

# Expected move request waiting times in single-device, polling-based material handling systems

YAVUZ A. BOZER<sup>1</sup> and JUNG H. PARK<sup>2</sup>

<sup>1</sup>*Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI 48109-2117, USA*  
E-mail: yabozer@umich.edu

<sup>2</sup>*ePocrates, Inc., 981 Industrial Road, Suite F, San Carlos, CA 94070, USA*  
E-mail: jpark@epocrates.com

Received July 1997 and accepted September 1999

We present an analytical approach to approximate the expected waiting times of move requests (customers) served by a single handling device (server) that operates according to the First-Encountered-First-Served (FEFS) rule, which is a common rule employed for polling-based material handling systems. Under the FEFS rule, the device inspects each station according to a prespecified polling sequence and serves the first move request it encounters. Polling resumes as soon as the device completes serving a move request. The expected waiting times are important for estimating the expected Work-In-Process (WIP) levels at individual stations and to gauge the overall performance of the system. Moreover, the polling sequence itself can affect the expected waiting times. If the device meets the throughput requirement under more than one polling sequence, the results we present can also be used to evaluate alternative polling sequences. In fact, using the analytical results and a numerical example, we show that alternative polling sequences, even if they impose the same “workload factor” on the device, can lead to significantly different expected WIP levels.

## 1. Introduction

In this paper we are concerned with developing an analytical approach to approximate the expected waiting time experienced by move requests (“customers”) served by a single-device handling system. The move requests arrive randomly, one at a time, at one of the “stations” served by the device (“server”), and the device is assumed to serve (i.e., move) only one move request (i.e., one load) at a time.

There are a number of single-device handling systems used in industry. They include overhead cranes (such as bridge cranes), gantry robots, microload Automated Storage/Retrieval (AS/R) systems, (tandem) Automated Guided Vehicle (AGV) systems, freight elevators, and vertical reciprocating conveyors, among others. In each of the above systems, there is a single-device that serves a set of stations and the device moves one load at a time. (For further information on the handling systems above, the reader may refer to Tompkins *et al.* (1996)).

There are two fundamental approaches to device control in a material handling system. Under the first approach, known as centralized control, a central computer keeps track of the state of the system and it decides when each move request will be served (and by which device if there are multiple-devices); i.e. the device(s) are “dispatched” by the central computer.

Under decentralized control, on the other hand, a central computer is not required; each device is preprogrammed to follow a certain set of instructions based on the state of the station they visit. A common method to implement decentralized control is based on the well-known polling concept in queueing, where the device “polls” each station until it locates a move request. This method is also known as the First-Encountered-First-Served (FEFS) rule, which has been used in material handling (Bartholdi and Platzman, 1989) and data retrieval in computer systems (Stone and Fuller, 1973). Following service, i.e., when the load is deposited at, say, station  $j$ , the device resumes polling starting at station  $j$ .

There are certain cost and performance tradeoffs between centralized and decentralized control. Although examining such tradeoffs is well beyond the scope of our paper, centralized control, generally speaking, offers a potential gain in system throughput but with an increase in cost, while decentralized control offers simplicity and reduced cost but a potential loss in system throughput. (Note that, under centralized control, the computer must exchange information with each device, which often results in radio dispatched devices). The device in our study is assumed to operate under the FEFS control rule.

In the next section we introduce basic definitions and review related past work. In Section 3, we present the

“basic throughput model” that was developed in Bozer and Srinivasan (1991). In Section 4, we state all our assumptions (including those that are unique to the expected waiting time model) and, building on the results given in Bozer and Srinivasan (1991), we derive the expected waiting times. Last, in Section 5, we present numerical results and state our conclusions.

## 2. Basic definitions and literature review

Consider the FEFS rule and  $M$  stations that are polled by the device in a cyclic fashion following the sequence  $1, 2, \dots, M$ . That is, if there are no move requests at station  $i$  when it is polled by the device, the device travels (empty) to station  $i + 1$  and polls it. Otherwise, the load at the head of the queue at station  $i$  is served; i.e., the device picks up the appropriate load at station  $i$ , travels (loaded) to station  $j$ , deposits the load and resumes polling at station  $j$ . Since it can move only one load at a time, no polling is performed as the device travels loaded from station  $i$  to  $j$ .

Note that, if  $i = M$  and there is no move request at station  $M$ , the next station to be polled is station 1 (i.e.,  $M + 1 = 1$ ). Also note that each station is polled exactly once as the polling sequence is repeated. Such a (cyclic) polling sequence – where each station is polled exactly once – is suitable only in those cases where the stations are located around a closed-loop path or similar configuration, which is the focus of this study. (We will later discuss the polling sequence itself and its impact).

A single device operating under the FEFS rule is a special type of nondeterministic polling system. Using the notation given by Srinivasan (1991), in a nondeterministic polling system, if there are no requests at station  $i$  when it is polled, the “server” next polls station  $j$  with probability  $e_{ij}$ . Letting  $p_{ij}$  denote the probability that a load picked up at station  $i$  is destined to station  $j$ , the FEFS policy we study here describes a polling system where  $e_{i,i+1} = 1$ , and  $p_{ij}$  takes arbitrary values between zero and one. (It is assumed that  $p_{ii} = 0$  since a station does not send loads to itself).

For user-defined  $p_{ij}$  and  $e_{ij}$  values, Srinivasan (1991) derives cycle times and stability conditions for single-server, nondeterministic polling systems under various service disciplines. (The cycle time,  $C_i$ , denotes the expected time between two successive visits by the server to station  $i$ ). He also develops conservation laws and obtains expected waiting times. However, for the nonexhaustive service discipline (i.e., exactly one customer is served if the queue at station  $i$  is nonempty at the polling instant), the expected waiting time is derived only for systems with two stations. (The material handling system we are concerned with falls under the nonexhaustive service discipline). Srinivasan also treats the special case where  $e_{ij} = p_{ij}$ , which does not apply to our system.

