engineering proceedings

MDPI

Proceeding Paper

Application of Tabu Search for Job Shop Scheduling Based on Manufacturing Order Swapping †

Jung-Sing Jwo [1], Cheng-Hsiung Lee [1], Jian-Tan Chen [1], Ching-Sheng Lin [1,*], Chun-Yu Lin [2], Wen-Kai Cheng [2], Chun-Hao Chang [2] and Jen-Kai King [2]

[1] Master Program of Digital Innovation, Tunghai University, Taichung 40704, Taiwan
[2] Aerospace Industrial Development Corporation, Taichung 407803, Taiwan
* Correspondence: cslin612@thu.edu.tw
† Presented at the IEEE 5th Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability, Tainan, Taiwan, 2–4 June 2023.

Abstract: Due to the characteristics of small-volume-large-variety production in the aircraft industry, determining how to intelligently take men, machines, materials, methods, and environments into account is critical for production, posing the challenge of scheduling the job shop. The existing scheduling system is slow to adapt when the requirements change and, most importantly, slow to consider the current situation of the production line. In this study, we apply the Tabu search algorithm to resolve the scheduling problem. This method swaps manufacturing orders with the consideration of the due date, working hours, and current delays via the finite capacity assumption. Meanwhile, we set up extreme bounds using the infinite capacity assumption for comparison. The case study is conducted in a work center that uses pliers to trim parts for use in the aircraft industry. The results show the closeness to the extreme bounds and indicate this method's practical feasibility for industrial applications.

Keywords: job shop scheduling; Tabu search; manufacturing orders; extreme bounds; aircraft industry

check for updates

Citation: Jwo, J.-S.; Lee, C.-H.; Chen, J.-T.; Lin, C.-S.; Lin, C.-Y.; Cheng, W.-K.; Chang, C.-H.; King, J.-K. Application of Tabu Search for Job Shop Scheduling Based on Manufacturing Order Swapping. *Eng. Proc.* **2023**, *55*, 51. https://doi.org/10.3390/engproc2023055051

Academic Editors: Teen-Hang Meen, Kuei-Shu Hsu and Cheng-Fu Yang

Published: 5 December 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the rapid development of Internet of Things (IoT) technology, the concepts and implementations of smart manufacturing have been widely studied. Since the production of composite materials is a crucial manufacturing process in the aircraft industry, it is important to introduce smart manufacturing-related technologies into current manufacturing processes. Properly arranging the manufacturing orders (MOs) of the jobs is the key to increasing the efficiency of product life cycles and successfully moving toward smart manufacturing. The main characteristic of processes in the aircraft industry is small-volume-large-variety production, posing the challenge of scheduling the job shop.

Job shop scheduling is considered to be combinatorial optimization and belongs to the class of NP-hard problems [1]. Many different methods have been explored in relation to this research topic, including analytical algorithms and artificial intelligence-based methods [2]. Queuing theory and simulation models are popular in analytical algorithms. Artificial intelligence-based optimization techniques mainly include Tabu search and genetic algorithms. In this study, we apply the Tabu search algorithm to address the job shop scheduling by swapping MOs. Using the finite capacity assumption, we consider the due date, working hours, and current delays for each job. To evaluate the effectiveness of the proposed solution, we compute extreme bounds based on the infinite capacity assumption to determine how far the proposed method is from optimality. The empirical study is conducted in a work center which uses pliers to trim parts for use in the aircraft industry.

The rest of this paper is structured as follows. Section 2 reviews several prior studies. Section 3 discusses the proposed scheduling method. The experiments are described in Section 4 and the conclusions are provided in Section 5.

2. Literature Review

Prior studies related to this paper are discussed with regard to Tabu search-based approaches and AI-related models.

