

A NEW MODEL FOR OPTIMIZATION OF CELL SCHEDULING CONSIDERING INTER-CELL MOVEMENT

Huang, Z. & Yang, J. J.

Department of Industrial and Manufacturing Systems Engineering, Beihang University,
Beijing 100191, China

E-Mail: buaa1609@buaa.edu.cn, jjyang@buaa.edu.cn

Abstract

Focusing on a cellular manufacturing system with inter-cell operations, a new scheduling framework is proposed in which a novel mathematical model is provided considering the cell scheduling sequence and an extended disjunctive graph model is established with multi-cell. In addition, an improved harmony search algorithm is proposed, comparison experiments show that the proposed solution can obtain the best results in most cases. To enhance further the generalizability of our strategy, an adaptive neuro-fuzzy inference system (ANFIS) structure is generated. ANFIS is based on the improved harmony search algorithm and takes the cell scheduling properties as input and the results of our proposed solution as output. Then, the optimal ANFIS structure is obtained by comparing different ANFIS structures. Finally, the ANFIS-predicted values are compared with those of the proposed solution and get a high fitting degree, which verifies the validity of the ANFIS structure.

(Received in August 2021, accepted in January 2022. This paper was with the authors 1 month for 1 revision.)

Key Words: Inter-Cell Scheduling, Harmony Search, Cell Scheduling Sequence, Adaptive Neuro-Fuzzy Inference System, Extended Disjunctive Graph Model

1. INTRODUCTION

The proposal and implementation of Industry 4.0 are promoting the development of intelligent manufacturing, of which shop scheduling is an important part. Shop scheduling problem is a NP-hard problem. Since its emergence, scholars have done a lot of research on it, especially at the algorithm level. Zhang and Zhang [1] proposed a multi-level dynamic scheduling model to solve the job-shop scheduling problem considering various emergencies. Shen and Chen [2] put forward a dynamic hybrid ant colony optimization to solve blocking flow shop scheduling problems. Shi et al. [3] established a dynamic scheduling model for the flexible job-shop scheduling problem considering fuzzy delivery time. However, along with the changing needs of customers, cellular manufacturing (CM) has been widely used to produce small- and medium-batch products with high personalization and complexity, and the scheduling problem for manufacturing cells has been widely considered [4-7].

In order of importance, there are three major problems to be solved in a cellular manufacturing system (CMS): cell formation (CF), cell layout (CL) and cell scheduling (CS) [1]. The first two problems mainly concern the division of machines and jobs into cells and the layout of the machines in each cell, which specify the initial conditions needed to solve the CS. According to the results of the first two problems, each cell is usually designated as either a flow-line manufacturing cell or a job shop manufacturing cell, and the corresponding scheduling problems can also be divided into flow-line manufacturing cell scheduling and job shop manufacturing cell scheduling problems. The effective implementation of CS is the key step for realizing the benefits of a CMS.

From the perspective of different cell types, the existing literature can be mainly divided into two groups: work focusing on flow-line manufacturing cells and work focusing on job shop manufacturing cells.

Regarding the scheduling problem for flowline manufacturing cells, Lin and Ying [8] proposed an efficient metaheuristic for solving the no-wait flowline manufacturing cell

scheduling problem considering sequence-dependent family setup times with the aim of minimizing the makespan. A hybrid harmony search algorithm was put forward by Li et al. [9] to solve the scheduling problem of flowline manufacturing cells with the aim of minimizing the total tardiness and mean total flow time. More related articles can be seen in Neufeld et al. [10] who reviewed flowshop group scheduling and its solution method.

However, the scheduling constraints for flowline manufacturing cells mainly focus on their sequence-dependent setup times, in addition, there has been less research on multi-cell scheduling, and inter-cell operations have also often been ignored, although they will inevitably be encountered in a CMS environment. Moreover, heuristic algorithms are mainly used as the solution algorithms, while the applications of learning algorithms are rarely used.

In sharp contrast to the cell scheduling for flowline manufacturing cells, there have been only a few studies on cell scheduling for job shop manufacturing cells. Li et al. [11] considered multiple single-processing machines and one batch-processing machine in a CMS and proposed a combined ant colony optimization algorithm to solve the intercell scheduling problem. A new mathematical model was proposed by Deliktas et al. [12] to solve the flexible job shop scheduling problem in a CMS considering exceptional jobs, intercell movement, intercell transportation time, sequence-dependent setup times and re-entrant jobs. Elmi et al. [13] developed an integer linear programming model for scheduling problems in job shop cell manufacturing systems considering exceptional and re-entrant jobs. Zeng et al. [14] put forward an integer nonlinear programming (ILP) model to describe the blocking job shop scheduling problem with AGVs and solved it using a two-stage heuristic algorithm and a local search algorithm. At present, there are two representative solution strategies for inter cell scheduling problems. One is Solimanpur et al. [15], who divided the cell scheduling problem into intra cell scheduling and inter cell scheduling. However, this solution method is to conduct inter cell scheduling after intra cell scheduling, and the sequence of cells are only considered in inter cell scheduling. The other is Tang et al. [16] who developed a scatter search approach addressing the scheduling problem in a CMS considering exceptional parts, but it solves multiple cells problem as a whole workshop. This method does not take the characteristics of CMS into account. Moreover, when the number of cells increases, the number of machines and processes increases accordingly, the solution efficiency is not high.

It can be seen from the above discussion that there are several non-negligible deficiencies in most of the existing literature on cell scheduling. 1) A major drawback is the regardless of inter-cell movement. 2) There are no articles concerning hybrid cells in a CMS. 3) Heuristic algorithms are mainly used as the solution algorithms, instead of learning algorithms. However, machine learning algorithms tend to offer good generalizability. 4) The optimization of the overall CMS cannot guarantee the optimization of each cell, and the optimization of each cell also does not necessarily guarantee the overall optimization of the system.

