Management Science and Financial Engineering
Vol 21, No 1, May 2015, pp.1-9
ISSN 2287-2043 | EISSN 2287-2361 |
http://dx.doi.org/10.7737/MSFE.2015.21.1.001

Min-Max Regret Version of an *m*-Machine Ordered Flow Shop with Uncertain Processing Times

Myoung-Ju Park
Department of Industrial and Management Systems Engineering, Kyung Hee University
Byung-Cheon Choi*
School of Business, Chungnam National University

(Received: April 29, 2015 / Accepted: May 8, 2015)

ABSTRACT

We consider an m-machine flow shop scheduling problem to minimize the latest completion time, where processing times are uncertain. Processing time uncertainty is described through a finite set of processing time vectors. The objective is to minimize maximum deviation from optimality for all scenarios. Since this problem is known to be NP-hard, we consider it with an ordered property. We discuss optimality properties and develop a pseudo-polynomial time approach for the problem with a fixed number of machines and scenarios. Furthermore, we find two special structures for processing time uncertainty that keep the problem NP-hard, even for two machines and two scenarios. Finally, we investigate a special structure for uncertain processing times that makes the problem polynomially solvable.

Keywords: Scheduling, Flow Shop, Uncertainty, Computational Complexity

* Corresponding Author, E-mail: polytime@cnu.ac.kr

1. INTRODUCTION

Many scheduling problems assume that data associated with a job are known in advance. In practice, however, the data cannot be specified exactly for many reasons, such as the introduction of a new machine or machine failure, which especially affect processing times. To reflect this reality, this paper considers a scheduling problem with uncertain processing times. The uncertainty of job processing times is structured through the concept of a *scenario* that corresponds to the assignment of plausible values to processing times. The objective is to minimize maximum deviation from optimality over all scenarios, referred to as the min-max regret version.

Daniels and Kouvelis (1995) have introduced two ways of describing the set of scenarios. The *interval* scenario case assumes that a processing time exists within its own lower and upper bounds, independent of other processing times, whereas the *discrete* scenario case assumes a finite set of processing time vectors, each specifying the processing time of each job. This paper considers only the discrete scenario case.

We consider an *m*-machine flow shop scheduling problem, in which the performance measure is the makespan. Let $J=\{1, 2, \cdots, n\}$ be a set of n independent jobs. The uncertainty for the processing times is described through the scenario set $S=\{1, 2, \cdots, v\}$. For each scenario $s \in S$, let $\left(p_{1,1}^s, p_{1,2}^s, \cdots, p_{m,n}^s\right)$ be the vector of processing times under scenario s, where $p_{i,j}^s$ is the processing time of job j on machine i under scenario s. Let $\sigma=\left(\sigma(1), \sigma(2), \cdots, \sigma(n)\right)$ be the schedule, where $\sigma(j)$ is the j-th job in σ. Let $C_{i,j}^s(\sigma)$ be the completion time of job j on machine i and $C_{max}^s(\sigma)=C_{m,\sigma}^s(\sigma)$ be the makespan of σ under scenario s. Our objective is to find an optimal schedule σ^* to minimize

$$z(\sigma^*) = max_{s \in S} \left\{ C_{max}^s(\sigma^*) - C_{max}^s(\sigma_s^*) \right\},$$

where σ_s^* is a schedule to minimize the makespan under scenario s. Since this problem is known to be NP-hard (Kouvelis and Daniels, 2000), however, we consider the problem satisfying *ordered properties*, defined as follows: Under each scenario $s \in S$,

- If, for some $i \in \{1, 2, \cdots, m\}$, $p_{i,j}^s \le p_{i,j'}^s$, then $p_{i',j}^s \le p_{i',j'}^s$, $i' = 1, 2, \cdots, m$;
- If, for some $j \in \{1, 2, \cdots, n\}$, $p_{i,j}^s \le p_{i',j}^s$, then $p_{i,j'}^s \le p_{i',j'}^s$, $j' = 1, 2, \cdots, n$.

Let the problem with ordered properties be referred to as *Problem OP*. Without loss of generality, henceforth, assume that for each scenario $s \in S$,

$$p_{i,1}^s \le p_{i,2}^s \le \cdots \le p_{i,n}^s, \ i = 1, 2, \cdots, m.$$

Furthermore, we consider three special cases of Problem OP, where

- The processing time of each job on each machine is represented by a function of two parameters (one depends on the job and the other depends on the machine);
- The job-depend parameter is certain, but the machine-depend parameter is uncertain;
- That is, $p_{i,j}^s = f(p_j, q_i^s)$ where *f* is some function, p_j is the processing requirement of job *j*, and q_i^s is the characteristic value of machine *i* under scenario *s*.

First, we consider the case with $p_{i,j}^s = p_j / q_i^s$, where q_i^s means the speed of machine *i* under scenario *s*. In this case, the processing times are inversely proportional to the machine speeds. Let this case be referred to as *Problem OP-Divide*. In the second case, let $p_{i,j}^s = \max\{p_j, q_i^s\}$, where q_i^s can be interpreted as the time associated with machine *i*, such as in setting or fine-tuning the machine. In this case, both processing jobs and the setting machine can start simultaneously. Let the second case be referred to as *Problem OP-Max*. In the third case, let $p_{i,j}^s = p_j + q_i^s$, respectively, where q_i^s can be interpreted as the time associated with machine *i*, such as in setting or fine-tuning the machine. In this case, processing jobs can start after machine set-up is completed. Let the third case be referred to as *Problem OP-Plus*.

To the best of our knowledge, the ordered flow shop problem with uncertain processing times has not yet been studied.

The rest of the paper is organized as follows. Section 2 discusses the related literature. Section 3 presents some optimality conditions and a pseudo-polynomial time approach for Problem OP with a fixed number of machines and scenarios. Section 4 shows the NP-hardness of Problems OP-Divide and OP-Max and the polynomiality of Problem OP-Plus. Finally, Section 5 discusses our conclusions and future work.

