

Article

A Multi-Objective and Multi-Dimensional **Optimization Scheduling Method Using a Hybrid Evolutionary Algorithms with a Sectional Encoding Mode**

Wenxiang Xu^D and Shunsheng Guo *

School of Mechanical and Electronic Engineering, Wuhan University of Technology, Wuhan 430070, China; xuwenxiang910327@126.com

* Correspondence: guoshunsheng@whut.edu.cn

Received: 22 January 2019; Accepted: 22 February 2019; Published: 4 March 2019



Abstract: Aimed at the problem of the green scheduling problem with automated guided vehicles (AGVs) in flexible manufacturing systems (FMS), the multi-objective and multi-dimensional optimal scheduling process is defined while considering energy consumption and multi-function of machines. The process is a complex and combinational process, considering this characteristic, a mathematical model was developed and integrated with evolutionary algorithms (EAs), which includes a sectional encoding genetic algorithm (SE-GA), sectional encoding discrete particle swarm optimization (SE-DPSO) and hybrid sectional encoding genetic algorithm and discrete particle swarm optimization (H-SE-GA-DPSO). In the model, the encoding of the algorithms was divided into three segments for different optimization dimensions with the objective of minimizing the makespan and energy consumption of machines and the number of AGVs. The sectional encoding described the sequence of operations of related jobs, the matching relation between transfer tasks and AGVs (AGV-task), and the matching relation between operations and machines (operation-machine) respectively for multi-dimensional optimization scheduling. The effectiveness of the proposed three EAs was verified by a typical experiment. Besides, in the experiment, a comparison among SE-GA, SE-DPSO, H-SE-GA-DPSO, hybrid genetic algorithm and particle swarm optimization (H-GA-PSO) and a tabu search algorithm (TSA) was performed. In H-GA-PSO and TSA, the former just takes the sequence of operations into account, and the latter takes both the sequence of operations and the AGV-task into account. According to the result of the comparison, the superiority of H-SE-GA-DPSO over the other algorithms was proved.

Keywords: green scheduling; automated guided vehicle; flexible manufacturing system; multi-objective and multi-dimensional; energy consumption; genetic algorithm; discrete particle swarm optimization

1. Introduction

With the development of the manufacturing industry and the changes of the competitive market, meeting customer's diverse demands and improving service quality has become the main direction of shifting their strategy for manufacturing enterprises, and flexible manufacturing system (FMS) become an effective way to meet these needs. At the same time, intelligent manufacturing has become a major trend in the development of manufacturing [1-3]. Under the background, FMS provides great flexibility for the intelligent manufacturing workshop and plays more and more important roles in intelligent manufacturing, especially to meet concurrent production of various parts on one or more pieces of large equipment. An FMS consists of material handling devices (automated guided



vehicles (AGVs) and robots), workstations, automated storage systems, and so on. An AGV is an automatic transport vehicle, which can navigate along a planned route with different guidance ways and systems [4]. It is widely used to transfer material in modern production systems and enhance the efficiency of it [5–7]; all AGVs can have unified scheduling using a central computer control system so that all shop floor operations would be controlled through an AGVs system.

AGVs scheduling of FMS is an important problem, which can affect the productivity, delivery cost and service quality to a great extent; it also determines the efficiency of the whole production system [8]. In most of the existing studies, the sequence of operation of related jobs and the matching relation between transfer tasks and AGVs (AGV-task) are the main ways to optimize the AGV scheduling problem; they are two related dimensions, which are important and effective in AGV scheduling [9]. Deroussi et al. proposed a solution method for the simultaneous scheduling for the sequence of operation and allocation of AGVs [10]. Lacomme et al. introduced a framework based on a disjunctive graph to model the joint scheduling problem of machines and AGVs [11]. Baruwa et al. proposed a simultaneous scheduling method of machines and AGVs based on a timed colored petri net approach [12]. Besides, there are also some practical conditions in the workshop that have been taken into consideration by researchers in their studies, such as battery charging of AGVs, cost of operation, time of operation, and so on [13–16]. Cai et al. developed a mixed regional control model for production tasks to solve the problem of task scheduling and coordination control presented by a multi-AGV system [17]. Novas et al. proposed a novel approach by taking resource-constraints, the loaded and the empty movements of the device, into account [18]. Mchaney proposed several methods that can be used to account for the impact of various battery usage schemes on AGV simulations [19]. Kabir et al. investigated how the duration of battery charging for AGVs can be varied to increase flexibility and manufacturing capacities of a manufacturing system [20]. Yan et al. investigated the capability to evaluate reliability issues in AGVs by considering the health management of these vehicles and their optimal mission configuration [21]. Although a lot of studies has been carried out for AGV scheduling by building different models, there are also some other conditions that need to be considered for further optimization of AGVs scheduling. In FMS, a lot of machines are versatile; it means that a single machine can process multiple operations and has different processing capabilities for different operations. Therefore, it is useful to optimize machine selection in the process of manufacturing by optimization of AGV scheduling. A similar problem has been studied by many researchers for manufacturing resource combinatorial optimization (MRCO) [22–25] and has provided important references for the optimization of machine selection in FMS by AGVs.

In the aspect of optimization objectives of AGVs scheduling in FMS, earlier studies have mainly concentrated on minimizing the makespan of all related jobs, and have been effectively applied on increasing productivity and the resource utilization rate [8,15,16]. Sometimes, the makespan is presented in the form of the travel distance of all AGVs [26]. Caridá et al. proposed a method for AGV scheduling in FMS using fuzzy systems with the objective of minimizing the makespan [27]. Chang et al. proposed a hybrid genetic algorithm (HGA) to improve the makespan solution for the distributed FMS scheduling problem [28]. Achmad et al. proposed a non-dominated sorting biogeography-based optimization (NSBBO) scheduling method in FMS with the objective of minimizing the makespan problem and total earliness [29]. However, as one of the most important parts in FMS, the number of them heavily influences the profitability in an FMS and increasing the utilization rate of AGVs is important for optimizing the performance of the FMS [30-32]. Some studies have been carried out to meet the demand for better optimal scheduling of AGVs in FMS. Mousavi et al. proposed an optimal scheduling model for AGVs that the objectives of which are the minimization of the makespan and number of AGVs while considering the AGVs battery charge; two optimization algorithms were also developed in the model [15,16]. Vivaldini et al. presented a methodology for the estimation of the minimum number of AGVs required to execute a given transportation order within a specific time-window [33]. With the development of green manufacturing, studies have been done for a green workshop by considering the energy consumption of the machines, which is also an important running

In summary, although a lot of effective studies have been done on the AGV scheduling problem in FMS, there is still value and a need for further research into the two problems as follows:

(1) It is of great value for enterprises to take more objectives such as energy consumption of machines into account for synthetically optimizing the manufacturing process and benefit the FMS.

(2) In FMS, optimizing the sequence of operations and the matching relation between transfer tasks and AGVs are just two dimensions to improve the performance of the production system, but, as we know, many machines have a variety of process capabilities, and optimizing machine selection should be an important part for increasing the resource utilization rate and the productivity in FMS.

To address the above concerns, this research developed a multi-objective and multi-dimensional optimization scheduling mathematical model while considering energy consumption and multi-function of the machines. In the model, the multi-objective is to minimize the makespan and energy consumption of the machines and the number of AGVs; the multi-dimensional objective is to simultaneously optimize the sequence of operation of related jobs, the matching relation between transfer tasks and AGVs (AGV-task) and the matching relation between operations and machines (operation-machine) for the multi-objective. To solve this kind of problem, evolutionary algorithms (EAs) are the common methods [9,37–39]. Some distinct research in the scheduling context that can also provide some new ideas for FMS with AGVs [40–42]. The model proposed in this research will be optimized in three-dimensions, as mentioned earlier using three evolutionary algorithms (sectional encoding genetic algorithm (SE-GA), sectional encoding discrete particle swarm optimization (SE-DPSO) and hybrid sectional encoding genetic algorithm and discrete particle swarm optimization (H-SE-GA-DPSO)) and the corresponding experiments and a comparison among them are introduced. Besides, the hybrid genetic algorithm, particle swarm optimization (H-GA-PSO) and tabu search algorithm (TSA) are also applied for the comparison; the former just takes the sequence of operation into account, which is proposed in literature [15], and the latter takes both the sequence of operation and AGV-tasks into account, which is proposed in literature [9].

The rest of this paper is organized as follows. The model, problem description and assumptions are detailed in Section 2. The algorithm design is explained in Section 3. The experiment and discussion are presented to verify the effectiveness of the algorithms proposed in this paper in Section 4. Especially, a comparison among results SE-GA, SE-DPSO, H-SE-GA-DPSO, H-GA-PSO and TSA is provided to prove the superiority of H-SE-GA-DPSO. Finally, Section 5 provides the conclusion of this paper and indicates the research direction to a further extent.

