processes

Article

Research on Inbound Jobs' Scheduling in Four-Way-Shuttle-Based Storage System

Zhaoyun Wu [1,*], Yingxu Zhang [1], Li Li [1,2], Zhongwei Zhang [1,*], Binbin Zhao [1], Yehao Zhang [1] and Xuewu He [1]

[1] School of Mechanical & Electrical Engineering, Henan University of Technology, Zhengzhou 450001, China
[2] School of Automobile and Transportation, Henan Polytechnic University, Zhengzhou 450046, China
* Correspondence: wuzhaoyun@haut.edu.cn (Z.W.); zzw_man@haut.edu.cn (Z.Z.)

Abstract: The four-way-shuttle-based storage and retrieval system is a recent innovative intelligent vertical warehousing system that has been widely applied in manufacturing and e-commerce environments due to its high flexibility and density. As a complex multi-device cooperative operational system, this system features the parallel operation of multiple elevators and four-way shuttles. During large-scale-batch inbound operations, the quality of scheduling solutions for inbound-operation equipment significantly impacts the system's efficiency and performance. In this paper, a detailed analysis of the inbound-operation process in the system is conducted, taking into consideration the motion characteristics of both the elevators and four-way shuttles. Furthermore, we establish operational time constraints that account for equipment acceleration and deceleration characteristics and introduce a flexible flow-shop-scheduling model to address the scheduling problem in the system. Additionally, we propose an improved genetic algorithm based on double-layer encoding to solve this problem. Comparative experiments with a traditional genetic algorithm and ant-colony algorithm demonstrate the superior efficiency and accuracy of our approach. Finally, the effectiveness of the proposed algorithm is validated through comparisons with large-scale practical experiments.

Keywords: four-way shuttle; elevator; job scheduling; improved genetic algorithm; FFSP problem

Citation: Wu, Z.; Zhang, Y.; Li, L.; Zhang, Z.; Zhao, B.; Zhang, Y.; He, X. Research on Inbound Jobs' Scheduling in Four-Way-Shuttle-Based Storage System. *Processes* **2024**, *12*, 223. https://doi.org/10.3390/pr12010223

Academic Editor: Olympia Roeva

Received: 4 December 2023
Revised: 31 December 2023
Accepted: 9 January 2024
Published: 19 January 2024

1. Introduction

In the field of the warehouse and logistics, "technology, intelligent transformation and flexible leap" have recently emerged as the primary development trend. The old extensive-management model has not been able to fulfill the demands of the precise management of warehouse cargo due to the fast development of company size, the increasing kinds of materials, and the complicated business process. As a result, the transformation of the cargo-storage system into a three-dimensional, automated, and intelligent method is unavoidable. A cutting-edge automated warehousing system called the four-way-shuttle-based storage and retrieval system (FWSBS/RS) is currently available on the market. Vertical elevators, horizontal four-way shuttles, three-dimensional shelves, together with additional hardware, and control management software, make up the system [1]. The benefits of this storage system are its high storage density, adaptable operating pattern, and quick reaction time. Since it has successfully solved the issues caused by traditional automated storage equipment's large area occupation, high energy consumption, low efficiency, and weak robustness [2], manufacturing companies and internet e-commerce companies have started to prioritize FWSBS/RS.

Inbound and outbound are the two primary operational modes in FWSBS/RS. During inbound operations, the system's elevator runs vertically to move the chosen cargo to the desired layer and the cargo can be delivered to the desired destination using a four-way shuttle, which can travel in four directions: forward, backward, left, and right. The system optimizes the storage position of cargo based on the actual situation during inbound operations, following a fixed procedure. In contrast, during outbound operations, the process is reversed, with a more randomized sequence and batch selection, prioritizing tasks differently. This paper primarily addresses inbound jobs' scheduling, specifically focusing on the characteristics of randomly arriving cargo. For the parallel operation of both elevators and shuttles, lack of an efficient scheduling mechanism would result in poor equipment utilization, an increasingly high energy consumption, and a long operating time, which runs counter to the idea of intelligent and low-carbon development [3].

At present, research in the field of job scheduling focuses on an automated storage and retrieval system (AS/RS) and a shuttle-based storage and retrieval system (SBS/RS). Due to these vertical storage systems' typical reliance on stacker cranes in each aisle for handling cargo or for transferring cargo from stacker cranes to reciprocating shuttles in corresponding aisles, their inbound and outbound operations are executed sequentially using respective devices, lacking the feature of multi-device parallel operations. However, there is relatively limited research on inbound-job scheduling for FWSBS/RS where multiple elevators and four-way shuttles serve the entire system in parallel. Owing to the unique characteristics of FWSBS/RS, existing methods are not directly applicable.

Therefore, this study draws inspiration from mature approaches in flow-shop scheduling and delves into their operational strategies and system performance. We apply an improved genetic algorithm to provide a more versatile and viable inbound-job-scheduling solutions for FWSBS/RS. This approach enhances warehousing efficiency, reduces operational costs, and opens new avenues for research in the scheduling of intelligent vertical storage systems.

The main contributions of this paper include two aspects. Firstly, a thorough analysis is conducted on the inbound-job-flow process in FWSBS/RS, and the motion patterns of both the elevator and four-way shuttles are presented, taking into account the equipment's acceleration and deceleration characteristics. Then, a precise inbound-job-scheduling model is created using a hybrid-flow-shop-scheduling approach to describe the system's properties. Secondly, this paper designs an improved genetic algorithm based on double-layer coding, and introduces the simulated annealing algorithm into the genetic algorithm to solve the inbound jobs' scheduling problem for the FWSBS/RS.

The remaining sections of this paper are organized as follows. In Section 2, we review the relevant literature. Section 3 describes the system layout and job-flow process. In Section 4, we transform the inbound-job-scheduling problem into a hybrid-flow-shop-scheduling problem and develop a corresponding model. Section 5 presents DELO-GA. Section 6 provides simulation experiments. Finally, Section 7 concludes the paper and discusses future research directions.

2. Literature Review

AS/RS, SBS/RS, and AVS/RS are the principal intelligent storage systems; AS/RS is the storage and pickup task of the system with the tunnel stacker working in parallel. AVS/RS and SBS/RS are shuttle vehicles driving along the track, and cooperate with the elevator to complete cross-layer and cross-roadway storage and pickup tasks. Currently, the research of FWSBS/RS is limited. Its operational flow is comparable to that of the AVS/RS and SBS/RS systems because the basis for in-depth study of the two systems can be inferred using FWSBS/RS.

Many researchers are currently studying the scheduling of AVS/RS and SBS/RS. The primary concerns are the optimization of storage space [4,5]; the order-sequence-optimization problem [6,7]; and path planning [8,9].

Using conceptualizing tools, Malmborg and his team [10–12] established a methodology for estimating system performance. The functional link between predicted access time and key system parameters was provided through modeling important properties of a three-dimensional storage system, such as storage capacity, shelf configuration, number of automated cars, number of elevators, etc. The storage–retrieval-cycle time, system utilization, and throughput are all estimated using the state equation model, which offers theoretical support for the technical selection and performance-feasibility decision of AS/RS and AVS/RS systems. In order to examine how the quantity of elevators and shuttles affected the effectiveness of the AVS/RS system, Marchet et al. [13] developed a simulation model and established a formula for evaluating throughput. Kuo et al. [14] developed a computationally efficient periodic time model for estimating resource utilization in AVS/RS. They analyzed 12 different scenarios to demonstrate the model's performance. Despite the model displaying some significant errors, it can provide accuracy in estimating vehicle usage and systems' costs.

It has recently been popular to use the queuing-network model to determine the operation time and waiting time more correctly in the three-dimensional storage system. The queueing-network model (QN), initially suggested by Malmborg's team [15], was used to model vehicle flow in various situations and assess the performance of various system characteristics. Their method offers the groundwork for further study and applications while offering an efficient way to conceive tools for AVS/RS systems. The sorting effectiveness of the system under a single instruction cycle and a double instruction cycle was also taken into account by Fukunari's team [16,17]. They also provided an iterative calculation approach for the queuing model under a random storage strategy and calculated the operation time of the instructions. In addition, a closed-queuing-network (CQN) model was put forth to fix the state-equation model's computational flaws in determining the system's equipment-utilization rate. By using dynamic data from shuttle and elevator operations as input for the discrete service-time distribution of the queuing network and taking into account various system factors and variables, Martin et al. [18] presented a discrete-time open-queueing-network approach. The system's performance metrics, such as the arrival rate, service rate, queue length, etc., were analyzed. In their initial research, Heragu et al. [19] researched different automated-warehouse-storage and -retrieval systems (AS/RS) and autonomous vehicle storage and retrieval systems (AVS/RS), and through viewing them as open queuing networks (OQN) and analyzing job cycle times using tools from existing manufacturing-system-performance analyzers (MPA) AVS/RS and AS/RS's performance is compared and examined.

Open-loop queuing networks (OQNs) and closed-loop queuing networks (CQNs) are now the two most used modeling techniques for AVS/RS systems. The actual case cannot be adequately described using CQN or OQN models because they overlook the device waiting time and task waiting time, respectively. A semi-open-loop queuing network (SOQN) is, therefore, more appropriate for the operating scenario of a three-dimensional storage system when taking into account the mutual waiting process during the actual operation of the system. The semi-open queuing-network (SOQN) model was created for the first time by Roy et al. [20,21], and the analysis-task completion time was solved using the decomposition-based method (DM). However, the impact of the lifting mechanism was disregarded. The effect of the stationing strategy on system performance is investigated and just one shelf operation range is taken into account. In order to account for the effect of lifting mechanisms in the three-dimensional warehouse system, Ekren et al. [22,23] constructed a semi-open queue-network (SOQN) model. They then used approximation methods to examine the system's task-completion time, waiting time, and other indicators. The topic of maximizing the number of warehouse cars under specified system parameters is examined using the matrix-geometric method (MGM). To assess the performance of the multi-layer AVS/RS system, Cai et al. [24] created a semi-open queue-network model, represented the vehicle travel time of AVS/RS as the service time of the service node in SOQN, and utilized the matrix-geometry approach for analysis.