As a possible device control rule in material handling, the FEFS rule was first proposed and analyzed by Bartholdi and Platzman (1989) for a single AGV traveling around a simple, closed-loop path. The authors show that the FEFS rule is particularly effective in such configurations. (Of course, the polling sequence  $1, 2, \dots, M$  corresponds to the physical sequence of the stations around the loop).

The FEFS rule was also studied by Bozer and Srinivasan (1991) in the context of tandem AGV systems, where the stations are first divided into nonoverlapping zones and each zone is served by a single AGV. The authors derive the cycle times and the stability condition for a single AGV serving a set of stations under FEFS. (In Bozer and Srinivasan (1991), the stations are not required to be located around a loop). In deriving the expected waiting times, we will build on the results obtained in Bozer and Srinivasan (1991).

Previous work in expected waiting times is also reported for handling systems running under centralized control. Assuming First-Come-First-Served (FCFS) device dispatching (i.e., the oldest move request in the system always gets served first regardless of the location of the empty device), Chow (1986) models a microload AS/R system – which is a single-device system – as an M/G/1/FCFS queue. With such an approach, the expected waiting time is averaged across all the stations; i.e., expected waiting times for individual stations are not captured. Bozer and Kim (1996) obtain the expected waiting times (at individual stations) for single- and multiple-device systems operating under the FCFS rule. Last, Bozer *et al.* (1994) derive the expected waiting times for a single device operating under the MOD FCFS rule, which tends to reduce empty device travel compared to FCFS. (With the MOD FCFS rule, when the device delivers a load at station  $i$ , it first inspects station  $i$ ; if a load is waiting at station  $i$ , it is served regardless of its status; otherwise, the oldest move request in the system is served).

## 3. The basic throughput model and system stability

The results presented in this section are taken from Bozer and Srinivasan (1991); we made minor changes to simplify or generalize them. Recall that the polling sequence is given by  $1, 2, \dots, M$ , where  $M$  denotes the number of stations. Let  $\lambda_i$  denote the rate at which loads arrive at the move request queue of station  $i$  (for brevity we will refer to it as queue  $i$ ). Likewise, let  $\Lambda_i$  denote the rate at which the device delivers loads to station  $i$ . In steady state,  $\lambda_i$  may or may not be equal to  $\Lambda_i$  for each  $i$ . However, flow is assumed to be globally conserved; i.e.,  $\sum_i \Lambda_i = \sum_i \lambda_i$ , provided that the device is able to meet the demand placed on it.

Suppose the number of loaded trips per unit time that the device must perform from queue  $i$  to station  $j$  is

denoted by  $f_{ij}$ . Note that  $\lambda_i = \sum_{j=1}^M f_{ij}$  and  $\Lambda_i = \sum_{j=1}^M f_{ji}$ , by definition. Let  $\tau_{ij}$  denote the mean travel time for the device to perform a loaded trip from queue  $i$  to station  $j$ ; this consists of the time it takes to pick-up the load at queue  $i$ , transport it to station  $j$ , deposit the load at station  $j$ , and inspect queue  $j$  for move requests. If queue  $j$  is empty, the device travels empty from station  $j$  to  $j+1$ . The time that it takes to perform this empty trip, which includes the time to inspect queue  $j+1$  is a random variable with mean  $\sigma_{j,j+1}$ . Lastly, let  $\sigma_{j,i}$  denote the expected empty device travel time from station  $j$  to station  $i$  following the polling sequence, that is:

$$\sigma_{j,i} = \begin{cases} \sum_{k=j}^{i-1} \sigma_{k,k+1}, & \text{if } j < i \\ \sum_{k=j}^M \sigma_{k,k+1} + \sum_{k=1}^{i-1} \sigma_{k,k+1}, & \text{if } j > i, \end{cases} \quad (1)$$

where  $M+1=1$ .

The throughput requirement (or workload) imposed on the device is determined by computing the workload factor  $\omega$ . Let  $\alpha_f$  denote the expected fraction of time the device is traveling loaded. Since the flow data and mean travel times are given,  $\alpha_f$  is given by Bozer and Srinivasan (1991) as:

$$\alpha_f = \sum_{i=1}^M \sum_{j=1}^M f_{ij} \tau_{ij}. \quad (2)$$

Note that  $\alpha_f$  is independent of the device dispatching rule. If  $\alpha_f \geq 1$ , the device will not be able to meet the required throughput (regardless of the dispatching rule used).

Let  $C_i$  denote the expected time between two successive inspections by the device at station  $i$ . The term  $C_i$  is derived in Bozer and Srinivasan (1991) as

$$C_i = \frac{\chi}{1 - \alpha_f - \phi_i + \lambda_i \chi}, \quad (3)$$

where  $\chi = \sum_{i=1}^M \sigma_{i,i+1}$  and  $\phi_i = \sum_{j \neq i} (\Lambda_j - \lambda_j) \sigma_{j,i}$ .

Let  $q_i$  denote the probability that queue  $i$  is empty at the polling instant. In Bozer and Srinivasan (1991) it is shown that:

$$q_i = 1 - \lambda_i C_i. \quad (4)$$

Note that  $q_i$  is non-negative, which implies that we must have  $\lambda_i C_i < 1$ . Using Equation (3), in Bozer and Srinivasan (1991) it is shown that under the FEFS dispatching rule, the device will meet the required throughput (i.e., the system will be stable) if the workload  $\omega$  meets the following condition:

$$\omega = \alpha_f + \phi < 1, \quad (5)$$

where

$$\phi = \max_i \phi_i = \max_i \left[ \sum_{j \neq i} (\Lambda_j - \lambda_j) \sigma_{j,i} \right]. \quad (6)$$

Note that, if flow is conserved at a station, i.e.,  $\lambda_j = \Lambda_j$ , its contribution to  $\phi_i$  is zero. In Bozer and Srinivasan

(1991),  $\phi$  is interpreted as ‘‘mandatory’’ empty device travel which results from empty device travel between stations with unbalanced flow. Also, the larger the  $\omega$  value in Equation (5), the ‘‘harder’’ the device must work to satisfy throughput. However, as we show later in the paper, substantially different expected waiting times may be incurred under the same  $\omega$  value.