2. Problem Description and Mathematical Model

2.1. Problem Description

On the basis of the current AGV multi-objective optimization scheduling problems in FMS, a multi-objective and multi-dimensional scheduling model of AGVs is presented in this paper, as shown in Figure 1. Let us assume that there are several jobs assigned to an FMS, which can be described as a set of jobs $\{J_1, J_2, ..., J_n\}$. Each of the jobs contains several operations. For instance, J_i is composed of the sequence of operations $\{O_i(1), O_i(2), ..., O_i(m_i)\}$. Some of the related machines $\{M_1, M_2, ..., M_s\}$ in this FMS are selected to execute these operations with limited AGVs $\{A_1, A_2, ..., A_t\}$, and each of them can meet one or more of the operations $\{O_a(b), O_c(d), ..., O_e(f)\}$. The same operation can be executed by different machines with different energy consumption. Therefore, the corresponding candidate machines for $O_i(j)$ are marked as $\{M_x, M_y, ..., M_z\}$. In this model, the operating time and energy consumption of each operation executed on each candidate machine, standby energy consumption of each machine per unit time, logistics time among all machines are pre-known. The assumptions in the model are as follows:

- All AGVs have a unit-load capacity.
- There are no battery charge problems on any AGV.
- AGVs and machines operate continuously without breakdown.
- There are no traffic problems, collision, deadlock.
- AGV loading and unloading times are fixed and considered as travel times.
- AGVs can always park at their unloading locations.
- The velocity of AGVs is constant.
- The start point (*SP*) of each job is in the home position (*H*) of the AGV.
- The machine-to-machine distances and SP-to-machine distances are known.
- Each machine operates only one product at a time.
- The setup times are included in the time of production.



Figure 1. Schematic diagram of multi-objective optimization scheduling problems in the model by considering multiple optimization dimensions.

To formulate the mathematical model of the problem, the related parameters and variables are summarized in the Appendix A attached to this paper.

2.2. Mathematical Model

Three objects, which include the makespan, energy consumption and the number of AGVs are taken into account in the mathematical model. It can be formulated as follows.

Minimizing the makespan. The mathematical expression of the makespan (*MS*) can be expressed by:

$$MS = \max\{MFT_k\} = \max\{AFT_l + T(k, N_k)\} = \max\{ot(i)\}$$

= max{tt(i)+ot(i, m_i)} (1)

$$MFT_{k} = \sum_{p=1}^{N_{k}} \left(T(k, p) + WT(k, p) \right)$$
(2)

$$AFT_{l} = \sum_{q=1}^{N_{l}} \left(T(l,q) + WT(l,q) \right)$$
(3)

$$ot(i) = ot_s(i,1) + \sum_{j=1}^{m_i} (ot(i,j) + otw(i,j))$$
(4)

$$tt(i) = tt_s(i,1) + \sum_{j=1}^{m_i} (tt(i,j) + ttw(i,j))$$
(5)

$$WT(k,p) = T_s(k,p) - T_e(k,p-1)$$
 (6)

$$WT(l,q) = T_s(l,q) - T_e(l,q-1)$$
(7)

$$otw(i,j) = ot_s(i,j) - ot_e(i,j-1)$$
(8)

$$ttw(i,j) = tt_s(i,j) - tt_e(i,j-1)$$
(9)

$$ot(i, j), T(k, p) = ot_e(i, j) - ot_s(i, j),$$

if $O_i(j)$ is assigned to $M(k)$ and index of it for $M(k)$ is p (10)

$$tt(i, j), T(l, q) = tt_e(i, j) - tt_s(i, j),$$

if TA_{ij} is assigned to AGV *l* and index of it for AGV *l* is *q* (11)

Subject to:

$$MS \le DT$$
 (12)

$$\sum_{i=1}^{n} m_i = \sum_{k=1}^{s} N_k \tag{13}$$

$$\sum_{i=1}^{n} m_i = \sum_{l=1}^{t} N_l.$$
(14)

Here, Equation (1) defines the calculation method of MS, which can be represented by the maximum finish time of the operations assigned to each machine (i.e., $max\{MFT_k\}$), the maximum finish time of traveling tasks assigned to each AGV (i.e., $max\{AFT_l+T(k,N_k)\}$), the maximum finish time of operations of each job (i.e., $max\{ot(i)\}$) or the maximum finish time of traveling for each job (i.e., $max\{ot(i)\}$) or the maximum finish time of traveling for each job (i.e., $max\{tt(i)+ot(i,m_i)\}$). Equations (2) and (6) define the calculation method of related values, which determine the result of $max\{MFT_k\}$. Equations (3) and (7) define the calculation method of related values which determines the result of $max\{AFT_l+T(k,N_k)\}$. Equations (4), (8) and (10) define the calculation method of related values, which determines the result of $max\{ot(i)\}$. Equations (5), (9) and (11) define the calculation method of related values, which determines the result of $max\{tt(i)+ot(i,m_i)\}$.

Constraint number (12) ensures the feasibility of completion time by meeting the demand of DT. Constraint numbers (13) and (14) define the numerical relationship of m_i , N_k and N_l in Equations (1)–(5).

Minimizing energy consumption. The mathematical expression of energy consumption can be expressed by:

$$ES = esw + ese \tag{15}$$

$$esw = \sum_{k=1}^{s} esw(k) = \sum_{k=1}^{s} \sum_{p=1}^{N_{k}} WT(k, p) \times wesu(k) = \sum_{i=1}^{n} \sum_{j=1}^{m_{i}} otw(i, j) \times wesu(k)$$
(16)

$$ese = \sum_{k=1}^{s} ese(k) = \sum_{k=1}^{s} \sum_{p=1}^{N_k} es(k,p) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} es(i,j).$$
(17)

Here, Equation (15) defines the *ES* of all machines by completing all jobs. Equations (16) and (17) define the calculation methods of *esw* and *ese*, which compose *ES*. In Equation (16), $otw(i,j) \times wesu(k)$ means the M_k is chosen to executing the operation number j of job number i.

Minimizing the number of AGVs. The number of AGVs is denoted by *NA*. Generally, more AGVs means a smaller makespan while it also means increased costs. Therefore, it is important to minimize the number of AGVs to optimize the performance of the FMS.

Multi-objective evaluation. The decision is usually made by considering a comprehensive result when there are several objectives that have to be taken into account. Pareto has given an effective method for us to solve multi-objective optimization problems. In the method, Pareto-optimal set and

Pareto-front refer to a group of best trade-off schedules and a set of Pareto solutions respectively [15]. The overall fitness function formulation for the three objectives is described by:

$$f(x) = \varphi_1 f_1(x) + \psi_1 \varphi_2 f_2(x) + \psi_2 \varphi_3 f_3(x)$$
(18)

$$\varphi_1 + \varphi_2 + \varphi_3 = 1\varphi_1 > 0, \ \varphi_2 > 0, \ \varphi_3 > 0.$$
 (19)

 φ_1 , φ_2 and φ_3 are the weights of three objectives respectively and they are constrained by Equation (19). $f_1(x)$, $f_2(x)$ and $f_3(x)$ in Equation (18) correspond to the fitness functions of the three objectives. Therefore, the overall fitness function is described by:

$$f(x) = \varphi_1 M S + \psi_1 \varphi_2 E S + \psi_2 (1 - \varphi_1 - \varphi_2) N A.$$
(20)

 ψ_1 and ψ_2 in Equation (18) are the ratios to balance among the objectives with different ranges of value [41]. To facilitate the calculation and analysis by considering the main part of the *ES* and *NA*, they are defined by:

$$\psi_1 = \frac{DT}{max(ese)} \tag{21}$$

$$\psi_2 = \frac{DT}{max(NA)}.\tag{22}$$

The expression *max(ese)* in Equation (21) is the maximum energy consumption by completing the operation of all jobs.

3. Algorithm Design

The genetic algorithm (GA) and particle swarm optimization (PSO) are the most typical evolutionary algorithms (EAs) to solve AGV scheduling problems. The GA is a search algorithm based on the mechanics of the natural selection process. It has the capability of simultaneous evaluation of many points in the search area, which increases the probability of finding the global solution to the problem [15]. Besides, the encoding method of the GA is very flexible [43–46], and it can be used to solve different types of problems using a suitable encoding format. PSO is a population EA, which is inspired by social behavior of bird flocking; this kind of algorithm makes use of the population wisdom to search cooperatively so as to find the optimal result in the solution space. Compared with other algorithms, it is more robust as it can work with limited information such as the fitness evaluation of each particle [47]. The encoding method of PSO can meet the continuous and discrete variable values, therefore, it can be used to solve both the extremum approximation and MRCO problems [48,49]. To improve the performance of the algorithms, hybrid EAs have also been studied in a lot of research by integrating the advantages of the compensatory properties of each algorithm to obtain better results [50–52].