The study mentioned above is mostly based on the queuing-network model, which primarily assesses system performance and efficiency and takes task arrival rate, service rate, and task dependence into account. However, it becomes challenging to represent and solve the scheduling problem when the queuing-network model is used to analyze a vast and intricate three-dimensional storage system. Currently, most researchers typically use intelligent optimization algorithms, such as the ant-colony algorithm [25], genetic algorithm [26], particle swarm optimization algorithm [27], etc., to solve the scheduling issues of large-scale three-dimensional storage systems. These algorithms can handle more complex scheduling issues and typically have the qualities of global search and adaptability. To address the order of single-access cargo operation, Cao et al. [28] used a technique similar to TSP, developed the appropriate mathematical model, and showed that this technique may shorten the hoist's operating time. To solve the scheduling issue of four-way shuttles in three-dimensional storage, Song et al. [29] proposed a scheduling optimization model based on the genetic algorithm coupled with cross-mutation operators. They demonstrated through simulation experiments that the optimized route scheduling could increase the operating efficiency of the system. In order to solve the scheduling problem for the four-way shuttle, Li et al. [30] took into account the randomness of storage and retrieval in the three-dimensional storage system, established a mathematical model for the equipment in FWSBS/RS, proposed a gray wolf optimization algorithm, and demonstrated the superiority of the proposed algorithm by comparing it to the outcomes of the genetic algorithm and particle swarm optimization algorithm. In order to overcome the problem of task scheduling, Wang et al. [31] developed a sequential mathematical model of task operation based on the features of parallel shuttle picking and serial elevator operation in three-dimensional storage systems.

In conclusion, FWSBS/RS offers significant flexibility, and it is challenging to apply a scheduling model to other kinds of storage systems. Few relevant studies have been conducted on the cooperative scheduling of four-way shuttles and elevators. The majority of studies focus on the scheduling analyses of one kind of equipment. The pertinent research for the three-dimensional storage system's equipment makes the assumption that it moves at a constant speed and does not take into account the effects of acceleration and direction change during the four-way-shuttle scheduling process, which makes it challenging to adapt the research to the three-dimensional storage system's actual operating conditions.

For the multi-elevators' and four-way shuttles' parallel operation in the system, the allocation and optimization of jobs between different parallel devices for inbound operations of cargo constitute a typical NP-Hard problem [32]. The exponential computational complexity arises from determining optimal job sequences and machine selection for the interdependent transporting and storing subtasks between elevators and four-way shuttles. To address this, this paper draws from well-established research on flow-shop-scheduling problems [33]. The flexible flow-shop-scheduling problem (FFSP), which is challenging to solve using exact algorithms, is an NP-Hard issue, as Wang et al. [34] have demonstrated. Greedy algorithms are typically employed to address issues with little computational complexity. Intelligent optimization techniques like the genetic algorithm and particle swarm optimization algorithm are typically employed to handle large-scale and difficult issues. This work introduces the FFSP issue to answer the inbound scheduling problem of FWSBS/RS because the FFSP problem contains the dual features of flow-shop scheduling and parallel machines.

3. Layout and Presumptions of FWSBS/RS

3.1. Layout Description

Vertical elevators, four-way shuttles, three-dimensional shelves, cargo pickers, as well as other machinery, and control-management software, make up FWSBS/RS. More than one elevator and one four-way shuttle are coordinated when the system handles inbound operations. The elevator is in charge of moving cargo from the bottom of the inbound picking station to the buffer zone of the target layer. The four-way shuttle is in charge of moving the cargo from the buffer zone to the target storage location on that layer. It travels down the horizontal main aisles and longitudinal sub-aisles. We investigate the scheduling of a single inbound operation in FWSBS/RS, since a three-dimensional storage system is characterized by small numbers, numerous categories, and great flexibility.

Figure 1 depicts the layout of FWSBS/RS. In the system, the layout of each layer is the same. The main aisles are those that run above and below the storage area, and the sub-aisles are those that run through the storage space. It is presumed that the elevator runs in the Z direction; the main aisle and the longitudinal sub-aisle are in the X and Y direction, respectively. When there is cargo to be stored in the system, the shuttle with the cargo can run along the main aisle to the sub-aisle and then to the target position When the four-way shuttle has finished the inbound operation, it returns to the buffer zone to prepare the next operation.

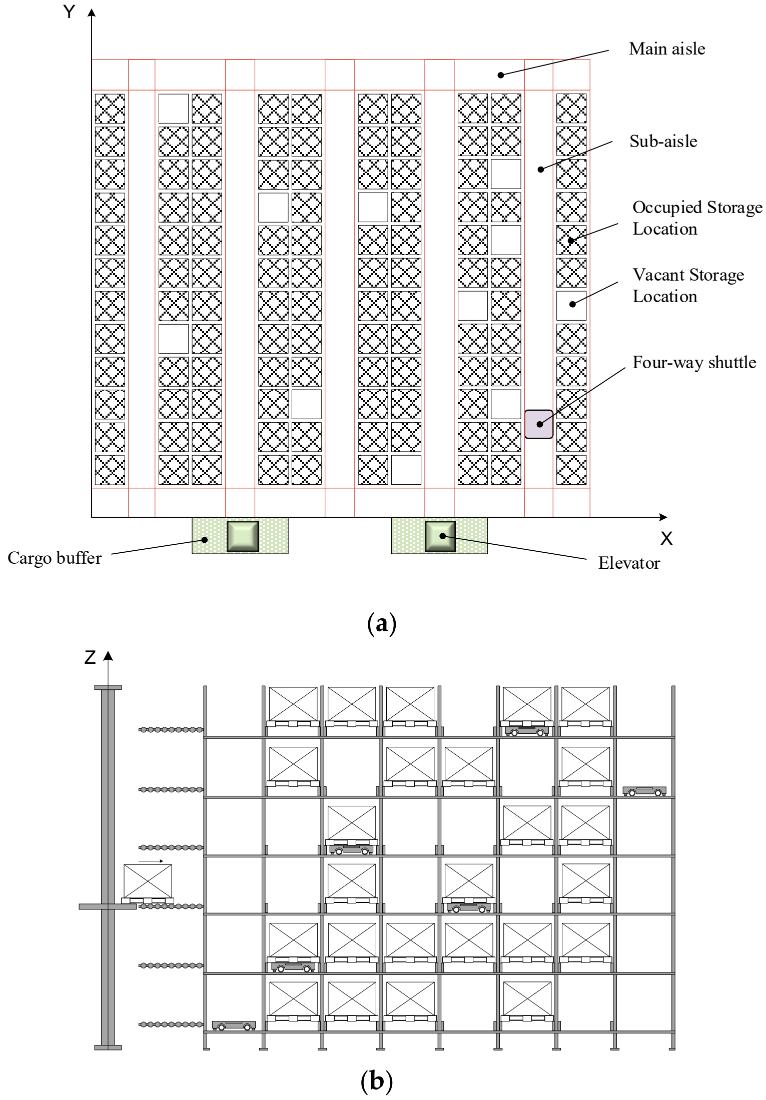

Figure 1. Layout of the FWSBS/RS: (**a**) floor plan; (**b**) elevation plan.

3.2. Inbound-Operation Flow

The inbound operation flow in FWSBS/RS is listed as follows:

1. The system requests scheduling of the elevator when the inbound operation is initiated. If there is an idle elevator, let it pick up the cargo at the I/O station;
2. If there is not, the task must wait. The elevator transports the cargo in a vertical direction to the cargo buffer on the target layer, releases the cargo, and then returns to the I/O station to wait for the next operation;
3. The system checks to see if there is an idle shuttle at the present layer, and, if there is, the four-way shuttle moves to the buffer zone to pick up cargo, but, if there is not, the task must wait;
4. After picking up cargo, the four-way shuttle runs along the main aisle to the sub-aisle where the target storage position is located; when the shuttle arrives at the destination,

it releases the cargo and then returns to the buffer along the shortest path to wait for the next task.

The inbound-process flow chart of FWSBS/RS is illustrated in Figure 2.

