via simulation.

Layout (col. 1)	Best 5 solutions by the WIP model			Obj. value w/ simulation (col. 5)	Best 5 solutions by simulation			% error (col. 5 - col. 7)/(col. 7) (col. 9)
	Rank (col. 2)	TBS (col. 3)	Obj. value (col. 4)		TBS (col. 6)	Obj. value (col. 7)	95% C.I. (col. 8)	
1	1	2 3 3 2 2 2	44.011	39.525	2 3 3 2 2 2	39.525	2.189	0.00
	2	2 3 3 2 3 2	44.157	39.835	2 3 3 2 2 2	39.531	1.900	0.77
	3	2 3 3 3 2 2	44.223	39.531	2 3 3 2 2 3	39.796	1.971	0.67
	4	3 3 3 2 2 2	44.271	40.238	2 3 3 2 3 2	39.835	2.402	1.01
	5	2 3 3 2 2 3	44.412	39.796	2 4 3 2 2 2	39.971	1.329	0.44
	Avg.		39.785		39.732			0.58
2	1	3 2 2 3 2	99.351	87.490	2 3 2 3 2	86.727	4.200	0.88
	2	2 3 2 3 2	100.105	86.727	4 2 2 3 2	87.115	2.743	0.45
	3	2 2 3 3 2	100.411	87.581	3 2 2 3 2	87.490	3.710	0.10
	4	4 2 2 3 2	100.628	87.115	3 2 3 3 2	87.559	3.059	0.51
	5	3 2 3 3 2	100.730	87.559	2 2 3 3 2	87.581	3.567	0.03
	Avg.		87.294		87.294			0.39
3	1	4 4 4 4 3	62.501	56.374	3 3 4 3 4	53.596	4.345	5.18
	2	3 4 4 4 3	62.548	56.085	3 4 3 3 4	53.761	5.693	4.32
	3	3 4 3 3 4	62.646	53.761	2 4 3 3 4	54.195	5.617	0.80
	4	4 4 3 4 3	62.692	57.042	4 4 3 3 4	54.453	5.774	4.75
	5	3 3 4 4 4	62.746	55.183	2 5 3 3 4	54.515	6.194	1.23
	Avg.		55.689		54.104			3.26

Using a string of integers to represent the TBS vector, we constructed a simple GA with “elitist reproduction” and “biased mutation.” *Elitist reproduction* is a technique where the best solutions from the current generation are “automatically” copied over to the next generation. *Mutation* is the random alteration (with a small probability) of the value of a string position. In biased mutation, we alter the value of a string position based on its current value; that is, if a string position’s value lies in the lower half of the feasible region, then its value is altered to a random integer in the upper half of the feasible region and vice versa.

We stop the algorithm either when it reaches the maximum number of generations (1,000 in our tests) or the 15th current best solution is not improved for 50 consecutive generations. For all three layouts (each tested with one, three, and five devices), using a population size of 50, an elitist reproduction rate of 8% (i.e., best 4 solutions out of 50), multiple crossovers (with a crossover probability of 0.70), and a mutation probability of 0.01, the GA obtained the same “optimal” solutions we obtained earlier via exhaustive enumeration. (The reader may refer to Kim 1995 for further details.)

## 7. Conclusions

The WIP model we present here establishes a formal, analytical relationship between the expected WIP level in a manufacturing system and the capacity of the material handling system that supports it. Traditional thinking (or “conventional wisdom”) dictates that, being a “non-value added” operation, investment in material handling should be minimized. Such thinking has led to research and analytical/simulation models where the objective is to determine the minimum fleet size to meet a given throughput requirement. In fact, it is fair to say that even material handling equipment vendors follow the same line of thinking and use simulation to design systems with minimum required number of devices for a given from-to chart.

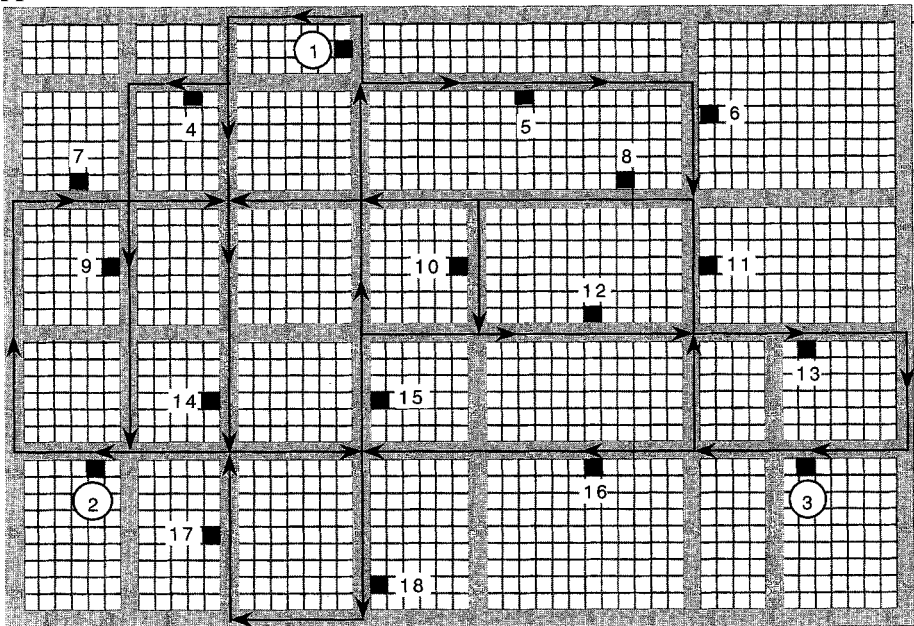
Our results based on the WIP model clearly suggest that such thinking is flawed. For example, if a single device meets the required throughput (with large TBSs), adding, say, two devices to the system is likely to yield a major reduction in total expected WIP (since the TBSs will be reduced). Given all the known manufacturing problems and costs associated with excessive WIP, it is very likely that the additional investment required to add two devices would be well-justified. Of course, adding more devices may be unnecessary; such diminishing returns would be indicated by the WIP model and the GA-based algorithm we present here.