On this basis, a hybrid sectional encoding genetic algorithm and discrete particle swarm optimization (H-SE-GA-DPSO) have been developed to meet the model proposed in this paper. It is a hybrid algorithm, which has the characteristics of the genetic algorithm and particle swarm optimization. In the algorithm, both the chromosome encoding and particle encoding are divided into three segments for multi-dimensional optimization scheduling. Correspondingly, sectional encoding genetic algorithm (SE-GA), sectional encoding discrete particle swarm optimization (SE-DPSO), hybrid genetic algorithm and particle swarm optimization (H-GA-PSO) and a tabu search algorithm (TSA) are also proposed and compared with H-SE-GA-DPSO. Here, H-GA-PSO has been proven to be a more effective algorithm in optimizing the sequence of operations [15]. Besides, as two related dimensions, both the sequence of operation and AGV-tasks have been taken into account for further optimization in studies by TSA [9]. The related steps and configurations of SE-GA and SE-DPSO respectively are shown in Sections 3.1 and 3.2. Figure 2 illustrates the flowchart of H-SE-GA-DPSO, which is explained in detail in Sections 3.1 and 3.2.



Figure 2. The main flowchart and methods of hybrid sectional encoding genetic algorithm and discrete particle swarm optimization (H-SE-GA-DPSO).

3.1. Sectional Encoding Genetic Algorithm

The sectional encoding genetic algorithm (SE-GA) is the main algorithm of the proposed H-SE-GA-DPSO; it can also provide optimized solutions independently. The basis of it is the GA; the difference is that the chromosome encoding it is divided into three segments for multi-dimensional optimization scheduling. Therefore, the main steps of it are similar to the GA that can be described as follows:

Step 1. Initializing parameters. It involves setting the crossover rate (*CR*), the mutation rate (*MR*), population size (*PS*), length of a chromosome (*LC*), the maximum number of iterations (*NC*_{max}), related basic data and so on. Table 1 shows the data structure for the model proposed in this research. The column 'Gene code(g)' of 'SgSo', the column 'Gene code(g)' of 'SgOm' and the column 'Gene code(g)' of 'SgAt' show the code of the segment of sequence of operations (SgSo), segment of operation-machine (SgOm), segment of AGV-task (SgAt) of a chromosome (*C*_r) respectively, which will be discussed later.

Step 2. Initializing population. A set of chromosomes will be generated in this step.

Chromosome encoding. The encoding used in this paper is real number coding according to the needs of the problem. As shown in the column 'Gene code(g)' of 'SgSo' of Table 1, each gene code defines an operation related to a job. The order of genes represents the priority of operations, which decreases from left to right. In the column 'Gene code(g)' of 'SgOm', each gene code defines a machine to execute an operation. Different from SgSo, the order of it is fixed according to the original sequence of operations. In the column 'Gene code(g)' of 'SgAt', each gene code defines an AGV to execute a traveling task. The order of it is the same with SgOm.

			Chromos	some (C _r)			
	Sg	So	Sg	Om	Sg	At	-
	Gene Number (G _e)	Gene Code (g)	Gene Number (G _e)	Gene Code (g)	Gene Number (G _e)	Gene Code (g)	Operations (<i>O_i(j</i>))
SE-GA	G_1 G_2	1 1	$\begin{array}{c} G_{\theta+1} \\ G_{\theta+2} \end{array}$	1 3	$\begin{array}{c} G_{2\theta+1} \\ G_{2\theta+2} \end{array}$	1 3	$O_1(1) O_1(2)$
(<i>C</i> ₁)	Gm_1	 1 2	$G_{\theta^+} m_1$	 2 s	$G_{2\theta+} m_1$	 2 5	$O_1(m_1) O_2(1)$
		2		 5			$O_2(m_2)$
		n		 k		NA	$O_n(1)$
	$G_{ heta}$	 n	$G_{2\theta}$	 4	$G_{3\theta}$	 1	$O_n(m_n)$
			Partic	le (<i>P_r</i>)			
	Sg	So	Sg	Om	Sg	At	_
	Dimension Number (dn)	Dimension Code (<i>dc</i>)		Code of Dir	nension (<i>dc</i>)		- Operations (<i>O_i(j</i>))
	1	2		1		1	<i>O</i> ₁ (1)
	2	3		3	3	3	<i>O</i> ₁ (2)
(P_1)	· · · · 1111	 C	•	 2	•	· · 7	$O_1(m_1)$
(4 I)	1	U		s	Į	5	$O_2(1)$
			-	 5	•	 4	$O_2(m_2)$
		θ	-	 k	N	 IA	$O_n(1)$
	θ	1		 4	•	 1	\dots $O_n(m_n)$

A chromosome (C_r) is expressed by:

$$C_{r} = \{(O_{i}(j)), (M_{i}(j), (A_{i}(j))) | i = 1, 2, ..., n, j = 1, 2, ..., m_{i}\} = \{\{\underbrace{O_{2}(1), O_{1}(2), ..., O_{i}(j), ..., O_{n}(m_{n}), ..., O_{i}(m_{i})}_{\theta}\}, \\ \{\underbrace{(\underbrace{M_{1}(1), M_{1}(2), ..., M_{1}(m_{1})}_{m_{1}}, (\underbrace{M_{2}(1), M_{2}(2), ..., M_{2}(m_{2})}_{m_{2}}), ..., (\underbrace{M_{n}(1), M_{n}(2), ..., M_{n}(m_{n})}_{m_{n}})\}, \\ \{\underbrace{(\underbrace{A_{1}(1), A_{1}(2), ..., A_{1}(m_{1})}_{m_{1}}, (\underbrace{A_{2}(1), A_{2}(2), ..., A_{2}(m_{2})}_{m_{2}}), ..., (\underbrace{A_{n}(1), A_{n}(2), ..., A_{n}(m_{n})}_{m_{n}})\}\}\}$$

$$(23)$$

It meets the conditions as follows:

$$\sum_{i=1}^{n} m_i = \theta.$$
(24)

The length of a chromosome is expressed by:

$$LC = 3 * \theta. \tag{25}$$

Chromosome coding is explained by an example of three jobs (J_1 , J_2 , J_3), and each job has two, two, and three operations respectively, and C_r is a random construct of

Г

$$\underbrace{\begin{array}{c} 2312331 \\ SgSo} \underbrace{\begin{array}{c} 21 \\ J_1 \\ J_2 \\ SgSo} \underbrace{\begin{array}{c} 21 \\ J_1 \\ J_2 \\ SgSo} \\ \underbrace{\begin{array}{c} 21 \\ J_1 \\ J_2 \\ SgSo} \\ \underbrace{\begin{array}{c} 21 \\ J_1 \\ SgSo} \\ SgSo \\ SgSO$$

٦

of J_1 , J_2 , and J_3 respectively, and from the left, the first '1' represents the first operation of J_1 , the second '1' represents the second operation of J_1 , the first '2' represents the first operation of J_2 , and so on. In part of the SgOm, from the left, the first '2' represents the first operation of J_1 , executed by machine number 2. The first '1' represents the second operation of J_1 , executed by machine number 1. The first '4' represents the first operation of J_2 , executed by machine number 4, and so on. In part of the SgAt, from the left, the first '1' means that the traveling task of the first operation of J_1 is executed by AGV number 1. The second '3' means that the traveling task of the second operation of J_1 is executed by AGV number 3 and so on.

Step 3. Fitness evaluation. Each chromosome is evaluated by the makespan, energy consumption and number of AGVs in accordance with Equations (1) to (17). In addition, the total fitness values were calculated based on Equations (20) to (22).

Step 4. Generating a new population. A new population is generated by selection, crossover, and mutation operation.

Selection. The roulette method, which is a probability random selection method, was used in this study for the selection operator.

Crossover. A one-point crossover and a two-point crossover based on a partial string exchange were employed, where the two-point crossover is illustrated in Figure 3a,b based on the example in step 2. The offspring generated by the crossover may be an illegal encoding. For example, the uncorrected code of the operation of a job may be seen in part of the SgSo. Therefore, these illegal SgSo codes should be repaired as shown in Figure 3.



Figure 3. Example of two-point crossover of chromosomes.

The number of crossovers is calculated by:

Number of crossovers =
$$\frac{CR \times PS}{2}$$
. (26)

Mutation. The number of mutations in each generation is calculated by:

Number of mutations
$$\cong PS \times MR.$$
 (27)

A one-point mutation operator was used in this study as shown in Figure 4. If the mutation point belonged to SgSo, the offspring generated by the mutation may be illegal coding, which should be repaired.





Step 5. Termination. When the number of iterations reaches the maximum, the best individual will be chosen as the result of the algorithm.

3.2. Sectional Encoding Discrete Particle Swarm Optimization

Step 1. Initializing parameters. The main content of it is setting the parameters in SE-DPSO, which include the number of the swarm (*NS*), the learning factor 1 (*LF*₁), the learning factor 2 (*LF*₂), the dimension of a particle (*DP*), the maximum number of iterations (*NC*_*P*_{max}), the inertia weight (ω), the related basic data and so on. The encoded description method is present in part of the SE-DPSO of Table 1. The 2nd column shows the dimension number of SgSo of a particle (*P*_r), and the column 'dimension code (dc)', the column 'SgOm' and the column 'SgAt' show the dimension codes of SgSo, SgOm, SgAt of a *P*_r respectively, which will be discussed later.

