Design Engineering

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

A Multi-Rule Algorithm for Multi-Shop Integrated Scheduling Problem

Zhiqiang Xie[1*], Xiaowei Zhang[1], Yingchun Xia[1], Yu Xin[2]
[1]School of Computer Science and Technology, Harbin University of Science and Technology, Harbin China
[2]Faculty of Electrical Engineering and Computer, Ningbo University, Ningbo, China
*Corresponding Author: Zhiqiang Xie

Abstract:

Aiming at the problems that the existing two-shop integrated scheduling algorithm cannot deal with the case of multiple-shop, and that the existing algorithm is too sensitive to the characteristic of instance to obtain a better solution, a multi-rule algorithm (MRA) is proposed to solve the two issues. MRA firstly uses multi-rule combined sorting strategy to generate a variety of sorting schemes, then schedules the sorting schemes according to the multi-shop tentative strategy, and finally determines the scheduling solution with the shortest completion time as the scheduling result. By analysing the simulation instances, the effectiveness of the MRA algorithm is verified. It can be seen from the simulation results that the MRA algorithm is better than the control algorithm in terms of completion time and calculation time.

***Keywords**: Nonstandard scheduling, Integrated scheduling, Multi-rule sorting strategy, Multi-shop tentative.*

I. INTRODUCTION

Because the production scheduling level of an enterprise directly affects the production and delivery capacity, the production scheduling problem has become one of the hot research fields after years of development. Traditionally, the production scheduling focuses on the efficiency of large-scale production by first decomposing the product process tree into unconstrained workpieces, and then carrying out subsequent operations. Due to the emphasis on large-scale production and high repetition probability of parts in the production process, it is generally used to reserve standard parts for auxiliary production. The representative algorithms are as follows: 1) Small batch products are scheduled by the job shop scheduling algorithm, where there is only a linear serial constraint relationship between the jobs, so it is a pure processing scheduling [1-5]. 2) Large batch of the same products are scheduled by flow shop scheduling algorithm [6-9]. But in reality, there are some order-oriented small and medium-sized enterprises that organize production in order, with a large

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

number of single small batch of non-standard products that need marking, manual grinding, scraping and precision samples in the production process, whose parts parameters are basically unpredictable, carry out real-time production completely in order, and often need assistance from multi-shop. As a result, such enterprises need to deal with a large number of non-standard parts. If they adopt the method of decomposition process tree, they will ignore the fact that processing and assembly can be processed in parallel, waste the idle processing capacity of external multi-shop, and reduce the flexibility of production. To solve the scheduling problems in real-time production, some scholars began to pay attention to the research of multi-shop integrated scheduling theory (MPIST) [10-14]. The main features of MPIST theory are: the scheduling granularity is further refined, the product process tree is not decomposed, and the constraints between operations are directly represented by the process tree, as shown in Figure 1; processing and assembly are directly scheduled according to the process tree; multi-shop collaborative processing is adopted; and the gap time of processing and assembly is used as much as possible. The scheduling of multiple products is equivalent to merging multiple process trees into a new virtual process tree, with the difference only in the size and shape of nodes.

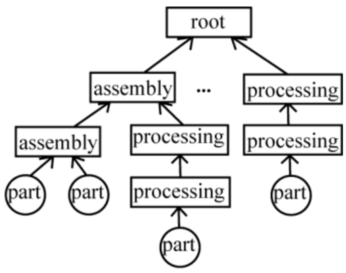

Figure 1. Instance of constraint relationship of a process tree.

The research of multi-shop scheduling focuses on the scheduling algorithm for two shops, for instance: in reference [15], an algorithm based on optimal time (OT) is proposed. In OT algorithm, the operation is first divided into several operation series by using the critical path rule, and then the crucial operation series are used as the basic scheduling scheme, and then the remaining operations are tentatively scheduled on the basic scheme in order, and so on, until all operations are scheduled. However, the tentative scheduling strategy has a high complexity, i.e. cubic polynomial; only a single rule is adopted to determine the scheduling order; although the next operation is considered in each tentative scheduling, it is only optimized locally, because only one complete scheduling scheme is generated in the algorithm life cycle; moreover, OT algorithm is designed for the symmetrical processing resources of two shops, and it can't handle the asymmetric resources. In reference [16], an algorithm based on allied critical path method (ACPM) [16] is proposed. During the process of scheduling, ACPM takes priority to the operation on the critical path, because the length of the path is the lower limit of the processing time, which has an important impact on the completion time. From the perspective of process tree, ACPM algorithm only optimizes the process tree longitudinally, which makes it very