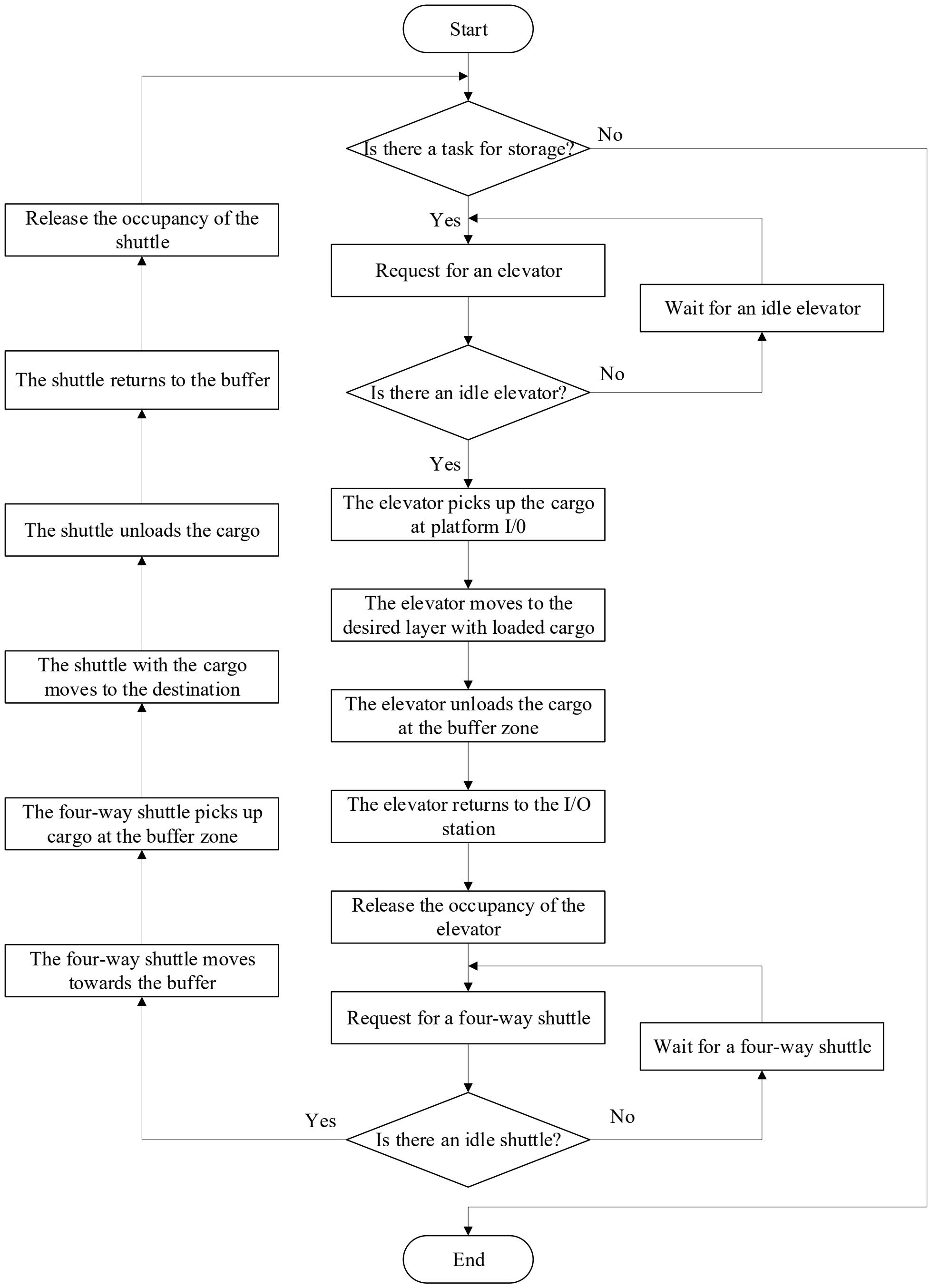

Figure 2. Flow chart of inbound operation.

3.3. Presumptions and Parameters

To make the system closer to the real operating conditions and to reduce idle-equipment time while satisfying constraints on the sequence of inbound jobs, the following presumptions are made which facilitate the modeling and research of FWSBS/RS:

1. The four-way shuttle with cargo can only move along the main aisles and sub-aisles;
2. The four-way shuttle without cargo can also run below cargo compartments besides the main aisle and sub-aisle;

3. Since the single-pick time for all cargo on the picking machine varies only slightly, it can be assumed as a constant value;
4. The storage system's space occupancy state is determined before executing the inbound tasks, and all target spaces are unoccupied;
5. During operation, the four-way shuttle and elevator can only move a single load at a time, and they are not allowed to interrupt before finishing the current task;
6. The four-way shuttle follows the POSC (Point of Service Completion) dwell strategy [14]; that is, they will stay at the position where the last task ended and wait for the next task;
7. Requests for the idle shuttle and elevator obey the FCFS (First Come First Service) rules [35];
8. The shelves and compartments in the whole storage system are identical, as is the width of each aisle.

In order to facilitate subsequent modeling and computations, the parameters and their meanings for the FWSBS/RS model are defined as presented in Table 1.

Table 1. The parameters of FWSBS/RS.

Parameters	Meaning	Parameters	Meaning
H	Shelf-layer height	C_i	Sub-aisle number
L	Cargo-location length	C_{xyz}	Aisle of destination for inbound cargo
W_C	Cargo-location width	S_0	Running-distance limit
W_m	Main-aisle width	t_s	Time for cargo picking
W_s	Sub-aisle width	$t_C^{C_1}$	Time for a four-way shuttle in main-aisle
T	Number of sub-aisles	$t_y^{y_1}$	Time for a four-way shuttle in sub-aisle
M	Number of layers	$t_z^{z_1}$	Time for an elevator in the vertical direction
N	Number of columns per layer	T_{Sn1}	Picker single-job time
P	Number of storage positions per column	T_{En2}	Elevator single-job time
a_R	Four-way-shuttle acceleration/deceleration	t_E	Elevator loading/unloading time
$v_{R\max}$	Maximum speed of four-way shuttle	T_{Rn3}	Four-way-shuttle single-job time
a_E	Elevator acceleration/deceleration	t_R	Four-way-shuttle loading/unloading time
$v_{E\max}$	Maximum speed of elevator	T_k	Individual-cargo inbound-job-completion time
H^k	Vacant-cargo locations	T_{sum}	Batch inbound-jobs-completion time

4. Scheduling Modeling of Inbound Jobs for FWSBS/RS

4.1. Description of the Problem

The scheduling issue for flexible flow shops in production systems is characterized by parallel machines and flow-shop scheduling, and is best explained as follows: there are $m_i(m_i \geq 2)$ machines at stage i; n workpieces are continually processed on the $c(c \geq 2)$ stages; and there is at least one stage with many machines.

A batch of inbound jobs for FWSBS/RS involves picking with the picking machine, and transportation with the elevator and the four-way shuttle. The inbound workflow for a batch of goods can be divided into three phases: goods picking, goods lifting, and goods transportation to the target location. The different incoming goods can be viewed as workpieces awaiting processing, and their handling by different equipment can be seen as different processing steps for these workpieces. The picking equipment, elevator, and four-way shuttle can be considered as parallel production equipment. Therefore, the scheduling problem for FWSBS/RS can be viewed as a flexible flow-shop-scheduling problem, and the scheduling goal is to minimize the total-job-completion time for this batch of inbound jobs and determine the order of inbound jobs.

4.2. Establishment of FWSBS/RS Model

Assuming that there are M layers in FWSBS/RS, each layer of shelves has N columns of shelves and T sub-aisles, and there are P compartments in each column. The coordinates of the cargo space located in row y, column x of layer z, can be written as

$\{(x, y, z)|x \in (1, 2, \ldots, N), y \in (1, 2, \ldots, P), z \in (1, 2, \ldots, M)\}$. The set of empty cargo spaces in the system is $H^k = \left\{\left(x_i^k, y_i^k, z_i^k\right)|k = 1, 2, \ldots, n\right\}$. The set of the sub-aisle in the system is noted as $C_i = \{(C_1, C_2, C_3, \ldots, C_{N/2})|i = 1, 2, \ldots, n\}$; the aisle C_1 is defined as the space between the first and second columns of shelves, while aisle $C_{N/2}$ is defined as the space between column N and column $(N-1)$ of shelves. The coordinates of the storge destination are (x, y, z); the sub-aisle in which the destination bay is located is C_{xyz}, $C_{xyz} = \left\lceil \frac{x}{2} \right\rceil$.

4.3. Equipment-Operating-Time Analysis

The main equipment in FWSBS/RS are four-way shuttles and elevators, which accelerate and decelerate during the stationary-to-stop operation. In practical operational scenarios, their operational processes are regarded as a combination of uniformly accelerated, decelerated motion, and constant linear motion [36]. To minimize collisions while maximizing equipment efficiency, it is essential to impose reasonable acceleration and deceleration constraints, considering adequate acceleration and deceleration distances within the limited travel range of the equipment. The motion process can be divided into two patterns: (a) stationary–uniform, acceleration–uniform, deceleration–stationary; (b) stationary–uniform, acceleration–uniform, motion with maximum speed–uniform, deceleration–stationary. Figure 3 depicts the two patterns.

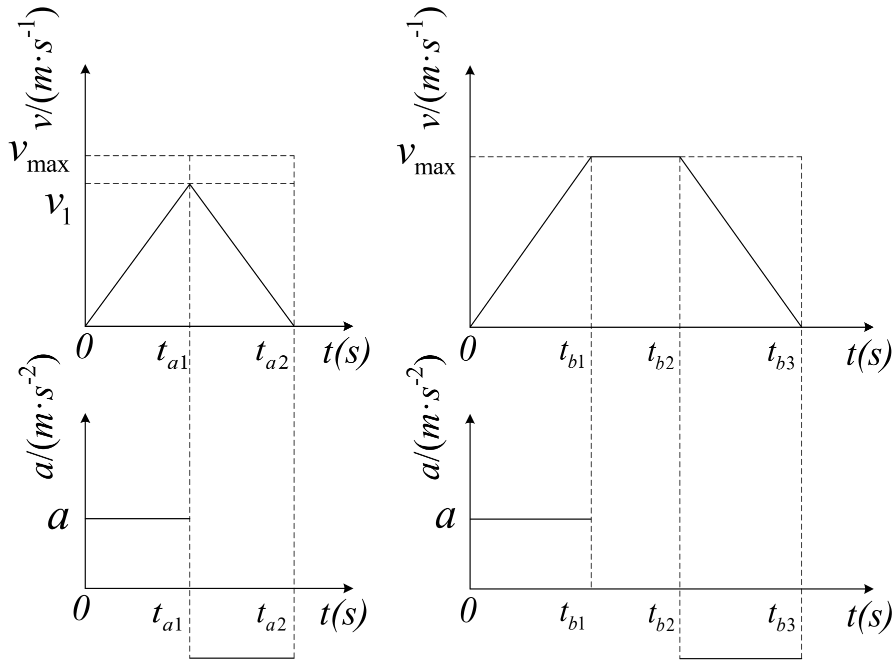

Figure 3. Motion–time relationship of the equipment: (**a**) equipment not running at maximum speed; (**b**) equipment at maximum speed.