Step 2. Initializing swarm. A group of an initial swarm was created in this step. The initial position (*p*) and initial velocity (*v*) of a particle were generated by:

$$p^0_{\alpha\beta} = rand(0,1) \tag{28}$$

$$v^0_{\alpha\beta} = rand(0,1). \tag{29}$$

Here, α , β are the index of a particle and the index of a dimension in the particle.

Step 3. Particle encoding. Considering the discreteness of the dimension codes of a particle in this model, the smallest position value (SPV) rule and a discrete interval (DI) rule were applied to transfer the position of a particle to a set of dimension codes. The representations of the codes in SE-DPSO were the same as the codes in SE-GA. Two sub-steps for encoding a particle are as follows:

Applying SPV and DI rule. The SPV rule was used to transform the continuous codes in PSO to discrete codes that are applicable to all types of scheduling problems [15], so as the DI rule. The SPV rule was applied in SgSo while the DI rule was applied in the SgOm and SgAt of a particle.

Define the sequence of operations, operation-machine and AGV-task relationship. An example with two jobs, three machines, and six available AGVs is shown in Table 2 for the explanation; both of the jobs include three operations. The assumed position of a particle for this example is shown in the 1st, 6th and 8th rows of Table 2. The definition method of the dimension codes of SgSo, SgOm, SgAt in a particle will be explained as follows respectively.

	Particle example	0.21	0.32	0.43	0.18	0.66	0.89
	Applying SPV rule	2	3	4	1	5	6
SgSo	Job codes	1	1	2	1	2	2
-	Corresponding operations in each job	<i>O</i> ₁ (1)	<i>O</i> ₁ (2)	$O_2(1)$	<i>O</i> ₁ (3)	<i>O</i> ₂ (2)	<i>O</i> ₂ (3)
		0.1	0.3	0.5	0.7	0.3	0.7
SaOm	Corresponding machines	M_1, M_2, M_3	M ₂ , M ₃	M ₂	M ₁ , M ₃	M ₂ , M ₃	M ₁ , M ₂
5gOm	Applying DI rule (Index of machines)	1	2	2	3	2	2
	Corresponding operations in each job	<i>O</i> ₁ (1)	<i>O</i> ₁ (2)	<i>O</i> ₁ (3)	<i>O</i> ₂ (1)	<i>O</i> ₂ (2)	<i>O</i> ₂ (3)
		0.93	0.21	0.49	0.37	0.86	0.18
SgAt	Applying DI rule (Index of AGVs)	6	2	3	3	6	2
(Corresponding operations in each job	<i>O</i> ₁ (1)	<i>O</i> ₁ (2)	<i>O</i> ₁ (3)	<i>O</i> ₂ (1)	<i>O</i> ₂ (2)	$O_2(3)$

Table 2. Encoding of an example particle.

As shown in the 1st and 2nd rows of Table 2, the continuous codes of SgSo in the particle are (0.21, 0.32, 0.43, 0.18, 0.66, 0.89). According to the SPV rule, the corresponding discrete codes should be (2, 3, 4, 1, 5, 6). In descending order, 0.18 is the smallest value and its discrete code should be 1; 0.89 is the largest value and its discrete code should be 6 and so on. The 3rd row of Table 2 shows the job codes of SgSo in the particle; the smallest three numbers of the discrete codes in the 2nd row of Table 2 are assigned to the first job, so the job code should be 1. The largest three numbers are assigned to the second job, so the job code should be 2.

As shown in the 5th and 7th rows of Table 2, the position of SgOm in the particle is $SgOm_P = (0.1, 0.3, 0.7, 0.2, 0.3, 0.7)$. According to the DI rule, the corresponding discrete codes should be $SgOm_C = (1, 2, 3, 2, 2, 2)$. As shown in the 6th and 8th rows of Table 2, $O_1(1)$ can be executed by M_1 , M_2 and M_3 . As a result, if $SgOm_P(1)$ is between 0 and 0.33, then $SgOm_C(1)$ should be 1; if $SgOm_P(1)$ is between 0.34 and 0.66, then $SgOm_C(1)$ should be 2 and so on. In the example, $SgOm_P(1)$ is 0.1, which is between 0 and 0.33, so $SgOm_C(1)$ should be 1. $O_1(2)$ can be executed by M_2 and M_3 . As a result, if $SgOm_P(2)$ is between 0 and 0.50, then $SgOm_C(2)$ should be 2, otherwise, $SgOm_C(2)$ should be 3. In the example, $SgOm_P(2)$ is 0.3, which is between 0 and 0.50, so $SgOm_C(2)$ should be 2. $O_1(3)$ can only be executed by M_2 , so $SgOm_C(3)$ should be 2.

As shown in the 9th and 10th rows of Table 2, the position of SgAt in the particle is (0.93, 0.21, 0.49, 0.37, 0.86, 0.18), according to the DI rule, the corresponding discrete codes are (6 2 3 3 6 2). Here, the number of available AGVs is six and the value 0.93 is between 5/6 and 6/6. Thus, the corresponding code should be 6; the value 0.21 is between 1/6 and 2/6, hence the corresponding code should be 2, and so on.

Step 4. Fitness evaluation. Each particle is evaluated by the makespan, total energy consumption and number of AGVs by Equations (1) to (17). In addition, the total fitness values will be calculated based on Equations (20) to (22).

Step 5. New swarm. The position and velocity of the particles were updated to generate a new swarm. Step 3, step 4 and step 5 were repeated up until the termination criterion.

The velocity of each particle was updated by:

$$v_{\alpha\beta}^{t+1} = \omega v_{\alpha\beta}^{t} + c_1 r_1 (p_{\alpha\beta}^{best} - p_{\alpha\beta}^{t}) + c_2 r_2 (g_{\beta}^{best} - p_{\alpha\beta}^{t}).$$
(30)

Here, c_1 is self-confidence and c_2 is swarm confidence. ω is the inertia weight, which is varying with a random method as shown in Equation (31).

$$\begin{cases} \omega = \mu + \sigma * N(0, 1) \\ \mu = \omega_{min} + (\omega_{max} - \omega_{min}) * rand(0, 1) \end{cases}$$
(31)

The position of the particle was updated by:

$$p_{\alpha\beta}^{t+1} = p_{\alpha\beta}^t + v_{\alpha\beta}^{t+1}.$$
(32)

Step 6. Termination. The iteration was terminated when the number of iterations reached its maximum, then the g^{best} was returned as the best solution.

4. Simulation Experiments and Discussion

4.1. Initial Data

To demonstrate the effectiveness of the proposed model, a numerical experiment was used in this paper. This experiment included eight jobs (J_1 , ..., J_8) with 10 machines (M_1 , ..., M_{10}), and there were three to five operations in each job. In the experiment, up to six AGVs were available. Table 3 shows the associated data between the operations and machines. Table 4 shows the AGV travel time among H points and the machines.

$I_{ob}(I)$		N	I _k / ot(i,j)/ ec(i,j)/ wecu	(k)	
J00(j _i)	<i>O_i</i> (1)	<i>O_i</i> (2)	<i>O</i> _{<i>i</i>} (3)	<i>O</i> _{<i>i</i>} (4)	<i>O_i</i> (5)
J_1	$\begin{array}{c} M_1/28/3.22/0.012 \\ M_4/25/2.91/0.012 \\ M_6/31/3.84/0.017 \end{array}$	M ₂ /21/3.82/0.015 M ₈ /23/3.15/0.014	$\begin{array}{c} M_5/22/2.67/0.011 \\ M_7/25/3.08/0.016 \\ M_{10}/27/2.72/0.021 \end{array}$		
J ₂	M ₃ /12/2.15/0.020 M ₉ /16/2.23/0.019	M ₂ /24/2.53/0.015	M ₁ /11/1.79/0.012 M ₈ /15/1.93/0.014	M ₁₀ /16/2.37/0.021	M ₅ /18/2.44/0.011 M ₇ /13/2.21/0.016
J ₃	M ₉ /17/3.22/0.019	$\begin{array}{c} M_2/18/2.78/0.015 \\ M_5/23/3.62/0.011 \\ M_7/21/2.50/0.016 \end{array}$	M ₄ /19/3.14/0.012 M ₈ /15/2.93/0.014	M ₁ /13/2.34/0.012 M ₆ /16/2.45/0.017	
J_4	M ₂ /22/3.25/0.015 M ₈ /27/3.23/0.014	M ₁ /19/3.33/0.012 M ₅ /21/3.32/0.011	M ₃ /23/3.67/0.020 M ₁₀ /21/3.52/0.021	M ₆ /28/4.13/0.017 M ₇ /33/4.21/0.016	
J ₅	M ₅ /31/3.67/0.011 M ₁₀ /27/3.55/0.021	$\begin{array}{c} M_3/19/2.21/0.020\\ M_4/24/2.32/0.012\\ M_8/22/2.24/0.014 \end{array}$	$\begin{array}{c} M_1/14/2.54/0.012\\ M_2/13/2.56/0.015\\ M_9/16/2.63/0.019 \end{array}$		
J ₆	M ₄ /25/3.11/0.012 M ₇ /27/3.23/0.016	M ₃ /30/4.11/0.020	M ₁ /21/3.76/0.012 M ₅ /18/3.68/0.011	M ₂ /12/1.97/0.015 M ₉ /15/1.91/0.019	M ₁₀ /27/3.41/0.021
J7	M ₃ /15/2.23/0.020 M ₆ /19/2.27/0.017	M ₄ /30/3.95/0.012 M ₇ /33/3.87/0.016	M ₁ /16/2.08/0.012 M ₈ /17/2.11/0.014		
J ₈	M ₅ /23/3.54/0.011 M ₁₀ /21/3.42/0.021	M ₃ /25/3.73/0.020 M ₆ /22/3.82/0.017	M ₂ /27/4.21/0.015 M ₄ /32/4.33/0.012		

Table 3. Basic information about the relationship between operations and machines.

