Int J Adv Manuf Technol (2016) 86:2219–2229
DOI 10.1007/s00170-016-8340-y

ORIGINAL ARTICLE

A travel time model for order picking systems in automated warehouses

Yacob Khojasteh[1] · Jae-Dong Son[2]

Received: 18 February 2015 / Accepted: 8 November 2015 / Published online: 21 January 2016

Abstract This paper addresses an order picking problem in a multi-aisle automated warehouse, in which a single storage/retrieval (S/R) machine performs storage and retrieval operations. When retrieval requests consist of multiple items and the items are in multiple stock locations, the S/R machine must travel to several storage locations to complete a customer order. The objective is to minimize the total time traveled by the S/R machine to complete the retrieval process of customer orders at the shortest time. First, we formulate the problem as a nonlinear programming model. Then, we propose a heuristic to solve the problem. Finally, we provide numerical experiments to evaluate the performance of the proposed heuristic. The results show that as the number of items in customer order increases, the heuristic shows a better performance by obtaining solutions close to optimal but in very small amount of times.

Keywords Multi-aisle automated warehouse · Order picking · Travel time analysis · Performance analysis

1 Introduction

The warehousing system plays an important role in the successful implementation of a supply chain. Automated warehouses are widely used in manufacturing, warehousing, and distribution applications. Automated storage and retrieval systems (AS/RSs) as basic components of automated warehouses are used in manufacturing, warehousing, and distribution applications. An AS/RS usually consists of one or more aisles; each aisle has storage racks on either side, a storage and retrieval (S/R) machine, input/output (I/O) stations, and accumulating conveyors. The S/R machine can simultaneously move in vertical and horizontal directions and access the storage rack on either side of the aisle. The number of S/R machines depends on the throughput capacity. When throughput capacity is high, one S/R machine should be employed in each storage aisle.

One important operational aspect of the AS/RSs is to minimize the total time/distance traveled by the S/R machine to complete the retrieval process of customer orders. Warehouse managers are interested in finding the most economical way of picking orders, which minimizes the costs involved in terms of travel distance or travel time. Order picking, which is a fundamental component of the retrieval function performed in warehouses, is a process by which products are retrieved from specified storage locations with respect to customer orders. Order picking represents only a subset of the material handling operations performed in warehousing. Improving the performance of order picking generally can lead to significant saving in warehousing cost [21].

Effective sequencing of the retrievals results in improvements in the overall throughput of the AS/RS. The list of retrievals is continuously changing over time. Performed retrievals are deleted from the list and new retrieval orders are added. Han et al. [5] suggested two ways to deal with this dynamic problem. Firstly, select a block of the most urgent storage and retrieval requests, sequence them, and when they are completed, select the next block, and so on. This is called *block sequencing*. Secondly, we can resequence the whole list of requests every time a new request is added and use due

✉ Jae-Dong Son
son88@ssu.ac.kr

1 Graduate School of Global Studies, Sophia University, Chiyoda-ku, Tokyo 102-8554, Japan

2 Department of Industrial and Information Systems Engineering, Soongsil University, Dongjak-Gu, Seoul 156-743, Korea

times or priorities. This kind of sequencing is called *dynamic sequencing*. The performance of both approaches differs. For example, Eben-Chaime [3] concluded that in a specified non-deterministic environment, the block sequencing strategy might be inappropriate. However, a block sequencing approach is more transparent and simpler with respect to implementation.

Various algorithms and heuristics are used to schedule storage and retrieval requests within a block. The main objectives in those approaches are to minimize total travel times or total travel distances. Van den Berg and Gademann [22] developed a transportation problem (TP) model for a block sequencing in an AS/RS with dedicated storage and a single-load machine. Ratliff and Rosenthal [19] developed a graph-based algorithm to find the shortest path to visit a set of pick locations in a ladder layout. Roodbergen and de Koster [20] extended the work of Ratliff and Rosenthal [19]. They considered the order picking problem in a parallel aisle warehouse in which order pickers can cross over the aisles at the end of aisles as well as at a middle cross aisle. They developed a dynamic programming algorithm to solve the problem.

Order picking problem has been addressed by many studies. Bozer and White [1], Han et al. [5], and Lee and Schaefer [9, 10] proposed procedures to optimize the sequencing of retrieval requests based on the solution of a linear assignment problem. Mahajan et al. [18] developed a retrieval sequencing scheme aimed at improving the throughput of mini-load AS/RSs. They proposed a nearest-neighbor retrieval sequencing heuristic and developed an analytical model to predict its performance. Khojasteh-Ghamari and Son [7] studied the order picking problem in an AS/RS, where each item can be stocked at several storage locations within the warehouse. They presented a heuristic and a suitable genetic algorithm, which searches for a better retrieval solution for each customer order. Khojasteh-Ghamari [6] extended the work by formulating the problem mathematically. Through numerical studies, the performances of the only two algorithms were compared, whereas the validity of the mathematical model is missing.

Lerher et al. [16] and Lerher et al. [17] proposed some analytical travel time models for multi-aisle AS/RSs and used simulation to compare the performances of those models. In the latter one, they considered the operating characteristics of the storage and retrieval machine such as acceleration and deceleration and the maximum velocity. Other related works addressing the travel time models and evaluating the proposed models are Lerher et al. [14] for double-deep AS/RSs and Lerher et al. [12, 13] for shuttle-based storage and retrieval systems. Kouloughli and Sari [8] presented analytical models of the cycle time for multi-aisle AS/RS and evaluated the optimal dimensions of the system for a minimum cycle time. As recent studies on mini-load AS/RSs, a simulation analysis in mini-load multi-shuttle AS/RSs and an energy efficiency model were presented in Lerher et al. [15] and Lerher et al. [11], respectively.

In this paper, we address an order picking problem in a multi-aisle AS/RS with aisle transferring S/R machine, in which a single S/R machine performs storage and retrieval operations. When retrieval requests consist of multiple items and the items are in multiple stock locations, the S/R machine must travel to several storage locations to complete a customer order. The objective is to minimize the total time traveled by the S/R machine to complete the retrieval process of customer orders. We formulate the problem as a nonlinear programming model and propose a heuristic called the shortest travel distance (STD) heuristic. In addition, we provide numerical studies to compare the solutions obtained by the STD heuristic with optimal solutions obtained by Lingo solver, a software tool designed to solve nonlinear programming models.

