

ORIGINAL ARTICLE

Open Access



Anomalies in Special Permutation Flow Shop Scheduling Problems

Lin Gui, Liang Gao*  and Xinyu Li

Abstract

Recent researches show that there are some anomalies, which are not satisfied with common sense, appearing in some special permutation flow shop scheduling problems (PFSPs). These anomalies can be divided into three different types, such as changing the processing time of some operations, changing the number of total jobs and changing the number of total machines. This paper summarizes these three types of anomalies showing in the special PFSPs and gives some examples to make them better understood. The extended critical path is proposed and the reason why these anomalies happen in special PFSPs is given: anomalies will occur in these special PFSPs when the time of the operations on the reverse critical path changes. After that, the further reason for these anomalies is presented that when any one of these three types of anomalies happens, the original constraint in the special PFSPs is destroyed, which makes the anomalies appear. Finally, the application of these anomalies in production practice is given through examples and also with the possible research directions. The main contribution of this research is analyzing the initial reason why the anomalies appear in special PFSPs and pointing out the application and the possible research directions of all these three types of anomalies.

Keywords: Scheduling, Permutation flow shop, Anomaly

1 Introduction

The permutation flow shop scheduling problem (PFSP), which has the same machine permutation to each job and the same job permutation to each machine [1–3], is a common type of flow shop scheduling problem [4, 5]. Some other new types of shop scheduling problems will form if different constraints are added to it. For example, the no-wait flow shop scheduling problem (NWFSP) forms when the constraint, that a certain operation of a job finishes, and it must immediately enter the next processing machine, is added to PFSP [6–8]. With other constraints, different problems, like no-idle flow shop scheduling problem (NIFSP) [9–11], no-buffering flow shop scheduling problem (NBFSP) [12–14] and synchronous flow shop scheduling problem (SFSP) [15–17] can be got.

In these special PFSPs mentioned above, scholars have found some interesting phenomena. The first who observed this were Abadi, Hall, and Sriskandarajah [18] in 2000. They found in the NWFSP if they slowed down (i.e., increasing the processing time) of some operations may permit a reduction in makespan. Also, in 2005, Spieksma and Woeginger [19] found similar phenomena that if they improved one of the machine speed, the makespan may be increased, or if they reduced the processing time of all operations of a job by a certain proportion, the makespan may be increased. In 2007, Kalczynski and Kamburowski [20] found the anomaly was not only existed in NWFSP but also NIFSP. They mainly analyzed the NWFSP and gave the method to calculate which operations can increase the processing time and how much, and with this method, the completion time can be decreased to the minimum. In 2008, Pan, Zhao and Qu [21] introduced another way to calculate the increase of the processing time to reduce the makespan. In 2017, Waldherr, Knust and Briskorn [22] discussed how to insert voluntary idle time in SFSP to reduce the

*Correspondence: gaoliang@mail.hust.edu.cn
State Key Laboratory of Digital Manufacturing Equipment
and Technology, Huazhong University of Science and Technology,
Wuhan 430074, China

objective functions, like minimization of makespan, total completion time, maximum lateness. In 2018, Panwalkar and Koulamas [23] explained these anomalies with schematic representation. They also did a systematic analysis in 2020 [24], dividing these anomalies into three categories: increasing the processing time of some operations, increasing the number of total jobs and increasing the number of total machines.

This paper aims to explain the anomalies in these special PFSPs further and discuss the role of these anomalies in actual production. In order to have a better understanding, this paper gives some examples of all three types of anomalies in Section 2. Then the Gantt chart and critical path are used to explain the existence of anomalies in the special PFSPs in Section 3. Lastly, a discussion about how to use these anomalies in real-world production is given in Section 4.

2 Examples of Anomalies in Special PFSPs

In special PFSPs, as described in Ref. [24], there exist three types of anomalies as follows.