2. LITERATURE REVIEW

Johnson (1954) presented a seminal algorithm, referred to as *Johnson's rule*, that efficiently solves the problem of minimizing the makespan in a two-machine flow shop. Since the problem of minimizing the makespan in a flow shop with more than two machines is strongly NP-hard (Garey and Johnson, 1976), however, researchers have considered a flow shop with special structures for processing times, such as the ordered flow shop (Smith *et al*., 1975, 1976) and the proportionate flow shop (Allahverdi, 1999; Hou and Hoogeveen, 2003; Ow, 1985).

Smith *et al*. (1975, 1976) considered an *m*-machine ordered flow shop. Smith *et al*. (1976) showed that an optimal schedule has an *inverted V-shape*, which is defined as a sequence in which all jobs scheduled before the job with the longest processing time are in shortest processing time (SPT) order and all jobs scheduled after the job with the longest processing time are in longest processing time (LPT) order. Smith *et al*. (1975) considered the case where the job processing times on the first (last) machine are the longest and showed that the problem can then be solved in polynomial time. They showed that if the first (last) machine is the slowest, the LPT (SPT) order is an optimal permutation schedule.

In a proportionate flow shop environment where the processing times of job *j* on all *m* machines are equal to p_j (Ow, 1985; Pinedo, 2006), the makespan is a constant, independent of the sequence. The proportionate flow shop model can be generalized by describing the processing time on each machine as a function of a processing requirement and a machine-dependent parameter q_i. Hou and Hoogeveen (2003) and Choi *et al*. (2007) showed that the problem with $p_{i,j}^s = p_j / q_i^s$ is NP-hard even for the three-machine case. Choi *et al*. (2007) proposed a heuristic for which a tight worst-case bound of 3/2 was demonstrated by Koulamas and Kyparisis (2009). Choi *et al*. (2010) showed that the problem with $p_{i,j}^s = \max\{p_j, q_i^s\}$ is NP-hard even for the three-machine case, and the makespan of the problem with $p_{i,j}^s = \min\{p_j, q_i^s\}$ is a constant, independent of the sequence. Choi *et al*. (2011) considered permutation schedules and showed that the problem with $p_{i,j}^s = p_j + q_i^s$ is polynomially solvable, when the number of machines is fixed.

Scheduling problems under scenario-based uncertainty have been studied extensively (Aissi *et al*., 2007, 2009, 2011; Johnson, 1954; Kouvelis and Yu, 1997). Daniel and Kouvelis (1995) considered a min-max regret version of the single-machine scheduling problem with uncertain processing times using the total completion time as the performance measure. They showed that the problem is NP-hard and developed exact and heuristic algorithms to solve it. Yang and Yu (2002) later showed that a min-max version of the same problem is NP-hard and presented a dynamic programming method and two heuristic algorithms where min-max version minimizes maximum cost over all scenarios. Aloulou and Croce (2008) considered a min-max version of the single-machine scheduling problem with uncertain due dates, weights, and processing times. The performance measures considered were total weighted completion times, the general maximum cost function, and the number of late jobs. The authors established the computational complexity for various combinations of uncertain job parameters and performance measures except for the case with the uncertain due dates and the number of late jobs as the performance measure. Aissi *et al.* (2011) later showed the case to be NP-hard. Kouvelis *et al.* (2000) and Kasperski *et al.* (2012) considered a min-max regret version of the two-machine flow shop scheduling problem with uncertain processing times, using the makespan as the performance measure of interest. Kouvelis *et al.* (2000) proved NP-hardness and developed exact and heuristic methods. Kasperski *et al.* (2008) showed strong NP-hardness and established approximability.

3. EXACT ALGORITHM FOR PROBLEM OP

This section investigates optimality conditions and develops a pseudo-polynomial time approach for Problem OP with a fixed number of machines and scenarios.

Observation 1: Smith *et al.* (1976) Given a schedule σ of an ordered flow shop with deterministic processing times (that is, there is a single scenario s), let σ' be a schedule constructed from σ by sequencing the jobs before (after) job n in SPT (LPT) order. Then, we have

$$C^s_{max}(\sigma') \le C^s_{max}(\sigma).$$

Theorem 1: In Problem OP, there exists an optimal schedule that has an inverted V-shape.

Proof: It immediately holds from Observation 1. ■

By Theorem 1, henceforth, we consider only the schedules with an inverted V-shape. Now, we reduce Problem OP with a fixed number of machines and scenarios to the shortest path problem in pseudo-polynomial time. We construct a schedule backwards from job n, by placing job $n-1$ either before or after job n, and likewise for the others. By Theorem 1, this construction is enough to find determine an optimal schedule.

Let h_s be the slowest machine under scenario s and let $P = max_{s \in S} \sum_{j=1}^{n} p^s_{h_s,j}$. For each $k, l \le n$ and L^s_i, $R^s_i \le P$, let $N(k,\ l;\ ((L^s_i)_{i=1,\cdots,h_s-1}, (R^s_i)_{i=h_s+1,\cdots,m})_{s\in S}$ denote a node such that

- Jobs in $\{n, n-1, \cdots, k\}$ have been scheduled already;
- l is the number of jobs positioned strictly before job n in $\{n-1, n-2, \cdots, k\}$;
- For each $s \in S$, L^s_i is the difference between the start times of the first job on machine i and i+1, i = 1, 2, $\cdots$, h_s-1;
- For each $s \in S$, R^s_i is the difference between the completion times of the last job on machine i-1 and i, i = h_s+1, $\cdots$, m.

