

Research Article

Heuristic for Stochastic Online Flowshop Problem with Preemption Penalties

Mohammad Bayat, Mehdi Heydari, and Mohammad Mahdavi Mazdeh

Department of Industrial Engineering, Iran University of Science and Technology, Tehran 17347-93138, Iran

Correspondence should be addressed to Mohammad Bayat; mohammadbayat@iust.ac.ir

Received 20 May 2013; Accepted 26 August 2013

Academic Editor: Oswaldo Luiz do Valle Costa

Copyright © 2013 Mohammad Bayat et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The deterministic flowshop model is one of the most widely studied problems; whereas its stochastic equivalent has remained a challenge. Furthermore, the preemptive online stochastic flowshop problem has received much less attention, and most of the previous researches have considered a nonpreemptive version. Moreover, little attention has been devoted to the problems where a certain time penalty is incurred when preemption is allowed. This paper examines the preemptive stochastic online flowshop with the objective of minimizing the expected makespan. All the jobs arrive overtime, which means that the existence and the parameters of each job are unknown until its release date. The processing time of the jobs is stochastic and actual processing time is unknown until completion of the job. A heuristic procedure for this problem is presented, which is applicable whenever the job processing times are characterized by their means and standard deviation. The performance of the proposed heuristic method is explored using some numerical examples.

1. Introduction

The stochastic flowshop problems have not penetrated very far and remain challenging. The stochastic two-machine flowshop problem is inherently more complex than its deterministic counterpart. This complexity is much more when the preemptive online version of the problem is considered. In other words, the jobs are allowed to be preempted and restarted. We assume that preemption can occur only on machine 1 in the flowshop problem. Moreover, all the jobs arrive overtime, which indicates that the existence and the parameters of each job are unknown until its release date. In the previous stochastic studies, preemptions are usually assumed to be “free” which means that each job can be preempted at any point of time and resumed later without incurring a penalty. However, this is not usually the case in practice. In many cases, such as melting furnaces, the time that has been spent before the preemption on a job is lost and considered as the preemption penalty.

Scheduling has become a well-studied problem and there are literally tremendous efforts on providing solution strategies for various kinds of modeling formulations such as job

shop and flowshop. There are literally many applications for flowshop problem (e.g., Defersha and Chen [1]; Mahavi et al. [2]; Braglia et al. [3]). Two-machine flowshop problem with makespan objective function and deterministic processing time can be optimally solved by Johnson's rule [4]. This problem would be NP-hard if it consists of three or more machines [5]. Therefore, numerous heuristic algorithms have been presented for solving such problems in various studies. Framinan et al. have reviewed some of these articles [6]. Ruiz and Maroto referred to 53 articles on the heuristics presented for minimizing makespan in permutation flowshop problem [7].

In the stochastic flowshop problem, the processing time is a random variable. This simple difference leads to many complexities in stochastic problems. Makino developed a sequencing rule to find the schedule that minimizes the expected makespan in a flowshop problem with two jobs and exponentially distributed processing time [8]. Frostig and Adiri investigated three-machine flowshop stochastic scheduling with an objective of minimizing distribution of schedule length [9]. Sethi et al. offered feedback production planning in a stochastic two-machine flowshop based on asymptotic analysis and computational results [10]. Elmaghraby and Thoney

studied the two-machine stochastic flowshop problem with arbitrary processing time distributions [11]. In the stochastic flowshop problem with two machines and exponential processing times, the expected makespan value would be minimized in the case that the jobs are sorted nonincreasingly in terms of parameter $(1/\mu_{i1} - 1/\mu_{i2})$. This method was proposed by Talwar and is known as Talwar's Rule [12]. Later, Cunningham and Dutta proved its optimality [13]. Ku and Niu obtained a sufficient condition for stochastic dominance and showed that Talwar's Rule yields a stochastically minimal makespan [14]. Soroush and Allahverdi presented a stochastic two-machine flowshop scheduling problem with total completion time criterion [15]. Laha and Chakraborty presented an efficient stochastic hybrid heuristic for flowshop scheduling [16]. Portugal and Trietsch applied Johnson's Rule to stochastic problems [17]. They utilized, mean of the processing time of each job as its processing time in Johnson's Rule. Moreover, Kalczyński and Kamburowski applied Talwar's Rule for Weibull distribution [18]. Baker and Altheimer used three heuristic methods for the flowshop problem with m machines, supposing general distributions for processing times. They investigated the performance of these methods in a set of problems using simulation and noticed that these methods had near-optimal performance [19]. Baker and Trietsch also explored three heuristic methods for the two-machine stochastic model with a general distribution function. They compared Johnson's Heuristic method and Talwar's Heuristic method (applying mean of the processing time instead of job processing time in these two methods) and also the heuristic method of changing neighboring pairs (two neighboring jobs are separately considered and are displaced if their order can be optimized) and figured out that none of these methods dominate the other [20]. Heydari et al. proposed a heuristic method for minimization of expected value of the total weighted completion time in single machine problem with preemption penalties [21]. In this paper, the heuristic proposed by Baker and Trietsch has been applied for sequencing the jobs which are present in the shop.

In spite of the various studies in the last decades, a considerable research has not been performed on preemptive online flowshop problems with stochastic processing times. In this paper, a heuristic method to this problem is presented. The logic applied in this research is emphasizing on minimizing idling times in the second machine. In this method, decision about the preemption of one job at arrival of a new job is made based on the idling time produced in the second machine. In this research, we assume that the processing time is a random variable with normal distribution, and processing times of different jobs are independent from each other.

The remainder of this paper is organized as follows: research assumptions and definitions are presented in Section 2. In Sections 3 and 4, nonpreemptive and preemptive stochastic models are reviewed. Then, the proposed heuristic method is presented in Section 5. In Section 6, a numerical example has been solved with the proposed algorithm. The performance analysis of the proposed heuristic method is presented in Section 7. Finally, the conclusions are discussed in the last section.

