Int j simul model 16 (2017) 1, 145-156
ISSN 1726-4529 Original scientific paper

OPTIMAL SELECTION OF MOVABLE SHELVES UNDER *CARGO-TO-PERSON* PICKING MODE

Li, Z. P.[*]; Zhang, J. L.[**,#]; Zhang, H. J.[*] & Hua, G. W.[**]
[*] School of Information, Beijing Wuzi University, Beijing 101149, P. R. China
[**] School of Economics and Management, Beijing Jiaotong University,
Beijing 100044, P. R. China
E-Mail: zhangjl@bjtu.edu.cn ([#] Corresponding author)

Abstract
In recent years, some e-commerce companies such as Amazon have adopted the *cargo-to-person* picking mode to improve their pickup efficiency. Under this mode, a shelf can store several types of goods and a type of goods can be placed on some shelves. When orders arrive, the warehouse robots move one shelf or more containing the ordered items to a fixed platform, and the pickers select the items from the shelves. It is very important to decide which shelves should be moved to increase picking efficiency. This paper addresses the problem of optimal movable-shelf selection for the *cargo-to-person* picking mode. The goal of this study is to minimize the total time (costs) of moving the selected shelves to finish a batch of orders. We model this problem using 0-1 linear programming and show that the problem is NP-hard. Furthermore, we propose a three-stage hybrid heuristic algorithm with polynomial complexity to solve it. We conduct numerical experiments to show the efficiency of this algorithm.

(Received, processed and accepted by the Chinese Representative Office.)

Key Words: *Cargo-to-Person* Mode; Warehousing; 0-1 Linear Programming; Heuristic Algorithm

1. INTRODUCTION

In recent years, e-commerce has developed rapidly, and online sales have greatly increased, requiring the support of fast and efficient logistics. Different from traditional offline logistics, e-commerce logistics are characterized by variety [1], high frequency and small batches. To enhance the efficiency of order-picking, the designers of the Kiva system proposed a new picking mode: *cargo-to-person*. Based on this mode, they developed a new warehouse management system – Kiva. This system uses storage shelves which is movable and can be moved by robots. Moving the shelves containing products in a batch of orders to the pickers has increased warehousing efficiency greatly and improved flexibility and accountability simultaneously [2]. In March 2012, Amazon spent $775 million to buy the Kiva system, which it used in its warehouse management. Picking efficiency has greatly increased since then, and picking costs have been shown to help save $458 million to $916 million annually.

The layout of warehouses that use a *cargo-to-person* picking mode is depicted in Fig. 1. The movable shelves, on which goods are stored according to predetermined rules, are arranged in the upper part of Fig. 1. Each shelf can store several types of goods, and every type of goods can be stored on multiple shelves [3]. The working platform is located in the lower left corner, and the robots' parking and charging area is in the lower right corner. When orders arrive, they are partitioned into several batches according to predetermined rules. For each batch of orders, one shelf or more is selected and moved to the working platform by robots. After the workers pick items from these shelves, the shelves are returned to their original place. In the *cargo-to-person* picking mode, consumer goods that sell quickly might be placed on multiple shelves [4, 5] to increase picking efficiency and reduce the moving time needed for selected shelves. Shelf selection for a given batch of orders is a key issue for the *cargo-to-person* picking mode. This paper addresses this problem.

DOI:10.2507/IJSIMM16(1)CO2

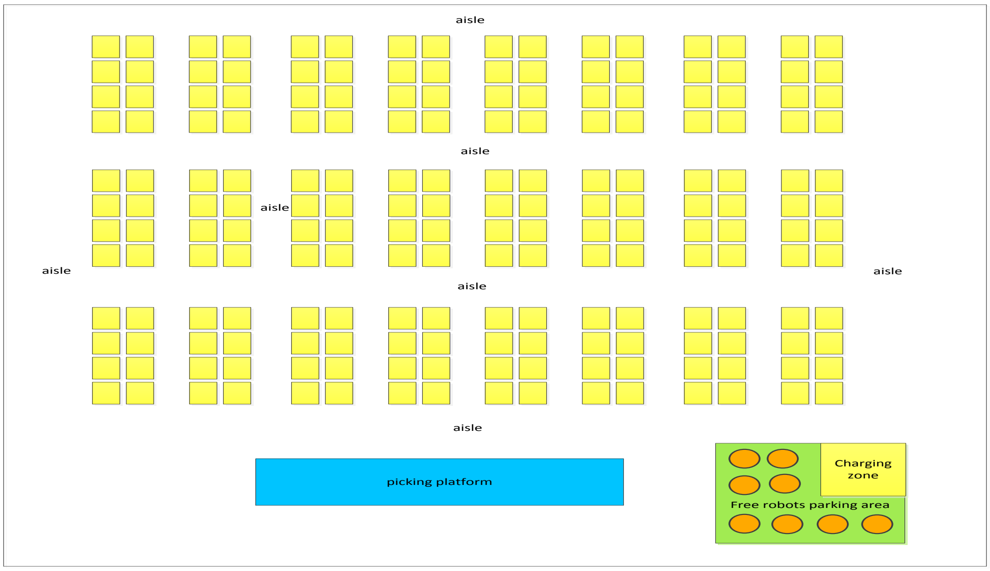

Figure 1: The layout of a warehouse with *cargo-to-person* picking mode.