#### 4. The expected waiting times

We first present the mathematical assumptions underlying the model:

- (1) The move requests (i.e., the loads) arrive at the move request queue of each station randomly, one at a time, following an independent Poisson process.
- (2) The device serves (i.e., moves) only one load at a time, and it operates under the First-Encountered-First-Served (FEFS) rule, following the cyclic polling sequence  $1, 2, \dots, M$ . (When traveling loaded, the device does not poll the stations.) Polling resumes always at the destination station of the last load delivered by the device.
- (3) A load picked up at station  $i$  is destined to station  $j$  with probability  $p_{ij}$  independent of the state of the system. A station does not send loads to itself (i.e.,  $p_{ii} = 0$ ).
- (4) Flow is not necessarily conserved at each station. However, flow is globally conserved; i.e.,  $\sum_i \Lambda_i = \sum_i \lambda_i$ .
- (5) The system is stable; that is,  $\alpha_f + \phi < 1$ .  
We next present an additional assumption that may be mathematically inconsistent but necessary to make the analysis tractable:
- (6) Loads are delivered at the stations by the device at random points in time.

Assumptions (2), (3), and (4) govern the movement of loads in the system, while assumption (5) can be easily checked using the results given in Bozer and Srinivasan (1991), which were repeated here. The validity of assumption (1) depends on how actual move requests arrive and it too can be checked if there are data on request arrivals. (However, as we discuss later in Section 5, in a manufacturing setting, assumption (1) may be a source of error for some of the stations).

Assumption (6), on the other hand, is needed for analytical tractability. Its effect (as a source of error) is difficult to assess but, as discussed in Srinivasan *et al.* (1994), it generally works well when the  $p_{ij}$ -matrix induces some randomness and there is a sufficient number of stations in the system. In Section 5 we will present simulation results to assess the impact of assumptions (1) and (6).

We now derive the approximate expected waiting time at station  $i$  using the ‘‘tagged customer’’ approach.

Consider a (tagged) load that arrives at queue  $i$ . Since loads are assumed to arrive in a Poisson fashion, and such arrivals see time averages (i.e., the well-known PASTA property applies (Wolff, 1982)), the tagged load finds the steady-state distribution of loads at queue  $i$ . If  $P_i(n)$  denotes the probability that  $n$  loads are present at queue  $i$ , and  $W_i(n)$  denotes the conditional expected waiting time for the tagged load given that  $n$  loads are present at queue  $i$  when the tagged load arrives, then the expected waiting time at queue  $i$  for the tagged load is given by

$$W_i = \sum_{n=0}^{\infty} P_i(n) W_i(n). \quad (7)$$

Equation (7) is divided into two cases,  $n = 0$  and  $n > 0$ , as follows:

$$W_i = P_i(0)W_i(0) + \sum_{n=1}^{\infty} P_i(n)W_i(n). \quad (8)$$

Consider first the case where  $n = 0$ . Let  $\alpha_{fj}$  denote the probability that the device is serving station  $j$  (i.e., the device is performing a *loaded* trip from station  $j$ ) at a random point in time. Then,

$$\alpha_{fj} = \sum_k f_{jk} \tau_{jk}. \quad (9)$$

Likewise, let  $\alpha_{ej}$  denote the probability that the device is performing an *empty* trip from station  $j$  to  $j+1$  at a random point in time. Using the results shown in Bozer and Srinivasan (1991), it is straightforward to show that

$$\alpha_{ej} = (q_j \sigma_{j,j+1}) / C_j, \quad (10)$$

where  $q_j$  (see Equation (4)) is the probability that queue  $j$  is empty at the polling instant, and  $C_j$  (see Equation (3)) is the expected cycle time between two successive inspections by the device at queue  $j$ .

Furthermore, let  $S_i^j$  represent the expected time required for the device to arrive at queue  $i$  and inspect it given that it was serving a move request from station  $j$  (when the tagged load arrived). Likewise, let  $\bar{S}_i^j$  represent the expected time required for the device to arrive at queue  $i$  and inspect it given that it was traveling empty from station  $j$  to  $j+1$  (when the tagged load arrived). Using  $S_i^j$  and  $\bar{S}_i^j$ , we obtain the expected waiting time for the tagged load when there are no other loads present in queue  $i$  as

$$W_i(0) = \sum_j \alpha_{fj} S_i^j + \sum_j \alpha_{ej} \bar{S}_i^j. \quad (11)$$

Note that  $W_i(0)$  also refers to the expected waiting time experienced by the first, or Head-Of-Line (HOL), load in the queue when the tagged load arrives.

Consider next the case where  $n > 0$ . For this case,  $W_i(n)$  is equal to the expected waiting time experienced by the

HOL load and the expected time required to pick-up the remaining  $n - 1$  loads plus the tagged load itself. That is,

$$W_i(n) = W_i(0) + nC_i. \quad (12)$$

The expected waiting time with  $n > 0$  is therefore given by

$$\sum_{n=1}^{\infty} P_i(n) W_i(n) = W_i(0) \sum_{n=1}^{\infty} P_i(n) + C_i \sum_{n=1}^{\infty} n P_i(n). \quad (13)$$

Since  $\sum_{n=1}^{\infty} P_i(n) = 1 - P_i(0)$ , and  $\sum_{n=1}^{\infty} n P_i(n) = \lambda_i W_i$  from Little's Law, we substitute Equations (11) and (13) into (8) and simplify the result to obtain

$$W_i = \frac{\sum_j \alpha_{fj} S_i^j + \sum_j \alpha_{ej} \bar{S}_i^j}{1 - \lambda_i C_i} = \frac{\sum_j \alpha_{fj} S_i^j + \sum_j \alpha_{ej} \bar{S}_i^j}{q_i}, \quad (14)$$

leaving  $S_i^j$  and  $\bar{S}_i^j$  as the unknowns to be determined.

