International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: https://www.tandfonline.com/loi/tprs20

Approximation algorithms for the min-max regret identical parallel machine scheduling problem with outsourcing and uncertain processing time

Shijin Wang & Wenli Cui

International Journal of Production Research, 2020
https://doi.org/10.1080/00207543.2020.1766721

Approximation algorithms for the min-max regret identical parallel machine scheduling problem with outsourcing and uncertain processing time

Shijin Wang* and Wenli Cui

School of Economics and Management, Tongji University, Shanghai, People's Republic of China

(*Received 2 February 2020; accepted 28 April 2020*)

We consider the robust (min-max regret) version of identical parallel machine scheduling problem, in which jobs may be outsourced to balance total cost against production efficiency. The total cost is measured in terms of the total completion time of jobs processed in-house and the cost of outsourcing the rest. Processing times of in-house jobs are uncertain and they are described as two types of scenarios: discrete and interval. The objective is to obtain a robust (min-max regret) decision that minimises the absolute deviation of total cost from the optimal solution under the worst-case scenario. We first prove the worst-case scenario for any feasible solution. For the interval scenario, we further prove that the maximum regret value can be obtained in polynomial time for any feasible schedule. We also prove that for any discrete scenario, the minimum total cost can be obtained in polynomial time. Since the problem with the interval scenario is strongly NP-hard, we then transform the problem into an equivalent robust single machine scheduling problem. Finally, we develop 2-approximation algorithms for the problem with discrete and interval scenarios, respectively. These results are helpful for bridging the scheduling theory and practice in identical parallel machining environments with outsourcing and uncertain processing times.

Keywords: identical parallel machine scheduling; min-max regret; uncertainty; approximation algorithms

1. Introduction

Nowadays, many companies, especially in electronics and motor industries, adopt outsourcing strategy under which they subcontract some operations to third-party companies with certain specialty or competitiveness, rather than process all operations by themselves. If such outsourcing strategy is utilised well, the companies can usually achieve lower production costs, shorter lead times, and higher ability to flexibly cope with a fluctuating market (Cachon and Harker 2002).

In canonical scheduling problems, processing times of operations or jobs are assumed to be deterministic beforehand (Aissi, Bazgan, and Vanderpooten 2009; Conde 2014; Gholami et al. 2019). However, in practice, processing times of operations or jobs may be uncertain due to various factors, such as the introduction of new machines, disruptions in the delivery of raw materials or parts, and sudden breakdown of machines. In this context, outsourcing, to some extent, is a good strategic choice to deal with operational risks and uncertainties. Hence, in this study, we consider a scheduling problem at operational level with outsourcing options and with the uncertainty of processing times of jobs within in-house manufacturing process.

Deterministic scheduling problems with an outsourcing option have been extensively studied in various manufacturing environments in the literature, for example, in the single machine (c.f., Lee and Sung 2008a, 2008b), in the parallel machine (c.f., Ruiz-Torres, Ho, and López 2006; Tavares Neto, Godinho Filho, and Da Silva 2015), in the flow shop (c.f., Choi and Chung 2011; Lee and Choi 2011; Chung and Choi 2013; Choi and Park 2014; Liu 2019), and in the job shop (c.f., Chung et al. 2005). However, there are few studies on the scheduling problems with an outsourcing option and with some uncertain problem parameters. To the best of our knowledge, only Choi and Chung (2016) and Park and Choi (2017) considered the single machine scheduling problem with outsourcing option under certain parameter uncertainty by robust (min-max regret) optimisation. Choi and Chung (2016) considered the problem in which the processing time is uncertain while the outsourcing cost is known in advance. They analysed the computational complexity for various cases, depending on whether the performance measurement of in-house jobs is makespan or total completion time, and whether the processing time uncertainty is described as interval or discrete scenarios. Park and Choi (2017) considered a single machine scheduling problem with an outsourcing option where the processing time and outsourcing cost are uncertain. The cost of processing

*Corresponding author. Email: shijinwang@tongji.edu.cn

jobs in-house is measured as the total weighted completion time. They deduced the computational complexity for two special cases of the problem.

In this study, we focus on an identical parallel machine scheduling problem with outsourcing and with uncertain processing times of jobs. Jobs can be outsourced to a subcontractor to improve performance measurement, which is expressed as the total cost for processing some jobs in-house measured by the total completion time and for paying for the processing of outsourced jobs. Furthermore, the objective of our scheduling problem is to minimise the maximum deviation from optimality over all scenarios defined by processing time uncertainty, referred to as the min-max regret version. In addition, parallel machine environment has an equal importance in both theoretical and practical aspects. Theoretically, it is a generalisation of the single machine and a special case of the flexible flow shop. From a practical viewpoint, it is important because the occurrence of resources in parallel is very common in the real world (Alon et al. 1998).

The main contributions of this paper are summarised as follows:

(1) For the defined robust (min-max regret) version of identical parallel machine scheduling problem with outsourcing and with uncertain processing times of jobs, we find and prove the worst-case scenario for any feasible solution.
(2) For the interval scenario of processing times, we prove that the maximum regret value can be obtained in polynomial time for any feasible schedule. We also prove that for any discrete scenario of processing times, the minimum total cost can be obtained in polynomial time.
(3) To further reduce the computational efforts, we transform the problem under study into an equivalent robust single machine scheduling problem for the interval scenario.
(4) We develop a 2-approximation algorithm for each case with discrete or interval scenarios of processing times, respectively.

The rest of the paper is organised as follows. In Section 2, the related literature are reviewed. In Section 3, we formally define the problem under study and some preliminaries results are given. In Section 4, we transform the problem with the interval scenario of processing times into an equivalent robust single machine scheduling problem. In Section 5, we develop a 2-approximation algorithm for each case with discrete or interval scenario of processing times, respectively. Finally, in Section 6, we conclude with our contributions and provide the direction for future research.

2. Literature review

2.1. Related studies on deterministic scheduling problems with outsourcing

In general, the studies in optimisation with outsourcing option could be broadly divided into two categories: game-theoretic analysis and scheduling problems in operational level. Game theory is applied to treat the subcontracting, in particular the subcontracting price. For each decision maker (i.e. manufacturer and subcontractor), their decisions are related to the scheduling problems with a subcontracting price to maximise their own revenue related objectives (Qi 2012; Xiao and Gaimon 2013; Wang, Geng, and Cheng 2018). While scheduling with outsourcing in operational level is to decide which jobs to be outsourced and how to schedule in-house jobs at the same time. By far, there are a lot of successful studies in this streamline in various machining environments, including single machine (e.g. Lee and Sung 2008a, 2008b), job shop (Chung et al. 2005) and flow shop (Choi and Chung 2011; Lee and Choi 2011; Chung and Choi 2013; Choi and Park 2014; Liu 2019).

In the following, we focus more on the highly-related scheduling problems with outsourcing in parallel machine environment. For a setting of in-house multiple identical parallel machines, Ruiz-Torres, Ho, and López (2006) studied the joint optimisation problem of job scheduling and outsourcing, and developed several heuristics to simultaneously minimise the number of late orders and the total outsource processing times. For a similar setting, Chen and Li (2008) studied the joint optimisation problem to minimise the total production and subcontracting cost with the limit of the maximum completion time of the orders. The computational complexity was analysed, and a heuristic was developed with its performance analysis. With the limit of a constant total outsource cost, Tavares Neto, Godinho Filho, and Da Silva (2015) studied the scheduling problem with in-house identical parallel machines and a third-party subcontractor to minimise the sum of the outsourcing cost and the total weighted tardiness. A linear programming model and an ant colony optimisation algorithm were developed to solve the problem effectively.