Let $N(n,\ 0;\ ((p^s_{i,n})_{i=1,\cdots,h_s-1}, (p^s_{i,n})_{i=h_s+1,\cdots,m})_{s\in S})$ and t be the source and sink nodes, respectively. For $k = n-1, \cdots,$ 1, let $N(k+1,\ l;\ ((L^s_i)_{i=1,\cdots,h_s-1}, (R^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$ be connected to two nodes with weight 0 as follows. First, let it be connected to $N(k,\ l;\ ((L^s_i)_{i=1,\cdots,h_s-1}, (R^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$ such that for each $s \in S$,

$$\bar{R}^s_{h_s+1} = \begin{cases} R^s_{h_s+1} - p^s_{h_s,k} + p^s_{h_s+1,k}, & if\ \ R^s_{h_s+1} > p^s_{h_s,k} \\ p^s_{i,k} & otherwise \end{cases}$$

and for $i = h_s$+2, $\cdots$, m,

$$\bar{R}^s_i = \begin{cases} \sum_{g=h_s+1}^{i} R^s_g - \sum_{g=h_s+1}^{i-1} \bar{R}^s_g - p^s_{h_s,k} + p^s_{i,k}, & \\ & if\ \ \sum_{g=h_s+1}^{i} R^s_g > \sum_{g=h_s+1}^{i-1} \bar{R}^s_g + p^s_{h_s,k} \\ p^s_{i,k} & otherwise \end{cases}$$

Note that $\bar{R}^s_i$ is recursively calculated from i = h_s+1 to m. Second, let it be connected to $N(k,\ l+1;\ ((\bar{L}^s_i)_{i=1,\cdots,h_s-1}, (R^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$ such that for each $s \in S$,

$$\bar{L}^s_{h-1} = \begin{cases} L^s_{h_s-1} - p^s_{h_s,k} + p^s_{h_s-1,k}, & if\ \ L^s_{h_s-1} > p^s_{h_s,k} \\ p^s_{i,k} & othewise \end{cases}$$

and for $i = h_s - 2, \cdots, m-$,

$$\bar{L}^s_i \begin{cases} \sum_{g=i}^{h_s-1} L^s_g - \sum_{g=i+1}^{h_s-1} \bar{L}^s_g - p^s_{h_s,k} + p^s_{i,k}, & \\ & if\ \ \sum_{g=i}^{h_s-1} L^s_g > \sum_{g=i+1}^{h_s-1} \bar{L}^s_g + p^s_{h_s,k} \\ p^s_{i,k} & otherwise \end{cases}$$

Note that $\bar{L}^s_i$ is recursively calculated from $i = h_s$-1 to 1. For l = 0, 1, $\cdots$, n-1, let $N(1,\ l;\ ((L^s_i)_{i=1,\cdots,h_s-1}, (R^s_i)_{i=h_s+1,\cdots,m})$ be connected to the sink node with weight

$$max_{s\in S} \left\{ \sum_{j=1}^{n} p^s_{h_s,j} + \sum_{i=1}^{h_s-1} L^s_i + \sum_{i=h_s+1}^{m} R^s_i - C^s_{max}(\sigma^*_s) \right\}$$

Note that when all jobs are determined, $\sum_{j=1}^{n} p^s_{h_s,j} + \sum_{i=1}^{h_s-1} L^s_i + \sum_{i=h_s+1}^{m} R^s_i$ is the makepspan under scenario s, and σ^*_s can be obtained in polynomial time by the algorithm in (Choi *et al.*, 2011). The objective is to find the shortest path between the source and the sink nodes. It is clear that the shortest path in the reduced graph represents the optimal schedule of Problem OP.

Theorem 2: Problem OP can be solved in $O(n^2P^{(m-1)v})$.

Proof: The number of nodes in the reduced graph is $O(n^2P^{(m-1)v})$. Since the number of edges emanating from each node is at most two, the total number of edges is at most $O(n^2P^{(m-1)v})$ We can use the algorithm of Ahuja *et al*. (1990) to obtain the shortest path problem between the source and the sink nodes. The running time of the algorithm becomes $O(n^2P^{(m-1)v})$. ■

4. COMPUTATIONAL COMPLEXITY OF PROBLEMS OP-DIVIDE, OP-MAX AND OP-PLUS

This section establishes the computational complexity of Problems OP-Divide, OP-Max, and OP-Plus.

4.1 NP-hardness of Problems OP-Divide and OP-Max

This subsection shows that Problems OP-Divide and OP-Max are NP-hard even for the case with two machines and two scenarios.

Theorem 3: Problem OP-Divide with two machines and two scenarios is NP-hard.

Proof: The proof is given in Appendix A. ■

Theorem 4: Problem OP-Max with two machines and two scenarios is NP-hard.

Proof: The proof is given in Appendix B. ■

Corollary 1: Problem OP with a fixed number of machines and scenarios is NP-hard in the ordinary sense.

Proof: It immediately holds from Theorems 1 and 3 (or 4). ■

4.2 Polynomiality of Problem OP-Plus

This subsection shows that Problem OP-Plus with a fixed number of machines and scenarios is polynomially solvable. To derive its polynomiality, we use $N=(k, l; ((L_i^s)_{i=1,\cdots,h_s-1}, (R_i^s)_{i=h_s+1,\cdots,m})_{s\in S}$ defined in the Section 3, and replace L_i^s and R_i^s with different information. Let $\sigma=(\sigma'_L, n, \sigma'_R)$ be the subschedule corresponding to this node, where $\sigma'_L(\sigma'_R)$ is the subschedule consisting of jobs sequenced before (after) job n. For simplicity of notations, let $\sigma_L=(\sigma'_L, n)$ and $\sigma_R=(n, \sigma'_R)$. Let z_1 and z_2 be the first and last jobs in σ, respectively. Let $C^s_{max,i}(\sigma_R)$ be the completion time of job z_2 on machine i under scenario s in σ_R. Note that for $i=h_s+1,\cdots,m$,

