

# Open Location Management in Automated Warehousing Systems

Yugang Yu

School of Management, the University of Science and Technology of  
China

René de Koster

Rotterdam School of Management, Erasmus University

## Abstract

A warehouse needs to have sufficient open locations to be able to deal with the change of item inventory levels, but due to ongoing storage and retrieval processes, open locations usually spread over storage areas. Unfavorable positions of open locations negatively impact the average load retrieval times. This paper presents a new method to manage these open locations such that the average system travel time for processing a block of storage and retrieval jobs in an automated warehousing system is minimized. We introduce the effective storage area (ESA), a well-defined part of the locations closest to the depot; where only a part of the open locations –the *effective* open locations-, together with all the products, are stored. We determine the optimal number of effective open locations and the ESA boundary minimizing the average travel time. Using the ESA policy, the travel time of a pair of storage and retrieval jobs can be reduced by more than 10% on average. Its performance depends hardly on the number or the sequence of retrievals. In fact, in case of only one retrieval, applying the policy leads already to beneficial results. Application is also easy; the ESA size can be changed dynamically during storage and retrieval operations.

*Keywords:* Distribution science, warehousing; AS/RS; storage and retrieval; open locations

## 1 Introduction

Warehouses or distribution centers are key nodes in supply chains. They decouple demand from supply in space, quantity, and time. They therefore play a crucial role in realizing high supply chain efficiency and service levels. Since the seminal papers of Hausmann et al. (1976), Graves et al. (1977), and Han et al., (1987), warehouse design

and management have received vast attention in management literature (Johnson and Meller, 2002; De Koster et al., 2007; Gu et al., 2007).

In order to run a warehouse storage system efficiently, a sufficient number of open locations (or empty slots) are necessary. This number depends on the change of the inventory levels over time and the way the system operates. If the number is insufficient, much time can be required in storage (and later in retrieval) because of lack of chance to find an open location close to the depot for incoming storage loads or difficulty in pairing open locations and retrieval locations. According to Tompkins et al. (2003, p. 403), the rule of thumb in practice is: “when a warehouse is more than 80% full, more space is needed” and “this rule is based on the fact that when a warehouse reaches this capacity, it takes longer to put something away”. For automated unit load warehousing systems (where pallets or totes are stored and retrieved by a storage and retrieval -S/R- machine) several researchers (e.g., Graves et al., 1977; Han et al., 1987; Meller and Mungwattana, 1997) have shown that, by sequencing a given block of storage and retrieval jobs, the average travel time of an S/R-machine for a pair of storage and retrieval jobs decreases with an increasing number of available open locations (see Figure 1). The more open locations the system has, the easier the system finds them to store incoming loads, and to optimally combine the storages with retrievals. However if the number of open locations increases beyond a bound, (e.g.  $m_e$  in Figure 1), increasing the number has little effect on further reducing the cycle travel time. Although present in academic results (reviewed in Section 2) in past decades, this effect has not been noted in past literature.

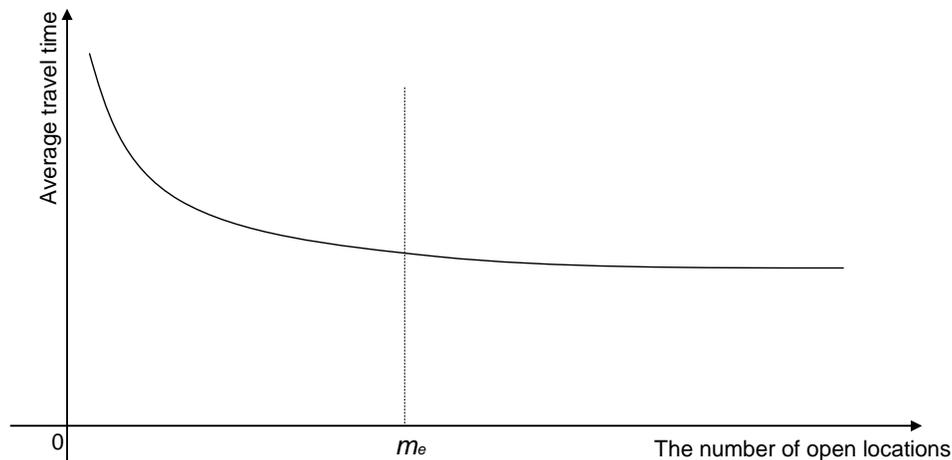


Figure 1: The Effect of the Number of Open Locations on the Average Travel Time of a Pair of Storage/Retrieval Jobs

We make use of this phenomenon by no longer allowing all the open locations (see Figure 2(a)) to be available to incoming storage jobs, but only those locations in an Effective Storage Area (ESA) (see Figure 2(b)). The ESA contains only a part of the open locations closest to the depot and all the products stored. The open locations within the

ESA are called *effective* open locations. The other, or *ineffective* open locations, are located outside the ESA and form an Ineffective Storage Area (ISA) (see Figure 2(b)). In order to create an ESA of a given size, unit loads stored outside have to be swapped with inside open locations. Once the ESA has been created, it is easy to maintain. In situations where only double plays are carried out (a storage job combined with a retrieval) we only store and retrieve unit loads within the ESA.

As stored products may have to be retrieved at any location within the entire ESA, after some time of operation, the open locations will be scattered randomly within the ESA and the ESA will look like in Figure 2(b).

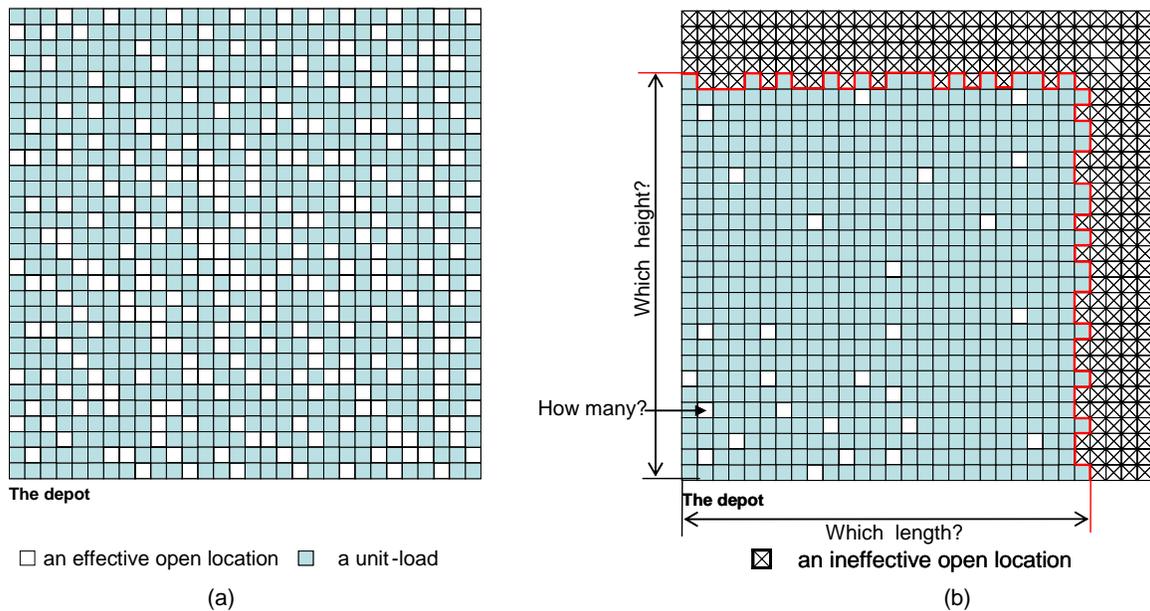


Figure 2: Effective Storage Area (ESA) and Ineffective Storage Area (ISA)

The number of effective open locations influences the average storage and retrieval time as it determines, together with the given number of stored unit loads, the size of ESA. If the ESA contains too many open locations it becomes large, which negatively impacts the average storage and retrieval time. If it contains too few open locations, it is difficult to match a retrieval to a close storage job leading to too long average storage and retrieval times. Our main research questions therefore are: (1) how many effective open locations ( $m_e$ ) should the ESA have and (2) what is the optimal boundary of the ESA to minimize the average travel time of a pair of S/R jobs. Equivalently, we want to minimize the makespan of a given block of S/R jobs that have to be carried out.

The idea of storing and retrieving loads using a shrunk and optimized ESA looks intriguingly simple and intuitive. However, it has not been studied in previous literature. Although in practice many different storage policies are used, depending on situation and possibilities (like class-based storage, or pre-shuffling known future retrievals in idle periods), we are not aware of companies that persistently manage the positions of open

locations. We show use of the ESA policy can lead to substantial reductions in average travel time for storage and retrieval jobs in case of random demand compared to the situation where open locations are not explicitly managed, i.e. they are scattered over the storage area.

We model the research problem for a given block of storage and retrieval jobs that have to be processed. The problem is complex due to the nonlinearity of the objective function, nonlinearity of constraints and integrality of the decision variable: the number of effective open locations, as we will show later. Fortunately, we can obtain the optimal solution numerically in an efficient way.

