International Journal of Industrial Engineering, 31(1), 46-64, 2024

INTEGRATING INTEGER PROGRAMMING WITH HEURISTIC ALGORITHM TO SOLVE THE WAREHOUSE RELOCATION PROBLEM OF AUTOMATED STORAGE AND RETRIEVAL SYSTEMS WITH MULTIPLE LOADING DEPOTS

Yung-Chia Chang[1], Kuei-Hu Chang[2,*], Hsien-Mi Meng[3], and Chih-Han Chang[1]

[1]Department of Industrial Engineering and Management
National Yang Ming Chiao Tung University
Hsinchu, Taiwan

[2]Department of Management Sciences
R.O.C. Military Academy
Kaohsiung, Taiwan
[*]Corresponding author's e-mail: evenken2002@yahoo.com.tw

[3]Department of Industrial Engineering and Systems Management
Feng Chia University
Taichung, Taiwan

The primary focus of the warehouse relocation problem is to systematically and efficiently move items already stored in the warehouse to newly planned storage locations, which is a critical issue for the successful operation of Automated Storage and Retrieval Systems (AS/RS). There is limited research on the warehouse relocation problem for AS/RS with multiple loading and unloading stations. To address this issue, this research proposed an approach to rearrange the materials stored in a unit-load AS/RS with multiple loading stations to a target assignment. A mathematical model based on integer programming was built to determine the optimal sequence for relocating storage locations and arrange the sequence of movement. Since this is at least NP-hard to solve, a heuristic algorithm was designed to solve this problem in several segments, enabling its application in practical scenarios with larger data scales to enhance the operational efficiency of automated storage systems. Computational experiments were conducted using various problem sizes to assess the performance of the proposed algorithm. Additionally, this study applied the proposed method to plan storage relocation for a large-scale automated AS/RS, which is operational at a computer hardware manufacturer in Taiwan, as a means to verify the feasibility and effectiveness of the method proposed in this research.

Keywords: Mathematical Programming; Heuristic Algorithm; Relocation Problem; Automated Storage and Retrieval Systems; Multiple Input/Output Stations

(Received on November 29, 2022; Accepted on February 3, 2024)

1. INTRODUCTION

With the upsurge of Industry 4.0, Artificial Intelligence (AI), and the Internet of Things (IoT), numerous manufacturers have started to construct smart factories. These smart factories possess smart production and intelligent logistic facilities. Smart production generally relates to the production logistics management of the enterprise, human-machine interaction, and 3D technology in the production process. In terms of intelligent logistics, IoT is integrated with logistics resources to increase in-plant material transport efficiency. Intelligent logistics is an integral part of planning the smart factory blueprint. How to increase the storage/retrieval and transport efficiency of materials and reduce the tying-down of capital and inventory are the key factors influencing enterprises' competitiveness. Intelligent logistics is built by combining IoT technology and Big Data analysis with AS/RS. It can replace traditional logistics by increasing delivery efficiency. Consumers can receive goods in a shorter time, which enhances the enterprise's competitiveness. Intelligent logistics covers in-plant material transfer, material purchase, and end-product delivery. It can be divided into internal and external logistics. Internal logistics refers to in-plant material transport and automatic guided vehicles (AGV). AGV distribution system is combined with a warehouse management system or manufacturing execution system. The sensors, computer, and wireless system are connected to the

AGV so that the system can guide the AGV to move the work-in-process (WIP) to the next production stage. This simplifies manual operation and reduces WIP inventory. In recent years, many companies use a just-in-time (JIT) production strategy to increase manufacturing efficiency and reduce waste and inventory costs. This means the production depends on orders. To implement a JIT production strategy and on-time delivery, external logistics must be well-coordinated with in-plant production flow. This simplifies the operation flow and is a further step into a smart factory.

Intelligent logistics is essential for implementing smart factories. Intelligent logistics guarantee smooth production flow while reducing WIP inventory and increasing customer satisfaction. In-plant intelligent logistics is a basic requirement for smart factories. Before the popularity of AS/RS, most factories used manpower to handle materials. Human handling has the potential to damage the materials. The materials could also be placed in the wrong locations due to human mistakes. Before the popularity of automated warehousing, the five key issues resulting from smart warehousing were discussed. The five key issues include warehouse inventory management, storage location layout, logistics efficiency, manpower allocation, and equipment operation. Recent research aims to optimize and predict the operation of warehousing systems using Big Data analysis. The warehousing system can coordinate with the product nature, supply chain management, and marketing policy of enterprises. After optimizing the operation, the system can be used with other intelligent logistics equipment. The in-plant logistics system is upgraded and developed toward intelligentization (e.g., Li *et al*., 2018; Wang *et al*., 2020; Zhou and Huang, 2022; Chen *et al*., 2022; Yang *et al*., 2022).

Many enterprises' production modes have changed from large-volume production with little variety to small-volume production with a wide range of varieties. When the AS/RS is used for storing the parts for production and semi-finished products, the frequency of pickup increases, but the quantity decreases. The material storage location assignment method is very important for AS/RS. The quality of placement principle will influence the overall operation of the system in the future. The storage location assignment problem is an NP-Hard problem (Frazelle and Sharp, 1989). Some scholars used mathematical modeling methods such as branch-and-bound (Muppani and Adil, 2008) and dynamic programming (Dijkstra and Roodbergen, 2017; Sharma *et al*., 2020) to solve this kind of problem. Some scholars used a meta-heuristic to solve the storage location assignment problem. For example, He *et al*. (2019) used particle swarm optimization (PSO) to solve the storage location assignment problem, Jiao *et al*. (2018) and Wang *et al*. (2019) used Genetic Algorithm (GA) to optimize the storage location assignment problem.

The method of storage location assignment should be related to actual demand. When the stored product item varies with demand, it is necessary to consider renewing the storage location layout and efficiently renewing the storage location content to meet the adjusted storage location planning (Gu *et al*., 2007). The retrieval operation has peak and off-peak periods. The product to be retrieved in the future is moved to a storage location near the exit during the off-peak period in advance. This increases the retrieval operating efficiency during the peak period. The problem of moving the materials stored in the warehouse to a new storage location is called the warehouse relocation problem. This concept was first proposed by Christofides and Colloff (1973). They assumed rearranging the storage locations in a warehouse with a single loading station and proposed a warehouse relocation algorithm that could minimize the total moving cost. Pazour and Carlo (2015) discussed quantizing the warehouse relocation problem for the warehousing system with a single loading station. They built a mathematical model for this problem and then designed several heuristics to solve the problem after relaxing some of the assumptions. Another research area relatively close to warehouse relocation is called the Block Relocation Problem (BRP). Containerized traffic consumes a lot of time and manpower for container handling operations at the wharf. This problem studies how to prearrange the containers according to the order of pickup to shorten the operating time of ships at the port. In turn, this helps the wharf attract more shipping companies. The BRP can minimize the container handling frequency by adjusting the unloading order or improving the method (Jin *et al*., 2015; Ting and Wu, 2017; Bacci *et al*., 2019; Tanaka and Mizuno, 2018; Azab and Morita, 2022; Liu *et al*., 2022). To shorten the ships' berth time, the BRP can shorten the pick up time and reduce the handling frequency by pre-marshaling the containers.

There is few research studying on the warehouse relocation problem for AS/RS. Among those, almost all of them discussed the problem with single loading/unloading station; none of them discussed AS/RS with multiple loading/unloading stations. The related research is reviewed in Section 2. This study aimed at the warehouse relocation problem of unit loading AS/RS with multiple loading/unloading stations. To be more precise, this paper delves into the warehouse relocation problem, drawing inspiration from a real-world AS/RS scenario. It addresses the challenge of relocating items within a warehouse while assuming that the current storage locations are known and the target storage locations have been identified. An integer programming model was presented to minimize the relocation cost. Given the inherent complexity of this problem, we propose a heuristic approach to enhance the solvability of larger instances. The following sections of this paper are arranged as follows. Section 2 is the preliminaries, introducing the research about block relocation and warehouse relocation problems. Section 3 shows the Research Method and Process, details the research problem, application context, and context assumption, builds a mathematical programming model, develops the warehouse relocation algorithm, and describes the decoding process. Section 4 shows the implementation and the result analysis, the performance of the warehouse relocation algorithm in various storage location backgrounds, and the application of the method in different scenarios. Section 5 shows the Conclusion and

Suggestions, detailing the conclusion and contribution of this study, and relevant suggestions are provided based on the results.

2. PRELIMINARIES

2.1 Block Relocation

The commonly seen materials management model is wharf container warehousing and the most familiar acquisition method for international materials. When the materials arrive at the port, the first step is to apply to the customs of the local port authority for unloading. Upon approval, the forwarder or container administrator is notified for unloading, storing, and incoming/outgoing. The importer sends transport vehicles to the wharf to get the containers of his required imported raw materials. The cranes are assigned to handle containers. The number of cranes is limited, and the containers are usually stacked in batches. The forwarder must minimize the container moving frequency to optimize the service efficiency of cranes and the waiting time of transport vehicles.

Ting and Wu (2017) combined the beam search algorithm with the smallest difference heuristic and virtual relocation heuristic. Their purpose was to solve a relatively large-scale block relocation problem within a short time while guaranteeing the accuracy of the result. They also aimed to maximize the manufacturing efficiency of a wharf, reduce costs, and use the warehouse space to the maximum extent. Bacci *et al*. (2020) used the Bounded Beam Search Algorithm to work out the initial feasible solution. It was combined with the Branch and Bound method to generate inequality and make the solving process faster and more efficient. The mathematical model was combined with the Branch and Cut (BC) method/algorithm to work out the preferred plan of each Node in the Branch and Cut Tree using linear relaxation. This method is called BC-RBRP. In addition, a partial constraint of the original model is relaxed when this method is used. Some of the original inequalities are deleted, and the Cutting Plane Strategy is used to check whether the current solution conforms to the constraints. The infeasible integer solution was eliminated to guarantee the feasibility of the integer linear programming solution.