$$R_i^s = C^s_{max,i}(\sigma_R) - C^s_{max,i-1}(\sigma_R)$$

For $i=h_s+1,\cdots,m$, let $\beta_i^s=\left(\beta^s_{i,h_s},\cdots,\beta^s_{i,i-1}\right)$ be the vector of the jobs for σ_R under scenario s, which determines the completion of z_2 on machine i under scenario s. For $g=h_s,\cdots,i-1$, and $i=h_s+1,\cdots,m$, let $b^s_{i,g}$ be the number of jobs positioned after job $\beta^s_{i,g}$, and $B^s_{i,g}$ be the set of jobs from job $\beta^s_{i,g-1}$ to $\beta^s_{i,g}$ where $\beta^s_{i,h_s-1}=n$ and $\beta^s_{i,i}=z_2$. Note that $\left|B^s_{i,g}\right|=b^s_{i,g-1}-b^s_{i,g}+1$. Then,

$$\begin{aligned}C^s_{max,i}(\sigma_R) &= \sum_{g=h_s}^{i}\sum_{j\in B^s_{i,g}} p^s_{g,j}\\ &= \sum_{j\in\sigma_R} p_j + \sum_{g=h_s}^{i-1} p_{\beta^s_{i,g}} + \sum_{g=h_s}^{i}(b^s_{i,g-1}-b^s_{i,g}+1)q^s_g\end{aligned}$$

Thus,

$$\begin{aligned}R_i^s = &\sum_{g\in h_s}^{i-1} p_{\beta^s_{i,g}} + \sum_{g=h_s}^{i}(b^s_{i,g-1}-b^s_{i,g}+1)q^s_g\\ &-\sum_{g\in h_s}^{i-2} p_{\beta^s_{i-1,g}} - \sum_{g=h_s}^{i-1}(b^s_{i-1,g-1}-b^s_{i-1,g}+1)q^s_g\end{aligned}$$

Hence, we can use the information of $\beta^s_{i,g}$ and b_i^s instead of R_i^s, $i=h_s+1,\cdots,m$.

For the consistency of notations, without loss of generality, assume that jobs before job n are processed from machines h_s to 1. In this case, job z_1 becomes the last job in σ_L. Let $C^s_{max,i}(\sigma_L)$ be the completion time of job z_1 on machine i under scenario s in σ_L. Note that for $i=h_s-1,\cdots,1$,

$$L_i^s = C^s_{max,i}(\sigma_L) - C^s_{max,i+1}(\sigma_L).$$

Similarly, for $i=h_s-1,\cdots,1$, let $\alpha_i^s=(\alpha^s_{i,i},\cdots,\alpha^s_{i,h_s-1})$ be the vector of the jobs for σ_L under scenario s, which determines the completion of z_2 on machine i under scenario s. For $g=i,\cdots,h_s-1$ and $i=h_s-1,\cdots,1$, let $\alpha^s_{i,g}$ be the number of jobs positioned after job $\alpha^s_{i,g}$, and $A^s_{i,g}$ be the set of jobs from job $\alpha^s_{i,g-1}$ to $\alpha^s_{i,g}$ where $\alpha^s_{i,g-1}=z_1$ and $\alpha^s_{i,h_s}=n$. Note that $\left|A^s_{i,g}\right|=a^s_{i,g}-a^s_{i,g-1}+1$. Then,

$$\begin{aligned}C^s_{max,i}(\sigma_L) &= \sum_{g=i}^{h_s}\sum_{j\in A^s_{i,g}} p^s_{g,j}\\ &= \sum_{j\in\sigma_L} p_j + \sum_{g=i}^{h_s-1} p_{\alpha^s_{i,g}} + \sum_{g=i}^{h_s}(a^s_{i,g}-a^s_{i,g-1}+1)q^s_g\end{aligned}$$

Thus,

$$L_i^s = \sum_{g=i}^{h_s-1} p_{\alpha^s_{i,g}} + \sum_{g=i}^{h_s}(a^s_{i,g}-a^s_{i,g-1}+1)q^s_g$$

$$-\sum_{g=i+1}^{h_s-1} p_{\alpha^s_{i+1,g}} - \sum_{g=i+1}^{h_s} (a^s_{i+1,g} - a^s_{i+1,g-1} + 1) q^s_g$$

Hence, we can use α^s_i and a^s_i instead of L^s_i for $i = 1, \cdots, h_s - 1$. Thus, instead of $N(k, l; ((L^s_i)_{i=1,\cdots,h_s-1}, (R^s_i)_{i=h_s-1}, (R^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$, we can use

$$N(k, l; ((\alpha^s_i, a^s_i)_{i=1,\cdots,h_s-1}, (\beta^s_i, b^s_i)_{i=h_s+1,\cdots,m})_{s\in S}),$$

For $k = n-1, \cdots, 1$, let $N(k+1, l; ((\alpha^s_i, a^s_i)_{i=1,\cdots,h_s-1}, (\beta^s_i, b^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$ be connected to two nodes with weight 0 as follows. First, let it be connected to $N(k, l; ((\bar{\alpha}^s_i, \bar{a}^s_i)_{i=1,\cdots,h_s-1}, (\bar{\beta}^s_i, \bar{b}^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$, such that $\bar{\alpha}^s_i = \alpha^s_i$ and $\bar{a}^s_i = a^s_i$ for $i = 1, \cdots, h_s - 1$ and

$$\begin{cases} \bar{\beta}^s_i = \beta^s_i \text{ and } \bar{b}^s_i = b^s_i + \mathbf{1}^s_i & \text{if } \sum_{g=h_s+1}^{i} R^s_g > \sum_{g=h_s+1}^{i-1} \bar{R}^s_g + p^s_{h_s,k} \\ \bar{\beta}^s_i = \left(\bar{\beta}^s_{i-1}, k\right) \text{ and } \bar{b}^s_i = (b^s_{i-1}, 0) & \text{otherwise} \end{cases}$$

where $\mathbf{1}^s_i$ is the $i - h_s$ dimensional one vector for $i = h_s + 1, \cdots, m$. Second, let it be connected to $N(k, l+1; ((\bar{\alpha}^s_i, \bar{a}^s_i)_{i=1,\cdots,h_s-1}, (\bar{\beta}^s_i, \bar{b}^s_i)_{i=h_s+1,\cdots,m})_{s\in S})$, such that $\bar{\beta}^s_i = \beta^s_i$ and $\bar{b}^s_i = b^s_i$ for $i = h_s + 1, \cdots, m$ and

$$\begin{cases} \bar{\alpha}^s_i = \alpha^s_i \text{ and } \bar{a}^s_i = a^s_i + \mathbf{1}^s_i & \text{if } \sum_{g=1}^{h_s-1} L^s_g > \sum_{g=i+1}^{h_s-1} \bar{L}^s_g + p^s_{h_s,k} \\ \bar{\alpha}^s_i = \left(k, \bar{\alpha}^s_{i-1}\right) \text{ and } \bar{a}^s_i = (0, \bar{a}^s_{i-1}) & \text{otherwise} \end{cases}$$

where $\mathbf{1}^s_i$ is the $h_s - i$ dimensional one vector for $i = h_s - 1, \cdots, 1$.