Time(min)	Н	M ₁	M ₂	M ₃	M ₄	M ₅	M ₆	M ₇	M ₈	M9	M ₁₀
Н	0	7	11	16	12	10	20	8	19	18	23
M_1	7	0	6	16	21	14	6	7	13	16	20
M ₂	11	6	0	21	9	7	11	18	23	15	22
M3	16	16	21	0	14	6	7	10	8	14	11
M_4	12	21	9	14	0	11	17	13	22	5	19
M_5	10	14	7	6	11	0	7	26	19	18	11
M_6	20	6	11	7	17	7	0	4	13	17	18
M_7	8	7	18	10	13	26	4	0	15	20	24
M_8	19	13	23	8	22	19	13	15	0	13	19
M9	18	16	15	14	5	18	17	20	13	0	18
M10	23	20	22	11	19	11	18	24	19	18	0

Table 4. AGV traveling time among H points and machines.

According to the data in Table 3 and Equation (17), the value of *max(ese)* in Equation (21) was calculated by:

$$max(ese) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} max(es(i,j)).$$
(33)

Therefore, the value of *max(ese)* is 94.25. Figure 5 shows the Gantt chart before optimization by a random execution sequence and traveling sequence with six AGVs for the experiment, where *Ai* represents AGV *i*. The nearest bars on the left side of each operation indicate the time that the AGV transports the job from the previous operation to that operation, where the first bar and second bar indicate the running time of the AGV with no-load and load respectively. The values of fitness, number of AGVs, makespan, and energy consumption are 6, 420.67, 433 and 134.51 respectively. It is just an available random scheme for this experiment and all of the six AGVs were applied; it is difficult to be chosen as a high-quality solution. To optimize the solution for this experiment in FMS, the EAs proposed in this paper were applied and verified in the next section.



Figure 5. Gantt chart of a random sequence of the experiment using six automated guided vehicles (AGVs) before optimization.

4.2. Parameters of the Algorithms Setting

The selection of parameters has a significant impact on the performance of the algorithms. To get the best settings and parameters of the H-SE-GA-DPSO that can provide the best performance within

a reasonable computation time, a parameter test experiment, based on an orthogonal design method which is a low cost and effective parameter test method, was performed with the data as shown in Section 4.1. In the experiment, four levels and eight factors, as shown in Table 5, were applied. The experimental results are shown in Table 6. Here, each group of data was run 20 times with 500 iterations and the average value was taken as the result.

T1-				Fa	actors			
Levels	PS	CR	MR	<i>c</i> ₁	<i>c</i> ₂	w_{min}	w _{max}	σ
1	100	0.2	0.05	0.01	0.3	0.01	0.3	0.2
2	150	0.4	0.08	0.05	0.5	0.05	0.5	0.4
3 4	200 300	0.6 0.8	0.1	0.1	0.7	0.1	0.7	0.6 0.8

Table 5. Factors and their levels.

Table 6. Experimental results of orthogonal test of the parameters of hybrid sectional encoding genetic algorithm and discrete particle swarm optimization (H-SE-GA-DPSO).

Index	PS	CR	MR	<i>c</i> ₁	<i>c</i> ₂	w_{min}	w _{max}	σ	Fitnesses	Time
1	100	0.2	0.05	0.01	0.3	0.01	0.3	0.2	274.93	84.67
2	100	0.4	0.08	0.05	0.5	0.05	0.5	0.4	273.90	85.12
3	100	0.6	0.1	0.1	0.7	0.1	0.7	0.6	271.24	85.18
4	100	0.8	0.2	0.2	0.9	0.2	0.9	0.8	299.52	85.37
5	150	0.2	0.05	0.05	0.5	0.1	0.7	0.8	276.37	120.07
6	150	0.4	0.08	0.01	0.3	0.2	0.9	0.6	272.72	117.92
7	150	0.6	0.1	0.2	0.9	0.01	0.3	0.4	273.69	119.26
8	150	0.8	0.2	0.1	0.7	0.05	0.5	0.2	272.06	124.39
9	200	0.2	0.08	0.1	0.9	0.01	0.5	0.6	272.02	168.32
10	200	0.4	0.05	0.2	0.7	0.05	0.3	0.8	273.08	168.21
11	200	0.6	0.2	0.01	0.5	0.1	0.9	0.2	268.01	169.05
12	200	0.8	0.1	0.05	0.3	0.2	0.7	0.4	271.51	168.23
13	300	0.2	0.08	0.2	0.7	0.1	0.9	0.4	275.29	247.08
14	300	0.4	0.05	0.1	0.9	0.2	0.7	0.2	263.67	249.13
15	300	0.6	0.2	0.05	0.3	0.01	0.5	0.8	270.88	251.51
16	300	0.8	0.1	0.01	0.5	0.05	0.3	0.6	269.30	249.74
17	100	0.2	0.2	0.01	0.9	0.05	0.7	0.4	276.15	82.77
18	100	0.4	0.1	0.05	0.7	0.01	0.9	0.2	275.87	96.53
19	100	0.6	0.08	0.1	0.5	0.2	0.3	0.8	270.46	92.71
20	100	0.8	0.05	0.2	0.3	0.1	0.5	0.6	272.73	93.06
21	150	0.2	0.2	0.05	0.7	0.2	0.3	0.6	267.76	131.14
22	150	0.4	0.1	0.01	0.9	0.1	0.5	0.8	285.97	125.89
23	150	0.6	0.08	0.2	0.3	0.05	0.7	0.2	277.42	146.37
24	150	0.8	0.05	0.1	0.5	0.01	0.9	0.4	271.09	151.61
25	200	0.2	0.1	0.1	0.3	0.05	0.9	0.8	276.93	174.32
26	200	0.4	0.2	0.2	0.5	0.01	0.7	0.6	273.46	169.82
27	200	0.6	0.05	0.01	0.7	0.2	0.5	0.4	272.31	168.94
28	200	0.8	0.08	0.05	0.9	0.1	0.3	0.2	277.92	170.27
29	300	0.2	0.1	0.2	0.5	0.2	0.5	0.2	280.28	259.37
30	300	0.4	0.2	0.1	0.3	0.1	0.3	0.4	272.58	246.45
31	300	0.6	0.05	0.05	0.9	0.05	0.9	0.6	274.05	247.34
32	300	0.8	0.08	0.01	0.7	0.01	0.7	0.8	277.16	252.66
33	100	0.2	0.08	0.01	0.9	0.01	0.5	0.6	272.94	95.31

As shown in Table 6, it can be observed that the data in group 21 were the best configuration of parameters, which provide the optimal solution within a reasonable computation time. According to the data in Table 5, it can be observed that a larger population leads to a longer computation time. Besides, compared with other groups of data, the values of *CR* and w_{max} of group 21 were smaller

while the values of *MR* and w_{min} were larger; they and the values of the other parameters affect the performance of the algorithm together. Therefore, to obtain optimal solutions and to ensure the effective comparison of the algorithms within a reasonable computation time, the parameters setting of each algorithm are shown in Table 7.

Algorithms	Parameters
SE-GA	PS = 150, CR = 0.2, MR = 0.2
SE-DPSO	$PS = 150, c_1 = 0.05, c_2 = 0.7, w_{min} = 0.2, w_{max} = 0.3, \sigma = 0.6$
H-SE-GA-DPSO	$PS = 150, CR = 0.2, MR = 0.2, c_1 = 0.05, c_2 = 0.7, w_{min} = 0.2,$
	$w_{max} = 0.5, v = 0.0$ $PS = 150 CR = 0.2 MR = 0.2 c_1 = 0.05 c_2 = 0.7 v_1 = -0.2$
H-GA-PSO	$w_{max} = 0.3, \sigma = 0.6$
TSA	Based on the related literature [9]

4.3. Experiment Results and Discussion

The parameters for the hardware and software platform are list as follows: Windows 10, IntelIntel[®]CoreTMi7-4700MQ CPU, 2.40 GHz, 8 GB of RAM and the MATLAB 2016a (MathWorks, Natick City, U.S.).