Quispe *et al*. (2018) compared two new methods for calculating the lower bound. These are LB-LIS and LB-PDB. All of the containers were moved according to the order priority. The goal was to minimize the total handling frequency. Compared to the lower bounds of Zhu *et al*. (2012) and Tanaka and Takii (2016), the algorithm could work out lower handling frequency using LB-LIS. Jovanovic and Voss (2014) used the improved Min-Max Heuristic algorithm to determine the movement of the container based on the next container. The Min-Max Heuristic helps preferentially move the containers to the stack with maximum residual capacity. In this condition, it is unnecessary to create a constraint of not placing the container on a full stack. In addition, whether the maximum capacity constraint will be reached if a stack is moved to another stack, the result shows that the degree of improvement increases with the container size. Compared to the traditional Min-Max, the large-scale problem can be improved by 5%-8%. This improvement method needs a little extra computing time. Jovanovic *et al*. (2019) integrated the Ant Colony Optimization into the Greedy Algorithm to improve the greedy algorithm's search speed and solve the restricted and unrestricted BRP. A small part of the information is stored. It is unnecessary to consider the whole Bay State.

Ku and Arthanari (2016a) proposed a search area-reducing algorithm of BRP (abstraction method) to improve the solution time of the search-based exact method. The abstraction method works in both forward and backward search directions in the search space. Ku and Arthanari (2016b) minimized the expected number of reshuffles for containers based on the truck appointment system in port container terminals by stochastic dynamic programming. They developed exact methods and also heuristics to improve the computational performance. Zehendner *et al*. (2017) proposed the online container relocation problem for the events without prior information. The "worst condition" and "average performance" of minimum container moving frequency were analyzed using the leveling heuristics. Galle *et al*. (2018) improved the existing binary coding and formulated the restricted BRP into a binary coding problem. The number of variables and constraints was reduced significantly compared to the existing equation.

The BRP can shorten the pick up time and reduce the handling frequency by pre-marshaling containers to shorten the ships' berth time. Caserta *et al*. (2012) used a mathematical model to calculate the complete feasible region of BRP and assumed the reduced feasible solution region to minimize the moving total. The intermediate and small-scale problems were solved. In their method, a simple heuristic rule was applied. The heuristic solution was compared with the best solution in a group of randomly generated instances to achieve rapid pickup (container placement priority), reduce the pickup cost, and increase the wharf efficiency. Scholl *et al*. (2018) proposed 12 prediction methods for pre-marshaling problems. The containers were put in the port to reduce the subsequent container relocation frequency based on the predictions. Da Silva *et al*. (2018) built a new integer programming model in the BRP and redefined the priority level of each container. The result showed that the new model had a shorter computing time and higher efficiency. Zweers *et al*. (2020) proposed B&B to work out the near-optimum solution. The computing time was too long. Therefore, they proposed the BB-H heuristic algorithm to work out a feasible solution within a finite time.

2.2 Warehouse Relocation

Manual order picking is a job that consumes different costs. These costs could be time, financial, labor, and manual picking. These costs may exceed half of a warehouse's total operating expenses. If the order-picking efficiency is low, the service quality of the overall supply chain may be degraded, and the operating expense can rise. With the popularization of automated equipment, the AS/RS can substitute traditional warehouses. The work hours of material storage and retrieval, manpower, and costs can be reduced by warehouse management and material moving. An important decision in the warehouse and delivery center is to plan the optimal storage location of materials, known as the storage location assignment problem (SLAP). Hausman *et al*. (1976) first proposed three storage location assignment strategies: the full turnover-based assignment, random assignment, and class-based assignment. The full turnover-based assignment is based on material turnover. This means that the material with higher turnover is placed near the loading station, and that with low turnover is placed farther. Random assignment means randomly selecting the material storage location. The class-based assignment effectively classifies the storage areas and materials based on customer requirement rates. The materials are stored in their classes. Pang and Chan (2017) used data mining to analyze the correlation between different products in the customer orders. The distance of replenishment was minimized, and the total travel distance of order picking was shortened. Wang *et al*. (2020) proposed a method to improve the order-picking operation. The order data were directly used for decision-making without any statistical treatment. The algorithm was developed to minimize the total travel distance.

Many strategies for solving the SLAP are decisions based on material requirements. These strategies vary with time. Requirements may change due to competition, product maturity, or product substitution. This will impact the optimum allocation of locations. Generally, the warehouse utilizes the off-peak time or works overtime to relocate materials from their current locations to the recommended ones. This practice is called warehouse relocation. The goal is to move from the initial storage location to the target storage location using the minimum travel time or distance. When the materials to be relocated and their target storage locations are known, the materials could be classified as cycle and non-cycle materials. The material that cannot be moved to the target storage location until other storage locations are moved is called cycle material; otherwise, it is non-cycle material. For example, as shown in Figure 1 (Pazour and Carlo, 2015), the initial storage location (a) needs to be rearranged to the target storage location (b). In this example, A and B are cycle materials, and C and D are non-cycle materials. The materials requiring the interchange of storage locations to reach the target storage locations are listed in the same cycle group. As shown in Figure 2 (a), there are three cycle groups; each group should be moved before the target storage location in Figure 2 (b) is reached (Carlo and Giraldo, 2012).

Carlo and Giraldo (2012) proposed the Rearrange-While-Working (RWW) strategy to optimize the process of warehouse relocation. They studied the following three conditions: (1) there is a vacancy in the warehouse, and the material handling equipment (MHE) is idle; (2) there is a vacancy in the warehouse under the RWW strategy; (3) there are multiple vacancies in the warehouse under the RWW strategy. In the first condition, the MHE can perform any required movement during idle time. In the other two conditions, the MHE is assumed to be non-idle, and the MHE cannot move until movement is required. They proposed multiple heuristic algorithms to minimize the relocation cost and shorten the computing time.

Loc 3 A	Loc 4 B	Loc 5 C
Loc 0 D	Loc 1 0_1	Loc 2 0_2

(a)

Loc 3 B	Loc 4 A	Loc 5 $0_{2'}$
Loc 0 $0_{1'}$	Loc 1 D	Loc 2 C

(b)

Figure 1. Storage rearrangement example

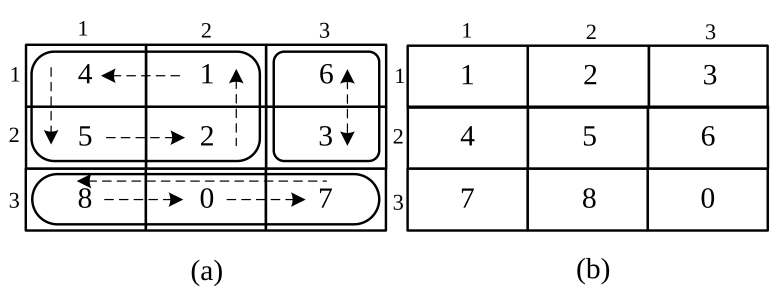


Figure 2. Initial storage for the three cycles

Chen *et al*. (2011) discussed the relocation problem in a warehousing system with dynamic storage operating in single command mode. They proposed a two-stage heuristic method in which an initial solution was generated and then improved by a tabu search. In their problem, the items to be relocated and their relocation destination are decision variables. Hence, they do not need to consider cycles. Pazour and Carlo (2015) studied the optimization of the warehouse relocation problem and quantized the influence of common assumptions. They built a mathematical programming model and designed heuristics under different circumstances and showed that their approach could reduce the average cost of relocation.

3. RESEARCH METHOD AND PROCESS

3.1 Problem description

This study took a unit-load AS/RS with multiple loadings as the subject. It aimed to develop a warehouse relocation algorithm so that the travel time of the stacker crane is minimized. The schematic diagram of a unit-load AS/RS with multiple loading stations is shown in Figure 3. There are Y racks, and each row of racks comprises vertical Z layers and horizontal X layers. The entire warehousing system has X*Y*Z storage locations. Each row of racks has only one stacker crane in charge of retrieval/storage operation, but each stacker crane can take charge of one row of racks (Figure 3(a)) or two rows of racks (Figure 3 (b)). The practice proposed in this study is described below, taking the AS/RS of this specification as an example.

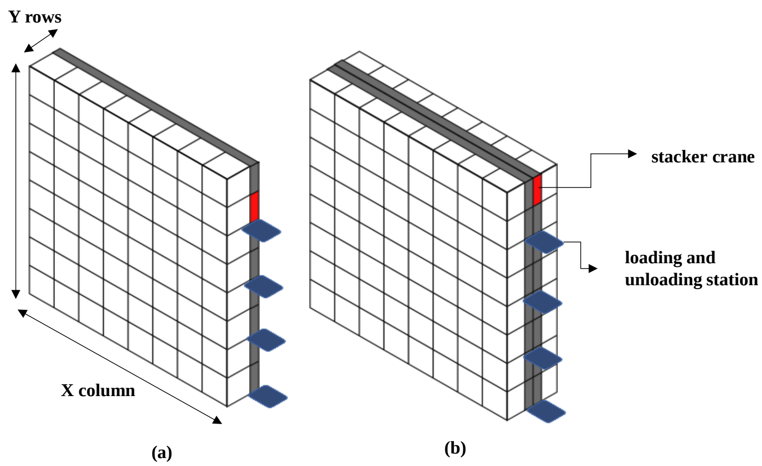


Figure 3. Schematic diagram of grid configuration of automated storage system

The warehouse relocation algorithm developed in this study can be applied in the following three scenarios:

(A) Exercise a storage location assignment in a running AS/RS with no specific storage assignment plan:
For a running AS/RS which has no specific storage assignment plan, applying the proposed method could help to generate a plan to move goods already stored in the warehouse to their planned location while minimizing the time to make such adjustments.