The WIP model and results we present here are also significant for facility layout. When an existing layout is improved through department relocations, it typically reduces the workload on the handling system. (In fact, that is one objective of layout improvement.) However, one often is in no position to reduce the handling workforce or to readily dispose of handling equipment. We show that savings still can be realized by reducing the TBS for all the jobs while maintaining the same handling workforce; that is, savings can be derived from reduced WIP levels rather than a reduced handling workforce. This option may prove critical in justifying layout improvements with a “fixed” handling capacity or workforce.

**Acknowledgment**

This study was partially supported by Bozer’s Presidential Young Investigator (PYI) Award under NSF Grant DDM-8858562 and the Center for Display Technology and Manufacturing (DTM) at the University of Michigan, Ann Arbor. We would also like to thank the referee who helped us improve the presentation of the paper and brought Karmarkar et al. (1985) and Jackman (1991) to our attention.

**Appendix**



Each grid represents one distance unit.

○ I/O station      ■ processing station

Figure A1. Layout 2.

Table A1. Job routes and throughput requirements 1 of layout 2.

Part type	Parts/min	Part route 1.L2												
1	0.027	1	-	8	-	6	-	10	-	11	-	3	-	
2	0.041	1	-	4	-	9	-	7	-	14	-	17	-	2
3	0.041	2	-	14	-	15	-	10	-	12	-	13	-	3
4	0.055	3	-	16	-	18	-	15	-	17	-	2	-	
5	0.027	3	-	11	-	8	-	5	-	6	-	13	-	3

Device speed: 35 (with three devices) to 15 (with seven devices) distance units/min.  
 P/D time = 5.1 (with three devices) to 12 (with seven devices) secs.  
 Processor utilization = 0.75.

Table A2. Job routes and throughput requirements 1 of layout 3.

Part type	Parts/min	Part route 1.L3*														
1	0.02	1	-	4	-	3	-	6	-	2						
2	0.08	1	-	7	-	6	-	5	-	8	-	2				
3	0.06	1	-	5	-	7	-	2								
4	0.04	1	-	8	-	3	-	4	-	2						
5	0.07	1	-	3	-	7	-	4	-	6	-	8	-	5	-	2

Device speed: from 270 (with five devices) to 150 (with nine devices) distance units/min.

P/D time = 8.34 (with five devices) to 15 (with nine devices) secs.

Processor utilization = 0.75.

\*We changed station labels in the original layout so that stations 1 and 2 become I/O stations.

(a) station 4 (7,9,14,16)

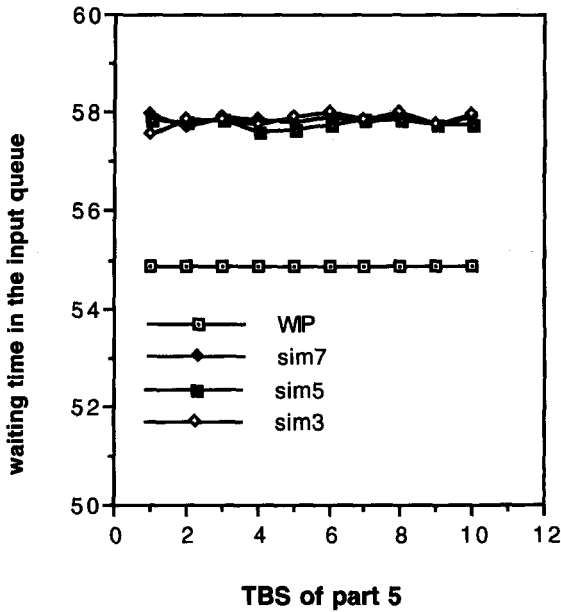
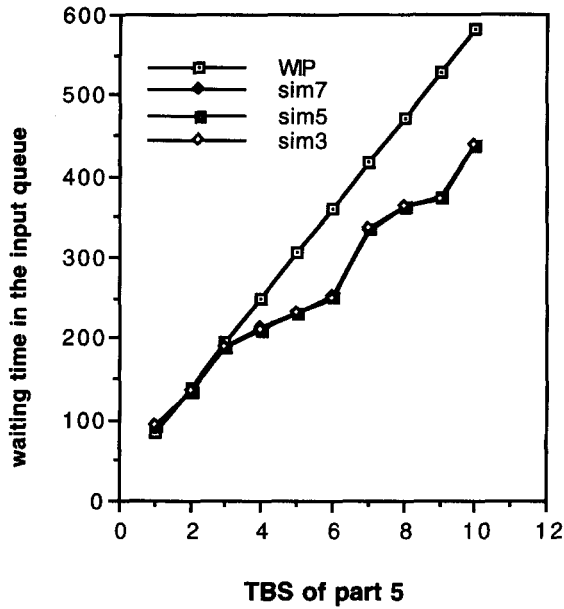


Figure A2. Expected waiting times in input queues of layout 2.