Consider first  $S_i^j$ , i.e., the expected time required for the device to arrive at queue  $i$  and inspect it given that it was serving a move request from station  $j$  when the tagged load arrived. Let  $k$  denote the destination of the move request picked up from station  $j$ . Furthermore, let  $\delta_{ki}$  denote the time required for the device to arrive at queue  $i$  and inspect it given that it just completed inspecting station  $k$ . Since we assume that loads arrive at random points in time, we have

$$S_i^j = \sum_{k \neq j} \left[ \tau_{jk}^{(2)} / (2\tau_{jk}) + \Delta_{ki} \right] p_{jk}, \quad (15)$$

where  $\Delta_{ki} = E[\delta_{ki}]$ , and  $\tau_{jk}^{(2)}$  is the second moment of  $\tau_{jk}$ . Note that the first term in the above summation represents the residual loaded travel time from station  $j$  to station  $k$  (Kleinrock, 1975). Also,  $\Delta_{ii} = 0$  by definition. Hence, given the  $\Delta_{ki}$  values (which we will derive shortly), one can obtain the  $S_i^j$  values.

Consider next  $\bar{S}_i^j$ , i.e., the expected time required for the device to arrive at queue  $i$  and inspect it given that it was traveling empty from station  $j$  to  $j+1$  when the tagged load arrived. Since loads are assumed to arrive at random points in time, we have:

$$\bar{S}_i^j = [\sigma_{j,j+1}^{(2)} / (2\sigma_{j,j+1})] + \Delta_{j+1,i}, \quad (16)$$

where  $\sigma_{j,j+1}^{(2)}$  is the second moment of  $\sigma_{j,j+1}$ .

We now derive the  $\Delta_{yx}$  values that are needed in Equations (15) and (16). Recall that  $\delta_{yx}$  is the time required for the device to arrive at queue  $x$  and inspect it given that it just completed inspecting station  $y$ . Let  $\bar{Q}_y$  ( $Q_y$ ) represent the event that, upon inspection, the device finds (does not find) a load at queue  $y$ . Hence, we have:

$$\Delta_{yx} = E[\delta_{yx}] = E[\delta_{yx} | \bar{Q}_y] P[\bar{Q}_y] + E[\delta_{yx} | Q_y] P[Q_y], \quad (17)$$

where  $y \neq x$ . Note that  $P[\bar{Q}_y] = \bar{q}_y$  and  $P[Q_y] = q_y$ , by definition. (See Equation (4) for  $q_y$ ).

The first conditional expression in Equation (17),  $E[\delta_{yx} | \bar{Q}_y]$ , represents the expected time the device spends

to deliver a load from station  $y$  and then arrive at queue  $x$  and inspect it. In other words,  $E[\delta_{yx}|\bar{Q}_y]$  is the expected loaded travel time from station  $y$  to station  $z$  ( $z \neq y$ ) plus the expected time required for the device to arrive at queue  $x$  and inspect it given that it just completed inspecting station  $z$ . Hence,

$$E[\delta_{yx}|\bar{Q}_y] = \sum_{z \neq y} (\tau_{yz} + \Delta_{zx}) p_{yz}. \quad (18)$$

Note that  $z$  may be equal to  $x$ , in which case  $\Delta_{xx}$  is equal to zero, by definition.

The second conditional term in Equation (17),  $E[\delta_{yx}|Q_y]$ , represents the expected time required for the device to travel empty from station  $y$  to  $y+1$ , plus the expected time required for the device to arrive at queue  $x$  and inspect it given that it just completed inspecting station  $y+1$ . That is,

$$E[\delta_{yx}|Q_y] = \sigma_{y,y+1} + \Delta_{y+1,x}. \quad (19)$$

Again, if  $y+1 = x$ ,  $\Delta_{xx} = 0$ .

Substituting Equations (18) and (19) into Equation (17), we rewrite the expression for  $\Delta_{yx}$  as follows:

$$\Delta_{yx} = \left[ \bar{q}_y \sum_{z \neq y} (\tau_{yz} + \Delta_{zx}) p_{yz} \right] + q_y (\sigma_{y,y+1} + \Delta_{y+1,x}), \quad (20)$$

where  $y \neq x$  on the left-hand side, and if  $z = x$  on the right-hand side,  $\Delta_{xx} = 0$ . Let  $\bar{\tau}_y$  denote the expected loaded travel time out of queue  $y$ ; that is,  $\bar{\tau}_y = \sum_{z \neq y} \tau_{yz} p_{yz}$ . Next, let  $\beta_y$  denote the expected travel time out of queue  $y$ . That is,  $\beta_y = \bar{q}_y \bar{\tau}_y + q_y \sigma_{y,y+1}$ . Rewriting the right-hand side of Equation (20), we obtain:

$$\Delta_{yx} = \beta_y + \left( \bar{q}_y \sum_{z \neq y} p_{yz} \Delta_{zx} \right) + q_y \Delta_{y+1,x}. \quad (21)$$

For  $M$  stations, the number of  $\Delta_{yx}$  values we need to compute is equal to  $M(M-1)$ . We also have  $M(M-1)$  linear equations, one for each  $\Delta_{yx}$  as obtained from Equation (21). Furthermore, the  $M(M-1)$  equations with  $M(M-1)$  unknowns partition into a set of  $M$  equations with each set containing  $M-1$  equations with  $M-1$  unknowns.