Theorem 5: Problem OP-Plus with a fixed number of machines and scenarios is polynomially solvable.

Proof: Since we use (α^s_i, a^s_i) instead of L^s_i for $i = 1, \cdots, h_s - 1$ and (β^s_i, b^s_i) instead of R^s_i for $i = h_s + 1, \cdots, m$ during the reduction, the number of nodes is

$$2 + \sum_{s\in S} \left\{ h_s(h_s - 1) + (m - h_s)(m - h_s + 1) \right\},$$

where is less than or equal to $m(m-1)v + 2$. Thus, the running time to solve the reduced shortest path problem is $O(n^{m(m-1)v+2})$. ■

Remark 1: Theorems 1-5 hold for the min--max version of Problem OP by replacing $C^s_{max}(\sigma^*_s)$ with 0.

5. CONCLUDING REMARKS AND FUTURE WORKS

We consider the min-max regret version of an *m*-machine ordered flow shop scheduling problem with uncertain processing times where the performance measure is the makespan. The uncertainty for the processing time is described through a set of discrete processing time scenarios. We show that the problem with a fixed number of machines and scenarios is NP-hard in the ordinary sense, and furthermore, the problem with two machines and two scenarios remains NP-hard, even if $p^s_{i,j} = p_j / q^s_i$ or $p^s_{i,j} = \max\left\{p_j, q^s_i\right\}$. Finally, we present the polynomial time approach to solve the problem with a fixed number of machines and scenarios if $p^s_{i,j} = p_j + q^s_i$.

For future research, it would be interesting to consider the interval scenario version of our problem.

REFERENCES

Ahuja, R. A., K. Mehlhorn, and J. B. Orlin, "Faster algorithm for the shortest path problem," *Journal of the Association for Computing Machinery* 37 (1990), 213-223.

Aissi, H., M. A. Alouloua, and M. Y. Kovalyov, "Minimizing the number of late jobs on a single machine under due date uncertainty," *Journal of Scheduling* 14 (2011), 351-360.

Aissi, H., C. Bazgan, and D. Vanderpooten, "Approximation of min-max and min-max regret versions of some combinatorial optimization problems," *European Journal of Operational Research* 179 (2007), 281-290.

Aissi, H., C. Bazgan, and D. Vanderpooten, "Min-max and min-max regret versions of combinatorial optimization problems: A survey," *European Journal of Operational Research* 197 (2009), 427-438.

Allahverdi, A., "Two-machine proportionate flowshop scheduling with breakdown to minimize maximum lateness," *Computers and Operations Research* 23 (1996), 909-916.

Aloulou, M. A. and F. D. Croce, "Complexity of single machine scheduling problems under scenario-based uncertainty," *Operations Research Letters* 36 (2008), 338-342.

Choi, B. C., J. Y. T. Leung, and M. L. Pinedo, "A note on makespan minimization in proportionate flow shops," *Information Processing Letters* 111 (2010), 77-81.

Choi, B. C., J. Y. T. Leung, and M. L. Pinedo, "Minimizing the makespan in a proportionate flow shop with machine-dependent processing times," *Journal of Combinatorial Optimization* 22 (2011), 797-810.

Choi, B. C., S. H. Yoon, and S. J. Chung, "Minimizing maximum completion time in a proportionate flow shop with one machine of different speed," *European Journal of Operational Research* 176 (2007), 964-974.

Daniels, R. L. and P. Kouvelis, "Robust scheduling to hedge against processing time uncertainty in single-stage production," *Management Science* 41 (1995), 363-376.

Garey, M. R. and D. S. Johnson, "The complexity of flowshop and jobshop scheduling," *Mathematics of Operations Research* 1 (1976), 117-129.

Hou, S. and H. Hoogeveen, "The three-machine proportionate flow shop with unequal machine speeds," *Operations Research Letters* 31 (2003), 225-231.

Johnson, S. M., "Optimal two- and three-stage production schedules with setup times included," *Naval Research Logistics Quarterly* 1 (1954), 62-67.

Kasperski, A., A. Kurpisz, and P. Zielinski, "Approximating a two-machine flow shop scheduling under discrete scenario uncertainty," *European Journal of Operational Research* 217 (2012), 36-43.

Kasperski, A. and P. Zeilinski, "A 2-approximation algorithm for interval data minmax regret sequencing problems with the total flow time criterion," *Operations Research Letters* 36 (2008), 343-344.

Koulamas, C. and G. J. Kyparisis, "A note on the proportional flow shop with a bottleneck machine," *European Journal of Operational Research* 193 (2009), 644-645.

Kouvelis, P. and R. L. Daniels, "Robust scheduling of a two machine flow shop with uncertain processing times," *IIE Transactions* 32 (2000), 421-432.

Kouvelis, P. and U. Yu, "Robust discrete optimization and its applications," *Kluwer Academic Publisher*, 1997.

Ow, P. S., "Focused scheduling in proportionate flow shops," *Management Science* 31 (1985), 851-869.

Pinedo, M. L., "Scheduling: Theory, Algorithms and Systems," *Prentice Hall*, 2006.