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

sensitive to the process characteristics of products, that is, the solution quality of different instances of the same problem is not stable. In addition, the strategy of algorithm design in reference [16] can only deal with two shops, in which the number of operation migration is defined instead of transportation time. However, the replacement of transportation time with migration times can only be established when there are two shops, because if there are more than two shops with different transportation distances, the migration times will be meaningless. As a result, the strategy designed in [16] is highly directional and limited to two workshops, because the adaptation cost is high in the case of any number of shops. According to the colouring strategy of graph theory, an algorithm based on the principle of the neighbourhood rendering (PNR) is proposed in literature [17]. In PNR algorithm, the shop is selected according to the calculated rendering factor and influence factor. However, this strategy pays too much attention to load balancing but ignores the influence of scheduled operations on parallel operations, which makes the algorithm sensitive to the characteristics of the examples and affects the solution quality. Like ACPM algorithm, in PNR, migration times are also used to calculate the cost of multi-shop collaboration, which results in the limited scheduling strategy to two shops, and a high adaptation cost in case of multi-shop.

Based on the analysis of the above algorithms, the following technical difficulties need to be overcome:

1) The existing methods replace the transportation time with migration count [15-17]. Actually, one operation needs to be delayed before it reaches the next station for processing. The final output is a Gantt chart that already contains such delay.

2) In the existing multi-shop integrated scheduling algorithm, the machines are divided into symmetrical and asymmetric processing resources, i.e. there is no flexible machine, and the scheduling algorithms are designed for these two cases respectively [15-17]. In fact, there are many workshops which are likely to be external collaborative mode, meanwhile, there is basically no shop with symmetrical machine resources. Therefore, the new design would be able to deal with any flexible machine resources.

To solve the above problems, a multi-rule algorithm (MRA) is proposed to solve the multi-shop integrated scheduling problem (MSISP), in which the up path, down path, layer properties and the long-time operation attributes are combined to generate four sequencing schemes, which meet the constraints of reverse scheduling and save the steps of converting the process tree to the reverse process tree. Then the multi-shop tentative scheduling strategy is used to schedule four sequencing schemes respectively to get four complete scheduling schemes, and finally output the scheme with the shortest completion time. This work focuses on the multi-shop integrated scheduling problem, to provide a manufacturing scheme that considers the transport time and the production structure.

II. PROBLEM DESCRIPTION

The main characteristic of MSISP is that the operation can be processed in multiple shops, but it takes a certain transportation time from one shop to another. For ease of

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

description, symbols and variables related to MSISP problems are defined as follows: n denotes the number of operations, P_i ($1 \le i \le n$) denotes operation i, l denotes the number of shops, S_w ($1 \le w \le l$) denotes shop w, $|S_w|$ denotes the machine count of S_w, M_{wk} ($1 \le k \le |S_w|$) denotes machine k in shop S_w, t_{iwk} denotes the processing time of operation i on machine M_{wk}, CT_{iwk} is the completion time when operation i is assigned to M_{wk}, ST_{iwk} is the start time when operation i is assigned to M_{wk}, E_{ij} is the directed edge (in Figure 1) which denotes operation i is the predecessor of operation j, D_{uv} denotes the transportation distance between shop S_u and S_v,

The essence of MSISP is a resource allocation problem, that is, design an algorithm to allocate each operation to the corresponding machine which distributed in different shops. Due to the complexity of the actual manufacturing process, it is necessary to abstract and assume the actual problems. Generally, the following hypotheses are made for MSISP:

1) The process tree is used to represent the constraint relationship between operations, as shown in Figure 1.

2) Each operation can only be started after the completion of its predecessor.

3) When the root operation has been finished, the corresponding time is the completion time of the product.

4) When an operation is finished, it is transported to the place of the successive operation. If the machine is set in a different shop, it can be delivered in place only after a fixed delivery time delay. The transfer of objects between shops is completed by special machine, with sufficient transport capacity.

5) The processing resources in each shop are arbitrary, either symmetrical or asymmetrical.