Although the Kiva system has worked well in Amazon's warehouses, there are still some problems need to be studied [2]. The designers of the Kiva system called for researchers to study the problems related to the *cargo-to-person* picking mode such as order-batching problems, movable-shelf selection problems, task-assignment and robot-motion planning problems, etc. [3, 4]. Among these problems, task-assignment and robot-motion planning problems have been studied extensively [5-9], but fewer studies have been conducted on order-batching problems and movable-shelf selection problems.

Although there are few studies of the picking methods used in the *cargo-to-person* picking mode, the picking methods under the classical *person-to-cargo* picking mode are well-studied [10-12]. In the *person-to-cargo* picking mode, the picking process consists of the following steps: order-batching, sequencing and routing, and sorting [10].

The order-batching problem is a special clustering problem, and most research about this problem focuses on designing efficient algorithms for different objective functions [10-12]. Pan et al. [13] developed an order-batching approach for a pick-and-pass system. Öncan [14] introduced mixed-integer programming formulations for traversal, return and midpoint routing policies, respectively, and designed some approximate algorithms for them. Menéndez et al. [15] designed a two-stage variable neighbourhood search method for the order-batching problem. Henn [16] studied the problem of minimizing the total delay for picking orders.

The sequencing and routing steps in the *person-to-cargo* method are very important since they consume more than 50 % of the total order-processing time. The static sequencing and routing problems are similar to the TSP and are NP-hard, and the dynamic sequencing and routing problem is more difficult. Most research on this problem focuses on designing efficient algorithms [10-12]. Lu et al. [17] studied dynamic order-picking route problem and proposed a new routing algorithm for it. Hong and Kim [18] introduced a route-selecting order-batching model with the S-shape routing method on the parallel-aisle order-picking problem. Schrotenboer et al. [19] proposed hybrid genetic algorithms for route determination that entail simultaneous pickup of consumers' orders and storage of returned products.

Since order-batching, sequencing and routing, and sorting problems are correlated, some studies have addressed the integration of these processes. Chen et al. [20] developed a nonlinear mixed-integer optimization model to study the integration including order-batching, batch sequencing, and pickers' routing. Chen et al. [21] proposed a heuristic approach for an integration problem of batch-picking and picker-routing problem.

Comparing the operational processes in a *cargo-to-person* mode and those in a *person-to-cargo* mode, we can find that if movable shelves are selected, the task-assignment and robot-motion planning problems in the *cargo-to-person* mode are similar to the picking-routing problem in the *person-to-cargo* mode. In the *person-to-cargo* mode, the locations of items to be picked are determined for a given batch of orders. Pickers want to find optimal picking routes that pass the locations of the selected items. However, in the *cargo-to-person* mode, there are multiple strategies for selecting movable shelves to pick the items in a batch of orders. The optimal combination of shelves should be selected to be moved before assigning tasks to the robots and planning routes for them. Hence, the optimal movable-shelves selection problem in the *cargo-to-person* mode is a new problem that does not exist in the *person-to-cargo* mode.

The optimal movable-shelf selection problem is a new combinatorial optimization problem, although its objective function is similar to that of the weighted set covering problem [22], since the constraints of these two problems are different. As we show in Section 2, the weighted set covering problem is a special case of the movable-shelf selection problem.

In this paper, we study the optimal movable-shelf selection problem in the *cargo-to-person* picking mode considering the types and the quantities of goods in a batch of orders, the types and quantities of goods placed on the shelves, and the time needed to move each shelf. The goal is to minimize the total moving time. We model this problem using 0-1 linear programming and prove that the problem is NP-hard. A hybrid heuristic algorithm with polynomial complexity O(MK^3) is designed for solving the model. We also conduct numerical experiments to demonstrate the effectiveness and efficiency of the model and algorithm.

This paper is arranged as follows: We state the problem and the corresponding mathematical model in Section 2. In Section 3, we state the algorithm for the problem, and discuss its complexity and worst-case analysis. In Section 4, an example is given to illustrate the algorithm. Extensive numerical experiments are conducted to demonstrate the effectiveness and efficiency of the model and algorithm in Section 5. Further conclusions and discussions are given in Section 6.

2. MATHEMATICAL MODEL AND BASIC PROPERTIES

Consider M types of goods and K movable shelves in a warehouse. Several types of goods might be stored on one shelf, and one type of goods might be stored on several shelves. Suppose that the quantity of goods i (i = 1, 2, …, M) stored on the j^{th} (j =1, 2, …, K) shelf is a_{ij} ($a_{ij} \geq 0$). The round-trip time for moving shelf j from its location to the picking platform is w_j. Given a batch of orders to be picked, the quantity of goods i in the batch of orders is q_i ($q_i \geq 0$) (i = 1, 2, …, M). For shelf j = 1, 2, …, K, the quantity of goods in the batch of orders might be $q_i > a_{ij}$. To pick all items in a batch of orders, we want to determine which shelves should be moved to minimize the total round-trip time.