For a setting of in-house unrelated parallel machines and multiple subcontractors each having a single machine, Mokhtari and Abadi (2013) investigated the joint scheduling of both in-house and outsourced jobs simultaneously to minimise sum of the total weighted completion time and total outsourcing cost. An integer programming model and a heuristic were developed for the problem.

Unlike these above deterministic parallel machine scheduling problem with outsourcing allowed, we focus more on the robust (min-max regret) version with outsourcing and with uncertain processing times of jobs to combine dynamic features of problem parameters in the real world practices.

2.2. Related studies on stability analysis of scheduling problems with uncertainty

Stability analysis concerns about how much can the numerical parameters of a scheduling problem be perturbed without loss of the original optimal solution or solution set. Such analysis could be used for transferring deterministic scheduling results under conditions of uncertainty like interval processing times of jobs, which could be used to bridge scheduling theory and practice.

Sotskov, Leontev, and Gordeev (1995) surveyed the earlier studies on stability analysis for discrete optimisation problems, in particular on the stability ball of optimal or approximate solutions. The properties of stability radius of an optimal makespan scheduling problem was analysed. Stability radius denotes the largest quantity of independent variations of the processing times of jobs such that an optimal schedule of the problem remains unchanged. An optimal schedule with a larger stability radius is better. The calculation of the stability radius is generally NP-hard.

In general, stability analysis on scheduling problems focuses on formulas, exact algorithms and heuristic methods for calculating the stability radius of an optimal or feasible schedule, necessary and sufficient conditions for zero stability radius, infinite stability radius, and lower and upper bounds of stability radius. Such researches have been conducted for various scheduling problems, including a general shop scheduling problem with the objective of minimising mean flow time (Bräsel, Sotskov, and Werner 1996), a job shop scheduling problem with the objective of minimising mean or maximum flow time (Sotskov, Sotskova, and Werner 1997), a general shop scheduling problem with bounded processing times (Lai et al. 1997), a scheduling problem with the objective of minimising the makespan with uncertain input data (Lai and Sotskov 1999), and various assembly line balance problems including SALBP-1 (Sotskov, Dolgui, and Portmann 2006), SALBP-1 and SALBP-2 (Sotskov et al. 2015), SALBP-2 (Lai et al. 2016) and SALBP-E (Gurevsky, Battaïa, and Dolgui 2012; Lai, Sotskov, and Dolgui 2019). SALBP-1 is to minimise a number of workstations within a fixed cycle time in the Simple Assembly Line Balancing Problem (SALBP), SALBP-2 is to minimise the cycle time with a fixed workstation set, and SALBP-E is to maximise the line efficiency in terms of the products given the number of opened workstations and the working time on the most loaded one, with neither fixed number of workstations nor fixed cycle time.

There are also researches on other stability measures like stability box. Sotskov and Lai (2012) derived an exact formula for the stability box for a single machine scheduling problem with interval processing times to minimise the total weighted flow time. Lai et al. (2018) developed an algorithm for constructing the optimality box for the same problem.

Unlike these above stability analysis on scheduling problems, we focus more on the robust (min-max regret) version with outsourcing and with uncertain processing times of jobs in parallel machine environment. In addition, min-max regret version is different from stability analysis in that the latter is 'post-optimal analysis' and is conducted after an optimal or feasible solution has been already found, whereas the former tries to find the solution with the minimum of the maximum regret values for all scenarios.

2.3. Related studies on robust (min-max regret) versions of scheduling problem with outsourcing

Robustness analysis is a theoretical framework that enables the decision maker to take into account uncertainty or imprecision in order to obtain decisions that will behave reasonably under any likely input data. The min-max regret criterion, which is suitable in situations where the decision maker may feel regret if he/she makes a wrong decision, is less conservative and its purpose is to obtain a solution minimising the maximum deviation between the value of the solution and the optimal value of the corresponding scenario over all possible scenarios (Aissi, Bazgan, and Vanderpooten 2009). The min-max regret versions of classical combinatorial optimisation problems have been widely studied (c.f., Feizollahi and Averbakh 2014; Furini et al. 2015; Conde 2019). In general, the min-max regret versions of combinatorial optimisation problems are harder than their classical counterparts (Aissi, Bazgan, and Vanderpooten 2009).

In the literature, there are continuous development and successful applications of min-max regret on machine scheduling problems in various environments, including in the single machine environment (Kasperski and Zieliński 2008; Sotskov, Egorova, and Lai 2014; Choi and Chung 2016; Drwal 2018; Yue et al. 2018; Kacem and Kellerer 2019), in the flow shop environment (Kouvelis, Daniels, and Vairaktarakis 2000; Averbakh 2006; Kasperski, Kurpisz, and Zieliński 2012), in the parallel machine environment (Xu et al. 2013; Conde 2014; Siepak and Józefczyk 2014; Xu, Lin, and Cui 2014; Drwal and Rischke 2016), and in the hybrid flow shop (Feng, Zheng, and Xu 2016).

In the following, we mainly review the most related works on min-max regret versions of scheduling problems in parallel machine environment and on min-max regret versions of scheduling problems with outsourcing. Xu et al. (2013)

investigated the interval data min-max regret (IDMR) version of the identical parallel machine scheduling problem with interval processing times to minimise the makespan (denoted by $C_{\max}$). Iterative relaxation based exact methods, local search and simulated annealing based heuristic methods were developed for the problem. Later, Xu, Lin, and Cui (2014) studied the IDMR version of uniform parallel machine scheduling problem with interval processing times to minimise total flow time (denoted by $\sum C_j$). A series of problem properties were analysed. They proved that the problem can be transformed into an equivalent robust single machine scheduling problem. Siepak and Józefczyk (2014) considered the IDMR version of unrelated machine scheduling problem with interval processing times to minimise $\sum C_j$. They developed a 2-approximate middle intervals time efficient algorithm and a scatter search based heuristic method for the problem. Conde (2014) proposed a mixed integer linear programming (MILP) formulation for the IDMR version of unrelated parallel machine scheduling with the interval processing times to minimise $\sum C_j$. Drwal and Rischke (2016) studied the IDMR version of the identical parallel machine scheduling problem with the interval processing times of jobs to minimise $\sum C_j$. They proved that the problem is strongly NP-hard, although the deterministic counterpart is polynomial-time solvable. They suggested that designing approximation algorithms for the problem with approximation ratio below two is an interesting open question.

Choi and Chung (2016) considered the min-max regret version of the single machine scheduling problem to determine which jobs should be processed by outsourcing under both interval and discrete scenarios with the objective of minimisation of the total cost for processing some jobs in-house and outsourcing the rest. Complexity results were deduced for the cost of in-house jobs as $C_{\max}$ and $\sum C_j$, respectively. For the discrete scenario case with the objective of makespan $C_{\max}$, they developed a 2-approximation algorithm. Park and Choi (2017) studied the min-max regret version of single machine scheduling problem with outsourcing, in which the in-house and outsourcing cost are uncertain and they are described with interval or discrete scenarios. The objective considered for in-house jobs are the total weighted completion time (denoted by $\sum w_jC_j$). The computational complexity for two special cases (identical weights and same processing times) were deduced.

The main features of the studies mentioned above are summarised in Table 1. As shown in the table, this study extends the existing researches and focuses on the min-max regret version of identical parallel machine scheduling problem with outsourcing and with uncertain in-house job processing time which is described in both discrete and interval scenarios. Approximation algorithms with approximation ratio equal to 2 are developed for the problem with discrete and interval scenarios.