To verify the effectiveness of the algorithms proposed in this paper and the advantages of H-SE-GA-DPSO over other algorithms, we performed a comparative experiment among the algorithms. All the algorithms were run 20 times with 500 iterations by considering the complexity of the problem.

In this experiment, the value of *DT* was 400. ψ_1 and ψ_2 were calculated based on Equations (21) and (22) respectively. The values of φ_1 , φ_2 and φ_3 were presumed to be 0.5, 0.2 and 0.3 respectively. In Table 8, the fitness values were calculated based on Equation (20).

Algorithms	Average Values of Fitness	Average Values of Makespan	Average Values of Energy Consumption	Average Numbers of AGVs	Values of Mean Computational Time
SE-GA	286.76	241.40	109.05	4.90	82.33
SE-DPSO	311.43	263.45	111.87	5.65	96.16
H-SE-GA-DPSO	267.76	211.85	104.95	4.85	131.14
H-GA-PSO	310.13	253.00	111.19	5.95	137.86
TSA	272.45	216.95	106.60	4.90	148.69

Table 8. The best optimization results of the five algorithms.

The best optimization results of the five algorithms are shown in Table 8, and the results are the average values after the algorithms being run 20 times. Compared with the result before optimization, as shown in Figure 5, the algorithms were proven to be effective in optimizing FMS scheduling with AGVs, and the number of AGVs decreases after optimization with SE-GA, H-SE-GA-DPSO and TSA. Compared with SE-GA and SE-DPSO, H-SE-GA-DPSO can get a better solution on all the three objectives even though they are all trying to solve the problem proposed on the three dimensions. Therefore, the performance of the hybrid algorithm was improved. Compared with H-GA-PSO and TSA, H-SE-GA-DPSO also can get a better comprehensive solution, although TSA can get a good result, the longer computational time discounted it; besides, compared with H-SE-GA-DPSO, the energy consumption of the result obtained by TSA was higher for ignoring the optimization of machine selection. Therefore, H-SE-GA-DPSO was verified to be superior to the other algorithms.

The 20 groups of data corresponding to the results of 20 runs of each algorithm are shown in Table 9. It can be observed that the optimal number of AGVs optimized with H-SE-GA-DPSO in this experiment can be four or five. Besides, according to the results of each algorithm in Table 9, compared

with the other algorithms, the results of H-SE-GA-DPSO had a smaller fluctuation range. As a result, the H-SE-GA-DPSO is more stable for optimizing fitness and minimizing the number of AGVs than the other algorithms.

Crown	SE-GA		SE-DI	PSO	H-SE-GA	-DPSO	H-GA-	PSO	TSA	
Numbers	Fitness Values	NA	Fitness Values	NA	Fitness Values	NA	Fitness Values	NA	Fitness Values	NA
1	283.97	5	302.61	6	267.02	5	311.34	6	268.14	4
2	288.22	5	303.77	6	272.47	5	303.71	6	268.05	4
3	280.47	5	297.10	6	264.84	5	319.92	6	257.24	5
4	290.13	5	314.55	6	272.16	5	298.77	6	274.32	6
5	286.47	5	304.89	6	262.00	5	308.00	6	266.79	5
6	286.60	4	318.13	5	270.36	5	297.91	6	272.21	5
7	288.38	5	305.45	5	271.25	5	318.64	6	259.46	5
8	297.83	6	317.55	6	260.91	5	312.78	6	276.08	5
9	303.52	5	325.18	6	271.62	5	311.86	6	262.57	5
10	279.32	5	316.09	5	264.26	4	309.95	6	270.11	4
11	283.85	5	299.63	6	274.43	5	303.21	6	276.49	4
12	277.01	5	322.33	6	261.51	4	309.85	6	275.45	6
13	283.42	4	291.04	6	269.45	5	310.76	6	267.69	4
14	290.57	6	316.45	6	258.78	5	315.64	6	281.90	5
15	279.99	4	309.10	6	266.22	5	315.64	6	281.97	6
16	290.95	6	342.85	4	270.40	5	315.07	6	281.75	5
17	280.45	5	307.36	5	276.77	4	316.68	6	268.66	4
18	289.96	5	314.90	5	262.12	5	310.88	5	266.19	5
19	277.39	4	313.54	6	268.98	5	302.03	6	296.97	6
20	296.76	4	306.07	6	269.60	5	309.90	6	277.03	5

Table 9. The 20 groups of data corresponding to the results of 20 runs of each algorithm.

As shown in Table 10, the experiments based on the data in Section 4.1 with different numbers of available AGVs were performed and the results are presented in the table. Here, each experiment was run 20 times using the H-SE-GA-DPSO. In Table 10, the numbers in parentheses indicate the frequencies that those numbers themselves appeared in the corresponding experiment. According to the results, four and five were further proven to be the optimal numbers of AGVs to meet the needs of the jobs in Section 4.1. As the number of available AGVs increases, six and seven were also selected as the optimal numbers of AGVs, which led to the increase of the values of fitness. Therefore, the results of H-SE-GA-DPSO can be influenced by the number of available AGVs to a certain extent. However, the H-SE-GA-DPSO was also proven to be stable and effective over a wide range.

Table 10. The experiments with different numbers of available AGVs using H-SE-GA-DPSO.

Data items	Numbers of Available AGVs									
Dutu items	6	7	8	9	10					
Average values of fitness	267.76	268.11	270.20	273.45	276.03					
Average numbers of AGV	4.85	4.95	5.15	5.20	5.40					
Number of AGVs	4(3), 5(17)	4(2), 5(17), 6(1)	4(1), 5(15), 6(4)	5(16), 6(4)	4(2), 5(10), 6(6), 7(2)					

Figure 6 shows the performance of SE-GA, SE-DPSO, H-SE-GA-DPSO, H-GA-PSO and TSA in the experiment above with six available AGVs. SE-DPSO was easy to fall into local convergence in this experiment and H-GA-PSO just took the sequence of operations into account; the convergence of the results was not satisfactory. Although SE-GA had a good convergence effect, its convergence rate was slower than that of H-SE-GA-DPSO and TSA. Compared with the other algorithms, H-SE-GA-DPSO and TSA had a good convergence rate and result, and the result of H-SE-GA-DPSO was slightly better than that of TSA, which can be attributed to the consideration of machine selection in H-SE-GA-DPSO.



Figure 6. Evolutionary curves of the best fitness using the five algorithms with six available AGVs.

The evolutionary curves of the makespan of the best fitness with six available AGVs are shown in Figure 7. The convergence trends of them are roughly the same as the curves in Figure 6; H-SE-GA-DPSO and TSA had a better optimization effect on the makespan.



Figure 7. Evolutionary curves of the makespan of the best fitness using the five algorithms with six available AGVs.

Figure 8 shows the evolutionary curves of energy consumption of the best fitness with six available AGVs. It can be easily observed that the values of energy consumption converge quickly and fluctuate within a certain range. In this aspect, the convergence result of H-SE-GA-DPSO was obviously better than the TSA. Therefore, it is significant to consider machine selection on AGVs scheduling in FMS.



Figure 8. Evolutionary curves of energy consumption of the best fitness using the five algorithms with six available AGVs.

Figure 9 shows the evolutionary curves of a number of AGVs of the best fitness with six available AGVs. The evolutionary process of them was relatively simple. The number of AGVs was minimized in the results obtained by H-SE-GA-DPSO and TSA. The numbers of AGVs obtained by other algorithms was influenced by the initial population to some extent.



Figure 9. Evolutionary curves of the number of AGVs of the best fitness using the five algorithms with six available AGVs.

Figure 10 shows the optimization sequence using only four AGVs, which was obtained by H-SE-GA-DPSO. Compared with the sequence in Figure 5, all jobs were completed in less time, although the number of AGVs was less. In Figures 6 and 8, the fitness and energy consumption were

also less than that of the sequence in Figure 5. Overall, the application of H-SE-GA-DPSO to solve a multi-objective and multi-dimensional scheduling problem is concluded to be an effective method.



Figure 10. Gantt chart of the experiment after optimization by H-SE-GA-DPSO.

To further verify the conclusion, a significance test of the results of the algorithms is provided. According to the data of the fitness values in Table 9, we can get the data as shown in Table 11 using the Shapiro-Wilk test and the corresponding box-plot is shown in Figure 11. In this test, the significance level was α =0.05 and all of the *p*-values were larger than α . Therefore, the data obtained by the five algorithms conform to a normal distribution. To analyze statistically significant differences in the data, The tukey HSD method was applied in this paper. Table 12 shows the results of the test of homogeneity of variances. Here, the significance level was also 0.05, so the variances don't satisfy the condition of homogeneity. The data of robust tests of equality of means obtained by the methods of Welch and Brown-Forsythe show that the *p*-values were 0.00, which is smaller than 0.05. Therefore, statistically significant differences among the fitness values were obtained with different algorithms in this paper. In summary, the advantages of H-SE-GA-DPSO over other algorithms were further verified.