In the model we use the same warehousing system as described in several seminal papers (e.g. Hausman et al., 1976; Graves et al., 1977; Han et al., 1987), namely an automated storage and retrieval system (**AS/RS**). These systems have been widely used to replace conventional manual warehouses since the 1950s (Lee and Schaefer, 1996). Typically, an AS/RS consists of a storage/retrieval (**S/R**) machine, a storage rack, a depot (or I/O point), and products stored on unit-loads (standardized pallets or totes). The unit-loads enter and leave the system at the depot. They are stored and retrieved by the S/R machine to and from the storage rack. In an AS/RS, the S/R machine's capacity normally is one unit-load. Therefore the system can operate in two command modes:

**Single-command Cycle (SC) mode:** In a travel cycle of the S/R machine a single job, either a storage or a retrieval, is performed. To store a unit-load, the S/R machine picks up a unit-load from the depot, moves, and deliveries it to an open location. After that, the S/R machine returns to the depot to complete the SC. To retrieve a unit-load, the process is reversed.

**Dual-command Cycle (DC) mode:** In a travel cycle of the S/R machine a storage is paired with a retrieval. The S/R machine picks up a unit load from the depot to store it at an open location, and then moves empty to a retrieval location to retrieve a unit-load. After this, the S/R machine returns to the depot and completes the DC. The empty travel time between the storage and retrieval location is called the *interleaving* travel time.

In operation, the DC mode is preferred because it can bring approximately 30% travel time reduction compared with the SC mode (Graves et al., 1977) for a pair of S/R jobs. Rather than comparing the performance of a single pair of storage and retrieval jobs, our research is based on processing a given block of dual command cycles (or S/R jobs). For this, a policy to sequence these DCs has to be selected.

For a given block of S/R jobs, the unit loads to be stored arrive and commonly wait on an accumulating conveyor in front of the depot. Only the first load can be picked up by the S/R machine. Therefore storage jobs can only be served in a first-come first-served (*FCFS*) sequence. The storage locations can be selected among all open locations in the rack. Retrievals can be sequenced freely as every retrieval location in the rack face is reachable by the S/R machine. Therefore the sequencing policies mainly focus on how to pair an open location and a retrieval as a DC and how to sequence multiple retrievals. This paper adopts the nearest neighbor policy, which sequences retrieval jobs based on the interleaving distance between storage and retrieval locations. The smallest one is processed next.

The remainder of the paper is organized as follows. In Section 2, we review literature. In Section 3, we formulate a mathematical model to determine the optimal number of effective open locations and the boundary of the ESA. In section 4, we develop an algorithm to obtain the optimal solution of the model based on some solution properties. In section 5, we evaluate the ESA policy by various numerical examples. In Section 6, we discuss how to implement the ESA policy in practice. Lastly, in Section 7, we conclude the paper and provide some future research directions.

## 2 Literature Review

This section reviews papers related to open location selection, retrieval sequencing policies and the decreasing marginal effect of adding open locations illustrated in Figure 1.

**Open location selection.** To our knowledge, no academic literature directly focuses on open location management in terms of positioning and numbering of open locations. However there is some literature about open location selection where each time one open location is selected from all open locations for a storage job. For example, the COL (closest open location) storage policy stores every incoming pallet at the open location closest to the depot (Schwarz et al. 1978). The implementation of the policy can form a forward area (closer to the depot) with pallets and few open locations, and a backward area (further from the depot) with open locations. However, due to lack of proper management of the open locations, Schwarz et al. (1978) and some others have shown that eventually the COL performs almost at the same level as a pure random storage policy with a FCFS policy for storages and retrievals. All sequencing policies mentioned in the next two paragraphs also contain some rules about selecting open locations for storages. However, none consider explicitly managing open locations.

**Retrieval sequencing.** Optimal sequencing a block of storage and retrieval jobs is a NP-hard problem under random or class-based storage (Han et al., 1987; Bozer and White, 1990; Gu et al., 2007). Therefore most literature focuses on solving the problem with various sequencing heuristic policies and storage policies such as FCFS (first-come-first-served) (Graves et al., 1977; Han et al., 1987; Gu et al., 2007), NN (nearest-neighbor) (Han et al., 1987), SDC (shortest dual command cycle) (Lee and Schaefer, 1996), and  $1+\varepsilon$  optimization where  $\varepsilon$  indicates a tolerance gap between the objective value of a solution and a problem lower bound (Lee and Schaefer, 1996). In the above papers, the most frequently cited and effective policy to sequence a block of storage and retrieval jobs one is NN. Compared with FCFS for sequencing storage jobs and retrievals and COL for storage location selection, the results in the above papers show NN can increase the throughput by 10-15%, by reducing the *travel-between* time with 50% or more. The travel time savings depend on the number of open locations and the block size: the number of storage and retrieval jobs. The savings decrease for increasing numbers of open locations and increasing block size. Mahajan et al. (1998) have shown numerically that NN provides near optimal solutions with only 3-6% gaps from a lower bound. In

experiments of Lee and Schaefer (1996) gaps are mostly within 4% from the optimal solution. Therefore NN has been adopted as retrieval sequencing heuristic by many researchers for different system configurations and demand patterns. Meller and Mungwattana (1997) for example, use NN and a variant (RNN-reverse nearest neighbor) for a multi-shuttle AS/RS where the S/R machine has twin- and triple-shuttles with quadruple-command or sextuple-command operational modes, respectively. Eben-Chaime (1992) uses NN as a dispatching rule in an AS/RS with stochastic demand and finds similar results. This paper therefore selects NN.

*Decreasing marginal effect of increasing the number of open locations* (see Figure 1). This phenomenon, appears to exist in unit-load warehousing systems with different sequencing heuristics (e.g. Graves et al., 1977; Han et al., 1987; Lee and Schaefer, 1996), storage policies (e.g. Graves et al., 1977; Lee and Schaefer, 1997), different types of retrieval machines (e.g. Meller and Mungwattana, 1997), and demand patterns (e.g. Schwarz et al., 1978; Eben-Chaime, 1992). As an example, Graves et al. (1977) study retrieval sequencing for an AS/RS with class-based storage. An inbound load has to be stored in its appropriate class. Then the first  $K$  jobs in the retrieval queue are sequentially examined to find a retrieval in the same class as the storage load to construct a DC. If no such load can be found the first job in the retrieval queue is selected. Schwarz et al. (1978) apply the same sequencing policy, under a dynamic setting. The block size changes dynamically due to stochastic product demand. Lee and Schaefer (1996) propose a shortest dual command cycle (SDC) heuristic and an  $\varepsilon$ -optimum algorithm for the sequencing problem, where  $\varepsilon$  indicates a tolerance gap between the objective value of a solution and a problem lower bound. They compare these heuristics and several other ones, such as NN, and shortest leg, for different system shapes and block sizes. Lee and Schaefer (1997) discuss a sequencing problem in an AS/RS using dedicated storage. Six algorithms, including a static assignment algorithm (ASSTA), a static heuristic algorithm (HRSTA), and a dynamic assignment algorithm (ASDYN), are tested. More examples supporting the curve in Figure 1 can be found in (Sarker et al., 1991; Eynan and Rosenblatt, 1993; Van den Berg and Gademann, 1999), and some relevant review papers (Van den Berg, 1999; De Koster et al., 2007; Gu et al., 2007). However, to our knowledge, disadvantages of having too many open locations have not been studied yet.

### **3 Model Formulation**

#### **3.1 Assumptions and Notations**

The assumptions for the system described in the introduction are as follows (see also Graves et al., 1977; Han et al., 1987):

- Storage and retrieval jobs are carried out one block after another. The system handles a new block of storage and retrieval jobs only when its preceding blocks have been completed. This assumption is relaxed in Section 6.

- The system operates in dual command cycle mode.
- The system objective is to minimize the expected travel time per DC.
- The total storage capacity, the speed of the depth movement mechanisms, and the S/R machine's speeds in the horizontal and vertical directions are known and constant.
- The depot is located at the lower left-hand corner of the rack.
- The rack is considered to be a continuous rectangular pick face.
- The machine can move simultaneously in horizontal and vertical directions, so that travel time is the maximum of horizontal travel time and vertical travel time. When the S/R machine is idle, it stops at the depot. The pick-up and deposit time of a load is not considered (this time is fairly constant for real systems).
- Storage jobs are performed in a FCFS sequence.
- The effective open locations for storage jobs and the retrieval locations for retrieval jobs are selected by the *NN* policy to form pairs of DCs.