6) The processing machine is exclusive. Only after one operation is finished can the machine process the next one.

Therefore, in this work, the objective function of MSISP is as follow:

$$\min(\max(CT_{iwk})) \tag{1}$$

Equation (1) is the objective function. Product completion time is minimized. Because the constraint relationship between operations is tree shaped, the root operation is the largest at the end of processing, and its completion time is the product completion time.

For any operation, it must be started after its predecessor is finished, that is:

$$ST_{iux} \ge \max(CT_{jvy} + D_{vu}), \forall E_{ji} \tag{2}$$

Equation (2) indicates the constraints of tree structure, that is, when an operation is finished, it will be transported to the place of the successive operation after the corresponding transportation time.

The processing machine is exclusive. The i-th operation on machine M_{wk} must be processed after the previous operation j on the same machine is finished, and the constraint equation is as follow:

$$ST_{iwk} \ge CT_{jwk} \tag{3}$$

III. METHODOLOGY

Design Engineering

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

In general, except for a few small-scale exceptions, most production scheduling problems have no optimal neighbourhood structure, and no precise algorithm of polynomial complexity has been found so far. The optimal solutions of all problems can be obtained by the exhaustive method, but the computational cost is too high to be borne by the computers today. Therefore, most of the existing algorithms are based on heuristic approximation algorithm [10-17]. In order to simplify the solution process, the MSISP problem is usually divided into two subproblems: 1) determining the sequence of operations; 2) determining the shop of operations. The concepts related to MSISP issues are defined as follows:

Definition 1 Operation sorting scheme, symbolized by Q, represents an operation sequencing scheme. The sorting scheme in MRA algorithm refers to reverse sequencing scheme, which meets the constraints of reverse scheduling and the assumptions of MSISP.

Definition 2 Up path. H_i denotes the up path of operation i. The operation string that runs up the target operation to the root node of the process tree is called the up path of the target operation, and the sum of the processing time of all the operations on the operation string is called the up-path length of the target operation.

Definition 3 Down path. L_i denotes the up path of operation i. There may be multiple operation strings running down the target operation to the leaf node. The string with the largest sum of operation string processing times is selected as the down path length of the target operation.

Definition 4 Tree tiers. A process tree is tiered from the root operation down one by one, that is, the tier attribute Tier(i) of each operation in the process tree is obtained.

For ease of understanding, the process tree shown in Figure 2 is used for description. The data in box are operation id, machine id and the processing time on the machine respectively. Where, the up path of operation 2 is {P_2, P_1}, and the length is H_2 = 23. There are two paths for the down path of operation 2: one is {P_2, P_3, P_5} with a path length of 27; the other is {P_2, P_4} with a length of 16, and 27 is the longest as the length of down path of operation 2, that is L_2 = 27. Operation 1 is at the root of the process tree, with the tier of 1, and operation 5 is at the bottom of the tree, with the tier of 4. In the case of reverse sequencing, the operation in tier 1 is always the first one to be sequenced.

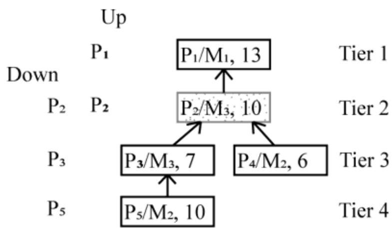

Figure 2. Instance of paths and tiers.

3.1 Multi-rule Combined Sorting Strategy

Generally, according to the basic characteristics of the tree in the graph theory, there is no constraint between the operations of the same tier, and there are pre and post constraints between operations at adjacent tiers. Therefore, the constraint relationship of the process tree can be met by sorting according to the tier attributes of the operations. On the other hand,

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

the path length represents the position of the operation in the process tree. The longer path length of the operation string is, which has an important impact on the production completion time. As a result, the strategy of preferentially scheduling the length path can shorten the product completion time in some cases. Therefore, the length of the path represents a sorting relation of the operation.

To measure the weight of scheduling order from many aspects, the MRA algorithm uses four operation attributes, i.e. up and down path, operation tier attribute and operation processing time, as the key words of operation sequencing, and finally generates four sorting schemes, equivalent to expanding the search space, that is, using a variety of sorting strategies to order operations and explore multiple points in the solution space, which can increase the probability of solution. Based on the above analysis, the specific combination of multi-rule combined sorting methods for MSISP are described in detail as follows:

1) Sorting in ascending order of operation tier attribute as the primary key, and ascending order of operation processing time as the secondary key.