According to the principle of kinematics, when the distance is greater than the critical distance, the motion state of the equipment changes from pattern (a) to pattern (b); the limit distance is shown in Formula (1).

$$S_0 = \frac{v_{\max}^2}{a} \tag{1}$$

The operating time of the equipment is divided into the running time of the four-way shuttle in the horizontal direction and the running time of the elevator in the vertical direction. The running time of the four-way shuttle is divided into the time $t_c^{C_1}$ running from the buffer to the sub-aisle where the cargo position is located along the x direction, and the time $t_y^{y_1}$ running from the main aisle entrance to the target cargo position along the Y direction, and the running time of the elevator is the time $t_z^{z_1}$ running from the bottom I/O station to the target-cargo-position layer. The time for each stage is shown in Formulas (2)–(4).

$$t_C^{C_1} = \begin{cases} 2\sqrt{\frac{|C-C_1|(2W_c+W_s)}{a_R}} & , |C-C_1|(2W_c+W_s) \le S_0 \\ \frac{|C-C_1|(2W_c+W_s)}{v_{R\max}} + \frac{v_{R\max}}{a_R} & , |C-C_1|(2W_c+W_s) > S_0 \end{cases} \tag{2}$$

$$t_y^{y_1} = \begin{cases} 2\sqrt{\frac{|y-y_1|L+W_m}{a_R}} & , (|y-y_1|L+W_m) \le S_0 \\ \frac{|y-y_1|L+W_m}{v_{R\max}} + \frac{v_{R\max}}{a_R} & , (|y-y_1|L+W_m) > S_0 \end{cases} \tag{3}$$

$$t_z^{z_1} = \begin{cases} 2\sqrt{\frac{|z-z_1|H}{a_E}} & , |z-z_1|H \le S_0 \\ \frac{|z-z_1|H}{v_{E\max}} + \frac{v_{E\max}}{a_E} & , |z-z_1|H > S_0 \end{cases} \tag{4}$$

4.4. Inbound Jobs' Scheduling Model

The problem of inbound scheduling for FWSBS/RS can be described as that, to accomplish inbound scheduling, n cargos to be inbound are assigned to m_j four-way shuttles and m_i elevators. The set of cargo to be stored is expressed as $H = \{H_1, H_2, H_3, \ldots, H_n\}$; the elevator set is represented as $E = \{E_1, E_2, E_3, \ldots, E_n\}$; the set of four-way shuttles is marked as $R = \{R_1, R_2, R_3, \ldots, R_n\}$; and the inbound time to complete a cargo can be expressed as follows:

$$T_{Sn1} = t_s \tag{5}$$

$$T_{En2} = 2t_z^{z_1} + t_E \tag{6}$$

$$T_{Rn3} = t_C^{C_1} + t_y^{y_1} + t_R \tag{7}$$

$$T_k = T_{Sn1} + T_{En2} + T_{Rn3} \tag{8}$$

In the above formulae, Formula (5) represents the first process, which is the picking time of the picking equipment, determined by the quantity of goods to be picked. Formula (6) stands for the second process, comprising the operation time of the elevator and the loading/unloading time of the elevator. Formula (7) denotes the operating time for the third process, involving the operation time of the four-way shuttle, including the time it takes for the four-way shuttle to travel to the main aisle where the goods are located, the time it takes to traverse the sub-aisle to the target location, and the time taken for loading/unloading of cargo at the location. Formula (8) represents the total time required for each job to complete the warehousing process.

4.4.1. Objective Function

The optimization goal is to minimize the maximum operational time while scheduling a batch of cargo for storage; the objective function can be written as Formula (9).

$$T_{sum} = \min(\max(T_k)), 1 \le k \le n \tag{9}$$

4.4.2. Constraint Conditions

The following constraint conditions are put in place in FWSBS/RS to guarantee the continuation of scheduling a batch of arriving cargo, the continuity of equipment operation, and the cooperative operation of multiple pieces of equipment.

$$t_n < t_{n+1} \tag{10}$$

$$P = \begin{cases} T(1,n) < T(2,n) < T(3,n), \ P(n) = 1,2,3 \\ G_n \in H_n^k \end{cases} \tag{11}$$

$$E(i) = \begin{cases} \text{argmin}(|D(i) - T|) & , \forall E(i) = 0 \\ 1 & , \exists E(i) = 0 \\ 0 & , \forall E(i) = 1 \end{cases} \tag{12}$$

$$t_{E_n} \geq T_{E(n-1)},\ t_{R_n} \geq T_{R(n-1)} \tag{13}$$

Formula (10) shows that the inbound time for each load must be increased in the order of inbound; Formula (11) shows the uniqueness constraint on the equipment's time as well as the target cargo position; Formula (12) restricts the choice of elevator; Formula (13) limits the operating times of a single elevator and the four-way shuttles for each layer.

5. Design of a Solution Approach for FWSBS/RS Inbound Jobs' Scheduling

FWSBS/RS's inbound-scheduling problem has a large number of solutions, making it difficult to solve using an exact method. The search for optimal solutions using the genetic algorithm (GA) is based on the laws of evolution that govern the biological world. For the inbound-scheduling problem of FWSBS/RS, this paper improves the GA's local-search capability and prevents it from accidentally hitting the local optimum, by using the double-layer encoding method to create the chromosome and integrating the simulated annealing algorithm into the GA's process. The following are our improvements to the traditional genetic algorithm to overcome its premature convergence and weak local-search ability.

5.1. Construct Individuals and Initialization

The double-layer encoding method applied to represent the chromosome segments of equipment and cargo for FWSBS/RS inbound-operation scheduling, augments the search-space dimensions, thereby enhancing the algorithm's parallel search capabilities. Figure 4 is an example of the method. There is an order-sorting (OS) layer and machine-sorting (MS) layer in a chromosome. In the OS layer, the genes (1, 2, 3, 4) indicate the amount of cargo to be inbound; in the MS layer, the genes (1), (2, 3) (4, 5, 6) indicate the number of pickers, elevator numbers, and four-way-shuttle numbers, respectively. The sequencing of cargo arrival and task completion by equipment can be depicted, since each chromosome corresponds to a specific scheduling scheme.

OS	2	1	4	3	1	3	2	4	3	1	4	2
MS	1	1	1	1	2	3	3	2	4	5	5	6

Figure 4. Double-layer encoding method.

The initial population is produced utilizing the random-generating and greedy-initialization methods following the completion of the encoding [37]. For the random-generation approach, the initial population is created by randomly permuting the MS-section numbers and their corresponding OS-section numbers while following the given constraint conditions. For the greedy-initialization strategy, the device with the quickest current completion-of-operation time is chosen in the device-selection portion of the MS section, and random-ordering generation is used to produce the final initial population in the OS section.

5.2. Decode Method

In order to determine the start and stop times for the devices involved in warehousing the cargo, the double-layer encoded chromosome must be decoded. The current inbound cargo is scheduled at the earliest opportunity, and the cumulative calculation of the completion time of the last inbound cargo is carried out to obtain the scheduling time corresponding to the chromosome [38]. A semi-active scheduling decoding strategy is used for decoding. When the constraints are satisfied, the inbound order of the double-layer encoded OS is traversed sequentially and corresponds to the order of the equipment in the MS section.

5.3. Adaptation Function and Selection Operation

Formula (14) illustrates the aim in this study, which is to minimize the operation time for finishing a batch jobs which are inbound. The fitness function is set to be the inverse of the objective function, where $f(q)$ indicates the q-th chromosome's fitness value. In the selection process, the fitness of the population's members is first standardized. The normalized value of fitness P is shown in Formula (15), and a roulette wheel method is used to generate a random-probability value based on the normalized fitness value of individual i, and where f_i denotes the fitness value of individual i, and $f_{\max}$ and $f_{\min}$ denote the maximum and minimum values of fitness in the population, respectively. The individual is chosen as the parent, and the selection procedure is repeated until a sufficient number of parent people have been chosen if this probability value is smaller than the individual's p value. The population as a whole becomes more fit thanks to this strategy, which keeps fitter individuals around and enhances the likelihood that they will be chosen in future generations.

$$f(q) = \frac{1}{T_{sum}} \tag{14}$$

$$P = \frac{f_i - f_{\min}}{f_{\max} - f_{\min}} \tag{15}$$

5.4. Crossover Operation

The POX crossover operator suggested in the literature [39] is used to make sure that the individual after cross-operation is still a workable solution. The steps for POX based on double-layer encoding are as follows:

Step 1. Randomly divide the n jobs in the OS layer into two non-empty complementary subsets I_1, I_2;

Step 2. Copy the genes that belong to I_1 from parent P_1 to child C_1 in the same location. Reposition the parts from I_1 in parent P_2 location in child C_2;

Step 3. Copy the remaining elements in parent P_2 to the remaining empty positions in child C_1 in order. Copy the remaining elements in parent P_1 to the remaining empty positions in child C_2 in order.