The length ( $L$ ) and the height ( $H$ ) of the storage rack form the horizontal and vertical dimensions of the system. The speeds of the S/R machine in the horizontal and vertical directions are  $s_c$  and  $s_h$ , respectively. We define  $t_h = L / s_h$  as length (in time) of the rack and  $t_v = H / s_v$  as height (in time) of the rack. Let  $T = \max\{t_h, t_v\}$  and  $b = \min\{t_h, t_v\} / T$  represent the shape factor of the rack. If  $b=1$ , we call the rack shape square-in-time (**SIT**), and **NSIT** (non-SIT) otherwise. Without loss of generality, we discuss the problem in the dimensions of  $t_h \times t_v = 1 \times b$  by setting  $T=1$ . The corresponding results can then be generalized to other rectangular rack dimensions by multiplying them by  $T \neq 1$  (see also Hausman et al., 1976; Han et al., 1987). The other key notations are:

Sets and parameters:	
$C_N$	The system storage capacity expressed in number of unit loads.
$S$ and $R$	The sets of initial effective open and retrieval locations, respectively.
$n$	The number of retrievals $ R $ , or the block size, or the number of DCs.
$u$	Storage space utilization of the system.
$m$	The total number of open locations in the rack, $m = C_N(1 - u)$ .
$b_e$	The shape factor of the ESA, $b_e = y_e / x_e$ .
$ETB$	Expected travel-between (i.e., interleaving) time in a DC.
$ESC$	Expected SC travel time.
$EDC$	Expected DC travel time of a DC using the ESA policy.
$EDC^{NN}$	Expected DC travel time of a DC without using the ESA policy (the NN policy is used).
Decision variables:	
$x_e$	The length (in time) of the ESA, $0 < x_e \leq 1$ .
$y_e$	The height (in time) of the ESA, $0 < y_e \leq \min(b, x_e)$ . Without loss of generality, we assume $y_e \leq x_e$ .
$m_e$	The number of effective open locations in the ESA, $m_e =  S $ .

## 3.2 Model

We obtain a model, denoted as M, to optimally dimension the ESA and determine  $m_e$  as follows:

Model M:

$$\min EDC(x_e, y_e, m_e) = ESC(x_e, y_e, m_e) + ETB(x_e, y_e, m_e) \quad (1)$$

$$\frac{(m_e + C_N \cdot u)}{C_N} (b \times 1) = x_e \cdot y_e \quad (2)$$

$$0 < x_e \leq 1 \quad (3)$$

$$0 < y_e \leq \min(b, x_e) \quad (4)$$

$$1 \leq m_e \leq m \quad (5)$$

Decision variables are  $x_e$ ,  $y_e$ ,  $m_e$ , and  $m_e$  is an integer.

The objective (1) is to minimize the expected DC travel time for the case where a block of  $n$  DCs is processed by the NN policy. Following the results for  $ESC$  and  $ETB$  in Bozer and White (1984) and Han et al. (1987), we can obtain:

$$ESC(x_e, y_e, m_e) = y_e^2 / (3x_e) + x_e, \quad (6)$$

$$ETB(x_e, y_e, m_e) = \frac{x_e}{n} \sum_{k=m_e}^{n+m_e-1} \int_0^1 z \bar{g}_k(z) dz, \quad (7)$$

where 
$$\bar{g}_k(z) = k[1 - F_e(z)]^{k-1} f_e(z) \quad 0 < z \leq 1, \quad (8)$$

with 
$$F_e(z) = \begin{cases} (2z - z^2) \left(\frac{z}{b_e}\right) \left[2 - \left(\frac{z}{b_e}\right)\right] & 0 < z \leq b_e \\ 2z - z^2 & b_e < z \leq 1 \end{cases}, \quad (9)$$

and 
$$f_e(z) = \begin{cases} 2(1-z) \left(\frac{z}{b_e}\right) \left[2 - \left(\frac{z}{b_e}\right)\right] + z(2-z) \left(\frac{2}{b_e}\right) \left(1 - \frac{z}{b_e}\right) & 0 < z \leq b_e \\ 2(1-z) & b_e < z \leq 1 \end{cases}. \quad (10)$$

are the cumulative density function (CDF) and the probability density function (PDF), respectively.  $b_e = x_e / y_e$  is the shape factor of the ESA.

Constraint (2) implies the ESA stores all unit-loads ( $C_N \cdot u$ ) and effective open locations ( $m_e$ ). Constraints (3), (4) and (5) determine the lower and upper bounds for variables  $x_e$ ,  $y_e$ , and  $m_e$ , respectively.

Both the objective function (1) and constraint (2) are nonlinear.  $m_e$  is an integer variable. The model therefore is non-linear integer. With standard software such as Mathematica5.2 (2005),  $ETB(x_e, y_e, m_e)$  cannot be evaluated analytically within a reasonable time (e.g. less than one hour) for realistic sizes of the rack and retrieval blocks. We therefore have to rely on numerical methods.

## 4 Algorithm and Properties

Because  $x_e$  and  $y_e$  are continuous variables, it is impossible to enumerate every combination to obtain an optimal solution of Model M. To simplify the computation process, in subsection 4.1 an algorithm is developed to obtain the optimal solution of Model M based on some properties. After that, Section 4.2 proves some other properties (Theorems 3-6) helpful in explaining numerical results.

### 4.1. Algorithm

The best ratio of  $x_e$  and  $y_e$  for Model M is proved in Lemma 2 by making use of Lemma 1 below.

**LEMMA 1.** *Let  $B(k) = \int_0^1 u(z) \bar{g}_k(z) dz$  where  $u(z)$  is a continuous, increasing function of  $z$ ,  $0 \leq z \leq 1$  where  $k$  is a positive integer and  $\bar{g}_k(z)$  is defined by (8). We have  $B(k) \geq B(k+1)$  and “=” only holds for  $u(z)$  being a constant.*

**Proof.** See Appendix A.

**LEMMA 2.** *Consider a given system with fixed  $n$ ,  $m$  and  $m_e$ , and two different shapes of the ESA with the same area size. The shape of the ESA closer to SIT leads to a smaller value of  $EDC(x_e, y_e, m_e)$ .*

**Proof.** See Appendix B.

Lemma 2 is intuitive, since the S/R machine can drive and lift simultaneously, implying locations placed on a square (in time) around the depot require the same travel time.

Based on Lemma 2, we obtain Theorem 1 to determine the optimal  $x_e^*$  and  $y_e^*$  as a function of  $m_e$ .

**THEOREM 1.** *For a given rack of dimensions  $l \times b$ , and a given  $m_e$  between  $l$  and  $m$ ,*

(a) *if  $1 \leq m_e \leq C_N(b-u)$ , the shape of the optimal ESA is SIT and the optimal ESA dimensions as a function of  $m_e$  are*

$$x_e^*(m_e) = y_e^*(m_e) = \sqrt{(m_e + C_N \cdot u)b / C_N}. \quad (11)$$

(b) *if  $\max\{C_N(b-u), l\} < m_e \leq m$ , the shape of the optimal ESA is NSIT and the optimal ESA dimensions are*

$$x_e^*(m_e) = (m_e + C_N \cdot u) / C_N, \quad (12)$$

$$y_e^*(m_e) = b. \quad (13)$$

**Proof.** According to Lemma 2, if the ESA can be feasibly constructed in SIT shape, it will be the optimum, otherwise the feasible non-SIT shape closest to SIT (i.e. with the

largest  $b_e = y_e^*(m_e) / x_e^*(m_e)$ ) will be the optimum. That is, for a given  $m_e$  the optimal  $x_e^*(m_e)$  and  $y_e^*(m_e)$  can be determined by the feasible ESA shape closest to SIT.

For Theorem 1(a), because  $m_e \leq C_N(b-u)$  (i.e.,  $(m_e + C_N \cdot u)b / C_N \leq b \cdot b$ ), we can construct a SIT shape ( $b_e=1$ ) with  $x_e^*(m_e) = y_e^*(m_e) = \sqrt{(m_e + C_N \cdot u)b / C_N}$ , which does not violate any Constraints (1)-(5). Therefore, Equation (11) holds and Theorem 1(a) has been proved.

For Theorem 1(b), because  $\max\{C_N(b-u), 1\} < m_e \leq m$  (i.e.,  $(m_e + C_N \cdot u)b / C_N > b \cdot b$ ), the shape of the ESA closest to SIT with feasible  $y_e$  makes Constraint (4) binding with  $y_e^*(m_e) = b$ , while  $x_e^*(m_e) = ((m_e + C_N \cdot u)b / C_N) / b = (m_e + C_N \cdot u) / C_N$ . This results in the largest value of  $b_e$ . Therefore, Equations (12) and (13) hold and Theorem 1(b) has been proved.

Using Equations (11)-(13), we can simplify Model M by eliminating  $x_e$  or  $y_e$ . Model M can therefore be split into two sub-models; one is for the case:  $1 \leq m_e \leq C_N(b-u)$  (denoted by Model M1) where the ESA is SIT, and the other is for the case:  $\max\{C_N(b-u), 1\} \leq m_e \leq m$  (denoted by Model M2) where the ESA is NSIT. We treat these models subsequently.

For Model M1 (only applicable if  $C_N(b-u) \geq 1$ ), substituting Equation (11) into Equation (6), we obtain  $ESC$  as a function of  $m_e$  equaling

$$ESC(m_e) = 4\sqrt{(m_e + C_N \cdot u)b / C_N} / 3, \quad 1 \leq m_e \leq C_N(b-u). \quad (14)$$

By substituting Equation (11) into Equation (7), we can obtain  $ETB$  as a function of  $m_e$  equaling

$$ETB(m_e) = \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} [\sqrt{(m_e + C_N \cdot u)b / C_N} \int_0^1 zk[1 - F_e(z)]^{k-1} f_e(z) dz], \quad 1 \leq m_e \leq C_N(b-u). \quad (15)$$

The objective function of Model M1 is now a function of  $m_e$ , denoted as  $EDC(m_e)$ , and is the sum of Equations (14) and (15).

For the constraints of Model M1, we replace Constraints (2)-(4) of Model M with  $1 \leq m_e \leq C_N(b-u)$ .