2. Research Assumptions and Definitions

In the stochastic case, t_{ip} is used to denote (random) processing time of job i on machine p , but we retain $E(t_{ij})$ to represent its expected value. And, r_i is the release time of job i , which can only be known right on or after r_i . The processing time of each job is a random variable with normal distribution $t_{ip} \sim N(\mu_{ip}, \sigma_{ip}^2)$ and jobs are independent of each other. Our objective is to find a schedule to minimize the expected makespan. The applied symbols are as follows:

- t_{ip} : processing time of job i on machine p .
- μ_{ip} : mean of processing time of job i on machine p .
- σ_{ip}^2 : variance of processing time of job i on machine p .
- I_i : idle time of the second machine from completing the job $i - 1$ until starting the job i .

The considered assumptions in this article are as follows:

- (i) Machines have constant speed that cannot be varied.
- (ii) The order of processing the jobs is the same on the first and second machines.
- (iii) Machines are ready to be utilized at zero time.
- (iv) Every machine can operate at most one job at a time.
- (v) Initiation of any job on the second machine would be after completion of the job on the first machine.
- (vi) Preemption of the jobs can be occurred only on machine 1.

We consider the preemption-repeat model in which the job currently being processed may be preempted at any point of time. However, by preempting a job, all the ongoing progress is considered to be lost. Therefore, if the job is restarted at some later moment in time, then it has to be processed from the beginning.

3. Nonpreemptive Stochastic Flowshop Problem

For general distributions without any special conditions on processing times, only one thorough solution is known for the stochastic flowshop problem, proposed by Makino [8].

Theorem 1. *In the two-machine stochastic flowshop problem with two jobs, job i precedes job j in an optimal sequence if*

$$E(\min\{t_{i1}, t_{j2}\}) \leq E(\min\{t_{i2}, t_{j1}\}). \quad (1)$$

Based on Theorem 1, Baker and Trietsch proposed a method that uses the properties of an adjacent pair wise interchange (API) [20]. They assume that the processing times are random variables with normal distribution. In this method, the condition for job i to precede job j takes the following form:

$$\begin{aligned} & \mu_{i1} - \sigma_{i1j2} [\phi(z_{i1j2}) + z_{i1j2} \cdot \Phi(z_{i1j2})] \\ & \leq \mu_{i2} - \sigma_{i2j1} [\phi(z_{i2j1}) + z_{i2j1} \cdot \Phi(z_{i2j1})]. \end{aligned} \quad (2)$$

This form of the API procedure uses the properties of normal distributions and is called the API Heuristic. In this method, the expected minimum of two variables with normal distribution is given by

$$E[\min\{X, Y\}] = \mu_X - \sigma_{XY} [\phi(z_{XY}) + z_{XY} \cdot \Phi(z_{XY})], \quad (3)$$

where ϕ and Φ denote the density function and the cdf of the standard normal. In addition, $\mu_{XY} = \mu_X - \mu_Y$ and $\sigma_{XY}^2 = \sigma_X^2 + \sigma_Y^2$ and $z_{XY} = \mu_{XY}/\sigma_{XY}$ [22]. In this paper, for sequencing the jobs present at the shop, this heuristic method has been utilized.

4. Preemptive Stochastic Flowshop Problem

Since the objective function of this study is expected makespan minimization, and considering the fact that makespan increase is due to increase in idling time in the second machine, this paper focused on minimizing the sum of expected idling times of the second machine.

As shown in Figure 1 the value of makespan is as follows:

$$\begin{aligned} C_{\max} &= \sum_{i=1}^n I_i + \sum_{i=1}^n t_{i2} \\ \implies E[C_{\max}] &= E\left[\sum_{i=1}^n I_i\right] + E\left[\sum_{i=1}^n t_{i2}\right], \end{aligned} \quad (4)$$

where I_i is the idling time of machine 2 before the starting of job i . Since the value of $E[\sum_{i=1}^n t_{i2}]$ is constant; minimizing $E[\sum_{i=1}^n I_i]$ leads to minimum value of makespan. Idling time of machine 2 is as follows:

$$\begin{aligned} I_1 &= t_{11}, \\ I_2 &= \max\{0, t_{11} + t_{21} - t_{12} - I_1\}, \\ I_2 &= \max\{0, t_{11} + t_{21} + t_{31} - t_{12} - t_{22} - I_1 - I_2\}, \\ &\vdots \\ I_k &= \max\left\{0, \sum_{i=1}^k t_{i1} - \sum_{i=1}^{k-1} t_{i2} - \sum_{i=1}^{k-1} I_i\right\}. \end{aligned} \quad (5)$$

Thus, the total idle time of the second machine will be computed as follows:

$$\begin{aligned} I_1 + I_2 &= \max\{t_{11}, t_{11} + t_{21} - t_{12}\}, \\ I_1 + I_2 + I_3 &= \max\{t_{11}, t_{11} + t_{21} - t_{12}, \\ &\quad t_{11} + t_{21} + t_{31} - t_{12} - t_{22}\}, \\ &\vdots \\ \implies \sum_{i=1}^n I_i &= \max\left[t_{11}, \max_{2 \leq k \leq n} \left\{\sum_{i=1}^k t_{i1} - \sum_{i=1}^{k-1} t_{i2}\right\}\right]. \end{aligned} \quad (6)$$

Assume that

$$U_k = \sum_{i=1}^k t_{i1} - \sum_{i=1}^{k-1} t_{i2}. \quad (7)$$

Therefore, the sum of idling times of machine 2 is given by

$$\sum_{i=1}^n I_i = \max_{1 \leq k \leq n} (U_k). \quad (8)$$

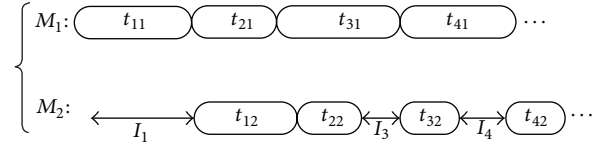


FIGURE 1: Two machine flowshop problem.