Smith, M. L., S. S. Panwalkar, and R. A. Dudek, "Flow shop sequencing problem with ordered processing time matrices," *Management Science* 21 (1975), 544-549.

Smith, M. L., S. S. Panwalkar, and R. A. Dudek, "Flow shop sequencing problem with ordered processing time matrices: A general case," *Naval Research Logistics Quarterly* 23 (1976), 481-486.

Yang, J. and G. Yu, "On the robust single machine scheduling problem," *Journal of Combinatorial Optimization* 6 (2002), 17-33.

Appendix A: Proof of Theorem 3

We reduce the partition problem, defined below, to Problem OP-Divide: Given n integers $\{a_1, a_2, \cdots, a_n\}$ such that $\sum_{j=1}^{n} a_j = 2A,$ is there a set B such that $\sum_{j\in B} a_j = A$?

Without loss of generality, assume that all integers in $\{a_1, a_2, \cdots, a_n\}$ are the multiples of 10.

Given an instance of the partition problem, we can construct an instance of Problem OP-Divide with two machines and two scenarios as follows:

- $p_j = a_j,\ j = 1, 2, \cdots, n,\ p_{n+1} = A^2$ and $p_{n+2} = p_{n+3} = 1;$
- $q_1^1 = 1/A$ and $q_2^1 = 1;$
- $q_1^2 = 1$ and $q_2^2 = 1/A.$

Clearly, this reduction can be done in polynomial time. We, henceforth, will show that there is a schedule $\overline{\sigma}$ such that $z(\overline{\sigma}) = 0$ if and only if there is a solution to the partition problem. It is observed from this reduction that

- The optimal schedules under scenarios 1 and 2 are LPT and SPT orders, respectively;
- Since integers in $\{a_1, a_2, \cdots, a_n\}$ are the multiples of 10, $C_{max}^1(\sigma_1^*)$ and $C_{max}^2(\sigma_2^*)$ can be described as follows:

$$C_{max}^1(\sigma_1^*) = C_{max}^2(\sigma_2^*) = A^3 + 2A^2 + 2A + 1.$$

Suppose that there exists a set $\overline{B}$ in the partition problem such that $\sum_{j\in\overline{B}} a_j = A.$ Then we can construct a new schedule $\overline{\sigma} = (n+2,\ \sigma_{\overline{B}},\ n+1,\ \sigma_{\overline{B}'},\ n+3),$ where $\sigma_{\overline{B}}$ is the SPT subschedule of jobs in $\overline{B}$ and $\sigma_{\overline{B}'}$ is the LPT subschedule of jobs in $\{1, 2, \cdots, n\} \setminus \overline{B}.$ Note that $\sum_{j\in\overline{B}} p_j = \sum_{j\in\overline{B}'} p_j = A.$ Under scenario 1, since $\sum_{j\in\overline{B}'} p_j = p_{n+2} = A^2$ and $p_{1,n+3} = \sum_{j\in\overline{B}'} p_{2,j} = A,$ schedule $\overline{\sigma}$ can be described as Figure 1 below.

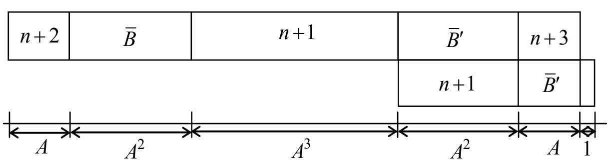

Figure 1. The Makespan of $\overline{\sigma}$ under Scenario 1

Thus, the makespan of $\overline{\sigma}$ under scenario 1 is

$$\begin{aligned} &p_{1,n+2} + \sum_{j\in\overline{B}} p_{1,j} + p_{1,n+1} + \sum_{j\in\overline{B}'} p_{1,j} + p_{1,n+3} + p_{2,n+3} \\ &= A^3 + 2A^2 + 2A + 1. \end{aligned}$$

Since $q_1^1 = q_2^2$ and $q_2^1 = q_1^2$, it is observed from the reversibility of flow shops that $C_{max}^1(\overline{\sigma}) = C_{max}^2(\overline{\sigma}).$ Thus, since $C_{max}^1(\overline{\sigma}) = C_{max}^2(\overline{\sigma}) = C_{max}^1(\sigma_1^*) = C_{max}^2(\sigma_2^*),$

$$z(\overline{\sigma}) = max_{s=1,2}\left\{C_{max}^s(\overline{\sigma}) - C_{max}^s(\sigma_s^*)\right\} = 0.$$

Suppose that there exists a schedule $\hat{\sigma} = (\sigma_{B_1}, n+1, \sigma_{B_2})$ such that

- $z(\hat{\sigma}) = 0;$
- $B_1(B_2)$ is the set of jobs positioned before (after) job $n+1;$
- By Theorem 1, σ_{B_1} is the SPT subschedule of jobs in B_1 and σ_{B_2} is the LPT subschedule of jobs in $\{1, 2, \cdots, n+3\} \setminus B_1.$

Note that since $C_{max}^1(\sigma_1^*)$ and $C_{max}^2(\sigma_2^*)$ have the constant term 1 and integers in $\{p_1, p_2, \cdots, p_n\}$ are all multiples of 10,

- $n+2 \in B_1$ and $n+3 \in B_2$, or $n+2 \in B_2$ and $n+3 \in B_1.$

Without loss of generality, henceforth, we assume that $n+2 \in B_1$ and $n+3 \in B_2$ in $\hat{\sigma}.$ Then, we can show $\sum_{j\in B_1} p_j = A+1$ which implies $B_1 \setminus \{n+2\}$ is a solution of partition problem. Consider two cases.