For Model M2, in analogy to Model M1,  $ESC$ , as a function of  $m_e$ , equals

$$ESC(m_e) = b^2 / (3(m_e + C_N \cdot u) / C_N) + (m_e + C_N \cdot u) / C_N, \quad (16)$$

$$\text{and} \quad ETB(m_e) = \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} [(m_e + C_N \cdot u) / C_N \int_0^1 zk[1 - F_e(z)]^{k-1} f_e(z) dz], \quad (17)$$

where  $\max\{C_N(b-u), 1\} \leq m_e \leq m$ .

The objective function for Model M2 is the sum of Equations (16) and (17).

The constraints of Model M2 are obtained by replacing Constraints (2)-(4) with  $\max\{C_N(b-u), 1\} \leq m_e \leq m$ .

Therefore, the overall optimal solution is the optimal solution of either Model M1 or Model M2 that provides the minimum objective value. As Model M1 and Model M2 both

are a function of a single decision variable  $m_e$ , they then can be solved optimally by enumerating  $m_e$  between 1 and  $m$  considering their constraints on  $m_e$ .

Theorem 2 below shows it is not necessary to enumerate every  $m_e$  for finding the optimal  $m_e^*$ .

**THEOREM 2.** *EDC( $m_e$ ) has a unique global minimum in Models M where EDC( $m_e$ ) is the combination of objective functions of Models M1 and M2 given in Equations (14)-(17).*

**Proof.** See Appendix C.

Normally  $EDC(m_e)$  is not a convex function of  $m_e$  although it has a “U” (or partly “U”) -shape.

According to Theorem 2, we can choose golden section search (Press et al., 2007) to find  $m_e^*$  and then use Theorem 1 to determine  $(x_e^*, y_e^*)$ .

## 4.2 Some Other Properties

The phenomenon described in Figure 1 is proved by Theorem 3, and a similar phenomenon for EDC as a function of  $n$  is proved by Theorem 4. In these two theorems, the ESA policy is not used (by setting  $m_e = m$ ,  $x_e = 1$  and  $y_e = b$  in Model M).

**THEOREM 3.** *For a fixed  $n$ , with an increase in  $m$ ,*

- (a)  $EDC^{NN}$  decreases,
- (b) the marginal reduction in  $EDC^{NN}$  decreases.

The proof is omitted here.

**THEOREM 4.** *For a fixed  $m$ , with an increase in  $n$ ,*

- (a)  $EDC^{NN}$  decreases,
- (b) The marginal reduction in  $EDC^{NN}$  decreases.

The proof is omitted here.

Theorem 5 below shows that an increase in  $m$  has more effect on reducing  $EDC^{NN}$  than the same increase in  $n$  does.

**THEOREM 5.** *Based on a given  $m$  and  $n$ , an increase in  $m$  brings a larger reduction of  $EDC^{NN}$  than the same amount of increase in  $n$  does.*

The proof is omitted here.

Although Theorem 5 is implicit in the numerical results of some literature, like Han et al. (1987), it has not been explicitly noted before. It can be understood by noticing that an added open location ( $m$ ) can provide one more candidate storage location for all the DCs in the block of DCs, to reduce its travel time. However, an added retrieval can only reduce its own DC travel time by selecting a proper open location while it may increase the total travel time of its previous DCs.

If the ESA policy is used ( $m_e \leq m$ ), we can derive a theorem similar to Theorem 4:

**THEOREM 6.** *The minimal expected DC travel time,  $EDC^*(x_e^*, y_e^*, m_e^*)$ , is a decreasing function of  $n$ .*

The proof is omitted here.

However,  $EDC^*(x_e^*, y_e^*, m_e^*)$  generally is not a convex function of  $n$  because, with an increase in  $n$ ,  $m_e^*$  and the optimal ESA shape factor  $b_e^* = y_e^* / x_e^*$  change simultaneously.

## 5 Numerical Examples

This section conducts numerical experiments to evaluate how much the ESA policy can outperform NN under various combinations of different parameters: the rack shape  $b$ , the system capacity  $C_N$ , the system utilization  $u$ , and the number of retrievals  $n$ .

### 5.1 Experimental Setup

We start with a base example with the parameter values:  $u=0.75$ ,  $b=0.75$ ,  $C_N=1500$ , and  $n=20$ . After that we vary  $C_N$ ,  $b$ ,  $u$ , and  $n$  in the ranges of  $[500, 3000]$ ,  $[0.25, 1]$ ,  $[0.55, 0.95]$ , and  $[1, 100]$ , respectively. These values are based on expert judgments, and cover parameter values used in previous papers (e.g., Han et al., 1987; Lee and Schaefer, 1996). Moreover, we have tested examples with all possible combinations of  $C_N=500, 1000, 1500, 2000, 2500, 3000$ ;  $b=0.25, 0.5, 0.75, 1$ ;  $u=0.55, 0.65, 0.75, 0.85, 0.95$ ; and  $n=1, 5, 10, 20, 30, 50, 100$ .

The algorithm steps are programmed in C++, and run on a DELL D630 notebook with CPU Duo 2.4 GHz, and 2GB of RAM. All results are normalized to a rack area of 1 square seconds by setting  $T = (1 / \sqrt{b})$ .

### 5.2 Results

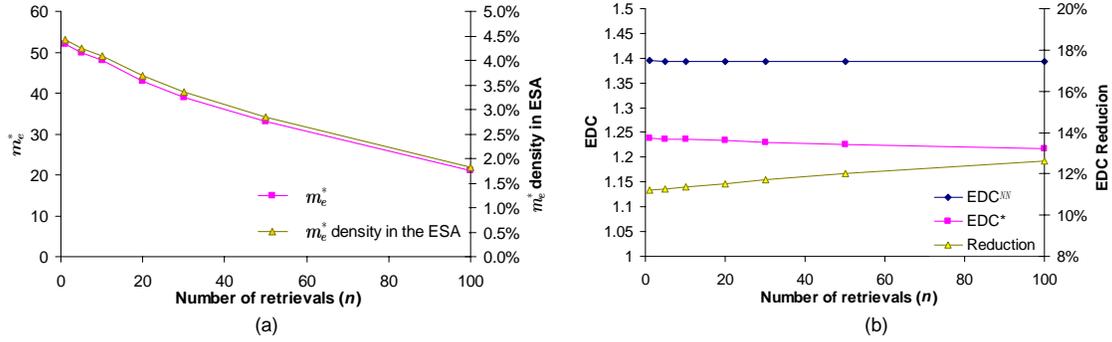
For the base example, the results are  $(x_e^*, y_e^*, m_e^*) = (0.90, 0.87, 43)$ , and  $EDC^* = 1.23$  when using the ESA policy. Using NN yields an expected DC travel time of  $EDC^{NN} = 1.39$ . Hence,  $EDC^*$  outperforms  $EDC^{NN}$  by 11.5%. Obviously, the phenomenon described by Figure 1 (proved by Theorem 3 for the NN policy) does not hold if the ESA policy is used.

The results corresponding to the sensitivity analyses of  $C_N$ ,  $b$ ,  $u$ , and  $n$  are shown in Figures 3-6 respectively. In each figure, the optimal number of open locations  $m_e^*$ ,  $EDC^*$ , and  $EDC^{NN}$  are provided. The computation time of each instance evaluated is within a second.

From Figures 3-6 and the other related results, we make the following observations.

1) Figures 3-6 show the ESA policy reduces the DC travel time significantly. For all the possible combinations we have tested, the ESA policy outperforms NN by 14.5% on average.

2) The optimal number of effective open locations,  $m_e^*$ , depends on  $n$ ,  $C_N$ ,  $b$ , and  $u$ .  $m_e^*$  decreases with an increase in  $n$  while it increases with an increase in  $C_N$ , and  $b$ .

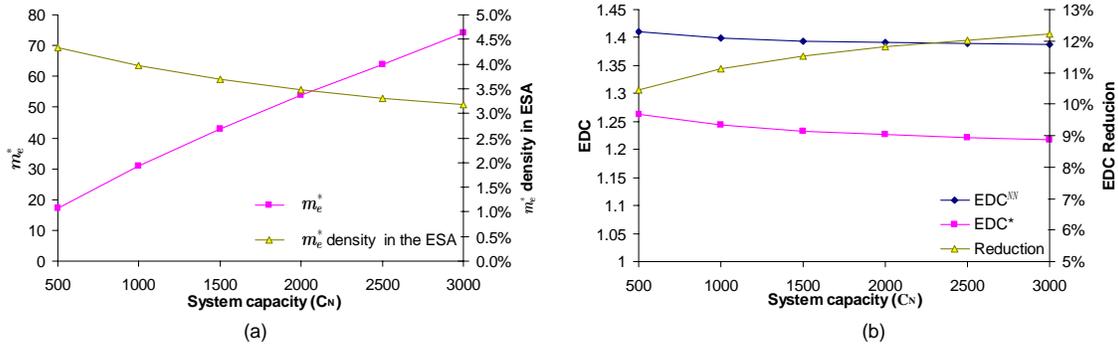


Note.  $m_e^*$  density in the ESA =  $(m_e^*/(\text{the capacity of the ESA in number of unit loads})) \times 100\%$ .