3. Problem description and preliminaries

3.1. Problem description

There are a set $\mathcal{N} = \{1, 2, \ldots, n\}$ of n independent jobs that can be either processed on in-house identical parallel machines or outsourced to a subcontractor. Let $\mathcal{M} = \{1, 2, \ldots, m\}$ denote the set of m in-house identical parallel machines. Each job is available at time 0 and can be processed non-preemptively only by one arbitrary machine. No machine can process more than one job at the same time. It is assumed that the capacity of the subcontractor is infinite. The processing time uncertainty is described as the set of scenarios denoted by S. Let p_j^s denote the processing time of job j under scenario $s \in S$. In the interval scenario, the processing time of job j can take any value from the interval $[p_j^-, p_j^+]$, i.e. $p_j^s \in [p_j^-, p_j^+]$ $(j \in \mathcal{N})$. A discrete scenario $s \in S$ is described as a vector of processing times $(p_1^s, p_2^s, \ldots, p_n^s)$.

Let $(\mathcal{I}, \mathcal{O}, x)$ be a schedule such that $\mathcal{I}$ and $\mathcal{O}$ are the sets of in-house and outsourced jobs, respectively, where $\mathcal{I} \cup \mathcal{O} = \mathcal{N}$ and $\mathcal{I} \cap \mathcal{O} = \emptyset$, and x is the assignment and schedule of jobs in $\mathcal{I}$. Let $C_j^s(x)$ be the completion time of job $j \in \mathcal{I}$ in x under scenario s and let o_j be the outsourcing cost of job $j \in \mathcal{O}$. In this paper, we study the problem to minimise the total cost which is expressed as the sum of the total completion time and the total outsourcing cost.

A schedule can be represented by a matrix $x = [x_{ijk}]_{m\times n\times n}$, where $x_{ijk} = 1$ if (and only if) the task j is scheduled as the kth to the last job on machine i, and $x_{ijk} = 0$ otherwise. Let X be the set of all feasible schedules. The deterministic problem of minimising the total cost in schedule x under scenario s is as follows:

$$F^s(x) = \sum_{j\in\mathcal{N}} \sum_{i\in\mathcal{M}} \sum_{k\in\mathcal{N}} kp_j^s x_{ijk} + \sum_{j\in\mathcal{N}} o_j \left(1 - \sum_{i\in\mathcal{M}} \sum_{k\in\mathcal{N}} x_{ijk}\right) \tag{1}$$

We denote the minimum total cost under scenario s as $F^s(*)$:

$$F^s(*) = \min_{x\in X} F^s(x) \tag{2}$$

Given a feasible schedule $x \in X$, its maximum regret under scenario s is defined as:

$$R_{\max}(x) = \max_{s\in S}(F^s(x) - F^s(*)) \tag{3}$$

Table 1. Summary of the highly related min-max regret works in the literature.

Year	Authors	Objective	Machine	Uncertainty	Outsource	Complexity	Approaches
2013	Xu et al.	$C_{\max}$	IPM	Interval PT	N	Y	Exact & heuristics
2014	Xu, Lin, and Cui	$\sum C_j$	UPM	Interval PT	N	Y	MILP, Approximation
2014	Siepak and Józefczyk	$\sum C_j$	URPM	Interval PT	N	Y	Approximation, heuristics
2014	Conde	$\sum C_j$	URPM	Interval PT	N	Y	MILP
2016	Choi and Chung	total cost, $\sum C_j$, $C_{\max}$	SM	In-house PT discrete & interval	Y	Y	Approximation
2016	Drwal and Rischke	$\sum C_j$	IPM	Interval PT	N	Y	–
2017	Park and Choi	total cost, $\sum w_j C_j$	SM	In-house PT or outsourcing cost interval & discrete	Y	Y	–
This	paper	total cost, $\sum C_j$	IPM	In-house PT interval & discrete	Y	Y	Approximation

Note: 'SM': single machine; 'IPM': identical parallel machine; 'UPM': uniform parallel machine; 'URPM': unrelated parallel machine; 'PT': processing time.

The scenario of maximising the regret across all possible scenarios is called the worst-case scenario of x. The robust (min-max regret) parallel machine scheduling problem with outsourcing (denoted by **R-PMO** afterwards) can be formulated as:

$$R = \min_{x \in X} R_{\max}(x) = \min_{x \in X} \max_{s \in S}(F^s(x) - F^s(*)) \tag{4}$$

The problem **R-PMO** with the interval scenario is strongly NP-hard since the min-max regret version of $P_m \parallel \sum C_j$ has been proven to be strongly NP-hard (Drwal and Rischke 2016), which is a reduction version of this problem without consideration of outsourcing. The problem **R-PMO** with the discrete scenario is NP-hard since the min-max regret version of the single machine with outsourcing has been proven to be NP-hard (Choi and Chung 2016), even when the number of possible scenarios is only two.

3.2. The worst-case scenario

Let $x_j = \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{N}} k x_{ijk}$ represent the position weight of job j for schedule x. Let s_x be the worst-case scenario for x. We denote the optimal schedule under s_x by $y = [y_{hjf}]_{m \times n \times n}$ such that $y_{hjf} = 1$ if job j is in the fth position to the last job on machine h, otherwise $y_{hjf} = 0$. Then, let $y_j = \sum_{h \in \mathcal{M}} \sum_{f \in \mathcal{N}} f y_{hjf}$ and the maximum regret of schedule x can be expressed as

$$\begin{aligned} R_{\max}(x) &= F^{s_x}(x) - F^{s_x}(*) = F^{s_x}(x) - F^{s_x}(y) \\ &= \left[\sum_{j \in \mathcal{N}} \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{N}} k p_j^{s_x} x_{ijk} + \sum_{j \in \mathcal{N}} o_j \left(1 - \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{N}} x_{ijk}\right)\right] \\ &\quad - \left[\sum_{j \in \mathcal{N}} \sum_{h \in \mathcal{M}} \sum_{f \in \mathcal{N}} f p_j^{s_x} y_{hjf} + \sum_{j \in \mathcal{N}} o_j \left(1 - \sum_{h \in \mathcal{M}} \sum_{f \in \mathcal{N}} y_{hjf}\right)\right] \\ &= \sum_{j \in \mathcal{N}} \left[(x_j - y_j) p_j^{s_x} + \left(\sum_{h \in \mathcal{M}} \sum_{f \in \mathcal{N}} y_{hjf} - \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{N}} x_{ijk}\right) o_j\right] \end{aligned} \tag{5}$$

Lemma 1 *For any schedule $x \in X$, there exists a worst-case scenario s_x for x, which can be defined as*:

$$p_j^{s_x} = \begin{cases} p_j^+ & \text{if } x_j > y_j \\ p_j^- & \text{otherwise} \end{cases} \tag{6}$$

Proof Given any schedule x, we can calculate x_j $(j \in \mathcal{N})$. The maximum regret can be written as (5), which is increasing with $p_j^{s_x}$ when $x_j - y_j > 0$, and non-increasing with $p_j^{s_x}$ when $x_j - y_j \leq 0$. ■

3.3. Computing the maximum regret value for the interval scenario

We show that it is possible to compute the value of maximum regret $R_{\max}(x) = \max_{s \in S}(F^s(x) - F^s(*))$ in polynomial time for the interval scenario for a fixed schedule x. The method is similar to the one presented in Conde (2014) for unrelated parallel machines. The main difference is that in the identical parallel machines with outsourcing, the input data contains a single interval $[p_i^-, p_i^+]$, while m intervals are given in the case of unrelated parallel machines, and the outsourcing decision should be additionally considered. Thus, we have the following theorem:

Theorem 1 *For any feasible schedule x, the maximum regret can be computed as*:

$$R_{\max}(x) = \sum_{j \in \mathcal{N}} \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{N}} (k p_j^+ - o_j) x_{ijk} - \min_{y \in X} \sum_{j \in \mathcal{N}} \sum_{h \in \mathcal{M}} \sum_{f \in \mathcal{N}} c_{hjf}(x) y_{hjf} \tag{7}$$

where, for a fixed job j, we have,

$$c_{hjf}(x) = f p_j^- + (p_j^+ - p_j^-) \sum_{i \in \mathcal{M}} \sum_{k \in \mathcal{N}} \min\{k, f\} x_{ijk} - o_j \tag{8}$$

Proof If $x_{ijk} = 1$, then $x_j = k$ and if $y_{hjf} = 1$, then $y_j = f$. One can write the expression $(x_j - y_j)p_j^{s_x} + (\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk})o_j$ in Equation (5) as a function of the variables x_{ijk} and y_{hjf},

$$(x_j - y_j)p_j^{s_x} + \left(\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk}\right) o_j = \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (kp_j^+ - o_j)x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} c_{hjf}(x)y_{hjf} \tag{9}$$

Next, we will show the correctness of Equation (9) for different cases that can arise.