In view of the above shortcomings, a mathematical model considering the sequence of the cells in the whole scheduling process is established here, and the operations in intra- and inter-cells are solved at the same time by means of a newly proposed improved harmony search (IHS) algorithm in a mixed-cell environment. The new solution method is named HWHS. In addition, to enhance its generalizability, an intelligent system is also developed based on the concept of an adaptive neuro-fuzzy inference system (ANFIS).

The structure of this paper is as follows: Section 2 describes the corresponding scheduling framework. Section 3 introduces the scheduling framework and solution method. In Section 4, the effectiveness of the proposed method is verified by comparison with two other methods on randomly generated data, and the optimal ANFIS structure is given. Finally, Section 5 is the conclusion.

2. PROBLEM DESCRIPTION

The mathematical model proposed in this paper considers the effect of the scheduling of the first cell on the results for cells that are scheduled later. This model can better capture the processing of inter-cell operations, more closely represent reality, and better highlight the characteristics of a CMS.

For convenience of description, the variable indices used in the formulas are listed in Table I. With the aim of minimizing the makespan, the objective function is defined in Eq. (1).

$$\text{Min max } (C_{kji}) \quad (1)$$

Table I: Indices of variables used in formulas.

Variable	Meaning
M, N, K	the numbers of machines, parts, and cells
m	the index for machines, $m = 1, \dots, M$
j	the index for jobs, $j = 1, \dots, N$
k	the index for cells, $k = 1, \dots, K$
i	the sequence number of an operation
O_j	the total number of operations for job j
o_{ji}	the index for operations, $o_{ji} = 1, \dots, O_j$
fo_{ji}	an inter-cell operation from a foreign cell to the current cell
$intrao_{ji}$	an intracell operation in the current cell
$intero_{ji}$	an inter-cell operation from the current cell to a foreign cell
$S_{o_{ji}}, S_{fo_{ji}}, S_{intrao_{ji}}, S_{intero_{ji}}$	the start times of operations $o_{ji}, fo_{ji}, intrao_{ji},$ and $intero_{ji}$
$E_{o_{ji}}, E_{fo_{ji}}, E_{intrao_{ji}}, E_{intero_{ji}}$	the end times of operations $o_{ji}, fo_{ji}, intrao_{ji},$ and $intero_{ji}$
$P_{ji}, P_{intero_{ji}}$	the processing times of operations o_{ji} and $intero_{ji}$
C_{kji}	the completion time of operation o_{ji} in cell k

In the proposed solution, the scheduling of a cell depends on the scheduling results for the previous cell. For the cell to be scheduled first, the corresponding time conditions are given in Eqs. (2) and (3).

$$S_{fo_{ji}} \geq \sum_{i'=0}^{i-1} P_{ji'} \quad (2)$$

This condition limits the start time of operation fo_{ji} (an operation from another cell to the current cell) by defining an earliest possible start time, that is, the sum of the processing times of all operations before this inter-cell operation.

$$S_{intrao_{ji+1}} - E_{intrao_{ji-1}} \geq P_{intero_{ji}} \quad (3)$$

This condition indicates that for operation $intero_{ji}$ (from the current cell to another cell), the time difference between the start time of the subsequent operation and the start time of the previous operation must be greater than or equal to the processing time of this inter-cell operation.

For a cell to be scheduled later, the corresponding time conditions are described by Eqs. (4) and (5).

$$E_{o_{ji-1}} \leq S_{fo_{ji}} \text{ and } S_{o_{ji+1}} \geq E_{fo_{ji}} \quad (4)$$

These conditions represent the possible time range of operation fo_{ji} , that is, between the end time of the previous operation and the start time of the subsequent operation.

$$E_{intrao_{ji-1}} \leq S_{intero_{ji}} \text{ and } S_{intrao_{ji+1}} \geq E_{intero_{ji}} \quad (5)$$

These conditions indicate that the latest end time of the previous operation is determined by the start time of $intero_{ji}$ and the earliest start time of the subsequent operation is limited by the end time of $intero_{ji}$.

For convenience of description, a typical CMS with three cells is considered. The processing times of the parts on the corresponding machines follow a uniform distribution on

the interval of [1, 10], as shown in Table II. There are 17 parts, P1-P17, and 13 machines, M1-M13. The parts and machines represented in the same colour belong to the same cell. The last row lists the cell to which each part belongs, and the last column lists the cell to which each machine belongs. For example, P1-P5 and M1-M4 are represented in red and belong to cell 1. Similarly, P6-P11 and M5-M8, in green, belong to cell 2, and P12-P17 and M9-M13, in blue, belong to cell 3. The sequences in which the parts pass through the machines are shown in Table III. Cell 1 is a flow-line manufacturing cell, and the other two are job shop manufacturing cells. In addition, there are inter-cell operations. The third operation on P2 in cell 1 needs to be processed on machine 12 in cell 3, as indicated in red; similarly, the third operation on P7 and the first operation on P15 are inter-cell operations, which need to be processed on machine 9 in cell 3 and machine 3 in cell 1, respectively.

Table II: Processing time of each part on each machine (s).