Figure 3: Influence of the Number of Retrievals (i.e. Block Size  $n$ ) on  $m_e^*$  and  $EDC$

3) Open locations have more impact in reducing the average travel time than the same number of retrievals; In Figure 3, the values of  $EDC^*$  at  $(n=1, m=52)$  and  $(n=100, m=21)$  are almost the same, which means that the contribution of  $100-1=99$  retrievals in reducing  $EDC$  can be approximately replaced by  $52-21=31$  open locations if the ESA policy is applied. To some extent, this can be explained by Theorem 5.

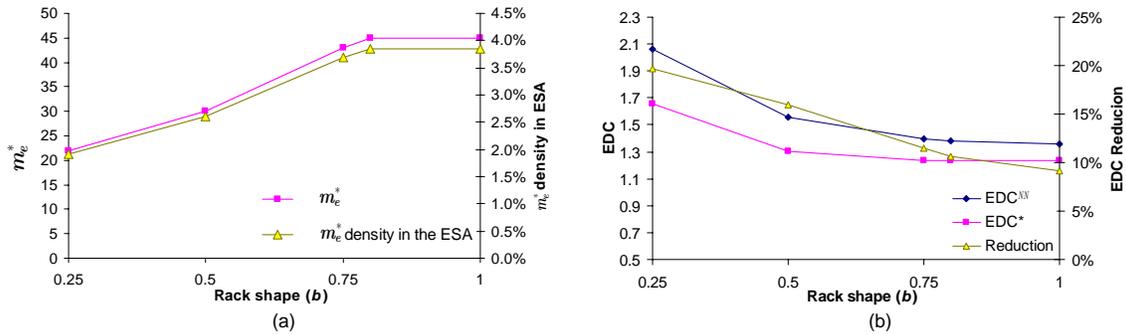
4)  $EDC^*$  is quite insensitive to the number of retrievals  $n$  (i.e., the block size of DCs) although Theorem 6 holds. Figure 3(b) shows  $EDC^*$  decreases by less than 1% if  $n$  changes from 1 to 100. Executing retrievals one by one (i.e.  $n=1$ ) is only slightly worse than cleverly sequencing them in a block size as large as 100. A similar phenomenon happens to  $EDC^{NN}$  in spite of Theorem 4. However, in many past papers (e.g., Han et al., 1987), it is stated that an increase in number of retrievals or a good sequence of retrievals can bring a significant reduction in the DC travel time. The “contradiction” states that for large  $m$  there is a much smaller marginal effect of increasing  $n$  than for smaller values of  $m$ . In those previous papers fewer open locations are tested (mostly less than 15) than the number ( $m_e$ ) in the ESA policy (between 21 and 52 in Figure 3(a)). In practice,  $m$  normally is more than 50 even for a system with a high space utilization of 90% and low capacity of 500 unit loads. In conclusion, the larger number of open locations in our cases makes increasing retrievals have little impact on reducing  $EDC$ .



Note.  $m_e^*$  density in the ESA =  $(m_e^*/(\text{the capacity of the ESA in number of unit loads})) \times 100\%$ .

Figure 4: Influence of System Capacity ( $C_N$ ) on  $m_e^*$  and EDC

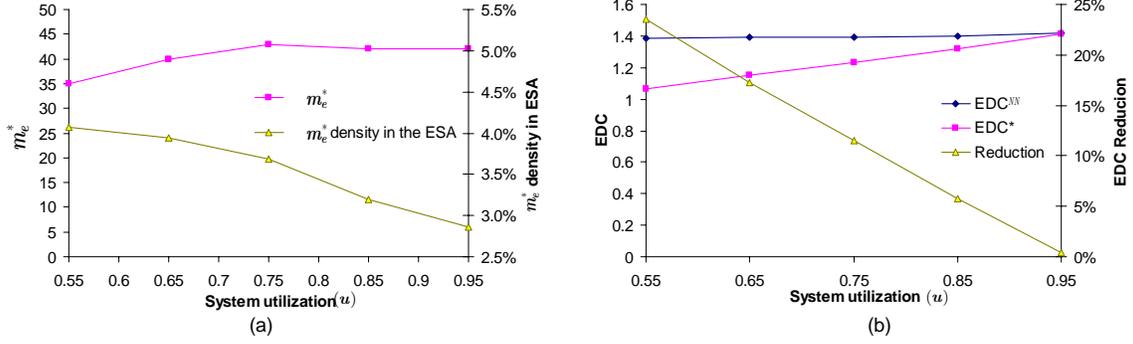
5) With an increase in the system capacity ( $C_N$ ), the DC travel time reduction of the ESA policy over NN increases. In Figure 4(b), it can be seen that the reduction increases from 10.5% at  $C_N=500$  to 12.2% at  $C_N=3000$ . The reason can be found in Figure 4(a); with increasing  $C_N$ , the density of  $m_e^*$  in the ESA decreases from 4.3% at  $C_N=500$  to 3.2% at  $C_N=3000$ . This leads to a relatively smaller size of the ESA, and contributes to the increase of the reduction.



Note.  $m_e^*$  density in the ESA =  $(m_e^*/(\text{the capacity of the ESA in number of unit loads})) \times 100\%$ .

Figure 5: Influence of Rack Shape ( $b$ ) on  $m_e^*$  and EDC

6) Figure 5 shows the performance of the ESA policy is less sensitive to a change of the rack shape factor  $b$  compared with that of NN. Moreover, the skewer (smaller  $b$ ) the rack shape is, the larger improvement the ESA policy obtains over NN (see Figure 5(b)).



Note.  $m_e^*$  density in the ESA =  $(m_e^*/(\text{the capacity of the ESA in number of unit loads})) \times 100\%$ .

Figure 6: Influence of System Utilization ( $u$ ) on  $m_e^*$  and EDC

7) Figure 6 shows the reduction of the DC travel time of  $EDC^*$  over  $EDC^{NN}$  highly depends on the system utilization ( $u$ ). If the system utilization  $u=0.55$ , the reduction reaches 23.5%. However, when  $u=0.95$ , the reduction is less than 1%. For common values of  $u$  between 0.65 and 0.85, the reductions are between 6% and 17%, which become larger if  $b$  becomes smaller.

8)  $m_e^*$  is not very sensitive to changes in  $u$ . Figure 6(a) shows increasing  $u$  from 0.75 to 0.95 does not cause an increment of  $m_e^*$  at all. This is due to the combined effect of the decrease in  $y_e^*/x_e^*$  (the shape of the ESA) and the increase of the ESA size. From observation 4 we know decreasing  $y_e^*/x_e^*$  reduces  $m_e^*$ . On the other hand, increasing the size of the ESA increases  $m_e^*$ . The two combined effects obviously outweigh each other for a great deal.

## 6 Implementing the ESA Policy in Practice

Figure 3(b) shows  $EDC^*$  is rather insensitive to the size of  $n$ :  $EDC^*$  at  $n=1$  hardly differs from  $EDC^*$  at  $n=50$ . This leads to a valuable suggestion for implementing the ESA policy in practice:

*Fix the block size  $n=1$ , and the number of effective open locations,  $m_e$  to be  $m_e^*$  (at given  $n=1$ ).*

With this we can obtain a near-optimal solution of the ESA policy, while the implementation becomes much easier as the number of effective open locations in the ESA is fixed regardless of the block size. We can even handle retrievals in block sizes of 1, for example by retrieving them in sequence of urgency, without noticeable impact on the expected DC travel time.

The suggestion is very easy to implement as  $n=1$ . Still, according to Theorem 4, a larger block size can reduce the DC travel time further. We can simply dynamically

change  $n$ , in line with the queue length of waiting storage and retrieval jobs. We therefore suggest the following implementation.

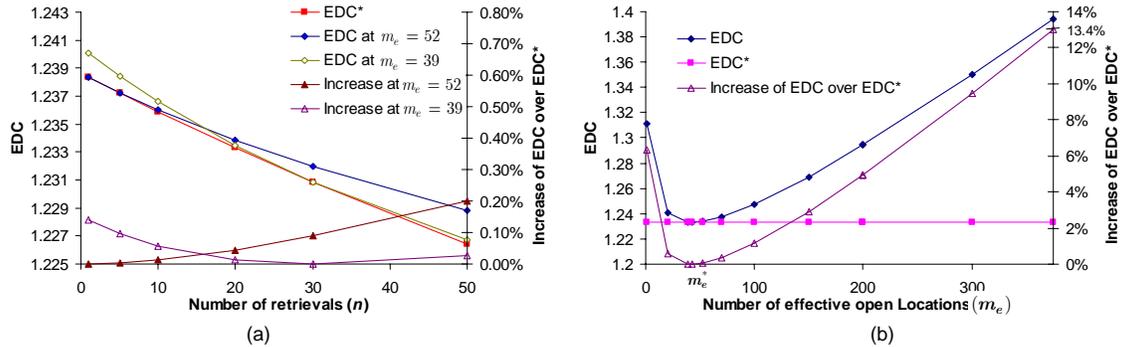
*Keep the block size  $n$ , equal to the queue length of storage and retrieval jobs, and fix the number of effective open locations ( $m_e$ ) to be  $m_e^*$  for  $n=1$ .*

For the base example in Section 5, the first suggestion obtains a near optimal  $EDC$  solution with maximum gaps of 0.97% from the optimum for  $n=50$ , where the “gap” is the gap between  $EDC^*$  at  $n=50$  and  $EDC^*$  at  $n=1$ . The gap becomes 0.2% by using the second suggestion.