2) Sorting in ascending order of operation tier attribute as the primary key, and descending order of operation processing time as the secondary key, which is also a horizontal optimization on the process tree.

3) Sorting in ascending order of up path as the primary key, ascending order of operation tier attribute as the secondary key, and descending order of operation processing time as the tertiary key.

4) Sorting in descending order of down path as the primary key, ascending order of operation tier attribute as the secondary key, and descending order of operation processing time as the tertiary key.

Finally, four sorting schemes {Q1, Q2, Q3, Q4} are obtained according to the above 4 methods. Among them, the first two sorting methods are typical horizontal optimization, and the last two sorting methods are mainly vertical and horizontal. Therefore, the multi-rule combined sorting method is mainly divided into two parts, one is to calculate operation attributes, the other is to combine different operation attributes to present four sorting schemes.

3.2 Multi-shop Tentative Scheduling Strategy

When scheduling the target operation, the multi-shop tentative scheduling strategy traverses all the devices that can deal with the target operation flexibly, and determines the scheme with the earliest target completion time as the scheduling scheme of the target operation. By multi-shop tentative scheduling strategy, the proposed algorithm in this work is not limited by the machine resources, such as count of machines. Taking Figure 3 as an instance, shop a has 3 machines, shop b has 2 machines, and A_1, A_2 and A_3 are scheduled operations. Then, according to the analysis of Gantt chart, the multi-shop tentative scheduling strategy is to find the gap that can accommodate the target operation on all the machine in shop a and b, and schedule the target operation to the gap with the earliest finish time. Therefore, the target operation in Figure 3 is finally scheduled to machine 2 in the shop a, that is gap1, and the start time is 0.

Design Engineering
ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

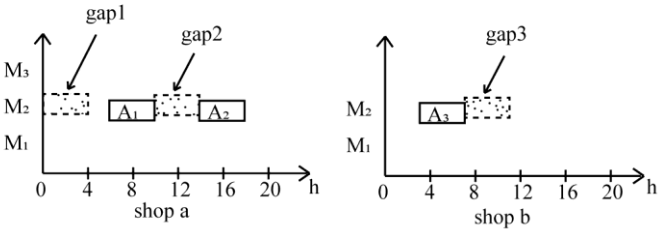

Figure 3. Instance of paths and tiers.

3.3 Algorithm Structure

Based on the previous analysis, the proposed MRA algorithm is mainly divided into three parts: calculation attributes of operations, combined sorting and multi-shop tentative scheduling, that is, the algorithm structure chart shown in Figure 4.

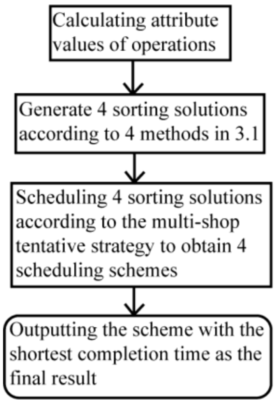

Figure 4. The structure of MRA.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

At present, the experimental in most of the integrated scheduling literatures are based on tree constraints [10-17]. In order to test the effectiveness of the proposed MRA algorithm, a variety of new instances with different scales and types are constructed based on the existing literature. The MRA in this paper is implemented in C#. The main parameters of the experimental computer are: Intel Atom 3735f 1.33GHz, 2GB RAM, Windows 10 operating system.

MSISP benchmark instance includes 4 aspects: constraints of operations, processing time data, machine and shop data, among which the constraint relationship between operations is represented by process tree frequently. For an example in [15], the processing data and constraint relationship are shown in Figure 5. In order to generate a large number of benchmark instances, in this paper, based on the instance in Figure 5, the problem scale is expanded by merging trees, and the node data in the process tree is filled with random data.

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

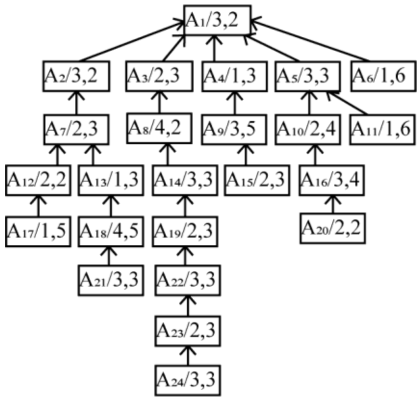

Figure 5. Instance with 24 operations and 2 shops.