The notations used in this paper are listed as follows:

$R_j = (a_{1j}, a_{2j}, \ldots, a_{Mj})^T$ (j = 1, 2, …, K): vector of goods' quantities on the shelf j;

$Q = (q_1, q_2, \ldots, q_M)^T$: vector of goods' quantities in the batch of orders to be picked;

$q_{\max} = \max\limits_{1 \leq i \leq M} q_i$;

$P_j = (p_{1j}, p_{2j}, \ldots, p_{Mj})^T$ (j = 1, 2, …, K): vector of quantities of the goods that will be picked from shelf j. Then, $p_{ij} = \min\{a_{ij}, q_i\}$;

$b_j = \sum\limits_{i=1}^{M} p_{ij}$: total quantity of all types of goods that should be picked up from shelf j if shelf j is moved to the picking platform;

$U_j = (u_{1j}, u_{2j}, \ldots, u_{Mj})^T$, $u_{ij} = \begin{cases} 1, & p_{ij} > 0 \\ 0, & otherwise \end{cases}$;

$c_j = \sum_{i=1}^{M} u_{ij}$: the number of types of goods that can be picked from shelf *j*;

$g = \max_j c_j$: the maximum number of types of goods that can be picked from one shelf;

$W = (w_1, w_2, \ldots, w_K)$: the vector of the round-trip time;

S: the set of selected shelves to move, which corresponds to a feasible solution;

F: the set of candidates that have one or more types of goods to be picked;

N: the set of shelves that have no goods to be picked;

Cost: the total time for moving all selected shelves.

The decision variables are defined as follows:

$$x_j = \begin{cases} 1, & shelf\ \ j\ is\ selected\ to\ move \\ 0, & otherwise \end{cases} \qquad j = 1, 2, \ldots, K.$$

We can formulate the optimal movable-shelf selection problem as the following 0-1 programming:

$$\text{(IP)} \quad min\ z = \sum_{j=1}^{K} w_j x_j \qquad (1)$$

$$s.t. \begin{cases} \sum_{j=1}^{K} a_{ij} x_j \ge q_i & i = 1,2\ldots,M \qquad (2) \\ x_j \in \{0,1\} & j = 1,2,\ldots,K \qquad (3) \end{cases}$$

The objective function is the total round-trip time of moving all selected shelves. Constraints (2) ensure that all items in the orders can be picked up from the selected shelves. Constraints (3) are the binary constraints on the decision variables.

For the model (IP), if $a_{ij} = 0$ or 1, and $q_i = 1$, the quantity of each type of goods on one shelf is at most 1, and the quantity of each type of goods in the batch of orders is exactly 1, the problem is reduced to the well-studied WSCP[23]. If $a_{ij} = 0$ or 1, $q_i = 1$ and $w_j = 1$, then the problem is reduced to the classical set cover problem (SCP)[24]. Therefore, the model (IP) includes both SCP and WSCP as special cases.

It is well-known that WSCP is strongly NP-hard when some sets are allowed to have more than two elements [22]. Therefore, the problem (IP) is also strongly NP-hard when more than two types of goods are stored on a shelf. Since most shelves store more than two types of goods in a *cargo-to-person* picking mode [3], the problem studied in this paper is strongly NP-hard.

It is well-known that there are some heuristic algorithms for SCP and WSCP [22, 23, 25-28], and their worst-case approximate ratio is not very good. The problem (IP) is more complicated since the constraints coefficient matrices of SCP and WSCP are a 0-1 matrix, but the constraint coefficient matrix of the problem (IP) is no longer a 0-1 matrix.

3. THE HYBRID HEURISTIC ALGORITHM

Since the optimal movable-shelf selection problem is strongly NP-hard, for the small-scale problem, we can use some algorithms, such as branch and bound algorithms, to solve it. However, for large-scale practical problems that contain a large number of shelves and goods, the exact algorithms are time-consuming. Therefore, it is necessary to design a fast, efficient heuristic approach to get an approximate solution. In the following, we shall design a hybrid heuristic method to get an approximate optimal solution.

3.1 The philosophy of the hybrid heuristic algorithm

The hybrid heuristic algorithm consists of three stages. The first stage is to find a feasible solution using a heuristic algorithm. A feasible solution entails a set of movable shelves from which all items in the batch of orders can be picked. The second stage is to delete redundant shelves from the feasible solution to enhance its precision. The third stage adopts an exchange strategy to further improve the solution precision. That is, for each unselected shelf, we add the shelf to the solution and delete the redundant shelves. If the objective function of the new solution is smaller than that of the original one, we execute this exchange strategy; otherwise, we do not.

