

THREE-MACHINE JOB SHOP SCHEDULING WITH INTERMEDIATE TRANSFER

Li, Y.^{*,**}; Shi, S. Y.^{*} & Huang, Q. D.^{*,#}

^{*} College of Mathematics, Jilin University, Changchun 130000, China

^{**} Jilin Institute of Chemical Technology, Jilin 132022, China

E-Mail: yan15@mails.jlu.edu.cn, shisy@jlu.edu.cn, huangqd@jlu.edu.cn ([#] Corresponding author)

Abstract

With the purpose of minimizing the makespan of three-machine job shop scheduling problem with intermediate transfer, in this study, a new model was built to perform the makespan minimization, which was proved to be strongly NP-hard. From this model a heuristic algorithm was subsequently derived with a worst-case error bound. The proposed model and derived algorithm were verified through computational experiments carried out on a couple of small size random instances. The results show that the heuristic algorithm possesses a tight worst-case bound of 2. This research presents an approach for obtaining the optimal solutions of the NP-hard problem within a reasonable time. (Received, processed and accepted by the Chinese Representative Office.)

Key Words: Job Shop Scheduling, Heuristic Algorithm, Worst-Case Performance, Coordinated Scheduling

1. INTRODUCTION

Intermediate transfer and final delivery are reported as two critical phases through the job transport. The former transfer the semi-finished jobs from one machine to another for subsequent processes, and the latter delivers the completed jobs to the customers [1]. Both phases have been paid much attention by industry and academia. In order to save cost, jobs are commonly transported in batches rather than individually, which makes the batch capacity a major constraint on transport. Considering the intermediate transfer and final delivery, Cheng et al. minimized the total transport cost and total earliness penalty of batch job in a single-machine job shop. With the aim to minimize the makespan, Lee and Chen discussed the conditions under which all jobs are of the same size and transported in a limited batch capacity and fixed transfer time. The transport of jobs with different sizes in a single-machine job shop and a two-parallel machines job shop was investigated by Chang and Lee. In their work, a polynomial-time approximation algorithm with a worst-case performance ratio of $5/3$ was presented. Given the same conditions, Zhong et al., developed an approximation algorithm of which the worst-case performance ratio reaches $3/2$ [2-8].

The intermediate transfer mainly takes place in the job shop, for semi-finished jobs in some manufacturing systems (e.g. steelmaking system) need to go through multiple production stages. Maggu and Das examined a two-machine job shop where several transporters are available to move jobs from one machine to the other. Chen and Lee proved that two-machine job shop scheduling is either an NP-hard problem or has an accurate polynomial-time solution. For the minimal makespan, Lee and Strusevich put forward a best possible approximation algorithm that transports jobs in two batches where a single interstage transporter with unlimited capacity was considered. Tang and Liu explored job shop scheduling problems where the jobs are transported via a batcher and a transporter. To obtain the minimization of the makespan, the strongly NP-hard problems were simulated by employing a mixed integer programming model and computing experiments were performed, from which a heuristic algorithm and a worst-case error constraint were derived. The makespan minimization of the two-machine job shop scheduling for a single transporter with

a limited capacity was reported in the work of Gong and Tang, where a heuristic algorithm was proposed with a worst-case ratio of 2 and $7/3$ respectively for two scenarios: each job occupies the same space and different spaces of the transporter, respectively [9-19]. For the latter case, the asymptotic worst-case ratio of $20/9$ was obtained.

Based on the aforementioned works, this study focus mainly on a three-machine job shop scheduling problem, where each job occupies a specific space during the intermediate transfer among the three machines. During the transfer, the jobs are transported in batch on two transporters with a limited space capacity, i.e., the total space of jobs being transported simultaneously cannot surpass the space limit of each transporter. The purpose of this research is to minimize the time cost for processing all jobs on the next machine through rational sequencing and batching of the jobs on two adjacent machines. The said problem is detailed as follows. M_1 , M_2 and M_3 are supposed to be the three machines, respectively, and $N = J_1, J_2, \dots, J_n$ are assumed to be the set of jobs. There are three steps in processing each job J_i : (1) processing on M_1 for p_{1j} units of time; (2) transfer to and processing on M_2 for p_{2j} units of time, which are not predetermined; (3) transfer to and processing on M_3 for p_{3j} units of time. Each transporter has a fixed capacity c which is evaluated by the total space of the jobs it can transfer in one batch. The capacity equals to the batch processing capacity. Despite the identical capacity, the two transporters may differ in transport duration. The transport durations of them are denoted as T_1 and T_2 , respectively. The objective of the target problem is to estimate the minimization of the makespan, C_{max} , i.e., the maximum time cost for completing the jobs on M_3 . Following Pinedo's three-field notation [20], our problem can be expressed as $TF_3||C_{max}$, which includes several variations of the cases studied in Lee and Chen.