2.1. Tabu Search Overview

Tabu search was created by Glover in 1986 for optimization problems [3,4]. There are two properties, local search and the Tabu list, in the Tabu search that enhance its optimization ability. The local search technique includes movements which do not improve the performance if no better moves are found. These worse but acceptable moves help the optimization process escape from local optima by exploring the solution space more thoroughly. The Tabu list is a form of short-term memory that records the most recently visited moves and discourages the Tabu search from revisiting them to prevent cycles [5]. To generate more neighborhoods, a method to resolve the distributed permutation flow-shop scheduling problem (DPFSP) is proposed, which involves swapping the sub-sequences of jobs [6]. A parallel computing approach based on Tabu search is used to speed up the computation time taken to resolve the blocking job shop scheduling problem [7]. The experiment was conducted on a cluster-based supercomputer using 512 CPU cores.

2.2. Other AI Approaches

Due to the booming development of deep learning techniques, neural network-based approaches have also been investigated in relation to the scheduling problem. Actor–Critic is a model-free reinforcement learning technique in which the actor network is taught how to behave in given environments, whereas the critic network learns to give feedback to the actor network [8]. An actor–critic deep reinforcement learning model is proposed to solve the job shop scheduling problem, in which the actor network assigns a job and the critic network provides rewards according to the processing time of the job [9]. A Deep Q-Network (DQN) is adopted for semiconductor manufacturing scheduling by using an agent to perform job assignments for each work center [10]. Another DQN method based on edge computing is proposed to solve the scheduling problem for a smart semiconductor manufacturing factory [11]. Unlike the traditional DQN, which supports only one decision, this method outputs multiple dispatching rules for multiple edge devices. To better handle the highly dynamic and changeable conditions in the factory, a dueling double DQN is designed to deal with the adaptive job shop scheduling problem by training a CNN to approximate the state-action value function [12].

3. Scheduling by Manufacturing Order Swapping

The proposed Tabu search scheduling based on manufacturing order swapping is introduced to discuss the research problem, the concept of the proposed solution, and the objectives. The detailed algorithm used for Tabu search scheduling is also described in the second section.

3.1. Problem Definition

We focus on scheduling one of the work centers in the aircraft process. For each day, there are new MOs to be dispatched. The unprocessed MOs from the previous day, they are combined with new MOs for the current day for dispatch. The initial sorting of MOs each day is performed using First in First out (FIFO) and Earliest Due Date (EDD) processes. The notations are defined as follows.

WO: working hours per day for the current work center (CWC).
i: MO, i = 1, 2, . . .I.
PT(i): processing time (scaled by hours) for MO i.

LT(i): average lead time (scaled by days) for MO i after CWC.
SD(i): the scheduled day (noted by 1, 2, 3, . . .) for the MO i where 1 means today, 2 means tomorrow, and so on.
DD(i): due date for MO i.
BPDD(i): number of days the MO i before or past the due date at the CWC by computing (Today-DD(i)).
EDD_FCA(i): expected delay days for MO i under finite capacity assumption by computing BPDD (i) + SD(i) + LT(i) where a positive value represents a delay and a negative value is one that occurs before the start of the schedule.

Due to the specific production requirements of composite materials in the aerospace industry, there are three minimum optimization problems to overcome: (i) total expected early completion days (TEECD), (ii) total expected delay days (TEDD), and (iii) total number of delayed manufacturing orders (TNDMO). For each MO i, we calculate the EDD_FCA(i). The TEECD is defined as the total sum of those EDD_FCA which are smaller than 0, as in "(1)", whereas TEDD is defined as the total sum of those EDD_FCA which are greater than 0, as in "(2)". The TNDMO is the number of MOs which have EDD_FCA greater than 0, as in "(3)". Another notation is the total number of early completion manufacturing orders (TNECMO), as in "(4)". After each schedule, we assign SD(i) by accumulating PT(i) and considering WO. The estimated values of TEECD, TEDD, and TNDMO are calculated as well.

$$\text{TEECD} = \sum_{i=1}^{I} \text{EDD_FCA}(i), \text{where EDD_FCA}(i) < 0 \quad (1)$$

$$\text{TEDD} = \sum_{i=1}^{I} \text{EDD_FCA}(i), \text{where EDD_FCA}(i) \geq 0 \quad (2)$$

$$\text{TNDMO} = |\{i | \text{EDD_FCA}(i) \geq 0 \forall i\}| \quad (3)$$

$$\text{TNECMO} = |\{i | \text{EDD_FCA}(i) < 0 \forall i\}| \quad (4)$$

The workflow of the scheduling process is shown in Figure 1. For each daily scheduling, we set the initial sorting (FIFO or EDD) for MOs. The Tabu search is applied to the schedule to compute TEECD, TEDD, and TNDMO. These values are used to evaluate the effectiveness of the method. Those MOs that are unable to be processed are moved to the next day for further scheduling.