The remainder of this paper is organized as follows. Description of the problem and assumptions are given in Section 2. A mathematical model is developed in Section 3. A heuristic is presented in Section 4. Numerical studies are presented in Section 5. Section 6 summarizes the paper and gives an overview of future works.

2 Problem description and assumptions

In this paper, we consider an end-of-aisle AS/RS with unit loads, where there are one or more pick aisles. Each aisle contains a storage rack on both sides of the aisle, and there is an I/O station at the end of each aisle. The I/O stations are at the height of the first layer (row) of the storage racks. There is a single S/R machine dedicated to all aisles of the system. The S/R machine is able to travel in cross warehouse aisle through a transfer car called "traverser" so that it can enter any pick aisle (see Fig. 1). This is the main structural difference between the system considered in this paper and that in Khojasteh-Ghamari [6], which makes it more practical in real-world applications.

If an item is retrieved from a pick aisle by the S/R machine, then it is delivered to the output station of the same aisle. In other words, in a retrieval operation, the item retrieved from an aisle is delivered to the output station of that aisle. In a deposit operation, when a storage location of an aisle is allocated for an item, the item should be loaded through the input station of that aisle. Therefore, the machine has no load when changing aisles. Storage locations are assigned to inbound parts in a random fashion (that is, no class-based or time-based storage location schema is used).

The following assumptions are also made about the S/R machine and the system operation:

- The rack face of each aisle is considered to be a continuous rectangular pick face.

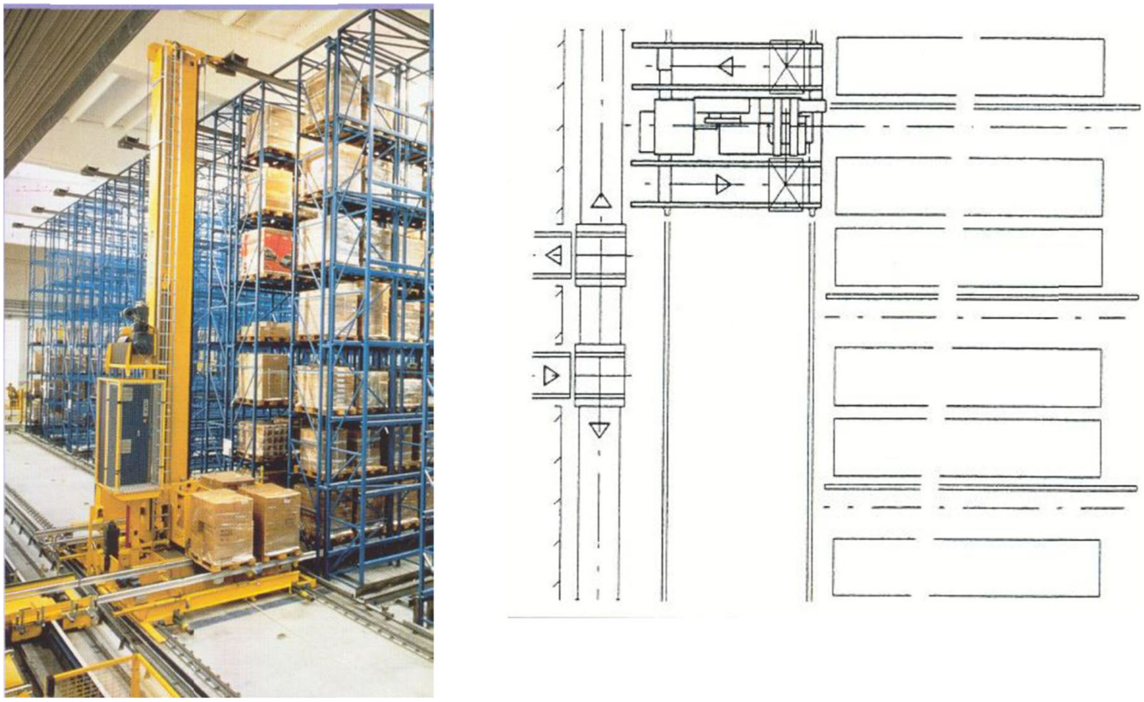


Fig. 1 Multi-aisle AS/RS with aisle transferring S/R machine

- The rack length and height of each aisle are known and identical.
- The S/R machine velocity in the horizontal and vertical directions is known.
- The S/R machine moves simultaneously in horizontal and vertical directions in order to reduce the travel time, which is called Tchebychev travel. The simultaneous movement begins in the picking aisle.
- Constant velocities are considered for horizontal and vertical travels, which include average acceleration and deceleration.
- Pickup and deposit times associated with load handling are assumed constant and, therefore, are ignored.
- The AS/RS can operate under both the single-command cycle and dual-command cycle operational policies.

The S/R machine is positioned at one of the pick aisles (at the respective I/O station) before receiving an order. This aisle is decided by the storage location (aisle) of the last item of the previous order. That is, after accomplishing the last retrieve of an order, the S/R machine stays in that pick aisle awaiting a new order. Figure 2 shows a schematic overhead view of an end-of-aisle order picking system with four pick aisles, 25 storage locations in a row of each rack, and a single S/R machine, which is located at the end of aisle number 2. Both input and output stations are shown by shaded rectangles.

When retrieval requests are made for multiple items and the items are in multiple stock locations, there will be a huge number of feasible solutions with different retrieval times. Our objective is to develop a mathematical model to find the optimal solution, which is a sequence of the requested items to be retrieved from the warehouse at the shortest time. Consider an order consists of s distinct types of items, in which n_k (k=1, 2,$\cdots$, s) items of type k are requested. The total number of feasible solutions to pick the order is given by

$$N!\prod_{k=1}^{s}\binom{m_k}{n_k} = N!\prod_{k=1}^{s}\left(\frac{m_k!}{n_k!(m_k-n_k)!}\right), \tag{1}$$

where m_k is the total number of items of type k that exists in the warehouse and $N=\sum_{k=1}^{s} n_k$.