The scheduling problem for two-machine job shop stems from the burn-in operation, one of the five main phases in semiconductor manufacturing. In the first two phases (referred to as wafer fabrication and testing), the integrated circuits are developed, tested and invaded individual circuits, so that the ones with defection could be discarded. Then comes the assembly phase, in which the leads are attached to each individual circuit and paled in a plastic or ceramic product. The assembly phase could be regarded as a single-machine case that processes one circuit at a time. After that, the chips are transferred in batch to receive the operation of burn-in. Within the burn-in, each chip is subjected to thermal stressing in a constant temperature oven for a period of time defined by the customer. Of course, the time used for burn-in should not exceed the designed value for the chip. The batch processing time is equal to the maximum time cost among all the chips in the batch. The last phase is branding: the manufacturer's logo and the product serial number are stamped on the completed circuits. This phase can also be viewed as a single machine [21-31].

The remaining part of this paper is organized as: Section 2 describes the target problem and gives some preliminary results; Section 3 puts forward a fast heuristic algorithm specific to the target problem; Section 4 discusses the worst-case performance of the proposed algorithm through computational experiments; Section 5 wraps up this research with some meaningful conclusions.

2. PROBLEM DESCRIPTION AND PRELIMINARIES

Below is a formal description of our problem. As mentioned before, any job J_i in the job set $N = J_1, J_2, \dots, J_n$ should be firstly processed on M_1 for an unspecified period of time, secondly be transferred to M_2 via a transporter for further processing, and finally be transferred to M_3 by the other transporter for final processing. All jobs are assumed to be available for processing at time 0. The time cost for processing job j on machine M_i ($i = 1, 2, 3$) are denoted as p_i ($i = 1, 2, 3$). Let s_j be the size of job j , i.e. the transporter space occupied by the job. The

total size of all jobs in a transport batch is not allowed to exceed the capacity of the transporter c , which can be expressed as the number of jobs, provided that all jobs share the same size. The two transporters are of identical capacity, but differ in transfer time. The transfer time of the two transporters are denoted as T_1 and T_2 , respectively. Moreover, our problem is subjected to the following two preliminaries.

Preliminary 2.1: As for a given job let $N = J_1, J_2, \dots, J_n$ and a three-machine job shop, there exists an optimal schedule satisfying the following conditions:

- (1) Jobs are consecutively processed on machine M_1 and idle time is not permitted.
- (2) Within the same batch, jobs transported are processed consecutively without idle time on three machines.

The resulting schedule is optimal pertaining to the minimization of the makespan for the classical two-machine job shop scheduling problem [32, 33]. In the classical problem, there is at most one job at a time on each machine. Thus, the scheduling structure is a different way from that of two-machine problem. Johnson suggested that it is adequate to consider permutation schedules for the three-machine problem.

Preliminary 2.2: The optimal sequence is that the n items on each machine obey the same sequence. In other words, the sequence on the first and third machines should be the same as that of the second machine.

3. THREE-MACHINE JOB SHOP SCHEDULING CONSIDERING INTERMEDIATE TRANSFER

This section mainly discusses the computational complexity of the problem $TF_3|s_j=1|C_{max}$ in which each job occupies same space during transport, i.e. $s_j = 1$, and each transporter can handle with up to c jobs in one batch. Based on the discussion, the author presented a heuristic algorithm specific to the problem.

3.1 Strong NP-hardness

Theorem 3.1: Three-machine job shop scheduling considering intermediate transfer is strongly NP-hard.

The theorem proof starts from a reduction of the following 3-partition problem being strongly NP-hard.

Suppose item $j \in H$ ($H = 1, 2, \dots, 3h$) has a positive integer size a_j subject to $a/4 < a_j < a/2$, $1 \leq j \leq 3h$ and $\sum_{j=1}^{3h} a_j = ha$ for an integer a . The problem aims to determine the existence or absence of disjoint subsets h, H_1, H_2, \dots, H_h , such that $|H_1| = |H_2| = \dots = |H_h| = 3$ and $\sum_{j \in H_i} a_j = a, i = 1, 2, \dots, h$.

For the target problem, an instance was created below: $N = 3h, c = 3, T = p = 2a, p_j = 2a_j, j = 1, 2, \dots, 3h$, and the threshold value is obtained by $y = (3h+3)a$. Then, the rest is to prove that when and only when there exists a feasible solution to the constructed instance with a makespan not greater than y the 3-partition problem has a solution.

(Necessity) If a solution to the 3-partition problem exists, there must be a schedule to the target problem whose makespan is not greater than y . For a given solution to the 3-partition problem (H_1, H_2, \dots, H_h) , a schedule can be constructed for the target problem, in which a transporter leaves from the batching machine to transfer the jobs in $J_j|j \in H_i$ to the single machine where the processing starts at time $3a$. The last job is predicted to be finished on the single machine at time $y = (3h+3)a$. Therefore, the above mentioned schedule could be considered feasible and the makespan is y .