**(b) station 5 (6,8,13)**



**(c) station 10**

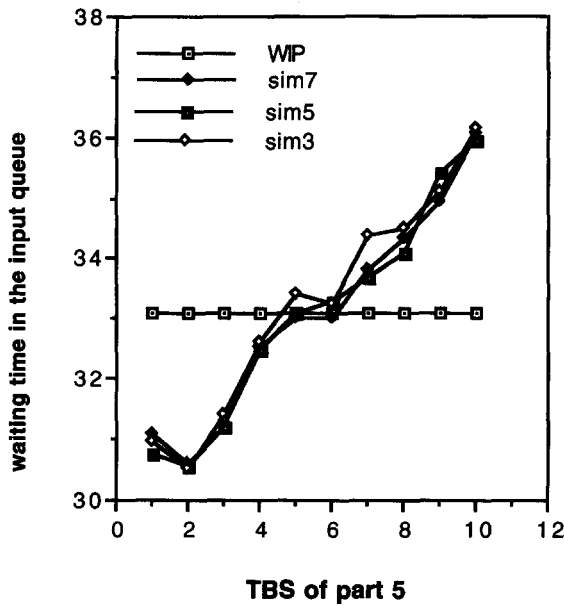
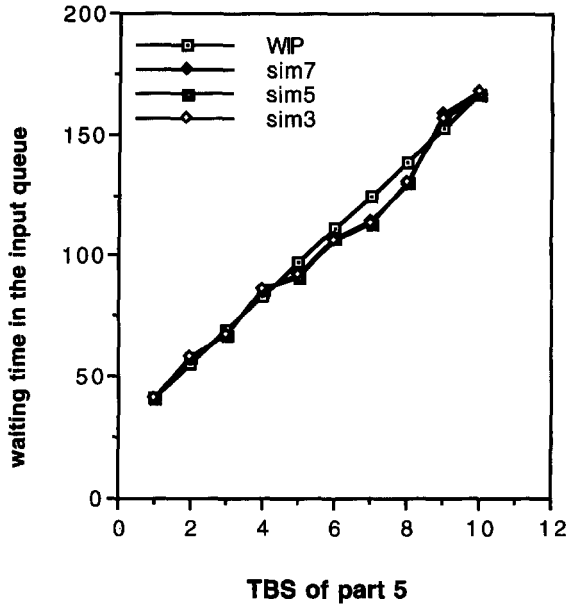


Figure A2. Continued.

**(d) station 11**



**(e) station 12 (15)**

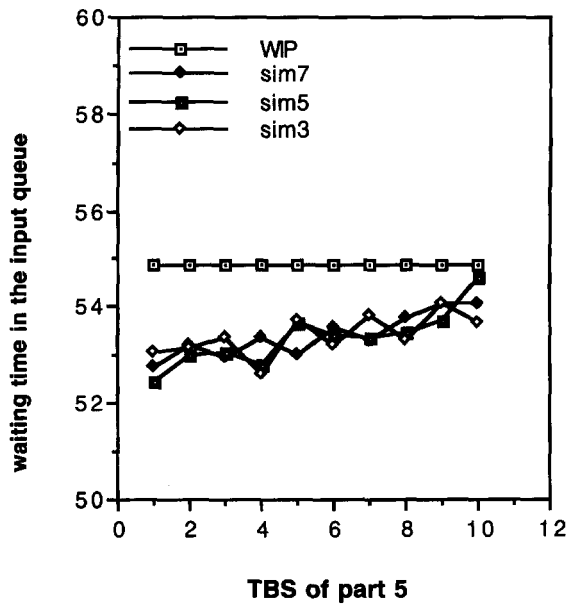


Figure A2. Continued.



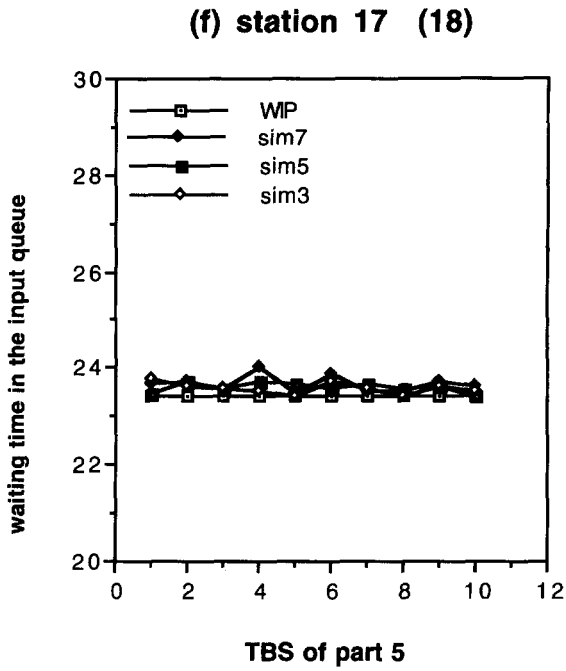


Figure A2. Continued.

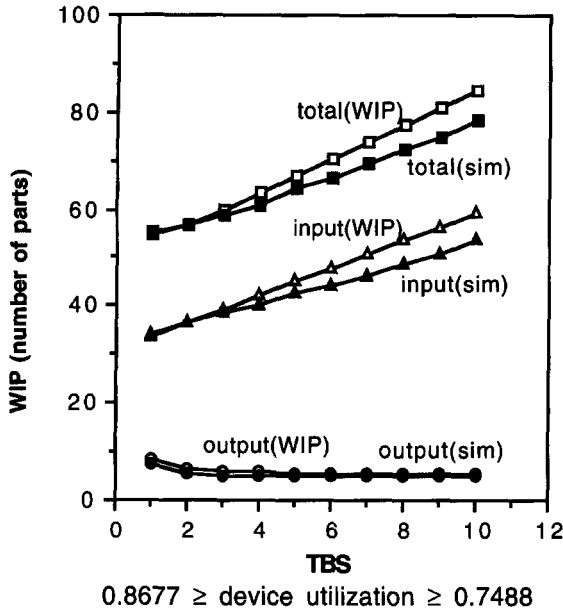
Table A3. Expected waiting times in output queues of layout 2.