Table 11. The results of the normal distribution test using the Shapiro-Wilk method.

Algorithms	SE-GA	SE-DPSO	H-SE-GA-DPSO	H-GA-PSO	TSA
Significance (p-value)	0.319	0.388	0.599	0.335	0.365



Figure 11. The box-plot of fitness values obtained by the five algorithms.

Table 12. The results of the test of homogeneity of variances.

Levene Statistic	degree of Freedom 1 (df1)	degree of Freedom 2 (df2)	p-Value
2.743	4	95	0.033

5. Conclusions and Future Work

Aimed at the AGV optimization scheduling problem in an FMS environment, this research developed a multi-objective and multi-dimensional optimization scheduling mathematical model while considering energy consumption and multi-function of the machines. In this model, the multi-objective was to minimize the makespan and energy consumption of machines and the number of AGVs. The multi-dimensional objective was to simultaneously optimize the sequence of operations of related jobs, the matching relation between transfer tasks and AGVs (AGV-task) and the matching relation between operations and machines (operation-machine) for the multi-objective.

To meet the needs of the above model, three evolutionary algorithms (SE-GA, SE-DPSO and H-SE-GA-DPSO) have been developed to realize multi-objective and multi-dimensional optimization scheduling, which have been proven to be effective. According to the results of the comparison, the superiority of H-SE-GA-DPSO over the other algorithms was proven. Overall, H-SE-GA-DPSO is a good optimization method in multi-objective and multi-dimensional scheduling of FMS with AGVs, and it can also be applied to more situations with the development of intelligent manufacturing and green manufacturing.

In the future, it will be interesting to investigate the following issues:

The model proposed in this paper should be stretched from a workshop level to an enterprise level and more factors should be taken into account for deeper optimization. H-SE-GA-DPSO should be applied in a distributed computing environment to meet more complex computing needs.

Author Contributions: Conceptualization, S.G.; Methodology, W.X. and S.G.; Software, W.X.; Validation, W.X. and S.G.; Formal analysis, W.X. and S.G.; Investigation, W.X.; Resources, S.G.; Data curation, W.X.

Funding: This work is supported by the Science and Technology Supporting Plan of Hubei, China (No.2015BAA063), and the Fundamental Research Funds for the Central Universities, China (No.2016-YB-020), and the National key R & D project, China (2017YFB0404204).

Acknowledgments: The authors would like to express their great appreciation for the valuable comments and constructive suggestions by the anonymous reviewers and the editor.

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

Appendix A

i	Index of jobs
j	Index of operations in a job
k	Index of machines
1	Index of AGVs
п	Number of jobs
m _i	Number of operations for Job number <i>i</i>
S	Number of machines
NA	Number of AGVs
Ji	Job number i
$O_i(j)$	Operation number <i>j</i> of job <i>i</i>
M_k	Machine number k
$M_i(j)$	Assigned machine for $O_i(j)$
A_l	AGV number <i>l</i>
$A_i(j)$	Assigned AGV for traveling task of $O_i(j)$
Н	Home of AGVs
N_k	Number of operations assigned to M_k
р	Index of operations assigned to M_k
$T_s(k,p)$	Start time of executing operation number p assigned to M_k
$T_e(k,p)$	End time of executing operation number p assigned to M_k
T(k,p)	Executing time of operation number p assigned to M_k
WT(k,p)	Waiting time before executing operation number p and after executing operation number p -1 assigned to M_k
es(k,p)	Energy consumption of operation number p assigned to M_k
wesu(k)	Standby energy consumption of M_k in unit time
N _l	Number of traveling tasks assigned to A_1
q	Index of traveling tasks assigned to A_1
$T_s(l,q)$	Start time of executing task number q assigned to A_1
$T_e(l,q)$	End time of executing task number q assigned to A_1
<i>T(l,q)</i>	Traveling time of task number q assigned to A_1

Table A1. Related parameters and variables.

WT (l,q)	Waiting time before executing the traveling task number q and after executing the traveling task number q -1 assigned to A_1
ot _s (i,j)	Start time of executing $O_i(j)$
ot _e (i,j)	End time of executing $O_i(j)$
ot(i,j)	Executing time of $O_i(j)$
otw(i,j)	Waiting time before executing operation number j and after executing operation number j -1
ot(i)	Finish time of operations of J_i
TA _{ij}	Related traveling task to $O_i(j)$ (Moving from the previous point of AGV to $M_i(j-1)$ and then to $M_i(j)$ or H to $M_i(j)$)
$tt_s(i,j)$	Start traveling time of executing TA_{ij}
$tt_e(i,j)$	End traveling time of executing TA_{ij}
tt(i,j)	Traveling time of executing <i>TA_{ij}</i>
ttw(i,j)	Waiting time before executing TA_{ij} and after executing $TA_{i(j-1)}$
tt(i)	Finish time of traveling of J_i
es(i,j)	Energy consumption of executing $O_i(j)$
esw(k)	Standby energy consumption of <i>M</i> _k
ese(k)	Energy consumption of executing operations of M_k
esw	Standby energy consumption
ese	Energy consumption of executing operations
ES	Energy consumption of all jobs
AFT_l	Finish time of traveling tasks assigned to A_1
MFT_k	Finish time of operations assigned to M_k
MS, DT	Makespan, Time of delivery

Table A1. Cont.

References

- 1. Guo, Q.; Zhang, M. A novel approach for multi-agent-based Intelligent Manufacturing System. *Inf. Sci.* 2009, 179, 3079–3090. [CrossRef]
- 2. Kang, H.S.; Ju, Y.L.; Choi, S.S.; Kim, H.; Park, J.; Son, J.; Kim, B.; Noh, S. Smart manufacturing: Past research, present findings, and future directions. *Int. J. Precis. Eng. Manuf. Green Technol.* **2016**, *3*, 111–128. [CrossRef]
- 3. Huang, S.; Guo, Y.; Zha, S.; Wang, F.; Fang, W. A Real-time Location System Based on RFID and UWB for Digital Manufacturing Workshop. *Procedia CIRP* **2017**, *63*, 132–137. [CrossRef]
- 4. Confessore, G.; Fabiano, M.; Liotta, G. A network flow based heuristic approach for optimising AGV movements. *J. Intell. Manuf.* **2013**, *24*, 405–419. [CrossRef]
- 5. Srivastava, S.C.; Choudhary, A.K.; Kumar, S.; Tiwari, M.K. Development of an intelligent agent-based AGV controller for a flexible manufacturing system. *Int. J. Adv. Manuf. Technol.* **2008**, *36*, 780. [CrossRef]
- 6. Singh, N.; Sarngadharan, P.V.; Pal, P.K. AGV scheduling for automated material distribution: A case study. *J. Intell. Manuf.* **2011**, *22*, 219–228. [CrossRef]
- Ramana, B.; Reddy, S.S.; Ramprasad, B. Quantitative Analysis of AGV System in FMS Cell Layout. *Def. Sci.* J. 2013, 47, 75–81. [CrossRef]
- Murata, T. Makespan Minimization of Machines and Automated Guided Vehicles Schedule Using Binary Particle Swarm Optimization. In Proceedings of the Proceedings of the International Multiconference of Engineers and Computer Scientists, Hong Kong, China, 17–19 March 2010; pp. 268–272.
- 9. Zheng, Y.; Xiao, Y.; Seo, Y. A tabu search algorithm for simultaneous machine/AGV scheduling problem. *Int. J. Prod. Res.* **2014**, *52*, 5748–5763. [CrossRef]