Proof by contradiction: Supposing that a schedule specific to the target problem with the makespan not greater than $y = (3h+3)a$ exists. Owing to the limited batching machine capacity,

the minimum account of batches is h on that machine. In general, it can be assumed that there are $h+1$ batches if the number of batches, h , is exceeded in the schedule. Then, the total time cost for processing batches on the machine is $(h+1)3a$ and the one-way transfer time cost on the last batch is a . However, this is a conflicting result because all jobs in the last batch should be considered to be handled on a single machine. Therefore, it can be seen that the schedule is exactly composed of h batches on the batching machine. The batches are denoted as H_1, H_2, \dots, H_h each of which includes exactly three jobs. The total cost time on the single machine is equal to $3ha$, and the earliest possible time starting to process the job on the single machine is $P+T/2=3a=y-3ha$, which means that on the single machine idle time in the schedule during $[3a, (3h+3)a]$ is not allowed and three jobs are carried by each transporter at a time period $[a, (h+1)a]$ without idleness. This in turn proves that, for $i=1, 2, \dots, h$, the processing of the batch $\{J_j|j \in H_i\}$ starts at time $(3i+1)a$ and completes at time $(3i+3)a$ on the single machine. Thus, $\sum_{j \in H_i} p_j = 2a$ with $i=1, 2, \dots, h$, which reveals that $\sum_{j \in H_i} a_j = a$ with $i=1, 2, \dots, h$ is a solution specific to the 3-partition problem. The combination of the if part and the only if part is able to prove the validation of theorem.

Corollary 1: Problem $TF_3|s_j=1|C_{max}$ is strongly NP-hard.

3.2 Three-machine production schedule

Respect to a strongly NP-hard problem $TF_3|s_j=1|C_{max}$, this study proposed a heuristic algorithm whose upper bound is on the worst case performance ratio. The construction of the algorithm is based on the principle of Johnson's algorithm.

Algorithm 3.1 (Heuristic algorithm)

Step 1: Split the jobs into three sets A_1, A_2 and A_3 , with $A_1 = \sum_{j \in p_i} p_{1j}$ containing all the jobs satisfying $A_1 = \{p_{ij} | p_{1j} \leq p_{2j}\}$, $A_2 = \{p_{ij} | p_{1j} \geq p_{2j} + T_1/2\}$ and $A_3 = \{p_{ij} | p_{2j} \leq p_{1j} \leq p_{2j} + T_1/2\}$ considering $i=1, 2$.

Step 2: Rank the jobs in $A_1 = \sum_{j \in p_i} p_{1j}$ in ascending order on M_1 . Further divide the jobs in A_3 into two sets B_1 , and B_2 , with $B_1 = \{p_{ij} | p_{2j} \leq p_{1j} \leq p_{2j} + T_1/2, p_{3j} \leq p_{2j}\}$, $B_2 = \{p_{ij} | p_{2j} \leq p_{1j} \leq p_{2j} + T_1/2, p_{2j} \leq p_{3j}\}$.

Step 3: Rank the jobs in B_1 in ascending order of p_{3j} on M_3 . Further divide the jobs in B_2 into two sets C_1 and C_2 , with $C_1 = \{p_{ij} | p_{2j} \leq p_{1j} \leq p_{2j} + T_1/2, p_{2j} + T_2/2 \leq p_{3j}\}$ and $C_2 = \{p_{ij} | p_{2j} \leq p_{1j} \leq p_{2j} + T_1/2, p_{2j} < p_{3j} < p_{2j} + T_2/2\}$. Similarly, rank the jobs in C_2 in ascending order of d_t denoting the sum of the absolute difference of time cost for processing job t on the first and second machines and that cost for dealing with job t on second and third machines, i.e. $d_t = |p_{1t} - p_{2t}| + |p_{2t} - p_{3t}|$.

Step 4: Arrange A_1, A_2, B_1, C_1 and C_2 in a chain $A_1 < B_1 < C_2 < A_2 < C_1$ in a manner that each job in A_1 precedes its corresponding job in any other set and each job in B_1 precedes its corresponding job in any other set except A_1 . The rest may be deduced by analogy.

Step 5: Create an empty batch R_1 , assign job P_j in A_1, B_1 and C_2 to batch R_{1k} , where $k = \text{argmin}_{l=1, 2, \dots, H} \{c \text{ jobs are not included in } P_l | R_l\}$, and let $P_k \leftarrow P_k + P_j, j=1, 2, \dots, n$.

Step 6: Assign the jobs of A_2 and C_1 into separate batches: place the first jobs of A_2 and C_1 into batch $H+1$, the second jobs of A_2 and C_1 into batch $H+2$, and so on. Perform the above procedure until at least one of A_2 and C_1 become empty. Let h batches be the new batches of the jobs in A_2 and C_1 , so that $h = \min_{i=1, 2} b_i + |b_1 - b_2|$.

Step 7: Rank the $H+h$ batches on the basis of Johnson's rule for the artificial three-machine job shop problem, where batch P_j is a job whose processing time are A_i, B_i and C_i in M_1, M_2 and M_3 , respectively. The final sequence of processing batch on all three machines is $\pi = \{R_1, R_2, \dots, R_H, R_{H+1}, \dots, R_{H+h}\}$. Rank the batches on three machines by the order π . Based on the order, each transporter delivers one batch at a time.