For example, if  $M = 4$  we have 12 unknowns and 12 equations. However, the first three equations contain only the terms  $\Delta_{21}$ ,  $\Delta_{31}$ , and  $\Delta_{41}$ ; the next three equations (fourth through sixth) contain only the terms  $\Delta_{12}$ ,  $\Delta_{32}$ , and  $\Delta_{42}$ ; the next three equations (seventh through ninth) contain only the terms  $\Delta_{13}$ ,  $\Delta_{23}$ , and  $\Delta_{43}$ ; and the last three equations contain only the terms  $\Delta_{14}$ ,  $\Delta_{24}$ , and  $\Delta_{34}$ . Hence, instead of solving 12 equations in 12 unknowns, we need to solve four sets of equations where each set contains only three equations in three unknowns.

Despite the above simplification, proving that the equations in question will always have a (unique) solution for arbitrary  $\bar{\tau}_y$  and  $p_{yz}$  ( $0 \leq p_{yz} \leq 1$ ) values is not straightforward even when we know that the system is

stable; i.e.,  $\alpha_f + \phi < 1$ . We solved a large number of problems and consistently obtained a unique solution for the  $\Delta_{yx}$  values as long as the system is stable.

Hence, based on empirical observations, we can only state that the linear system of equations given by Equation (21) appears to have a unique solution as long as the system is stable. Given the  $\Delta_{yx}$  values, we simply substitute Equations (9), (10), (15), and (16) into Equation (14) to obtain the expected waiting time at queue  $i$  as follows:

$$W_i = \frac{1}{q_i} \sum_j \left[ \left( \sum_k f_{jk} \tau_{jk} \right) \left( \sum_{k \neq j} \frac{\tau_{jk}^{(2)} p_{jk}}{2\tau_{jk}} + \sum_{k \neq j} \Delta_{ki} p_{jk} \right) \right] + \frac{1}{q_i} \sum_j \left[ \left( \frac{q_j \sigma_{j,j+1}}{C_j} \right) \left( \frac{\sigma_{j,j+1}^{(2)}}{2\sigma_{j,j+1}} + \Delta_{j+1,i} \right) \right]. \quad (22)$$

The expected queue length at queue  $i$  is equal to  $\lambda_i W_i$  by Little's Law. In the next section we present two numerical examples to show empirically the performance of the model.

## 5. Numerical examples and conclusions

The two examples we present in this section are inspired by tandem AGV systems (Bozer and Srinivasan, 1991) used in a manufacturing setting, where each station may serve either as an "I/O station" (i.e., entry and exit point for "jobs" processed through the system) or a "processor station" (where the loads are processed one at a time). The device picks up a load from one of the I/O stations and moves it to the appropriate processor station, say, station  $i$ . As they are delivered by the device, each load queues up in front of the processor at station  $i$  and waits for processing. When processing at station  $i$  is complete, the load enters the move request queue of station  $i$  (i.e., queue  $i$ ) and waits for the device to move it to the next processor station or an I/O station, whichever is appropriate. Loads delivered to an I/O station exit the system immediately.

The processing times at each processing station are assumed to be exponentially distributed. The mean processing time at each processing station is such that the average processor utilization is equal to 0.80. Note that, at processor station  $i$ , even with exponential processing times, move requests will not necessarily enter queue  $i$  in a Poisson fashion (unless of course the loads delivered by the device happen to follow a Poisson distribution). Hence, in manufacturing applications with processor stations, assumption (1) (which assumes Poisson arrivals at each move request queue) becomes a source of error for processor stations. (The other source of error is assumption (6), which assumes that loads are delivered by the device at random points in time).

The jobs processed through the system and the stations visited by each job type is specified by the user. In both

examples, flow is assumed to be conserved at each processor station. However, since a job may enter from one I/O station and exit from another, flow is not necessarily conserved at the I/O stations.

We used a GPSS/H-based simulation model to obtain simulated values for the expected WIP at each queue. All simulation results are based on 10 replications and 3000 loaded trips per replication. In the simulation model, move requests arrive at the I/O stations in a Poisson fashion. However, move request arrivals at processor stations are governed by the simulation model – we did not force them to follow a Poisson pattern. Hence, given Poisson external arrivals at the I/O stations, and exponential processing times at the processor stations, the simulation model mimics “actual” load/device movements in the system. Therefore, errors observed between the simulation and analytical results are due to assumption (6) at I/O stations, and assumptions (1) and (6) at processor stations.

The first example is based on a single, bidirectional AGV serving a set of eight stations as shown in Fig. 1a. Stations 1 and 8 serve as I/O stations while the remaining stations are processor stations. Each grid in Fig. 1 (a and b) measures 10 feet. In traveling from one station to another, the AGV is required to follow the (shaded) guidepath. For example, the distance from station 1 to station 2 is 3 grids (30 feet). The AGV travels at a speed of 150 fpm (empty or loaded), and it takes 0.15 minutes to pick-up or deposit a load. (The pick-up/deposit time is simply added to the travel time when the AGV moves a load).

Six job types are processed in the system; the flow rate and stations visited by each job type are given as follows: Job 1 (3/hr) 1-2-3-4-8; Job 2 (7/hr) 8-7-6-5-1; Job 3 (2/hr) 1-2-3-4-7-6-5-1; Job 4 (2.5/hr) 8-7-6-5-2-3-4-8; Job 5 (2/hr) 1-5-6-7-8; and Job 6 (1/hr) 8-4-3-2-1. The resulting from-to chart for the above problem is presented as Table A1 in the Appendix.

Suppose the polling sequence is given by 1-2-3-4-8-7-6-5; i.e., the AGV polls the stations in a near-clockwise fashion. The AGV is able to meet the throughput requirement; i.e., the system is stable since  $\alpha_f + \phi = 0.814$ .