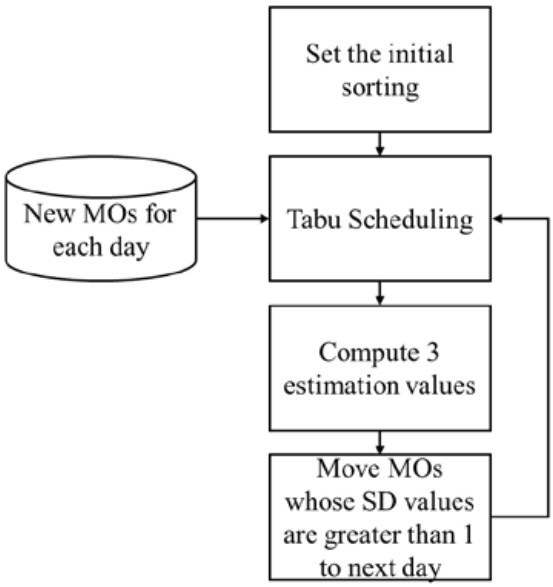

Figure 1. Process of scheduling.

3.2. Scheduling Algorithm

The concept of the Tabu search scheduling is to switch the MOs and obtain the best TEECD, TEDD, and TNDMO. To be specific, for each MO i with EDD_FCA$(i) \geq 0$, we switch i with all other MOs and re-calculate SD to obtain new EDD_FCA. With the new EDD_FCA, we compute new TEECD, TEDD, and TNDMO for comparison. The switch frequencies are TNDMO $\times$ (TNDMO + TNECMO $-$ 1). The above process operates iteratively.

The details of the scheduling algorithm are shown in Algorithm 1. The input S_init is an initial sorting by FIFO or EDD. The variable assignment occurs from lines 1 to 5, where sBest denotes the overall best scheduling and iterBest denotes the best scheduling in each iteration. Each day, we perform an N iteration to find the best schedule. The function getNeighbors is used to find out all possible schedules by swapping the elements in setDelay with the elements in set_All and storing all possible schedulings into sNeighborhood (lines 7–10). Then, we loop over each scheduling in sNeighborhood to find the best scheduling for the given iteration (lines 11–16). The function checkMinOptimization checks whether the current scheduling sCandidate has lower TEECD, TEDD, and TNDMO than iterBest. If the current schedule already exists in the tabuList, we skip the comparison. In the next step, we use the same function to compare iterBest with sBest (lines 17–18). Lastly, iterBest is put into tabuList to avoid being trapped in the local optimum (lines 19–20).

Algorithm 1: Tabu Search Scheduling by Manufacturing Order Swapping.

Input: S_init which is an initial scheduling by FIFO or EDD

```
1:  sBest ← S_init
2:  iterBest ← S_init
3:  tabuList ← []
4:  tabuList.push(S_init)
5:  iteration ← N
6:  while iteration:
7:    set_Delay = { MO i | EDD_FCA(i) ≥ 0 ∀ i based on iterBest}
8:    set_All = {MO i based on iterBest}
9:    #Get all possible schedulings based on iterBest
10:   sNeighborhood ← getNeighbors(set_Delay, set_All)
11:   #Find the best scheduling in this iteration
12:   for (sCandidate in sNeighborhood)
13:       if (not tabuList.contains(sCandidate))
14:           iterBest ← checkMinOptimization(sCandidate, iterBest)
15:       end
16:   end
17:   #Is iterBest better than sBest?
18:   sBest ← checkMinOptimization(iterBest, sBest)
19:   #Put iterBest into tabuList to avoid trapped in the local optimum
20:   tabuList.push(iterBest)
21: End
```

4. Experimental Results

For each day, there are new manufacturing orders (NMO) sent to the work center: total manufacturing orders (TMO), finished manufacturing orders (FMO), finished but delayed manufacturing orders (FDMO), and total number of delayed manufacturing orders (TNDMO). The TMO is the NMO combined with the previous unprocessed MOs. The FMO represents the number of MOs which finish on that day, while FDMO indicates those FMO that pass the due dates. Based on the scheduling, TNDMO is used to represent the total number of MOs that pass the due dates.