3.2 The algorithm

Input: The number of types of goods M, the number of shelves K, the quantity vector of goods stored on a shelf $R_j = (a_{1j}, a_{2j}, \ldots, a_{Mj})^T$ ($j = 1, 2, \ldots, K$), the quantity vector of goods in the batch of orders to be picked $Q = (q_1, q_2, \ldots, q_M)^T$, and the round-trip time to move shelf j to the picking platform w_j ($j = 1, 2, \ldots, K$).

```
Initialization: Let $F = \{1, 2, \ldots, K\}$, $S = \varnothing$, and $N = \varnothing$. Set $Cost = 0$.
Stage 1. Find a feasible solution.
 Step 1.1 For each $j \in F$
     Calculate $P_j = (p_{1j}, p_{2j}, \ldots, p_{Mj})^T$, where $p_{ij} = \min\{a_{ij}, q_i\}$, $i = 1, 2, \ldots, M$
         $b_j = \sum_{i=1}^{M} p_{ij}$
     If $b_j = 0$, $F \Leftarrow F \setminus \{j\}$;   $N \Leftarrow N \cup \{j\}$
     else
     $rate(P_j) = \frac{w_j}{b_j}$
 Step 1.2 Set $j^* = \arg\min_{j \in F} \{rate(P_j)\}$ and
         $S \Leftarrow S \cup \{j^*\}$, $F \Leftarrow F \setminus \{j^*\}$, $Cost = Cost + w_{j^*}$, $Q = Q - P_{j^*}$
 Step 1.3 If $Q \neq 0$ and $F \neq \varnothing$, go to Step 1.1; else, go to Stage 2.
Stage 2. Delete the redundant shelves from the selected set $S$.
 Step 2.1 For each $j \in S$
     If $\sum_{t \in S \setminus \{j\}} R_t \geq Q$ then $S \Leftarrow S \setminus \{j\}, F \Leftarrow F \cup \{j\}$.
Stage 3. Adopt the exchange strategy to improve the quality of the solution further.
 Step 3.1 For each $k \in F \cup N$, add shelf $k$ to $S$ and delete the redundant shelves from $S$.
     $SS(k) = S \cup \{k\}$;
     $TF = \varnothing$;
     $Tcost = 0$;
     For every $j \in SS(k) \setminus \{k\}$
       If $\sum_{t \in SS(k) \setminus \{j\}} R_t \geq Q$,
         $SS(k) \Leftarrow SS(k) \setminus \{j\}, TF \Leftarrow TF \cup \{j\}$
         $Tcost = Tcost + w_j$
       If $Tcost > w_k$
         $S = SS(k)$;
         $Cost = Cost + w_k - Tcost$;
Output: $S$, $Cost$.
```

3.3 Complexity of the algorithm

Theorem 1. *The time complexity of the hybrid heuristic algorithm is* O(MK^3).

Proof. By simple calculation, we know that Step 1.1 runs KM times, and Step 1.2 and Step 1.3 in each iteration run $(2M + 1)K$ times and $2M + K$ times, respectively. Note that the maximum number of iterations in Stage 1 is no more than K; then the total complexity in Stage 1 is no more than $KM + ((2M + 1)K + (2M + K))K = \mathrm{O}(MK^2)$. By similar arguments, we know that the complexity of Stage 2 is O(MK^2), and the complexity of Stage 3 is O(MK^3). Therefore, the total time complexity of the hybrid heuristic algorithm is O(MK^3).

Theorem 2 *Let x^0 be the solution obtained from Stage 1 of the hybrid heuristic algorithm and x^* be the optimal solution of problem (IP). We have*

$$\frac{\sum_{j=1}^{K} w_j x_j^0}{\sum_{j=1}^{K} w_j x_j^*} \le q_{\max} \cdot H(g)$$

where $g = \max_j \left(c_j\right) = \max_j \left(\sum_{i=1}^{M} u_{ij}\right)$, $H(g) = \sum_{i=1}^{g} \frac{1}{i}$ *and* $q_{\max} = \max_{1 \le i \le M} q_i$.

Proof: Since $p_{ij} = \min \{a_{ij}, q_i\}$, the problem (IP) is equivalent to the following 0-1 linear programming model (IP-1):

$$\text{(IP-1)} \quad \begin{aligned} & min\ \ z = \sum_{j=1}^{K} w_j x_j \\ & s.t. \begin{cases} \sum_{j=1}^{K} p_{ij} x_j \ge q_i & i = 1,2...,M \\ x_j \in \{0,1\} & j = 1,2,...,K \end{cases} \end{aligned}$$

Its relaxed linear programming model is as follows:

$$\text{(LP)} \quad \begin{aligned} & min\ \ z = \sum_{j=1}^{K} w_j x_j \\ & s.t. \begin{cases} \sum_{j=1}^{K} p_{ij} x_j \ge q_i & i = 1,2...,M \\ x_j \ge 0 & j = 1,2,...,K \end{cases} \end{aligned}$$

The dual of (LP) is:

$$\text{(DP)} \quad \begin{aligned} & max\ f = \sum_{i=1}^{M} q_i y_i \\ & s.t. \begin{cases} \sum_{i=1}^{M} p_{ij} y_i \le w_j & j = 1,2...,K \\ y_i \ge 0 & i = 1,2,...,M \end{cases} \end{aligned}$$

Assume that the solution x^0 is found at iteration s of Stage 1. This means that there are s shelves selected corresponding to the solution x^0. Let the index of the selected shelf in iteration r be denoted by (r).