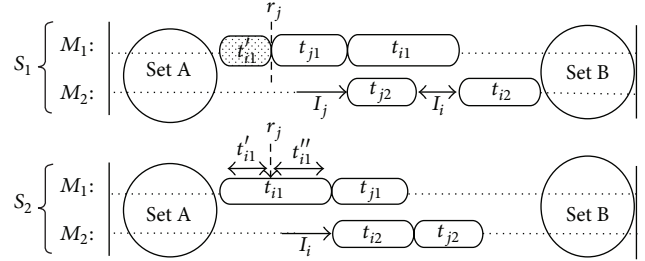


FIGURE 2: Schedules S_1 and S_2 at arrival of job j in preemption-repeat mode.

In the two-machine stochastic online flowshop problem, if in schedule 1 (S_1) job i precedes job j and in schedule 2 (S_2), these two jobs are replaced, then $E(C_{\max}^{S_1}) \leq E(C_{\max}^{S_2})$ if

$$E\left[\max_{1 \leq k \leq n} \{U_k(S_1)\}\right] \leq E\left[\max_{1 \leq k \leq n} \{U_k(S_2)\}\right], \quad (9)$$

where $E[\max_{1 \leq k \leq n} \{U_k(S_1)\}]$ and $E[\max_{1 \leq k \leq n} \{U_k(S_2)\}]$ are the expected values of the sum of idling time on machine 2 in schedule 1 and 2.

As mentioned before, minimizing the expected value of total idling time in machine 2 leads to expected makespan minimization. Since the expected value of makespan in S_1 is less than its value in S_2 , the idle time of machine 2 in S_1 will be less than its value in S_2 .

In preemptive online problem, assume that job i with processing time t_{i1} is being processed on machine 1. Let t'_{i1} be the remaining processing time of job i and the time spent for processing of job i is t''_{i1} . Therefore, we have

$$t_{i1} = t'_{i1} + t''_{i1}. \quad (10)$$

Suppose that the new job j with the processing time t_{jp} and the release time r_j arrives at the shop (Figure 2). The new job j becomes known at its release time.

Let A be the subset of completed jobs on machine 1 and B the subset of uncompleted jobs. First, the priority of job j is determined with the API heuristic method. If the priority of job j is more than the job i then one of the two S_1 and S_2 schedules would occur as shown in Figure 2; otherwise, it will be added to list of uncompleted jobs. In S_1 job i is preempted because of the higher priority for job j , but in S_2 job j will be processed after completion of job i . As mentioned before,

total idle time of machine 2 until start of job i and j in S_1 and S_2 schedules is as follows:

$$\begin{aligned}
U_j(S_1) &= \sum_{h \in A} t_{h1} + t'_{i1} + t_{j1} - \sum_{h \in A} t_{h2}, \\
U_{i'}(S_1) &= \sum_{h \in A} t_{h1} + t'_{i1} - \sum_{h \in A} t_{h2}, \\
U_i(S_1) &= \sum_{h \in A} t_{h1} + t'_{i1} + t_{j1} + t_{i1} - t_{j2} - \sum_{h \in A} t_{h2}, \\
U_i(S_2) &= \sum_{h \in A} t_{h1} + t_{i1} - \sum_{h \in A} t_{h2}, \\
U_j(S_2) &= \sum_{h \in A} t_{h1} + t_{i1} + t_{j1} - t_{i2} - \sum_{h \in A} t_{h2}.
\end{aligned} \tag{11}$$

Therefore, the expected value of makespan in S_1 is less than its value in S_2 and preemption will occur if

$$\begin{aligned}
&E[\max\{U_{i'}(S_1), U_j(S_1), U_i(S_1)\}] \\
&< E[\max\{U_i(S_2), U_j(S_2)\}] \\
&\xrightarrow{U_j(S_1) > U_{i'}(S_1)} E[\max\{U_j(S_1), U_i(S_1)\}] \\
&< E[\max\{U_i(S_2), U_j(S_2)\}] \\
&\implies E[\min\{-U_j(S_1), -U_i(S_1)\}] \\
&> E[\min\{-U_j(S_2), -U_i(S_2)\}].
\end{aligned} \tag{12}$$

The processing of the jobs in the subset A has been completed on machine 1, and their total processing time is a constant value. Thus, we assume that

$$\sum_{h \in A} t_{h1} = \lambda, \tag{13}$$

where λ is a constant value. Some of the jobs in the subset A have been completed and some of them have not been completed until time r_j . Let A' be the subset of completed jobs on machine 2, and let A'' be the subset of uncompleted jobs on this machine. Thus,

$$\begin{aligned}
A' \cup A'' &= A, \\
\sum_{h \in A} t_{h2} &= \sum_{h \in A'} t_{h2} + \sum_{h \in A''} t_{h2}.
\end{aligned} \tag{14}$$

Therefore, $\sum_{h \in A'} t_{h2}$ is constant value $\sum_{h \in A''} t_{h2}$ is a random variable with normal distribution

$$\begin{aligned}
\sum_{h \in A'} t_{h2} &= \beta, \\
\sum_{h \in A''} t_{h2} &\sim N\left(\sum_{h \in A''} \mu_{h2}, \sum_{h \in A''} \sigma_{h2}^2\right),
\end{aligned} \tag{15}$$

where β is a constant value. Hence, $U_j(S_1)$ and $U_i(S_1)$ are random variables with normal distribution and the following means and variances:

$$\begin{aligned}
U_j(S_1) &\sim N\left(\lambda - \beta + t'_{i1} - \sum_{h \in A''} \mu_{h2} + \mu_{j1}, \sigma_{j1}^2 + \sum_{h \in A''} \sigma_{h2}^2\right), \\
U_i(S_1) &= N\left(\lambda - \beta + t'_{i1} - \sum_{h \in A''} \mu_{h2} + \mu_{j1} + \mu_{i1} \right. \\
&\quad \left. - \mu_{j2}, \sigma_{j1}^2 + \sigma_{i1}^2 + \sigma_{j2}^2 + \sum_{h \in A''} \sigma_{h2}^2\right).
\end{aligned} \tag{16}$$

