Robust optimization for identical parallel machine scheduling with uncertain processing time

Kwang-Kyu SEO* and Byung Do CHUNG**

*Department of Management Engineering, Sangmyung University, 31 Sangmyungdae-gil, Dongnam-gu, Cheonan, Choongnam 330-720, Republic of Korea
**Department of Systems Management Engineering, Sungkyunkwan University, 2066 Seobu-ro, Jangan-gu, Suwon 440-746, Republic of Korea
E-mail: bd.chung@skku.edu
Received 7 October 2013

Abstract
In this paper, we propose to apply robust optimization approaches to the problem of identical parallel machine scheduling with processing time uncertainty. Box uncertainty and cardinality-constrained uncertainty are considered, and robust counterpart is reformulated as deterministic MILP problems. We explore the impact of the protection level, and show the trade-off between robustness and conservativeness. The results of numerical experiments demonstrate that the robust counterpart with cardinality-constrained uncertainty outperforms that with box uncertainty with respect to the mean and standard deviation of realized objective values. However, the robust counterpart with box uncertainty has an advantage in that it requires less computational efforts to solve the problem.

Key words: Identical parallel machine, Machine scheduling, Uncertain processing time, Robust optimization, Makespan minimization

1. Introduction

In the identical parallel machine scheduling problem, a number of jobs are assigned to identical machines without preemption to minimize the maximum makespan. The optimal scheduling problem has been importantly considered since it can be applied to various industries other than production lines, like social science, computer science, and healthcare (Mokotoff, 2004). The problem has been modified into several variants due to its theoretical and practical importance (for example, Chang et al., 2004; Chung et al., 2009; Hu et al., 2010, Özlören and Webster, 2010).

A typical approach to the parallel machine scheduling problem assumes all data are known deterministically. However, this assumption may be too strict in practice due to machine conditions or worker skill, especially in the manufacturing systems with human operator (Iwamura et al., 2012). Ben-Tal and Nemirovski (2000) showed that a deterministic optimal solution can be severely affected by a small perturbation of parameters. For the reason, stochastic programming has been adopted to overcome such drawbacks of deterministic approaches. For example, Skutella and Uetz (2005) and Bouyahia et al. (2010) considered probability distributions for the processing time of jobs to minimize the expected weighted completion time. Al-Khamis and M’Hallah (2011) developed a two-stage stochastic programming model to maximize the expected net profit under due date uncertainty. A solution algorithm based on sampled scenarios was developed to deal with uncertainty. Unfortunately, it may not be able to identify the probability distribution or sampled scenarios when sufficient information or historical data is not available. In addition, the computational complexity tends to increase when a stochastic approach is applied.

We propose to apply two robust optimization (RO) approaches to address the identical parallel machine scheduling problem with processing time uncertainty, which belongs to a crude uncertainty set. The idea of RO was first proposed by Soyster (1973) and the fields of RO methodologies have been actively carried out after the works by Ben-Tan and Nemirovski (1999, 2000) and Bertsimas and Sim (2003, 2004). RO differs from stochastic programming in that the objective is to find the best worst-case solution, which is a solution that performs well even in a worst case scenario. RO has several advantages. First, RO does not require probability distribution information or scenarios to
describe parameter uncertainty. For instance, when a new product is developed, we should estimate uncertain processing time without appropriate probability distribution until substantial data are available. Second, the robust counterpart of an uncertain mixed integer linear programming (MILP) model can be reformulated as a deterministic MILP problem. Thus, it is computationally less expensive compared with stochastic programming approaches or heuristic algorithms to handle uncertainty. It can be efficiently solved by commercial software packages. Third, RO provides an uncertainty-immunized solution, which remains feasible in any realization of parameter values from a pre-determined uncertainty set (Refer to the more detailed reviews on robust optimization by Ben-Tal et al. (2009) and Bertsimas et al. (2011)).

Thanks to the advantages, RO research has been successfully applied in various fields, including transportation planning (Yao et al., 2009), supply chain management (Bertsimas and Thiele, 2006), and portfolio management (Fabozzi et al., 2010). However, there has been scarce research on RO approaches to machine scheduling problems. Lu et al. (2012) presented an RO model for a single machine to minimize the total flow time. A simulated annealing-based heuristic algorithm was developed. Lin et al. (2004) addressed a scheduling problem of chemical processes under a box uncertainty set, and derived an equivalent second-order cone mixed integer program. Since it is a nonlinear mixed integer program, it may not be an attractive approach to solve large-scale robust discrete optimization problems (Bertsimas and Sim, 2004). For the identical parallel machine scheduling problem, Xu et al. (2013) considered a robust min-max regret problem for identical parallel machine scheduling with uncertain processing time interval. They provided two algorithms based on a general iterative relaxation procedure. Since their approach generates the worst case scenario iteratively, this approach may be computationally expensive. Therefore, we propose to apply robust optimization approaches to find more efficiently the best worst case solution under uncertain processing time.