- 10. Deroussi, L.; Gourgand, M.; Tchernev, N. A simple metaheuristic approach to the simultaneous scheduling of machines and automated guided vehicles. *Int. J. Prod. Res.* **2008**, *46*, 2143–2164. [CrossRef]
- 11. Lacomme, P.; Larabi, M.; Tchernev, N. Job-shop based framework for simultaneous scheduling of machines and automated guided vehicles. *Int. J. Prod. Econ.* **2013**, *143*, 24–34. [CrossRef]
- 12. Baruwa, O.T.; Piera, M.A. A coloured Petri net-based hybrid heuristic search approach to simultaneous scheduling of machines and automated guided vehicles. *Int. J. Prod. Res.* **2016**, *54*, 1–20. [CrossRef]
- 13. Udhayakumar, P.; Kumanan, S. Task scheduling of AGV in FMS using non-traditional optimization techniques. *Int. J. Simul. Model.* **2010**, *9*, 28–39. [CrossRef]
- 14. Pan, X.Y.; Wu, J.; Zhang, Q.W.; Lai, D.; Xie, H.; Zhang, Z. A Case Study of AGV Scheduling for Production Material Handling. *Appl. Mechan. Mater.* **2013**, *411–414*, 2351–2354. [CrossRef]
- Mousavi, M.; Yap, H.J.; Musa, S.N.; Tahriri, F.; Dawal, S.Z.M. Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization. *PLoS ONE* 2017, *12*, e0169817. [CrossRef] [PubMed]
- 16. Mousavi, M.; Yap, H.J.; Musa, S.N. A fuzzy hybrid GA-PSO algorithm for multi-objective AGV scheduling in FMS. *Int. J. Simul. Model.* **2017**, *16*, 58–71. [CrossRef]
- 17. Cai, Q.; Tang, D.; Zheng, K.; Zhu, H.; Wu, X.; Lu, X. Multi-AGV scheduling optimization based on: Neuro-endocrine coordination mechanism. *Int. J. Smart Sens. Intell. Syst.* **2013**, *7*, 1613–1630. [CrossRef]
- 18. Novas, J.M.; Henning, G.P. Integrated scheduling of resource-constrained flexible manufacturing systems using constraint programming. *Expert Syst. Appl.* **2014**, *41*, 2286–2299. [CrossRef]
- Mchaney, R. Modelling battery constraints in discrete event automated guided vehicle simulations. *Int. J. Prod. Res.* 2007, 33, 3023–3040. [CrossRef]
- 20. Kabir, Q.S.; Suzuki, Y. Increasing manufacturing flexibility through battery management of automated guided vehicles. *Comput. Ind. Eng.* **2018**, *117*, 225–236. [CrossRef]
- 21. Yan, R.; Jackson, L.M.; Dunnett, S.J. Automated guided vehicle mission reliability modelling using a combined fault tree and Petri net approach. *Int. J. Adv. Manuf. Technol.* **2017**, *92*, 1825–1837. [CrossRef]
- 22. Tao, F.; Hu, Y.; Zhou, Z. Correlation-aware resource service composition and optimal-selection in manufacturing grid. *Eur. J. Oper. Res.* 2010, 201, 129–143. [CrossRef]
- Guo, S.; Du, B.; Peng, Z.; Huang, X.; Li, Y. Manufacturing resource combinatorial optimization for large complex equipment in group manufacturing: A cluster-based genetic algorithm. *Mechatronics* 2015, 31, 101–115. [CrossRef]
- 24. Du, B.; Guo, S.; Huang, X.; Li, Y.; Guo, J. A Pareto supplier selection algorithm for minimum the life cycle cost of complex product system. *Expert Syst. Appl.* **2015**, *42*, 4253–4264. [CrossRef]
- 25. Wang, L.; Guo, S.; Li, X.; Du, B.; Xu, W. Distributed manufacturing resource selection strategy in cloud manufacturing. *Int. J. Adv. Manuf. Technol.* **2018**, *94*, 3375–3388. [CrossRef]
- 26. Udhayakumar, P.; Kumanan, S. Integrated scheduling of flexible manufacturing system using evolutionary algorithms. *Int. J. Adv. Manuf. Technol.* **2012**, *61*, 621–635. [CrossRef]
- 27. Caridá, V.F.; Morandin, O.; Tuma CC, M. Approaches of fuzzy systems applied to an AGV dispatching system in a FMS. *Int. J. Adv. Manuf. Technol.* **2015**, *79*, 615–625. [CrossRef]
- 28. Chang, H.C.; Liu, T.K. Optimisation of distributed manufacturing flexible job shop scheduling by using hybrid genetic algorithms. *J. Intell. Manuf.* **2017**, *28*, 1973–1986. [CrossRef]
- Rifai, A.P.; Nguyen, H.T.; Aoyama, H.; Dawal, S.Z.M.; Masruroh, N.A. Non-dominated sorting biogeography-based optimization for bi-objective reentrant flexible manufacturing system scheduling. *Appl. Soft Comput.* 2018. [CrossRef]
- 30. Proth, J.M.; Sauer, N.; Xie, X. Optimization of the number of transportation devices in a flexible manufacturing system using event graphs. *IEEE Trans. Ind. Electron.* **2006**, *44*, 298–306. [CrossRef]
- 31. Saidi-Mehrabad, M.; Dehnavi-Arani, S.; Evazabadian, F.; Mahmoodian, V. An Ant Colony Algorithm (ACA) for solving the new integrated model of job shop scheduling and conflict-free routing of AGVs. *Comput. Ind. Eng.* **2015**, *86*, 2–13. [CrossRef]
- 32. Pjevcevic, D.; Nikolic, M.; Vidic, N.; Vukadinovic, K. Data envelopment analysis of AGV fleet sizing at a port container terminal. *Int. J. Prod. Res.* 2017, *55*, 4021–4034. [CrossRef]
- 33. Vivaldini, K.; Rocha, L.F.; Martarelli, N.J.; Becker, M.; Moreira, A.P. Integrated tasks assignment and routing for the estimation of the optimal number of AGVS. *Int. J. Adv. Manuf. Technol.* **2016**, *82*, 719–736. [CrossRef]

- 34. Yan, H.E. Job scheduling model of machining system for green manufacturing. *Chin. J. Mech. Eng.* **2007**, *43*, 27–33.
- 35. Govindan, K.; Diabat, A.; Shankar, K.M. Analyzing the drivers of green manufacturing with fuzzy approach. *J. Clean. Prod.* **2015**, *96*, 182–193. [CrossRef]
- 36. Singh, A.; Philip, D.; Ramkumar, J.; Das, M. A simulation based approach to realize green factory from unit green manufacturing processes. *J. Clean. Prod.* **2018**, *182*, 67–81. [CrossRef]
- 37. Liang, Y.; Lin, L.; Gen, M.; Chien, C. A hybrid evolutionary algorithm for fms optimization with AGV dispatching. *Comput. Ind. Eng.* **2012**, *2*, 1115–1128.
- 38. Chen, R.M.; Shen, Y.M. Dynamic search control-based particle swarm optimization for project scheduling problems. *Adv. Mech. Eng.* **2016**, *8*, 1687814016641837. [CrossRef]
- 39. Lu, H.; Zhou, R.; Fei, Z.; Shi, J. A multi-objective evolutionary algorithm based on Pareto prediction for automatic test task scheduling problems. *Appl. Soft Comput.* **2018**. [CrossRef]
- Wang, Z.; Si, L.; Tan, C.; Liu, X. A Novel Approach for Shearer Cutting Load Identification through Integration of Improved Particle Swarm Optimization and Wavelet Neural Network. *Adv. Mech. Eng.* 2014, 2014, 521629. [CrossRef]
- 41. Giagkiozis, I.; Fleming, P.J. Methods for multi-objective optimization: An analysis. *Inf. Sci.* **2015**, *293*, 338–350. [CrossRef]
- 42. Han, D.; Yang, B.; Li, J.; Wang, J.; Sun, M.; Zhou, Q. A multi-agent-based system for two-stage scheduling problem of offshore project. *Adv. Mech. Eng.* **2017**, *9*, 1–17. [CrossRef]
- 43. Kim, J.W.; Sang, W.K. New Encoding/Converting Methods of Binary GA/Real-Coded GA. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2005**, *88*, 1554–1564. [CrossRef]
- 44. Yamamoto, H.; Qudeiri, J.A.; Yamada, T.; Rizauddin, R. Production layout design system by GA with one by one encoding method. *Artif. Life Robot.* **2008**, *13*, 234–237. [CrossRef]
- 45. Wang, H.F.; Hsu, H.W. A closed-loop logistic model with a spanning-tree based genetic algorithm. *Comput. Oper. Res.* **2010**, *37*, 376–389. [CrossRef]
- Zhang, Y.; Liu, M. An improved genetic algorithm encoded by adaptive degressive ary number. *Soft Comput.* 2018, 22, 6861–6875. [CrossRef]
- 47. Wu, J.; Zhang, C.H.; Cui, N.X. PSO algorithm-based parameter optimization for HEV powertrain and its control strategy. *Int. J. Automot. Technol.* **2008**, *9*, 53–59. [CrossRef]
- Ishaque, K.; Salam, Z.; Amjad, M.; Mekhilef, S. An Improved Particle Swarm Optimization (PSO)-Based MPPT for PV With Reduced Steady-State Oscillation. *IEEE Trans. Power Electron.* 2012, 27, 3627–3638. [CrossRef]
- 49. Tian, Y.; Liu, D.; Yuan, D.; Wang, K. A discrete PSO for two-stage assembly scheduling problem. *Int. J. Adv. Manuf. Technol.* **2013**, *66*, 481–499. [CrossRef]
- 50. Shi, X.H.; Liang, Y.C.; Leeb, H.P.; Lu, C.; Wang, L.M. An improved GA and a novel PSO-GA-based hybrid algorithm. *Inf. Process. Lett.* **2005**, *93*, 255–261. [CrossRef]
- Li, J.; Yang, B.; Zhang, D.; Zhou, Q.; Li, L. Development of a multi-objective scheduling system for offshore projects based on hybrid non-dominated sorting genetic algorithm. *Adv. Mech. Eng.* 2015, *7*, 1687814015573785.
 [CrossRef]
- 52. Su, Y.; Chu, X.; Zhang, Z.; Chen, D. Process planning optimization on turning machine tool using a hybrid genetic algorithm with local search approach. *Adv. Mech. Eng.* **2015**, *7*, 1687814015581241. [CrossRef]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).