Case 1: All jobs are outsourced for both schedules x and y. In this case, the left-hand side of Equation (9) is equal to its right-hand side and they are both equal to 0.

Case 2: Job j is not processed in house in schedule x but it is processed in house in schedule y, that is, $x_j = 0$, i.e. $x_{ijk} = 0$ for $i \in \mathcal{M}$, $k \in \mathcal{N}$, and $y_j = \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} fy_{hjf} > 0$. Since $x_j < y_j$, $p_j^{s_x} = p_j^-$ according to Equation (6). In this case, the left-hand side of Equation (9) is equal to $-p_j^- \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} fy_{hjf} + o_j \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} = -\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}}(fp_j^- - o_j)y_{hjf}$, which coincides with the right-hand side of Equation (9) after substituting $c_{hjf}(x)$ in Equation (8) into Equation (9).

Case 3: Job j is processed in house in schedule x, but it is not processed in house in schedule y. Since $x_j > y_j$, $p_j^{s_x} = p_j^+$ according to Equation (6). In this case, both the left- and right-hand sides of Equation (9) are equal to $p_j^+ \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} kx_{ijk} - o_j \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} = \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}(kp_j^+ - o_j)x_{ijk}$.

Case 4: Job j is processed in house at the position kth position to the last job in schedule x and at the position fth position to the last job in schedule y, i.e. $x_{ijk} = y_{hjf} = 1$, then $\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} \min\{k,f\}x_{ijk} = \min\{k,f\}$ in Equation (8). In this case, the right-hand side of Equation (9) is equal to

$$\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (kp_j^+ - o_j)x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} c_{hjf}(x)y_{hjf} = \begin{cases} p_j^+(k-f) + O & \text{if } k > f \\ p_j^-(k-f) + O & \text{otherwise} \end{cases}$$

where $O = (\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk})o_j$, which coincides with the left-hand side of Equation (9).

This completes the proof. ■

The minimisation in (7) is equivalent to the minimum assignment problem, thus it can be solved in polynomial time by using the Hungarian method (Papadimitriou and Steiglitz 1998). Recall that solving an assignment problem of size $n \times nm$ requires $O((nm)^3)$ time (using the Hungarian method). Hence, the maximum regret of any given feasible schedule can be obtained in $O((nm)^3)$ time.

3.4. *Computing the minimum total cost for any discrete scenario*

It is clear that in-house jobs should be processed in the increasing order of p_j in an optimal schedule for all scenarios since the total cost for in-house jobs is expressed as the total completion time. Without loss of generality, we assume that $p_1^s \le p_2^s \le \cdots \le p_n^s$ under scenario s.

We define binary variables $x_{ij} = 1$ $(i \in \mathcal{M}, j \in \mathcal{N})$ if job j is on machine i, and otherwise $x_{ij} = 0$. Then,

$$F^s(*) = \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}\sum_{k=1}^{j} p_k^s x_{ik} + \sum_{j=1}^{n} o_j\left(1 - \sum_{i=1}^{m} x_{ij}\right) \tag{10}$$

Let $x = (x_{11}, \ldots, x_{1n}, \ldots, x_{m1}, \ldots, x_{mn})^T$ be the mn-dimensional vector. Since $x_{ij} = x_{ij}^2$, formulation (10) can be rewritten as

$$\begin{aligned} F^s(*) &= x^T Q_s x + \sum_{j=1}^{n} o_j \\ &= (x_{11}(p_1^s - o_1)x_{11} + p_1^s x_{11}x_{12} + \cdots + p_1^s x_{11}x_{1n} + \cdots + x_{m1}(p_1^s - o_1)x_{m1} + p_1^s x_{m1}x_{m2} + \cdots + p_1^s x_{m1}x_{mn}) + \cdots \\ &\quad + (x_{12}(p_2^s - o_2)x_{12} + p_2^s x_{12}x_{13} + \cdots + p_2^s x_{12}x_{1n} + \cdots + x_{m2}(p_2^s - o_2)x_{m2} + p_2^s x_{m2}x_{m3} + \cdots + p_2^s x_{m2}x_{mn}) + \cdots \\ &\quad + (x_{1n}(p_n^s - o_n)x_{1n} + \cdots + x_{mn}(p_n^s - o_n)x_{mn}) + \sum_{j=1}^{n} o_j \end{aligned} \tag{11}$$

where $Q_s = [q^s_{ij}]_{nm\times nm}$ is an $nm \times nm$ matrix, defined as follows:

$$q^s_{i+kn,j+kn} = \begin{cases} p^s_i & \text{if } 0 < i < j \le n, \text{ for } k = 0,1,\dots,m-1 \\ p^s_j - o_j & \text{if } 0 < i = j \le n, \text{ for } k = 0,1,\dots,m-1 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Then, we define,

$$q^s_{ij} = \begin{cases} q^s_{i+kn,j+kn} & \text{for } k = 0,1,\dots,m-1,\ 0 < i,j \le n \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Therefore, for $s \in S$, an optimal schedule x^{s*} and the optimal value $F^s(*)$ can be obtained in $O((nm)^2)$ time by using the QP algorithm in Choi and Chung (2016).

4. Transforming the problem with interval scenario into an equivalent robust single machine problem

In this section, in order to reduce the computation time for solving the problem **R-PMO** with interval scenario, we develop a method to transform the problem **R-PMO** into an equivalent robust single machine problem (donated by **R-SMO**) based on the method in Xu, Lin, and Cui (2014), such that the properties of the optimal number of jobs assigned to each machine can be explored.

Let n_i denote the number of jobs assigned to machine $i, i \in \mathcal{M}$ and n_I denote the number of in-house jobs. We have $\sum_{i\in\mathcal{M}} n_i = n_I$ and $n_i \cap n_h = \emptyset$ $(i,h \in \mathcal{M}, i \neq h)$. Then we define an allocation vector $V_x(n,m,i) = \{n_i, i \in \mathcal{M}\}$ to denote the number of jobs allocated to machine $i \in \mathcal{M}$ in schedule x.

Let $V^s(n,m,i)(*) = \{n^s_i(*), i \in \mathcal{M}\}$ denote the allocation vector of an optimal schedule under scenario s, i.e. the optimal allocation vector. If the optimal allocation vector for the worst-case scenario s_x of schedule x is known in advance and it is denoted by $V^{s_x}(n,m,h)(*) = \{n^{s_x}_h(*), h \in \mathcal{M}\}$, the assignment problem in the second term of Equation (7) can be reformulated as the following $n \times n_I$ assignment problem instead of an $n \times mn$ assignment problem:

$$\min_{y\in X} \sum_{j\in\mathcal{N}} \sum_{h\in\mathcal{M}} \sum_{f=1}^{n^{s_x}_h} c_{hjf}(x) y_{hjf} \tag{14}$$

A standard bipartite graph with n job nodes and n_I position nodes can be used to describe this assignment problem. A job node can be denoted by j $(j \in \mathcal{N})$ and a position node can be denoted by (h,f) $(h \in \mathcal{M}; f = 1,2,\dots,n^{s_x}_h)$.