The main contributions of our work are as follows:

- We first develop two RO research models for the identical parallel machine problem under processing uncertainty based on the mathematical theories on RO, which provides a computational advantage compared with iterative algorithms and sampling based approaches.
- We compared two RO models based on box uncertainty sets and cardinality constrained uncertainty sets. The solution from box uncertainty is commonly criticized that it is more conservative than cardinality constrained uncertainty. However, it is observed that the conservativeness issue depends on the problem size and uncertainty level and the RO model with cardinality constrained uncertainty is computationally expensive.
- The numerical results contribute to the growing body of robust optimization and production scheduling problems under uncertainty. The set-based uncertainty is the most appropriate notion in cases where a production system is not stable, or sufficient data is not available to estimate the probability distribution. We have found that RO can be an important and effective way to solve the scheduling problem with the assumptions of box or cardinality-constrained uncertainty sets.

The rest of this paper is organized as follows. Section 2 describes the deterministic identical parallel machine scheduling model and its robust counterpart. In Section 3, we explain two uncertainty sets – a box uncertainty set and a cardinality uncertainty set. Then, robust counterpart is reformulated as equivalent deterministic MILP models. Section 4 provides numerical experiments and compares the two robust counterparts. Finally, Section 5 discusses the proposed methods.

### 2. Identical parallel machine scheduling model

In this section, we present a deterministic identical parallel machine scheduling problem, as well as the robust counterpart of the problem after introducing processing time uncertainty. The main notations for the problem used throughout the paper are stated below:

\[ J_i \quad \text{Job } i, \quad i \in N = \{1,...,n\} \]

\[ M_j \quad \text{Machine } j, \quad j \in M = \{1,...,m\} \]

\[ p_i \quad \text{Nominal processing time of job } J_i \]

\[ \hat{p}_i \quad \text{Uncertain processing time of job } J_i \]

\[ \overline{p}_i \quad \text{Maximum processing time of job } J_i \]
\( p_i \)  
Minimum processing time of job \( J_i \)

\( C_j \)  
Completion time of machine \( M_j \)

\( x_{ij} \)  
Assignment decision variable

### 2.1 Deterministic model

Consider a set of \( n \) independent and non-preemptive jobs and a set of \( m \) identical machines with the same processing time for the same job. It is assumed that each job can be scheduled on any machine, and each machine can process only one job at a time. Then, the identical parallel machine scheduling problem becomes the following deterministic mixed-integer linear programming (D-MILP) problem:

\[
\begin{align*}
\min r \\
\text{s.t.} \sum_{i=1}^{n} p_i x_{ij} & \leq r, \quad j = 1, 2, \ldots, m \\
\sum_{j=1}^{m} x_{ij} & = 1, \quad i = 1, 2, \ldots, n \\
x_{ij} & \in \{0, 1\}, \quad i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, m
\end{align*}
\]

The binary decision variable \( x_{ij} \) is 1 if job \( i \) is assigned to the machine \( j \), and 0 otherwise. \( C_j \), the completion time of machine \( j \) is \( \sum_{i=1}^{n} p_i x_{ij} \), the left-hand side of Eq. (2). Since we are interested in minimizing the maximum makespan, the objective function is \( \min x \left( \max_{j \in M} C_j \right) \), which is equivalent to Eq. (1) and Eq. (2). Using Eq. (3), every job is assigned to exactly one machine.

### 2.2 Robust counterpart model

As mentioned in the previous section, it is clear that uncertain factors, especially perturbations in processing time, need to be considered in scheduling problems. In this paper, we assume that the uncertain processing time \( \tilde{p}_i \) belongs to a bounded uncertainty set \( U_{\tilde{p}} \) for all \( i \in N = \{1, \ldots, n\} \). RO finds the robust optimal solution satisfying constraints with all possible uncertain parameter values. An uncertain processing time appears in Eq. (2) and a robust counterpart (R-MILP) can be formulated as:

\[
\begin{align*}
\min r \\
\text{s.t.} \sum_{i=1}^{n} \tilde{p}_i x_{ij} & \leq r, \quad j = 1, 2, \ldots, m; \quad \tilde{p}_i \in U_{\tilde{p}}, \quad i = 1, 2, \ldots, n
\end{align*}
\]

However, R-MILP is a type of semi-infinite problem and cannot be solved directly, since it has an infinite number of constraints due to the bounded uncertainty set \( U_{\tilde{p}} \). Therefore, we need to reformulate R-MILP so that a finite number of constraints remains in the model.