When all quantity of goods i in the batch of orders can be taken from the selected shelves, we say that goods i is completely picked.

Assume that all items are completely picked according to the type of goods ordered 1, 2, …, M, which means that goods 1, 2, …, i-1 are completely picked before the type of goods i is completely picked. Therefore, before the last item of goods i is picked, the number of types of goods that are completely picked on shelf *j* is at most *i*-1. Then, the number of types of goods that can continue to be picked from shelf *j* is at least $c_j - i + 1$.

Let q_i^r be the quantity of goods *i* that have not been picked before iteration *r* of Stage 1. Then $P^r_j = (p^r_{1j}, p^r_{2j}, \dots, p^r_{Mj})^T$ is the vector of goods' quantities that have not been picked from shelf *j* before iteration *r* of Stage 1, and $b_j^r = \sum_{i=1}^{M} p_{ij}^r$ is the total quantity of all types of goods that have not been picked from shelf *j* before iteration *r* of Stage 1.

If the last item of goods *i* is picked in iteration *r*, then the time for picking the last item of goods *i* is $\frac{w_{(r)}}{b_{(r)}^r}$. Based on the rules for selecting shelves in Stage 1, we obtain:

$$\frac{w_{(1)}}{b_{(1)}^1} \le \frac{w_{(2)}}{b_{(2)}^2} \le \dots \le \frac{w_{(r)}}{b_{(r)}^r} \text{ and } \frac{w_{(r)}}{b_{(r)}^r} \le \frac{w_j}{b_j^r},\ j = 1, 2, \dots, K.$$

Let $y_i = \dfrac{w_{(r)}}{q_{\max} H(g) b_{(r)}^r}$. We can prove that y_i is a feasible solution of (DP). Since

$$y_i = \frac{w_{(r)}}{q_{\max} H(g) b_{(r)}^r} \le \frac{w_j}{q_i H(g) b_j^r} \le \frac{w_j}{q_i H(g)(c_j - i + 1)},$$

then

$$\sum_{i=1}^{M} p_{ij} y_i \le \sum_{i=1}^{M} p_{ij} \frac{w_j}{q_i H(g)(c_j - i + 1)} = \sum_{\{i: p_{ij} > 0\}} p_{ij} \frac{w_j}{q_i H(g)(c_j - i + 1)}$$

$$\le \sum_{\{i: p_{ij} > 0\}} \frac{w_j}{H(g)(c_j - i + 1)} = \frac{w_j}{H(g)} H(c_j) \le w_j$$

and

$$\sum_{i=1}^{M} q_i y_i = \sum_{i=1}^{M} q_i \frac{w_{(r)}}{q_{\max} H(g) b_{(r)}^r} = \frac{1}{q_{\max} H(g)} \sum_{i=1}^{M} q_i \frac{w_{(r)}}{b_{(r)}^r} \ge \frac{1}{q_{\max} H(g)} \sum_{r=1}^{s} w_{(r)}.$$

From the dual theory, we have that

$$\sum_{i=1}^{M} q_i y_i \le OPT_{LP}.$$

Note that $OPT_{LP} \le OPT_{IP-1}$, $\sum_{j=1}^{K} w_j x_j^0 = \sum_{r=1}^{s} w_{(r)}$ and $\sum_{j=1}^{K} w_j x_j^* = OPT_{IP-1} = OPT_{IP}$, we have

$$\frac{\sum_{j=1}^{K} w_j x_j^0}{\sum_{j=1}^{K} w_j x_j^*} \le q_{\max} \cdot H(g).$$

From Theorem 2, in the worst case, the ratio of the approximate solution obtained in Stage 1 over the exact solution is less than $q_{\max}$ ln(*g*). Moreover, the quality of the approximate solution can be further improved in Stage 2 and Stage 3.

4. A SIMPLE EXAMPLE

Example: There are 12 types of goods stored on 8 movable shelves, and the quantity of each of the goods on each shelf is represented as matrix *A*. Each row of matrix *A* corresponds to one type of goods. Each column of matrix *A* corresponds to a shelf.

$$A = \begin{pmatrix} 2 & 0 & 0 & 2 & 0 & 0 & 3 & 0 \\ 3 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\ 2 & 0 & 0 & 0 & 3 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 & 0 & 2 \\ 0 & 2 & 0 & 1 & 0 & 0 & 2 & 0 \\ 0 & 3 & 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 3 & 0 & 2 \\ 0 & 3 & 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 0 & 3 & 2 & 0 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 3 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 3 & 0 & 2 \end{pmatrix}$$

The round-trip moving times of the shelves are set as $W = (2\ \ 4\ \ 3\ \ 1\ \ 3\ \ 2\ \ 3\ \ 2)$. The quantities of goods in the batch of orders is $Q = (2\ \ 3\ \ 1\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 1\ \ 2\ \ 3\ \ 1)^T$.