3.3 Application of algorithm to a numerical example

The application of Algorithm 3.1 was implemented using a numerical example which was constructed with the number of jobs $n = 10$, the batching machine capacity $c = 3$ and transfer time of the transporters $T_1 = 6$ and $T_2 = 4$. Table I lists the time cost for processing jobs on each machine.

Table I: The processing time of jobs on each machine.

Job	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9	J_{10}
p_{1j}	3	5	4	6	5	2	5	3	3	3
p_{2j}	5	2	3	3	5	5	4	1	2	3
p_{3j}	4	3	3	1	4	3	3	2	4	5

According to Steps 1 and 2 the jobs should be firstly categorized. The sorted sets are $A_1 = [J_1, J_6]$, $A_2 = [J_4]$, $B_1 = [J_3, J_5, J_7]$, $C_1 = [J_9, J_{10}]$ and $C_2 = [J_2, J_8]$. The sequence of the batches is $\{A_1, B_1, C_2, A_2, C_1, \}$ and $H = 3$. By Steps 3 and 4, the sets and sequence of the jobs are obtained as shown in Tables II and III.

Table II: Assigning the jobs into 5 sets.

Batch	First position	Second position	Third position
A_1	J_6	J_1	
B_1	J_5	J_3	J_7
C_2	J_2	J_8	
A_2	J_4		
C_1	J_{10}	J_9	

Table III: Arranging the job sequence.

Job	J_6	J_1	J_5	J_3	J_7	J_2	J_8	J_4	J_{10}	J_9
p_{1j}	2	3	5	6	5	5	3	6	3	3
p_{2j}	5	5	5	3	4	2	1	3	3	2
p_{3j}	3	4	4	1	3	3	2	1	5	4

All the jobs should be assigned into the batches through Steps 5 and 6: $R_1 = [J_6, J_1, J_5]$, $R_2 = [J_3, J_7, J_2]$, and $R_3 = [J_5, J_8]$. Following Step 7, there are $R_4 = [J_4, J_{10}]$ and $R_5 = [J_9]$. The distribution of jobs to batches is done and shown in Table IV.

The makespan of the above solution was 62 by Algorithm 3.1 and the optimal makespan (60) can be obtained through the mixed integer programming.

Table IV: The distribution of jobs to batches.

Batch no.	Jobs	Processing time: Machine 1	Machine 2	Machine3
1	J_6, J_1, J_5	10	15	11
2	J_3, J_7, J_2	14	9	9
3	J_8	3	1	2
4	J_4	6	3	1
5	J_{10}, J_9	6	5	9

Theorem 3.2: Algorithm 3.1 can solve three-machine job shop scheduling problem in $O(c^4 n^4)$ time.

Proof: Since the heuristic algorithm has given a job sequence through Step 1 to Step 4, the time needed to complete job j on machine M_1 is $C_{1j} = \sum_{j \in P_i} p_{1j}$ for $j = 1, \dots, n$. It is now necessary to determine the departure time of each batch. For the first transporter, a finite number of possible time points for departure from M_1 are required. When the transporter

returns from M_2 to M_1 , two cases are under consideration, i.e., the transporter either carries a new batch immediately or waits for the completion time of a job and then transports the batch containing that job. As for the second transporter, the number of possible time points to leave from M_2 is limited. The same procedure occurred to the first transporter is repeated. In the previous case, the leaving time of batch R_k is $d_{1k-1}+T_2$. In the second case, the leaving time is C_{2j} for job j . Not more than n^2 candidate time points are estimated considering all the possible leaving time for the transporter from M_2 . As for M_3 , it begins to handle a new batch R_k at $d_{1k-1}+T_2/2$ or the time cost in the previous batch on M_2 . Therefore, the increment of the makespan keeps unchanged. The same application to the second transporter was conducted. At most n^2 candidate points were predicted for all the possible leaving time needed for the transporter from M_3 , and the completion time can be computed as a finite number.

In Algorithm 3.1, in Step 6 the Johnson's rule requires $O(n \log n)$ time. The time complexity of the algorithm from Step 1 to Step 5 can be established as $O(c^4 n^4)$. It indicates that the time complexity of Algorithm 3.1 is thought to be $O(c^4 n^4)$.

4. PERFORMANCE ANALYSIS AND COMPUTATIONAL EXPERIMENTS

4.1 Performance analysis

To estimate the performance of heuristic algorithm proposed in this study, the first step is assigned a lower bound on the makespan C_{max}^* pertaining to an optimal schedule π^* .

Theorem 4.1: A value of 2 is imposed to the worst-case ratio of Algorithm 3.1, which is considered as a tight bound.