As an example, assume that four items A, B, C, and D and one from each are requested for retrieval. As shown in Fig. 2, assume that the total number of items A, B, C, and D currently exist in the warehouse are six, four, four, and five, respectively. So, in this example, s=4, n_1=n_2=n_3=n_4=1, N=4, m_1=6, m_2=4, m_3=4, m_4=5, and the total number of feasible solutions is 11,520 which is obtained as follows.

$$4!\prod_{i=1}^{4}\binom{m_i}{n_i} = 4!\binom{6}{1}\binom{4}{1}\binom{4}{1}\binom{5}{1} = 11,520$$

The optimal solution of this example will be provided in Section 5.

The S/R machine in the considered system operates dual-command cycle, and single-command cycle if there is no item to be deposited. In Fig. 2, suppose that two items E and B are requested in an order for retrieval and that $E2$ and $B4$ are selected for retrieval. Since the S/R machine is already in

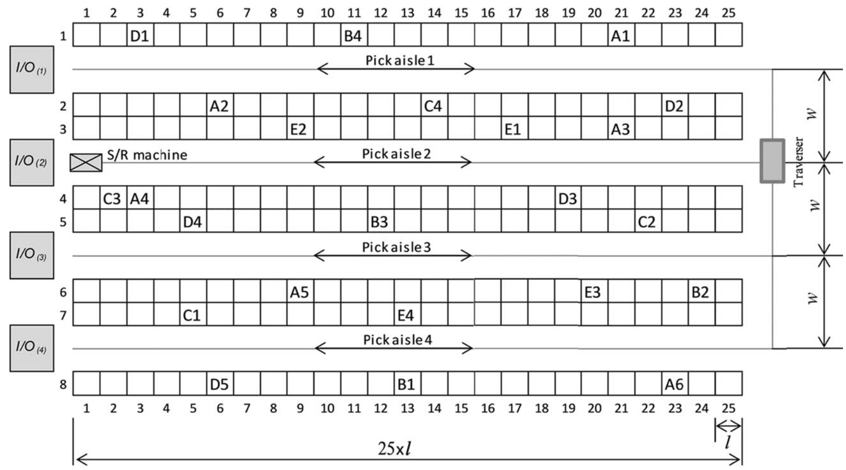


Fig. 2 A four-aisle AS/RS with a single S/R machine

aisle 2 (as shown in the figure), the machine moves to retrieve *E2* first which is located in the current aisle and drops it off at the I/O station of the aisle. Next, it needs to exit the current aisle and to enter into the aisle 1 to retrieve *B4*. However, before this move takes place, the machine checks the input buffer of the I/O station of the current aisle whether there is an item to be stored. If there is such an item, then the machine will pick it and store it in the nearest available storage location during the retrieval process without effecting the total travel time of the machine. In this case, the machine performs a dual-command cycle. However, if there is no any item to be stored, then the machine will move to aisle 1 to perform an only retrieval operation (single-command cycle).

In the following section, we develop a model, analyze the travel time of the S/R machine, and formulate the problem as a nonlinear mathematical model.

3 Model formulation

3.1 Notations

The following notations are introduced:

r	Number of rows of a storage rack (i.e., the number of layers in a rack)
c	Number of columns of a storage rack (i.e., the number of storage bins in each layer)
i	The horizontal location of a storage rack
j	The vertical location of a storage rack
l	The length of a storage bin
h	The height of a storage bin
b_{ij}	The storage bin position of column i and row j
w	The distance between two consecutive aisles
v_h	The horizontal velocity of the S/R machine w
v_v	The vertical velocity of the S/R machine
v_w	The velocity of the traverser in the cross direction
a	The pick aisle number, $a=1,2,\cdots,A_{\text{aisle}}$
a_c	The current aisle number where the S/R machine is currently located
a_s	The smallest aisle number among the aisles where the S/R machine already visited
a_l	The largest aisle number among the aisles where the S/R machine already visited
t_{ij}	The total travel time of the machine to pick up an item at b_{ij} and deliver it to the I/O station
t_e	The time needed for S/R machine to exit/enter an aisle
$\mathrm{I/O}_{(a)}$	Input/output station of pick aisle a
k	Item type, $k=1, 2, \ldots, s$
n_k	The requested number of item type k in a customer order
f	An index for rack face; $f=1$ for the rack on the right side of an aisle, $f=2$ for the left side
R_{afijk}	Equals 1 if the item type k in location (a, f, i, j) is selected for retrieval; 0 otherwise
F_{afijk}	Equals 1 if the item type k in location (a, f, i, j) is selected to be retrieved first; 0 otherwise
X_{afijk}	Equals 1 if an item in location (a, f, i, j) is the item type k; 0 otherwise
I_a	Equals 1 if aisle a has an item selected by the S/R machine; 0 otherwise

3.2 S/R machine travel time analysis

The S/R machine has independent vertical and horizontal movement capabilities. Hence, the distance metric that defines the distance from and to arbitrary storage locations within the AS/RS is the Tchebychev distance metric. That is, the time that the machine travels from a storage location to another location will be the maximum of the horizontal and vertical travel times.

In the following, we first analyze the travel time of the S/R machine in multi-aisle AS/RS. Then, we formulate the problem and present the objective function with respective constraints.

3.2.1 Travel time within a single aisle

Assume that the S/R machine is located in aisle a (at the I/O$_{(a)}$) and the item at location b_{ij} is selected for retrieval. Machine should travel the horizontal distance of $i \times l$ and the vertical distance of $j \times h$, as depicted in Fig. 3.