	TBS	St. no	1	2	3	4	5	6	7	8	9
Three devices	1	WIP	5.591	5.370	5.962	5.932	5.780	6.044	6.007	5.524	5.791
		Sim	5.648	5.367	5.993	5.990	5.830	6.079	6.051	5.569	5.858
		95% C.I.	0.315	0.296	0.263	0.284	0.272	0.329	0.283	0.309	0.249
		% error	1.03%	0.04%	0.52%	0.96%	0.86%	0.57%	0.72%	0.80%	1.16%
	5	WIP	3.272	2.947	3.681	3.602	3.492	3.781	3.578	3.296	3.382
		Sim	3.249	2.907	3.681	3.584	3.444	3.676	3.563	3.148	3.367
		95% C.I.	0.097	0.085	0.097	0.112	0.150	0.121	0.097	0.091	0.094
		% error	0.73%	1.38%	0.01%	0.49%	1.41%	2.86%	0.42%	4.69%	0.42%
	10	WIP	3.116	2.775	3.530	3.444	3.340	3.633	3.405	3.152	3.212
		Sim	3.150	2.829	3.512	3.493	3.335	3.613	3.415	2.978	3.238
		95% C.I.	0.082	0.064	0.087	0.063	0.122	0.120	0.092	0.081	0.057
		% error	1.07%	1.88%	0.51%	1.41%	0.14%	0.55%	0.30%	5.85%	0.83%
Five devices	1	WIP	6.117	5.749	6.736	6.686	6.433	6.874	6.811	6.006	6.451
		Sim	6.216	5.773	6.794	6.794	6.509	6.937	6.918	6.090	6.558
		95% C.I.	0.397	0.345	0.302	0.358	0.341	0.376	0.336	0.377	0.325
		% error	1.59%	0.42%	0.86%	1.59%	1.17%	0.90%	1.55%	1.38%	1.64%
	5	WIP	3.874	3.332	4.555	4.423	4.240	4.723	4.383	3.913	4.056
		Sim	3.901	3.293	4.540	4.403	4.219	4.572	4.375	3.764	4.073
		95% C.I.	0.068	0.076	0.111	0.076	0.242	0.086	0.074	0.065	0.102
		% error	0.69%	1.17%	0.34%	0.45%	0.50%	3.30%	0.19%	3.96%	0.42%
	10	WIP	3.727	3.160	4.417	4.274	4.101	4.589	4.209	3.788	3.887
		Sim	3.775	3.168	4.444	4.334	4.348	4.412	4.238	3.505	3.919
		95% C.I.	0.094	0.110	0.098	0.102	0.222	0.165	0.130	0.116	0.116
		% error	1.26%	0.26%	0.61%	1.39%	5.67%	4.02%	0.67%	8.08%	0.82%
Seven devices	1	WIP	6.737	6.221	7.603	7.534	7.179	7.796	7.708	6.581	7.204
		Sim	6.894	6.319	7.712	7.747	7.345	7.901	7.845	6.735	7.444
		95% C.I.	0.285	0.295	0.280	0.284	0.273	0.306	0.253	0.314	0.226
		% error	2.28%	1.55%	1.40%	2.76%	2.27%	1.33%	1.74%	2.28%	3.23%
	5	WIP	4.569	3.810	5.522	5.338	5.082	5.757	5.282	4.624	4.824
		Sim	4.618	3.834	5.515	5.290	5.143	5.476	5.275	4.408	4.824
		95% C.I.	0.103	0.069	0.120	0.106	0.175	0.137	0.095	0.079	0.120
		% error	1.05%	0.63%	0.13%	0.91%	1.19%	5.13%	0.13%	4.89%	0.01%
	10	WIP	4.432	3.638	5.398	5.197	4.955	5.639	5.107	4.517	4.656
		Sim	4.471	3.663	5.397	5.252	5.130	5.393	5.123	4.267	4.702
		95% C.I.	0.075	0.097	0.114	0.117	0.363	0.159	0.093	0.213	0.092
		% error	0.87%	0.68%	0.02%	1.04%	3.41%	4.56%	0.32%	5.85%	0.97%

Table A3. Continued.

10	11	12	13	14	15	16	17	18	Avg.	Max
5.819	5.448	5.398	5.521	5.533	5.341	6.215	5.885	5.549		
5.862	5.449	5.374	5.545	5.549	5.363	6.243	5.930	5.594		
0.338	0.320	0.301	0.279	0.318	0.292	0.291	0.241	0.280		
0.74%	0.02%	0.44%	0.43%	0.28%	0.41%	0.45%	0.76%	0.80%	0.61%	1.16%
3.580	3.157	3.060	3.228	3.163	2.950	3.918	3.490	3.167		
3.583	3.027	3.051	3.066	3.096	2.877	3.877	3.434	3.165		
0.111	0.081	0.141	0.116	0.086	0.081	0.102	0.081	0.078		
0.09%	4.30%	0.28%	5.29%	2.17%	2.53%	1.06%	1.62%	0.06%	1.66%	5.29%
3.435	3.005	2.900	3.075	2.999	2.783	3.764	3.322	3.001		
3.467	2.956	2.900	3.041	3.009	2.780	3.737	3.330	3.026		
0.091	0.145	0.098	0.133	0.072	0.069	0.092	0.091	0.098		
0.95%	1.65%	0.00%	1.12%	0.33%	0.10%	0.72%	0.25%	0.83%	1.03%	5.85%
6.497	5.879	5.796	6.001	6.021	5.701	7.158	6.608	6.048		
6.587	5.953	5.804	6.060	6.073	5.726	7.237	6.696	6.127		
0.375	0.356	0.316	0.333	0.348	0.363	0.314	0.339	0.322		
1.37%	1.24%	0.13%	0.97%	0.85%	0.44%	1.09%	1.32%	1.29%	1.10%	1.64%
4.386	3.682	3.520	3.800	3.692	3.337	4.950	4.237	3.699		
4.374	3.558	3.526	3.678	3.647	3.279	4.890	4.187	3.713		
0.105	0.075	0.076	0.112	0.082	0.073	0.068	0.075	0.085		
0.27%	3.50%	0.17%	3.33%	1.23%	1.77%	1.22%	1.19%	0.36%	1.34%	3.96%
4.259	3.542	3.368	3.659	3.533	3.172	4.808	4.072	3.537		
4.268	3.449	3.412	3.610	3.549	3.180	4.799	4.058	3.565		
0.123	0.126	0.115	0.172	0.067	0.085	0.103	0.083	0.085		
0.20%	2.70%	1.30%	1.36%	0.45%	0.26%	0.19%	0.34%	0.78%	1.69%	8.08%
7.269	6.403	6.287	6.575	6.602	6.154	8.194	7.424	6.640		
7.458	6.523	6.378	6.693	6.747	6.278	8.292	7.566	6.775		
0.321	0.314	0.298	0.304	0.324	0.272	0.245	0.254	0.271		
2.53%	1.84%	1.42%	1.76%	2.15%	1.96%	1.18%	1.88%	1.99%	1.98%	3.23%
5.287	4.301	4.073	4.465	4.315	3.817	6.075	5.078	4.324		
5.299	4.074	4.035	4.312	4.272	3.761	6.046	5.023	4.325		
0.107	0.103	0.112	0.122	0.079	0.086	0.102	0.074	0.109		
0.24%	5.55%	0.94%	3.56%	1.01%	1.48%	0.48%	1.08%	0.03%	1.58%	5.55%
5.176	4.173	3.929	4.337	4.160	3.656	5.945	4.915	4.166		
5.175	3.934	3.982	4.217	4.187	3.645	5.916	4.905	4.201		
0.077	0.120	0.116	0.133	0.088	0.076	0.087	0.087	0.078		
0.03%	6.08%	1.33%	2.83%	0.63%	0.29%	0.50%	0.21%	0.83%	1.69%	6.08%