### 3. Equivalent robust counterpart models

There have been several RO approaches to obtain an equivalent robust counterpart model that requires less computational effort (Ben-Tal et al., 2009; Bertsimas et al., 2011). We apply two approaches appropriate for the identical parallel machine scheduling problem by Ben-Tal and Nemirovski (2000) and Bertsimas and Sim (2003, 2004). Note that the RO approach by Ben-Tal and Nemirovski (2000) was originally proposed for uncertain linear programming problems, but it is also applicable to MILP models as shown in Section 3.1.
3.1 Robust counterpart with box uncertainty

First, we assume box uncertainty (or interval uncertainty) sets for uncertain parameters as shown in Eq. (6).

\[ U_i = \{ \tilde{p}_i : \underline{p}_i \leq \tilde{p}_i \leq \bar{p}_i \} \]  

(6)

, where \( \underline{p}_i \) and \( \bar{p}_i \) are the minimum and maximum processing times of job \( i \), respectively.

**PROPOSITION 1.** Given a pre-defined box uncertainty set \( U_i \), the robust counterpart R-MILP1 can be reformulated as a deterministic MILP problem with maximum processing time.

**Proof.** Equation (5) is the only constraint affected by the uncertain parameters. Since \( p_i \) belongs to a box uncertainty set and \( x_{ij} \) is a binary variable, meaning \( x_{ij} \) is non-negative, we have the following relationship.

\[
\sum_{i=1}^{n} \tilde{p}_i x_{ij} = \sum_{i=1}^{n} \underline{p}_i x_{ij} + \sum_{i=1}^{n} (\bar{p}_i - \underline{p}_i) \eta x_{ij} \\
\leq \sum_{i=1}^{n} \underline{p}_i x_{ij} + \sum_{i=1}^{n} (\bar{p}_i - \underline{p}_i) x_{ij} \\
= \sum_{i=1}^{n} \tilde{p}_i x_{ij} \leq r
\]

, where \( 0 \leq \eta \leq 1 \). Once the constraint \( \sum_{i=1}^{n} \tilde{p}_i x_{ij} \leq r \) is satisfied for all \( j \), \( \sum_{i=1}^{n} \tilde{p}_i x_{ij} \leq r \), \( \tilde{p}_i \in U_i \), \( i = 1, 2, ..., n \) holds for all \( j \).

Therefore, the robust counterpart R-MILP1 with the infinite number of constraints can be reformulated as a deterministic MILP problem (R-MILP2) with maximum processing time as:

\[
\begin{align*}
\min_{x,r} & \quad r \\
\text{s.t.} & \quad \sum_{i=1}^{n} \tilde{p}_i x_{ij} \leq r, \quad j = 1, 2, ..., m \\
& \quad (3) \text{ and } (4)
\end{align*}
\]

R-MILP2 provides an uncertainty-immunized optimal solution, which remains feasible whenever the realized processing time belongs to the pre-defined uncertainty set. However, the solution for a worst-case scenario in terms of processing time of all jobs (due to Eq. (7)) may be too conservative, because it is very unlikely to occur in practice.

3.2 Robust counterpart with cardinality constrained uncertainty

In this section, we apply the RO approach based on the cardinality uncertainty set, which enables us to adjust the robustness of the robust optimal solution. In order to control the level of robustness, we introduce a parameter \( \Gamma_j \) to Eq. (5) to avoid excessive conservativeness, and also assume box uncertainty sets in Eq. (6). The parameter can be a real value between 0 and \( n \). When \( \Gamma_j \) is an integer value, the following equation allows up to \( \lfloor \Gamma_j \rfloor \) of processing time to be perturbed within the uncertainty set.

\[
\sum_{i=1}^{n} \underline{p}_i x_{ij} + \beta_j \left( x_{ij}, \Gamma_j \right) \leq r, \quad j = 1, 2, ..., m
\]

\( \beta_j \left( x_{ij}, \Gamma_j \right) \) with decision variable \( y_{ij} \) in the \( j \)th constraint is defined as
\[
\beta_j(x_j, \Gamma_j) = \max_{y_j} \sum_{i=1}^{n} (p_i - p_j^*) x_j y_j
\]
(8)

\[
s.t. \sum_{i=1}^{n} y_j \leq \Gamma_j
\]

\[
0 \leq y_j \leq 1, \ i = 1, 2, ..., n
\]

The problem (8) becomes a deterministic problem D-MLIP with nominal value when \( \Gamma_j = 0 \) for all \( j \). Also, it becomes a deterministic problem R-MILP2 with maximum processing time when \( \Gamma_j = n \) for all \( j \), and it does not allow any constraint violation. We can adjust the robustness and conservativeness of the solution by changing the value of \( \Gamma_j \).