The horizontal and vertical travel times of the machine will be il/v_h and jh/v_v, respectively, where v_h and v_v are the horizontal and vertical velocities of the S/R machine, respectively. Since the machine moves simultaneously in vertical and horizontal directions, the time to travel from I/O$_{(a)}$ to the location b_{ij} will be the maximum value between the horizontal and vertical travel times, which is also equal to the return travel time from b_{ij} to I/O$_{(a)}$. Let t_{ij}^{a} be the total round trip travel time of the S/R machine to retrieve and deliver the item of location b_{ij} in aisle a. Then,

$$t_{ij}^{a} = 2\max\left(\frac{il}{v_h}, \frac{jh}{v_v}\right). \tag{2}$$

3.2.2 Travel time between aisles

Assume that the S/R machine is located in aisle a (at I/O$_{(a)}$) and the item in b_{ij} of the aisle a' is selected to be retrieved. In this case, the machine needs (i) to exit the current aisle a (traveling a horizontal distance of $c \times l$ before the exit); (ii) to reach aisle a' and enter the aisle, which includes traveling the distance between aisles a and a', plus an entrance time; and (iii) to reach to bin b_{ij} of aisle a'. If these three horizontal travel times are denoted by t_1, t_2, and t_3, respectively, then we have the following.

$$t_1 = \frac{cl}{v_h} + t_e, \quad t_2 = \frac{\left|a' - a\right| w}{v_w}, \quad t_3 = t_e + \frac{(c-i)l}{v_h} \tag{3}$$

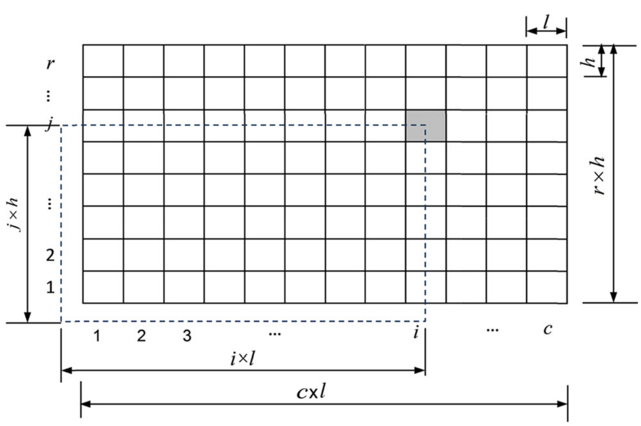


Fig. 3 Front view of a storage rack with r rows and c columns

Therefore, the horizontal travel time to reach the item in aisle a' is given by

$$t_1 + t_2 + t_3 = \frac{cl}{v_h} + \frac{(c-i)l}{v_h} + \frac{\left|a' - a\right| w}{v_w} + 2t_e. \tag{4}$$

To carry this item to I/O$_{(a')}$ (the I/O station of the current aisle, a'), $\max(il/v_h, jh/v_v)$ time unit is required (as shown in Eq. (2)). Thus, the total travel time of the S/R machine will be

$$T'_h = t_1 + t_2 + t_3 + t_{ij}^{a'}/2. \tag{5}$$

3.3 Problem formulation

In this section, we formulate the order picking problem as a nonlinear programming problem. The objective function is to minimize the total travel time of the S/R machine to retrieve and deliver all the items requested in an order. When the S/R machine must visit multiple aisles for retrievals, the total travel time is the sum of the travel time of the machine within and between the aisles. The components of the total travel time of the machine are described below:

i) *Travel time within the current aisle*:
If there is at least an item in the current aisle a_c to be selected (i.e., $\sum_f \sum_i \sum_j \sum_k R_{a_c fijk} X_{a_c fijk} \geq 1$), the travel time of the machine to retrieve the item(s) is

$$T_1 = \sum_{f=1}^{2} \sum_{i=1}^{c} \sum_{j=1}^{r} \sum_{k=1}^{s} R_{a_c fijk} X_{a_c fijk} t_{ij}. \tag{6}$$

ii) *Travel time within other aisle(s)*:
If there is an item at location b_{ij} in aisle a ($a \neq a_c$) for retrieval (i.e., $\exists i,j,f,k, F_{afijk}=1$), the machine travels the distance cl to go out of the current aisle a_c and distance $(c-i)l$ to reach the location within the aisle a. Also, the time t_e and $t_{ij}^{a}/2$ are needed for S/R machine to exit/enter an aisle and to carry the item to I/O$_{(a)}$. Therefore, the travel time of the machine to retrieve the item is

$$T_2 = \sum_{a \neq a_c}^{A} \sum_{f=1}^{2} \sum_{i=1}^{c} \sum_{j=1}^{r} \sum_{k=1}^{s} \left(F_{afijk} X_{afijk} \left(\frac{cl}{v_h} + \frac{(c-i)l}{v_h} + 2t_e + \frac{t_{ij}^{a}}{2} \right) + \left(1 - F_{afijk}\right) R_{afijk} X_{afijk} t_{ij} \right) I_a. \tag{7}$$

The first part of Eq. (7) is to retrieve and deliver the first item in the aisle a. The second part is the travel time required to retrieve and deliver the rest of the items within the aisle a, if any ($\exists i,j,f,k, \quad F_{afijk}=0$).

iii) *Travel time between the aisles*:

Based on the current location of the S/R machine, and in order to minimize the total travel time between aisles, we need to find and select the optimal sequence of the aisles to be visited by the machine. Let us define a_s and a_l as the smallest and largest aisle numbers to be visited by the S/R machine, respectively. That is,

$$a_s = \min\{a|I_a = 1\}, \text{ and } \quad a_l = \max\{a|I_a = 1\}. \quad (8)$$

For example, consider Fig. 2, where the S/R machine is located in aisle 2 (a_c=2). Assume that aisles 1 and 4 should be visited by the machine. So, a_s=1, and a_l=4. Then the shortest distance traveled between the aisles to retrieve all selected items is given by

$$d_w = w(a_l - a_s + \min\{|a_c - a_s|, |a_l - a_c|\}). \quad (9)$$

In the example,

$$d_w = w(4-1 + \min\{|2-1|, |4-2|\}) = w(3+1) = 4w,$$

where w is the distance between two consecutive aisles. This indicates that to minimize the total travel time between the aisles, the machine should visit aisle 1 first, then aisle 4. If the machine visits aisle 4 first, this distance will be $5w$, which is not minimum. Thus, the optimal travel time of the above distance is given by

$$T_3 = d_w/v_w = w(a_l - a_s + \min\{|a_c - a_s|, |a_l - a_c|\})/v_w. \quad (10)$$

Now, Eq. (10) can be written as follows:

$$T_3 = w(a_l - a_s + p|a_c - a_s| + (1-p) \ |a_l - a_c|)/v_w, \quad (11)$$

where p=1 if $|a_c - a_s| \le |a_l - a_c|$; 0 otherwise.