**(a) part 1 (0.027 parts/min)**



**(b) part 2 (0.041 parts/min)**

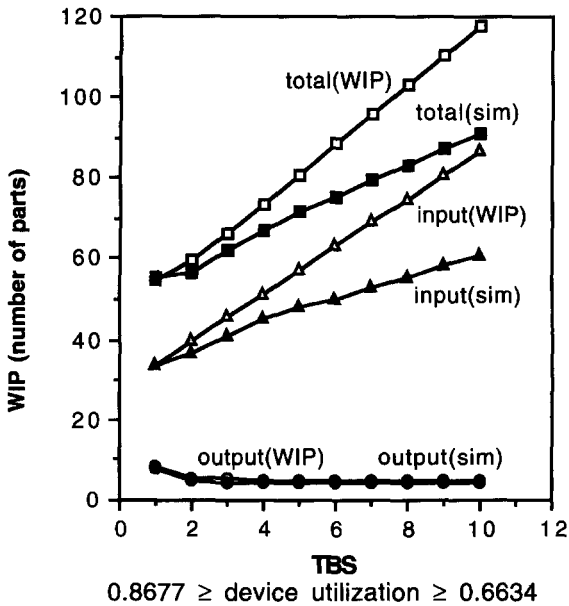
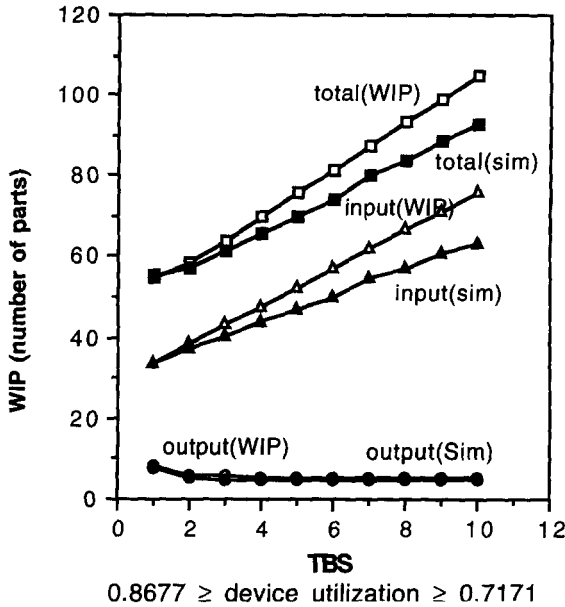
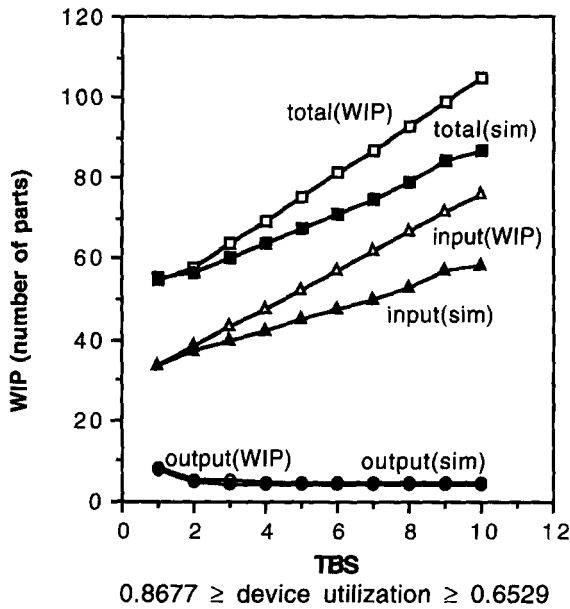


Figure A3. Expected total WIP in the system for seven devices of layout 2.