Now we can determine the optimal allocation vector $V^{s_x}(n,m,h)(*) = \{n^{s_x}_h(*), h \in \mathcal{M}\}$ for s_x. Xu, Lin, and Cui (2014) showed that for a given (n_I, m), the optimal allocation vectors for all scenarios have an identical optimal allocation vector independent of the scenario for uniform parallel machine. Thus for the identical parallel machine in this study, we can easily have the following lemma.

LEMMA 2 *Given (n_I, m), any allocation vector $V(n_I,m,h) = \{n_h, h \in \mathcal{M}\}$ that satisfies the following inequality is an optimal allocation vector for all scenarios*:

$$n_{h_2} - n_{h_1} \le 1 \quad (\forall\, h_1, h_2 \in \mathcal{M}, h_1 \neq h_2, n_{h_2} > 0) \tag{15}$$

Based on Lemma 2 above, we propose the following allocation algorithm to obtain an optimal allocation vector $V(n_I,m,h)(*)$ for all scenarios.

Algorithm 1 Allocation algorithm

1: Input m machines and the number of jobs processed in-house n_I;
2: Set $n_h = 0$ for $h \in \mathcal{M}$. Let $i_h \leftarrow n_h + 1$;
3: **while** $\sum_{h=1}^{m} n_h < n_I$ **do**
4: Let l be the smallest index such that $i_l = \min_{h\in\mathcal{M}} i_h$, then $n_l \leftarrow n_l + 1$ and $i_l \leftarrow n_l$
5: **end while**
6: Output the number of jobs assigned on machine h, n_h $(h \in \mathcal{M})$.

There are n possible outcomes for n_I (i.e. $n_I = 1, \ldots, n$). Given the values of n and m, we can convert the identical parallel machine problem of the maximum regret value into an equivalent single machine problem in $O(n)$ time.

Theorem 2 *Given (n, m), the maximum regret of any schedule x can be obtained in $O(n^4)$ time.*

Proof Given a combination (n_I, m), we can first derive an optimal allocation vector $V^{s_x}(n, m, h)$ by using the given allocation algorithm in Algorithm 1. Then, the maximum regret can be obtained by solving an $n \times n$ assignment problem by the Hungarian algorithm which requires $O(n^3)$ time (Papadimitriou and Steiglitz 1998) and there are n possible outcomes for n_I. Hence, the maximum regret of any schedule can be obtained in $O(n^4)$ time. ■

Based on the above results, if the allocation vector $V(n, m, i) = \{n_i, i \in \mathcal{M}\}$ of schedule x is given, the **R-PMO** problem can be reformulated as the following formulation:

$$\min_{x \in X} \left\{ \sum_{j \in \mathcal{N}} \sum_{i \in \mathcal{M}} \sum_{k=1}^{n_i} (kp_j^+ - o_j) x_{ijk} - \min_{y \in X} \sum_{j \in \mathcal{N}} \sum_{h \in \mathcal{M}} \sum_{f=1}^{n_h} c_{hjf}(x) y_{hjf} \right\} \tag{16}$$

where $c_{hjf}(x) = fp_j^- + (p_j^+ - p_j^-) \sum_{i \in \mathcal{M}} \sum_{k=1}^{n_i} \min\{k, f\} x_{ijk} - o_j$

Xu, Lin, and Cui (2014) showed that given (n_I, m), the optimal allocation vector $V^*(n_I, m, i) = \{n_i^*, i \in \mathcal{M}\}$ of the optimal robust schedule $x_{n_I}^*$ with the in-house jobs n_I is the same as that of the optimal schedule for the deterministic $P_m \| \sum C_j$ scheduling problem. It indicates that given the values of n_I and m, we can obtain $V^*(n_I, m, i)$ in **R-PMO** using the allocation algorithm in Algorithm 1. The optimal robust schedule $x^* = \min_{n_I \in \mathcal{N}} x_{n_I}^*$.

Let σ be the sequence of k in ascending order and g denote the position of (i, k) in σ, i.e. $\sigma(g) = k$. Then the set of n_{I_x} position nodes (i, k) ($k = 1, 2, \ldots, n_i$; $i = 1, 2, \ldots, m$; $\sum_{i \in \mathcal{M}} n_i = n_{I_x}$) can be replaced by g, $g = 1, 2, \ldots, n_{I_x}$. Moreover, the following condition holds: if $x_{ijk} = 1$, then $x_{jg} = 1$. Let σ be the sequence of f in ascending order and r denote the position of (h, f) in σ, i.e. $\sigma(r) = f$, $r = 1, 2, \ldots, n_{I_y}$. Then, y_{hjf} can be replaced by y_{jr}. Thus, (16) can be reformulated as follows:

$$\min_{n_{I_x} \in \mathcal{N}} \min_{x \in X} \left\{ \sum_{j \in \mathcal{N}} \sum_{g=1}^{n_{I_x}} (\sigma(g) p_j^+ - o_j) x_{jg} - \min_{n_{I_y} \in \mathcal{N}} \min_{y \in X} \sum_{j \in \mathcal{N}} \sum_{r=1}^{n_{I_y}} c_{jr}(x) y_{jr} \right\} \tag{17}$$

where $c_{jr}(x) = \sigma(r) p_j^- + (p_j^+ - p_j^-) \sum_{g=1}^{n_{I_x}} \min\{\sigma(g), \sigma(r)\} x_{jg} - o_j$

Given the values of n and m, we can convert the parallel machine problem of the maximum regret value into a single machine problem in $O(n^2)$ times since there are n possible outcomes for n_{I_x} and n_{I_y}, respectively. Once we convert the problem into an equivalent robust single machine problem, the properties in Choi and Chung (2016) can be used for the robust single machine problem.

5. Approximation algorithms

5.1. A 2-approximation algorithm for the discrete scenario

Since the problem **R-PMO** with the discrete scenario is NP-hard, we develop a 2-approximation algorithm.

The problem **R-PMO** can be formulated as the following integer programming (IP):

$$[\textbf{IP}] \text{ minimize} \quad R_{\max} \tag{18a}$$

$$\text{s.t.} \quad R_{\max} \geq F^s(x) - F^s(*) \quad \forall\, s \in S \tag{18b}$$

$$x_{ijk} \in \{0, 1\} \quad \forall\, i \in \mathcal{M},\, j \in \mathcal{N},\, k \in \mathcal{N} \tag{18c}$$

where $F^s(*)$ can be determined in $O((nm)^2)$ as shown in Subsection 3.4.

We can replace the integrality constraint (18c) in the problem **IP** with the following relaxed constraints: $x_{ijk} \geq 0, \forall\, i \in \mathcal{M}, j \in \mathcal{N}, k \in \mathcal{N}$. The resulting problem (referred to as **LP**) is a linear programming model. Let $x^{LP} = \{x_{ijk}^{LP}, i \in \mathcal{M}, j \in \mathcal{N}, k \in \mathcal{N}\}$ be an optimal solution for **LP**, which is the relaxation of the **IP** above. Then, we develop the following approximation algorithm:

Algorithm 2 Approximation algorithm for the problem with the discrete scenario

1: Obtain x^{LP} for the **LP** of the problem **R-PMO**.