In addition, for $U_j(S_2)$ and $U_i(S_2)$ we have

$$\begin{aligned}
U_i(S_2) &= N\left(\lambda - \beta - \sum_{h \in A''} \mu_{h2} + \mu_{i1}, \sigma_{i1}^2 + \sum_{h \in A''} \sigma_{h2}^2\right), \\
U_j(S_2) &= N\left(\lambda - \beta - \sum_{h \in A''} \mu_{h2} + \mu_{i1} + \mu_{j1} - \mu_{i2}, \right. \\
&\quad \left. \sigma_{i1}^2 + \sigma_{j1}^2 + \sigma_{i2}^2 + \sum_{h \in A''} \sigma_{h2}^2\right).
\end{aligned} \tag{17}$$

In order for decision making about preemption, the following inequality should be explored:

$$\begin{aligned}
&E[\min\{-U_j(S_1), -U_i(S_1)\}] \\
&> E[\min\{-U_j(S_2), -U_i(S_2)\}].
\end{aligned} \tag{18}$$

Assume that

$$\begin{aligned}
X &= -U_j(S_1), \\
Y &= -U_i(S_1), \\
X' &= -U_j(S_2), \\
Y' &= -U_i(S_2).
\end{aligned} \tag{19}$$

The expected minimum of two variables with normal distribution is calculated as discussed before. Thus, for computation of $E[\min\{X, Y\}]$ we have

$$\begin{aligned}
\mu_X &= -\lambda + \beta - t'_{i1} + \sum_{h \in A''} \mu_{h2} - \mu_{j1}, \\
\mu_{XY} &= \mu_{i1} - \mu_{j2}, \\
\sigma_{XY}^2 &= 2\sigma_{j1}^2 + \sigma_{i1}^2 + \sigma_{j2}^2 + 2 \sum_{h \in A''} \sigma_{h2}^2,
\end{aligned} \tag{20}$$

$$\begin{aligned}
z_{XY} &= \frac{\mu_{XY}}{\sigma_{XY}} \\
&\implies E[\min\{X, Y\}] \\
&= \mu_X - \sigma_{XY} [\phi(z_{XY}) + z_{XY} \cdot \Phi(z_{XY})],
\end{aligned}$$

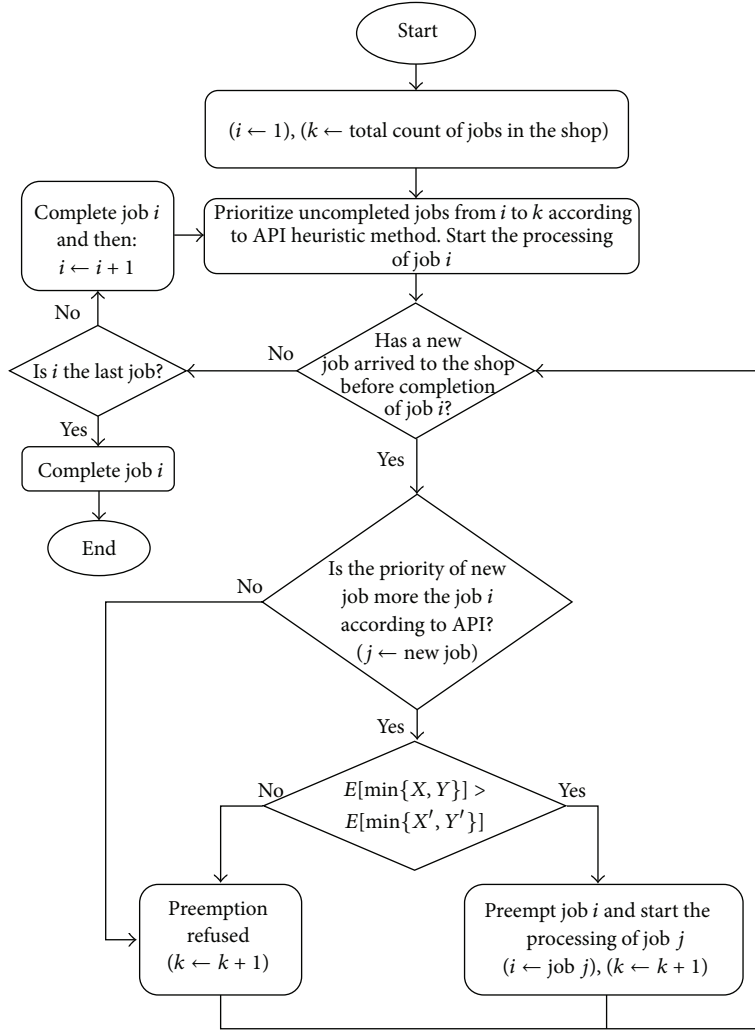


FIGURE 3: Proposed heuristic algorithm.

and $E[\min\{X', Y'\}]$ is as follows:

$$\begin{aligned} \mu_{X'} &= -\lambda + \beta + \sum_{h \in A''} \mu_{h2} - \mu_{i1}, \\ \mu_{X'Y'} &= \mu_{j1} - \mu_{i2}, \\ \sigma_{X'Y'}^2 &= 2\sigma_{i1}^2 + \sigma_{j1}^2 + \sigma_{i2}^2 + 2 \sum_{h \in A''} \sigma_{h2}^2, \\ z_{X'Y'} &= \frac{\mu_{X'Y'}}{\sigma_{X'Y'}} \end{aligned} \quad (21)$$

$$\begin{aligned} E[\min\{X', Y'\}] &= \mu_{X'} - \sigma_{X'Y'} [\phi(z_{X'Y'}) + z_{X'Y'} \cdot \Phi(z_{X'Y'})]. \end{aligned}$$

Therefore, the preemption condition will be as presented in Lemma 2.

Lemma 2. *In the preemption-repeat stochastic online flowshop problem with two machines, job i will be preempted at arrival of*

job j (assuming that the job j priority is greater than the job i based on API) if and only if

$$E[\min\{X, Y\}] > E[\min\{X', Y'\}]. \quad (22)$$

Proof. As mentioned before, by preemption of job i at arrival of job j the expected value of total idling time in machine 2 decreases if $E[\min\{X, Y\}] > E[\min\{X', Y'\}]$. \square

5. Proposed Algorithm

In this section, a heuristic algorithm is proposed to minimize the expected value of makespan for the preemptive stochastic online flowshop problem with two machines. In addition, a schematic illustration of the algorithm is provided in Figure 3.