**PROPOSITION 2.** Given the parameter \( \Gamma_j \) and the pre-defined box uncertainty set \( U_i \), the robust counterpart can be reformulated as a deterministic MILP Problem

**Proof.** In the sub-problem \( \beta_j(x_j^*, \Gamma_j) \), the decision variable is \( y_j \) given a vector of \( x_j^* \) and \( \Gamma_j \), and it becomes a linear programming problem. By considering the dual problem, we have:

\[
\min \sum_{i=1}^{n} q_y
\]
(9)

\[
s.t. \ z_j + q_y \geq (p_i - p_j^*) x_j^*, \ i = 1, 2, ..., n
\]

\[
q_y \geq 0, \ i = 1, 2, ..., n
\]

\[
z_j \geq 0
\]

where \( q_y \) and \( z_j \) are dual variables. By the strong duality theorem, Problem (8) has the same objective value as Problem (9), and therefore, we can derive an equivalent deterministic formulation R-MILP3 as follows:

\[
\min \sum_{i=1}^{n} q_y
\]
(10)

\[
s.t. \ z_j + q_y \geq (p_i - p_j^*) x_j^*, \ i = 1, 2, ..., n, \ j = 1, 2, ..., m
\]

\[
q_y \geq 0, \ i = 1, 2, ..., n, \ j = 1, 2, ..., m
\]

\[
z_j \geq 0, \ j = 1, 2, ..., m
\]

(3) and (4)

Finally, the robust identical parallel machine scheduling problem, R-MILP3, becomes a deterministic MILP with \( (2 + n)m \) more variables and \( 3n \) more constraints than D-MILP.
4. Experimental analysis

This section presents numerical experiments carried out to compare the two robust counterparts derived in the previous section. The test problems are generated from Xu et al. (2013). In the numerical experiments, we consider a problem of size \( m=4,5 \) and \( n=9,12,15 \). It is also assumed that \( U_i = \{ \tilde{p}_i : 10 \leq \tilde{p}_i \leq 10(1 + \theta) \} \), where \( \theta \) is randomly chosen for the interval \([0, \theta]\). The uncertainty level \( \theta \) is chosen from the set \{0.2,0.4,0.6,0.8,1.0\}. For a given \( m,n \) and \( \theta \), we generated 20 case problems, for a total of 600 case problems formulated as R-MILP2 and R-MILP3. All problems were coded in GAMS/CPLEX and solved on a PC with an Intel 1.87GHz CPU and 2GB of RAM.

4.1 Price of robustness

First, we solve test problems by adjusting the value of \( \Gamma \) to observe the price of robustness, for which a trade-off between robustness and conservativeness exists. In the model of R-MILP3, the protection level is not a decision variable but a parameter determined by a decision maker depending on the preference of robustness. Therefore, a common way to determine the value of \( \Gamma \) is sensitivity analysis by adjusting the protection level (Bertsimas and Sim, 2004). Since we are considering identical parallel machines, we set \( \Gamma_j = \Gamma \) for analytical simplicity. We note that each \( \Gamma_j \) can take a different value by considering several scenarios (Adida and Perakis, 2006).

Figure 1 illustrates the optimal value for each \( \Gamma \). The horizontal and vertical axes represent the value of \( \Gamma \) and optimal solution \( r^* \), respectively. In the case of \( n = 9 \), the optimal value is 31.63 with a protection level of zero, or \( \Gamma = 0 \). However, the optimal value is increased by 5.8% with maximum protection, \( \Gamma = [n] \). It is intuitively true that we need to pay a price for the increased protection level, in the form of the reduced constraint violation probability, as explained in Section 3.

It is also observed that the optimal value is not a linear function of the protection level. In other words, the increase of the optimal value is marginal when the value of \( \Gamma \) is large. Therefore, a risk-taking decision maker does not need to have a high protection level in order to avoid constraint violation. For example, the change of the optimal value is around 4.7% when \( \Gamma \) is changed from 0 to 2. However, it is only 0.5% when we alter \( \Gamma \) from 3 to 9.