The hybrid heuristic algorithm runs as follows:

Stage 1. Calculate the average moving time for picking one item from each shelf. The results are listed in Table I.

Table I: Average moving time for picking one item from each shelf.

Shelf	1	2	3	4	5	6	7	8
Average time	2/7	4/7	3/7	1/5	3/6	2/7	3/8	2/8

From Table I, we select Shelf 4 first and update Q and S as follows:

$Q = (0\ \ 2\ \ 1\ \ 2\ \ 1\ \ 2\ \ 1\ \ 2\ \ 0\ \ 2\ \ 3\ \ 1)^T$ and $S = \{4\}$

Recalculate the average moving time for picking one item from the remaining shelves (see Table II), and then select Shelf 8.

Table II: Average time for picking one item from the remaining shelves.

Shelf	1	2	3	5	6	7	8
Average time	2/4	4/6	3/6	3/6	2/7	3/4	2/8

Repeat these steps until $Q = (0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0)^T$. Then, we obtain a feasible solution $S = \{4, 6, 7, 8\}$ with the total round-trip moving time as 8. It is easy to check that it is also the global optimal solution of problem (IP).

5. EXTENSIVE NUMERICAL EXPERIMENTS

5.1 Random testing problems

In this section, we conduct extensive numerical experiments on testing problems generated randomly to validate the effectiveness and efficiency of the heuristic algorithm. Suppose that there are at most 10 types of goods placed on one shelf. For each type of goods, one shelf has at most 8 items. The moving time of each shelf is an integer between 1~10 generated randomly following uniform distribution.

The quantity of each type of goods on a shelf is an integer generated randomly between 1~8. The quantity of each type of goods in the order is an integer generated randomly from {0, 1, 2, 3}. For every group of parameters, we randomly generate 10 examples. We solve each example using both Lingo software, which generates the exact solution, and the hybrid heuristic algorithm. For every testing example, we record the running times of both methods and the approximate ratio of the hybrid heuristic algorithm. The average running time and approximate ratio of the two methods are the average of 10 testing examples. The worst approximate ratio is the largest approximate ratio among 10 examples. The results are summarized in Table III.

Table III: The numerical results.

M	*K*	The average running time of the heuristic algorithm (seconds)	The average running time of Lingo software (seconds)	The average approx-imate ratio of the heuristic algorithm	The worst approximate ratio
200	100	0.15	0.31	1.015	1.043
200	150	0.20	0.6	1.05	1.080
200	200	0.185	0.765	1.07	1.124
200	250	0.19	0.76	1.05	1.092
200	300	0.236	0.98	1.08	1.133
400	200	0.55	1.06	1.012	1.019
400	300	0.65	1.4	1.045	1.148
400	400	0.798	2.7	1.044	1.077
400	500	0.8	2.56	1.076	1.123
400	600	0.99	7.5	1.08	1.136
500	300	1.16	1.44	1.026	1.05
500	400	1.44	2.04	1.048	1.067
500	500	1.99	3.44	1.058	1.078
500	600	1.61	4.7	1.062	1.083
500	700	2.27	8.2	1.078	1.116
500	800	1.75	7.1	1.073	1.093
500	900	1.957	Run out of memory	-	-
800	600	8.06	Run out of memory	-	-
800	800	12.21	Run out of memory	-	-
800	1000	13.75	Run out of memory	-	-
1000	1000	21.72	Run out of memory	-	-
1000	1500	34.62	Run out of memory	-	-
1000	3000	191.98	Run out of memory	-	-

Table III shows that the average approximate ratio of the hybrid heuristic algorithm does not exceed 1.08 in all examples. The worst-case approximate ratio is no more than 1.15. This means that the hybrid heuristic algorithm is effective (i.e., the solution generated by the method is very close to the optimal solution). Furthermore, for all of the testing problems, the hybrid heuristic algorithm is faster than Lingo software (Lingo 11). Moreover, for medium- and large-scale problems, Lingo cannot obtain the optimal solution due to insufficient memory, while the hybrid heuristic algorithm can still solve the problem in a short amount of time. Therefore, the hybrid heuristic algorithm is efficient.

5.2 Comparing with the greedy algorithm for a set-covering problem

Since the problem studied in this paper is an extension of the set-covering problem, to demonstrate the effectiveness of the algorithm proposed in this paper, we compare its performance with that of the greedy algorithm stated in [23] and [27] on a typical extreme example of the set-covering problem (many researchers use this example to estimate the performance of the greedy algorithm) [23].