Therefore, the objective function is to minimize the sum of T_1, T_2, and T_3. That is,

$$\text{minimize } T = \sum_{a=1}^{A}\sum_{f=1}^{2}\sum_{i=1}^{c}\sum_{j=1}^{r}\sum_{k=1}^{s}\{F_{afijk}X_{afijk}(cl/v_h + 2t_e + (c-i)\ l/v_h + t_{ij}^a/2) + (1-F_{afijk})\left(R_{afijk}X_{afijk}t_{ij}^a\right)\}I_a$$
$$+w(a_l - a_s + p|a_c - a_s| + (1-p)|a_l - a_c|)/v_w \quad (12)$$

subject to

$$\sum_{a=1}^{A}\sum_{f=1}^{2}\sum_{i=1}^{c}\sum_{j=1}^{r}\left(F_{afijk}X_{afijk} + (1-F_{afijk})R_{afijk}X_{afijk}\right)I_a = n_k \quad (k = 1, \ 2, \ldots, \ s) \quad (13)$$

$$\sum_{f=1}^{2}\sum_{i=1}^{c}\sum_{j=1}^{r}\sum_{k=1}^{s} F_{afijk}X_{afijk} = 1 \ , \quad \forall a \ne a_c \quad (14)$$

$$\sum_{f=1}^{2}\sum_{i=1}^{c}\sum_{j=1}^{r}\sum_{k=1}^{s} F_{a_cfijk}X_{a_cfijk} = 0 \quad (15)$$

$$\sum_{f=1}^{2}\sum_{i=1}^{c}\sum_{j=1}^{r}\sum_{k=1}^{s}\left(F_{afijk}X_{afijk} + (1-F_{afijk})R_{afijk}X_{afijk}\right) \ge I_a \quad (a = 1, \ 2, \ldots, A_{\text{aisle}}) \quad (16)$$

$$R_{afijk} \in \{0,1\} \quad (a = 1,2,\ldots,A_{\text{aisle}}; \ f = 1,2, \ i = 1,2,\ldots,c; \ j = 1,2,\ldots,r; \ k = 1, \ 2,\ldots, \ s) \quad (17)$$

$$F_{afijk} \in \{0,1\} \quad (a = 1,2,\ldots,A_{\text{aisle}}; \ f = 1,2, \ i = 1,2,\ldots,c; \ j = 1,2,\ldots,r; \ k = 1, \ 2,\ldots, \ s) \quad (18)$$

$$I_a \in \{0,1\} \quad (a = 1,2,\ldots,A_{\text{aisle}}) \quad (19)$$

$$p \in \{0,1\} \quad (20)$$

Constraint (13) guarantees that the all items requested in the order are retrieved and delivered. Constraint (14) represents that if any aisle (other than the current one) has an item for retrieval, then there is only one item in that aisle which is selected first for retrieval. Constraint (15) prevents any items in current aisle a_c from being selected as a first retrieval item. Constraint (16) states that if an aisle has an item which is selected for retrieval, then that item must be retrieved.

4 Heuristic

Although the nonlinear programming model formulated in Section 3 provides the optimal solution, the number of constraints and variables increases dramatically when the number of items increases. Therefore, in this section, we present a heuristic to solve the order picking problem. We call it the shortest travel distance (STD) heuristic. In fact, the minimum travel time of the machine to retrieve all the items requested in a customer order is a key in developing this heuristic. If all requested items exist in the current aisle, then they will be selected for retrieval. Thus, all retrieval operations will be done within the current aisle and the S/R machine will not leave the aisle to accomplish the retrieval process. However, if at least one requested item does not exist in the current aisle, then the heuristic will search those items in the nearest aisles, and if an item is in multiple locations within the aisle, the priority for selection will be for those that are closer to the I/O station of the aisle. This minimizes the total travel time of the machine. When a requested item, which is already in the current aisle, also exists in another aisle, then the heuristic will select the one with a shorter travel time in total.

Before describing the steps of the heuristic, we define the following notations:

N_k^{order} Order quantity for item type k $(=1,\ 2,\ldots,\ s)$

N_k^a Total number of item k that exists in aisle $a(=1,\ 2,\ldots,\ A_{\text{aisle}})$

STD_d Set of aisles in $[a_s, a_l]$ satisfying $d=(a_l-a_s)+\min\{|a_l-a_c|,|a_c-a_s|\}$, where d denotes the distance traveled by the machine from current aisle a_c. The set can be expressed as follows.

$$\begin{aligned}\text{STD}_d = \{\forall a\in[a_s, a_l]|(a_l-a_s)\\ +\min\{|a_l-a_c|,\ |a_c-a_s|\}\\ = d,\ 0\le d\le d_{\max}\},\end{aligned} \tag{21}$$

where $d_{\max}=A_{\text{aisle}}-1+\min\{A_{\text{aisle}}-a_c, a_c-1\}$, the possible maximum travel distance from the current aisle a_c.

STD_d^δ $\delta(\delta=1,\ 2,\ldots)$ th type of STD_d for a given d if there exists more than one type of STD_d.

$R_{\text{STD}_d^\delta}^{\text{total}}$ Set of the locations of all the selected items for retrieval that are located in the aisles of set STD_d^δ. The total number of each selected item k should be less than or equal to the requested number of the item, N_k^{order}. Therefore, it can be defined as follows.

$$R_{\text{STD}_d^\delta}^{\text{total}} = \left\{(a,f,i,j,k)\left|\sum_{\beta=1}^{r}\left(\sum_{i=1}^{\beta\alpha}\sum_{f}\sum_{a\in\text{STD}_d^\delta}X_{afi\beta k}+\sum_{i=\beta\alpha+1}^{(\beta+1)\alpha}\sum_{j=1}^{\beta}\sum_{f}\sum_{a\in\text{STD}_d^\delta}X_{afijk}\right)\le N_k^{\text{order}},\ k=1,\ 2,\ldots,\ s\right.\right\}, \tag{22}$$

where $\alpha=\lceil hv_h/lv_v\rceil(>1)$. $\lceil x\rceil$ denotes the round-up of x to the nearest integer. The locations of $R_{\text{STD}_d^\delta}^{\text{total}}$ with respect to the requested order quantity N_k^{order} are selected based on the retrieval sequence shown in Table 1. In definition above, the first term of the inequality represents the summation of items located in the vertical direction column of Table 1, whereas the second term represents those in the horizontal direction column.