2: Construct $\widetilde{x}$ by letting

$$\widetilde{x}_{ijk} = \begin{cases} 1 & amp; \text{if } x^{LP}_{ijk} \geq \frac{1}{2} \\ 0 & amp; \text{otherwise} \end{cases} \tag{19}$$

Now, we prove the 2-approximability of Algorithm 2.

$$\begin{aligned} F^s(x) - F^s(*) &= \sum_{i\in\mathcal{M}}\sum_{j\in\mathcal{N}}\sum_{k\in\mathcal{N}} kp^s_j x_{ijk} + \sum_{j\in\mathcal{N}} o_j \left(1 - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk}\right) - \min_{y\in X} F^s(y) \\ &= \sum_{j\in\mathcal{N}} C^s_j \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} + \sum_{j\in\mathcal{N}} o_j \left(1 - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk}\right) - \sum_{j\in\mathcal{N}} \min\{C^s_j, o_j\} \end{aligned} \tag{20}$$

where C^s_j is the completion time of job j in schedule x for scenario s.

Let A be the set of jobs $j \in \mathcal{N}$ such that $x^{LP}_{ijk} \geq \frac{1}{2}$ and B be the set of the rest jobs. Let R^{*LP} be the optimal value of the relaxation model, **LP**. Then, for each $s \in S$,

$$\begin{aligned} R^{*LP} &\geq \sum_{j\in\mathcal{N}} o_j + \sum_{j\in\mathcal{N}}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk} - \sum_{j\in\mathcal{N}} \min\{C^s_j, o_j\} \\ &= \sum_{j\in\mathcal{N}} o_j + \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) \widetilde{x}_{ijk} - \sum_{j\in\mathcal{N}} \min\{C^s_j, o_j\} - \Delta^s \end{aligned} \tag{21}$$

where $\Delta^s = \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j)(1 - x^{LP}_{ijk}) - \sum_{j\in B}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk}$.

Now we show that $R^{*LP} \geq \Delta^s$ for each $s \in S$,

$$\begin{aligned} R^{*LP} - \Delta^s &\geq \sum_{j\in\mathcal{N}} o_j + \sum_{j\in\mathcal{N}}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk} - \sum_{j\in\mathcal{N}} \min\{C^s_j, o_j\} - \Delta^s \\ &= \sum_{j\in A} o_j + \sum_{j\in B} o_j + \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk} \\ &\quad + \sum_{j\in B}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk} - \sum_{j\in A} \min\{C^s_j, o_j\} - \sum_{j\in B} \min\{C^s_j, o_j\} - \Delta^s \\ &= \sum_{j\in A} o_j + \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk} - \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j)(1 - x^{LP}_{ijk}) - \sum_{j\in A} \min\{C^s_j, o_j\} \\ &\quad + \sum_{j\in B} o_j + \sum_{j\in B}\sum_{i\in\mathcal{M}}\sum_{k\in N} (C^s_j - o_j) x^{LP}_{ijk} + \sum_{j\in B}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j) x^{LP}_{ijk} - \sum_{j\in B} \min\{C^s_j, o_j\} \\ &= \sum_{j\in A} \left[o_j + \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j)(-1 + 2x^{LP}_{ijk}) - \min\{C^s_j, o_j\} \right] \\ &\quad + \sum_{j\in B} \left[o_j + \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} 2(C^s_j - o_j) x^{LP}_{ijk} - \min\{C^s_j, o_j\} \right] \end{aligned} \tag{22}$$

Four different cases should be considered and they are presented as follows:

- For $j \in A$ such that $C^s_j - o_j \geq 0$, since $x^{LP}_{ijk} \geq \frac{1}{2}$, the first term in the right-hand side of (22),

$$o_j + \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j)(-1 + 2x^{LP}_{ijk}) - \min\{C^s_j, o_j\} = \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} (C^s_j - o_j)(-1 + 2x^{LP}_{ijk}) \geq 0.$$

- For $j \in A$ such that $C_j^s - o_j \leq 0$, since $x_{ijk}^{LP} < \frac{1}{2}$, the first term in the right-hand side of (22),

$$o_j + (C_j^s - o_j)\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}(-1 + 2x_{ijk}^{LP}) - \min\{C_j^s, o_j\} = (o_j - C_j^s)\left[1 - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}(-1 + 2x_{ijk}^{LP})\right] \geq 0.$$

- For $j \in B$ such that $C_j^s - o_j \geq 0$, since $x_{ijk}^{LP} \geq 0$, the second term in the right-hand side of (22),

$$o_j + 2(C_j^s - o_j)\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}x_{ijk}^{LP} - \min\{C_j^s, o_j\} = 2(C_j^s - o_j)\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}x_{ijk}^{LP} \geq 0.$$

- For $j \in B$ such that $C_j^s - o_j \leq 0$, since $x_{ijk}^{LP} < \frac{1}{2}$, the second term in the right-hand side of (22),

$$o_j + 2(C_j^s - o_j)\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}x_{ijk}^{LP} - \min\{C_j^s, o_j\} = -(C_j^s - o_j)\left(1 - 2\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}x_{ijk}^{LP}\right) \geq 0.$$

Thus, we obtain that $R^{*LP} \geq \Delta^s$ for any $s \in S$, together with the transformation of inequality (21) $R^{*LP} + \Delta^s \geq \sum_{j\in\mathcal{N}} o_j + \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}(C_j^s - o^j)\widetilde{x}_{ijk} - \sum_{j\in\mathcal{N}}\min\{C_j^s, o_j\}$. Then, inequality (21) can be rewritten as follows: for each $s \in S$,

$$2R^{*LP} \geq \sum_{j\in\mathcal{N}} o_j + \sum_{j\in A}\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}}(C_j^s - o^j)\widetilde{x}_{ijk} - \sum_{j\in\mathcal{N}}\min\{C_j^s, o_j\} \tag{23}$$

By inequality (23) and $R^{*LP} \leq R^*_{\max}$, where $R^*_{\max}$ is the optimal regret value of the problem **IP**, we have

$$R(\widetilde{x}) \leq 2R^{*LP} \leq 2R^*_{\max} \tag{24}$$

This leads to $R(\widetilde{x})/R^*_{\max} \leq 2$. Since $R^*_{\max}$ is the optimal regret value of the problem **IP**, we can conclude that Algorithm 2 is a 2-approximation algorithm for the problem **R-PMO** with the discrete scenario. Hence, we have the following theorem:

THEOREM 3 *The problem **R-PMO** with the discrete scenario has a 2-approximation algorithm.*

5.2. A 2-approximation algorithm for the interval scenario

Next, we develop a 2-approximation algorithm for the problem **R-PMO** with the interval scenario for the processing time uncertainty.

Let s^M be the mid-point scenario, i.e. $p_j^{s^M} = \frac{1}{2}(p_j^+ + p_j^-)$ for all $j \in \mathcal{N}$, and let y be an optimal schedule under s^M. We will show that for any feasible schedule x it holds that $R_{\max}(y) \leq 2R_{\max}(x)$.