(B) Renew the storage location assignment of an AS/RS as requirement changes:
The material materials requirement condition may change over time due to competition, product maturity, or product substitution and the like. Therefore, the optimum allocation of locations varies with time. The proposed approach can be applied to respond to the material requirement variation.
with time. The proposed approach can be applied to respond to the material requirement variation.

(C) Marshal storage locations in advance according to requirements during off-peak hours:
In practice, the demands in various periods are different. The stacker crane may be very busy to satisfy the storage/retrieval requirements during peak hours. Using the proposed algorithm, materials to be retrieved could be relocated to locations close to the loading stations during off-peak hours to that the stacker crane could operate efficiently during peak hours.

The assumptions made to the study problem are outlined below:

(1) The stacker crane executes single command cycle only. It departs from one loading station for the specified storage location to store or retrieve the material. It then returns to a loading station, which may be the same or different from

the original departure station. Figure 4 shows an example of the layout of one layer of racks in the warehouse. The stacker crane is located in loading station_A. Let C_i and C_j represent the travel times of the stacker crane from the loading station_A to the storage location *i* and *j*, respectively. In the mode of single command cycle, suppose the materials in the storage location *i* need to be relocated to storage location *j*, the stacker crane needs to depart from the loading station_A, travel to storage location *i*, pick up the materials in the storage location *i* and then return to the loading station_A. Then the stacker crane needs to again depart from the loading station_A, travel to storage location j, place the material in the storage location j and the return to loading station_A. The total travel time of this stacker crane to complete this relation is by $2(C_i+C_j)$, in this example.

(2) Each loading station of AS/RS has a temporary storage area available for one pallet.
(3) The storage location assignment is given, meaning which type of materials need to be placed in which storage locations have been determined and named as “target storage location assignment”.
(4) The storage locations to be rearranged are given and named as “current storage location assignment”.
(5) The travel distance and time of the stacker crane between storage locations are known.
(6) One stacker crane is in charge of one or two rows of racks.
(7) The racks have multiple loading stations for input/output.
(8) The AS/RS is with single-deep racks.
(9) The quantities of different materials in the current and target storage locations could be different.

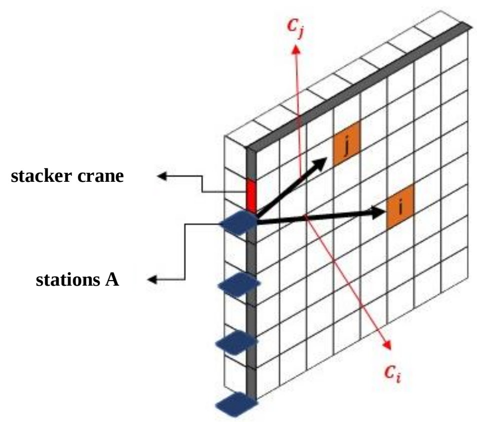


Figure 4. The configuration diagram of one-layer shelf

3.2 Building of mathematical model

Based on the above-mentioned assumptions, this study built an integer programming model for this problem. This model is called Model A. The symbols of the decision variables and parameters for building Model A are compiled in Table 1.

Table 1. Description of the symbols of variables and parameters used in model A

Symbol	Explanation
Set	
L	Set of all storage locations, $L = \{0,1,2,\dots,l\}$
R	Set of loading and unloading stations, $R = \{1,2,\dots,r\}$
H	Set of materials, $H = \{1,2,\dots,h\}, H = HU \cup HL$
HU	Set of materials, $U \coloneqq \{H \mid NI_k \geq NM_k\}, k \in H$
HL	Set of materials, $HL \coloneqq \{H \mid NI_k < NM_k\}, k \in H$
Parameter	
$C_{i,j,g}$	The time that stacker crane travels from storage location *i* to storage location *j* while using station g for loading/unloading.
$F_{i,k}$	If the target storage location *i* stores materials *k*, it is 1; otherwise, it is 0
NI_k	The number of storage locations material *k* occupied in the current storage location
NM_k	The number of storage locations material *k* occupied in the target storage locations

Symbol	Explanation
T	Total number of moves
Decision variables	
$x_{i,j,k,t}$	During the *t*th move, if the stacker crane moves the material *k* from location *i* to location *j*, it is 1; otherwise, it is 0
$IO_{g,t}$	During the *t*th move, if the loading and unloading station *g* is used, it is 1; otherwise, it is 0
$Q_{i,k,t}$	After the *t*th move, if the location *i* stores the material *k*, it is 1; otherwise, it is 0
$O_{i,t}$	After the *t*th move, if location *i* is empty, it is 1; otherwise, it is 0
$E_{i,j,t}$	After the *t*th move, if the stacker crane move from location *i* to location *j* without carrying any material, it is 1; otherwise, it is 0
$IOO_{g,t}$	After the *t*th move, if the stacker crane passes through the loading and unloading station *g* without carrying any material, it is 1; otherwise, it is 0

Model A is described as follows.

Minimize

$$\sum_{i}\sum_{j}\sum_{k}\sum_{g}\sum_{\substack{t\\t\neq 0}} x_{i,j,k,t} * C_{i,j,g} * IO_{g,t} + \sum_{i}\sum_{j}\sum_{g}\sum_{\substack{t\\t\neq T}} E_{i,j,t} * C_{i,j,g} * IOO_{g,t} \tag{1}$$

Subject to

$$\sum_{i\in L}\sum_{\substack{j\in L\\j\neq i}}\sum_{k\in H} x_{i,j,k,t} \leq 1, \quad \forall t = 1,2,3 \dots T \tag{2}$$
$$x_{i,j,k,t} \leq O_{j,t-1}, \ \forall i,j \in L, i \neq j, \forall k \in H, \forall t = 1,2,3 \dots T \tag{3}$$
$$x_{i,j,k,t} \leq O_{i,t}, \ \forall i,j \in L, i \neq j, \forall k \in H, \ \forall t = 1,2,3 \dots T \tag{4}$$
$$\sum_{\substack{i\in L\\i\neq j}} x_{i,j,k,t} \leq Q_{j,k,t}, \forall j \in L, \forall k \in H, \forall t = 1,2,3 \dots T \tag{5}$$
$$Q_{i,k,t} + \sum_{\substack{j\in L\\j\neq i}} x_{i,j,k,t} \leq 1, \forall i \in L, \forall k \in H, \forall t = 1,2,3 \dots T \tag{6}$$
$$Q_{i,k,t-1} \leq Q_{i,k,t} + \sum_{\substack{j\in L\\j\neq i}} x_{i,j,k,t}, \forall i \in L, \forall k \in H, \forall t = 1,2,3 \dots T \tag{7}$$
$$O_{it} + \sum_{k\in H} Q_{i,k,t} = 1, \forall i \in L, \forall t = 1,2,3 \dots T \tag{8}$$
$$\sum_{i\in L} Q_{i,k,t-1} = \sum_{i\in L} Q_{i,k,t}, \forall k \in H, \forall t = 1,2,3 \dots T \tag{9}$$
$$Q_{i,k,T} \geq F_{i,k}, \forall i \in L, \forall k \in HU \tag{10}$$
$$\sum_{i\in L} |F_{i,k} - Q_{i,k,T}| = NM_k - NI_k, \forall k \in HL \tag{11}$$
$$\sum_{i\in L}\sum_{\substack{j\in L\\j\neq i}}\sum_{k\in H} x_{i,j,k,t-1} \geq \sum_{i\in L}\sum_{\substack{j\in L\\j\neq i}}\sum_{k\in H} x_{i,j,k,t}, \forall t = 2,3,4 \dots T \tag{12}$$
$$\sum_{g\in R} IO_{g,t} = 1, \forall t = 1,2,3 \dots T \tag{13}$$
$$\sum_{g\in R} IOO_{g,t} = 1, \forall\ t = 0,1,2, \dots T-1 \tag{14}$$
$$\sum_{\substack{i\in L\\i\neq a}}\sum_{k\in H} x_{i,a,k,t-1} + \sum_{\substack{j\in L\\j\neq b}}\sum_{k\in H} x_{b,j,k,t} \leq E_{a,b,t-1} + 1, \forall a,b \in L, b \neq a, \ \forall t = 2,3,4 \dots T \tag{15}$$
$$\sum_{i\in L}\sum_{\substack{j\in L\\i\neq j}} E_{i,j,t} \leq 1, \forall\ t = 0,1,2, \dots T-1 \tag{16}$$
$$E_{0,j,0} \geq \sum_{k\in H}\sum_{\substack{i\in L\\i\neq j}} x_{j,i,k,1}, \forall j \in L, j \neq 0 \tag{17}$$
$$\sum_{\substack{j\in L\\j\neq 0}}\sum_{k\in H}\sum_{\substack{i\in L\\i\neq j}} x_{j,i,k,1} = 1 \tag{18}$$
$$\begin{aligned} &x_{i,j,k,t}, IO_{g,t} \in \{0,1\} \ \ \forall i,j \in L, k \in H, g \in R, \forall t = 1,2,3 \dots T, \\ &Q_{i,k,t}, O_{i,t} \in \{0,1\} \ \ \forall i \in L, k \in H, \forall t = 0,1,2 \dots T, \\ &E_{i,j,t}, IOO_{g,t} \in \{0,1\} \ \ \forall i,j \in L, g \in R, \forall t = 0,1,2, \dots T-1 \end{aligned} \tag{19}$$