We have tested the above suggestions for a wide variety of possible combinations of  $C_N$ ,  $b$ ,  $u$ , and  $n$  (see Section 5.1) with similar results.

These suggestions show the first assumption in Subsection 3.1 can be relaxed. We do not have to fix the block size beforehand, but we can dynamically adapt it with close to optimal results.

In line with past research (e.g., Graves et al., 1977; Han et al., 1987), we so far have only considered DC modes. In this mode, once the ESA has been created, it can be maintained automatically. However, in practice we may be forced to carry out single-command cycles sometimes. The system utilization and then  $(m_e^*, x_e^*, y_e^*)$  may have to be changed correspondingly. In case of storage jobs only,  $m_e$  drops below  $m_e^*$ , and in case of only retrieval jobs  $m_e$  exceeds  $m_e^*$ . To see what the impact is of this is, we deliberately let  $m_e$  deviate from  $m_e^*$  within a given range between  $m_e^*$  at  $n=1$  (i.e.,  $m_e^*=52$ ), and  $m_e^*$  at  $n=30$  (i.e.,  $m_e^*=39$ ). Figure 7(a) shows  $EDC$  increases by less than 0.2% for the base example. We have tested all the combinations of  $C_N$ ,  $b$ ,  $u$ , and  $n$  (given in Section 5.1). The results show that  $EDC$  deviates from  $EDC^*$  by less than 1% for  $C_N \geq 1000$ , and for most cases of  $C_N=500$ . Therefore, we can dynamically change  $(m_e, x_e, y_e)$ , while  $EDC$  deviates little from  $EDC^*$ . This small deviation of  $EDC$  from  $EDC^*$  can be explained by looking at Figure 7(b). The difference becomes large only when  $m_e$  is quite far from  $m_e^*$ .



Note.  $m_e=52$  and 39 correspond to the optimal  $m_e$  at  $n=1$  and 30, respectively;  $EDC$  represents the result by substituting a fixed  $m$  and the results of Theorem 1 into Equation (1); “Increase at  $m_e=39$ ” represents the increase of  $EDC$  at  $m_e=39$  over  $EDC^*$ .

Figure 7: Influence of the Number of Effective Open Locations ( $m_e$ ) on  $EDC$

If the number of effective open locations deviates too far from the optimum, the optimal situation can be restored in idle periods by the automated S/R machine.

In conclusion, the implementation of the ESA policy is quite easy and flexible. With a change of  $n$  and  $u$ , the number of effective open locations ( $m_e$ ), and the boundary,  $x_e \times y_e$ , of the ESA can be well managed without extra effort.

## 7 Conclusions and Further Research

To our knowledge, this paper is the first to focus on modeling the management of open locations in warehouses. We propose the ESA policy to manage open locations for minimizing the cycle travel time of a block of storage and retrieval jobs. A model, some properties and an algorithm are developed to determine the optimal number of open locations, and the boundary of the ESA. From the results of the paper, we obtain the following managerial insights:

- The ESA policy can outperform nearest neighbor (NN) by between 17% and 6% of the cycle travel time for realistic rack utilizations between 65 and 85%. Savings can be more than 20% if the system utilization is less than 65%.
- The optimal solution of the model depends on system capacity, system shape, and rack utilization. Our algorithm can determine the optimal number of open locations and the ESA boundary within a second for any real practical system size on an ordinary computer.
- Application of the ESA policy is fairly easy by using the suggestions in section 6. The suggestions make it possible to apply the ESA policy under varying conditions, even when the job queue changes or when there are only single commands to be performed.

We obtain several findings differing from those in previous research.

- The block size,  $n$ , of storage and retrieval jobs normally has little influence on the average DC travel time. However, all previous papers (e.g., Han et al., 1987; Lee and Schaefer, 1996; Mahajan et al., 1998) demonstrate that a large block size can significantly reduce the DC travel time. This can be explained as in these papers, due to calculation complexity, the number of open locations has only been evaluated for small values, e.g.  $<10$ . In practice the number of open locations in such a system is usually much larger. A typical aisle in an AS/RS system of about 70 pallets long and 10 pallets high contains 1400 pallets; with 95% rack utilization this implies 70 empty slots. The optimal number of effective open locations is normally less than 40 in such a case.
- Changing the sequence of retrieval jobs normally has little influence on the DC travel time. As sufficient effective open locations are readily available in practice, our research, supported by Theorems 5&6 and numerical examples, shows that NN reduces the DC travel time mainly due to the smart selection of open locations, rather than cleverly sequencing retrievals.

The above differences greatly reinforce the advantages of the ESA policy. According to our first suggestion in Section 6 it is not necessary to intentionally lengthen the queue of retrievals in practice, especially as accumulating retrievals needs waiting time. The policy can be used dynamically by considering new incoming jobs and job urgency with little impact on the travel time.

This paper may trigger a new sub-research area in warehousing and distribution science to study the management of open locations. Based on this paper, further research directions at least include:

- Open location management in different warehousing systems: AS/RSs with multiple shuttles, end-of-aisle AS/RS systems, carousel systems, and conventional multi-aisle systems.
- Open location management under different storage policies: class-based storage, dedicated storage, duration-of-stay (DOS), etc.
- Combinations of open location management with S/R machine dwell point selection, product prepositioning, etc.
- The possible effects of open location management on different warehousing system designs, such as system dimensions, and depot selection problems.

Finally, with the increasing application of compact 3D storage warehouses (Yu and De Koster, 2009b), open locations can also be managed for reducing the total cycle travel time in such systems. However, because open locations normally change dynamically during storage and retrieval processes in these systems, the related research could be much more challenging than in a conventional warehousing system.

## Appendices:

### Appendix A. Proof of Lemma 1

To prove  $B(k+1) - B(k)$ , we first prove:  $\bar{g}_{k+1}(z) \geq \bar{g}_k(z)$  for  $0 \leq z \leq F_e^{-1}(1/(k+1))$ , and  $\bar{g}_{k+1}(z) \leq \bar{g}_k(z)$  for  $F_e^{-1}(1/(k+1)) \leq z \leq 1$ . (18)

by changing  $k$  to  $k+1$ , we have  $\bar{g}_{k+1}(z) - \bar{g}_k(z) = (k+1)(1 - F_e(z))^k f_e(z) - k(1 - F_e(z))^{k-1} f_e(z) = [(k+1)(1 - F_e(z)) - k] \cdot (1 - F_e(z))^{k-1} f_e(z)$ . (19)

Letting  $\bar{g}_{k+1}(z) - \bar{g}_k(z) = 0$ , we have  $z = F_e^{-1}(1/(k+1))$  where  $0 < F_e^{-1}(1/(k+1)) < 1$  as  $0 < 1/(k+1) < 1$ .

In Equation (19), because  $f_e(z)(1 - F_e(z))^{k-1} > 0$ , and  $(1 - F_e(z))$  is a decreasing function of  $z$ , we have

$$\bar{g}_{k+1}(z) - \bar{g}_k(z) \begin{cases} \geq 0 & \text{if } 0 \leq z \leq a \\ < 0 & \text{if } a < z \leq 1 \end{cases} .$$

where  $a = F_e^{-1}(1/(k+1))$ . Therefore (a) has been proved.

With Equation (18),

$$\begin{aligned} B(k+1) - B(k) &= \int_0^1 u(z) \bar{g}_{k+1}(z) dz - \int_0^1 u(z) \bar{g}_k(z) dz = \int_0^1 u(z) [\bar{g}_{k+1}(z) - \bar{g}_k(z)] dz \\ &= \iint_{R_1} u(z) dy dz - \iint_{R_2} u(z) dy dz \text{ where } R_1 = \{(y, z) | 0 \leq z \leq a, \bar{g}_k(z) \leq y \leq \bar{g}_{k+1}(z)\}, \\ &R_2 = \{(y, z) | a \leq z \leq 1, \bar{g}_{k+1}(z) \leq y \leq \bar{g}_k(z)\}. \end{aligned}$$

Because  $u(z)$  is an increasing function of  $z$ , we have

$$\iint_{R_1} u(z) dy dz \leq \iint_{R_1} u(a) dy dz, \text{ and } \iint_{R_2} u(a) dy dz \leq \iint_{R_2} u(z) dy dz. \quad (20)$$

Where “=” only holds for  $u(z)$  being a constant.

Because  $\bar{g}_k(z)$  (i.e., Equation (8)) is a PDF,

$$\int_0^1 \bar{g}_k(z) dz = \int_0^1 k[1 - F_e(z)]^{k-1} f_e(z) dz = 1, \quad \forall k \text{ is a positive integer,} \quad (21)$$

we have  $\iint_{R_1} dy dz = \iint_{R_2} dy dz$ . As  $u(a)$  is a constant, we have

$$\iint_{R_1} u(a) dz dy = \iint_{R_2} u(a) dz dy. \quad (22)$$

Using Equations (20) and (22), we have  $\iint_{R_1} u(z) dz dy \leq \iint_{R_2} u(z) dz dy$ . That is

$B(k+1) - B(k) \leq 0$  (“=” only holds for  $u(z)$  being a constant) and Lemma 1 has been proved.