Since a single AGV (with no other vehicles sharing the guidepath) is likely to show little or no variation in travel times, we simulated two types of AGV travel times: deterministic and uniform (where the latter is assumed to be within  $\pm 20\%$  of the mean). The analytical and simulated expected WIP levels (i.e., expected number of loads waiting at each station) are shown in Table 1, where the “*abs. error*” column represents the absolute difference between the analytical result (“*Analy.*”) and the sample mean obtained from simulation (“*Sim.*”).

The analytical model performs reasonably well with small absolute errors (and acceptable percent errors). There is little or no difference in expected WIP levels between deterministic and uniform AGV travel. This is

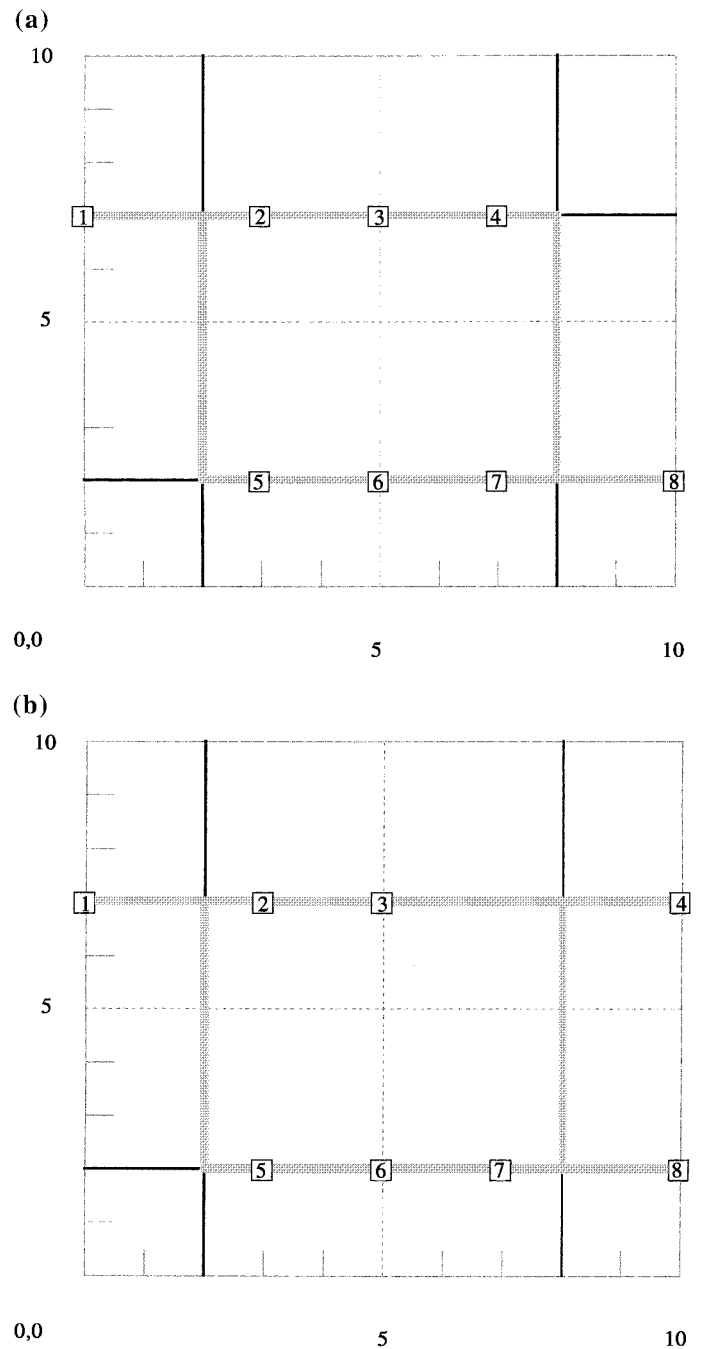


Fig. 1. Layouts (a and b) for Example 1 (with alternative locations for station 4).

perhaps due to the small variation we used in the uniform distribution.

Consider the opposite polling sequence, i.e., the near-counterclockwise sequence 1-5-6-7-8-4-3-2. It is straightforward to show that the above polling sequence has a workload factor of  $\alpha_f + \phi = 0.814$ , which is the *same workload factor* obtained under the original near-clockwise polling sequence. However, assuming deterministic travel times, in Table 2 (see the two columns labeled

**Table 1.** Analytical and simulated expected WIP results for Example 1

Station	Analy.	Deterministic/exponential		Abs. error	Analy.	Uniform/exponential		Abs. error
		Sim.	95% CI			Sim.	95% CI	
1	0.52	0.52	0.45–0.60	0.00	0.52	0.58	0.52–0.64	0.06
2	0.57	0.63	0.52–0.73	0.05	0.57	0.59	0.54–0.64	0.02
3	0.57	0.62	0.53–0.70	0.05	0.57	0.56	0.50–0.61	0.01
4	0.56	0.55	0.48–0.61	0.01	0.56	0.58	0.53–0.63	0.02
5	1.40	1.29	1.11–1.46	0.11	1.40	1.31	1.09–1.52	0.09
6	1.43	1.40	1.15–1.65	0.03	1.44	1.46	1.24–1.69	0.03
7	1.46	1.41	1.19–1.63	0.06	1.47	1.57	1.19–1.95	0.11
8	1.20	1.32	1.08–1.57	0.12	1.20	1.31	1.07–1.55	0.11
Average abs. error				0.05				0.06
Percentage abs. error (%)				5.35				5.44

“analy. – Fig. 1a”) we observe that the expected total WIP level obtained with the near-counterclockwise polling sequence is more than twice of that obtained with the near-clockwise sequence. (The analytical model correctly identifies the preferred polling sequence). Considering that the workload factor is the same between the two polling sequences, the large difference above in expected WIP levels underlines the importance of the proposed model.

We also use the first example to show that the expected WIP level is not necessarily proportional to the workload factor, which is a significant observation considering that the workload factor determines whether the system is stable or not. Suppose station 4 in Fig. 1a is moved out slightly, resulting in the layout shown in Fig. 1b. (This is the only change made).