![Fig. 1 Minimum makespan for different values of \( \Gamma \).](image)

4.2 Performance comparison

In this section, a simulation study is performed to examine the quality of the robust solutions from R-MILP2 and R-MILP3. In the first step of the simulation experiments, we solved the robust counterparts to find a robust solution of each approach. In the second step, uncertain processing time is randomly generated. Then, the minimum makespan as a
realized objective value is computed with the robust solution and the randomly generated processing time. Specifically, 1000 sets of random processing time for each set of $m$, $n$, and $\bar{\theta}$ are generated from a uniform distribution (i.e. $U(\underline{p}, \overline{p})$) to evaluate the performance of the robust solutions.

In the first set of experiments, we test the robust solution of R-MILP3 by varying the value of $\Gamma$. The mean and standard deviation (SD) of the realized objective values are compared in Table 1. It suggests that R-MILP3 solutions with $\Gamma = 3$ or $4$ outperform the robust solution with $\Gamma = 2$ in terms of mean and standard deviation in most cases. However, we see that the performance of $\Gamma$ depends on the problem size and other parameter values like $\bar{\theta}$ in this experiments. For instance, $\Gamma = 3$ dominates the case of $\Gamma = 4$ when there are 5 machines and 12 jobs with an uncertainty level 0.6. When the uncertainty level is changed to 0.8, $\Gamma = 4$ dominates the case of $\Gamma = 3$, in contrast to the previous example. Also, the standard deviation grows as the uncertainty interval rises.

Table 1  Comparison of R-MILP3 solutions with various $\Gamma$

| $m$ | $n$ | $\bar{\theta}$ | Mean | SD |
|-----|-----|-----------------|------|----|
|     |     | $\Gamma = 2$ | $\Gamma = 3$ | $\Gamma = 4$ | $\Gamma = 2$ | $\Gamma = 3$ | $\Gamma = 4$ |
| 4   | 9   | 0.2 | 30.22 | 30.22 | 30.22 | 0.26 | 0.26 | 0.26 |
|     |     | 0.4 | 30.44 | 30.44 | 30.44 | 0.52 | 0.52 | 0.52 |
|     |     | 0.6 | 30.67 | 30.67 | 30.67 | 0.78 | 0.78 | 0.78 |
|     |     | 0.8 | 30.90 | 30.89 | 30.90 | 1.05 | 1.05 | 1.05 |
|     |     | 1.0 | 31.17 | 31.16 | 31.17 | 1.33 | 1.32 | 1.32 |
| 12  |     | 0.2 | 31.24 | 31.24 | 31.25 | 0.64 | 0.62 | 0.63 |
|     |     | 0.4 | 32.42 | 32.40 | 32.42 | 1.14 | 1.13 | 1.13 |
|     |     | 0.6 | 33.63 | 33.57 | 33.58 | 1.70 | 1.66 | 1.65 |
|     |     | 0.8 | 34.84 | 34.79 | 34.79 | 2.25 | 2.22 | 2.24 |
|     |     | 1.0 | 36.53 | 36.02 | 35.96 | 3.06 | 2.79 | 2.75 |
| 15  |     | 0.2 | 41.24 | 41.21 | 41.22 | 0.64 | 0.59 | 0.60 |
|     |     | 0.4 | 42.35 | 42.32 | 42.36 | 1.17 | 1.15 | 1.15 |
|     |     | 0.6 | 43.50 | 43.49 | 43.48 | 1.76 | 1.72 | 1.70 |
|     |     | 0.8 | 44.71 | 44.63 | 44.60 | 2.34 | 2.27 | 2.26 |
|     |     | 1.0 | 46.42 | 45.98 | 45.83 | 3.23 | 2.89 | 2.84 |
| 5   | 9   | 0.2 | 20.84 | 20.83 | 20.83 | 0.48 | 0.47 | 0.47 |
|     |     | 0.4 | 21.65 | 21.66 | 21.68 | 0.94 | 0.95 | 0.94 |
|     |     | 0.6 | 22.50 | 22.53 | 22.52 | 1.42 | 1.42 | 1.42 |
|     |     | 0.8 | 23.37 | 23.33 | 23.31 | 1.90 | 1.88 | 1.87 |
|     |     | 1.0 | 24.19 | 24.17 | 24.11 | 2.38 | 2.37 | 2.33 |
| 12  |     | 0.2 | 30.47 | 30.46 | 30.45 | 0.35 | 0.33 | 0.32 |
|     |     | 0.4 | 30.92 | 30.92 | 30.90 | 0.68 | 0.66 | 0.66 |
|     |     | 0.6 | 31.40 | 31.35 | 31.39 | 1.02 | 0.98 | 0.98 |
|     |     | 0.8 | 31.85 | 31.83 | 31.80 | 1.40 | 1.29 | 1.29 |
|     |     | 1.0 | 32.41 | 32.33 | 32.39 | 1.72 | 1.65 | 1.69 |
| 15  |     | 0.2 | 31.43 | 31.41 | 31.41 | 0.65 | 0.62 | 0.62 |
|     |     | 0.4 | 32.72 | 32.69 | 32.74 | 1.13 | 1.10 | 1.14 |
|     |     | 0.6 | 34.13 | 34.06 | 34.02 | 1.72 | 1.66 | 1.62 |
|     |     | 0.8 | 35.48 | 35.41 | 35.39 | 2.24 | 2.20 | 2.17 |
|     |     | 1.0 | 37.56 | 36.73 | 36.75 | 3.07 | 2.72 | 2.73 |