During process of warehouse relocation, the stacker crane may move with or without carrying materials. Objective Equation (1) minimizes the total travel time of the stacker crane. After each move, the stacker crane can choose to stay at its current location or move to another, described by Constraint (2). Constraint (3) ensures the destination is vacant when a stacker crane is transporting materials. Constraint (4) updates the vacant storage location after each move. Constraint (5) updates the storage location of the moved material. Constraint (6) updates the original storage location to a vacant storage location after the stacker crane transports materials to another storage location. Constraint (7) keeps the unmoved material in its original storage location. Constraint (8) ensures each storage location only stores only one type of material or is vacant. Constraint (9) makes sure the total quantity of all types of materials remains unchanged before and after the movement of the stacker crane. According to context assumption 9, the quantities of various materials in the current and target storage locations can be different. Constraint (10) aims only at the type of materials whose total number of current storage locations is greater than or equal to that of target storage locations. It ensures that the material should be placed in the target storage location after the last move. Constraint (11) aims only at the type of material whose total number of current storage locations is smaller than the total number of target storage locations. It specifies that the planned quantity of the target storage locations should be less than or equal to the current storage locations. Constraint (12) ensures that the number of movements should be as small as possible. According to Context in assumption 1, since the stacker crane exercises single command cycle, each move needs to pass through one loading/unloading station. Constraints (13) and (14) ensure that the stacker crane can only move to one station in each move. Additionally, when a stacker crane transports materials to the target storage location, it needs to move to the next storage location to pick up materials. During those moves, the stacker crane won't carry any material. Therefore, Constraint (15) records the travel mode of stacker crane while moving without carrying materials. Constraints (17) and (18) record the initial and final locations of the first movement of the stacker crane without materials. Constraint (10) states the variables are binary.

Pazour and Carlo (2015) proved that the warehouse relocation problem with single loading station is at least NP-hard. The single loading station is a special problem of the AS/RS with multiple loading stations. Therefore, the problem under discussion in this study is at least NP-hard. It means that the problem scale of Model A would grow rapidly as the number of storage locations, material types, the given number of moves, and the number of loading stations increase, and the optimum solution could not be obtained within a reasonable time. Therefore, besides directly solving Model A, this study developed a heuristics, called the warehouse relocation algorithm, to solve problems with a larger scale found in practice.

3.3 Warehouse Relocation Algorithm

The proposed heuristic employs the concept of breaking down the problem into multiple subproblems and reducing the complexity of the mathematical model by reducing the number of decision variables and constraints. To achieve this, we categorize all materials into distinct types and relocate them accordingly, treating each material type as an individual subproblem. Model B and Model C, both simpler than Model A, are specifically designed to address each subproblem. Locations that do not require relocation are omitted from the relocation plan to further streamline the problem. Additionally, instead of using the total number of moves (T) as in Model A, we now limit the required moves based on the number of available movable storage locations within the heuristic. Furthermore, in order to minimize the size of the mathematical models used in the heuristic, we exclusively consider moves involving carrying materials, while moves without carrying materials can be computed separately using an algorithm. The primary distinction between Model B and Model C lies in the total number of required moves, and Table 2 provides comprehensive definitions of the variables and parameters utilized in both Model B and Model C.

Table 2. Definitions of symbols, parameters and decision variables of Model B and Model C

Symbol	Explanation
Set	
R	Set of loading and unloading stations, $R = \{1,2,\ldots,r\}$
L_{RS}	Set of storage locations that are not compatible with the storage assignment
L_{ME}	Set of target storage locations that are empty initially
L_E	Set of storage locations that are not target and are empty initially
L_O	Set of target storage locations that are not empty (occupied by other materials) initially
Parameter	
$C_{i,j,g}$	The total travel time for a stacker crane handles relocating materials from location i to location j while using location g as the loading and unloading station
N_{RS}	The total number of storage locations that are not compatible with the storage assignment
N_{ME}	The total number of empty locations in the target assignment
N_E	The total number of empty slots in non-target slots
N_O	The total number of storage spaces for other materials

Decision variables

$x_{i,j}$	If the stacker crane moves the material from the storage location *i* to the target storage location *j*, it is 1; otherwise, it is 0
$y_{i,j}$	If the stacker crane moves the material from the target storage location *i* that needs to be vacated to the temporary storage location *j*, it is 1; otherwise, it is 0
$z_{i,j}$	If the stacker crane moves the material from the storage location *i* to the target storage location *j* that needs to be vacated, it is 1; otherwise, it is 0
$IOX_{i,j,g}$	If the stacker crane moves the material from the storage location *i* to the target storage location *j* while using station *g* for loading/unloading, it is 1; otherwise, it is 0
$IOY_{i,j,g}$	If the stacker crane transports the material from the target storage location *i* that needs to be vacated to the temporary storage location *j* while using station *g* for loading/unloading, it is 1; otherwise, it is 0
$IOZ_{i,j,g}$	If the stacker crane moves the material from the temporary storage location *i* to the target storage location *j* that needs to be vacated while using station *g* for loading/unloading, it is 1; otherwise, it is 0

Model B is described as follows.
Minimize

$$\sum_{i\in L_{RS}}\sum_{j\in L_{ME}}\sum_{g\in R} x_{i,j} * C_{i,j,g} * IOX_{i,j,g} + \sum_{i\in L_O}\sum_{j\in L_E}\sum_{g\in R} y_{i,j} * C_{i,j,g} * IOY_{i,j,g} + \sum_{i\in L_{RS}}\sum_{j\in L_O}\sum_{g\in R} z_{i,j} * C_{i,j,g} * IOZ_{i,j,g} \tag{20}$$

Subject to

$$\sum_{j\in L_{ME}} x_{ij} + \sum_{j\in L_O} z_{ij} \leq 1, \forall\, i \in L_{RS} \tag{21}$$
$$\sum_{j\in L_E} y_{ij} \leq 1, \forall i \in L_O \tag{22}$$
$$\sum_{i\in L_O} y_{ij} \leq 1, \forall\, j\in L_E \tag{23}$$
$$\sum_{i\in L_{RS}} z_{ij} = \sum_{k\in L_E} y_{jk},\ \forall\, j\in L_O \tag{24}$$
$$\sum_{i\in L_{RS}} x_{ij} \leq 1, \forall j \in L_{ME} \tag{25}$$
$$\sum_{i\in L_{RS}} z_{ij} \leq 1, \forall\, j\in L_O \tag{26}$$
$$\sum_{i\in L_{RS}}\sum_{j\in L_{ME}} x_{ij} = N_{ME} \tag{27}$$
$$\sum_{i\in L_{RS}}\sum_{j\in L_O} z_{i,j} = Min(N_E, N_O) \tag{28}$$
$$\sum_{g\in R} IOX_{i,j,g} = 1, \forall\, i \in L_{RS}, \forall j \in L_{ME} \tag{29}$$
$$\sum_{g\in R} IOY_{i,j,g} = 1, \forall\, i \in L_O, \forall j \in L_E \tag{30}$$
$$\sum_{g\in R} IOZ_{i,j,g} = 1, \forall\, i \in L_{RS}, \forall j \in L_O \tag{31}$$
$$\begin{aligned} &x_{i,j}, IOX_{i,j,g} \in \{0,1\} \quad \forall\, i \in L_{RS}, j \in L_{ME}, g \in R \\ &y_{i,j}, IOY_{i,j,g} \in \{0,1\} \quad \forall\, i \in L_O, j \in L_E, g \in R \\ &z_{i,j}, IOZ_{i,j,g} \in \{0,1\} \quad \forall\, i \in L_{RS}, j \in L_O, g \in R \end{aligned} \tag{32}$$

Model B only considered the move for transporting materials. The objective equation (20) minimized the total travel time of the stacker crane with materials. Constraint (21) means that the stacker crane could directly move to the target storage location. The target location could be a vacant storage location, or it may be occupied by other materials. If it chooses to move to the target storage location occupied by other materials, the materials in the storage location should be removed first. Constraint (22) means the removed materials must be stored in a temporary location. Constraint (23) restricts each temporary storage location to be borrowed only once. Constraint (24) ensures that the storage location after removing the materials was the target storage location. Constraint (25) and Constraint (26) ensure that the material matching the target storage location would not be changed after movement. Constraint (27) restricts the number of direct moves of the stacker crane to the target storage location as the total number of the storage locations which are vacant in the initial state in the target storage assignment. Constraint (24) makes sure that the storage location vacated after removing the materials is the target storage location. Constraint (28) restricts the number of moves of the stacker crane to the target storage location occupied by other materials in the initial state as the minimum value of the number of storage locations occupied by other materials and the number of vacant storage locations of non-target storage locations. According to Context Assumption 1, the stacker crane works with a single-cycle command, and each move passes by a loading depot. Constraints (29)-(31) mean the stacker crane could move to only one loading station while transporting materials.

Model C is very similar to Model B by removing Constraints (27) and (28) and add Constraint (33) to form Model C.

$$\sum_{i \in L_{RS}} \sum_{j \in L_{ME}} x_{i,j} + \sum_{i \in L_{RS}} \sum_{j \in L_{O}} z_{i,j} = N_{RS} \tag{33}$$

Constraint (33) restricts the total number of moves of the stacker crane to the number of storage locations to be rearranged.

The proposed heuristic called the warehouse relocation algorithm, consists of five steps, discussed below:

Step 1: Assign material relocation order.