Case 1: $\sum_{j\in B_1} p_j = A+1$

Let $\sum_{j\in B_1} p_j = A+1+k,$ where $k \geq 1.$ Then, since $p_{1,n+1}^1 = A^3,\ p_{2,n+1}^1 = A^2$ and $\sum_{j\in B_2} p_{2,j}^1 = A+1-k$

$$\begin{aligned} C_{max}^1(\hat{\sigma}) &\geq \sum_{j\in B_1} p_{1,j}^1 + p_{1,n+1}^1 + p_{2,n+1}^1 + \sum_{j\in B_2} p_{2,j}^1 \\ &= A(A+1+k) + A^3 + A^2 + (A+1-k) \\ &= A^3 + 2A^2 + (k+2)A - k + 1 \geq A^3 + 2A^2 + 3A. \end{aligned}$$

Since $3A > 2A+1,$

$$C_{max}^1(\hat{\sigma}) > A^3 + 2A^2 + 2A + 1,$$

which implies

$$z(\hat{\sigma}) \geq C_{max}^1(\hat{\sigma}) - C_{max}^1(\sigma_1^*) > 0.$$

This is a contradiction.

Case 2: $\sum_{j\in B_1} p_j < A+1$

Let $\sum_{j\in B_1} p_j < A+1-k,$ where $k \geq 1.$ Then, since $p_{1,n+1}^2 = A^2,\ p_{2,n+1}^2 = A^3$ and $\sum_{j\in B_2} p_{2,j}^2 = A(A+1+k),$

$$\begin{aligned} C_{max}^2(\hat{\sigma}) &\geq \sum_{j\in B_1} p_{1,j}^2 + p_{1,n+1}^2 + p_{2,n+1}^2 + \sum_{j\in B_2} p_{2,j}^2 \\ &= A(A+1-k) + A^2 + A^3 + (A+1+k) \\ &= A^3 + 2A^2 + (k+2)A - k + 1 \\ &\geq A^3 + 2A^2 + 3A. \end{aligned}$$

Since $3A > 2A+1,$

$$C_{max}^2(\hat{\sigma}) > A^3 + 2A^2 + 2A + 1,$$

which implies

$$z(\hat{\sigma}) \ge C^2_{max}(\hat{\sigma}) - C^2_{max}(\sigma^*_2) > 0.$$

This is a contradiction.
By case 1 and case 2, the proof is complete.

Appendix B: Proof of Theorem 4

We reduce the equal cardinality partition problem, which is stated as follows: Given $2n$ integers $\{b_1, b_2, \cdots, b_{2n}\}$ such that $\sum_{j=1}^{2n} b_j = 2K$, is there a subset $B \subseteq \{1, 2, \cdots, 2n\}$ such that $\sum_{j \in B} b_j = K$ and $|B| = n$? Without loss of generality, assume that $n \ge 10$, and all integers in $\{b_1, b_2, \cdots, b_{2n}\}$ are the multiples of 10.

Given an instance of the equal cardinality partition problem, we can construct an instance of Problem OP-Max with two machines and two scenarios as follows:

- $p_j = b_j K,\ j = 1, 2, \cdots, 2n,\ p_{2n+1} = p_{2n+2} = 1$ and $p_{2n+3} = nK^2$;
- $q^1_1 = 0$ and $q^1_2 = K^2$;
- $q^2_1 = K^2$ and $q^2_2 = 0$.

Clearly, this reduction can be done in polynomial time. We, henceforth, will show that there is a schedule $\bar{\sigma}$ such that $z(\bar{\sigma}) = 0$ if and only if there is a solution to the equal cardinality partition problem. In this reduction, we observe that

- The optimal schedules under scenarios 1 and 2 are SPT and LPT orders, respectively;
- $C^1_{max}(\sigma^*_1)$ and $C^2_{max}(\sigma^*_2)$ can be described as follows:

$$C^1_{max}(\sigma^*_1) = C^2_{max}(\sigma^*_2) = (3n+2)K^2 + 1.$$

Suppose that there exists a set $\bar{B}$ in the equal cardinality partition problem such that $\sum_{j \in \bar{B}} b_j = K$ and $|\bar{B}| = n$. Then we can construct a new schedule $\bar{\sigma} = (2n+1, \sigma_{\bar{B}}, 2n+3, \sigma_{\bar{B}'}, 2n+2)$, where

- $\sigma_{\bar{B}}$ is the SPT subschedule of jobs in $\bar{B}$;
- $\sigma_{\bar{B}'}$ is the LPT subschedule of jobs in $\bar{B}' = \{1, 2, \cdots, 2n\} \setminus \bar{B}$.

Then, since $p^1_{1,2n+1} = 1,\ p^1_{2,2n+1} = K^2,\ p^1_{2,j} = K^2$ for $j \in \bar{B} \cup \bar{B}'$, $p^1_{2,2n+3} = nK^2$, $p^1_{2,2n+2} = K^2$ and $|\bar{B}| = |\bar{B}'| = n$,

$$\begin{aligned} C^1_{max}(\bar{\sigma}) &= p^1_{1,2n+1} + p^1_{2,2n+1} + \sum_{j \in \bar{B}} p^1_{2,j} \\ &\quad + p^1_{2,2n+3} + \sum_{j \in \bar{B}'} p^1_{2,j} + p^1_{2,2n+2} \\ &= 1 + K^2 + nK^2 + nK^2 + nK^2 + K^2 = (3n+2)K^2 + 1, \end{aligned}$$

and, since $p^2_{1,2n+1} = K^2,\ p^2_{1,2n+3} = nK^2,\ p^2_{1,j} = K^2$ for $j \in \bar{B} \cup \bar{B}'$, $p^2_{1,2n+2} = K^2$, $p^2_{2,2n+2} = 1$ and $|\bar{B}| = |\bar{B}'| = n$,

$$\begin{aligned} C^2_{max}(\bar{\sigma}) &= p^2_{1,2n+1} + \sum_{j \in \bar{B}} p^2_{1,j} + p^2_{1,n+3} \\ &\quad + \sum_{j \in \bar{B}'} p^2_{1,j} + p^2_{1,2n+2} + p^2_{2,2n+2} \\ &= K^2 + nK^2 + nK^2 + nK^2 + K^2 + 1 = (3n+2)K^2 + 1. \end{aligned}$$

Thus,

$$z(\bar{\sigma}) = max_{s=1,2}\left\{C^s_{max}(\bar{\sigma}) - C^s_{max}(\sigma^*_s)\right\} = 0.$$

Suppose that there exists a schedule $\hat{\sigma} = (\sigma_{B_1}, n+1, \sigma_{B_2})$ such that

- $z(\hat{\sigma}) = 0$;
- $B_1(B_2)$ is the set of jobs positioned before (after) job $2n+3$;
- By Theorem 1, σ_{B_1} is the SPT subschedule of jobs in B_1 and σ_{B_2} is the LPT subschedule of jobs in $\{1, 2, \cdots, 2n, 2n+1, 2n+2\} \setminus B_1$.