Since y donate the optimal schedule under the mid-point scenario s^M, we have, $F^{s^M}(x) - F^{s^M}(y) \geq 0$, i.e. $\sum_{j\in\mathcal{N}}(x_j - y_j)(p_j^+ + p_j^-) + 2\sum_{j\in\mathcal{N}}(\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk})o_j \geq 0$, which is equivalent to the following inequality:

$$\begin{aligned}&\sum_{\{j:x_j>y_j\}}(x_j - y_j)p_j^+ + \sum_{\{j:x_j<y_j\}}(x_j - y_j)p_j^- + \sum_{j\in\mathcal{N}}\left(\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk}\right)o_j \\ &\geq \sum_{\{j:x_j<y_j\}}(y_j - x_j)p_j^+ + \sum_{\{j:x_j>y_j\}}(y_j - x_j)p_j^- + \sum_{j\in\mathcal{N}}\left(\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf}\right)o_j \end{aligned} \tag{25}$$

According to Lemma 1 in Section 3, one can easily prove that for any two feasible schedules x and y, the following inequality holds:

$$R_{\max}(x) \geq \sum_{\{j:x_j>y_j\}}(x_j - y_j)p_j^+ + \sum_{\{j:x_j<y_j\}}(x_j - y_j)p_j^- + \sum_{j\in\mathcal{N}}\left(\sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf} - \sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk}\right)o_j \tag{26}$$

The following inequality follows from Lemma 1:

$$F^{s_y}(y) \leq F^{s_y}(x) + \sum_{\{j:y_j>x_j\}}(y_j - x_j)p_j^+ + \sum_{\{j:y_j<x_j\}}(y_j - x_j)p_j^- + \sum_{j\in\mathcal{N}}\left(\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf}\right)o_j \tag{27}$$

where s_y is the worse-case scenario of schedule y.

Subtracting $F^{s_y}(*)$ from both sides of (27) yields $R_{\max}(y) \leq F^{s_y}(x) - F^{s_y}(*) + \sum_{\{j:y_j>x_j\}}(y_j - x_j)p_j^+ + \sum_{\{j:y_j<x_j\}}(y_j - x_j)p_j^- + \sum_{j\in\mathcal{N}}(\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf})o_j$, which gives (28), together with $R_{\max}(x) \geq F^{s_y}(x) - F^{s_y}(*)$:

$$R_{\max}(y) \leq R_{\max}(x) + \sum_{\{j:y_j>x_j\}}(y_j - x_j)p_j^+ + \sum_{\{j:y_j<x_j\}}(y_j - x_j)p_j^- + \sum_{j\in\mathcal{N}}\left(\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf}\right)o_j \tag{28}$$

Now applying formula (26) to (25) we can obtain

$$R_{\max}(x) \geq \sum_{\{j:y_j>x_j\}}(y_j - x_j)p_j^+ + \sum_{\{j:y_j<x_j\}}(y_j - x_j)p_j^- + \sum_{j\in\mathcal{N}}\left(\sum_{i\in\mathcal{M}}\sum_{k\in\mathcal{N}} x_{ijk} - \sum_{h\in\mathcal{M}}\sum_{f\in\mathcal{N}} y_{hjf}\right)o_j \tag{29}$$

Now inequalities (29) and (28) yield $R_{\max}(y) \leq R_{\max}(x) + R_{\max}(x) = 2R_{\max}(x)$, which leads to $R_{\max}(y)/R_{\max}(x) \leq 2$. Since x is an any arbitrary feasible schedule (including the optimal one), we have $R_{\max}(y)/R(x^*) \leq 2$. Therefore, we have the following theorem:

THEOREM 4 *The problem **R-PMO** with the interval scenario has a 2-approximation algorithm.*

Since the maximum regret of any schedule can be obtained in $O((nm)^3)$ time as shown in Section 3, the time complexity of this 2-approximation algorithm is $O((nm)^3)$.

6. Conclusion

This study investigated the robust (min-max regret) version of identical parallel machine scheduling problem with an outsourcing option and uncertain processing times. The performance measurement was expressed as the total cost including the total completion time of precessing jobs in-house and the subcontracting cost of outsourced jobs. Two types of processing time uncertainty are considered: interval and discrete scenarios. The objective was to obtain robust decisions that minimise the absolute deviation of total cost from the optimal solution under the worst-case scenario. We proved the worst-case scenario for any feasible solution, which is helpful for reducing search efforts. For the interval scenario, we proved that the maximum regret value can be obtained in polynomial time for any feasible schedule. We also proved that for any discrete scenario, the minimum total cost can be obtained in polynomial time. To reduce the computational time, we further transformed the problem with interval scenario into an equivalent robust single machine scheduling problem. In addition, for both discrete and interval scenarios, 2-approximation algorithms were developed.

For future research, it would be interesting to investigate the min-max regret version of parallel machine scheduling problem with both uncertain outsourcing cost and uncertain processing time. Another interesting direction is to develop the exact algorithm for the identical parallel machine scheduling problem with an outsourcing option and with both uncertain processing time and outsourcing cost.

Acknowledgements

The authors would like to thank the editors and anonymous referees for their constructive comments which contributed to improve the quality of this paper.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 71571135 and 71971155 and also supported by the Fundamental Research Funds for the Central Universities.

References

Aissi, H., C. Bazgan, and D. Vanderpooten. 2009. "Min-Max and Min-Max Regret Versions of Combinatorial Optimization Problems: A Survey." *European Journal of Operational Research* 197: 427–438.

Alon, N., Y. Azar, G. J. Woeginger, and T. Yadid. 1998. "Approximation Schemes for Scheduling on Parallel Machines." *Journal of Scheduling* 1 (1): 56–66.

Averbakh, I. 2006. "The Minmax Regret Permutation Flow-Shop Problem with Two Jobs." *European Journal of Operational Research* 169: 761–766.

Bräsel, H., Y. N. Sotskov, and F. Werner. 1996. "Stability of a Schedule Minimizing Mean Flow Time." *Mathematical and Computer Modelling* 24 (10): 39–53.

Cachon, G. P., and P. T. Harker. 2002. "Competition and Outsourcing with Scale Economies." *Management Science* 48: 1314–1333.

Chen, Z. L., and C. L. Li. 2008. "Scheduling with Subcontracting Options." *IIE Transactions* 40 (12): 1171–1184.

Choi, B. C., and J. Chung. 2011. "Two-Machine Flow Shop Scheduling Problem with an Outsourcing Option." *European Journal of Operational Research* 213: 66–72.

Choi, B. C., and K. Chung. 2016. "Min-Max Regret Version of a Scheduling Problem with Outsourcing Decisions Under Processing Time Uncertainty." *European Journal of Operational Research* 252: 367–375.

Choi, B. C., and M. J. Park. 2014. "Outsourcing Decisions in m-Machine Permutation Flow Shop Scheduling Problems with Machine-Dependent Processing Times." *Asia-Pacific Journal of Operational Research* 31 (4): 1450028-1–1450028-10.

Chung, D. Y., and B. C. Choi. 2013. "Outsourcing and Scheduling for Two-Machine Ordered Flow Shop Scheduling Problems." *European Journal of Operational Research* 226 (1): 46–52.

Chung, D. Y, K. Lee, K. Shin, and J. Park. 2005. "A New Approach to Job Shop Scheduling Problems with Due Date Constraints Considering Operating Subcontracts." *International Journal of Production Economics* 98: 238–250.

Conde, E. 2014. "A MIP Formulation for the Minmax Regret Total Completion Time in Scheduling with Unrelated Parallel Machines." *Optimization Letters* 8 (4): 1577–1589.

Conde, E. 2019. "Robust Minmax Regret Combinatorial Optimization Problems with a Resource Dependent Uncertainty Polyhedron of Scenarios." *Computers & Operations Research* 103: 97–108.

Drwal, M. 2018. "Robust Scheduling to Minimize the Weighted Number of Late Jobs with Interval Due-Date Uncertainty." *Computers & Operations Research* 91: 13–20.

Drwal, M., and R. Rischke. 2016. "Complexity of Interval Minmax Regret Scheduling on Parallel Identical Machines with Total Completion Time Criterion." *Operations Research Letters* 44: 354–358.

Feizollahi, M. J., and I. Averbakh. 2014. "The Robust (Minmax Regret) Quadratic Assignment Problem with Interval Flows." *INFORMS Journal on Computing* 26 (2): 321–335.