**(c) part 3 (0.041 parts/min)**

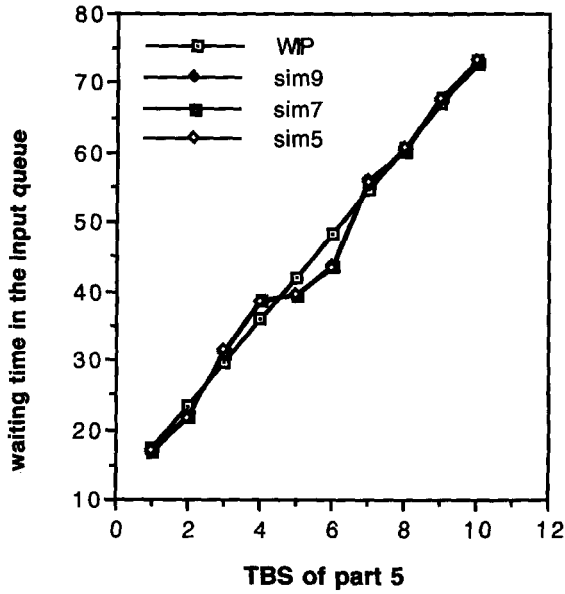


**(d) part 4 (0.055 parts/min)**



**(e) part 5 (0.027 parts/min) : same trend as in graph (a)**

(a) station 3



(b) station 4 (7)

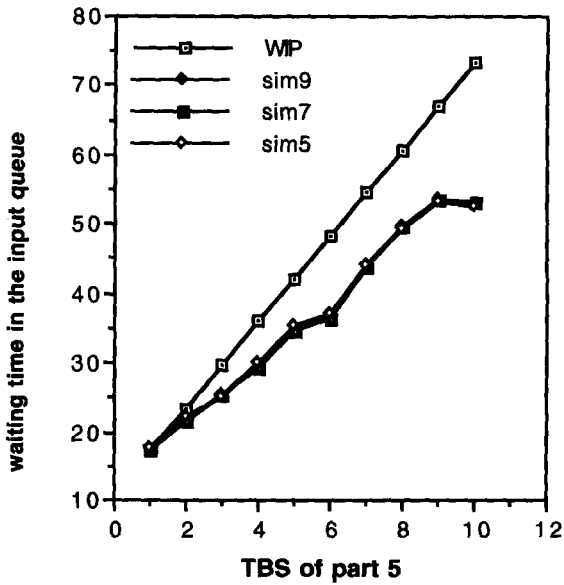
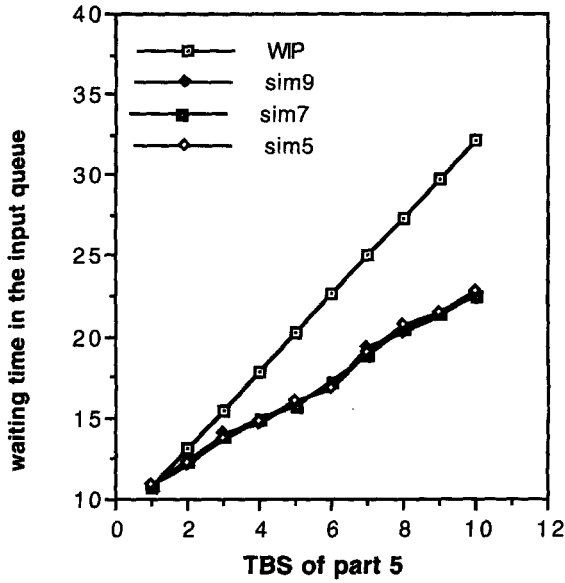


Figure A4. Expected waiting times in input queues of layout 3.

**(c) station 5 (8)**



**(d) station 6**

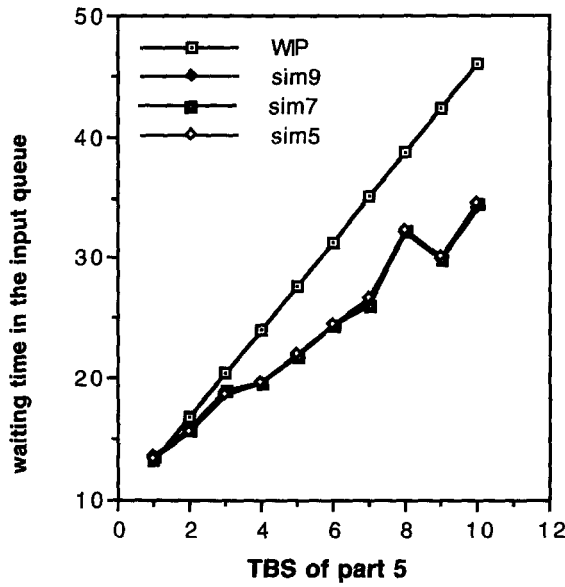


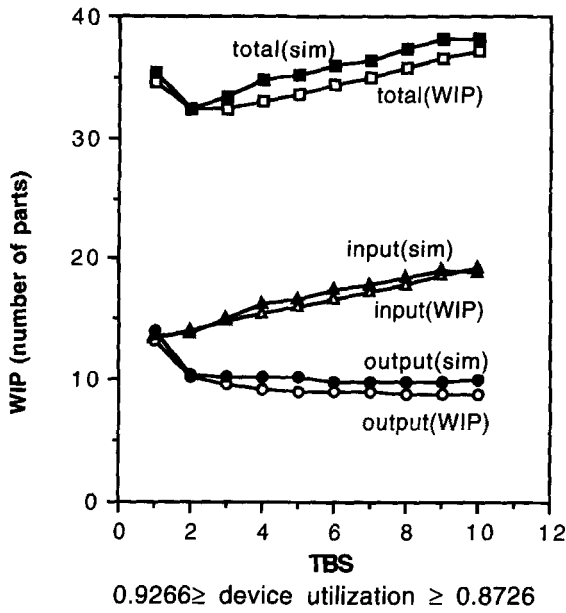
Figure A4. Continued.

Table A4. Expected waiting times in output queues of layout 3.