M	P																	Cell of M
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
1	9	2	10	2	10													1
2	7	7	8	8	8													1
3	9	7	10	8	2										7			1
4	7	7	10	9	10													1
5						4	2	5	4	3	8							2
6						4	2	5	6	4	8							2
7						7			3	6	6							2
8						2	6	8	9	4								2
9							10					8	7	9	9	3	7	3
10												4	2	10	9	6	6	3
11												8	2	3	3	1	4	3
12		5										5	9		5	7		3
13												2			3	2	9	3
Cell of P	1	1	1	1	1	2	2	2	2	2	2	3	3	3	3	3	3	

Table III: Processing routes of parts.

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Route	1, 2, 3, 4	1, 2, 12, 3, 4	1, 2, 3, 4	1, 2, 3, 4	1, 2, 3, 4	5, 6, 8, 7	8, 5, 9, 6	5, 8, 6	6, 7, 5, 8	8, 5, 7, 6	7, 5, 6	12, 9, 13, 11, 10	10, 9, 12, 11	11, 9, 10	3, 11, 12, 10, 9, 13	12, 11, 10, 9, 13	11, 10, 13, 9

3. THE PROPOSED METHOD

3.1 Scheduling framework

Our contribution is to build a scheduling framework that combines an extended disjunctive graph model to show the operation process and to express the solution ideas with mathematical formulation. The whole solution method consists of two components: one is the IHS algorithm, as shown in the left part of Fig. 1, and the other is the trained ANFIS, as shown in the right part of Fig. 1. These two components are described in detail in Sections 3.1 and 3.2, respectively.

3.2 Improved harmony search (IHS) algorithm

The basic HS algorithm is improved to solve the problem considered in this paper as described in steps 1 to 5 below.

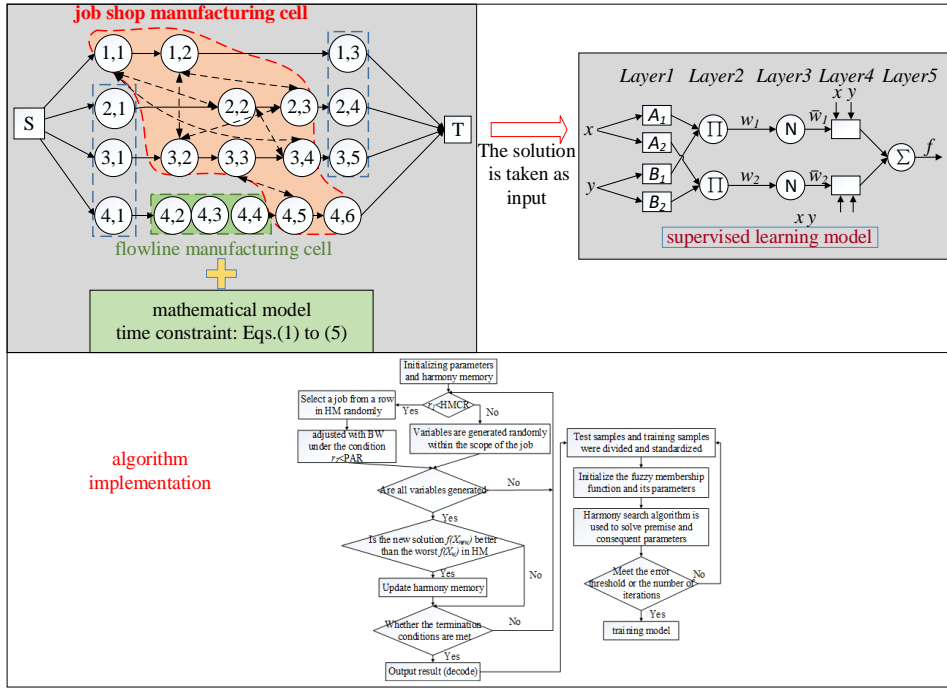


Figure 1: Scheduling framework.

Step 1. Parameter initialization.

The parameters involved in IHS include the harmony memory size (HMS), the harmony memory consideration rate (*HMCR*), the pitch adjustment rate (*PAR*), the tone fine-tuning bandwidth (*BW*) and the number of iterations (*NI*). For the setting of these parameters, please refer to the calibration experiment in Section 4.2.

Step 2. Initialization of the harmony memory.

Subject to the time constraints, the harmony memory is randomly generated based on operation-based encoding, and the corresponding value of the makespan is recorded. Taking cell 3 as an example, there are 5, 4, 3, 5, 5, and 4 intracell operations for P12, P13, P14, P15, P16 and P17, respectively, excluding the first operation for P15, namely, 26 intracell operations, and there are 2 foreign-cell operations, namely, the third operation for P2 and the third operation for P7. Based on operation-based encoding, the corresponding encoding values for P12, P13, P14, P15, P16, P17, P2 and P7 are 5, 4, 3, 5, 5, 4, 1 and 1, respectively.

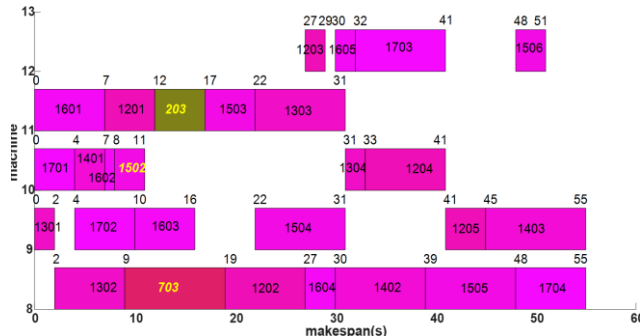
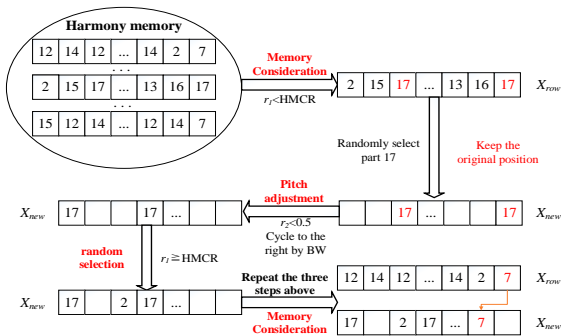


Figure 2: The flow chart of the proposed method. Figure 3: The scheduling results for cell 3.