Note that the vertical and horizontal travel times to reach the item located in b_{ij} are given by il/v_h and jh/v_v, respectively.

Table 1 The retrieval sequence to select the items located near $\text{I/O}_{(a)}$ station

Sequence	Vertical direction		Horizontal direction	
β	i	j	i	j
1	$1\le i\le 1\alpha$	1	$1\alpha+1<i\le(1+1)\alpha$	1
2	$1\le i\le 2\alpha$	2	$2\alpha+1\le i\le(2+1)\alpha$	$1\le j\le 2$
⋮	⋮	⋮	⋮	⋮
$r-1$	$1\le i\le(r-1)\alpha$	$r-1$	$(r-1)\alpha+1\le i\le r\alpha$	$1\le j\le r-1$
r	$1\le i\le r\alpha$	r	–	–

Then, since the machine can simultaneously move in horizontal and vertical directions, it can reach to the horizontal location $i=j\alpha$ first, while it is moving up to reach the vertical location j because the horizontal velocity is usually greater than the vertical one. The *vertical direction* column in the table represents the location of item (i,j), where $1\le i\le\beta\alpha$ and $j=\beta$. In fact, the travel time to reach this location is always given by jh/v_v, which depends only on the *vertical* location $j(=\beta)$ for a given sequence number β regardless of the horizontal location i $(1\le i\le\beta\alpha)$. For example, when $\beta=2$, the retrieval times for the items located in $(1\le i\le\beta\alpha,\beta)$, i.e., $(1\le i\le 2\alpha,\ 2)$ are all $\beta h/v_v=2h/v_v$. The *horizontal direction* column in Table 1 represents the location of item (i,j) such that $(\beta\alpha+1\le i\le(\beta+1)\alpha,\ 1\le j\le\beta)$, where the travel time to reach that location is always given by il/v_h, which depends only on *horizontal* location i $(\beta\alpha+1\le i\le(\beta+1)\alpha)$ for a given sequence number β regardless of the vertical location j $(1\le j\le\beta)$. When $\beta=2$, for instance, the retrieval times for the items located in $(2\alpha+1\le i\le(2+1)\alpha,\ 1\le j\le 2)$ are il/v_h regardless of j $(1\le j\le 2)$. Therefore, for a given retrieval sequence number β, the items located in (i,j) of the vertical direction (column) should be selected prior to those of the horizontal direction (column) according to the STD heuristic.

$R^{\text{first}}_{\text{STD}^{\delta}_{d}}$ The location of items to be visited first for retrieval in other aisles of STD^{δ}_{d}, which is defined as follows.

$$R^{\text{first}}_{\text{STD}^{\delta}_{d}} = \left\{ (a,f,i,j,k) \middle| \max_{t^a_{ij}} R^{\text{total}}_{\text{STD}^{\delta}_{d}},\ a\in\text{STD}^{\delta}_{d},\ a\neq a_c \right\} \quad (23)$$

Since the travel time of the first selected item is given by the total horizontal time of the machine to reach the item from the entrance of the aisle plus the time to $\text{I/O}_{(a)}$ station of $t^a_{ij}/2$ after retrieval, the item located farthest away from the $\text{I/O}_{(a)}$ station of the aisle $a\in\text{STD}^{\delta}_{d}$ will be selected as the first item for retrieval within the aisle. In other words, the first item will be the one with the largest travel time t^a_{ij} among all the selected items, which are the elements of $R^{\text{total}}_{\text{STD}^{\delta}_{d}}$ in aisle $a\in\text{STD}^{\delta}_{d}$.

$R_{\text{STD}^{\delta}_{d}}$ $R^{\text{total}}_{\text{STD}^{\delta}_{d}} - R^{\text{first}}_{\text{STD}^{\delta}_{d}}$

$T_{\text{STD}^{\delta}_{d}}$ The total travel time for STD^{δ}_{d}, which can be expressed as

$$T_{\text{STD}^{\delta}_{d}} = \sum_{(a,f,i,j,k)\in R_{\text{STD}^{\delta}_{d}}} 2X_{afijk} t^a_{ij} + \sum_{(a,f,i,j,k)\in R^{\text{first}}_{\text{STD}^{\delta}_{d}}} X_{afijk}\left(t^a_{ij} + ((m-i)+ml)/v_h + 2t_e\right) + wd/v_h. \quad (24)$$

Three steps of the STD heuristic:

Step 1. Input initial data: Initial data includes the information about the AS/RS structure such as horizontal and vertical velocities of the S/R machine, storage rack size, item locations, and order quantity for each item k (N^{order}_k, $k=1,\ 2,\ldots,\ s$).

Step 2. Count the number of ordered items in each aisle, N^a_k ($a=1,\ 2,\ldots,A;\ k=1,\ 2,\ldots,\ s$). If $\sum_a N^a_k \geq N^{\text{order}}_k$ for each item k, then set $d=0$ and go to the next step; otherwise, no feasible solution exists.

Step 3. If $d<d_{\max}$, then we have STD^{δ}_{d}, and follow the two procedures (a) and (b) below; otherwise, the minimum travel time of the machine is given by $T_{\text{total}} = \min\left(T_{\text{STD}^{\delta}_{d_{\max}}}, \forall\delta\right)$, and stop the algorithm.

(a) If at least one of STD^{δ}_{d} satisfies $\sum_{a\in\text{STD}_d} N^a_k \geq N^{\text{order}}_k$ for all k, then the minimum travel time is $T_{\text{total}} = \min\left(T_{\text{STD}^{\delta}_{d}}, \forall\delta\right)$, and stop the algorithm.

(b) If there exists any k such that $\sum_{a\in\text{STD}^{\delta}_{d}} N^a_k < N^{\text{order}}_k$ for all δ, set $d=d+1$ and go to *Step 3*.