Proof: Let $C_1 = \min_j p_{1j}$, $C_2 = \min_j p_{2j}$ and $C_3 = \min_j p_{3j}$. Consider three auxiliary problems AUX1, AUX2 and AUX3 of the target problem, where AUX1 is the case with $C_1 = \min_j p_{1j}$, AUX2 is the case with $C_2 = \min_j p_{2j}$ and AUX3 is the case with $C_3 = \min_j p_{3j}$. Let C_1^* , C_2^* and C_3^* be the optimal makespans for AUX1, AUX2 and AUX3, respectively. It is possible to derive the optimal schedules for AUX1, AUX2 and AUX3, respectively. Therefore, the first lower bound can be determined as $lb_1 = C_1^*, C_2^*, C_3^*$.

Ignoring the intermediate transfer, the target problem turns into the classical $TF_3||C_{max}$ problem. Respect to an optimal schedule for $TF_3||C_{max}$, the jobs are sequenced resorting to Johnson's rule. If the permutation π of jobs is known, the makespan of corresponding permutation schedule can be written as $\max\{\sum_{j=1}^u p_{1j} + \sum_{j=u}^v p_{2j} + \sum_{j=v}^{un} p_{3j}\}$ for $TF_3||C_{max}$. Following Algorithm 3.1, the second lower bound can be obtained as $lb_2 = \max\{\sum_{j=1}^u p_{1j} + \sum_{j=u}^v p_{2j} + \sum_{j=v}^{un} p_{3j}\} + k(T_1 + T_2)$. Finally, the derived lower bound on the optimal makespan can be expressed as $lb = \max\{lb_1, lb_2\}$.

The makespan obtained by the proposed algorithm could be presented using $C_{max}^* = \sum_{j=1}^u p_{1j} + \sum_{j=u}^v p_{2j} + \sum_{j=v}^{un} p_{3j} + k(T_1 + T_2) \leq a(N) + b(N) + c(N) + T_1 + T_2$, where jobs u, v, w are critical jobs in Algorithm 3.1, and k denotes the number of consecutive trips undergone by a transporter before the process of job u on M_2 . Hereinto, job u is referred to as the last processed job on M_1 before this sequence of trips; and job v is defined as the first job of the sequence, where jobs $v, v+1, \dots, w$ are continuously processed on M_2 ; similarly, job w is the first job of jobs $w, w+1, \dots, n$ to be continuously processed on machine M_3 . Time gap between adjacent jobs is not permitted. Because of the impossibility of other cases, the theorem is logically proved, which supports the result that the worst-case ratio of Algorithm 3.1 equals 2.

4.2 Computational experiments

A couple of computational experiments were carried out to validate the presented heuristic algorithm. The proposed algorithm was translated into C language and compiled with Matlab.

The executable file ran on a computer (512 MB RAM; 1,204 kB cache). To compare the results, the mixed integer programming model was solved using LINGO11 based on small-scale random instances on the same PC.

A total of five small-scale instances were created randomly. On the basis of uniform distribution within the range $[0, 50]$, the processing time of the jobs on three machines were randomly generated, and the transfer times of the transporters were calculated based on the obtained processing time as introduced in Section 3.2. The results yielded from the heuristic algorithm were contrasted with those from LINGO11.

The near optimal levels of parameters obtained by calibrating the algorithm as mentioned in Section 3.2 were chosen in this study. Therefore, 100, 0.9, 0.05 and 0.8 were assigned to the population size, crossover probability, mutation probability and *DCV*, respectively. The compared results can be seen in Table V and Fig. 1. It is clear that the proposed algorithm implemented with much higher speed than LINGO solver. Although the LINGO could find the optimal solution, the computation time increased exponentially with the increment of instance size. By contrast, the proposed algorithm obtained a near-optimal computing time.

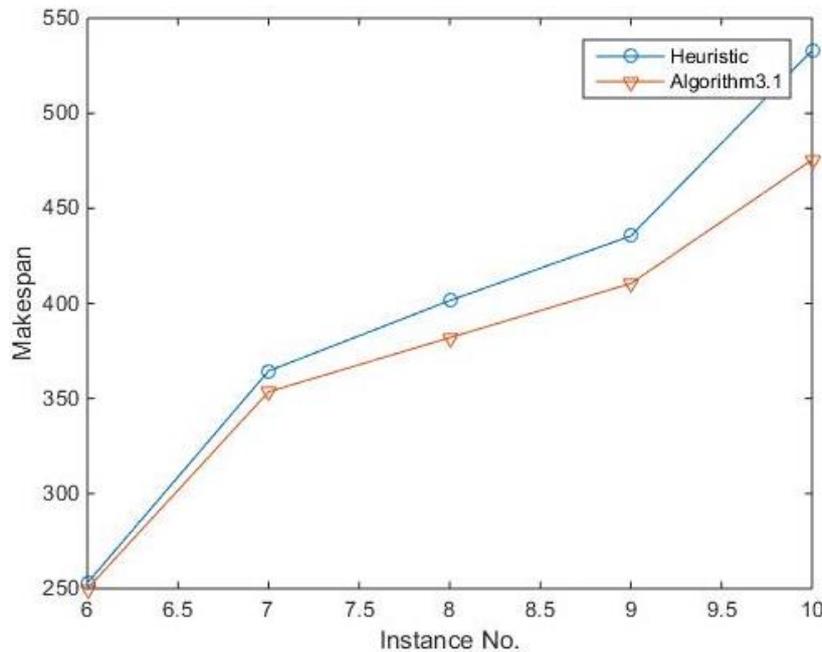


Figure 1: Comparison of the algorithms' performance for small-scale instance.