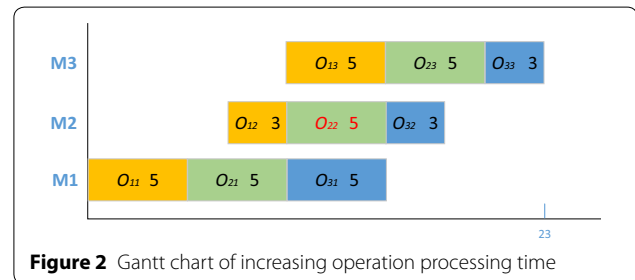
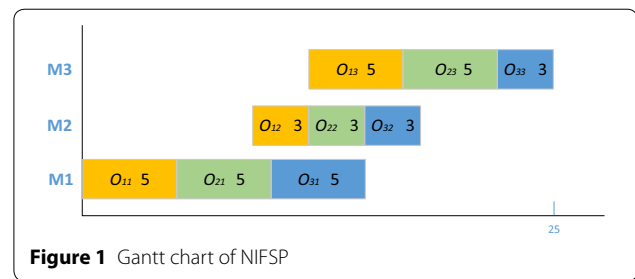
Type 1: The anomaly caused by changing processing time of operations: increasing the processing time of some operations in an optimal schedule causes a reduction in the minimum objective function value, or reducing the processing time of some operations in an optimal schedule causes an increase in the minimum objective function value.

Type 2: The anomaly caused by changing the number of total jobs: increasing the number of total jobs in an optimal schedule causes a reduction in the minimum objective function value, or reducing the number of total jobs in an optimal schedule causes an increase in the minimum objective function value.

Type 3: The anomaly caused by changing the number of total machines: increasing the number of total machines in an optimal schedule causes a reduction in the minimum objective function value, or reducing the number of total machines in an optimal schedule causes an increase in the minimum objective function value.

It is easy to know that when only two jobs or each job with only two operations are given, there is no possibility for any anomaly mentioned above to appear. The reason is given in Section 3. Based on that, the examples showed below use simplest version of these anomalies, which has three jobs J_1 , J_2 , J_3 , and each job has only three operations. The operations in each job are (O_{11}, O_{12}, O_{13}) , (O_{21}, O_{22}, O_{23}) and (O_{31}, O_{32}, O_{33}) respectively.

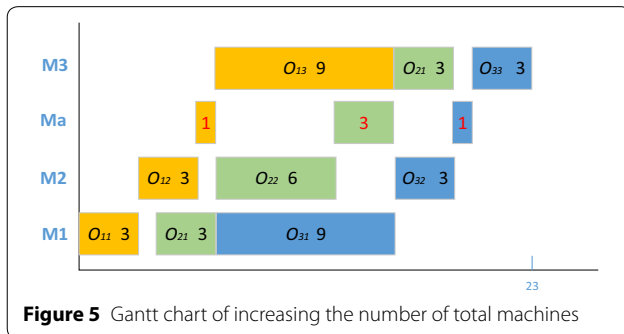
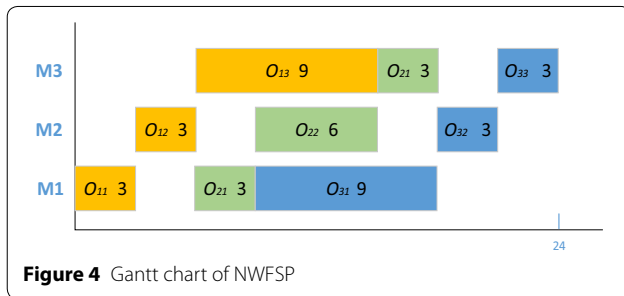
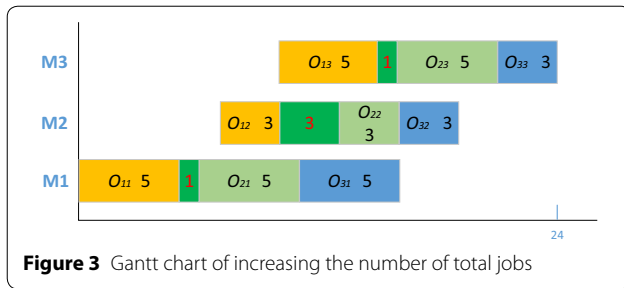
For the type 1 anomaly, it usually appears in NIFSP and NWFSP. This paper only uses the NIFSP to show the type 1 anomaly. The processing time of each operation is considered as (5, 3, 5), (5, 3, 5) and (5, 3, 3). It is easy to know that when the minimization of makespan is regarded as