### Appendix B. Proof of LEMMA 2

To keep the ESA equally sized, while changing its shape, without loss of generality we normalize the ESA area to have size=1 by setting  $x_e = 1/\sqrt{b_e}$  and  $y_e = \sqrt{b_e}$ .

According to Bozer and White (1984), for a storage area  $(1/\sqrt{b_e}) \times \sqrt{b_e}$ ,

$$ESC(x_e, y_e, m_e) = \frac{2}{\sqrt{b_e}} \left( \int_0^{b_e} \frac{2z^2}{b_e} dz + \int_{b_e}^1 z dz \right).$$

Next, from Equations (7)-(10), for a storage area  $(1/\sqrt{b_e}) \times \sqrt{b_e}$  we have

$$E(Z_k^{\min}(x_e, y_e, m_e)) = x_e \int_0^{b_e} P_A dz + x_e \int_{b_e}^1 P_B dz. \quad (23)$$

$$\text{where } P_A = \frac{2kz^2(4b_e - 3(1+b_e)z + 2z^2)((b_e^2 + 2b_e(-2+z)z^2 + (2-z)z^2)/b_e^2)^k}{b_e^2 - 4b_e z^2 + 2(1+b_e)z^3 - z^4},$$

$$\text{and } P_B = 2k(1-z)^{2k-1}z.$$

$$ETB(x_e, y_e, m_e) = \left( \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \int_0^{b_e} P_A dz + \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \int_{b_e}^1 P_B dz \right) / \sqrt{b_e}.$$

$EDC(x_e, y_e, m_e) = ESC(x_e, y_e, m_e) + ETB(x_e, y_e, m_e)$  can now be expressed as a function of  $b_e$  (denoted as  $EDC(b_e)$ ). We have

$$EDC(b_e) = \int_0^{b_e} \left( \frac{4z^2}{b_e \sqrt{b_e}} + \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \frac{P_A}{\sqrt{b_e}} \right) dz + \int_{b_e}^1 \left( \frac{2z}{\sqrt{b_e}} + \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \frac{P_B}{\sqrt{b_e}} \right) dz.$$

Lemma 2 can be proved by demonstrating that  $EDC(b_e)$  is a decreasing function of  $b_e$  (i.e.,  $dEDC(b_e)/db_e \leq 0$  with  $0 < b_e \leq 1$ ) which is intuitive and its proof can be found in Yu and De Koster (2009a). Therefore Lemma 2 has been proved.

## Appendix C. Proof of THEOREM 2

From Equations (14)-(17), the objective function (1), as a function of  $m_e$ , becomes

$$EDC(m_e) = ESC(m_e) + ETB(m_e).$$

We can prove Theorem 2 by demonstrating that  $dEDC(m_e)/dm_e$  either is a monotonous function of  $m_e$ , or has a unique global minimum. Without loss of generality, we assume  $m_e$  is a continuous variable,  $1 \leq m_e \leq m$  here. The theorem can be proved in three Steps. Step 1 calculates  $dEDC(m_e)/dm_e$  and analyzes some properties of Model M1 for further analysis; Steps 2, 3 and 4 prove that Model M1, M2 and then M have the property in Theorem 2 respectively.

**Step 1.** For Model M1, the ESA is SIT, and we have

$$\frac{dESC(m_e)}{dm_e} = \frac{d(4x_e^*(m_e)/3)}{dx_e} \cdot \frac{d(\sqrt{(m_e + C_N \cdot u)b/C_N})}{dm_e} = \frac{2b}{3x_e^*(m_e)C_N},$$

$$\text{For } ETB(m_e), \text{ we have } \frac{dETB(m_e)}{dm_e} = \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \int_0^1 \frac{d(x_e^*(m_e)zk[1-F_e(z)]^{k-1}f_e(z))}{dm_e} dz$$

$$\begin{aligned}
&= \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \left\{ \int_0^1 \frac{d(x_e^*(m_e))}{dm_e} \cdot zk[1-F_e(z)]^{k-1} f_e(z) dz + \int_0^1 x_e \frac{d(k)}{dm_e} z[1-F_e(z)]^{k-1} f_e(z) dz \right. \\
&\quad \left. + \int_0^1 x_e^*(m_e) k \cdot \frac{d(z[1-F_e(z)]^{k-1} f_e(z))}{dm_e} dz \right\} \\
&= \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \left\{ \int_0^1 \frac{d(x_e^*(m_e))}{dm_e} zk[1-F_e(z)]^{k-1} f_e(z) dz + \int_0^1 x_e^*(m_e) z[1-F_e(z)]^{k-1} f_e(z) dz \right. \\
&\quad \left. + \int_0^1 x_e^*(m_e) zk[1-F_e(z)]^{k-1} f_e(z) \ln(1-F_e(z)) dz \right\},
\end{aligned}$$

$$\text{where } \frac{dx_e^*(m_e)}{dm_e} = \frac{b}{2C_N \sqrt{b(m_e + C_N \cdot u) / C_N}} > 0.$$

Therefore we have  $\frac{dEDC(m_e)}{dm_e} = Part1 + Part2 + Part3$ , where  $Part1 = \frac{2b}{3x_e^*(m_e)C_N}$ ,

$$Part2 = \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \left\{ \int_0^1 \frac{d(x_e^*(m_e))}{dm_e} zk[1-F_e(z)]^{k-1} f_e(z) dz + \int_0^1 x_e^*(m_e) z[1-F_e(z)]^{k-1} f_e(z) dz \right\}, \text{ and}$$

$$Part3 = \frac{1}{n} \sum_{k=m_e}^{n+m_e-1} \int_0^1 x_e^*(m_e) zk[1-F_e(z)]^{k-1} f_e(z) \ln(1-F_e(z)) dz. \text{ Unfortunately, } dETB(m_e)/dm_e \text{ is}$$

not analytically integrable. We therefore analyze the properties of these three parts for the further proof.

$Part1 > 0$  as  $b > 0, x_e > 0, C_N > 0$ . Moreover, it converges to 0 *polynomially* with the increase in  $m_e$  as  $2b / (3x_e C_N)$  is positive with  $x_e$  given by Equation (11).

$Part2 > 0$  as  $f_e(z) \geq 0, 1-F_e(z) \geq 0, x_e^*(m_e) > 0$  and  $\frac{dx_e^*(m_e)}{dm_e} > 0$  where all the “=” can

only hold at the extreme points of  $z=0$  or 1. Moreover, it equals 0 if  $z=0$ . And if  $z < 1$   $Part2$  converges to 0 *exponentially* with the increase in  $m_e$  as  $m_e$  is on the power of  $1-F_e(z)$  and  $0 \leq 1-F_e(z) < 1$ .

$Part3 < 0$  as  $f_e(z) \geq 0, 0 \leq 1-F_e(z) \leq 1$ , and  $\ln(1-F_e(z)) \leq 0$  where all the “=” can only hold at the extreme points of  $z=0$  or 1. Moreover,  $Part3$  equals 0 if  $z=0$ . And if  $z < 1$   $Part3$  converges to 0 *exponentially* with the increase in  $m_e$  as  $m_e$  is on the power of  $1-F_e(z)$  and  $0 \leq 1-F_e(z) < 1$ .

**Step 2.** There are three possible cases for the result of  $dEDC(m_e)/dm_e$ ; Case 2.1:  $dEDC(m_e)/dm_e > 0$ , Case 2.2:  $dEDC(m_e)/dm_e < 0$ , and Case 2.3: Otherwise.

*Case 2.1.* In this case,  $EDC(m_e)$  reaches its minimum at  $m_e = 1$ .

*Case 2.2.* if  $dEDC(m_e)/dm_e < 0$  for  $m_e = 1, 2, \dots, m$ , then  $EDC(m_e)$  reaches its minimum at  $m_e = m$ .

*Case 2.3.* There is at least one  $m_e$  satisfying  $dEDC(m_e)/dm_e = 0$ . Denote the smallest  $m_e$  satisfying  $dEDC(m_e)/dm_e = Part1 + Part2 + Part3 = 0$  as  $m_0$ . That is  $Part1 + Part2 = |Part3|$  at  $m_0$ . From the above analysis, with the increase in  $m_e$ ,

$Part1 + Part2$  converges to 0 *polynomially* with the increase in  $m_e$  as  $Part1$  converges to 0 *polynomially* and  $Part2$  converges to 0 *exponentially*. Meanwhile,  $Part3=0$  for  $z=0$ , or converges to 0 *exponentially* with the increase in  $m_e$  for  $z>0$ . It means that ,with increasing  $m_e$  ,once  $Part1 + Part2 = |Part3|$  at  $m_0$  ,  $|Part3|$  will become less than  $Part1 + Part2$  (converging to 0 *polynomially*) at  $m_e > m_0$  . That is, for  $m > m_0$  ,  $Part1 + Part2 - |Part3| = Part1 + Part2 + Part3 = dEDC(m_e)/dm_e$  will be positive, and converges to 0. Moreover, as  $dEDC(m_e)/dm_e > 0$  for  $m_e > m_0$  , and  $m_0$  is the smallest to make  $dEDC(m_e)/dm_e = 0$  at  $m_e = m_0$  , we have  $dEDC(m_e)/dm_e \leq 0$  for  $m_e < m_0$  .