To obtain evaluation metrics with which to assess the approach, we introduce three bounds including UBTEECD, LBTEDD and LBTNDMO. The assumption is based on infinite capacity, which implies all new MOs are finished on one day without carrying

over to the next day. The UBTEECD is used to represent the maximum of total expected early completion days, LBTEDD represents the minimum of total expected delay days, and LBTNDMO represents the minimum total number of delayed manufacturing orders. As mentioned before, the minimum of TEECD is considered due to the special requirements of composite materials.

In this empirical study, five days of scheduling are presented. Since the total number of delayed manufacturing orders is the most critical factor in the factory, we start with the evaluation of TNDMO. Both FIFO and EDD are used for initial scheduling. The results are shown in Tables 1 and 2. The scheduling approach with FIFO initialization reaches the LBTNDMO every day, whereas EDD initialization is not promising. More detailed evaluations of TEECD, TEDD, and TNDMO with FIFO and EDD initializations are also conducted. Table 3 demonstrates the FIFO scheduling and the proposed scheduling with FIFO initialization. Directly applying FIFO for scheduling does not provide good results, while the proposed method achieves good performance in TEDD and TNDMO. In Table 4, we employ EDD and the proposed method with EDD initialization for comparison. Directly applying EDD for scheduling does not achieve good performance either, but the FIFO scheduling outperforms EDD scheduling. Similarly, the proposed method with FIFO initialization is better than EDD initialization. In conclusion, the proposed method works better in terms of TEDD and TNDMO, though there is still room to improve the values of the TEECD.

Table 1. Tabu search with initial scheduling via FIFO compared with the LB_{TNDMO}.

	Day1	**Day2**	**Day3**	**Day4**	**Day5**
NMO	369	66	94	123	121
TMO	369	393	402	508	594
FMO	42	85	17	35	56
FDMO	34	54	9	34	48
TNDMO	63	61	21	40	50
LB_{TNDMO}	63	61	21	40	50

Table 2. Tabu search with initial scheduling via EDD compared with the LB_{TNDMO}.

	Day1	**Day2**	**Day3**	**Day4**	**Day5**
NMO	369	66	94	123	121
TMO	369	397	451	557	655
FMO	38	40	17	23	48
FDMO	27	37	13	20	41
TNDMO	71	81	53	65	84
LB_{TNDMO}	63	68	45	60	84

Table 3. Comparison results between the FIFO, our method and bound (% is the comparison with the bound).

	Day1			Day2			Day3			Day4			Day5		
	TEECD	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**
FIFO	4524 (−12%)	1230 (18%)	86 (37%)	5114 (−11%)	1029 (17%)	67 (10%)	5869 (−12%)	572 (24%)	30 (43%)	7623 (−14%)	883 (17%)	43 (8%)	9144 (−13%)	876 (34%)	54 (8%)
Our method	4312 (−16%)	1079 (3%)	63 (0%)	4950 (−14%)	898 (2%)	61 (0%)	5509 (−18%)	475 (3%)	21 (0%)	7420 (−16%)	758 (1%)	40 (0%)	8737 (−17%)	655 (0%)	50 (0%)
Bound	5113	1043	63	5739	882	61	6682	463	21	8877	752	40	10510	653	50

Table 4. Comparison results between the EDD, our method and bound (% is the comparison with the bound).