5 Numerical studies

We conducted numerical studies in order to evaluate the performance of the proposed STD heuristic. We developed five scenarios given in Table 2. There are seven item types stored in the warehouse: A, B, C, D, E, F, and G. Each item can be found in six different locations in the warehouse (i.e., inventory of each item is 6). Initial locations of the all items in the warehouse are generated randomly. The scenarios are developed based on five different types of customer orders, so that in the scenario numbers 1 to 5, the number of required items for retrieval is one to five, respectively. We also set the warehouse parameters as follows: $A_{\text{aisle}}=4$, $a_c=2$, $l=1\text{m}$, $h=1\text{m}$, $w=3.5\text{m}$, $c=36$, $r=12$, $v_h=3\text{m/s}$, $v_v=1\ \text{m/s}$, $v_w=0.6\text{m/s}$, and $t_e=4\text{s}$.

In selecting the warehouse structural parameters, we considered the concept of shape factor (or the configuration of rack) defined by Bozer and White [1, 2] to improve the performance of the system. Given that the horizontal and vertical velocities of the S/R machine are 3 and 1 m/s, respectively, a rack with length of 36 and height of 12 m can achieve 1 for the shape factor as the best value. With this value, the machine will require the least amount of time to reach the furthest storage location in the rack.

In order to evaluate the performance of the STD heuristic, in each scenario, we first solve the problem as the nonlinear programming model given in Eqs. (13–22) using Lingo solver. Then, we solve the problem according to the STD heuristic. The simulation was performed using Mathematica 5.0. The results are summarized in Table 3.

Comparing the results given in Table 3 reveals the fact that in the Lingo, the elapsed runtime of CPU increases

Table 2 Scenarios for numerical experiment

Scenario	Number of items in the customer order	Items requested in the customer order	Number of feasible solution (see Eq. (1))
1	1	*A*	6
2	2	*B,C*	72
3	3	*C,E,G*	1296
4	4	*A,B, D,F*	31,104
5	5	*B,D,E,F,G*	933,120

Table 3 Comparison of results between STD heuristic and Lingo solver

Scenario	Lingo output			STD heuristic output		
	Location of selected items	Minimum travel time (s)	Elapsed runtime of CPU (s)	Selected items	Minimum travel time (s)	Elapsed runtime of CPU (s)
1	(2,1,3,4,A)	8	3:01	(2,1,3,4,A)	8	Within 5
2	(1,2,14,4,B) (1,1,16,5,C)*	49.17	16:32	(2,2,21,2,C) (3,2,33,8,B)*	53.67	
3	(1,1,16,5,C)* (1,1,16,1,E) (2,2,9,1,G)	56.50	45:37	(3,2,28,4,C)* (3,2,19,1,E) (2,2,9,2,G)	62.33	
4	(2,1,3,4,A) (1,2,14,4,B) (4,2,6,3,D) (4,1,7,11,F)*	82.83	1:32:29	(2,1,3,4,A) (2,2,2,10,D) (3,2,8,5,F) (4,2,26,3,B)*	91.67	
5	(2,2,14,4,B) (1,2,6,3,D) (3,1,16,1,E) (3,1,7,11,F)* (4,2,9,1,G)	103.50	2:03:24	(1,2,8,5,F)* (2,2,2,10,D) (3,2,9,2,G) (4,2,19,1,E) (4,2,26,3,B)	112.00	

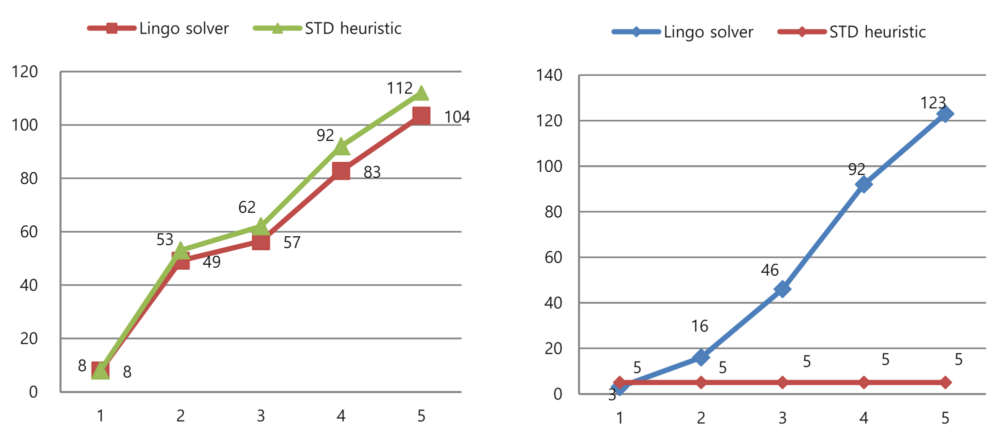
*( )** the first selected item for retrieval in each relevant aisle

dramatically as the number of requested items in the customer order (scenario number) increases, while the STD heuristic finds the solution in less than 5 s. However, there is a slight difference between the solutions of the heuristic and optimal one. In fact, when the problem size is small, as in scenario 1, the solution obtained by the heuristic is optimal or sub-optimal. However, as the size of the problem increases, because of increasing the number of feasible solutions, Lingo requires more time to find the optimal solution, while the heuristic obtains a solution which is very close to the optimal but in a much shorter time. The graphs of time comparison between Lingo solver and STD heuristic are depicted in Fig. 4.

By comparing the total travel times and CPU times of the both methods in Fig. 4, one can see that as the number of items in the customer order increases, the CPU time of the optimal solution increases dramatically while the differences between the total travel times change very slightly. The reason is that a small increase in the number of items in the customer order increases the number of feasible solutions dramatically. With a large number of feasible solutions, the Lingo requires a longer time to find the optimal solution. Thus, its CPU time increases. However, the STD heuristic regardless of the number of feasible solutions seeks the items in the nearest location which requires a small amount of CPU time, while the obtained solutions are very close to optimal values.

Consider the example given in Section 2 where we computed the total number of feasible solutions for the problem. Four items *A*, *B*, *C*, and *D* and one from each were requested for retrieval. Assume that the total number of items *A*, *B*, *C*, and *D* existed in the warehouse were respectively six, four, four, and five, as depicted in Fig. 2. After solving this problem by both STD heuristic and Lingo, the solutions of both

Fig. 4 Time comparison between STD heuristic and Lingo solver



(a) Machine travel time (b) CPU runtime

methods were identical, where the selected items for retrieval are (2, 2, 3, 1, *A*), (1, 1, 11, 1, *B*)*, (2, 2, 2, 1, *C*), and (1, 1, 3, 1, *D*). However, the elapsed runtime of the Lingo was about 40 s more than that of the STD heuristic. ( )* represents the first selected item for retrieval in each relevant aisle.