Feng, X., F. F. Zheng, and Y. F. Xu. 2016. "Robust Scheduling of a Two-Stage Hybrid Flow Shop with Uncertain Interval Processing Time." *International Journal of Production Research* 54 (12): 3706–3717.

Furini, F., M. Iori, S. Martello, and M. Yagiura. 2015. "Heuristic and Exact Algorithms for the Interval Min-Max Regret Knapsack Problem." *INFORMS Journal on Computing* 27 (2): 392–405.

Gholami, O., Y. N. Sotskov, F. Werner, and A. S. Zatsiupo. 2019. "Heuristic Algorithms to Maximize Revenue and the Number of Jobs Processed on Parallel Machines." *Automation and Remote Control* 80 (2): 297–316.

Gurevsky, E., O. Battaïa, and A. Dolgui. 2012. "Balancing of Simple Assembly Lines Under Variations of Task Processing Times." *Annals of Operations Research* 201: 265–286.

Kacem, I., and H. Kellerer. 2019. "Complexity Results for Common Due Date Scheduling Problems with Interval Data and Minmax Regret Criterion." *Discrete Applied Mathematics* 264: 76–89.

Kasperski, A., A. Kurpisz, and P. Zieliński. 2012. "Approximating a Two-Machine Flow Shop Scheduling Under Discrete Scenario Uncertainty." *European Journal of Operational Research* 217 (1): 36–43.

Kasperski, A., and P. Zieliński. 2008. "A 2-Approximation Algorithm for Interval Data Minmax Regret Sequencing Problems with the Total Flow Time Criterion." *Operations Research Letters* 36: 343–344.

Kouvelis, P., R. L. Daniels, and G. Vairaktarakis. 2000. "Robust Scheduling of a Two-Machine Flow Shop with Uncertain Processing Times." *IIE Transactions* 32 (5): 421–432.

Lai, T. C., and Y. N. Sotskov. 1999. "Sequencing with Uncertain Numerical Data for Makespan Minimization." *Journal of the Operational Research Society* 50: 230–243.

Lai, T. C., Y. N. Sotskov, and A. Dolgui. 2019. "The Stability Radius of An Optimal Line Balance with Maximum Efficiency for a Simple Assembly Line." *European Journal of Operational Research* 274: 466–481.

Lai, T. C., Y. N. Sotskov, A. Dolgui, and A. Zatsiupa. 2016. "Stability Radii of Optimal Assembly Line Balances with a Fixed Workstation Set." *International Journal of Production Economics* 182: 356–371.

Lai, T. C., Y. N. Sotskov, N. G. Egorova, and F. Werner. 2018. "The Optimality Box in Uncertain Data for Minimising the Sum of the Weighted Job Completion Times." *International Journal of Production Research* 56 (19): 6336–6362.

Lai, T. C., Y. N. Sotskov, N. Y. Sotskova, and F. Werner. 1997. "Optimal Makespan Scheduling with Given Bounds of Processing Times." *Mathematical and Computer Modelling* 26 (3): 67–86.

Lee, K., and B. C. Choi. 2011. "Two-Stage Production Scheduling with an Outsourcing Option." *European Journal of Operational Research* 213: 489–497.

Lee, I. S., and C. S. Sung. 2008a. "Minimizing Due Date Related Measures for a Single Machine Scheduling Problem with Outsourcing Allowed." *European Journal of Operational Research* 186: 931–952.

Lee, I. S., and C. S. Sung. 2008b. "Single Machine Scheduling with Outsourcing Allowed." *International Journal of Production Economics* 111: 623–634.

Liu, L. 2019. "Outsourcing and Rescheduling for a Two-Machine Flow Shop with the Disruption of New Arriving Jobs: A Hybrid Variable Neighborhood Search Algorithm." *Computers & Industrial Engineering* 130: 198–221.

Mokhtari, H., and I. N. K. Abadi. 2013. "Scheduling with an Outsourcing Option on Both Manufacturer and Subcontractors." *Computers & Operations Research* 40 (5): 1234–1242.

Papadimitriou, C., and K. Steiglitz. 1998. *Combinatorial Optimization: Algorithms and Complexity*. North Chelmsford, MA: Courier Corporation.

Park, M. J., and B. C. Choi. 2017. "A Single-Machine Scheduling Problem with Uncertainty in Processing Times and Outsourcing Costs." *Mathematical Problems in Engineering* 2017: 1–8.

Qi, X. 2012. "Decentralized Subcontractor Scheduling with Divisible Jobs." *Journal of Scheduling* 18: 497–511.

Ruiz-Torres, A. J., J. C. Ho, and F. J. López. 2006. "Generating Pareto Schedules with Outsource and Internal Parallel Machines." *International Journal of Production Economics* 103: 810–825.

Siepak, M., and J. Józefczyk. 2014. "Solution Algorithms for Unrelated Machines Minmax Regret Scheduling Problem with Interval Processing Times and the Total Flow Time Criterion." *Annals of Operations Research* 222 (1): 517–533.

Sotskov, Y. N., A. Dolgui, T. C. Lai, and A. Zatsiupa. 2015. "Enumerations and Stability Analysis of Feasible and Optimal Line Balances for Simple Assembly Lines." *Computers & Industrial Engineering* 90: 241–258.

Sotskov, Y. N., A. Dolgui, and M. C. Portmann. 2006. "Stability Analysis of An Optimal Balance for an Assembly Line with Fixed Cycle Time." *European Journal of Operational Research* 168: 783–797.

Sotskov, Y. N., N. G. Egorova, and T. C. Lai. 2014. "Minimizing Total Weighted Flow Time of a Set of Jobs with Interval Processing Times." *Mathematical & Computer Modelling* 50 (3): 556–573.

Sotskov, Y. N., and T. C. Lai. 2012. "Minimizing Total Weighted Flow Time Under Uncertainty Using Dominance and a Stability Box." *Computers & Operations Research* 39: 1271–1289.

Sotskov, Y. N., V. K. Leontev, and E. N. Gordeev. 1995. "Some Concepts of Stability Analysis in Combinatorial Optimization." *Discrete Applied Mathematics* 58: 169–190.

Sotskov, Y. N., N. Y. Sotskova, and F. Werner. 1997. "Stability of An Optimal Schedule in a Job Shop." *Omega* 25 (4): 397–414.

Tavares Neto, R. F., M. Godinho Filho, and F. M. Da Silva. 2015. "An Ant Colony Optimization Approach for the Parallel Machine Scheduling Problem with Outsourcing Allowed." *Journal of Intelligent Manufacturing* 26 (3): 527–538.

Wang, X. L., S. J. Geng, and T. C. E. Cheng. 2018. "Negotiation Mechanisms for an Order Subcontracting and Scheduling Problem." *Omega* 77: 154–167.

Xiao, W. L., and C. Gaimon. 2013. "The Effect of Learning and Integration Investment on Manufacturing Outsourcing Decisions: A Game Theoretic Approach." *Production & Operations Management* 22 (6): 1576–1592.

Xu, X. Q., W. T. Cui, J. Lin, and Y. J. Qian. 2013. "Robust Makespan Minimisation in Identical Parallel Machine Scheduling Problem with Interval Data." *International Journal of Production Research* 51 (12): 3532–3548.

Xu, X. Q., J. Lin, and W. T. Cui. 2014. "Hedge Against Total Flow Time Uncertainty of the Uniform Parallel Machine Scheduling Problem with Interval Data." *International Journal of Production Research* 52 (19): 5611–5625.

Yue, F., S. J. Song, Y. L. Zhang, J. N. D. Gupta, and R. Chiong. 2018. "Robust Single Machine Scheduling with Uncertain Release Times for Minimising the Maximum Waiting Time." *International Journal of Production Research* 56 (16): 5576–5592.