Next, we experiment with the same problems to compare R-MILP2 and R-MILP3. The value for $\Gamma$ is set to 3 in the performance comparison study. The computation time to find a robust solution is also an item of interest in this experiment. The average computation time is summarized in Table 2 together with the mean and standard deviation of the realized objective values. The trade-off between the solution quality and computation time is clearly shown. The
use of the R-MILP3 model allows for the achievement of smaller mean and standard deviation of the realized objective value. In contrast, less computation time is required when using the R-MIL2 model. For example, all R-MILP2 models are solved within 0.5 seconds. However, it takes over 360 seconds to solve the R-MILP3 model on average in the case of 5 machines, 12 jobs, and an uncertainty level of 0.4.

The common criticism of RO with box uncertainty is the robust solution may be too conservative. As shown in Proposition 1, R-MILP2 is a deterministic problem with maximum demand and, thus the criticism seems to be true in this problem as well. However, when the problem size or uncertainty level is small, we observe that R-MILP2 is more favorable than R-MILP3, since the solution performances are almost same but the computation time of MILP2 is much less than MILP3. For example, when \( m = 5 \), \( n = 12 \) and \( \bar{\theta} = 0.2 \), the means and standard deviations for both models are same. However, the computation time of R-MILP2 is only 0.2 seconds, which is much smaller than that of R-MILP3, 324.65 seconds.

| \( m \) | \( n \) | \( \bar{\theta} \) | Mean | SD | Time (sec.) |
|---|---|---|---|---|---|
| R-MILP2 | R-MILP3 | R-MILP2 | R-MILP3 | R-MILP2 | R-MILP3 |
| 4 | 9 | 0.2 | 30.22 | 30.22 | 0.26 | 0.26 | 0.19 | 7.86 |
| | | 0.4 | 30.44 | 30.44 | 0.52 | 0.52 | 0.15 | 9.13 |
| | | 0.6 | 30.67 | 30.67 | 0.78 | 0.78 | 0.16 | 6.18 |
| | | 0.8 | 30.93 | 30.89 | 1.06 | 1.05 | 0.13 | 3.07 |
| | | 1.0 | 31.26 | 31.16 | 1.43 | 1.32 | 0.10 | 2.36 |
| 12 | 2 | 0.2 | 31.33 | 31.24 | 0.74 | 0.62 | 0.06 | 0.05 |
| | | 0.4 | 32.52 | 32.40 | 1.30 | 1.13 | 0.06 | 0.10 |
| | | 0.6 | 33.67 | 33.57 | 1.78 | 1.66 | 0.10 | 0.36 |
| | | 0.8 | 35.19 | 34.79 | 2.64 | 2.22 | 0.09 | 0.98 |
| | | 1.0 | 37.51 | 36.02 | 3.63 | 2.79 | 0.10 | 1.72 |
| 15 | 2 | 0.2 | 41.34 | 41.21 | 0.75 | 0.59 | 0.09 | 0.30 |
| | | 0.4 | 42.57 | 42.32 | 1.34 | 1.15 | 0.09 | 10.02 |
| | | 0.6 | 43.80 | 43.49 | 1.94 | 1.72 | 0.10 | 16.54 |
| | | 0.8 | 46.85 | 44.63 | 3.53 | 2.27 | 0.10 | 27.05 |
| | | 1.0 | 50.85 | 45.98 | 2.89 | 2.89 | 0.10 | 17.34 |
| 5 | 9 | 0.2 | 20.90 | 20.83 | 0.54 | 0.47 | 0.09 | 1.56 |
| | | 0.4 | 21.72 | 21.66 | 0.98 | 0.95 | 0.10 | 1.44 |
| | | 0.6 | 22.61 | 22.53 | 1.48 | 1.42 | 0.12 | 2.28 |
| | | 0.8 | 23.43 | 23.33 | 1.96 | 1.88 | 0.09 | 1.73 |
| | | 1.0 | 24.23 | 24.17 | 2.40 | 2.37 | 0.09 | 1.90 |
| 12 | 0.2 | 30.46 | 30.46 | 0.33 | 0.33 | 0.20 | 324.65 |
| | | 0.4 | 30.93 | 30.92 | 0.66 | 0.66 | 0.18 | 361.16 |
| | | 0.6 | 31.41 | 31.35 | 1.00 | 0.98 | 0.15 | 257.79 |
| | | 0.8 | 31.84 | 31.83 | 1.36 | 1.29 | 0.14 | 173.78 |
| | | 1.0 | 32.46 | 32.33 | 1.77 | 1.65 | 0.10 | 61.01 |
| 15 | 0.2 | 31.51 | 31.41 | 0.72 | 0.72 | 0.07 | 0.08 |
| | | 0.4 | 32.84 | 32.69 | 1.26 | 1.10 | 0.09 | 0.57 |
| | | 0.6 | 34.16 | 34.06 | 1.77 | 1.66 | 0.11 | 8.88 |
| | | 0.8 | 35.79 | 35.41 | 2.65 | 2.20 | 0.13 | 32.74 |
| | | 1.0 | 38.04 | 36.73 | 3.29 | 2.72 | 0.12 | 57.21 |