Table V: Results of small-scale random instances.

Instance no.	Size (N)	Capacity (c)	T_1	T_2	<i>MIP</i> value	Other alg.	Algorithm 3.1
1	6	2	44	50	248.00	253.00	250.00
2	7	2	45	52	333.00	364.50	353.50
3	8	3	56	58	372.00	401.50	382.00
4	9	3	65	64	401.50	435.50	410.50
5	10	4	74	70	456.00	533.00	475.50

Next, the author evaluated the effectiveness of Algorithm 3.1 for problem $TF_3|s_j=1|C_{max}$. The test problems were randomly generated in the following steps:

- Set the number of jobs $n \in \{50, 100, 200, 300\}$ and transporter capacity $c \in \{1, 2, 3\}$.
- Draw transfer times T_1 and T_2 based on the uniform distributions $U[1, 40]$ and $U[1, 20]$.
- Derive job processing time p_{1j} , p_{2j} and p_{3j} from the uniform distributions $U[1, 50]$ and $U[1, 100]$.

With regard to each of the 12 combinations of (n, c) , 10 different test problems were randomly produced for further experiment. For each problem for test, the C_{max}^H and the lower bound C_{max}^* were computed, and the average and maximum relative gaps (labelled by *Avg_gap* and *Max_gap*, respectively) were obtained. $C_{max}^H - C_{max}^*(\%) = (\frac{C_{max}^H - C_{max}^*}{C_{max}^*}) \times 100$. Tables VI and VII list the results of computational experiments. It can be observed that Algorithm 3.1 increased with the value of n , while the relative gaps declined with the increment of n .

Table VI: Results for $T_1, T_2 \in U[1, 20], p_{1j}, p_{2j}, p_{3j} \in U[1, 50]$.

Number of jobs (n)	$c = 1$		$c = 2$		$c = 3$	
	<i>Avg_gap</i>	<i>Max_gap</i>	<i>Avg_gap</i>	<i>Max_gap</i>	<i>Avg_gap</i>	<i>Max_gap</i>
50	0.41	0.08	2.54	9.34	5.48	9.80
100	0.06	0.14	0.89	1.37	2.56	4.54
200	0.11	0.10	0.11	0.24	0.36	0.75
300	0.01	0.13	0.11	0.22	0.22	0.44

Table VII: Results for $T_1, T_2 \in U[1, 40], p_{1j}, p_{2j}, p_{3j} \in U[1, 100]$.

Number of jobs (n)	$c = 1$		$c = 2$		$c = 3$	
	<i>Avg_gap</i>	<i>Max_gap</i>	<i>Avg_gap</i>	<i>Max_gap</i>	<i>Avg_gap</i>	<i>Max_gap</i>
50	0.60	1.43	3.66	7.52	5.87	8.51
100	0.15	0.50	0.86	1.53	2.24	6.85
200	0.05	0.10	0.20	0.38	0.56	0.85
300	0.02	0.05	0.11	0.20	0.19	0.59

5. CONCLUSIONS

In this study, the coordination of three-machine job shop scheduling and intermediate transfer was explored. Each transporter only carries a limited number of jobs of which the total size is not allowed to exceed the transport capacity. The jobs are of the same size. Two transporters carrying the batches of jobs between machines were considered, and the capacity, transfer times of the transporters were taken into account. The longest time cost for processing jobs of a batch on the second machine was considered as the processing time of the batch. The objective is to minimize the makespan. It is proved that the problem is strongly NP-hard, and a heuristic algorithm was proposed for the problem with absolute worst-case ratio of 2. The proposed algorithm was validated in computational experiments on small-scale random instances. It is proved that the algorithm can access to the near optimal solutions during a reasonable time.

ACKNOWLEDGEMENTS

The authors thank two anonymous referees for their helpful comments and suggestions. This work is partly supported by National Natural Science Foundation of China (No. 41772247) and National Natural Science Foundation of China (Grant No. 60674084).