With reference to Table 2 (please see the two columns labeled “Analy. – Fig. 1b”), the near-clockwise polling sequence yields a workload factor of 0.865 and a total expected WIP of 11.73 loads (assuming deterministic travel times), while the near-counterclockwise polling sequence yields a workload factor of 0.852 and a total

expected WIP of 19.93 loads. Although the latter polling sequence has a smaller workload factor, it results in significantly larger expected WIP! (The workload factor is computed using mean travel times. Any difference in  $\alpha_f + \phi$ , no matter how small, is a “real” difference).

The second example is a two-zone tandem AGV system serving 12 stations numbered 1 through 12. Stations 1 and 9 serve as I/O stations and the remaining stations are processor stations. Also, two additional stations (13 and 14) serve as transfer stations between the two zones; please refer to Fig. A1 in the Appendix. Zone 1 consists of stations 1 through 6 (plus station 13), while zone 2 consists of stations 7 through 12 (plus station 14). Each grid in Fig. A1 measures 10 ft and each AGV travels at 150 fpm (empty or loaded). The time required to pick-up (or deposit) a load is equal to 0.20 minutes. All loads going from one zone to another must go through stations 13 and 14. (Two one-way conveyors are assumed to be provided between stations 13 and 14).

Three job types are handled by the AGVs; the flow rate and the stations visited by each job type are given as follows: Job 1 (2/hr) 1-6-3-13-14-7-11-10-8-9; Job 2 (2/hr)

**Table 2.** Impact of polling sequences on expected WIP, Example 1

Polling sequence	1,2,3,4,8,7,6,5 Analy.–Fig. 1a	1,5,6,7,8,4,3,2 Analy.–Fig. 1a	1,2,3,4,8,7,6,5 Analy.–Fig. 1b	1,5,6,7,8,4,3,2 Analy.–Fig. 1b
Station				
1	0.52	1.01	0.70	1.24
2	0.57	2.01	0.75	2.60
3	0.57	2.27	0.74	2.80
4	0.56	2.02	0.73	2.47
5	1.40	1.91	2.23	2.38
6	1.43	1.99	2.29	2.44
7	1.46	2.18	2.34	2.60
8	1.20	2.54	1.92	3.40
Analy. WIP	7.72	15.92	11.73	19.93
Sim. WIP	7.74	17.48	10.93	22.71

**Table 3.** Analytical and simulated expected WIP results for Example 2

Zone	Station	Analy. Deterministic/exponential			Abs. error	Analy. Uniform/exponential			Abs. error
		Sim.	95% CI	95% CI		Sim.	95% CI	95% CI	
1	1	1.97	2.12	1.47–2.77	0.15	1.97	2.20	1.32–3.07	0.23
	2	0.45	0.43	0.39–0.47	0.03	0.46	0.39	0.34–0.44	0.07
	3	0.35	0.36	0.33–0.39	0.01	0.36	0.35	0.31–0.39	0.00
	4	0.32	0.33	0.31–0.35	0.01	0.32	0.32	0.27–0.37	0.00
	5	1.75	1.98	1.28–2.67	0.23	1.75	2.01	1.02–3.00	0.26
	6	1.58	1.82	1.07–2.57	0.24	1.59	1.84	1.29–2.38	0.25
	13	0.27	0.29	0.26–0.33	0.02	0.27	0.29	0.25–0.33	0.02
Average abs. error					0.10				
Percentage abs. error (%)					7.24				
2	7	1.30	1.18	0.99–1.36	0.13	1.31	1.14	1.05–1.24	0.16
	8	0.83	0.85	0.71–0.99	0.02	0.83	0.82	0.69–0.94	0.02
	9	–	–	–	–	–	–	–	–
	10	0.74	0.68	0.60–0.76	0.06	0.74	0.72	0.63–0.81	0.02
	11	0.71	0.71	0.63–0.78	0.00	0.71	0.71	0.62–0.80	0.00
	12	0.17	0.18	0.16–0.19	0.01	0.17	0.17	0.15–0.18	0.00
	14	1.12	0.87	0.76–0.98	0.25	1.12	0.91	0.82–1.00	0.21
Average abs. error					0.08				
Percentage abs. error (%)					9.53				
Overall avg. abs. error					0.09				
Overall percentage abs. error (%)					8.30				

1-2-5-6-2-4-13-14-12-14-13-3-1; and Job 3 (2.5/hr) 1-5-13-14-7-8-7-10-7-11-9. (The resulting from-to chart is presented as Table A2 in the Appendix).

The polling sequence for zone 1 and zone 2 is given by 1-6-5-13-4-2-3 and 7-11-10-9-12-14-8, respectively. Both zones are stable since the  $\alpha_f + \phi$  value for zone 1 and zone 2 is equal to 0.851 and 0.708, respectively. As before, we simulated two types of travel times for the AGVs: deterministic and uniform (where the latter is assumed to be within  $\pm 20\%$  of the mean).

The results for each zone are shown in Table 3. Although there are only a few stations in each zone (and the move requests from processor stations do not necessarily follow a Poisson pattern as required by assumption (1)), the analytical model performs reasonably well. Of course, another source of error is the fact that the AGVs do not necessarily inspect the stations at random points in time as required by assumption (6).

In general, for the conditions under which we tested the model, it estimates the expected WIP levels reasonably accurately to show the relative differences between the stations and to evaluate alternative polling sequences. In those cases where the coefficient of variation for the processing times at the processor stations are small, however, the model accuracy may deteriorate since assumption (1) would be less tenable. Nevertheless, we believe such results will be useful in the design and evaluation of polling-based single-device handling systems.

A possible future research direction is to generalize the results presented in this study by allowing the device to poll some of the stations twice or more during a polling cycle. Such an extension would make it possible to analyze configurations where the stations are not necessarily located around a closed-loop or similar path.