The resultant child individual is always a workable solution since the device number of the MS layer at the crossover time is replicated together with the cargo number of the related OS layer to the corresponding child individual.

5.5. Mutation Operation

To maintain population diversity, different variants of the bilayer code are used to increase chromosome diversity and the ability to explore the search space.

By selecting two cargo numbers at random, a, and b, and discovering all of the positions of a, b in the chromosomes, positional variation was applied to the chromosomes in the OS layer of the cargo selection. The new chromosomes were obtained by swapping the OS in the chromosomes of a, b with the MS at the same time position.

For the MS layer of the device selection, the chromosomal genes are operated by single-point mutation. Using this approach, the population is mutated when the MS code corresponding to the OS code fulfills the machine selection-order constraints.

5.6. Simulated Annealing Strategy

After the above selection, crossover, and mutation operations to generate a new population of $P' = \left\{ \left[S'_1\right], [S'_2], \cdots, [S'_n] \right\}$ to improve the algorithm's optimization of search ability, the fitness function is used as the objective function of the simulated annealing operation, and the annealing increment is calculated as

$$\Delta f = fitness(S_i) - fitness(S_i') \tag{16}$$

$$\exp(-\Delta f / T_i) \tag{17}$$

In Formula (16), accept chromosome S_i' in the population of the following generation if $\Delta f < 0$; otherwise, accept chromosome S_i' with the probability given in Formula (17). The simulated-annealing-temperature updated rule is $T_{i+1} = \lambda T_i$, where T_i is the simulated annealing temperature and λ is the cooling rate.

6. Simulation Experiment and Verification

6.1. Algorithm Effectiveness Analysis

6.1.1. Analysis of Small-Scale Inbound Jobs

The single-depth FWSBS/RS is used as an example in this paper, and the initial system parameters are shown in Table 2. Three algorithms, namely, ant-colony optimization (ACO), genetic algorithm (GA), and the proposed double-encoded local optimization genetic algorithm (DELO-GA), are applied to validate the results for small-scale instances. The key parameters for these three algorithms are set as shown in Table 3, with some parameters of the ACO being adapted from the literature [40].

Table 2. Initial-system-parameters configuration.

Parameters	Value	Unit	Parameters	Value	Unit
H	0.8	m	P	12	-
L	1	m	t_s	10	s
W_C	1	m	t_E	2	s
W_m	1	m	t_R	2	s
W_s	1	m	a_R	2	m/s^2
T	5	-	$v_{R\max}$	2	m/s
M	6	-	a_E	1	m/s^2
N	10	-	$v_{E\max}$	2	m/s

Table 3. Parameter settings for the three algorithms.

Algorithms	Parameter Name	Value
ACO	Ant-colony size	50
	Pheromone-evaporation coefficient	0.8
	Information heuristic factor	1
	Expectation heuristic factor	5
	Degree of acceptance of random search	0.2
	Iteration count	400–1000
GA/DELO-GA	Initial population size	200
	Crossover probability	0.8
	Mutation probability	0.08
	Iteration count	400–1000
	Initial temperature	100
	Cooling coefficient	0.92

Small-scale instances (SI) are generated using a random approach, resulting in 10 sets of jobs, each comprising 10 cargos for inbounded, and the three algorithms are run on each set of data 30 times. The results are shown in Table 4, where 'Times' represents the completion time obtained using the enumeration method (EM) for each set of tasks; 'Avg' indicates the average completion time for each set of jobs; 'Best' denotes the best completion time achieved for each operation set; and 'Nun_opt' indicates the number of times the optimal solution was achieved using the algorithms.

Table 4. Comparison results of three algorithms for 10 small-scale instances.

SI	EM	ACO			GA			DELO-GA		
	Times/s	**Avg/s**	**Best/s**	**Num_opt**	**Avg/s**	**Best/s**	**Num_opt**	**Avg/s**	**Best/s**	**Num_opt**
1	144.80	146.74	144.80	26	145.13	144.80	25	144.80	144.80	27
2	136.80	137.71	136.80	25	138.80	136.80	24	137.13	136.80	29
3	111.31	115.04	111.31	14	113.53	111.31	23	111.95	111.31	28
4	131.28	134.10	131.28	19	132.50	131.28	22	132.03	131.28	26
5	149.46	150.84	149.46	18	150.54	149.46	25	149.87	149.46	27
6	162.41	162.75	162.41	26	163.08	162.41	22	162.50	162.41	29
7	147.07	149.28	147.07	23	149.58	147.07	21	147.92	147.07	26
8	168.31	171.19	168.31	22	169.98	168.31	25	168.98	168.31	28
9	114.83	120.61	118.66	17	120.06	118.66	23	114.83	114.83	30
10	151.63	153.44	151.63	24	152.96	151.63	26	152.29	151.63	28

Due to the fact that 10 instances are small-scale instances, an exhaustive enumeration method is employed to find the optimal solution for each group of jobs. As shown in Table 4, comparing the results of the enumeration method, ACO, GA, and the DELO-GA algorithm proposed in this paper, we observe that ACO and GA match the enumeration method in 9 sets but struggle to find the optimal solution in the 9th instance, being trapped in low efficiency and the local optimum. In contrast, the DELO-GA algorithm consistently discovers the optimal solution for all 10 instances.

As shown in Figure 5, the number of optimal solutions obtained using the DELO-GA algorithm is more than seven times that of the ACO and GA. Figure 6 shows that the difference between the average operating time and the optimal operating time of the 10 sets obtained is less than 2 s, validating the effectiveness of the algorithm proposed in this paper.

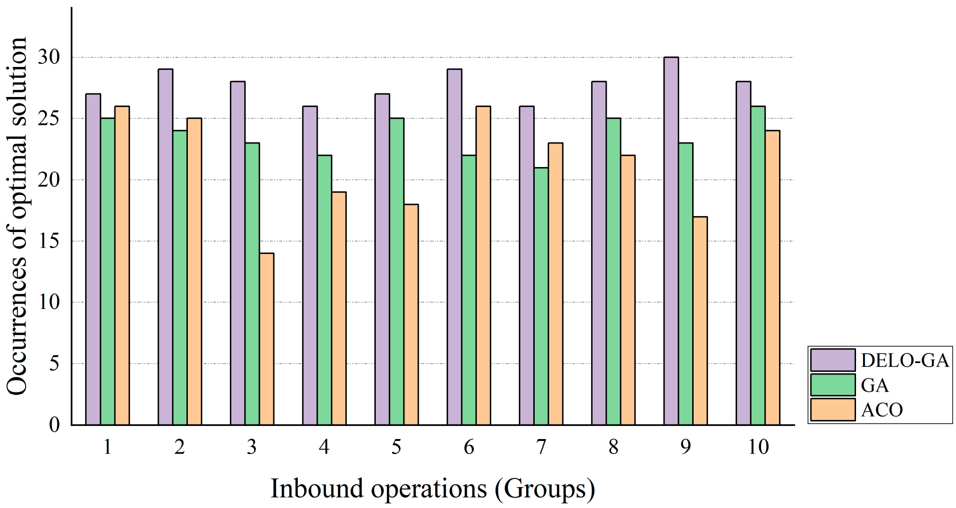

Figure 5. Occurrences of optimal solutions in 30 runs for three algorithms.

6.1.2. Analysis of Three Algorithms for FFSP Benchmark Instances

This paper investigates the inbound-scheduling problem of the FWSBS/RS by transforming it into a flexible flow-shop-scheduling problem. Due to the absence of standard benchmark instances for inbound scheduling in such systems, this study conducts comparative experiments using Taillard benchmark instances [41] to evaluate three proposed algorithms, and the website of these instances is: http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html (accessed on 20 December 2023). Additionally, the effectiveness of these algorithms is validated through comparative analysis with results obtained by Wei et al. [42].

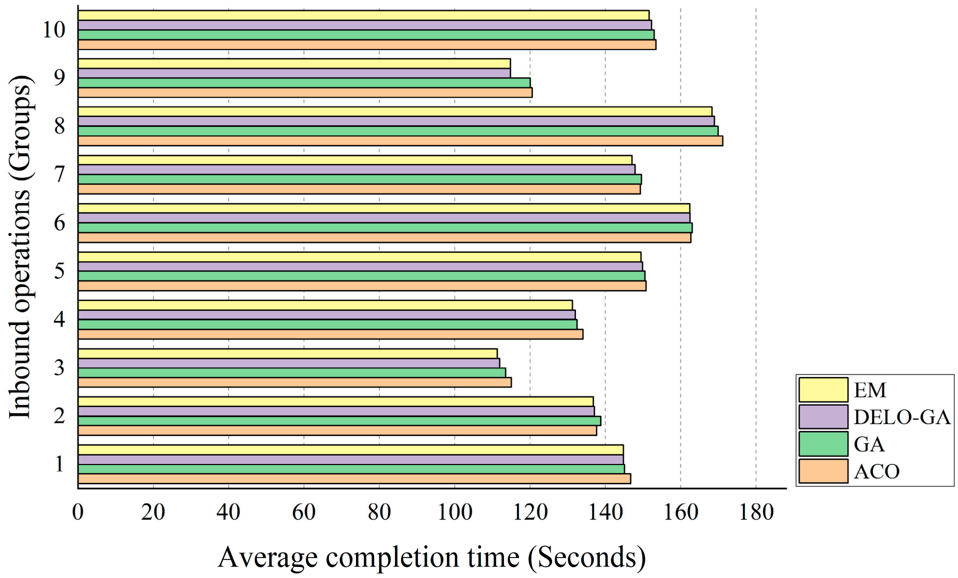

Figure 6. Comparison of results for 10 sets of inbound jobs using three algorithms and enumeration method.