Figure 2 through 4 depict examples of mean, standard deviation, and computation time corresponding to the various values of uncertainty level. The improvement of R-MILP3 is marginal when the uncertainty interval is small. The gap between mean values in both methods tends to increase as the interval rises. However, we cannot see such a
result in the cases of standard deviation and computation time. Also, the level of uncertainty interval rarely affects the computation time when R-MILP2 is applied.

\[
\begin{align*}
\text{(a) } m &= 4, n = 15, \overline{\theta} = 1 \\
\text{(b) } m &= 5, n = 15, \overline{\theta} = 1
\end{align*}
\]

Fig. 2 Average of the realized objective value.

\[
\begin{align*}
\text{(a) } m &= 4, n = 15, \overline{\theta} = 1 \\
\text{(b) } m &= 5, n = 15, \overline{\theta} = 1
\end{align*}
\]

Fig. 3 Standard deviation of the realized objective value.

\[
\begin{align*}
\text{(a) } m &= 4, n = 15, \overline{\theta} = 1 \\
\text{(b) } m &= 5, n = 15, \overline{\theta} = 1
\end{align*}
\]

Fig. 4 Computation time (Sec.)

5. Conclusions

This paper provides theory and computational analysis for the identical parallel machine scheduling problem with uncertain processing time. Two RO approaches have been used to find the best worst-case solutions with reasonable assumptions about the uncertain parameters including a box uncertainty set and a cardinality-constrained uncertainty set. Robust counterpart was reformulated as equivalent deterministic problems, which are still MILP. The most attractive feature of the proposed approach is that R-MILP2 does not increase the complexity of the problem compared to the nominal problem D-MILP, and R-MILP3 increases it only slightly.

Our computational results show two types of trade-offs in the RO framework: 1) the trade-off between robustness and conservativeness by varying the protection level, which is also known as the price of robustness, and 2) the trade-off between solution quality and computation time by comparing R-MILP2 and R-MILP3. R-MILP3 outperformed R-MILP2 with respect to the mean and standard deviation of the realized objective values, but took more time to solve the robust counterpart than R-MILP2.

There are several topics that need to be addressed in the future research. One of these topics is that the determination of protection level for R-MILP2. As mentioned in Section 4, in order to determine \( \Gamma \), sensitivity analysis and scenarios developments are conducted in advance. It may be helpful to measure the trade-off between uncertainty and robustness and a mathematical model to find proper value to achieve the level of measurement. Another interesting research topic is uncertain data with partial distribution information. In this paper, we assume that only
crude uncertainty set is known. However, in some cases where partial information about uncertain parameter can be obtained, other distributionally RO approaches may be applied (e.g. Calafiore and Ghaoui, 2006; Delage and Ye, 2010). Finally, additional work is needed to develop RO methods for various machine scheduling problems (e.g. a non-identical machine scheduling problem and a machine scheduling problem with blocking and failure).

Acknowledgment

This paper was supported by Faculty Research Fund, Sungkyunkwan University, 2013.

References

Adida, E. and Perakis, G., A robust optimization approach to dynamic pricing and inventory control with no backorders, Mathematical Programming, Vol. 107, No. 1-2(2006), pp. 97-129.

Al-Khamis, T. and M’Hallah, R., A two-stage stochastic programming model for the parallel machine scheduling problem with machine capacity, Computers & Operations Research, Vol. 38, No. 12 (2011), pp.1747-1759.