TBS	St. no	1	2	3	4	5	6	7	8	Avg.	Max
Five devices	WIP	8.341	0.000	9.400	9.412	8.542	9.129	8.906	9.610		
	Sim	8.562	0.000	9.644	9.692	8.819	9.414	9.085	9.947		
	95% C.I.	0.636	0.000	0.637	0.666	0.630	0.643	0.632	0.661		
	% error	2.58%	0.00%	2.53%	2.89%	3.14%	3.02%	1.96%	3.38%	2.79%	3.38%
	WIP	1.907	0.000	3.006	3.040	2.144	2.704	2.456	3.158		
	Sim	1.949	0.000	2.843	2.854	2.021	2.554	2.341	2.969		
10	95% C.I.	0.033	0.000	0.056	0.066	0.034	0.052	0.039	0.052		
	% error	2.14%	0.00%	5.75%	6.52%	6.12%	5.89%	4.91%	6.37%	5.39%	6.52%
	WIP	1.789	0.000	2.895	2.932	2.032	2.586	2.333	3.035		
	Sim	1.869	0.000	2.750	2.780	1.958	2.463	2.204	2.852		
	95% C.I.	0.038	0.000	0.053	0.065	0.039	0.049	0.039	0.041		
	% error	4.31%	0.00%	5.26%	5.47%	3.80%	5.03%	5.84%	6.44%	5.16%	6.44%
Seven devices	WIP	8.563	0.000	10.136	10.152	8.935	9.756	9.444	10.430		
	Sim	8.959	0.000	10.444	10.521	9.285	10.124	9.711	10.843		
	95% C.I.	0.722	0.000	0.727	0.759	0.689	0.725	0.691	0.710		
	% error	3.42%	0.00%	2.95%	3.51%	3.77%	3.63%	2.75%	3.81%	3.40%	3.81%
	WIP	2.380	0.000	3.918	3.965	2.711	3.495	3.147	4.130		
	Sim	2.414	0.000	3.679	3.695	2.526	3.235	2.931	3.827		
10	95% C.I.	0.029	0.000	0.000	0.000	0.021	0.048	0.032	0.053		
	% error	1.41%	0.00%	6.50%	7.30%	7.35%	8.03%	7.37%	7.92%	6.56%	8.03%
	WIP	2.275	0.000	3.823	3.876	2.616	3.392	3.037	4.020		
	Sim	2.350	0.000	3.509	3.536	2.474	3.144	2.831	3.691		
	95% C.I.	0.033	0.000	0.081	0.084	0.027	0.043	0.043	0.038		
	% error	3.20%	0.00%	8.96%	9.63%	5.71%	7.88%	7.29%	8.93%	7.37%	9.63%
Nine devices	WIP	9.052	0.000	10.959	10.980	9.415	10.471	10.070	11.338		
	Sim	9.352	0.000	11.261	11.299	9.732	10.799	10.326	11.723		
	95% C.I.	0.615	0.000	0.632	0.620	0.594	0.637	0.601	0.608		
	% error	3.21%	0.00%	2.68%	2.82%	3.26%	3.04%	2.48%	3.29%	2.97%	3.29%
	WIP	2.891	0.000	4.869	4.929	3.317	4.324	3.877	5.142		
	Sim	2.893	0.000	4.408	4.474	3.069	3.942	3.582	4.735		
10	95% C.I.	0.027	0.000	0.107	0.092	0.038	0.061	0.034	0.060		
	% error	0.08%	0.00%	10.46%	10.17%	8.08%	9.71%	8.24%	8.58%	7.90%	10.46%
	WIP	2.797	0.000	4.788	4.856	3.235	4.233	3.777	5.041		
	Sim	2.872	0.000	4.245	4.269	2.953	3.800	3.424	4.474		
	95% C.I.	0.049	0.000	0.074	0.072	0.031	0.045	0.049	0.048		
	% error	2.61%	0.00%	12.80%	13.75%	9.55%	11.40%	10.31%	12.68%	10.44%	13.75%



**(a) part 1 (0.02 parts/min)**



**(b) part 2 (0.08 parts/min)**

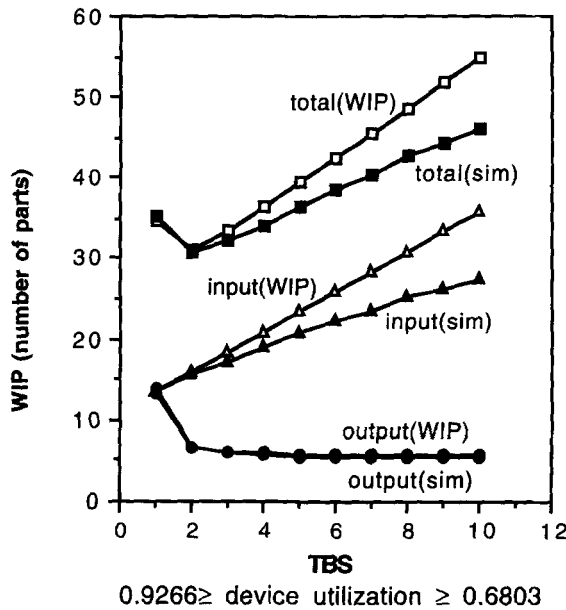
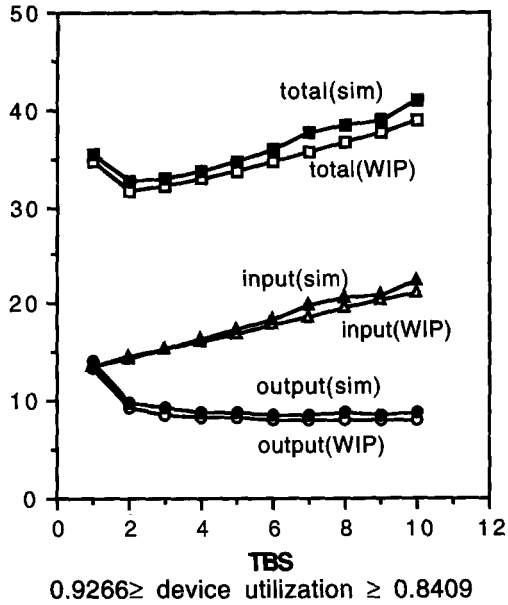
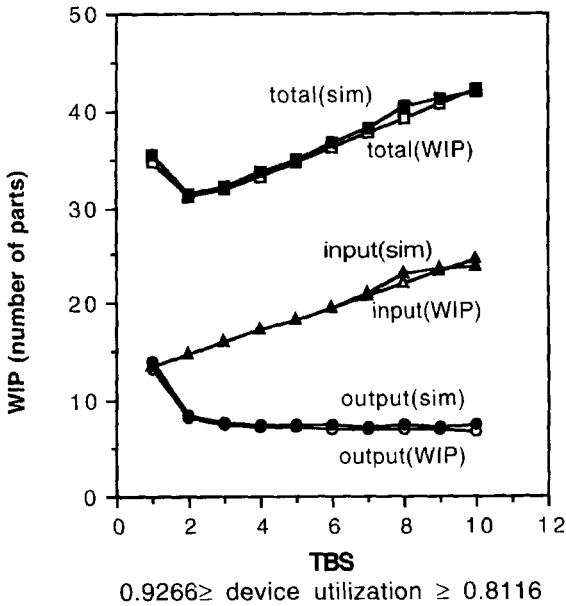