Step 3. Creation of a new harmony.

A new harmony is created on the basis of three rules, as illustrated in Fig. 2.

Memory consideration: When a random number r_1 is less than *HMCR*, a row X_{row} is randomly selected from the harmony memory, and a part j is randomly selected from the part range $[1, M]$. Based on the position of part j in X_{row} , the same number j is placed in the same

position in X_{new} . If an operation already exists in this location, it is instead placed randomly in a free location.

Pitch adjustment: By comparing a random number r_2 against PAR , the existing elements in X_{new} are adjusted by a cyclic shift. When $r_2 \leq 0.5$, all operations are shifted one bit to the right. When r_2 is greater than 0.5, all operations are shifted one bit to the left. The number of bits moved is BW .

Random selection: A part j is randomly selected from the part range $[1, M]$. If this part does not exist in X_{new} , a free position will be randomly selected, and the corresponding operation number will be placed there.

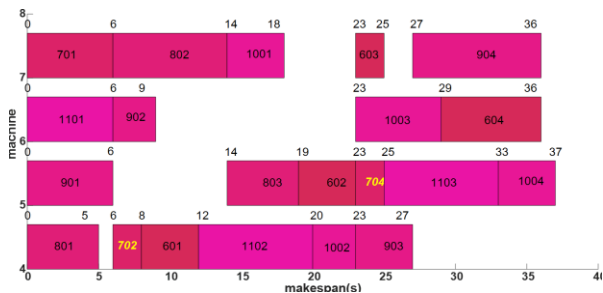
Step 4. Updating of the harmony memory.

With the minimum makespan as the objective, if the scheduling result represented by the new harmony vector X_{new} is better than that represented by the worst harmony vector X_w in the current harmony memory, X_w is replaced by X_{new} in the harmony memory.

Step 5. Repetition of steps 3 and 4 until the termination criteria are met.

The scheduling results are related to the scheduling sequence of the cells. Taking the CMS represented in Table I as an example, there are three cells; therefore, there are six possible CS sequences, (3, 2, 1), (3, 1, 2), (2, 3, 1), (2, 1, 3), (1, 3, 2) and (1, 2, 3). In any such sequence, the scheduling results for a previous cell serve as scheduling conditions for latter cell. For example, in the sequence (3, 2, 1), cell 3 is the first cell that will be scheduled. Table II shows that the first operation for P15, represented by 1501 (note that the rest of the operations are represented in the same way), is an inter-cell operation that is to be processed in cell 1, although the rest of the operations for P15 will be processed in cell 3. In addition, there are two inter-cell operations that will be processed in cell 3, 203 and 703, from cell 1 and cell 2, respectively.

Since 1502 is the operation subsequent to 1501, the earliest start time of 1502 is equal to the processing time of 1501, i.e., $S_{fo_{ji}}$ is equal to 7 s, according to Eq. (2). Similarly, the earliest start time of 203 is equal to the sum of the processing times of 201 and 202, that is, $S_{fo_{ji}}$ is equal to 9 s. The earliest start time of 703 is equal to the sum of the processing times of 701 and 702, that is, $S_{fo_{ji}}$ is equal to 8 s. Now, the above time conditions are used to schedule cell 3 via the IHS algorithm. The results are shown in Fig. 3. The values inside the rectangles in all the following Gantt charts represent the operations of the various jobs. In addition, different jobs are shown in different colours, while different operations of the same job are represented in the same colour. The numbers above the left and right sides of any rectangle represent the start and end times of the corresponding operation, respectively.



Once we have finished calculating the results for the other five CS sequences in the same way, we can find that the optimal scheduling sequence is (1, 3, 2), and the optimal value of makespan is 61 s in cell 1, 35 s in cell 2, 55 s in cell 3, respectively. So the optimal value of makespan in CMS is 61 s. To verify the proposed solution, scheduling results are also generated using the overall solution method and compared in Table IV. The comparison reveals that the makespans achieved with the HWHS method are shorter than those of the overall solution both for each cell and for the CMS as a whole. Thus, it can be seen that the HWHS method is better than the overall solution method not only in finding the optimal solution for each individual cell but also in finding the optimal result for the CMS.

Table IV: The scheduling results of two solution methods.

Cell No.	Solution method	
	Overall solution (s)	HWHS method (s)
cell 1	73	61
cell 2	47	35
cell 3	71	55
CMS	73	61

3.3 Adaptive neuro-fuzzy inference system (ANFIS) structure

The results of the new solution are better, but there are still some deficiencies. First, the run time will increase with the number of cells, so the parallel operation function *parfor* in MATLAB is used in place of the traditional for loop, and a multiprocessor computer is used to run multiple cases at a time to reduce the run time. Second, the generalizability is poor, and as the number of cells continues to increase, even multiprocessor technology becomes insufficient to meet the computing requirements. Take a problem with 6 cells as an example, according to the above method, the solution for each case requires 6! cycles to compute. Thus, when there are 16 cases, 16×720 cycles are required. To solve this problem, this paper adopts an ANFIS structure, which is a type of supervised learning model based on the Takagi–Sugeno (T-S) model. The whole system includes five layers, namely, a fuzzification layer, a rule layer, a normalization layer, a defuzzification layer and an output layer.