the objective function, the optimal schedule is J_1, J_2, J_3 , and the value is 25. The Gantt chart is shown in Figure 1. As known by common sense, when the processing time of any operation is increased, it is impossible to reduce the makespan of the optimal schedule. But here comes an interesting thing that when the processing time of operation O_{22} is increased to 5, then the makespan changes from 25 to 23. The Gantt chart is shown in Figure 2. On the contrary, if the schedule of Figure 2 is regarded as the initial one, the makespan will increase if the processing time of operation O_{22} is reduced from 5 to 3. Also, NWFSP has the same anomaly.

For the type 2 anomaly, it usually appears in NIFSP and SFSP. Similarly, only the NIFSP is used to show the type 2 anomaly. The processing time of the operations is the same as before. As known by common sense, when the number of total jobs is increased, it is impossible to reduce the makespan of the optimal schedule. But another interesting thing has happened. When a job, of which processing time is (1, 3, 1) respectively, is inserted between J_1 and J_2 , the makespan changes from 25 to 24. The Gantt chart is shown in Figure 3. On the contrary, if the schedule of Figure 3 is regarded as the initial one, the makespan will increase if the number of total jobs is reduced from 4 to 3. Also, the SFSP has the same anomaly.

For the type 3 anomaly, it usually appears in NWFSP, NBFSP, and SFSP. This paper only uses the NWFSP to show the type 3 anomaly. The processing time of each operation in each job is considered as (3, 3, 9), (3, 6, 3) and (9, 3, 3). It is easy to know that when the minimization of makespan is regarded as the objective function,

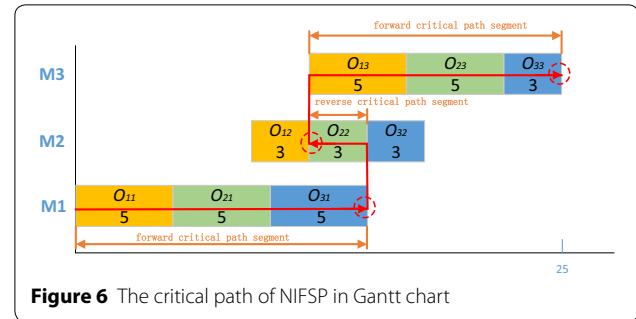


the optimal schedule is J_1, J_2, J_3 , and the makespan is 25. The Gantt chart is shown in Figure 4. From common sense, when the number of total machines is increased (each job increases the number of total operations), it is impossible to reduce the makespan of the optimal schedule. But when Ma is added between M2 and M3, and the processing time of each job on this machine is 1, 3, 1 respectively. The makespan changes from 24 to 23. The Gantt chart is shown in Figure 5. On the contrary, if the schedule of Figure 5 is regarded as the initial one, the makespan will increase if the number of total machines is reduced from 4 to 3. Also, NBFSP and SFSP have the same anomaly.

In conclusion, these three types of anomalies can appear in four kinds of special PFSPs. To express this problem more clearly, the correspondence between shop types and different anomalies is summarized in Table 1.

Table 1 Summarize of anomalies

| | Type 1 | Type 2 | Type 3 |
|-------|--------|--------|--------|
| NWFSP | ✓ | — | ✓ |
| NIFSP | ✓ | ✓ | — |
| NBFSP | — | — | ✓ |
| SFSP | — | ✓ | ✓ |



3 Analysis of Anomalies

3.1 Definitions and Theorems

In order to explore the causes of the anomalies, the critical path in other scheduling problems [25, 26] is expanded. The definitions are added as follows.

Definition 1 The critical path segment from the operation start time to the operation end time is called the forward critical path segment.

Definition 2 The critical path segment from the operation end time to the operation start time is called the reverse critical path segment.

So the critical path of NIFSP is shown in Figure 6.

As it can be easily seen from Figure 6, the completion time of the schedule is the sum of the forward critical path segments minus the sum of the reverse critical path segments. Therefore, it is easy to obtain the following theorems.