In this heuristic, one type of material is rearranged at a time. Therefore, the first step is to arrange these materials in random order by their types.

Step 2: Determine whether to terminate this heuristic

The termination condition is that all material types have completed warehouse relocation. If this termination condition was not met, the next material type was selected.

Step 3: Compare the current and target storage location assignment

If the total number of target storage locations is greater than the one of current storage locations, then check if the materials have been relocated. If the warehouse relocation has been done, return to Step 2. Otherwise, proceed to Step 4. If the total number of current storage locations was larger, check if all the target storage locations have been loaded with the kind of material. If all of the target storage locations have been loaded with the material, return to Step 2. Otherwise, proceed to Step 4.

Step 4: Warehouse relocation

If the total number of storage locations to be rearranged was smaller than the total number of vacant and temporary storage locations where the stacker crane could move, use Model C solved the problem. Otherwise, use Model B. Afterward, arrange the loading stations obtained from traveling processes while the stacker crane was carrying materials in descending order. The corresponding traveling process without carrying materials could be found out, and the travel time in each material transport process were calculated according to this order. The loading stations passed by were recorded. This is the warehouse relocation task completed with a restricted number of moves, so Step 5 should be executed to renew the state of the storage location.

Step 5: Renew the state of the storage location

The state of storage location was renewed according to the computing result of Model B or Model C and then returned to Step 3.

Figure 5 illustrates the decision flow of the warehouse relocation algorithm, as explained in the previous steps. Initially, we must determine the sequencing order for relocating the materials. After identifying the material type to be relocated, we compare the total number of locations that need to be relocated with the total number of target locations. If the total number of target locations exceeds the total number of locations to be relocated, we apply Model B to address the problem with some parameter updates. Otherwise, Model C is utilized. As the model determines moves involving material transportation, we can subsequently compute the corresponding moves when no material is being carried.

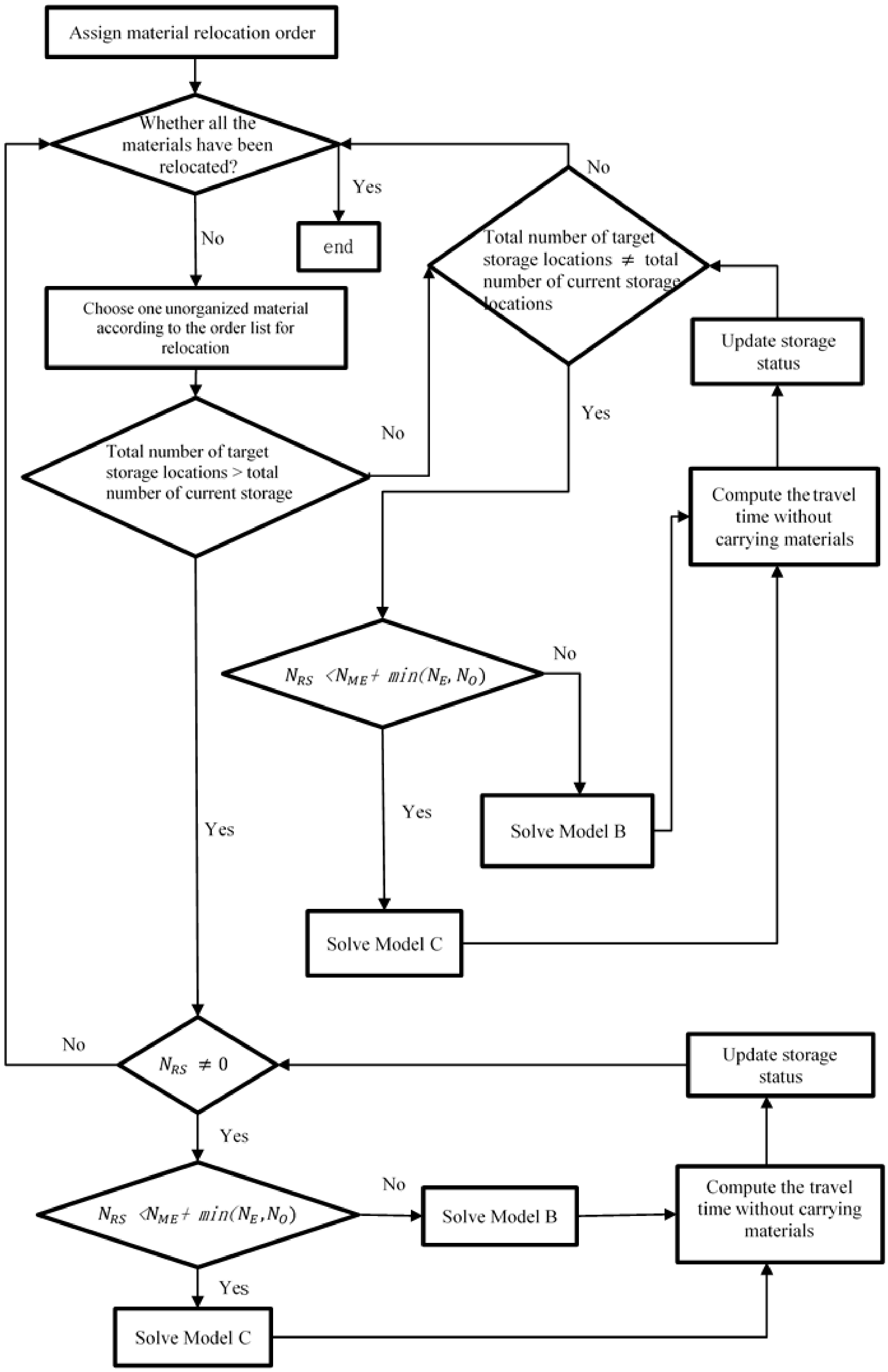
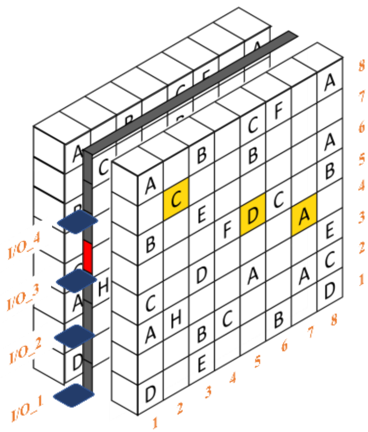


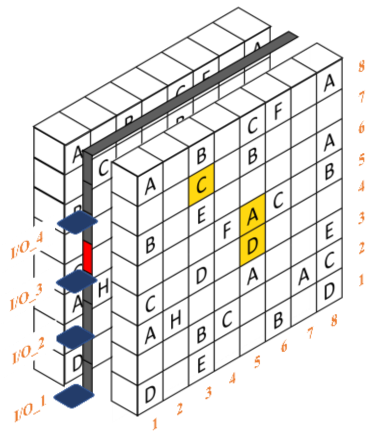
Figure 5. Flowchart of warehouse relocation algorithm

3.4 Decoding

After implementing the relocation algorithm, the outcome should be decoded into a comprehensible warehouse relocation plan for actual execution. An example with two-layer shelf is illustrated in Figure 6 (a), the current storage location, and Figure 6 (b), the target storage location. After comparing the current and target storage locations, Material C in the storage location (2,7), Material D in the storage location (5,5), and Material A in the storage location (7,4) in Figure 6 (a) were the materials to be relocated. The storage location (3,7), storage location (5,4), and storage location (5,5) were the target storage locations of Material C, Material D, and Material A, respectively. Apply the proposed algorithm to plan the storage assignment in Figure 6(a) to Figure (b). After decoding, the warehouse relocation plan is organized in Table 3. Table 3 illustrates that the stacker crane needs to travel from coordinates (1,1) through the loading station 1 to the storage location (5,5) and picked up Material D. It then moved Material D to the storage location (5,4) through the loading station 3 and to the storage location (7,4) through the loading station 3 to pick up Material A. Then it moved Material A through the loading station 3 to the storage location (5,5), and through the loading station 3 to the storage location (2,7) to pick up Material C. It then moved Material C to the storage location (3,7) through the loading station 3. This completed the overall warehouse relocation task. The moving sequence of stacker crane can be expressed as from initial storage location (1,1) -> I/O_1 -> (5,5)-> I/O_3 -> (5,4)-> I/O_3 -> (7,4)-> I/O_3 -> (5,5)-> I/O_3 -> (2,7)-> I/O_3 -> (3,7).



(a)



(b)

Figure 6. Initial storage location and target storage location of a two-layer shelf relocation example

Table 3. Decoding results: the relocation plan

No.	Material type	Start	End	I/O
1	-	(1,1)	(5,5)	1
2	D	(5,5)	(5,4)	3
3	-	(5,4)	(7,4)	3
4	A	(7,4)	(5,5)	3
5	-	(5,5)	(2,7)	3
6	C	(2,7)	(3,7)	3

4. COMPUTATIONAL EXPERIMENTS

This study used Python and Gurobi Optimization software to implement the algorithms running on the computer with a 2.80 GHz processor with 16GB RAM for computational experiments.

4.1 Design of experiments and results analysis

To examine the performance of Model A and the proposed heuristic, this study designed a three-factor full factorial experiment. The first factor is the rack size. It represents the number of storage locations. Seven levels of 9, 25, 49, 64, 100, 400, and 900 are included in this factor. The second factor is the storage location utilization, representing the proportion of storage locations occupied by materials, including three levels of 60%, 80%, and 97%. The third factor is the rack organization, representing the proportion of current storage locations mismatching the target storage locations i.e., the proportion of the number of storage locations to be rearranged, including two levels of 60% and 80%. Table 4 integrated the settings of these factors. 42 test problems were generated by this method. In these test problems, the AS/RS storage location was assumed to be 1.45 m long, 1.45 m wide, and 1.8 m high. The horizontal and vertical speeds of the stacker crane were 80 m/min and 15 m/min, respectively. Additionally, this study used Chebyshev distance to calculate the travel time and distance of the stacker crane. The number of material types was set at 3 for all test problems.