The parameters can be adjusted by a learning algorithm, either the backpropagation (BP) algorithm or least squares estimation (LSE). The BP algorithm has certain shortcomings, such as a slow convergence speed, susceptibility to falling into local extreme points, and poor generalization ability. Therefore, much effort has been made to develop improved learning algorithms. Similarly, the IHS algorithm is applied in this paper to optimize the parameters in the ANFIS training process.

4. COMPUTATIONAL RESULTS

4.1 Experimental design

According to [14], a CMS has an average of 5.9 cells, and each cell contains an average of 6 machines. So, the ranges of Number of cells, Number of machines in each cell and Number of parts in each cell are (3, 4, 5), (5, 6, 7, 8) and (8, 12, 20, 30) respectively. Random cases were generated to verify the effectiveness of the HWHS solution since there is no standard case to use. The processing times obey a uniform distribution on the interval [1, 10], and each case includes at least one job shop cell and at least one flow-line manufacturing cell.

The relative deviation (*RD*) is used as the test index to judge the performance of HWHS, as shown in Eq. (6).

$$RD = (B - B_i) / B \times 100 \% \quad (6)$$

where B_i is the solution obtained by the i^{th} solution method and B is the optimal solution obtained by all solution methods.

Table V: Experimental results.

Instance $c \times m \times p$	HWHS		Overall solution [16]		SVS [15]	
	Value	RD	Value	RD	Value	RD
3×5×8	(2,1,3) (76,54,48)	0.00	80,65,82	0.08	89,86,92	0.21
3×5×12	(2,1,3) (115,76,69)	0.00	112,118,121	0.05	129,119,132	0.15
3×5×20	(3,1,2) (181,112,111)	0.00	174,199,171	0.10	180,177,216	0.19
3×5×30	(3,2,1) (159,309,149)	0.05	204,231,312	0.06	294,240,225	0.00
3×6×8	(2,1,3) (99,76,59)	0.00	93,107,91	0.08	73,90,101	0.02
3×6×12	(3,1,2) (91,155,76)	0.04	149,133,131	0.00	120,155,146	0.04
...
3×7×30	(2,3,1) (349,226,209)	0.01	251,291,345	0.00	244,375,310	0.09
3×8×8	(1,2,3) (87,113,67)	0.00	108,115,107	0.02	92,111,117	0.04
3×8×12	(1,3,2) (168,108,94)	0.01	189,154,164	0.13	150,167,167	0.00
3×8×20	(1,3,2) (243,203,131)	0.00	271,174,223	0.12	237,211,263	0.08
...
4×6×12	(4,3,2,1) (121,89,110,78)	0.00	117,133,130,117	0.10	91,129,121,132	0.09
4×6×20	(3,2,1,4) (125,159,183,120)	0.00	197,181,198,217	0.08	201,192,182,195	0.10
4×6×30	(3,1,4,2) (191,397,210,180)	0.19	275,272,241,336	0.01	334,237,329,266	0.00
4×7×8	(3,4,1,2) (109,89,81,73)	0.00	93,105,116,88	0.06	90,90,110,98	0.01
4×7×12	(2,3,4,1) (105,85,126,164)	0.09	172,127,138,122	0.14	117,133,141,151	0.00
...
4×8×30	(4,3,2,1) (366,214,194,216)	0.00	287,384,272,273	0.05	327,280,394,267	0.08
5×5×8	(4,3,1,2,5) (52,53,75,50,45)	0.15	78,72,76,68,63	0.20	63,63,57,60,65	0.00
5×5×12	(5,4,1,3,2) (74,85,70,75,61)	0.00	95,88,92,94,94	0.12	90,90,85,90,84	0.06
5×5×20	(2,3,4,1,5) (121,109,113,113,95)	0.00	134,136,127,137,132	0.13	125,123,123,133,132	0.10
5×5×30	(5,1,2,4,3) (162,178,170,168,158)	0.00	191,196,188,181,188	0.10	181,183,171,177,173	0.03
...
5×7×30	(4,3,5,2,1) (193,197,229,199,186)	0.02	215,196,236,221,215	0.05	221,219,220,213,224	0.00
5×8×8	(1,3,5,4,2) (87,68,73,91,62)	0.05	94,87,87,78,95	0.09	82,84,81,87,87	0.00
5×8×12	(1,2,4,5,3) (101,92,95,106,80)	0.00	109,108,112,99,117	0.10	111,110,113,111,112	0.07
5×8×20	(4,1,2,3,5) (130,148,149,151,133)	0.00	180,180,174,170,171	0.19	142,158,154,148,155	0.05
5×8×30	(1,3,5,4,2) (213,198,192,210,177)	0.00	233,200,234,238,207	0.12	220,223,234,232,234	0.10
Average		0.03		0.08		0.05

4.2 Results and discussion of HWHS

All calculations were performed using MATLAB software on a personal computer with a 2.60 GHz CPU and 8 GB of RAM. To speed up the program, the parallel operation function *parfor* in MATLAB was used in place of the traditional for loop so that multiple cases could be run at the same time. The corresponding results are shown in Table V above. The values in red represent the maximum makespan among the cells, that is, the makespan of the CMS.

The results were analysed from the perspectives of both the CMS as a whole and individual cells. For ease of viewing, the results for the CMS are displayed as a graph in Fig. 6. In most cases (approximately 70 %), the new method HWHS is obviously superior to the other two solutions. The same conclusion can also be drawn from Table V, which shows that the average *RD* is 0.03, which is markedly smaller than the *RD*s of 0.08 and 0.05 of the other two solutions.