Step 0 (parameter definition). The parameter k denotes the total number of jobs in the shop and i is the job counter parameter with the primary value of 1. Processing time of each job is a random variable with normal distribution.

TABLE 1: Specifications of jobs.

i	μ_{i1}	σ_{i1}	μ_{i2}	σ_{i2}	r_i
1	17	1.5	10	2.3	0
2	11	2	13	2.1	0
3	16	2	15	2.4	0
4	10	2.4	14	1.6	0
5	11	2.6	8	1.1	15
6	9	1.2	14	2	24

Arrival time of job i is denoted by r_i . Parameter j denotes the new job that arrives at the shop.

Step 1. Prioritize all uncompleted jobs with API heuristic method, from i to k .

Step 2. Process the job i until it is completed, or a new job arrives at the shop. If a new job arrives at the shop then assign index j to its characteristics and go to Step 4, otherwise go to next step.

Step 3. After completion of job i , if i is the last job then finish the algorithm; otherwise, increment the counter i by one and go to Step 1.

$$i \leftarrow i + 1. \quad (23)$$

Step 4. If the priority of the new job according to API heuristic method is more than job i then follow the algorithm, otherwise go to Step 2.

Step 5. If $E[\min\{X, Y\}] > E[\min\{X', Y'\}]$ then go to next step, otherwise go to Step 7.

Step 6. The priority of job j is more than job i . Thus, job i will be preempted and job j will be preferred. And since then, the index i will be used for job j . Increment the counter k by one, and then go to Step 2.

$$\begin{aligned} k &\leftarrow k + 1, \\ i &\leftarrow \text{job } j. \end{aligned} \quad (24)$$

Step 7. Preemption is not allowed and the processing of job i will be continued. Increment the counter k by one, and then go to Step 2.

$$k \leftarrow k + 1. \quad (25)$$

6. Numerical Example

In this section, the performance of the proposed algorithm is evaluated through a numerical example. Suppose a stochastic two-machine flowshop problem with 6 jobs and specifications given in Table 1.

In Table 1, r_i is arrival time of job i . Moreover, we assume that the actual processing time of a job on each machine is equal to its mean processing time on that machine. According to the proposed algorithm, first, the jobs present at the shop at

TABLE 2: Differences between $E[\min\{t_{i1}, t_{j2}\}]$ and $E[\min\{t_{i2}, t_{j1}\}]$.

i	j			
	1	2	3	4
1		3.7134	4.6283	5.2399
2	-3.7134		-1.9221	1.1447
3	-4.6283	1.9221		3.7873
4	-5.2399	-1.1447	-3.7873	

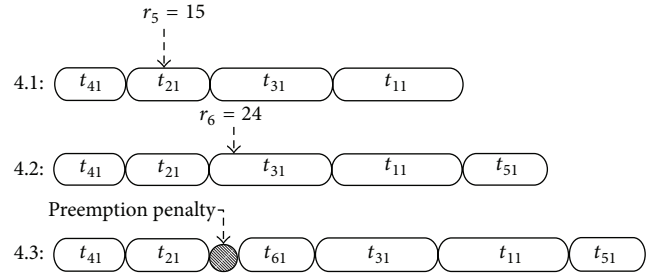


FIGURE 4: Scheduling a sample problem with proposed algorithm.

time zero should be scheduled with API heuristic method. We run the API algorithm by starting with the sequence 1-2-3-4. Table 2 shows the differences between $E[\min\{t_{i1}, t_{j2}\}]$ and $E[\min\{t_{i2}, t_{j1}\}]$ for these jobs, with job i corresponding to a row and j to a column. For instance, the calculation of $E[\min\{t_{11}, t_{22}\}]$ is as follows:

$$\begin{aligned} E[\min\{t_{11}, t_{22}\}] &= \mu_{11} - \sigma_{(11)(22)} [\phi(z_{(11)(22)}) + z_{(11)(22)} \cdot \Phi(z_{(11)(22)})], \\ \mu_{11} &= 17, \\ \mu_{(11)(22)} &= \mu_{11} - \mu_{12} = 17 - 13 = 4, \\ \sigma_{(11)(22)}^2 &= \sigma_{11}^2 + \sigma_{22}^2 = 1.5^2 + 2.1^2 = 2.5807^2, \\ z_{(11)(22)} &= \frac{\mu_{(11)(22)}}{\sigma_{(11)(22)}} = 1.55, \\ E[\min\{t_{11}, t_{22}\}] &= 17 - 2.5807 [0.12 + 1.55 \times 0.9394] = 12.9326. \end{aligned} \quad (26)$$

Similarly, $E[\min\{t_{12}, t_{21}\}] = 9.2192$, and the first entry in Table 2 is the difference $12.9326 - 9.2192 = 3.7134$.

In Table 2, according to API method, negative values indicate stable sequences. For instance, 1-2 is not a stable sequence because the value of row 1 and column 2 is positive. Therefore, these two jobs should be substituted. Hence, API heuristic with altered order of the jobs yields the stable sequence 4-2-3-1 as shown in Figure 4 (4.1).

The machines will start to process the jobs according to the sequence 4-2-3-1. Job 5 with release time $r_5 = 15$ arrives at the shop as shown in Figure 4 (4.1). Job 2 is being processed at this time. According to API, The priority of job 5 is less than the priority of job 2 because if we schedule jobs 2, 5 with this

method, then the job 5 will has less priority. Thus, the preemption is not allowed. Processing of the jobs is continued and during the processing of job 3, job 6 with release time $r_6 = 24$ arrives to the shop as shown in Figure 4 (4.2). According to API, The priority of job 6 is more than the priority of job 3 at this time. Thus, the parameters at this time are as follows:

$$\begin{aligned} i &= 3, \\ i &\leftarrow \text{job 3}, \\ j &\leftarrow \text{job 6}, \\ t'_{31} &= 3. \end{aligned} \quad (27)$$