For effective comparison, we utilized the optimal solutions of three scales of Taillard benchmark instances provided by Wei et al. [42] as the benchmarks for evaluating the effectiveness of our proposed algorithms. Each algorithm was applied to solve each case 30 times. ARPD represents the average relative percentage deviation between the results obtained using the algorithm and the optimal values. Here, C_{xi} denotes the solution of algorithm x on instance i, and $C_{\mathrm{Opt}i}$ stands for the optimal solution provided by the referenced literature. The calculation of ARPD can be completed using the Formula (18):

$$\mathrm{ARPD} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{C_{xi} - C_{\mathrm{Opt}i}}{C_{\mathrm{Opt}i}}\right| \times 100\% \tag{18}$$

The three algorithms proposed in this paper were individually executed 30 times on each benchmark instance. Table 5 presents the problem size (jobs × machines) for each instance, along with the best solution obtained using each algorithm, their respective maximum values, and the ARPD.

Table 5. Comparison of results for three algorithms on benchmark instances.

Instances	PS (J × M)	$C_{\mathrm{Opt}i}$	ACO			GA			DELO-GA		
			Best	Max	ARPD	Best	Max	ARPD	Best	Max	ARPD
Ta001	20 × 5	1324	1339	1357	2.51%	1357	1364	3.43%	1322	1360	1.26%
Ta011	20 × 10	1698	1713	1775	4.02%	1761	1812	5.92%	1658	1680	2.35%
Ta031	50 × 5	2731	2834	2979	5.19%	2879	3106	6.13%	2723	2799	3.98%

As analyzed from the data in Table 5, compared to GA and ACO, the DELO-GA algorithm has smaller deviations between the obtained solution and the optimal, with an ARPD less than 4% for larger scale problems (e.g., Ta031). These results indicate that DELO-GA has better robustness in addressing this problem as the size increases. Moreover, transforming the FWSBS/RS's inbound scheduling into an FFSP verifies the applicability and validity of DELO-GA in solving this type of scheduling problem.

6.2. Comparative Analysis of Three Algorithms on Various-Scale Storage Systems

To validate the applicability of the three proposed algorithms to storage systems with different layouts, 18 cases are configured as shown in Table 6, varying in the number of layers (M), sub-aisles (T), and storage positions per column (P).

Table 6. Layout parameters of different storage systems.

<table>
<tr><td>M</td><td colspan="2">4</td><td colspan="2">6</td><td colspan="2">8</td></tr>
<tr><td>T</td><td colspan="2">3</td><td colspan="2">5</td><td colspan="2">9</td></tr>
<tr><td>P</td><td colspan="3">10</td><td colspan="3">12</td></tr>
</table>

As indicated in Table 7, 50 random inbound jobs are generated for each layout. The three algorithms are executed 30 times, respectively, to solve each case, and the mean deviation (MD) of each algorithm is calculated to verify their applicability in inbound-scheduling operations for storage systems with various scales. The calculation of MD is defined using Formula (19).

$$\mathrm{MD} = \frac{1}{n}\sum_{i=1}^{n}|x_i - \overline{x}| \quad (19)$$

Table 7. Mean deviation of three algorithms for various-storage-system layouts.

Instances	**Layouts**			**MD**		
	M	***T***	***P***	**ACO**	**GA**	**DELO-GA**
1	4	4	10	2.90	3.10	2.23
2	4	4	12	3.59	4.50	3.59
3	4	5	10	4.36	4.78	3.65
4	4	5	12	4.11	5.20	3.78
5	4	8	10	4.64	5.03	3.52
6	4	8	12	4.00	4.82	3.56
7	6	4	10	2.82	3.93	2.37
8	6	4	12	2.97	3.90	2.88
9	6	5	10	2.90	3.76	2.35
10	6	5	12	2.12	2.19	1.91
11	6	8	10	3.19	2.85	2.48
12	6	8	12	3.48	4.01	2.92
13	8	4	10	3.41	3.22	2.23
14	8	4	12	2.77	3.67	2.59
15	8	5	10	2.58	3.06	2.16
16	8	5	12	2.62	3.34	2.02
17	8	8	10	2.48	2.96	2.56
18	8	8	12	2.71	3.46	2.36

From Table 7 and Figure 7, it is evident that the DELO-GA algorithm proposed in this paper consistently maintains a mean deviation of less than 4 across 30 independent tests while solving inbound operations for 18 layouts of the FWSBS/RS. Notably, for the 10th layout, all three algorithms produced minimal and closely aligned results. Therefore, further comparative analyses of increased batch quantities for inbound jobs are conducted in Section 6.3, focusing on Instance 10 (M = 6, T = 5, P = 12).

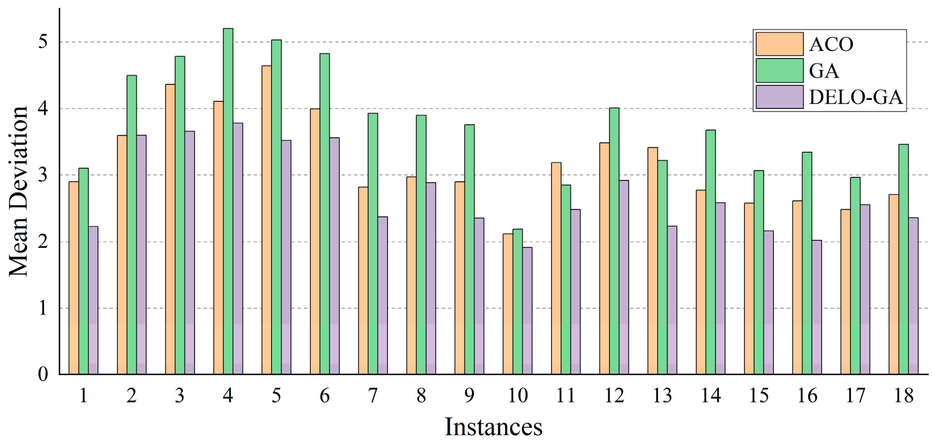

Figure 7. Mean deviation plot of storage-system solutions across different layouts.

6.3. Comparative Analyses of Three Algorithms for Large-Scale Jobs

For larger scale instances, including 20 or more inbound-operation jobs [43], the exact solution cannot be found using the enumeration method. Three algorithms, ACO, GA, and DELO-GA, are used to solve each group of inbound operation with the number of jobs of being 20, 40, 60, 80, and 100, respectively, and each group of jobs is solved using the three algorithms for 30 times. The three algorithms are programmed in Python, and are run and tested on a Windows 11 computer configured with Intel(R) Core(TM) i9-9900K CPU @3.60GHz and 32 GB RAM.

Table 8 shows the completion times for batch inbound jobs obtained by running three algorithms. The optimal-job-completion times are 210.83 s, 411.31 s, 613.83 s, 811.31 s, and 1013.83 s for five different scales, respectively. When comparing the results produced by the ACO and the conventional GA, the suggested DELO-GA algorithm's average optimization efficiency is 2.28% and 2.11%, respectively.

Table 8. Comparison of three algorithms for large-scale instances.

Scales	Algorithms	CPU Times/s	Avg/s	Best/s	MD
20	ACO	17.38	221.89	217.83	2.39
	GA	14.85	219.72	217.66	1.66
	DELO-GA	12.81	212.54	210.83	1.18
40	ACO	31.27	436.88	422.83	8.28
	GA	27.92	434.46	422.83	6.89
	DELO-GA	24.93	417.05	411.31	1.78
60	ACO	74.62	641.17	626.49	7.51
	GA	65.62	635.85	620.49	6.23
	DELO-GA	57.41	616.89	613.83	1.78
80	ACO	103.08	841.94	827.00	5.73
	GA	98.78	840.36	822.00	7.65
	DELO-GA	75.95	818.98	811.31	2.81
100	ACO	146.22	1063.31	1026.83	13.62
	GA	135.94	1075.83	1035.14	14.86
	DELO-GA	92.25	1021.95	1013.83	2.78

To analyze the results of 30 independent tests for each job group, standardization of the results is performed due to significant variations in average job times among different groups. Therefore, this paper employs a standardization procedure for the results of each group, computed using the following Formula (22):

$$\mu = \frac{1}{n}\sum_{i=1}^{n} t_i \tag{20}$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(t_i - \mu)^2} \tag{21}$$

$$Z_i = \frac{t_i - \mu}{\sigma} \tag{22}$$

where μ represents the mean of a set of test results; σ stands for the standard deviation of a set of test results; Z_i denotes the standardized value of test i; and the boxplot of standardized values for all sets is depicted in Figure 8.

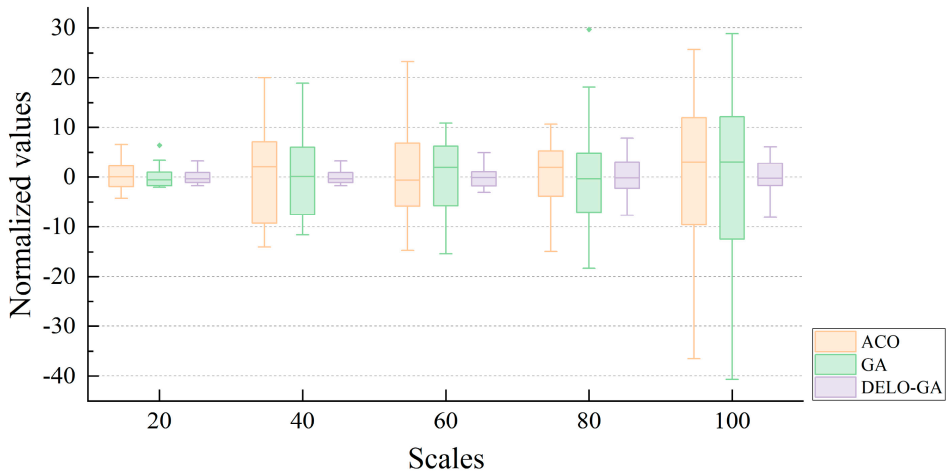

Figure 8. Boxplots of standardized results using three algorithms for different inbound jobs.