Note that since $C^1_{max}(\sigma^*_1)$ and $C^2_{max}(\sigma^*_2)$ have the constant term 1 and integers in $\{p_1, p_2, \cdots, p_{2n}, p_{2n+3}\}$ are all multiples of 10,

- $2n+1 \in B_1$ and $2n+2 \in B_2$, or $2n+2 \in B_2$ and $2n+1 \in B_1$.

Without loss of generality, henceforth, we assume that $2n+1 \in B_1$ and $2n+2 \in B_2$ in $\hat{\sigma}$. Let $\bar{B}_1 = B_1 \setminus \{2n+1\}$ and $\bar{B}_2 = B_2 \setminus \{2n+2\}$. Then, we will show $|\hat{B}_1| = |\hat{B}_2|$ and $\sum_{j \in \hat{B}_1} p_j = K^2$ which imply $\hat{B}_1$ is a solution of equal cardinality partition problem.

Claim 1: $|\hat{B}_1| = |\hat{B}_2|$.
Proof: Suppose that $|\hat{B}_1| > |\hat{B}_2|$. Then, it is observed that since $p^2_{1,j} = K^2$ for $j \in \hat{B}_1$, $|\hat{B}_1| \ge n+1$ and $n \ge 10$,

$$\sum_{j \in \hat{B}_1} p^2_{1,j} + \sum_{j \in \hat{B}_2} p^2_{2,j} \ge (n+1)K^2 + 1.$$

Then, by this inequality, $p^2_{1,2n+1} = K^2$, $p^2_{1,2n+3} = p^2_{2,2n+3} = nK^2$ and $p^2_{2,2n+2} = 1$,

$$\begin{aligned} C^2_{max}(\hat{\sigma}) &\ge p^2_{1,2n+1} + \sum_{j \in \hat{B}_1} p^2_{1,j} \\ &\quad + p^2_{1,2n+3} + p^2_{2,2n+3} + \sum_{j \in \hat{B}_2} p^2_{2,j} + p^2_{2,2n+2} \\ &\ge K^2 + (n+1)K^2 + nK^2 + nK^2 + 1 = (3n+2)K^2 + 2, \end{aligned}$$

Thus, $z(\hat{\sigma}) \ge C^2_{max}(\hat{\sigma}) - C^2_{max}(\sigma^*_2) \ge 1$. This is a contradiction.

The argument above can be similarly applied to the case with $|\hat{B}_1| < |\hat{B}_2|$. The proof is complete. ■

Henceforth, we consider only the schedules satisfying Claim 1. Consider two cases.

Case 1: $\sum_{j \in \hat{B}_1} p_j > K^2$

It is observed that since for $j \in \hat{B}_1$, $|\hat{B}_1| \ge n+1$ and $n \ge 10$, $p_j = b_j K,\ j = 1, 2, \cdots, 2n$, $\sum_{j \in \hat{B}_1} p^1_{1,j} \ge K^2 + K$. Thus, since $p^1_{1,2n+1} = 1$, $p^1_{1,2n+3} = p^1_{2,2n+3} = nK^2$, $\sum_{j \in \hat{B}_2} p^1_{2,j} = nK^2$ and $p^1_{2,2n+2} = K^2$,

$$\begin{aligned} C^1_{max}(\hat{\sigma}) &\geq p^1_{1,2n+1} + \sum\nolimits_{j\in\hat{B}_1} p^1_{1,j} + p^1_{1,2n+3} \\ &+ p^1_{2,2n+3} + \sum\nolimits_{j\in\hat{B}_2} p^1_{2,j} + p^1_{2,2n+2} \\ &\geq 1 + K^2 + K + nK^2 + nK^2 + nK^2 + K^2 \\ &= (3n+2)K^2 + K + 1. \end{aligned}$$

Thus, $z(\hat{\sigma}) \geq C^1_{max}(\hat{\sigma}) - C^1_{max}(\sigma^*_1) \geq K$. This is a contradiction.

Case 2: $\sum_{j\in\hat{B}_1} p_j < K^2$

It is observed that since $p_j = b_j K,\ j = 1, 2, \cdots, 2n,\ \sum_{j\in\hat{B}_2} p^2_{2,j} \geq K^2 + K$. Thus, since $p^2_{1,2n+1} = K^2$, $\sum_{j\in\hat{B}_1} p^2_{1,j} \geq nK^2$, $p^2_{1,\,2n+3} = p^2_{2,2n+3} = nK^2$ and $p^2_{2,2n+2} = 1$,

$$\begin{aligned} C^2_{max}(\hat{\sigma}) &\geq p^2_{1,\,2n+1} + \sum\nolimits_{j\in\hat{B}_1} p^2_{1,\,j} + p^2_{1,\,2n+3} \\ &+ p^2_{2,\,2n+3} + \sum\nolimits_{j\in\hat{B}_2} p^2_{2,\,j} + p^2_{2,\,2n+2} \\ &\geq K^2 + nK^2 + nK^2 + nK^2 + K^2 + k + 1 \\ &= (3n+2)K^2 + K + 1. \end{aligned}$$

Thus, $z(\hat{\sigma}) \geq C^2_{max}(\hat{\sigma}) - C^2_{max}(\sigma^*_2) \geq K$. This is a contradiction.

By cases 1 and 2, the proof is complete.