According to the proposed algorithm, satisfaction of the inequality $E[\min\{X, Y\}] > E[\min\{X', Y'\}]$ should be explored in order for decision making about the preemption:

$$\begin{aligned} \mu_X &= -\lambda + \beta - t'_{i1} + \sum_{h \in A''} \mu_{h2} - \mu_{j1} \\ &= -21 + 14 - 3 + 13 - 9 = -6, \\ \mu_{XY} &= \mu_{i1} - \mu_{j2} = 16 - 14 = 2, \\ \sigma_{XY}^2 &= 2\sigma_{j1}^2 + \sigma_{i1}^2 + \sigma_{j2}^2 + 2 \sum_{h \in A''} \sigma_{h2}^2 \\ &= 2 \times 1.2^2 + 2^2 + 2^2 + 2 \times 2.1^2 = 4.4385^2, \\ z_{XY} &= \frac{\mu_{XY}}{\sigma_{XY}} = 0.4506 \\ &\implies E[\min\{X, Y\}] \\ &= \mu_X - \sigma_{XY} [\phi(z_{XY}) + z_{XY} \cdot \Phi(z_{XY})] \\ &= -6 - 4.4385 (0.3604 + 0.4506 \times 0.6739) = -8.947. \end{aligned} \quad (28)$$

Similarly, $E[\min\{X', Y'\}] = -10.262$. Since $E[\min\{X, Y\}] > E[\min\{X', Y'\}]$, preemption is allowed. Job 3 will be preempted and job 6 will start. Consequently, the scheduling of jobs on machine 1 will be as shown in Figure 4 (4.3).

The computation of completion time of the jobs on machine 1 and 2 are provided in Table 3, where C_{ij} is the completion time of job i on machine.

As mentioned before, it is assumed that the actual processing time of the job is equal to the mean of processing time. Therefore, the expected value of makespan is equal to 85 for this example.

7. Performance Analysis

In order to evaluate the performance of the proposed algorithm, it was applied on a variety of problems with different sizes. Thus, 20,000 problems in 20 categories with different quantities for the number of jobs and the processing time specifications have been produced. The results are compared

TABLE 3: Completion time of jobs on machines 1 and 2.

i	R_{i1}	C_{i1}	R_{i2}	C_{i2}	r_i
4	10	10	14	24	0
2	11	21	13	37	0
6	9	33	14	51	25
3	16	49	15	66	0
1	17	66	10	76	0
5	11	77	8	85	15

to a lower bound of the optimum makespan that is defined as follows:

$$\begin{aligned} C_{\max}^{\text{OPT}} &\geq \max_{i \in \{1,2,3,\dots,n\}} [r_i + E(t_{i1}) + E(t_{i2})], \\ C_{\max}^{\text{OPT}} &\geq \min_{i \in \{1,2,3,\dots,n\}} [r_i + E(t_{i1})] + \sum_{i=1}^n E(t_{i2}), \\ C_{\max}^l &= \max \left\{ \max_{i \in \{1,2,3,\dots,n\}} [r_i + E(t_{i1}) + E(t_{i2})], \right. \\ &\quad \left. \min_{i \in \{1,2,3,\dots,n\}} [r_i + E(t_{i1})] + \sum_{i=1}^n E(t_{i2}) \right\} \\ &\implies C_{\max}^{\text{OPT}} \geq C_{\max}^l. \end{aligned} \quad (29)$$

Parameter ρ is the performance guarantee of the proposed method (MB) and called approximation factor if

$$C_{\max}^{\text{MB}} \leq \rho \times C_{\max}^l \implies C_{\max}^{\text{MB}} \leq \rho \times C_{\max}^{\text{OPT}}, \quad (30)$$

where C_{\max}^{MB} and C_{\max}^{OPT} are the expected values of makespan that the MB method and an optimal method, respectively, achieve on each instance.

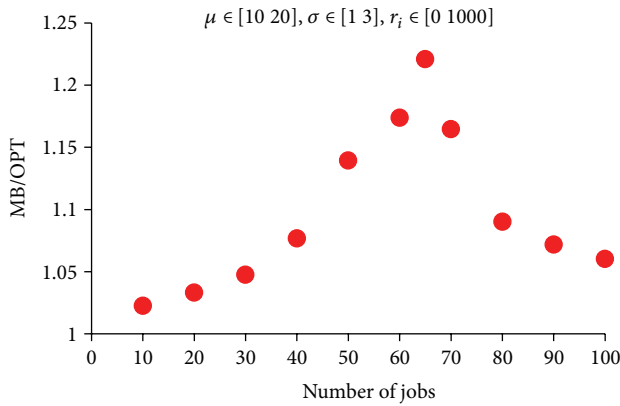
Other assumptions are as follows:

- (1) Release dates are generated using uniform distribution within the interval $[0, 1000]$.
- (2) All jobs are preemption-restart.
- (3) The actual processing time is assumed to be equal to the mean of processing time.
- (4) We have produced 1000 problems for each category and based on the comparison between results of each method, the maximum value of factor ρ is calculated for these problems (worse case).

To compare the performance of the proposed method (MB) against OPT; first, the performance is analyzed with respect to the number of jobs which is varies from 10 to 100 and 10 categories are produced. The processing time follows a normal distribution. We assume that the mean of processing time and standard deviation of processing time are random values within the interval $[10, 20]$ and $[1, 3]$, respectively. Since the mean of processing time follows uniform distribution, the expected value of the processing time is $E(\mu_{ij}) = 15$. Table 4 and Figure 5 summarize the details of our implementations. Second and third columns of Table 4 are the mean of makespan for 1000 produced problems in each category. Whereas

TABLE 4: Makespan value of the proposed algorithm versus lower bound of optimum.

No. of jobs	$C_{\max}^{\text{MB}} \times 10^{-3}$	$C_{\max}^{\text{OPT}} \times 10^{-3}$	ρ_{\max}
10	0.9316	0.9308	1.0225
20	0.9912	0.9874	1.0332
30	1.0039	0.9959	1.0476
40	1.0183	1.0046	1.0766
50	1.0361	1.0096	1.1394
60	1.0641	1.0129	1.1737
65	1.1054	1.0279	1.2209
70	1.1693	1.1016	1.1646
80	1.3257	1.2823	1.0901
90	1.496	1.4648	1.0718
100	1.6662	1.6367	1.0602

FIGURE 5: ρ versus the number of jobs.