Theorem 1 Changing the processing time of the operations on the non-critical paths will not cause the completion time to change until a new critical path is generated.

Theorem 2 Changing the processing time of the operations on the critical path will definitely cause the completion time to change until a new critical path is generated.

Proof The two theorems above can be easily proved by contradiction. Assume that changing the processing time of operations on non-critical paths can cause the

completion time to change. Due to the completion time of the schedule equals to the sum of the forward critical path segments minus the sum of the reverse critical path segments, if the processing time change of one operation can cause the change of the completion time, the operation is on the critical path, which conflicts with the previous assumption. Similarly, assume that changing the processing time of operations on critical paths can not cause the completion time to change. If the processing time change of one operation can not cause the change of the completion time, the operation is on the non-critical path, which conflicts with the previous assumption.

With the Theorem 2, it is easy to draw the following two corollaries.

Corollary 1 *When increasing the processing time of the operations in the forward critical path segment, the completion time will increase until a new critical path is generated; when reducing the processing time of the operations in the forward critical path segment, the completion time will reduce until a new critical path is generated.*

Corollary 2 *When increasing the processing time of the operations in the reverse critical path segment, the completion time will reduce until a new critical path is generated. When reducing the processing time of the operations in the reverse critical path segment, the completion time will increase until a new critical path is generated.*

It can be known that only the change of the processing time of the operations in the reverse critical path segment can cause an anomaly in the special PFSPs. The reverse critical path segment can only be generated in the schedule which has at least three jobs and each job has at least three operations. That is why only two jobs or each job with only two operations can not cause anomalies. Of course, when there are multiple critical paths in a schedule, all the critical paths work together.

3.2 Explanation of Anomalies

For type 1 anomaly, the example mentioned above is used. From Figure 7, only increase the processing time of operation O_{22} , can the completion time be reduced until some other critical path is generated. It is easy to know that the maximum value, used to increase the processing time of the operation O_{22} to reduce the completion time, is related to ε , which show in Figure 7, and it can be calculated as follows:

$$\Delta_{a,b} = \min \left\{ \min \{ \varepsilon_{a,j} \} \left(\text{in which } j = \{1, \dots, b\} \right), \min \{ \varepsilon_{a+1,j} \} \left(\text{in which } j = \{b, \dots, n\} \right) \right\}, \quad (1)$$

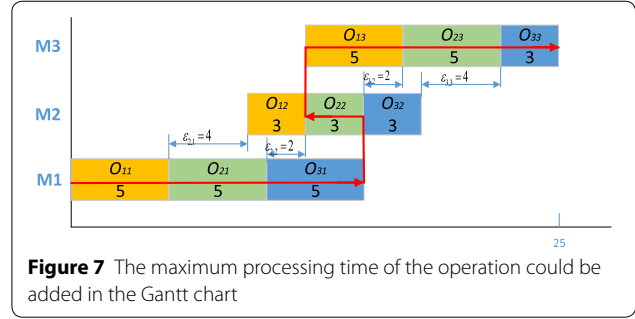


Figure 7 The maximum processing time of the operation could be added in the Gantt chart