Ben-Tal, A. and Nemirovski, A., Robust solutions of uncertain linear programs, Operations Research Letters, Vol. 25, No. 1 (1999), pp. 1-13.

Ben-Tal, A. and Nemirovski, A., Robust solutions of linear programming problems contaminated with uncertain data, Mathematical Programming Vol. 88, No. 3 (2000), pp. 411-424.

Ben-Tal, A., El Ghaoui, L. and Nemirovski, A., Robust optimization (2009), Princeton.

Bertsimas, D., Brown, D.B. and Caramanis, C., Theory and applications of robust optimization, SIAM Review, Vol. 53, No. 3 (2011), pp. 464-501.

Bertsimas, D. and Sim, M., Robust discrete optimization and network flows, Mathematical Programming, Vol. 98, No. 1-3 (2003), pp. 49–71.

Bertsimas, D. and Sim, M., The price of robustness, Operations Research, Vol. 52, No. 1 (2004), pp.35-53.

Bertsimas, D. and Thiele, A., A robust optimization approach to inventory theory, Operations Research, Vol. 54, No. 1 (2006), pp. 150-168.

Bouyahia, Z., Bellalouma, M., Jailet, P. and Ghedira, K., A priori parallel machines scheduling, Computers & Industrial Engineering, Vol. 58, No. 3 (2010), pp.488-500.

Calafiore, G.C., and El Ghaoui, L., On distributionally robust chance-constrained linear programs, Journal of Optimization Theory and Applications, Vol. 130, No. 1 (2006), pp. 1-22.

Chang, P.Y., Damodaran, P. and Melouk, S., Minimizing makespan on parallel batch processing machines, International Journal of Production Research, Vol. 42, No. 19 (2004), pp. 4211-4220.

Chung, S.H., Tai, Y.T. and Pearm, W.L., Minimising makespan on parallel batch processing machines with non-identical ready time and arbitrary job sizes, International Journal of Production Research, Vol. 47, No. 18 (2009), pp. 5109-5128.

Delage, E. and Ye, Y., Distributionally robust optimization under moment uncertainty with application to data-driven problems, Operations Research, Vol. 58, No. 3 (2010), pp. 595-612.

Fabozzi, F.J., Huang, D. and Zhou, G., Robust portfolios: contributions from operations research and finance, Annals of Operations Research, Vol. 176, No. 1 (2010), pp. 191-220.

Hu, X., Bao, J.S. and Jin, Y., Minimising makespan on parallel machines with precedence constraints and machine eligibility restrictions, International Journal of Production Research, Vol. 48, No. 6 (2010), pp. 1639-1651.

Iwamura, K., Yoshioka, T., Tanimizu, Y. and Sugimura, N., Development of scheduling system considering statistical distributions of processing times for human operators, Journal of Advanced Mechanical Design, Systems, and Manufacturing, Vol. 6, No. 5 (2012), pp. 600-610.

Lin, X., Janak, S.L. and Floudas, C.A., A new robust optimization approach for scheduling under uncertainty: I. Bounded uncertainty, Computers & Chemical Engineering, Vol. 28, No. 6 (2004), pp. 1069-1085.

Lu, C.C., Lin, S.W. and Ying, K.C. Robust scheduling on a single machine to minimize total flow time, Computers & Operations Research, Vol. 39, No. 7 (2012), pp. 1682-1691.

Mokotoff, E., An exact algorithm for the identical parallel machine scheduling problem, European Journal of Operational Research, Vol. 152, No. 3 (2004), pp. 758-769.
Özlen, M. and Webster, S. Minimising total flow-time on two parallel machines with planned downtimes and resumable jobs, International Journal of Production Research, Vol. 48, No. 1 (2010), pp. 201-226.
Skutella, M. and Uetz, M., Stochastic machine scheduling with precedence constraints, SIAM Journal on Computing Vol. 34, No. 4 (2005), pp. 788-802.
Soyster, A.L., Convex programming with set-inclusive constraints and applications to inexact linear programming, Operations research, Vol. 21, No. 5 (1973), pp.1154-1157.
Xu, X., Cui, W., Lin, J. and Qian, Y., Robust makespan minimisation in identical parallel machine scheduling problem with interval data, International Journal of Production Research, Vol. 51, No. 2 (2013), pp.3532-3548.
Yao, T., Mandala, S.R. and Chung, B.D., Evacuation transportation planning under uncertainty: a robust optimization approach, Networks and Spatial Economics, Vol. 9, No. 2 (2009), pp. 171-189.