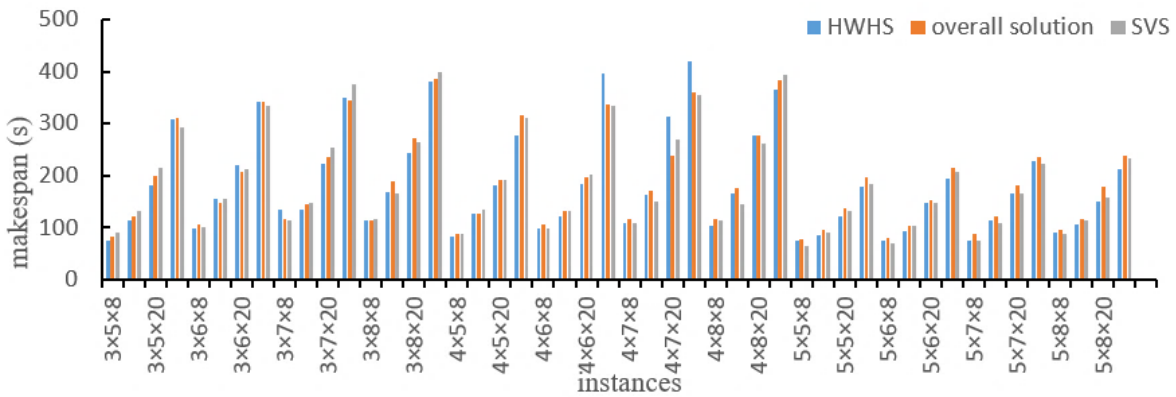


Figure 6: Performance comparison of different scale instances.

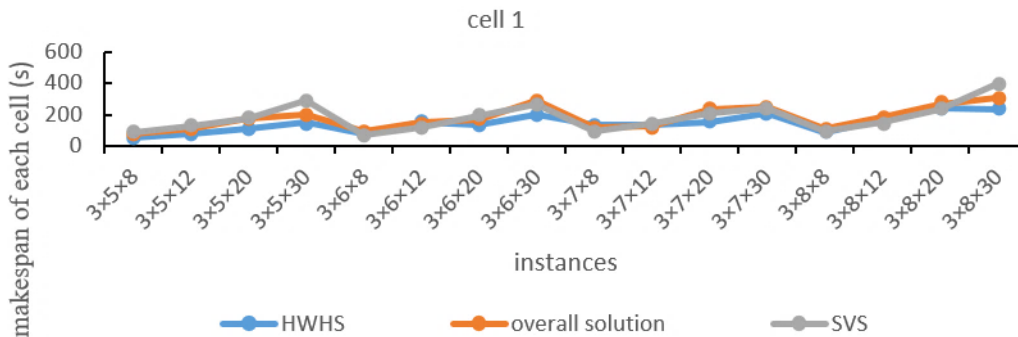


Figure 7: Performance comparison for cell 1 in cases with three cells.

The results for the individual cells are presented in Fig. 7. This figure shows that the results of HWHS are obviously better than those of the other two solutions for most cases in each cell. The same conclusions were obtained for four cells and five cells.

There are two possible reasons for the above results. One is that in contrast to the other two solutions, HWHS considers the interactions between the cells, determines the appropriate sequence in which to sort the cells for scheduling, and addresses the overall optimization of the CMS on the basis of optimizing the makespan of each cell. The second is the HS algorithm adopted in HWHS, which not only makes full use of the existing harmony information in the harmony memory but also considers each segment of individual information separately.

4.3 Optimal ANFIS structure

Using the proposed HWHS method, the makespan values of 400 scheduling sequences in $6 \times 5 \times 8$ cases were calculated. 85 % of the values were used as training data, and the remaining

15 % were used as test data. The input parameters were the total number of operations, the total number of machines, the total processing times of the operations and the CS sequence, and the output was the optimal makespan value for each cell. To account for the different ranges and different units of the input parameter data, the data were first normalized.

Different ANFIS structures based on the IHS algorithm (ANFIS-IHS) were designed, and the value of the mean absolute percentage error (*MAPE*) relative to the actual value under each structure was calculated. Each ANFIS-IHS structure was run 10 times, and the average *MAPE* value was calculated by combining different parameters. For the membership function, the membership centre range was set to [-2, 2], and the width range was set to (0, 2]. Consequently, the ANFIS-IHS structure with the minimal *MAPE* value (0.118) was selected as the optimal structure, as seen in Table VI.

Table VI: Different ANFIS-IHS structures and their corresponding *MAPE* values.

Input MF type	Output MF type	Generate FIS	Number of MFs	Opt.	And	Or	<i>MAPE</i>
gauss	linear	sub. clustering	2	hybrid	min	max	0.128
trap	constant	grid partition		hybrid	prod	probor	0.156
tri	constant	sub. clustering		hybrid	prod	max	0.187
gbell	linear	sub. clustering	2	hybrid	min	probor	0.141
gauss	constant	grid partition		hybrid	prod	probor	0.180
trap	linear	sub. clustering		hybrid	prod	probor	0.118
tri	linear	grid partition	2	hybrid	min	max	0.174
gbell	constant	sub. clustering	3	hybrid	min	probor	0.182
gauss	linear	sub. clustering		hybrid	prod	max	0.197
trap	linear	grid partition		hybrid	prod	probor	0.202
tri	linear	grid partition	3	hybrid	min	probor	0.124
gbell	constant	sub. clustering	3	hybrid	min	max	0.167

Table VII: Predicted makespan values with different ANFIS structures.