4.1 Experiment 1

The OT, ACPM, PNR algorithms in [15-17] and the proposed MRA algorithm in this paper are used to schedule the instance in Figure 5. The processing machine resources are 2 shops, named S1, S2, and 4 machines each shop. Because the control algorithm can't deal with the transportation time, the shop transportation time is calculated according to 1 hour assumed in the instance. The scheduling results are shown in Figures 6-9.

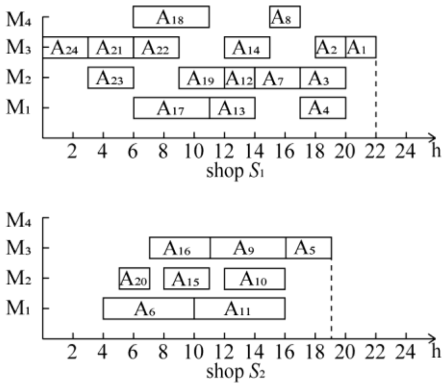

Figure. 6 Gantt chart by OT algorithm.

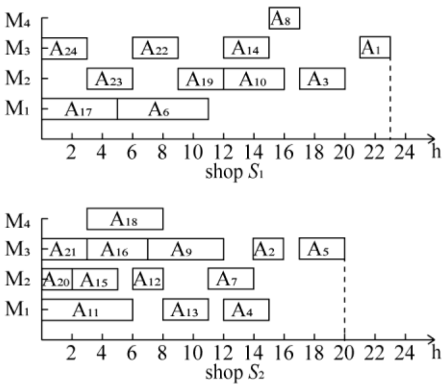

Figure 7. Gantt chart by ACPM algorithm.

In Figure 5, the critical path length of the instance is 22, so 22 is the optimal solution of the instance. The scheduling result of 22 hours is obtained by OT algorithm [15], and the

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

same result is obtained by the proposed MRA algorithm in this paper. The other two algorithms are 23 hours.

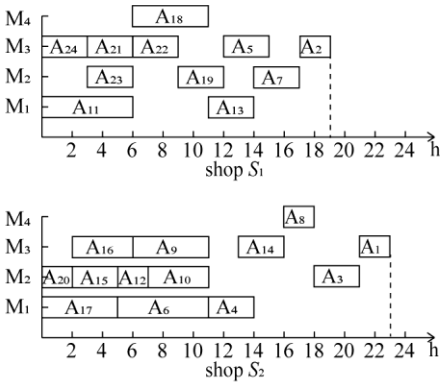

Figure 8. Gantt chart by PNR algorithm.

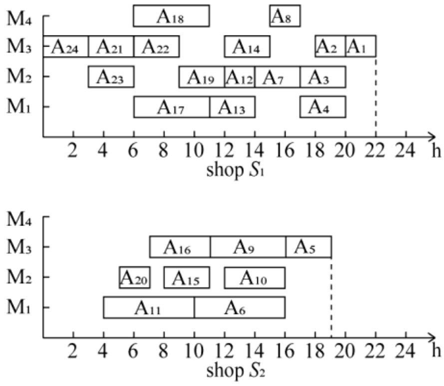

Figure 9. Gantt chart by MRA algorithm.

4.2 Experiment 2

In order to test the performance of the MRA algorithm in the statistical sense, it is necessary to further explore the quality of the algorithm. Based on the instance in Figure 5, a new process tree with 96 operations is generated by using 4 trees in Figure 5, and the process tree is filled with random data. The instances R1-R10 are generated according to this method, and the specific data is shown in TABLE I, in which the processing time is random in the interval [2,10].

TABLE I. Data of R1-R10

Id	Shop count	Transportation time	Machine resource	Operation count
R1	2	1	5 machines/shop	96
R2	2	2	5 machines/shop	96
R3	2	3	5 machines/shop	96
R4	2	4	5 machines/shop	96
R5	2	5	5 machines/shop	96
R6	2	1	5 machines/shop	96
R7	2	2	5 machines/shop	96
R8	2	3	5 machines/shop	96
R9	2	4	5 machines/shop	96

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

R10	2	5	5 machines/shop	96

The scheduling results of all the instances are recorded in TABLE II and Figure 10. It is observed that in 10 random instances, the proposed MRA algorithm is better than the OT algorithm in the scheduling results of 9 instances, except that the results of R1 are the same. Theoretically, the solution space is not expanded by the OT algorithm because only one complete scheduling scheme is generated in its algorithm life cycle. MRA algorithm is used to generate four complete scheduling schemes in the algorithm life cycle, and explore multiple schemes to improve the solution quality.