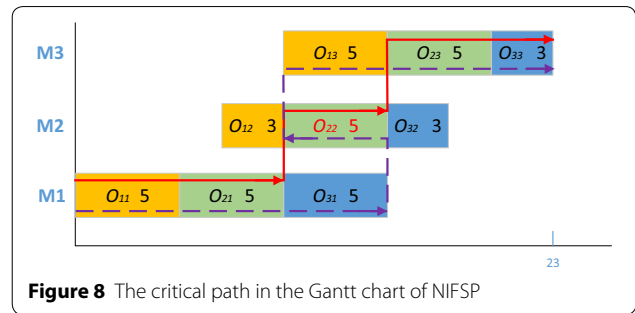
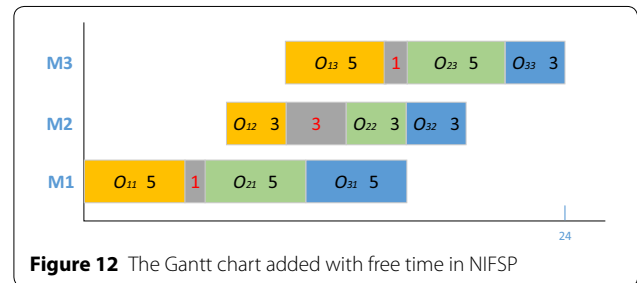
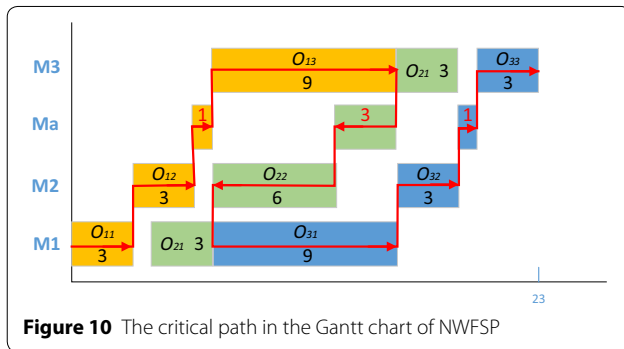
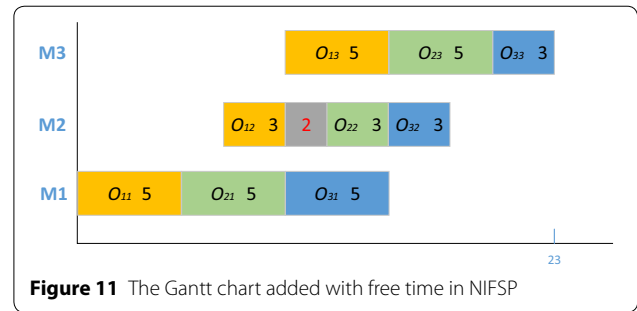
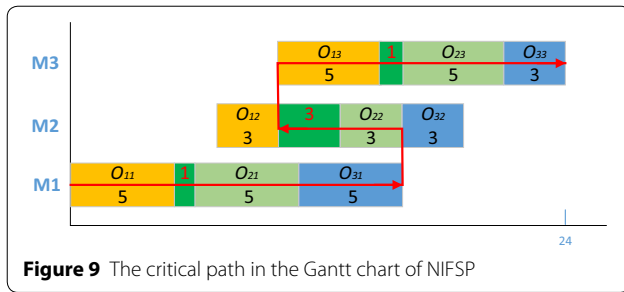


Figure 8 The critical path in the Gantt chart of NIFSP

where n is the number of total jobs; m is the number of total machines; O_{ij} is the i th operation in the j th job; ε_{ij} is the difference between the start time of O_{ij} and the end time of $O_{i,j-1}$; $O_{a,b}$ is the operation on the reverse critical path segment, the a th operation in the b th job; $\Delta_{a,b}$ is the maximum value $O_{a,b}$ could increase to reduce the completion time.

With Eq. (1), it is easy to know that the maximum value increased in the operation O_{22} is 2. After that, a new critical path is generated, shown in Figure 8 with the red solid line. O_{22} changes from the reverse critical path segment to the forward critical path segment. In this case, if it continues to increase the processing time of the operation O_{22} , the completion time will increase. Also, there is another critical path in the Gantt chart, shown in Figure 8 with the purple dotted line. In this critical path, the operation O_{22} is on the reverse critical path segment, but when the processing time increases tiny value, this critical path does not exist.

For type 2 and type 3 anomalies, the examples mentioned above are also used. From Figure 9, the job inserted between $J1$ and $J2$ increases the value of the forward critical path segment and the reverse critical path segment at the same time. The latter increases 3 while the former increases 2, so the completion time reduces by 1. The calculation of the maximum value to reduce the



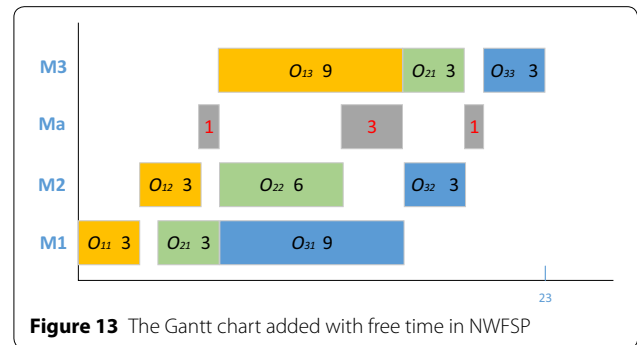
makespan is similar to the type 1 anomaly, and when processing time of the inserted job is (0, 2, 0), the makespan will be the minimum.