	Day1			Day2			Day3			Day4			Day5		
	TEECD	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**	**TEECD**	**TEDD**	**TNDMO**
EDD	4193 (−18%)	1154 (11%)	95 (51%)	4704 (−17%)	1146 (13%)	97 (43%)	5595 (−19%)	751 (28%)	80 (78%)	7488 (−18%)	819 (18%)	90 (50%)	8877 (−17%)	939 (24%)	109 (30%)
Our method	4399 (−14%)	1131 (8%)	71 (13%)	4669 (−18%)	1085 (7%)	81 (19%)	5511 (−20%)	657 (12%)	53 (18%)	7428 (−18%)	860 (8%)	65 (5%)	8779 (−18%)	814 (7%)	84 (0%)
Bound	5113	1043	63	5668	1015	68	6893	585	45	9088	692	60	10721	760	84

5. Conclusions

We present a Tabu search schedule based on swapping the manufacturing orders. We conduct empirical studies in the aerospace industry with FIFO and EDD initializations. The proposed approach achieves good performance with regard to minimizing the total expected delay days and the total number of delayed manufacturing orders. In future work, we will focus on the improvement of the total expected early completion days and also expand our method to different work centers to evaluate its effectiveness and robustness.

Author Contributions: Supervision, J.-S.J.; methodology, J.-T.C., C.-S.L. and C.-H.L.; investigation, J.-S.J., J.-T.C., C.-Y.L., W.-K.C., C.-H.C. and J.-K.K.; writing—review and editing, C.-S.L.;. All authors have read and agreed to the published version of the manuscript.

Funding: This work is financially supported by the Ministry of Science and Technology (MOST) of Taiwan under Grant MOST 110-2622-E-029-015 - .

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

References

1. Garey, M.R.; Johnson, D.S.; Sethi, R. The complexity of flowshop and jobshop scheduling. *Math. Oper. Res.* **1976**, *1*, 117–129. [CrossRef]
2. Zhang, Z.; Wang, W.; Zhong, S.; Hu, K. Flow shop scheduling with reinforcement learning. *Asia-Pac. J. Oper. Res.* **2013**, *30*, 1350014. [CrossRef]
3. Glover, F. Tabu search—Part I. *ORSA J. Comput.* **1989**, *1*, 190–206. [CrossRef]
4. Navarro, A.; Rudnick, H. Large-scale distribution planning—Part II: Macro-optimization with Voronoi's diagram and tabu search. *IEEE Trans. Power Syst.* **2009**, *24*, 752–758. [CrossRef]
5. Caldeira, R.H.; Gnanavelbabu, A.; Joseph Solomon, J. Solving the Flexible Job Shop Scheduling Problem Using a Hybrid Artificial Bee Colony Algorithm. In *Trends in Manufacturing and Engineering Management*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 833–843.
6. Gao, J.; Chen, R.; Deng, W. An efficient tabu search algorithm for the distributed permutation flowshop scheduling problem. *Int. J. Prod. Res.* **2013**, *51*, 641–651. [CrossRef]
7. Dabah, A.; Bendjoudi, A.; AitZai, A.; Taboudjemat, N.N. Efficient parallel tabu search for the blocking job shop scheduling problem. *Soft Comput.* **2019**, *23*, 13283–13295. [CrossRef]
8. Konda, V.; Tsitsiklis, J. Actor-critic algorithms. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 29 November–4 December 1999; Volume 12.
9. Liu, C.L.; Chang, C.C.; Tseng, C.J. Actor-critic deep reinforcement learning for solving job shop scheduling problems. *IEEE Access* **2020**, *8*, 71752–71762. [CrossRef]
10. Waschneck, B.; Reichstaller, A.; Belzner, L.; Altenmüller, T.; Bauernhansl, T.; Knapp, A.; Kyek, A. Deep reinforcement learning for semiconductor production scheduling. In Proceedings of the 2018 29th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC), Saratoga Springs, NY, USA, 30 April–3 May 2018; pp. 301–306.
11. Lin, C.C.; Deng, D.J.; Chih, Y.L.; Chiu, H.T. Smart manufacturing scheduling with edge computing using multiclass deep Q network. *IEEE Trans. Ind. Inform.* **2019**, *15*, 4276–4284. [CrossRef]
12. Han, B.A.; Yang, J.J. Research on adaptive job shop scheduling problems based on dueling double DQN. *IEEE Access* **2020**, *8*, 186474–186495. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.