ID	No.	HWHS	ANFIS-IHS (this work)	ANFIS-ABC [17]	ANFIS-GA [18]	ANFIS-HTGA [19]
1	6 5 4 3 2 1	106	104.5	105.7	104.5	107.9
2	6 5 4 3 1 2	128	124.56	135.7	139.1	146.4
3	6 5 4 2 3 1	151	153	154.2	154.6	158.9
4	6 5 4 2 1 3	141	142	144.1	143.8	146.2
5	6 5 4 1 3 2	129	126.6	127.8	126.8	125.3
6	6 5 4 1 2 3	158	159.2	160	157.2	163.1
7	6 5 3 4 2 1	173	169	178.3	175	180.4
8	6 5 3 4 1 2	189	187	188.2	187.9	200.3
9	6 5 3 2 4 1	143	146.3	147.5	148.9	156.2
10	6 5 3 2 1 4	201	198.2	195.7	186.5	187.8
11	6 5 3 1 4 2	189	187.7	188	187.3	178.7
12	6 5 3 1 2 4	127	124.5	126.9	129.9	131.1
13	6 5 2 4 3 1	158	157.1	161.2	167.3	154.3
14	6 5 2 4 1 3	107	104.3	103.3	106.9	109
15	6 5 2 3 4 1	113	112	111.5	115.4	134.1
16	6 5 2 3 1 4	165	167.2	163.5	165.2	177.2
R²	-	-	0.976	0.947	0.725	0.814

To further verify the effectiveness of the ANFIS-IHS structure, it was compared with other improved ANFIS structures from the existing literature. For the selected configuration of $6 \times 5 \times 8$, 320 untested combinations remained ($6! = 720$ in total), which were used to verify the performance of the ANFIS-IHS structure. The maximum makespan values for each combination as obtained by HWHS are compared with the values predicted under the different ANFIS structures in Table VII.

The goodness of fit R^2 is used as the parameter to judge the degree of fitting. Let y be the value to be fitted; its mean value is \bar{y} , and the fitted value is \hat{y} , as shown in Eq. (7).

$$R^2 = \frac{SSR}{SST} = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = 1 - \frac{SSE}{SST} \quad (7)$$

By applying Eq. (7) to the data in Table VII, we can find that the R^2 value of the optimal ANFIS-IHS structure is 0.976, which is closest to 1 compared with the R^2 values of the other ANFIS structures, which are 0.947, 0.725 and 0.814. That is, the values predicted by the proposed ANFIS-IHS and found by the HWHS algorithm have a high fitting degree. Thus, it can be seen that the data for each cell as obtained via HWHS can serve as a suitable training set to obtain an optimal ANFIS-IHS structure that can predict new cases well.

5. CONCLUSIONS AND OUTLOOK

This paper focuses on the solution of cell scheduling considering inter-cell movement, and we propose a new scheduling framework in which a novel mathematical model is provided considering the sequence of cells and an extended disjunctive graph model is established with multi-cell. The improved HS algorithm is adapted to minimize makespan by modifying three key aspects: memory consideration, pitch adjustment and random selection. To confirm the applicability of the proposed solution, it is compared with the two main existing methods on random cases of different scales. The results show that the new solution is superior to the other two methods, for both on the CMS as a whole and each individual cell. In addition, to enhance the general applicability of our solution, an intelligent system is developed based on the ANFIS concept. The attributes of the CS problem to be solved are provided as input, and the scheduling results obtained by the new solution are output. Different ANFIS-IHS structures are constructed, and the structure with the minimum *MAPE* is regarded as the optimal ANFIS-IHS. To further verify the effectiveness of the ANFIS-IHS structure, it is compared with other improved ANFIS structures from the existing literature. Finally, a high fitting degree is obtained between the predicted results of the proposed ANFIS-IHS and our new strategy, verifying the validity of the ANFIS-IHS structure.

In reality, there may be many complex constraints, establishing a model that can capture as many of these practical constraints as possible is a key problem that urgently needs to be solved. At the algorithm level, various machine learning algorithms have been well applied in the fields of data mining, image recognition, speech recognition and character recognition. How to apply them to the field of scheduling is an interesting direction of exploration. In terms of hardware, GPUs can be used to solve scheduling problems and improve the solution speed.

ACKNOWLEDGEMENT

This work was supported by Guangdong science and technology innovation strategy plan, China with the project number 202011020005.

REFERENCES

- [1] Zhang, H.; Zhang, Y. Q. (2020). A discrete job-shop scheduling algorithm based on improved genetic algorithm, *International Journal of Simulation Modelling*, Vol. 19, No. 3, 517-528, doi:10.2507/IJSIMM19-3-CO14
- [2] Shen, C.; Chen, Y. L. (2020). Blocking flow shop scheduling based on hybrid ant colony optimization, *International Journal of Simulation Modelling*, Vol. 19, No. 2, 313-322, doi:10.2507/IJSIMM19-2-CO7