### Acknowledgements

Study was partially supported by the Center for Display Technology and Manufacturing (DTM) at the University of Michigan. The presentation of the paper, including explanation of the assumptions and their role as well as the examples we used, benefited greatly from the feedback we received from the referees. We are grateful for their careful review and suggestions for improvement.

### References

- Bartholdi, J.J., III and Platzman, L.K. (1989) Decentralized control of automated guided vehicles on a simple loop. *IIE Transactions*, **21**(1), 76–81.
- Bozer, Y.A., Cho, M.S. and Srinivasan, M.M. (1994) Expected waiting times in single-device trip-based material handling systems. *European Journal of Operational Research*, **75**, 200–216.
- Bozer, Y.A. and Kim, J. (1996) Determining transfer batch sizes in trip-based material handling systems. *International Journal of Flexible Manufacturing Systems*, **8**, 313–356.



Bozer, Y.A. and Srinivasan, M.M. (1991) Tandem configurations for automated guided vehicle systems and the analysis of single-vehicle loops. *IIE Transactions*, **23**(1), 72–82.

Chow, W.M. (1986) Design for line flexibility. *IIE Transactions*, **18**(1), 95–108.

Kleinrock, L. (1975) *Queueing Systems, Volume 1: Theory*, Wiley, New York, p. 169.

Srinivasan, M.M. (1991) Nondeterministic polling systems. *Management Science*, **37**(6), 667–681.

Srinivasan, M.M., Bozer, Y.A. and Cho, M.S. (1994) Trip-based material handling systems: throughput capacity analysis. *IIE Transactions*, **26**(1), 70–89.

Stone, H.S. and Fuller, S.H. (1973) On the near-optimality of the shortest-latency-time-first drum scheduling discipline. *Communications of the ACM*, **16**(6), 352–353.

Tompkins, J.A., White, J.A., Bozer, Y.A., Frazelle, E.H., Tanchoco, J.M.A. and Trevino, J. (1996) *Facilities Planning*, 2nd edn, Wiley, New York.

Wolff, R.W. (1982) Poisson arrivals see time averages. *Operations Research*, **30**, 223–231.

**Appendix**

**Table A1.** Flow matrix for Example 1 (Loads/hr)

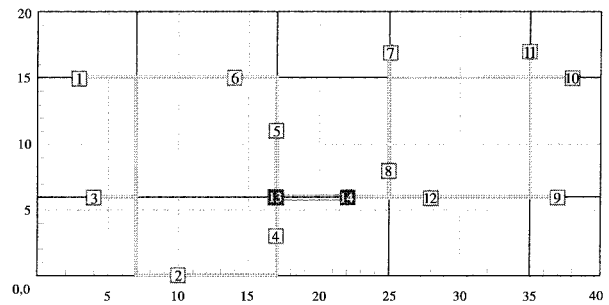
To/From	1	2	3	4	5	6	7	8
1	0	5	0	0	2	0	0	0
2	1	0	7.5	0	0	0	0	0
3	0	1	0	7.5	0	0	0	0
4	0	0	1	0	0	0	2	5.5
5	9	2.5	0	0	0	2	0	0
6	0	0	0	0	11.5	0	2	0
7	0	0	0	0	0	11.5	0	2
8	0	0	0	1	0	0	9.5	0

Device Speed: 150 ft/min.  
 Pickup/Deposit Time: 0.15 min.

**Table A2.** Flow matrix for Example 2 (Loads/hr)

To/From	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1		2			2.5	2								
2				2	2									
3	2												2	
4													2	
5						2							2.5	
6		2	2											
7							2.5	2.5	4.5					
8							2.5	2						
9														
10							2.5	2						
11									2.5	2				
12													2	
13			2										6.5	
14							4.5					2	2	

Device Speed: 150 ft/min.  
 Grid Conversion: 10 ft/distance unit.  
 Pickup/Deposit Time: 0.20 min.



**Fig. A1.** Layout for Example 2, two-zone tandem AGV system.

**Biographies**

Yavuz A. Bozer received his B.S.I.E. degree from the Middle East Technical University (METU), Turkey and his M.S. (1978) and Ph.D. (1985) degrees in Industrial and Systems Engineering from Georgia Institute of Technology. Currently, he is a Professor with the Department of Industrial and Operations Engineering at the University of Michigan, Ann Arbor, and the Goff Smith Engineering Co-Director of the Tauber Manufacturing Institute (TMI). He is also the Engineering Global Leadership (EGL) program advisor and Co-Director of the Lean Manufacturing Certificate Program. His teaching and research interests focus on the design, efficient operation, and performance evaluation of material flow and storage systems within facilities and across facilities/suppliers as well as single-floor and multi-floor facility layout and design problems. Professor Bozer’s professional experience includes his full-time employment with the SysteCon Division of Coopers & Lybrand and subsequently the Material Handling Research Center at Georgia Tech. He received the 1987 Outstanding Dissertation Award from the Institute of Industrial Engineers (IIE). In 1988 he was named a Presidential Young Investigator (PYI) by NSF. Professor Bozer was inducted into the Council of Outstanding Young Engineering Alumni at Georgia Tech in 1995, and he received the Technical Innovation Award in Industrial Engineering from IIE in 1999. He is a member of IIE, SME, and INFORMS. His publications have appeared in leading refereed journals in his area. He is also a co-author of a well-regarded textbook entitled *Facilities Planning*, second edition, Wiley.

Jung H. Park is the Director of Marketing Analytics at ePocrates in San Carlos, California. He earned his Master’s Degree and Ph.D. in Industrial and Operations Engineering from The University of Michigan and a B.S. in Engineering from Harvey Mudd College. He has a business and consulting background, having provided services to over twenty companies in the financial, pharmaceutical, electronic, and automotive industries.

*Contributed by the Facilities Layout and Material Handling Department*