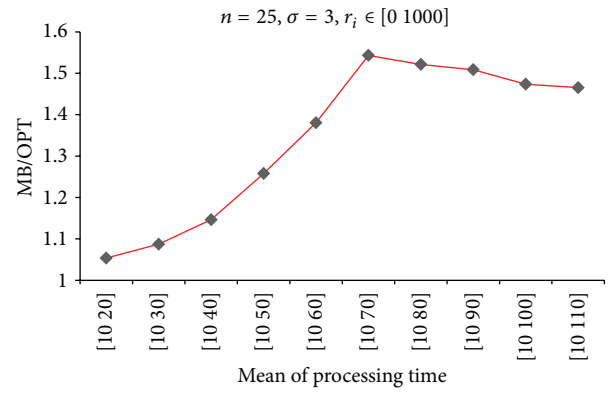
the forth column is the maximum value of factor ρ among 1000 problems in each category. As can be observed, the proposed method (MB) performs up to 1.2209 times worse than the other one when number of jobs approaches 65. Note that when n extends beyond 65, the density of jobs will increase illogically and we have $n \times E(\mu_{ij}) > 1000$ while $r_i < 1000$. When the density of jobs increases illogically, the waiting time of jobs increases and the idle time of machine 2 decreases. Thus, the performance of proposed algorithm almost improves.

The performance of the proposed method has also been evaluated for different values of the mean of processing time. Ten categories have been produced and the number of jobs and the value of standard deviation have been fixed at constant values ($n = 25$, $\sigma = 3$). The results are shown in Table 5 and Figure 6. When the mean of processing time approaches to the interval [10, 70], the performance of the algorithm is in the worst condition because $n \times E(\mu_{ij})$ and $\max\{r_i\}$ are almost equal to 1000. For the next intervals, the density of jobs increase illogically and the idle time of machine 2 decrease. Hence, the performance of MB algorithm almost improves.

Moreover, the performance of the proposed method has been evaluated with respect to distribution of process times as

TABLE 5: The performance of the proposed algorithm (MB) against OPT where the number of jobs is 25.

Mean of processing time	$C_{\max}^{\text{MB}} \times 10^{-3}$	$C_{\max}^{\text{OPT}} \times 10^{-3}$	ρ_{\max}
[10 20]	0.9958	0.9907	1.0537
[10 30]	1.0156	1.0037	1.0874
[10 40]	1.0391	1.0151	1.1464
[10 50]	1.0783	1.0246	1.2579
[10 60]	1.1439	1.0365	1.3808
[10 70]	1.2466	1.0879	1.5433
[10 80]	1.3709	1.1847	1.5215
[10 90]	1.5226	1.3341	1.5087
[10 100]	1.6871	1.4994	1.4736
[10 110]	1.8537	1.6595	1.4653

FIGURE 6: ρ versus mean of processing time.

the uniform distribution [1, 100] with the coefficient of variation being 0.05, 0.1, and 0.2. The number of jobs varies from 10 to 100, and ten categories have been produced. This mode of fixing the process-time distribution can yield better insight about the evaluation of computational performance of the algorithm. The processing times are generated using uniform distribution within the interval [1, 100]. It is assumed that the release dates are random values within the interval [0, 3000]. Taking into consideration the fact that the mean of processing time is uniformly distributed, the expected value of the processing time is $E(\mu_{ij}) = 50.5$. The details of implementations have been summarized in Table 6 and Figure 7. When n takes a value more than 60, an illogical increase will occur in the density of jobs and we have $n \times E(\mu_{ij}) > 3000$ while $r_i < 3000$. Hence, an increase is observed in the waiting time of jobs and the idle time of the second machine decreases. Therefore, the performance of the proposed algorithm improves.

8. Concluding Remarks

In this paper, preemptive stochastic online scheduling problem was investigated for two-machine flowshop. The objective function was minimizing the expected value of makespan. Inspired by Johnson's heuristic method for stochastic flowshop problem with an emphasis on minimal idle times of second machine, a heuristic method was presented.

TABLE 6: The performance of the proposed algorithm versus lower bound of optimum.

Coefficient of variation No. of jobs	0.05			0.1			0.2		
	$C_{max}^{MB} \times 10^{-3}$	$C_{max}^{OPT} \times 10^{-3}$	ρ_{max}	$C_{max}^{MB} \times 10^{-3}$	$C_{max}^{OPT} \times 10^{-3}$	ρ_{max}	$C_{max}^{MB} \times 10^{-3}$	$C_{max}^{OPT} \times 10^{-3}$	ρ_{max}
10	2.7968	2.7897	1.0219	2.8459	2.8405	1.0254	2.8307	2.825	1.0227
20	2.964	2.9525	1.0326	2.9743	2.9622	1.0418	2.971	2.9589	1.0306
30	3.0341	3.007	1.0625	3.0188	2.989	1.0803	3.0386	3.0048	1.1078
40	3.0838	3.0329	1.0841	3.0749	3.0237	1.1027	3.0739	3.0203	1.1109
50	3.1526	3.0436	1.1444	3.1592	3.0535	1.1857	3.1721	3.046	1.1915
60	3.3512	3.1049	1.2379	3.3944	3.1656	1.2431	3.4213	3.1543	1.1947
70	3.7758	3.6411	1.1272	3.8438	3.7081	1.1174	3.8903	3.7421	1.1521
80	4.2539	4.1585	1.0656	4.3728	4.2702	1.079	4.4989	4.3706	1.0951
90	4.8665	4.6817	1.0641	4.8975	4.8162	1.0453	5.0656	4.9662	1.0567
100	5.2566	5.1918	1.0401	5.4189	5.3534	1.0376	5.6141	5.5313	1.0455

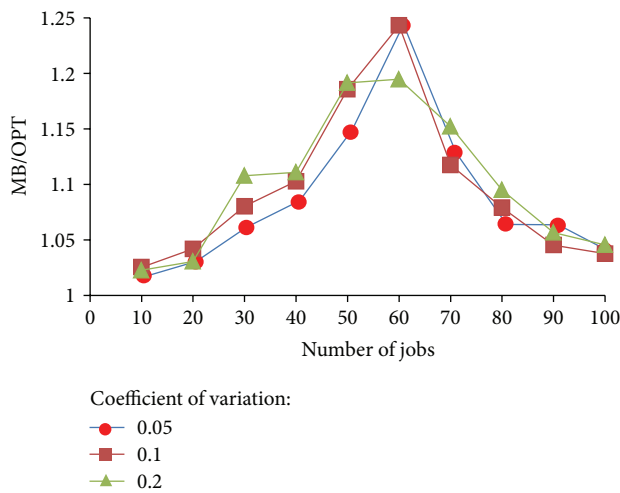


FIGURE 7: ρ versus the number of jobs and the coefficient of variation.