The mean-deviation (MD) index is proposed to compare the stability of the solutions generated using the three algorithms. As depicted in Table 8 and Figure 8, as the scale of FWSBS/RS's inbound operation increases, the solution stability of DELO-GA for large-scale-operation instances is superior to that of the other two algorithms, and the optimal operation time obtained is shorter compared to the optimal solution time of the other two algorithms. Further evidence of the superiority of the algorithm proposed in this study can be observed in the increasing improvement of its optimal result as the operation size grows, in contrast to the other two algorithms.

To comprehensively analyze the convergence rates of the ACO, GA, and DELO-GA when solving problems of different job sizes, we have plotted the iterative convergence curves of the three algorithms using the optimal solutions obtained, as shown in Figure 9. It is observed that, for job sizes of 20 and 40, the ACO exhibits a similar search capability as the GA. However, the ACO has a slower convergence rate compared to the GA. When solving problems with 60 and 80 jobs, the ACO is more prone to premature convergence, resulting in poorer optimal solutions. Notably, when dealing with 100 jobs, the GA quickly falls into a local optimum.

An analysis of the optimization results shows that the ACO and GA have average solution errors of 2.51% and 2.40%, respectively. In contrast, the DELO-GA exhibits an average solution error of 0.88%. Furthermore, the DELO-GA demonstrates faster convergence rates for all five job sizes, requiring less time to find the optimal batch inbound-job schedule. It can also escape the local optimal time more efficiently. Therefore, this study confirms the superior overall performance of the proposed algorithm, making it suitable for solving large-scale FWSBS/RS's inbound-scheduling problems.

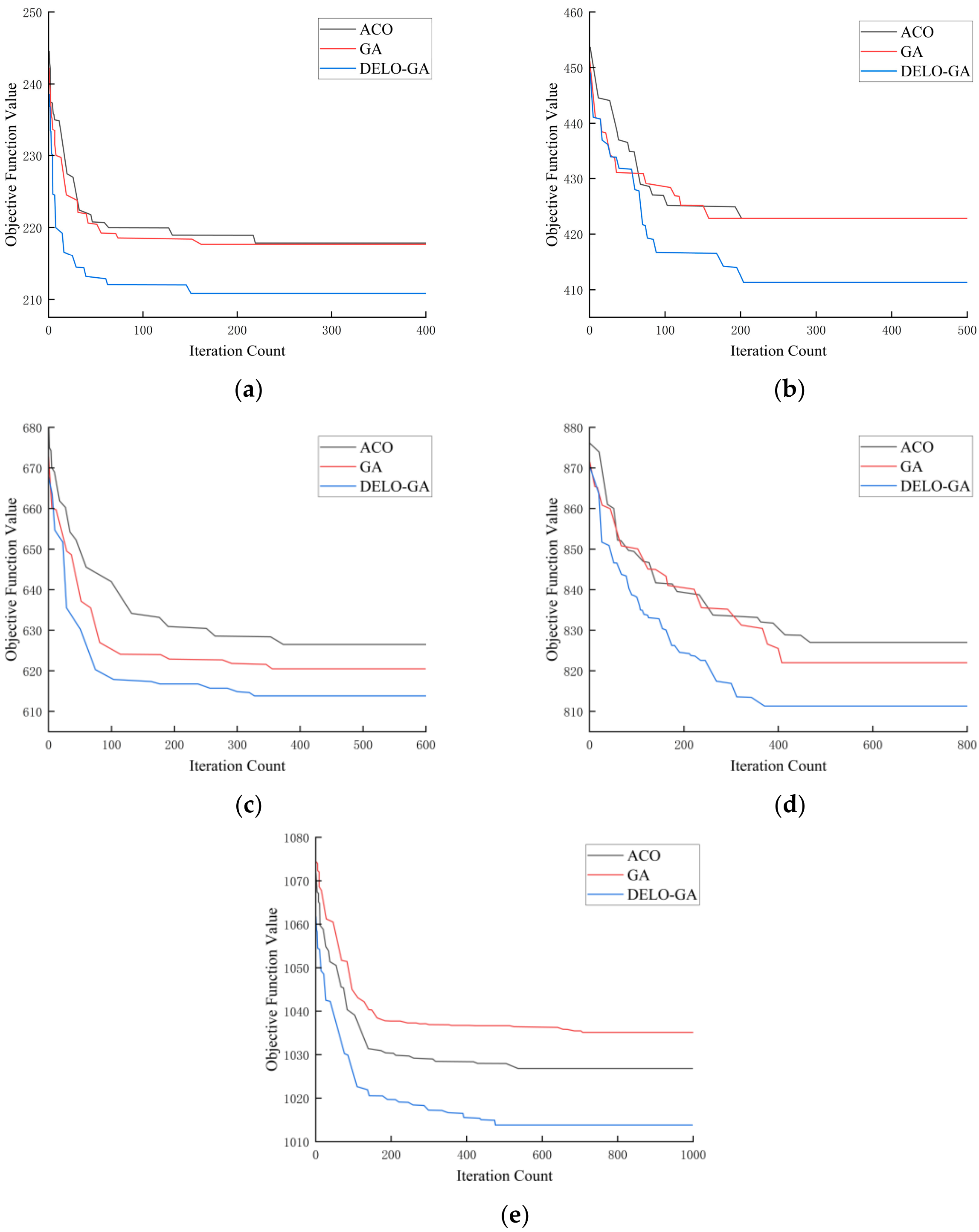

Figure 9. Convergence curves of the optimal solutions of the three algorithms under different job sizes: (**a**) 20 inbound jobs; (**b**) 40 inbound jobs; (**c**) 60 inbound jobs; (**d**) 80 inbound jobs; and (**e**) 100 inbound jobs.

7. Conclusions

At the core of this paper lies the examination of equipment's parallel-operation characteristics, inbound-job-scheduling modeling, and performance evaluation for FWSBS/RS.

Theoretically, we model the inbound-job-scheduling problem of FWSBS/RS as an FFSP model with the object of minimizing the maximum job-completion time, thereby enhancing the overall system efficiency and performance. By meticulously analyzing the inbound-job processes, considering the motion patterns of the elevators and four-way shuttles, and introducing job time constraints accounting for equipment acceleration and deceleration, we further enhance the model's precision.

Methodologically, to address the inbound-job-scheduling problem in FWSBS/RS, we propose an improved genetic algorithm based on double-layer encoding, termed DELO-GA. Experiment results demonstrate that when handling large-scale inbound jobs, DELO-GA reduces computational time by 26.6% and 18.4% compared to the ACO and traditional GA, confirming its superiority in terms of computational efficiency and accuracy. The introduction of this algorithm not only reduces batch inbound-job-completion times but also exhibits superior optimization capabilities and faster convergence rates, thereby enhancing the overall system performance. This contribution furnishes modern warehousing systems with a potent tool and methodology.

There are still areas for potential improvement in this study. Firstly, further consideration of practical constraints and factors such as equipment maintenance and failures are warranted to better reflect real-world operational conditions. Secondly, exploring more complex multi-objective optimization problems to balance different performance metrics is an avenue for future research. Lastly, validating the algorithm's applicability and robustness across a broader spectrum of warehousing systems is essential. Subsequent research could delve deeper into the application of artificial intelligence and deep-learning technologies in warehouse-system scheduling, and environmental sustainability, to optimize energy utilization and reduce environmental impact, could be an important focus area.