From Figure 10, the machine Ma is inserted between M2 and M3. Similar to type 2 anomaly, the value of the forward critical path segment and the reverse critical path segment at the same time. The latter increases 3 while the former increases 2, so the completion reduces by 1. The calculation of the maximum value to reduce the makespan is similar to the type 1 anomaly, and when processing time of the operations in Ma is (0, 2, 0), the makespan will be the minimum.

3.3 Cause of the Anomalies

Knowing from the above analysis, no matter which kind of anomalies, the completion time can be reduced when the value increasing in the reverse critical path segment is more than the value increasing in the forward critical path segment. In other words, if only the value of the reverse critical path segment is increased, the completion time may be reduced the most, just like it mentioned in Section 3.2.

While looking at this problem from another aspect, if the time of the operation, the number of total jobs or the number of total machines is not increased, the completion time can also be reduced when the operation waits for some time instead of being processed immediately. The examples used are shown in Section 3.2, and Figure 11, Figure 12, Figure 13 are generated. The



grey blocks represent the waiting time of the operation, and it is easy to know that it has the same effect as the anomalies.

However, in this case, the constraints of the special PFSPs are not satisfied. In Figure 11, if the operation of O_{22} does not process immediately when the operation O_{12} finishes, the schedule is not the NIFSP. Similarly, in Figure 13, if the operation of O_{13} does not process immediately when the operation O_{12} finishes, the schedule is not the NWFSP. It is also the same situation in NBFSP and SFSP. In another word, the anomalies in special PFSPs can be seen as a way to reduce the completion time by destroying the constraints of the special PFSPs. It is easy to know the lower bound of the anomalies in these special PFSPs is the optimal solution in PFSPs which has the same data with these special PFSPs.

Except for the anomalies in scheduling problems, there are also some anomalies in other problems, like

the transportation anomaly [27–29], the Braess anomaly [30], Belady's anomaly [31] and Graham's multiprocessing anomaly [32]. They may have the same cause with scheduling problems to produce the anomalies.

3.4 Application of the Anomalies

Through the analysis above, it is known that anomalies are caused by destroying constraints, while it is common in real-world production.

Use NIFSP as an example. In some actual manufacture, when the devices are expensive to use, the production model of no-idle is always accepted to reduce the total cost. But it may have a conflict if the product has an urgent due date. In this case, the company should balance the constraints of the production cost and the due date. The NIFSP data above is used to show the application of the type 1 anomaly. Suppose the unit time cost of all machines is 5, and the due date is 20. When the completion time exceeds 20, the compensation for each unit time shall be 10. Then, it is easy to know the cost is $(15+9+13) \times 5 + 10 \times \max\{0, (25-20)\} = 235$. Since the type 1 anomaly appears in NIFSP, the waiting time can be inserted into the scheduling to reduced the completion time, as shown in Figure 11, and the cost is changed to $(15+11+13) \times 5 + 10 \times \max\{0, (23-20)\} = 225$. It is known that the total costs are reduced by using the type 1 anomaly in NIFSP.

In real-world production, the process may be more complicated than the instance above. There may exist more jobs, more machines, and more reverse critical path segments. At this time, if the cost of the devices is different, the company should decide which device to insert free time to get the maximum benefit. On the contrary, the company may consider to speed up some machine, which may increase the makespan under the meets of the due date, to reduce the total cost.

Also in NWFSP, the company should decide how to control the speed of the machine to balance the total cost and the makespan. In NBFSP, the company should decide whether and where to add a buffer area to reduce the makespan. In SFSP, the company should decide when should each operation be processed to get the minimum completion time.