REFERENCES

- [1] Lee, C.-Y.; Chen, Z.-L. (2001). Machine scheduling with transportation considerations, *Journal of Scheduling*, Vol. 4, No. 1, 3-24, doi:[10.1002/1099-1425\(200101/02\)4:1<3::AID-JOS57>3.0.CO;2-D](https://doi.org/10.1002/1099-1425(200101/02)4:1<3::AID-JOS57>3.0.CO;2-D)
- [2] Cheng, T. C. E.; Gordon, V. S.; Kovalyov, M. Y. (1996). Single machine scheduling with batch deliveries, *European Journal of Operational Research*, Vol. 94, No. 2, 277-283, doi:[10.1016/0377-2217\(96\)00127-0](https://doi.org/10.1016/0377-2217(96)00127-0)

- [3] Chang, Y.-C.; Lee, C.-Y. (2004). Machine scheduling with job delivery coordination, *European Journal of Operational Research*, Vol. 158, No. 2, 470-487, doi:[10.1016/S0377-2217\(03\)00364-3](https://doi.org/10.1016/S0377-2217(03)00364-3)
- [4] Yang, Z.; Feng, M.-Q. (2015). Hei river flood risk analysis based on coupling hydrodynamic simulation of 1-D and 2-D simulations, *International Journal of Heat and Technology*, Vol. 33, No. 1, 47-54, doi:[10.18280/ijht.330107](https://doi.org/10.18280/ijht.330107)
- [5] Iuga, M. V.; Kifor, C. V.; Rosca, L. I. (2015). Shop floor key performance indicators in automotive organizations, *Academic Journal of Manufacturing Engineering*, Vol. 13, No. 2, 96-103
- [6] Koch, H. J. (2017). Quantum biology: unit membrane reduces entropy due to wave particle duality, *NeuroQuantology*, Vol. 15, No. 1, 42-44, doi:[10.14704/nq.2017.15.1.1010](https://doi.org/10.14704/nq.2017.15.1.1010)
- [7] Banerjee, S.; Ghosh, A.; Mitra, S. K. (2017). A modified mathematical model for lifetime enhancement in wireless sensor network, *Mathematical Modelling of Engineering Problems*, Vol. 4, No. 2, 84-90, doi:[10.18280/mmep.040204](https://doi.org/10.18280/mmep.040204)
- [8] Zhong, W.; Dosa, G.; Tan, Z. (2007). On the machine scheduling problem with job delivery coordination, *European Journal of Operational Research*, Vol. 182, No. 3, 1057-1072, doi:[10.1016/j.ejor.2006.09.059](https://doi.org/10.1016/j.ejor.2006.09.059)
- [9] Gong, H.; Tang, L. (2011). Two-machine flowshop scheduling with intermediate transportation under job physical space consideration, *Computers & Operations Research*, Vol. 38, No. 9, 1267-1274, doi:[10.1016/j.cor.2010.10.018](https://doi.org/10.1016/j.cor.2010.10.018)
- [10] Maggu, P. L.; Das, G. (1980). On $2 \times n$ sequencing problem with transportation times of jobs, *Pure & Applied Mathematics Sciences (PAMS)*, Vol. 12, No. 1-2, 1-6
- [11] Supsomboon, S.; Vajasuviwon, A. (2016). Simulation model for job shop production process improvement in machine parts manufacturing, *International Journal of Simulation Modelling*, Vol. 15, No. 4, 611-622, doi:[10.2507/IJSIMM15\(4\)3.352](https://doi.org/10.2507/IJSIMM15(4)3.352)
- [12] Li, Y.; Xie, B.-C. (2017). Application of triple partition model to the proof of supply chain scheduling problem, *Journal of Interdisciplinary Mathematics*, Vol. 20, No. 5, 1255-1263, doi:[10.1080/09720502.2017.1311041](https://doi.org/10.1080/09720502.2017.1311041)
- [13] Zhong, Y.; Li, J. M.; Zhu, S. Z. (2017). Research on the multi-objective optimized scheduling of the flexible job-shop considering multi-resource allocation, *International Journal of Simulation Modelling*, Vol. 16, No. 3, 517-526, doi:[10.2507/IJSIMM16\(3\)CO13](https://doi.org/10.2507/IJSIMM16(3)CO13)
- [14] Wang, J.-B.; Xia, Z.-Q. (2005). Scheduling jobs under decreasing linear deterioration, *Information Processing Letters*, Vol. 94, No. 2, 63-69, doi:[10.1016/j.ipl.2004.12.018](https://doi.org/10.1016/j.ipl.2004.12.018)
- [15] Ruiz, R.; Maroto, C.; Alcaraz, J. (2006). Two new robust genetic algorithms for the flowshop sequencing problem, *Omega*, Vol. 34, No. 5, 461-476, doi:[10.1016/j.omega.2004.12.006](https://doi.org/10.1016/j.omega.2004.12.006)
- [16] Zhang, L.; Wang, L.; Zheng, D.-Z. (2006). An adaptive genetic algorithm with multiple operators for flowshop scheduling, *The International Journal of Advanced Manufacturing Technology*, Vol. 27, No. 5-6, 580-587, doi:[10.1007/s00170-004-2223-3](https://doi.org/10.1007/s00170-004-2223-3)
- [17] Chen, B.; Lee, C.-Y. (2008). Logistic scheduling with batching and transportation, *European Journal of Operational Research*, Vol. 189, No. 3, 871-876, doi:[10.1016/j.ejor.2006.11.047](https://doi.org/10.1016/j.ejor.2006.11.047)
- [18] Lee, C.-Y.; Strusevich, V. A. (2005). Two-machine shop scheduling with an uncapacitated interstage transporter, *IIE Transaction*, Vol. 37, No. 8, 725-736, doi:[10.1080/07408170590918290](https://doi.org/10.1080/07408170590918290)
- [19] Tang, L.; Liu, P. (2009). Flowshop scheduling problems with transportation or deterioration between the batching and single machines, *Computers & Industrial Engineering*, Vol. 56, No. 4, 1289-1295, doi:[10.1016/j.cie.2008.07.022](https://doi.org/10.1016/j.cie.2008.07.022)
- [20] Pinedo, M. L. (1995). *Scheduling: Theory, Algorithms, and Systems*, Prentice-Hall, Englewood Cliffs
- [21] He, Y.; Zhong, W.; Gu, H. (2006). Improved algorithms for two single machine scheduling problems, *Theoretical Computer Science*, Vol. 363, No. 3, 257-265, doi:[10.1016/j.tcs.2006.04.014](https://doi.org/10.1016/j.tcs.2006.04.014)
- [22] Cheng, B.; Yang, S.; Hu, X.; Li, K. (2014). Scheduling algorithm for flowshop with two batch-processing machines and arbitrary job sizes, *International Journal of Systems Science*, Vol. 45, No. 3, 571-578, doi:[10.1080/00207721.2012.724107](https://doi.org/10.1080/00207721.2012.724107)
- [23] Tang, L.; Guan, J.; Hu, G. (2010). Steelmaking and refining coordinated scheduling problem with waiting time and transportation consideration, *Computers & Industrial Engineering*, Vol. 58, No. 2, 239-248, doi:[10.1016/j.cie.2009.07.014](https://doi.org/10.1016/j.cie.2009.07.014)