The implementation of the proposed method was demonstrated using a numerical example. The performance evaluation of the proposed method has been done with comparing to a lower bound of optimum. Primary results indicated that the proposed method had the approximation factor less than 2 for a wide range of problems. The proposed method utilized the properties of the normal distributions, and this method can be used as a heuristic method for other distributions, as long as their means and variances are available. Scheduling with preemption penalties is a novel research area in scheduling field especially for online stochastic problems and we strongly believe that this research could be well extended for problems with stochastic setup times, other distribution function for processing time, and flowshop problem with more than 2 machines.

References

[1] F. M. Defersha and M. Chen, "Mathematical model and parallel genetic algorithm for hybrid flexible flowshop lot streaming problem," *International Journal of Advanced Manufacturing Technology*, vol. 62, no. 1-4, pp. 249-265, 2012.

[2] M. Mahavi Mazdeh, F. Zaerpour, and F. Firouzi Jahantigh, "A fuzzy modeling for single machine scheduling problem with deteriorating jobs," *International Journal of Industrial Engineering Computations*, vol. 1, no. 2, pp. 147-156, 2010.

[3] M. Braglia, M. Frosolini, R. Gabbriellini, and F. Zammori, "CONWIP card setting in a flow-shop system with a batch production machine," *International Journal of Industrial Engineering Computations*, vol. 2, no. 1, pp. 1-18, 2011.

[4] S. M. Johnson, "Optimal two- and three-stage production schedules with setup times included," *Naval Research Logistics Quarterly*, vol. 1, no. 1, pp. 61-68, 1954.

[5] M. R. Garey, D. S. Johnson, and R. Sethi, "The complexity of flowshop and jobshop scheduling," *Mathematics of Operations Research*, vol. 1, no. 2, pp. 117-129, 1976.

[6] J. M. Framinan, J. N. D. Gupta, and R. Leisten, "A review and classification of heuristics for permutation flow-shop scheduling with makespan objective," *Journal of the Operational Research Society*, vol. 55, no. 12, pp. 1243-1255, 2004.

[7] R. Ruiz and C. Maroto, "A comprehensive review and evaluation of permutation flowshop heuristics," *European Journal of Operational Research*, vol. 165, no. 2, pp. 479-494, 2005.

[8] T. Makino, "On a scheduling problem," *Journal of the Operations Research Society Japan*, vol. 8, pp. 32-44, 1965.

[9] E. Frostig and I. Adiri, "Three-machine flowshop stochastic scheduling to minimize distribution of schedule length," *Naval Research Logistics Quarterly*, vol. 32, no. 1, pp. 179-183, 1985.

[10] S. Sethi, H. Yan, Q. Zhang, and X. Y. Zhou, "Feedback production planning in a stochastic two-machine flowshop: asymptotic analysis and computational results," *International Journal of Production Economics*, vol. 30-31, pp. 79-93, 1993.

[11] S. E. Elmaghraby and K. A. Thoney, "The two-machine stochastic flowshop problem with arbitrary processing time distributions," *IIE Transactions*, vol. 31, no. 5, pp. 467-477, 1999.

[12] P. P. Talwar, "A note on sequencing problems with uncertain job times," *Journal of the Operations Research Society of Japan*, vol. 9, pp. 93-97, 1967.

[13] A. A. Cunningham and S. K. Dutta, "Scheduling jobs, with exponentially distributed processing times, on two machines of a flow shop," *Naval Research Logistics Quarterly*, vol. 20, pp. 69-81, 1973.

[14] P. S. Ku and S. C. Niu, "On Johnson's two-machine flow shop with random processing times," *Operations Research*, vol. 34, no. 1, pp. 130-136, 1986.

- [15] H. M. Soroush and A. Allahverdi, "Stochastic two-machine flowshop scheduling problem with total completion time criterion," *International Journal of Industrial Engineerin*, vol. 12, no. 2, pp. 159–171, 2005.
- [16] D. Laha and U. K. Chakraborty, "An efficient stochastic hybrid heuristic for flowshop scheduling," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 6, pp. 851–856, 2007.
- [17] V. Portugal and D. Trietsch, "Johnson's problem with stochastic processing times and optimal service level," *European Journal of Operational Research*, vol. 169, no. 3, pp. 751–760, 2006.
- [18] P. J. Kalczynski and J. Kamburowski, "A heuristic for minimizing the expected makespan in two-machine flow shops with consistent coefficients of variation," *European Journal of Operational Research*, vol. 169, no. 3, pp. 742–750, 2006.
- [19] K. R. Baker and D. Altheimer, "Heuristic solution methods for the stochastic flow shop problem," *European Journal of Operational Research*, vol. 216, no. 1, pp. 172–177, 2012.
- [20] K. R. Baker and D. Trietsch, "Three heuristic procedures for the stochastic two-machine flowshop problem," *Journal of Scheduling*, vol. 14, no. 5, pp. 445–454, 2011.
- [21] M. Heydari, M. Mahavi Mazdeh, and M. Bayat, "Stochastic online scheduling with preemption penalties," *Journal of Mathematics and Computer Science*, vol. 6, pp. 238–250, 2013.
- [22] K. R. Baker and D. Trietsch, *Principles of Sequencing and Scheduling*, John Wiley & Sons, Hoboken, NJ, USA, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