In order to show the superiority of the proposed MRA algorithm from the time complexity, it is analysed from the experimental point of view. In order to prevent interference and accurately record the running time of the algorithm, the following method is used for the experiment: run the target 24 times, record the running time values, remove the 2 maximum and 2 minimum values, and store the remaining 20 records for drawing the box diagram, as shown in Figure 11. From the experimental result, the MRA algorithm is faster than the control algorithm. The reason is that the optimal time strategy of OT cost too much time on moving the affected operations.

TABLE II. Data of R1-R10

Id	MRA (h)	OT (h)
R1	2	1
R2	2	2
R3	2	3
R4	2	4
R5	2	5
R6	2	1
R7	2	2
R8	2	3
R9	2	4
R10	2	5

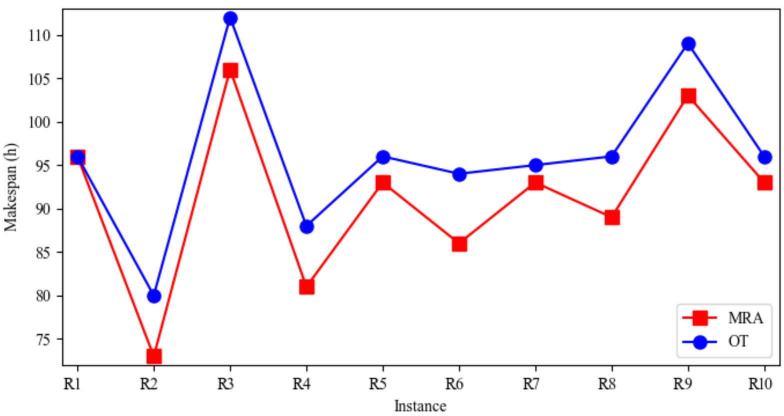

Figure. 10 Scheduling results of R1-R10.

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

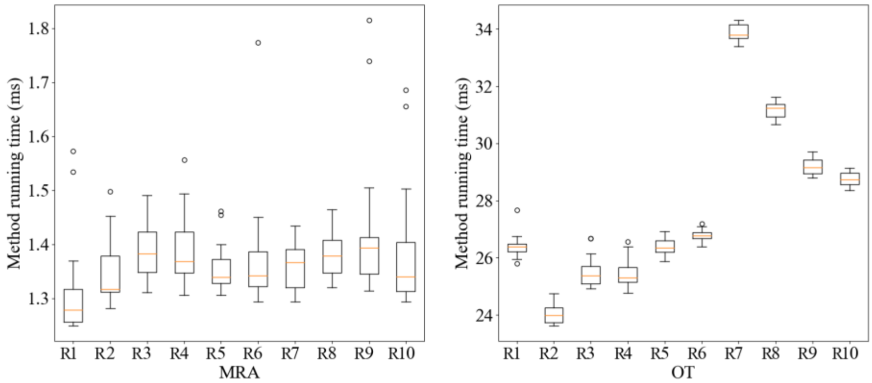

Figure. 11 Method running time of R1-R10.

Based on the above analysis, the MRA algorithm has not only a better solution quality, but also a lower time complexity than the control algorithm.

V. CONCLUSION

This study presented a multi-rule-based strategy on multi-shop integrated scheduling problem. Compared with the existing algorithms, the proposed MRA algorithm can generate 4 complete scheduling schemes in the life cycle, which is equivalent to the multi-point exploration in the solution space and enhances the solution quality. The scheduling results of several instances verified the effectiveness of the proposed MRA algorithm. The conclusions are as follows: (1) Compared with the control algorithms, the MRA multi-shop tentative strategy can deal with any number of processing resources. (2) According to the results of 10 random instances, MRA gets the same result in one instance, and get better results than the control algorithm in the rest of 9 instances, by which it indicates the quality of MRA algorithm in the statistical sense. (3) From the perspective of time complexity, MRA algorithm is faster than the control algorithm. At last, this paper can be used as the basis for further study of multi-shop scheduling problems and other research of scheduling algorithm.

ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China (Grant No.: 61772160, 61602133), the China Postdoctoral Science Foundation (Grant No.: 2016M591541), the Research Fund for the Doctoral Program of Higher Education of China (Grant No.: 20122304110012), the Scientific and Technological Research Project of Heilongjiang Provincial Education Department(Grant No.: 12531105), the Postdoctoral Scientific Research Start Fund Project in Heilongjiang Province(Grant No.: LBH-Q13092), the Postdoctoral Science Foundation of Heilongjiang Province(Grant No.: LBH-Z15096).

REFERENCES

ISSN: 0011-9342
Issue: 11 | Pages: 968 - 979

[1] Xu W H, Ji Z C, Wang Y. (2018) A flower pollination algorithm for flexible job shop scheduling with fuzzy processing time, Modern Physics Letters B. 32, 1840113: 34–36. https://doi.org/10.1142/S02179849 18401139
[2] Pinedo M, Hadavi K. (2016). Scheduling: theory, algorithms, and systems, IIE Transactions 28(8): 695–697
[3] Gu M Z, X. Lu W, Gu J W. (2015) An asymptotically optimal algorithm for large-scale mixed job shop scheduling to minimize the makespan, Journal of Combinatorial Optimization 33(2): 1–23
[4] Hao X C, Gen M, Lin L. (2015) Effective multi-objective EDA for bi-criteria stochastic job-shop scheduling problem, Journal of Intelligent Manufacturing 28(3): 833–845
[5] Sundar S, Suganthan P N, Jin C T. (2017) A hybrid artificial bee colony algorithm for the job-shop scheduling problem with no-wait constraint, Soft Computing 21(5): 1193–1202
[6] Lu C, Li X Y, Gao L. (2017) An effective multi-objective discrete virus optimization algorithm for flexible job-shop scheduling problem with controllable processing times, Computers and Industrial Engineering. 104: 156–174. https://doi.org/10.1016/j.cie.2016.12.020
[7] Behnamian J, Ghomi S M T F. (2016) A survey of multi-factory scheduling, Journal of Intelligent Manufacturing 27(1): 231–249
[8] Ye H F, Li Z S, Chen C. (2014) Adaptive and self-learning pso-based algorithm for job shop scheduling problem, Journal of Jilin University 52(1): 93–97
[9] Zhao S K, Fang S L. (2013) Operation-based encoding and neighbourhood search genetic algorithm for job shop scheduling optimization, Journal of Mechanical Engineering 49(16): 160–169
[10] Gui Z Y, Yang J, Xie Z Q. (2017) Scheduling algorithm for flexible job shop based on pruning and layering, Control and Decision 32(11): 1921–1932
[11] Xie Z Q, Hao S Z, Ye G J. (2009) A new algorithm for complex product flexible scheduling with constraint between jobs, Computers & Industrial Engineering 57(3): 766–772
[12] Lei Q, Guo W F, Song Y C. (2018) Integrated scheduling algorithm based on an operation relationship matrix table for tree-structured products, International Journal of Production Research 56(16): 5437–5456
[13] Shi F, Zhao S K. (2017) Product comprehensive scheduling problems solved by genetic algorithm based on operation constraint chain coding, China Mechanical Engineering 28(20): 2483–2492
[14] Shi F, Zhao S K, Meng Y. (2019) Hybrid algorithm based on improved extended shifting bottleneck procedure and GA for assembly job shop scheduling problem, International Journal of Production Research. 1–22. https://doi.org/10.1080/00207543.2019.1622052
[15] Zhang X H, Xie Z Q, Xin Y. (2017) Integrated scheduling algorithm of two shops based on optimal time, Computer Integrated Manufacturing Systems 23(9): 1938–1949
[16] Xie Z Q, Zhou H X, Gui Z Y. (2013) Integrated scheduling algorithm of two shop based on ACPM, Computer Science 40(4): 193–198
[17] Xie Z Q, Yu J, Chen D Y. (2016) Integrated scheduling algorithm of two shops based on the principle of the neighbourhood rendering, Journal of Mechanical Engineering 52(1): 149-159.