4 Conclusions

This paper summarizes three types of anomalies in the special PFSPs. By using the extended critical path, the rule of anomalies is explained, and that is, when the time of the operation on the reverse critical path changes, anomalies will occur in these special PFSPs. After that, the primary cause of these anomalies is presented. When the processing time of the process increased, or the

number of jobs increased, or the number of machines increased, the constraint in the original special PFSPs is destroyed, which makes the anomalies appear. Finally, this paper points out the application of these anomalies in production practice through examples.

As the essential cause of the anomalies is revealed, there are some research directions could be done at the problem analysis level. For example, is the original optimal scheduling still the optimal solution after increasing the processing time of a certain operation (or inerting the free time to the scheduling), or which one has less completion time when the original optimal solution and the original sub-optimal solution increase the processing time of a certain operation (or inert the free time to the scheduling). Research on these issues will be beneficial to actual production.

Acknowledgements

Not applicable

Authors' contributions

LG was in charge of the whole trial; LG wrote the manuscript; XL assisted with problem analyses. All authors read and approved the final manuscript.

Authors' Information

Lin Gui, born in 1995, is currently a master candidate at *State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, China*. He received his bachelor degree from *Shan Dong University, China*, in 2018. His main research interests are shop scheduling and algorithm optimization.

Liang Gao, born in 1974, is currently a professor at *State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, China*. He received his PhD degree in mechatronic engineering from *Huazhong University of Science and Technology, China*, in 2002. His main research interests are intelligent optimization method and its application in design and manufacturing

Xinyu Li, born in 1985, is currently a professor at *State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, China*. He received his PhD degree in industrial engineering from *Huazhong University of Science and Technology, China*, in 2009. His main research interests are intelligent manufacturing systems, shop scheduling, intelligent optimization and machine learning.

Funding

Supported by National Natural Science Foundation of China (Grant No. 51825502).

Competing interests

The authors declare no competing financial interests.

Received: 1 February 2020 Revised: 23 May 2020 Accepted: 27 May 2020
Published online: 09 June 2020

Reference

- [1] Kunkun Peng, Long Wen, Ran Li, et al. An effective hybrid algorithm for permutation flow shop scheduling problem with setup time. *Procedia CIRP*, 2018: 1288-1292.
- [2] J Gmys, M Mezma, N Melab, et al. A computationally efficient Branch-and-Bound algorithm for the permutation flow-shop scheduling problem. *European Journal of Operational Research*, 2020, 283(3): 814-833.