Therefore we have, for Model M1,  $EDC(m_e)$  has a unique global minimum. It is a decreasing function of  $m_e$  if  $m_e$  is less than  $m_0$  , and an increasing function of  $m_e$  if  $m_e$  is larger than  $m_0$  .

**Step 3.** For Model M2, using a similar process, we can prove the property for Model M2. Note that for the proof,  $Part3$  will become more complex as  $b_e$  will not be 1, and is a function of  $m_e$  . However, it does not change the property that  $Part3=0$  or converges to 0 *exponentially* with the increase in  $m_e$  .

**Step 4.** From the result of Steps 2&3, if  $C_N(b-u) \leq 1$  or  $C_N(b-u) \geq m$  , Model M1 or M2 is equivalent to Model M. The properties in Theorem 2 hold for Model M. Otherwise,  $1 < C_N(b-u) < m$  . In this case,  $m_e = C_N(b-u)$  is the intersection point (denoted as  $m_i$  ) of the objective functions of Model M1 & M2. At  $m_e = m_i$  , there are four possible cases for  $dEDC(m_e)/dm_e$  . Case 4.1:  $dEDC(m_e)/dm_e \leq 0$  for both Model M1&M2, Case 4.2:  $dEDC(m_e)/dm_e \geq 0$  for both Model M1&M2, Case 4.3:  $dEDC(m_e)/dm_e \geq 0$  for Model M1, but  $\leq 0$  for Model M2, and Case 4.4:  $dEDC(m_e)/dm_e \leq 0$  for Model M1, but  $\geq 0$  for Model M2.

*Case 4.1.* For Model M1, if  $dEDC(m_e)/dm_e \leq 0$  at  $m_e = C_N(b-u)$  , according to the proof in Step 2, we have  $dEDC(m_e)/dm_e \leq 0$  for all  $m_e \leq C_N(b-u)$  .  $EDC(m_e)$  obtains its minimum at  $m_e = C_N(b-u)$  .

For Model M2, because  $dEDC(m_e)/dm_e \leq 0$  at  $m_e = C_N(b-u)$  and  $EDC(m_e)$  is a continuous function of  $m_e$  for  $m_e \in [1, m]$  , the minimal value  $EDC(m_e)$  of Model M1 is only a feasible solution of Model M2. With the result in Step 3, the unique global optimal value  $m_e$  of Model M2 will become  $m_e^*$  of Model M.

*Case 4.2.* For Model M2, according to the proof in Step 2,  $dEDC(m_e)/dm_e \geq 0$  for all  $m_e \geq C_N(b-u)$  , and  $EDC(m_e)$  gets the minimum at  $m_e = C_N(b-u)$  . For Model M1, because  $dEDC(m_e)/dm_e \geq 0$  at  $m_e = C_N(b-u)$  and  $EDC(m_e)$  is a continuous function of  $m_e$  for  $m_e \in [1, m]$  , the minimal value  $EDC(m_e)$  of Model M2 is only a feasible solution of Model M1. With the result in Step 2, the unique global optimal value  $m_e$  of Model M1 will become the optimal value  $m_e^*$  of Model M.

*Case 4.3.* We prove that this case does not exist by reduction to absurdity. For Model M1,  $dEDC(m_e)/dm_e \geq 0$  at  $m_e = C_N(b-u)$ . According to the proof in Case 2, if  $m_e$  increases further to an  $m_e$  while the ESA shape still keeps in SIT,  $EDC(m_e)$  would increase to a high value (denoted as  $EDC'$ ). Moreover, according to Lemma 2, this value is less than any other value of  $EDC(m_e)$  in NSIT, which corresponds to a solution of Model M2. Denoting the NSIT value of Model M2 as  $EDC''$ , we obtain  $EDC'' > EDC' > EDC(m_e)|_{m_e=C_N(b-u)}$ , which contradicts  $dEDC(m_e)/dm_e \leq 0$  for Model M2. We therefore can eliminate this case.

*Case 4.4.* For Model M1, if  $dEDC(m_e)/dm_e \leq 0$  at  $m_e = C_N(b-u)$ , according to the proof in Step 2, we have  $dEDC(m_e)/dm_e \leq 0$  for all  $m_e \leq C_N(b-u)$ . Similarly, for Model M2,  $dEDC(m_e)/dm_e \geq 0$  for all  $m_e \geq C_N(b-u)$ . We therefore have  $m_e = C_N(b-u)$  is the global optimal solution  $m_e^*$  of Model M.

Summarizing the cases 4.1-4.4, we obtain that  $EDC(m_e)$  is a decreasing function of  $m_e$  if  $m_e$  is less than  $m_e^*$ , and an increasing function of  $m_e$  if  $m_e$  is larger than  $m_e^*$ . Therefore Step 4 is done and Theorem 2 has been proved.

## References

- [1] Bozer, Y. A. and White, J. A. (1984), "Travel-time models for automated storage/retrieval systems," *IIE Transactions*, 16, 329-338.
- [2] Bozer, Y. A. and White, J. A. (1990), "Design and performance models for end-of-aisle order picking systems," *Management Science*, 36(7), 852-866.
- [3] De Koster, R., Le-Duc, T. and Roodbergen, K. J. (2007), "Design and control of warehouse order picking: A literature review," *European Journal of Operational Research*, 182(2), 481-501.
- [4] Dooly, D. R., and Lee, H. F. (2008), "A shift-based sequencing method for twin-shuttle automated storage and retrieval systems," *IIE Transactions*, 40(6), 586-594.
- [5] Eben-Chaime, M. (1992), "Operations Sequencing in Automated Warehousing Systems," *International Journal of Production Research*, 30(10), 2401-2409.
- [6] Eynan, A. and Rosenblatt, M. J. (1993), "An Interleaving Policy in Automated Storage-Retrieval Systems," *International Journal of Production Research*, 31(1), 1-18.
- [7] Graves, S. C., Hausman, W. H. and Schwarz, L. B. (1977), "Storage-retrieval interleaving in automatic warehousing systems," *Management Science*, 23, 935-945.

- [8] Gu, J. X., Goetschalckx, M. and McGinnis, L. F. (2007), "Research on warehouse operation: A comprehensive review," *European Journal of Operational Research*, 177(1), 1-21.
- [9] Han, M. H., McGinnis, L. F., Shieh, J. S., and White, J. A. (1987), "On sequencing retrievals in an automated storage/retrieval system," *IIE Transactions*, 19(1), 56-66.
- [10] Hausman, W. H., Schwarz, L. B., and Graves, S. C. (1976), "Optimal storage assignment in automatic warehousing systems," *Management Science*, 22(6), 629-638.
- [11] Johnson, M. E. and Meller, R. D. (2002), "Performance analysis of split-case sorting systems," *Manufacturing & Service Operations Management*, 4(4), 258-274.
- [12] Lee, H. F. and Schaefer, S. K. (1996), "Retrieval sequencing for unit-load automated storage and retrieval systems with multiple openings," *International Journal of Production Research*, 34(10), 2943-2962.
- [13] Lee, H. F. and Schaefer, S. K. (1997), "Sequencing methods for automated storage and retrieval systems with dedicated storage," *Computers & Industrial Engineering*, 32(2), 351-362.
- [14] Mahajan, S., Rao, B. V. and Peters, B. A. (1998), "A retrieval sequencing heuristic for miniload end-of-aisle automated storage/retrieval systems," *International Journal of Production Research*, 36(6), 1715-1731.
- [15] Meller, R. D. and Mungwattana, A. (1997), "Multi-shuttle automated storage/retrieval systems," *IIE Transactions*, 29(10), 925-938.
- [16] Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. (2007), *Numerical Recipes: The Art of Scientific Computing (3 ed.)*, Cambridge: Cambridge University Press.
- [17] Sarker, B. R., Sabapathy, A., Lal, A. M. and Han, M. H. (1991), "Performance evaluation of a double shuttle automated storage and retrieval system," *Production Planning & Control*, 2(3), 207-213.
- [18] Schwarz, L. B., Graves, S. C. and Hausman, W. H. (1978), "Scheduling policies for automatic warehousing systems: simulation results," *IIE Transactions*, 10(3), 260-270.
- [19] Tompkins, J. A., White, J. A., Bozer, Y. A. and Tanchoco, J. M. A. (2003), *Facilities Planning*, John Wiley and Sons, New York.

- [20] Van den Berg, J. P. (1999), "A literature survey on planning and control of warehousing systems," *IIE Transactions*, 31(8), 751-762.
- [21] Van den Berg, J. P. and Gademann, A. J. R. M. (1999), "Optimal routing in an automated storage/retrieval system with dedicated storage," *IIE Transactions*, 31(5), 407-415.
- [22] Yu, Y. and De Koster, R. B. M. (2009a), Open location management in automated warehousing systems. *working paper, Erasmus University*, accessible at <http://hdl.handle.net/1765/14615>.
- [23] Yu, Y. and De Koster, R. B. M. (2009b), "Optimal zone boundaries for two-class-based compact three-dimensional automated storage and retrieval systems," *IIE Transactions*, 41(3), 194-208.