- [3] Shi, D. L.; Zhang, B. B.; Li, Y. (2020). A multi-objective flexible job-shop scheduling model based on fuzzy theory and immune genetic algorithm, *International Journal of Simulation Modelling*, Vol. 19, No. 1, 123-133, doi:[10.2507/IJSIMM19-1-CO1](https://doi.org/10.2507/IJSIMM19-1-CO1)
- [4] Zeng, C. K.; Tang, J. F. (2014). Blocking job shop cell scheduling with automated guided vehicles, *Proceedings of the 11th World Congress on Intelligent Control and Automation*, 438-442, doi:[10.1109/WCICA.2014.7052753](https://doi.org/10.1109/WCICA.2014.7052753)
- [5] Li, D. N.; Wang, Y.; Xiao, G. X.; Tang, J. F. (2013). Dynamic parts scheduling in multiple job shop cells considering intercell moves and flexible routes, *Computers & Operations Research*, Vol. 40, No. 5, 1207-1223, doi:[10.1016/j.cor.2012.11.024](https://doi.org/10.1016/j.cor.2012.11.024)
- [6] Tesic, Z.; Stevanov, B.; Jovanovic, V.; Tomic, M.; Kafol, C. (2016). Period batch control – a production planning system applied to virtual manufacturing cells, *International Journal of Simulation Modelling*, Vol. 15, No. 2, 288-301, doi:[10.2507/IJSIMM15\(2\)8.337](https://doi.org/10.2507/IJSIMM15(2)8.337)
- [7] Nie, X. D.; Chen, X. D.; Chen, X. (2016). Simulation study of flexible manufacturing cell based on token-oriented Petri net model, *International Journal of Simulation Modelling*, Vol. 15, No. 3, 566-576, doi:[10.2507/IJSIMM15\(3\)CO14](https://doi.org/10.2507/IJSIMM15(3)CO14)
- [8] Lin, S.-W.; Ying, K.-C. (2019). Makespan optimization in a no-wait flowline manufacturing cell with sequence-dependent family setup times, *Computers & Industrial Engineering*, Vol. 128, 1-7, doi:[10.1016/j.cie.2018.12.025](https://doi.org/10.1016/j.cie.2018.12.025)
- [9] Li, Y. Z.; Li, X. P.; Gupta, J. N. D. (2015). Solving the multi-objective flowline manufacturing cell scheduling problem by hybrid harmony search, *Expert Systems with Applications*, Vol. 42, No. 3, 1409-1417, doi:[10.1016/j.eswa.2014.09.007](https://doi.org/10.1016/j.eswa.2014.09.007)
- [10] Neufeld, J. S.; Gupta, J. N. D.; Buscher, U. (2016). A comprehensive review of flowshop group scheduling literature, *Computers & Operations Research*, Vol. 70, 56-74, doi:[10.1016/j.cor.2015.12.006](https://doi.org/10.1016/j.cor.2015.12.006)
- [11] Li, D. N.; Meng, X. W.; Li, M.; Tian, Y. N. (2016). An ACO-based intercell scheduling approach for job shop cells with multiple single processing machines and one batch processing machine, *Journal of Intelligent Manufacturing*, Vol. 27, No. 2, 283-296, doi:[10.1007/s10845-013-0859-2](https://doi.org/10.1007/s10845-013-0859-2)
- [12] Deliktas, D.; Torkul, O.; Ustun, O. (2019). A flexible job shop cell scheduling with sequence-dependent family setup times and intercellular transportation times using conic scalarization method, *International Transactions in Operational Research*, Vol. 26, No. 6, 2410-2431, doi:[10.1111/itor.12407](https://doi.org/10.1111/itor.12407)
- [13] Elmi, A.; Solimanpur, M.; Topaloglu, S.; Elmi, A. (2011). A simulated annealing algorithm for the job shop cell scheduling problem with intercellular moves and reentrant parts, *Computers & Industrial Engineering*, Vol. 61, No. 1, 171-178, doi:[10.1016/j.cie.2011.03.007](https://doi.org/10.1016/j.cie.2011.03.007)
- [14] Zeng, C. K.; Tang, J. F.; Yan, C. J. (2015). Job-shop cell-scheduling problem with inter-cell moves and automated guided vehicles, *Journal of Intelligent Manufacturing*, Vol. 26, No. 5, 845-859, doi:[10.1007/s10845-014-0875-x](https://doi.org/10.1007/s10845-014-0875-x)
- [15] Solimanpur, M.; Vrat, P.; Shankar, R. (2004). A heuristic to minimize makespan of cell scheduling problem, *International Journal of Production Economics*, Vol. 88, No. 3, 231-241, doi:[10.1016/S0925-5273\(03\)00196-8](https://doi.org/10.1016/S0925-5273(03)00196-8)
- [16] Tang, J. F.; Wang, X. Q.; Kaku, I.; Yung, K.-L. (2010). Optimization of parts scheduling in multiple cells considering intercell move using scatter search approach, *Journal of Intelligent Manufacturing*, Vol. 21, No. 4, 525-537, doi:[10.1007/s10845-008-0236-8](https://doi.org/10.1007/s10845-008-0236-8)
- [17] Qi, C. C.; Ly, H.-B.; Le, L. M.; Yang, X. Y.; Guo, L.; Pham, B. T. (2021). Improved strength prediction of cemented paste backfill using a novel model based on adaptive neuro fuzzy inference system and artificial bee colony, *Construction and Building Materials*, Vol. 284, Paper 122857, 14 pages, doi:[10.1016/j.conbuildmat.2021.122857](https://doi.org/10.1016/j.conbuildmat.2021.122857)
- [18] Azad, A.; Manoochehri, M.; Kashi, H.; Farzin, S.; Karami, H.; Nourani, V.; Shiri, J. (2019). Comparative evaluation of intelligent algorithms to improve adaptive neuro-fuzzy inference system performance in precipitation modelling, *Journal of Hydrology*, Vol. 571, 214-224, doi:[10.1016/j.jhydrol.2019.01.062](https://doi.org/10.1016/j.jhydrol.2019.01.062)
- [19] Ho, W.-H.; Chen, J.-X.; Lee, I.-N.; Su, H.-C. (2011). An ANFIS-based model for predicting adequacy of vancomycin regimen using improved genetic algorithm, *Expert Systems with Applications*, Vol. 38, No. 10, 13050-13056, doi:[10.1016/j.eswa.2011.04.109](https://doi.org/10.1016/j.eswa.2011.04.109)