Author Contributions: Conceptualization, Z.W. and L.L.; data curation, B.Z.; formal analysis, Y.Z. (Yingxu Zhang) and L.L.; methodology, Z.W.; software, Y.Z. (Yingxu Zhang); validation, B.Z. and X.H.; visualization, Y.Z. (Yingxu Zhang); writing—original draft, Y.Z. (Yingxu Zhang); writing—review and editing, Z.Z. and Y.Z. (Yehao Zhang). All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Science and Technology Project of Henan Province in China (No. 232102221007); the Science and Technology Project of Henan Province in China (No. 242102220029); the Foundation of the Education Department of Henan Province (Grant No. 23A460003); and the fund of Henan Key Laboratory of Superhard Abrasives and Grinding Equipment (Grant No. JDKFJJ2022012).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Azadeh, K.; De Koster, R.; Roy, D. Robotized and Automated Warehouse Systems: Review and Recent Developments. *Transp. Sci.* **2019**, *53*, 917–945. [CrossRef]
2. Boysen, N.; Stephan, K. A survey on single crane scheduling in automated storage/retrieval systems. *Eur. J. Oper. Res.* **2016**, *254*, 691–704. [CrossRef]
3. Li, H.; Lyu, J.; Zhen, L.; Zhuge, D. A joint optimisation of multi-item order batching and retrieving problem for low-carbon shuttle-based storage and retrieval system. *Clean. Logist. Supply Chain* **2022**, *4*, 100042. [CrossRef]
4. Silva, A.; Coelho, L.; Darvish, M.; Renaud, J. Integrating storage location and order picking problems in warehouse planning. *Transp. Res. Part E Logist. Transp. Rev.* **2020**, *140*, 102003. [CrossRef]
5. Wang, H.; Ji, S.; Su, G. Research on Autonomous Vehicle Storage and Retrieval System Cargo Location Optimization in E-commerce Automated Warehouse. *IOP Conf. Ser. Mater. Sci. Eng.* **2020**, *790*, 012165. [CrossRef]
6. D'Antonio, G.; Paolo, C. Analytical models for cycle time and throughput evaluation of multi-shuttle deep-lane AVS/RS. *Int. J. Adv. Manuf. Technol.* **2019**, *104*, 1919–1936. [CrossRef]
7. Zhao, X.; Wang, Y.; Wang, Y.; Huang, K. Integer Programming Scheduling Model for Tier-to-Tier Shuttle-Based Storage and Retrieval Systems. *Processes* **2019**, *7*, 223. [CrossRef]
8. Luo, J.; Yin, H.; Li, B.; Wu, C. Path planning for automated guided vehicles system via interactive dynamic influence diagrams with communication. In Proceedings of the 2011 9th IEEE International Conference on Control and Automation (ICCA), Santiago, Chile, 19–21 December 2011; pp. 755–759.
9. Meng, T.; Liu, X.F. The AVS/RS modeling and path planning. *J. Appl. Sci. Eng.* **2015**, *18*, 245–250. [CrossRef]
10. Malmborg, C. Interleaving dynamics in autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* **2003**, *41*, 1057–1069. [CrossRef]

11. Malmborg, C.; Al-Tassan, K. An integrated performance model for orderpicking systems with randomized storage. *Appl. Math. Model.* **2000**, *24*, 95–111. [CrossRef]
12. Eldemir, F.; Graves, R.; Malmborg, C. A comparison of alternative conceptualizing tools for automated storage and retrieval systems. *Int. J. Prod. Res.* **2003**, *41*, 4517–4539. [CrossRef]
13. Marchet, G.; Melacini, M.; Perotti, S.; Tappia, E. Development of a framework for the design of autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* **2013**, *51*, 4365–4387. [CrossRef]
14. Kuo, P.-H.; Krishnamurthy, A.; Malmborg, C. Design models for unit load storage and retrieval systems using autonomous vehicle technology and resource conserving storage and dwell point policies. *Appl. Math. Model.* **2007**, *31*, 2332–2346. [CrossRef]
15. Malmborg, C. Conceptualizing tools for autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* **2002**, *40*, 1807–1822. [CrossRef]
16. Fukunari, M.; Malmborg, C. An efficient cycle time model for autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* **2008**, *46*, 3167–3184. [CrossRef]
17. Fukunari, M.; Malmborg, C. A network queuing approach for evaluation of performance measures in autonomous vehicle storage and retrieval systems. *Eur. J. Oper. Res.* **2009**, *193*, 152–167. [CrossRef]
18. Epp, M.; Wiedemann, S.; Furmans, K. A discrete-time queueing network approach to performance evaluation of autonomous vehicle storage and retrieval systems. *Int. J. Prod. Res.* **2016**, *55*, 960–978. [CrossRef]
19. Heragu, S.; Cai, X.; Krishnamurthy, A.; Malmborg, C. Analytical models for analysis of automated warehouse material handling systems. *Int. J. Prod. Res.* **2011**, *49*, 6833–6861. [CrossRef]
20. Roy, D.; Krishnamurthy, A.; Heragu, S.; Malmborg, C. Performance Analysis and Design Tradeoffs in Warehouses with Autonomous Vehicle Technology. *IIE Trans.* **2012**, *44*, 1045–1060. [CrossRef]
21. Roy, D.; Krishnamurthy, A.; Heragu, S.; Malmborg, C. Queuing models to analyze dwell-point and cross-aisle location in autonomous vehicle-based warehouse systems. *Eur. J. Oper. Res.* **2015**, *242*, 72–87. [CrossRef]
22. Ekren, B.; Heragu, S.; Krishnamurthy, A.; Malmborg, C. An Approximate Solution for Semi-Open Queueing Network Model of an Autonomous Vehicle Storage and Retrieval System. *Autom. Sci. Eng. IEEE Trans.* **2013**, *10*, 205–215. [CrossRef]
23. Ekren, B.Y.; Heragu, S.S.; Krishnamurthy, A.; Malmborg, C.J. Matrix-geometric solution for semi-open queuing network model of autonomous vehicle storage and retrieval system. *Comput. Ind. Eng.* **2014**, *68*, 78–86. [CrossRef]
24. Cai, X.; Heragu, S.; Liu, Y. Modeling and evaluating the AVS/RS with tier-to-tier vehicles using a semi-open queueing network. *IIE Trans.* **2014**, *46*, 905–927. [CrossRef]
25. Tang, H.-Y.; Juan, L. *An Improved Ant Colony Algorithm for Order Picking Optimization Problem in Automated Warehouse*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 2, pp. 1537–1547.
26. Zou, M.; Wang, Q.; Liu, S.-A. Optimization of Parking Space Allocation for Automated Parking System of Paternoster Type by Genetic Algorithm. In Proceedings of the 2019 Chinese Control and Decision Conference (CCDC), Nanchang, China, 3–5 June 2019; pp. 3834–3838.
27. Yang, D.; Wu, Y.; Huo, D. Research on Design of Cross-Aisles Shuttle-Based Storage/Retrieval System Based on Improved Particle Swarm Optimization. *IEEE Access* **2021**, *9*, 67786–67796. [CrossRef]
28. Cao, W.; Zhang, M. The Optimization and Scheduling Research of Shuttle Combined Vehicles in Automated Automatic Three-dimensional Warehouse. *Procedia Eng.* **2017**, *174*, 579–587. [CrossRef]
29. Song, J.; Yang, M.; Zhou, X. Scheduling Optimization of Automated Storage and Retrieval System Based on Four-Way Shuttles. In Proceedings of the 2020 IEEE International Conference on Mechatronics and Automation (ICMA), Beijing, China, 13–16 October 2020; pp. 524–529.
30. Li, M.; Li, L.; Zhang, C.; Jiang, L.; Liu, H.; Lin, Z.; Wei, L. A Four-Way Shuttle Scheduling Method Based on Grey Wolf Algorithm. In Proceedings of the 2022 China Automation Congress (CAC), Xiamen, China, 25–27 November 2022; pp. 1778–1784.
31. Wang, Y.; Mou, S.; Wu, Y. Task scheduling for multi-tier shuttle warehousing systems. *Int. J. Prod. Res.* **2015**, *53*, 5884–5895. [CrossRef]
32. Ganbold, O.; Kundu, K.; Li, H.; Zhang, W. A Simulation-Based Optimization Method for Warehouse Worker Assignment. *Algorithms* **2020**, *13*, 326. [CrossRef]
33. Zhan, X.N.; Xu, L.Y.; Ling, X.F. Task Scheduling Problem of Double-Deep Multi-Tier Shuttle Warehousing Systems. *Processes* **2021**, *9*, 41. [CrossRef]
34. Wang, H. Flexible flow shop scheduling: Optimum, heuristics and artificial intelligence solutions. *Expert Syst.* **2005**, *22*, 78–85. [CrossRef]
35. Ekren, B. Performance evaluation of AVS/RS under various design scenarios: A case study. *Int. J. Adv. Manuf. Technol.* **2011**, *55*, 1253–1261. [CrossRef]
36. Zhao, N.; Luo, L.; Lodewijks, G. Scheduling two lifts on a common rail considering acceleration and deceleration in a shuttle based storage and retrieval system. *Comput. Ind. Eng.* **2018**, *124*, 48–57. [CrossRef]
37. Zhang, Z.; Lu, R.; Zhao, M.; Luan, S.; Bu, M. Robot path planning based on genetic algorithm with hybrid initialization method. *J. Intell. Fuzzy Syst.* **2021**, *42*, 2041–2056. [CrossRef]
38. Kim, J.W. The Decoding Approaches of Genetic Algorithm for Job Shop Scheduling Problem. *J. Inf. Syst.* **2016**, *25*, 105–119. [CrossRef]

39. Fang, Y.; Peng, C.; Lou, P.; Zhou, Z.; Hu, J.; Yan, J. Digital-Twin-Based Job Shop Scheduling Toward Smart Manufacturing. *IEEE Trans. Ind. Inform.* **2019**, *15*, 6425–6435. [CrossRef]
40. Huang, M.; Guo, D.; Liang, X.; Liang, X. An Improved Ant Colony Algorithm is Proposed to Solve the Single Objective Flexible Job-shop Scheduling Problem. In Proceedings of the 2020 IEEE 8th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 20–22 November 2020; pp. 16–21.
41. Taillard, E. Benchmarks for Basic Scheduling Problems. *Eur. J. Oper. Res.* **1993**, *64*, 278–285. [CrossRef]
42. Wei, H.; Li, S.; Jiang, H.; Hu, J.; Hu, J. Hybrid Genetic Simulated Annealing Algorithm for Improved Flow Shop Scheduling with Makespan Criterion. *Appl. Sci.* **2018**, *8*, 2621. [CrossRef]
43. Li, J.; Gu, X.; Zhang, Y.; Zhou, X. Distributed Flexible Job-Shop Scheduling Problem Based on Hybrid Chemical Reaction Optimization Algorithm. *Complex Syst. Model. Simul.* **2022**, *2*, 156–173. [CrossRef]