K types of goods are stored in $K+1$ movable shelves. For the first K shelves, only one type is placed on each shelf. The last shelf stores all types of goods. The quantities of goods stored on shelves can be expressed in the following matrix:

$$A = \begin{pmatrix} 1 & 0 & \dots & 0 & 1 \\ 0 & 1 & \dots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 1 \end{pmatrix}_{K\times(K+1)}$$

The round-trip moving time of shelves is set as $W = (1, 1/2, 1/3, \dots, 1/K, 1+\varepsilon)$, where ε is a small positive real number. The quantity of each type of goods in the order to be picked is exactly 1; $Q = (1, 1, 1, \dots, 1)^T$. It is easy to check that the optimal solution is $S = \{K + 1\}$, and the optimal value is $1 + \varepsilon$.

When this example is solved by greedy algorithms, we obtain the approximate solution $S = \{1, 2, \dots, K\}$ with the value of $H(K) = \sum_{i=1}^{K} \frac{1}{i}$ [23, 27]. However, the hybrid heuristic algorithm can get the exact solution.

We also note that if we only execute Stage 1 of the algorithm, we can also get the approximate solution $S = \{1, 2, \dots, K\}$. However, after we execute Stages 2 and 3, we can obtain the global optimal solution. Therefore, Stages 2 and 3 of the algorithm can effectively improve the solution precision and avoid falling into a local optimal solution.

6. CONCLUSION

The *cargo-to-person* picking mode is a new e-commerce warehouse-management mode. Amazon's successful application of the Kiva system illustrates the advantages of a system based on the new picking mode. To reduce picking time and improve picking efficiency, it is important to select optimal shelves for moving a given batch of orders. In this paper, we studied the optimal movable-shelf selection problem under the *cargo-to-person* picking mode. We modelled this problem using 0-1 linear programing and proved that the problem is NP-hard. Then, a hybrid heuristic algorithm with polynomial complexity is proposed for the problem. We also conduct numerical experiments to show the effectiveness and efficiency of the proposed algorithm. Based on the results summarized in this paper, additional topics need to be further studied.

In this paper, we only consider the problem of picking one batch of orders and assume that the position of each shelf in the warehouse is fixed. In reality, the orders to be picked are often partitioned into multiple batches. In the process of picking two successive batches of orders, some shelves might be moved twice. To reduce the total round-trip moving time of shelves, we should adjust the positions of shelves in the warehouse according to the information of all batches of orders. In other words, after a shelf is moved to the platform for picking one batch of orders, its position in the warehouse might be adjusted to reduce the round-trip moving time of shelves for picking the next batch of orders. In the future, we will study multi-stage optimal movable-shelf selection problems and the optimal strategy for adjusting shelves' positions.

The optimal movable-shelf selection problem discussed in this paper occurs at the operational level, which is a static problem. In fact, different processes of *cargo-to-person* warehouse systems, such as location allocation, order-batching, shelf selection, robot task assignment and path routing, are correlated. Studying the optimal movable-shelf selection problem considering other processes is an interesting topic that needs to be studied further.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (Grant No.71390334, 71540028, 71661167009) and a high-level scientific research project of Beijing Wuzi University (GJB20164005) and Beijing Logistics Informatics Research Base.

REFERENCES

[1] Qi, Y.; Tang, M.; Zhang, M. (2014). Mass customization in flat organization: The mediating role of supply chain planning and corporation coordination, *Journal of Applied Research and Technology*, Vol. 12, No. 2, 171-181, doi:10.1016/S1665-6423(14)72333-8

[2] Wurman, P. R.; D'Andrea, R.; Mountz, M. (2008). Coordinating hundreds of cooperative, autonomous vehicles in warehouses, *AI Magazine*, Vol. 29, No. 1, 9-19

[3] Hazard, C. J.; Wurman, P. R.; D'Andrea, R. (2006). Alphabet Soup: A test bed for studying resource allocation in multi-vehicle systems, *Proceedings of the 2006 AAAI Workshop on Auction Mechanisms for Robot Coordination*, 23-30

[4] Enright, J. J.; Wurman, P. R. (2011). Optimization and coordinated autonomy in mobile fulfillment systems, *Workshops at the 25th AAAI Conference on Artificial Intelligence: Automated action planning for autonomous mobile robots*, 33-38

[5] Wang, Y.; De Silva, C. W. (2008). A machine-learning approach to multi-robot coordination, *Engineering Applications of Artificial Intelligence*, Vol. 21, No. 3, 470-484, doi:10.1016/j.engappai.2007.05.006

[6] Jose, K.; Pratihar, D. K. (2016). Task allocation and collision-free path planning of centralized multi-robots system for industrial plant inspection using heuristic methods, *Robotics and Autonomous Systems*, Vol. 80, 34-42, doi:10.1016/j.robot.2016.02.003

[7] Liu, S.; Sun, D.; Zhu, C. (2014). A dynamic priority based path planning for cooperation of multiple mobile robots in formation forming, *Robotics and Computer-Integrated Manufacturing*, Vol. 30, No. 6, 589-596, doi:10.1016/j.rcim.2014.04.002