6 Conclusions

In this paper, we addressed an order picking problem in a multi-aisle unit-load AS/RS served by a single S/R machine, in which each item can be found in several storage locations. There is a large number of feasible solutions with different retrieval times. We first formulated the problem as a nonlinear programming model and developed a heuristic called the STD heuristic. Then, through the numerical experiments, we compared the performances of the STD heuristic and the mathematical model in a set of five different scenarios. The results showed that the STD heuristic performs better in terms of the CPU runtime which is needed to obtain a solution, and the solutions were very close to optimal.

In the Lingo, the elapsed runtime of CPU increases dramatically as the number of requested items in the customer order increases. This makes the method impractical. However, the STD heuristic finds the solutions in much shorter times. The travel time of the machine obtained by the STD heuristic is optimal in scenario 1 where the problem size is small. However, when the number of items in the customer order increases, the travel times of the machine obtained by the heuristic are slightly higher than optimal values. In fact, adding one more item into the customer order increases the total number of feasible solutions, and hence the CPU time of the Lingo to find the optimal solution will increase significantly. With a large number of feasible solutions, Lingo requires longer time to find the optimal solution. However, the STD heuristic, regardless of the number of feasible solutions, seeks the items in the nearest location which requires a small amount of CPU time, while the obtained solutions are sub-optimal. This makes the STD heuristic practical, because the differences are insignificant compared to those of the CPU times.

In the future, the performance of the proposed STD heuristic would be compared with the other algorithms existing in the literature including meta-heuristics such as genetic algorithms. Also, considering acceleration and deceleration for the S/R machine in the mathematical model could make it more practical. Furthermore, applying STD heuristic to the diverse end-of-aisle mini-load systems classified by Foley and Frazelle [4] would be an interesting topic.

Although the throughput requirement is not directly considered in our model, it can be obtained by setting the maximum number of aisles "A_{aisle}" based on some other parameters such as machine velocities. However, it is possible to define it as a parameter based on which other parameters can be determined. This would be a different and interesting model and can be developed in a future study. We also considered that when the S/R machine operates a dual-command cycle, an item can be stored in the nearest available storage location during the retrieval process. However, to minimize the total travel time of the machine, a more effective storage policy could be employed.

Compliance with ethical standards

Conflict of interest The authors declare that they have no competing interests.

References

1. Bozer YA, White JA (1984) Travel-time models for automated storage/retrieval systems. IIE Trans 16(4):329–338
2. Bozer YA, White JA (1990) Design and performance models for end-of-aisle order picking systems. Manag Sci 36(7):852–866
3. Eben-Chaime M (1992) Operations sequencing in automated warehousing systems. Int J Prod Res 30(10):2401–2409
4. Foley RD, Frazelle E (1991) Analytical results for miniload throughput and the distribution of dual command travel time. IIE Trans 23:273–281
5. Han M-H, McGinnis LF, Shieh JS, White JA (1987) On sequencing retrievals in an automated storage/retrieval system. IIE Trans 19:56–66
6. Khojasteh-Ghamari Y (2012) Warehouse management: productivity improvement in automated storage and retrieval systems. In: Manzini R (ed) Warehousing in the global supply chain. Springer, London
7. Khojasteh-Ghamari Y, Son JD (2008) Order picking problem in a multi-aisle automated warehouse served by a single storage/retrieval machine. Int J Inf Manag Sci 19(4):651–665
8. Kouloughli S, Sari Z (2015) Multi-aisle AS/RS dimensions optimization for cycle time minimization. Int J Adv Manuf Technol. doi: 10.1007/s00170-014-6709-3, published online: Feb. 13, 2015
9. Lee HF, Schaefer SK (1996) Retrieval sequencing for unit-load automated storage and retrieval systems with multiple openings. Int J Prod Res 34:2943–2962
10. Lee HF, Schaefer SK (1997) Sequencing methods for automated storage and retrieval systems with dedicated storage. Comput Ind Eng 32:351–362
11. Lerher T, Edl M, Rosi B (2014) Energy efficiency model for the mini-load automated storage and retrieval systems. Int J Adv Manuf Technol 70(1–4):97–115
12. Lerher T, Ekren BY, Dukic G, Rosi B (2015) Travel time model for shuttle-based storage and retrieval systems. Int J Adv Manuf Technol 40(1–3):101–121
13. Lerher T, Ekren YB, Sari Z, Rosi B (2015) Simulation analysis of shuttle based storage and retrieval systems. Int J Simul Modell 14(1):11–23
14. Lerher T, Potrc I, Šraml M, Tollazzi T (2010) Travel time models for automated warehouses with aisle transferring storage and retrieval machine. Eur J Oper Res 205:571–583
15. Lerher T, Sraml M, Potrc I (2011) Simulation analysis of mini-load multi-shuttle automated storage and retrieval systems. Int J Adv Manuf Technol 54(1–4):337–348

16. Lerher T, Sraml M, Kramberger J, Potrc I, Borovinsek M, Zmazek B (2006) Analytical travel time models for multi aisle automated storage and retrieval systems. Int J Adv Manuf Technol 30(3–4): 340–356
17. Lerher T, Sraml M, Potrc I, Tollazzi T (2010) Travel time models for double-deep automated storage and retrieval systems. Int J Prod Res 48(11):3151–3172
18. Mahajan S, Rao BV, Peters BA (1998) A retrieval sequencing heuristic for miniload end-of-aisle automated storage/retrieval systems. Int J Prod Res 36:1715–1731
19. Ratliff HD, Rosenthal AS (1983) Order-picking in a rectangular warehouse: a solvable case of the traveling salesman problem. Oper Res 31:507–521
20. Roodbergen KJ, De Koster R (2001) Routing order pickers in a warehouse with a middle aisle. Eur J Oper Res 133:32–43
21. Tompkins JA, White JA, Bozer YA, Tanchoco JMA (2003) Facilities planning, 3rd edn. John Wiley and Sons, New York
22. Van den Berg JP, Gademann AJRM (1999) Optimal routing in an automated storage/retrieval system with dedicated storage. IIE Trans 31:407–415