Table 4 Factors and levels of the test problems

Factor	Level values						
Rack size (Size)	3*3	5*5	7*7	8*8	10*10	20*20	30*30
Utilization (Util.)	60%	80%	97%				
Rack organization (Org.)	60%	80%					

The experimental results are organized in Table 5. Columns (1) to (6) in Table 5 represent the storage location state of each test problem. Column (5) represents the number of loading stations of AS/RS, which was generated randomly based on the principle of the higher the shelf, the larger number of the loading stations. Column (6) shows the total number of moves (T) to be assigned in advance when solving Model A. This number was determined according to the storage location utilization and the proportion of the storage locations to be rearranged. The total travel time of the stacker crane required for each test problem was organized in columns (7) to (11). Column (7) was the solution obtained by Model A. The solution marked with "*" represented that the test problem could be solved to optimal using Model A. The solution marked with "#" indicated the best feasible solution found before running out of memory. The solution without any mark represented the best feasible solution that could be found after running the algorithm for 24 hours. If no feasible solution could be found after running the algorithm for 24 hours, the program was stopped and use "-" to represent the solution. The runtime for obtaining the solutions shown in column (7) are shown in column (12). Columns (13) to (16) show the minimum, maximum, average, and standard deviation of the solution obtained from the 10 replications of executing the proposed warehouse relocation algorithm. Column (17) shows the average gap between the best solutions found from the exact approach (i.e., running Model A) and the heuristic solutions using the proposed algorithm of the test problems.

As shown in Table 5, the number of variables in Model A grew exponentially as the rack size (i.e., the number of storage locations), storage location utilization, and the proportion of the storage locations to be rearranged increased, which increased the runtime to solve the test problems. The largest problem that could be solved using Model A is with rack size of 7 * 7, i.e., 49 storage locations. Actually, in the study made by Pazour and Carlo (2015) on single loading/unloading station, the average runtime to solve the eight test problems with rack size of 3*3 is 4.96 hours using their general formulation running on a 3.4GHz processor with 4.0GB of RAM personal computer with IBM ILOG CPLEX Optimization Studio 12.5. The eight test problems of 3*3 storage locations tested by Pazour and Carlo (2015) are accessible from http://dx.doi.org/10.1016/j.tre.2014.11.002). This study also applied Model A to solve the same eight test problems. The average runtime of obtaining the optimum solution was only 0.07 hours.

Furthermore, the proposed heuristic, the warehouse relocation algorithm, can find the optimum solutions for four out of the six test problems on a smaller scale (rack size of 3 * 3). For the rest two, the gap was within 10%. The longest runtime for the heuristics is 0.3 sec. Also, Model A can find the optimum solutions for four out of the six test problems with the rack size of 5 * 5, but the runtime was extended to more than ten hours. The runtime for the heuristics to solve all these test problems was less than 1 second, while the gap is at most 41%. When the scale of the test problems increased to rack size of 8*8, the runtime of the heuristic was all within 3 seconds. Whereas Model A was not able to find even feasible solutions within 24 hours.

Table 5. Experimental results of three materials

Experimental results								*: Solved to optimality #: Best-found feasible solution before running out of memory								
(1)	(2)	(3)	(4)	(5)	(6)	Travel time (in secs)					runtime (in secs)					(17)
Inst.	Size	Util.	Org.	I/O	*T*	Optimal	Heuristic				Optimal	Heuristic				Gap
						(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)	
							Min	Mean	Max	STD		Min	Mean	Max	STD	
1	3*3	60%	60%	2	10	*57.6	57.6	70.3	88.3	10.7	9.2	0.1	0.1	0.2	0.0	0%
2	3*3	60%	80%	2	10	*70.0	76.2	85.1	92.3	4.9	13.6	0.1	0.2	0.2	0.0	9%
3	3*3	80%	60%	2	10	*50.4	50.4	57.6	68.9	9.2	3.6	0.1	0.1	0.2	0.0	0%

4	3*3	80%	80%	2	10	*76.8	82.2	87.5	111.2	9.8	20.9	0.2	0.2	0.2	0.0	7%
5	3*3	97%	60%	2	10	*65.0	65.0	90.3	112.5	21.0	2.7	0.1	0.2	0.2	0.0	0%
6	3*3	97%	80%	2	10	*83.7	83.7	118.0	161.6	32.0	6.9	0.2	0.2	0.3	0.0	0%
7	5*5	60%	60%	2	10	*192.7	213.6	227.4	250.3	14.3	5292.0	0.2	0.2	0.2	0.0	11%
8	5*5	60%	80%	2	12	*254.4	307.2	327.5	349.3	14.3	66168.0	0.2	0.2	0.3	0.0	21%
9	5*5	80%	60%	2	12	*222.6	276.2	361.2	419.0	56.8	35640.0	0.2	0.3	0.3	0.0	24%
10	5*5	80%	80%	2	15	325.0	457.5	509.2	553.9	35.1	86400.0	0.3	0.3	0.4	0.0	41%
11	5*5	97%	60%	2	15	*252.0	346.1	374.8	424.3	27.5	38261.5	0.4	0.4	0.5	0.0	37%
12	5*5	97%	80%	2	20	438.0	553.7	569.9	582.0	10.8	86400.0	0.6	0.6	0.7	0.0	26%
13	7*7	60%	60%	3	20	1380.0	509.4	530.1	551.6	15.1	86400.0	0.4	0.4	0.5	0.0	-63%
14	7*7	60%	80%	3	25	1484.4	684.7	724.9	757.9	28.1	86400.0	0.6	0.6	0.7	0.0	-54%
15	7*7	80%	60%	3	25	#1926.0	718.9	734.2	753.5	11.6	9721.0	0.5	0.5	0.6	0.0	-63%
16	7*7	80%	80%	3	30	#858.0	950.4	978.0	1026.7	21.7	73249.0	0.6	0.7	0.9	0.1	11%
17	7*7	97%	60%	3	30	#2019.6	874.5	913.5	951.9	21.3	83973.1	1.3	1.4	1.5	0.1	-57%
18	7*7	97%	80%	3	40	#3501.0	1134.0	1226.9	1343.8	77.1	70961.9	2.0	2.2	2.8	0.2	-68%
19	8*8	60%	60%	3	25	-	736.1	759.8	772.5	13.3	86400.0	0.5	0.5	0.5	0.0	-
20	8*8	60%	80%	3	35	-	1010.4	1052.4	1080.0	20.2	86400.0	0.7	0.7	0.8	0.0	-
21	8*8	80%	60%	3	35	-	1199.7	1294.3	1334.5	42.1	86400.0	0.8	0.8	0.9	0.0	-
22	8*8	80%	80%	3	40	-	1579.8	1702.6	1830.9	101.4	86400.0	1.0	1.1	1.2	0.1	-
23	8*8	97%	60%	3	40	-	1742.2	1766.5	1797.8	19.4	86400.0	1.4	1.6	1.7	0.1	-
24	8*8	97%	80%	3	50	-	1943.7	1998.5	2156.3	68.7	86400.0	2.0	2.2	2.4	0.2	-
25	10*10	60%	60%	4	40	-	776.2	812.5	851.4	24.8	86400.0	0.7	0.8	0.8	0.0	-
26	10*10	60%	80%	4	50	-	1007.4	1050.6	1100.2	33.7	86400.0	1.0	1.0	1.1	0.0	-
27	10*10	80%	60%	4	50	-	1057.8	1112.4	1134.7	19.1	86400.0	0.8	0.9	0.9	0.0	-
28	10*10	80%	80%	4	65	-	1617.6	1626.3	1639.0	8.5	86400.0	1.3	1.3	1.3	0.0	-
29	10*10	97%	60%	4	60	-	2233.2	2280.8	2617.4	114.4	86400.0	2.0	2.2	2.5	0.1	-
30	10*10	97%	80%	4	80	-	2705.6	2947.4	3219.2	142.9	86400.0	3.6	4.0	5.1	0.5	-
31	20*20	60%	60%	7	150	-	4521.6	4686.1	4754.4	75.0	86400.0	77.4	141.2	180.2	31.7	-
32	20*20	60%	80%	7	200	-	6146.5	6270.0	6330.7	73.9	86400.0	202.2	272.3	306.2	28.7	-
33	20*20	80%	60%	7	200	-	6178.8	6529.6	6799.0	210.6	86400.0	37.1	72.5	151.5	36.7	-
34	20*20	80%	80%	7	260	-	9440.0	9617.4	9943.2	162.4	86400.0	177.8	308.9	405.2	83.9	-
35	20*20	97%	60%	7	240	-	12535.6	13405.2	14636.4	843.6	86400.0	331.2	406.6	522.1	75.0	-
36	20*20	97%	80%	7	320	-	18265.2	18702.6	19438.8	433.8	86400.0	1208.9	1382.1	1806.7	209.6	-
37	30*30	60%	60%	10	325	-	11726.4	11871.6	12195.6	147.0	86400.0	1552.8	2760.9	4156.8	1117.6	-
38	30*30	60%	80%	10	435	-	17380.2	17518.2	17942.4	160.8	86400.0	3213.9	4470.6	6906.1	1225.4	-
39	30*30	80%	60%	10	435	-	17518.8	17896.2	18310.8	376.9	86400.0	1155.3	2758.7	4205.0	1249.1	-
40	30*30	80%	80%	10	580	-	27921.1	28134.2	28402.1	145.5	86400.0	4120.8	6776.3	8536.3	1599.4	-
41	30*30	97%	60%	10	525	-	35683.3	36817.6	38365.1	642.1	86400.0	8590.7	9939.3	11672.4	1123.9	-
42	30*30	97%	80%	10	700	-	51434.1	52972.0	55270.4	1561.1	86400.0	33855.0	39362.3	49640.4	5142.5	-