- [24] Coffman, E. G.; Galambos, G.; Martello, S.; Vigo, D. (1999). Bin packing approximation algorithms: Combinatorial analysis, Du, D. Z.; Pardalos, P. M. (Eds.), *Handbook of Combinatorial Optimization*, Springer, Boston, 151-207, doi:[10.1007/978-1-4757-3023-4_3](https://doi.org/10.1007/978-1-4757-3023-4_3)
- [25] Fernandez de la Vega, W.; Lueker, G. S. (1981). Bin packing can be solved within $1+\epsilon$ in linear time, *Combinatorica*, Vol. 1, No. 4, 349-355
- [26] Gong, D.; Tang, M.; Liu, S.; Li, Q. (2017). Reconsidering production coordination: a principal-agent theory-based analysis, *Advances in Production Engineering & Management*, Vol. 12, No. 1, 51-61, doi:[10.14743/apem2017.1.239](https://doi.org/10.14743/apem2017.1.239)
- [27] Tang, L.-X.; Gong, H. (2008). A hybrid two-stage transportation and batch scheduling problem, *Applied Mathematical Modelling*, Vol. 32, No. 12, 2467-2479, doi:[10.1016/j.apm.2007.09.028](https://doi.org/10.1016/j.apm.2007.09.028)
- [28] Ascione, F.; Bianco, N.; De Stasio, C.; Mauro, G. M.; Vanoli, G. P. (2016). A methodology to assess and improve the impact of public energy policies for retrofitting the building stock: application to Italian office buildings, *International Journal of Heat and Technology*, Vol. 34, Special Issue 2, S277-S286, doi:[10.18280/ijht.34Sp0213](https://doi.org/10.18280/ijht.34Sp0213)
- [29] Sun, K.; Li, Y.-P.; Roy, U. (2017). A PLM-based data analytics approach for improving product development lead time in an engineer-to-order manufacturing firm, *Mathematical Modelling of Engineering Problems*, Vol. 4, No. 2, 69-74, doi:[10.18280/mmep.040201](https://doi.org/10.18280/mmep.040201)
- [30] Medina, Y. C.; Khandy, N. H.; Fonticiella, O. M. C.; Morales, O. F. G. (2017). Abstract of heat transfer coefficient modelation in single-phase systems inside pipes, *Mathematical Modelling of Engineering Problems*, Vol. 4, No. 3, 126-131, doi:[10.18280/mmep.040303](https://doi.org/10.18280/mmep.040303)
- [31] Kuric, I.; Císar, M. (2015). Machine tool errors and its simulation on experimental device, *Academic Journal of Manufacturing Engineering*, Vol. 13, No. 4, 17-21
- [32] Pradhan, R. K. (2017). Minimal neural recruitment from Stevens coding and Fechner decoding in the brain, *NeuroQuantology*, Vol. 15, No. 1, 86-91, doi:[10.14704/nq.2017.15.1.1007](https://doi.org/10.14704/nq.2017.15.1.1007)
- [33] Johnson, S. M. (1954). Optimal two- and three-stage production schedules with setup times included, *Naval Research Logistics Quarterly*, Vol. 1, No. 1, 61-68, doi:[10.1002/nav.3800010110](https://doi.org/10.1002/nav.3800010110)