- [3] Fernandezviagas, Victor, Jose M Molinapariante, Jose M Framinan. Generalised accelerations for insertion-based heuristics in permutation flowshop scheduling. *European Journal of Operational Research*, 2020, 282(3): 858-872.
- [4] Enda Jiang, Ling Wang. An improved multi-objective evolutionary algorithm based on decomposition for energy-efficient permutation flow shop scheduling problem with sequence-dependent setup time. *International Journal of Production Research*, 2019, 57(6): 1756-1771.
- [5] Ning Zhao, Song Ye, Kaidian Li, et al. Effective iterated greedy algorithm for flow-shop scheduling problems with time lags. *Chinese Journal of Mechanical Engineering*, 2017, 30(3): 652-662.
- [6] Allahverdi Ali, Harun Aydilek, Asiye Aydilek. No-wait flow shop scheduling problem with separate setup times to minimize total tardiness subject to makespan. *Applied Mathematics and Computation*, 2020, 365(15): 124688.
- [7] Fuqing Zhao, Huan Liu, Yi Zhang, et al. A discrete water wave optimization algorithm for no-wait flow shop scheduling problem. *Expert Systems With Applications*, 2018, 91: 347-363.
- [8] Xueqi Wu, Ada Che. Energy-efficient no-wait permutation flow shop scheduling by adaptive multi-objective variable neighborhood search. *Omega*, 2019, 94: 102117.
- [9] Weishi Shao, Dechang Pi, Zhongshi Shao. Memetic algorithm with node and edge histogram for no-idle flow shop scheduling problem to minimize the makespan criterion. *Applied Soft Computing*, 2017, 54: 164-182.
- [10] Jingnan Shen, Ling Wang, Shengyao Wang. A bi-population EDA for solving the no-idle permutation flow-shop scheduling problem with the total tardiness criterion. *Knowledge Based Systems*, 2015, 74(1): 167-175.
- [11] Riahi Vahid, Raymond Chiong, Yuli Zhang. A new iterated greedy algorithm for no-idle permutation flowshop scheduling with the total tardiness criterion. *Computers & Operations Research*, 2020.
- [12] Rossit Daniel Alejandro, Fernando Tohmé, Mariano Frutos. The non-permutation flow-shop scheduling problem: a literature review. *Omega*, 2018, 77: 143-153.
- [13] Sana Abdollahpour, Javad Rezaeian. Minimizing makespan for flow shop scheduling problem with intermediate buffers by using hybrid approach of artificial immune system. *Applied Soft Computing*, 2015: 44-56.
- [14] Newton M. A Hakim, Vahid Riahi, Abdul Sattar. Makespan preserving flowshop reengineering via blocking constraints. *Computers & Operations Research*, 2019.
- [15] Stefan Waldherr, Sigrid Knust. Decomposition algorithms for synchronous flow shop problems with additional resources and setup times. *European Journal of Operational Research*, 2017, 259(3): 847-863.
- [16] Stefan Waldherr, Sigrid Knust. Complexity results for flow shop problems with synchronous movement. *European Journal of Operational Research*, 2015, 242(1): 34-44.
- [17] Matthias Bultmann, Sigrid Knust, Stefan Waldherr. Synchronous flow shop scheduling with pliable jobs. *European Journal of Operational Research*, 2018, 270(3): 943-956.
- [18] Abadi, IN Kamal, Nicholas G. Hall, Chelliah Sriskandarajah. Minimizing cycle time in a blocking flow shop. *Operations Research*, 2000, 48(1): 177-180.
- [19] Spieksma Frits CR, Gerhard J Woeginger. The no-wait flow-shop paradox. *Operations Research Letters*, 2005, 33(6): 603-608.
- [20] Kalczynski Pawel Jan, Jerzy Kamburowski. On no-wait and no-idle flow shops with makespan criterion. *European Journal of Operational Research*, 2007, 178(3): 677-685.
- [21] Quanke Pan, Baohua Zhao, Yugui Qu. Heuristics for the no-wait flow shop problem with makespan criterion. *Chinese Journal of Computers*, 2008, 31(7): 1147-1154.
- [22] S Waldherr, S Knust, D Briskorn. Synchronous flow shop problems: How much can we gain by leaving machines idle? *Omega*, 2017, 72: 15-24.
- [23] S S Panwalkar, Christos Koulamas. The evolution of schematic representations of flow shop scheduling problems. *Journal of Scheduling*, 2019, 22(4): 379-391.
- [24] S S Panwalkar, Christos Koulamas. Analysis of flow shop scheduling anomalies. *European Journal of Operational Research*, 2020, 280(1): 25-33.
- [25] Aya Ishigaki, Yuki Matsui. Effective neighborhood generation method in search algorithm for flexible job shop scheduling problem. *International Journal of Automation Technology*, 2019, 13(3): 389-396.
- [26] Dunbing Tang, Min Dai. Energy-efficient approach to minimizing the energy consumption in an extended job-shop scheduling problem. *Chinese Journal of Mechanical Engineering*, 2015, 28(5): 1048-1055.
- [27] G M Appa. The transportation problem and its variants. *Journal of the Operational Research Society*, 1973, 24(1): 79-99.
- [28] David J Robb. The "More for Less" paradox in distribution models: An intuitive explanation. *IIE Transactions*, 1990, 22(4): 377-378.
- [29] Szwarc Włodzimierz. The transportation paradox. *Naval Research Logistics Quarterly*, 1971, 18(1): 185-202.
- [30] Roughgarden Tim, Éva Tardos. How bad is selfish routing? *Journal of the ACM (JACM)*, 2002, 49(2): 236-259.
- [31] Belady Laszlo A, Robert A Nelson, Gerald S Shedler. An anomaly in space-time characteristics of certain programs running in a paging machine. *Communications of the ACM*, 1969, 12(6): 349-353.
- [32] Graham Ronald L. Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*, 1969, 17(2): 416-429.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)