Table 6. Experimental results of five materials (rack size of 3*3 and 5*5)

						*: Solved to optimality										
(1)	(2)	(3)	(4)	(5)	(6)	Travel time (in secs)					runtime (in secs)					(17)
Inst.	Size	Util.	Org.	I/O	T	Optimal	Heuristic				Optimal	Heuristic				Gap
						(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)	
							Min	Mean	Max	STD		Min	Mean	Max	STD	
43	3*3	60%	60%	2	10	*60.2	60.2	69.0	88.8	9.6	6.8	0.2	0.2	0.2	0.0	0%
44	3*3	60%	80%	2	10	*77.8	77.8	94.6	111.0	10.7	29.3	0.2	0.2	0.2	0.0	0%
45	3*3	80%	60%	2	10	*58.4	58.4	65.3	97.2	12.2	10.0	0.2	0.2	0.2	0.0	0%
46	3*3	80%	80%	2	10	*63.2	63.2	66.3	74.3	3.3	18.3	0.2	0.2	0.2	0.0	0%
47	3*3	97%	60%	2	10	*88.1	88.1	102.4	121.8	14.7	13.2	0.2	0.2	0.3	0.0	0%
48	3*3	97%	80%	2	10	*100.9	100.9	116.9	145.6	14.4	36.3	0.2	0.3	0.3	0.0	0%
49	5*5	60%	60%	2	10	*238.8	262.6	289.9	319.5	20.8	48600.0	0.3	0.3	0.3	0.0	10%
50	5*5	60%	80%	2	12	*292.7	346.8	362.7	397.1	13.8	39240.0	0.3	0.3	0.3	0.0	19%
51	5*5	80%	60%	2	12	232.8	279.8	365.3	430.2	63.5	86400.0	0.3	0.3	0.4	0.0	20%
52	5*5	80%	80%	2	15	376.8	436.7	499.3	575.0	41.9	86400.0	0.3	0.4	0.4	0.0	16%
53	5*5	97%	60%	2	15	292.2	366.6	456.0	530.3	55.0	86400.0	0.4	0.5	0.6	0.1	25%
54	5*5	97%	80%	2	20	476.1	587.6	702.6	804.6	75.6	86400.0	0.6	0.7	0.8	0.1	23%

As shown in Table 5, the number of variables in Model A grew exponentially as the rack size (i.e., the number of storage locations), storage location utilization, and the proportion of the storage locations to be rearranged increased, which increased the runtime to solve the test problems.

Moreover, not only the rack size, storage location utilization, the proportion of the storage locations to be rearranged, and the number of loading stations affect the solution runtime obtained by using Model A, the total number of material types is also a factor. The runtime is increased with the increase of the number of material types. Table 6 shows the results of solving problems with five types of materials under the rack size of 3*3 and 5*5. The runtimes of these problems are obviously a lot longer than the ones when solving similar problems under three types of material. Table 7 shows the runtime difference between three and five kinds of materials. When the number of types of materials was increased from three to five, the runtime of Model A was doubled. In contrast, the computing time of the warehouse relocation algorithm was increased by at most 0.1 sec.

To sum up the discussion of the experimental results, Model A can deliver the optimum solution effectively when the problem scale is small. As the rack size, storage location utilization, proportion of the storage locations to be rearranged, the number of loading stations, and the total number of types of materials increase, the runtime to get an optimal solution increases exponentially. When the rack size is increased to 8*8, Model A could not even find a feasible solution within 24 hours of runtime.

Table 7 Comparison of the runtime under two different number of material types

Size	Types of Materials	Ave. runtime in secs.				
		Optimal	Heuristic			
			Min	Mean	Max	STD
3*3	3	9.5	0.1	0.2	0.2	0.0
3*3	5	19.0	0.2	0.2	0.2	0.0

4.2 Actual Case

In this section, the AS/RS being used by a famous computer hardware manufacturer in Taiwan (hereinafter referred to as Company G) was taken as an actual case to demonstrate how the proposed warehouse relocation algorithm be applied to solve practical instance. Figure 7 is the layout of an AS/RS used in one of the manufacturing sites of Company G. This AS/RS is a unit-load AS/RS with multiple loading/unloading stations working under a single command cycle. The entire system comprised 8 columns, 23 rows, and 16 layers. There are a total of 2,944 storage locations for storing materials. Four stacker cranes, each of which is in charge of the retrieval/storage operations of two racks, are available. The length and width of each storage location are both 1.45m, respectively, and the height is 1.8 m. The horizontal and vertical speeds of the stacker crane are 80 m/min and 15 m/min, respectively. This AS/RS was connected to a seven-story building in which different production stages were designed on various floors. The AS/RS serves to smooth the hardware manufacturing. There were eight loading/unloading stations on each floor.

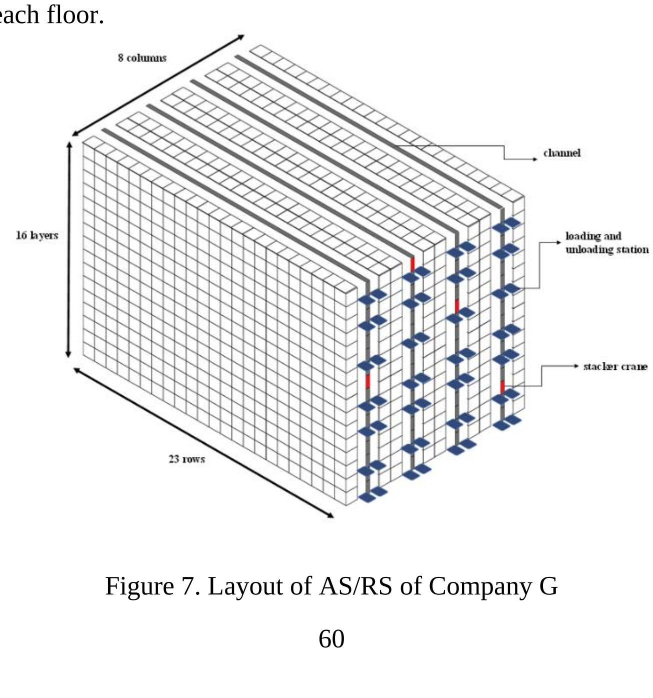


Figure 7. Layout of AS/RS of Company G

Lee (2020) analyzed the operational data of this AS/RS and divided the materials into seven types. An improved harmony search algorithm (IHSA) was proposed to find the storage location assignment to improve the system's running efficiency. In this section, the target storage location assignment was as recommended by Lee (2020). Furthermore, the storage status at 23:59 on May 18, 2020, was used as the current storage location. The storage location status was summarized in Table 8. Each stacker crane worked independently and took charge of 736 storage locations. The Area column in Table 8 represents the storage location area in the charge of the four stacker cranes. The Util. and Org. columns represent the storage location utilization and the proportion of storage locations to be rearranged in each area. The I/O column represents the number of loading stations in each storage area.

Table 8. Summary of the current storage location status of AS/RS of Company G

Area	Size	Util.	Org.	I/O
1	23*16*2	98%	83%	7
2	23*16*2	98%	82%	7
3	23*16*2	99%	79%	7
4	23*16*2	97%	81%	7

The proposed heuristic was applied to solve the warehouse relocation in each area. The result was decoded to provide a warehouse relocation plan of AS/RS for company G. Table 9 shows the result of warehouse relocation. The study suggested Company G to rearrange the storage locations according to the decoding result during off-peak hours to arrange the storage location to fit the storage location assignment recommended by Lee (2020). Since four stacker crane could operate simultaneously within each area, it is estimated that the relocation task could be completed within 10 hours.

Table 9. The results of warehouse relocation for the AS/RS

Experimental results								
Area	Travel time (in secs).				Runtime (in secs).			
	Heuristic				Heuristic			
	Min	Mean	Max	STD	Min	Mean	Max	STD
1	34928.5	37854.1	41343.3	1611.5	2418.6	4945.2	8367.8	1990.1
2	35382.0	36973.8	38443.2	969.6	3249.5	5785.6	9561.0	2165.6
3	35404.2	38074.2	39803.4	1413.6	3505.0	7194.0	10885.6	2344.9
4	31420.8	33920.4	36081.6	1638.0	2672.4	5690.1	8934.7	2430.5

5. CONCLUSION AND SUGGESTIONS

This study aimed at the warehouse relocation problem of unit-load AS/RS with multiple loading stations operating under single command cycle. Given the initial and target storage location assignments, we developed an optimization model utilizing integer programming techniques. Our proposed optimization model can efficiently solve problems involving up to 25 storage locations optimally. When compared to the average runtime of 4.96 hours reported by Pazour and Carlo (2015) for solving test problems with rack sizes of 3*3 to optimality, our study achieves the same optimal solutions in an average time of just 0.07 hours.

Given that the storage relocation problem has been demonstrated to be NP-hard, this study introduces a heuristic called the warehouse relocation algorithm. The purpose of this algorithm is to partition the original problem into multiple subproblems, thus accelerating the solution process and enabling its application to tackle industry-scale problems. This heuristic was applied to solve the warehouse relocation problem for an AS/RS that is currently running in a manufacturing facility in Taiwan. The AS/RS comprises a total of 2,944 storage locations, 28 loading/unloading stations and 4 stacker cranes. The proportion of storage locations to be rearranged was about 80%. This study successfully found a warehouse relocation plan for this practical problem and suggested a relocation plan that could be completed within 10 hours. This demonstrated the practical feasibility of the method developed by this study.