[8] Li, Z.; Li, W.; Jiang, L. (2016). Task assignment problem of robots in a smart warehouse environment, *Management Studies*, Vol. 4, No. 4, 167-175, doi:10.17265/2328-2185/2016.04.004

[9] Li, Z.; Li, W. (2015). Mathematical model and algorithm for the task allocation problem of robots in the smart warehouse, *American Journal of Operations Research*, Vol. 5, No. 6, 493-502, doi:10.4236/ajor.2015.56038

[10] Gu, J.; Goetschalckx, M.; McGinnis, L. F. (2007). Research on warehouse operation: A comprehensive review, *European Journal of Operational Research*, Vol. 177, No. 1, 1-21, doi:10.1016/j.ejor.2006.02.025

[11] Davarzani, H.; Norrman, A. (2015). Toward a relevant agenda for warehousing research: literature review and practitioners' input, *Logistics Research*, Vol. 8, No. 1, 18 pages, doi:10.1007/s12159-014-0120-1

[12] Gong, Y.; de Koster, R. B. M. (2011). A review on stochastic models and analysis of warehouse operations, *Logistics Research*, Vol. 3, No. 4, 191-205, doi:10.1007/s12159-011-0057-6

[13] Pan, J. C.-H.; Shih, P.-H.; Wu, M.-H. (2015). Order batching in a pick-and-pass warehousing system with group genetic algorithm, *Omega*, Vol. 57, Part B, 238-248, doi:10.1016/j.omega.2015.05.004

[14] Öncan, T. (2015). MILP formulations and an iterated local search algorithm with tabu thresholding for the order batching problem, *European Journal of Operational Research*, Vol. 243, No. 1, 142-155, doi:10.1016/j.ejor.2014.11.025

[15] Menéndez, B.; Pardo, E. G.; Alonso-Ayuso, A.; Molina, E.; Duarte, A. (2017). Variable neighborhood search strategies for the order batching problem, *Computers & Operations Researc*h, Vol. 78, 500-512, doi:10.1016/j.cor.2016.01.020

[16] Henn, S. (2015). Order batching and sequencing for the minimization of the total tardiness in picker-to-part warehouses, *Flexible Services and Manufacturing Journal*, Vol. 27, No. 1, 86-114, doi:10.1007/s10696-012-9164-1

[17] Lu, W.; McFarlane, D.; Giannikas, V.; Zhang, Q. (2016). An algorithm for dynamic order-picking in warehouse operations, *European Journal of Operational Research*, Vol. 248, No. 1, 107-122, doi:10.1016/j.ejor.2015.06.074

[18] Hong, S.; Kim, Y. (2017). A route-selecting order batching model with the S-shape routes in a parallel-aisle order picking system, *European Journal of Operational Research*, Vol. 257, No. 1, 185-196, doi:10.1016/j.ejor.2016.07.017

[19] Schrotenboer, A. H.; Wruck, S.; Roodbergen, K. J.; Veenstra, M.; Dijkstra, A. S. (2016). Order picker routing with product returns and interaction delays, *International Journal of Production Research*, 13 pages, doi:10.1080/00207543.2016.1206982

[20] Chen, T.-L.; Cheng, C.-Y.; Chen, Y.-Y.; Chan, L.-K. (2015). An efficient hybrid algorithm for integrated order batching, sequencing and routing problem, *International Journal of Production Economics*, Vol. 159, 158-167, doi:10.1016/j.ijpe.2014.09.029

[21] Cheng, C.-Y.; Chen, Y.-Y.; Chen, T.-L.; Yoo, J .J.-W. (2015). Using a hybrid approach based on the particle swarm optimization and ant colony optimization to solve a joint order batching and picker routing problem, *International Journal of Production Economics*, Vol. 170, Part C, 805-814, doi:10.1016/j.ijpe.2015.03.021

[22] Yang, J.; Leung, J. Y.-T. (2005). A generalization of the weighted set covering problem, *Naval Research Logistics*, Vol. 52, No. 2, 142-149, doi:10.1002/nav.10093

[23] Chvatal, V. (1979). A greedy heuristic for the set-covering problem, *Mathematics of Operations Research*, Vol. 4, No. 3, 233-235

[24] Garfinkel, R. S.; Nemhauser, G. L. (1972). *Integer Programming*, John Wiley and Sons, New York

[25] Aickelin, U. (2002). An indirect genetic algorithm for set covering problems, *Journal of the Operational Research Society*, Vol. 53, No. 10, 1118-1126, doi:10.1057/palgrave.jors.2601317

[26] Umetani, S.; Yagiura, M. (2007). Relaxation heuristics for the set covering problem, *Journal of the Operational Research Society of Japan*, Vol. 50, No. 4, 350-375

[27] Sundar, S.; Singh, A. (2012). A hybrid heuristic for the set covering problem, *Operational Research*, Vol. 12, No. 3, 345-365, doi:10.1007/s12351-010-0086-y

[28] Wang, J. (2013). Model and algorithm for set covering problem (in Chinese), *Computer Engineering and Applications*, Vol. 49, No. 17, 15-17