Figure A5. Expected total WIP in the system for nine devices of layout 3.

**(c) part 3 (0.06 parts/min)**



**(d) part 4 (0.04 parts/min)**



**(e) part 5 (0.07 parts/min) : same trend as in graph (b)**

## References

- Bertrand, J.W.M., "Multiproduct Optimal Batch Size with In-Process Inventories and Multi Work Centers," *IIE Transactions*, Vol. 17, No. 2, pp. 157–163 (1985).
- Bozer, Y.A., Cho, M.S., and Srinivasan, M.M., "Expected Waiting Times in Single-Device Trip-Based Material Handling Systems," *European Journal of Operational Research*, Vol. 75, pp. 200–216 (1994).
- Cho, M.S., "Design and Performance Analysis of Trip-Based Material Handling Systems in Manufacturing," Ph.D. dissertation, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI (1990).
- Chow, W.M., "Design for Line Flexibility," *IIE Transactions*, Vol. 18, No. 1, pp. 95–108 (1986a).
- Chow, W.M., "An Analysis of Automated Storage and Retrieval Systems in Manufacturing Assembly Lines," *IIE Transactions*, Vol. 18, No. 2, pp. 204–214 (1986b).
- Egbelu, P.J., "The Use of Non-Simulation Approaches in Estimating Vehicle Requirements in an Automated Guided Vehicle Based Transport System," *Material Flow*, Vol. 4, Nos. 1–2, pp. 17–32 (1987).
- Egbelu, P.J., "Economic Design of Unit Load-Based FMSs Employing AGVs for Transport," *International Journal of Production Research*, Vol. 31, No. 12, pp. 2753–2775 (1993a).
- Egbelu, P.J., "Concurrent Specification of Unit Load Sizes and Automated Guided Vehicle Fleet Size in Manufacturing System," *International Journal of Production Economics*, Vol. 29, pp. 49–64 (1993b).
- Goldberg, D.E., *Genetic Algorithms*, Addison-Wesley Publishing Co., New York, NY (1989).
- Grasso, E.T. and Tanchoco, J.M.A., "Unit Load and Material Handling Considerations in Material Requirements Systems," *Material Flow*, Vol. 1, pp. 79–87 (1983).
- Holland, J.H., *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, MI (1975).
- Jackman, J., "Unit Load Size Determination Based on Economic Move Quantity," *Journal of the Operational Research Society*, Vol. 42, No. 8, pp. 703–711 (1991).
- Karmarkar, U.S., Kekre, S., Kekre, S., and Freeman, S., "A Novel Analytical Lot-Sizing Model (Q-LOTS)," *Interfaces*, Vol. 15, No. 2, pp. 1–9 (1985).
- Kim, J., "Transfer Batch Sizing in Trip-Based Material Handling Systems," Ph.D. dissertation, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI (1995).
- Kuehn, P.J., "Approximate Analysis of General Queueing Networks by Decomposition," *IEEE Transactions on Communications*, Vol. COM-27, No. 1, pp. 113–126 (1979).
- Mahadevan, B. and Narendran, T.T., "Determination of Unit Load Sizes in an AGV-Based Material Handling System for an FMS," *International Journal of Production Research*, Vol. 30, No. 4, pp. 909–922 (1992).
- Nozaki, S. and Ross, S., "Approximations in Finite Capacity Multiserver Queues with Poisson Arrivals," *Journal of Applied Probability*, Vol. 15, pp. 826–834 (1978).
- Ross, S.M., *Introduction to Probability Theory*, Academic Press Inc., Orlando, FL (1985).
- Solberg, J.J., "Capacity Planning with a Stochastic Workflow Model," *AIEE Transactions*, Vol. 13, No. 2, pp. 116–122 (1981).
- Solot, P. and Bastos, J.M., "MULTIQ: A Queueing Model for FMSs with Several Pallet Types," *Journal of the Operational Research Society*, Vol. 39, No. 9, pp. 811–821 (1988).
- Srinivasan, M.M., Bozer, Y.A., and Cho, M.S., "Trip-Based Material Handling Systems: Throughput Capacity Analysis," *IIE Transactions*, Vol. 26, No. 1, pp. 70–89 (1994).
- Yao, D.D. and Buzacott, J.A., "Modeling the Performance of Flexible Manufacturing Systems," *International Journal of Production Research*, Vol. 23, No. 5, pp. 945–959 (1985).
- Yao, D.D. and Buzacott, J.A., "Models of Flexible Manufacturing Systems with Limited Local Buffers," *International Journal of Production Research*, Vol. 24, No. 1, pp. 107–118 (1986).
- Yao, D.D. and Buzacott, J.A., "Modeling a Class of Flexible Manufacturing Systems with Reversible Routing," *Operations Research*, Vol. 35, No. 1, pp. 87–93 (1987).