Based on what we could find in the literature, this paper is the first one to discuss this storage relocation problem for an AS/RS with multiple loading/unloading stations. Previous literature has focused primarily on storage relocation problems with single loading/unloading stations. None of them discussed the scenario with multiple loading/unloading stations. Both exact and heuristic approaches were proposed in this paper. As the AS/RS scale increases, the number of storage locations and the number of storage locations to be rearranged increase accordingly, which increases the difficulty of problem-solving. Partitioning the warehouse into several subareas and solving each subproblem accordingly has been demonstrated to be a practical approach. The proposed warehouse relocation algorithm not only can be applied to rearrange the storage locations within a warehouse. It is also possible to be applied to the situation to move goods closer to the adjacent loading/unloading station during off-peak hours to shorten the good retrieval process during peak hours.

There are still more details to be studied in the future to meet the need from practice. In this study, the storage location plan has been determined (i.e., known target locations). It might be an interesting problem to consider the relocation cost while determining the storage location assignment. Furthermore, planning the AS/RS operations in conjunction with the automated guided vehicles to automate the entire warehouse operations is also a practical research problem to study to meet the current trend.

References

Azab, A., and Morita, H. (2022). The Block Relocation Problem with Appointment Scheduling. *European Journal of Operational Research*, 297(2): 680-694.

Bacci, T., Mattia, S., and Ventura, P. (2019). The Bounded Beam Search Algorithm for The Block Relocation Problem. *Computers & Operations Research*, 103: 252-264.

Bacci, T., Mattia, S., and Ventura, P. (2020). A Branch-And-Cut Algorithm for The Restricted Block Relocation Problem. *European Journal of Operational Research*, 287(2): 452-459.

Carlo, H.J., and Giraldo, G.E. (2012). Toward Perpetually Organized Unit-Load Warehouses. *Computers & Industrial Engineering*, 63(4): 1003-1012.

Caserta, M., Schwarze, S., and Voss, S. (2012). A Mathematical Formulation and Complexity Considerations for The Blocks Relocation Problem. *European Journal of Operational Research*, 219(1): 96-104.

Chen, J.W., Zhang, X.L., Peng, X.F., Xu, D.S., and Peng, J.C. (2022). Efficient Routing for Multi-AGV Based on Optimized Ant-Agent. *Computers & Industrial Engineering*, 167: 108042.

Christofides, N., and Colloff, I. (1973). The Rearrangement of Items in A Warehouse. *Operations Research*, 21(2): 577-589.

Da Silva, M.D., Toulouse, S., and Calvo, R.W. (2018). A New Effective Unified Model for Solving The Pre-Marshalling and Block Relocation Problems. *European Journal of Operational Research*, 271(1): 40-56.

Dijkstra, A.S., and Roodbergen, K.J. (2017). Exact Route-Length Formulas and A Storage Location Assignment Heuristic for Picker-To-Parts Warehouses. *Transportation Research Part E: Logistics and Transportation Review*, 102: 38-59.

Frazelle, E.A., and Sharp, G.P. (1989). Correlated Assignment Strategy Can Improve Any Order-Picking Operation. *Industrial Engineering*, 21(4): 33-37.

Galle, V., Barnhart, C., and Jaillet, P. (2018). A New Binary Formulation of The Restricted Container Relocation Problem Based on A Binary Encoding of Configurations. *European Journal of Operational Research*, 267(2): 467-477.

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.

Hausman, W.H., Schwarz, L.B., and Graves, S.C. (1976). Optimal Storage Assignment in Automatic Warehousing Systems. *Management Science*, 22(6): 629-638.

He, Y.Y., Wang, A.H., Su, H.L., and Wang, M.Y. (2019). Particle Swarm Optimization Using Neighborhood-Based Mutation Operator and Intermediate Disturbance Strategy for Outbound Container Storage Location Assignment Problem. *Mathematical Problems in Engineering*, 2019: 9132315.

Jiao, Y.L., Xing, X.C., Zhang, P., Xu, L.C., and Liu, X.R. (2018). Multi-Objective Storage Location Allocation Optimization and Simulation Analysis of Automated Warehouse Based on Multi-Population Genetic Algorithm. *Concurrent Engineering: Research and Applications*, 26(4): 367-377.

Jin, B., Zhu, W.B., and Lim, A. (2015). Solving The Container Relocation Problem by An Improved Greedy Look-Ahead Heuristic. *European Journal of Operational Research*, 240(3): 837-847.

Jovanovic, R., Tuba, M., and Voss, S. (2019). An Efficient Ant Colony Optimization Algorithm for The Blocks Relocation Problem. *European Journal of Operational Research*, 274(1): 78-90.

Jovanovic, R., and Voss, S. (2014). A Chain Heuristic for The Blocks Relocation Problem. *Computers & Industrial Engineering*, 75: 79-86.

Ku, D., and Arthanari, T.S. (2016a). On The Abstraction Method for The Container Relocation Problem. *Computers & Operations Research*, 68: 110-122.

Ku, D., and Arthanari, T.S. (2016b). Container Relocation Problem with Time Windows for Container Departure. *European Journal of Operational Research*, 252(3): 1031-1039.

Lee, S.T. (2020). Optimization of Storage Location Assignment of Unit-Load Automated Storage and Retrieval System with Multiple Input/Output Depots Using Improved Harmony Search Algorithm. National Yang Ming Chiao Tung University, Taiwan.

Li, N.L., Zhang, C., Xie, W.X., and Li, Y.P. (2018). Exceptional Events Classification in Warehousing Based on An Integrated Clustering Method for A Dataset with Mixed-Valued Attributes. *International Journal of Computer Integrated Manufacturing*, 31(11): 1078-1096.

Liu, S., Liu, S.X., Lu, C., Zhou, M.C., and Abusorrah, A. (2022). Valid Inequality and Variable Fixation for Unrestricted Block Relocation Problems. *IEEE Transactions on Intelligent Transportation Systems*, Early Access. DOI10.1109/TITS.2022.3151069

Muppani, V.R., and Adil, G.K. (2008). A Branch and Bound Algorithm for Class Based Storage Location Assignment. *European Journal of Operational Research*, 189(2): 492-507.

Pang, K.W., and Chan, H.L. (2017). Data Mining-Based Algorithm for Storage Location Assignment in A Randomised Warehouse. *International Journal of Production Research*, 55(14): 4035-4052.

Pazour, J.A., and Carlo, H.J. (2015). Warehouse Reshuffling: Insights and Optimization. *Transportation Research Part E: Logistics and Transportation Review*, 73: 207-226.

Quispe, K.E.Y., Lintzmayer, C.N., and Xavier, E.C. (2018). An Exact Algorithm for The Blocks Relocation Problem with New Lower Bounds. *Computers & Operations Research*, 99: 206-217.

Scholl, J., Boywitz, D., and Boysen, N. (2018). On The Quality of Simple Measures Predicting Block Relocations in Container Yards. *International Journal of Production Research*, 56(1-2): 60-71.

Sharma, S., Abouee-Mehrizi, H., and Sartor, G. (2020). Inventory Management Under Storage and Order Restrictions. *Production and Operations Management*, 29(1): 101-117.

Tanaka, S., and Mizuno, F. (2018). An Exact Algorithm for The Unrestricted Block Relocation Problem. *Computers & Operations Research*, 95: 12-31.

Tanaka, S., and Takii, K. (2016). A Faster Branch-And-Bound Algorithm for The Block Relocation Problem. *IEEE Transactions on Automation Science and Engineering*, 13(1): 181-190.

Ting, C.J., and Wu, K.C. (2017). Optimizing Container Relocation Operations At Container Yards with Beam Search. *Transportation Research Part E: Logistics and Transportation Review*, 103: 17-31.

Wang, M., Zhang, R.Q., and Fan, K. (2020). Improving Order-Picking Operation Through Efficient Storage Location Assignment: A New Approach. *Computers & Industrial Engineering*, 139: 106186.

Wang, W., Yang, J., Huang, L., Proverbs, D., and Wei, J.B. (2019). Intelligent Storage Location Allocation with Multiple Objectives for Flood Control Materials. *Water*, 11(8): 1537.

Yang, Z.G., Wang, R.Y., Wu, D.P., Wang, H.G., Song, H.N., and Ma, X.Q. (2022). Local Trajectory Privacy Protection in 5G Enabled Industrial Intelligent Logistics. *IEEE Transactions on Industrial Informatics*, 18(4): 2868-2876.

Zehendner, E., Feillet, D., and Jaillet, P. (2017). An Algorithm with Performance Guarantee for The Online Container Relocation Problem. *European Journal of Operational Research*, 259(1): 48-62.

Zhou, Y.L., and Huang, N.N. (2022). Airport AGV Path Optimization Model Based on Ant Colony Algorithm to Optimize Dijkstra Algorithm in Urban Systems. *Sustainable Computing: Informatics and Systems*, 35: 100716.

Zhu, W.B., Qin, H., Lim, A., and Zhang, H.D. (2012). Iterative Deepening A* Algorithms for The Container Relocation Problem. *IEEE Transactions on Automation Science and Engineering*, 9(4): 710-722.

Zweers, B.G., Bhulai, S., and Van Der Mei, R.D. (2020). Pre-Processing A Container Yard Under Limited Available Time. *Computers & Operations Research*, 123: 105045.