A Selective Scheduling Problem with Sequence-dependent Setup Times: A Risk-averse Approach

Maria Elena Bruni[1], Sara Khodaparasti[2] and Patrizia Beraldi[1]
[1]*Department of Mechanical, Energy and Management Engineering, Unical, Italy*
[2]*Department of Mathematics and Computer Science, Unical, Italy*

Keywords: Machine Scheduling, Sequence-dependent Setup Time, Distributionally Robust Optimization, Conditional Value-at-Risk, Metaheuristic.

Abstract: This paper addresses a scheduling problem with parallel identical machines and sequence-dependent setup times in which the setup and the processing times are random parameters. The model aims at minimizing the total completion time while the total revenue gained by the processed jobs satisfies the manufacturer's threshold. To handle the uncertainty of random parameters, we adopt a risk-averse distributionally robust approach developed based on the Conditional Value-at-Risk measure hedging against the worst-case performance. The proposed model is tested via extensive experimental results performed on a set of benchmark instances. We also show the efficiency of the deterministic counterpart of our model, in comparison with the state-of-the-art model proposed for a similar problem in a deterministic context.

1 INTRODUCTION

Machine scheduling (MS) is categorized as an outstanding combinatorial problem whose applications are far beyond the traditional problems arising in the manufacturing sector and includes other fields such as healthcare, logistics, computer science, and communications (Blazewicz et al., 1991; Chang et al., 2019). A typical parallel machine scheduling problem deals with the sequencing and scheduling of a set of jobs to be processed on a set of identical machines where each job is processed by only one machine and each machine can process only one job at a time. A processing time is assigned to each job. Switching from one processed job to the next one in the schedule might require setup times, which, generally, are sequence-dependent. The aim is, in general, to find a feasible scheduling decision optimizing a time-related performance criterion such as the total completion time. In classical scheduling contexts, it is explicitly assumed that all existing jobs have to be processed. In the last years, the MS literature focused on a different aspect, considering the contributions of researchers and practitioners who recognized that there is a trade-off between the revenue gained by processing a job and the increase in the total time completion and that, in practice, it is impossible to accept all customer orders (jobs) due to the firm limitations in terms of resource and/or time. This led to a new class of problems considering the order acceptance or rejection, called *order acceptance and scheduling problems*. (Oguz et al., 2010). Most contributions in the MS literature focus on deterministic models in which the uncertainty is completely ignored or, in the best case, is handled by replacing the random parameter with an estimated value, such as the mathematical expectation, as followed in the risk-neutral approaches. On the contrary, in practice, the time-dependent performance measures, such as the total completion times, are affected by the fluctuations in the processing times and/or the sequence-dependent setup times due to unexpected events such as the unavailability of raw materials and tools, machine failure, the variations in the worker skills or the changes in the working environment (Suwa and Sandoh, 2012; Bougeret et al., 2018). Clearly, the optimal scheduling decisions obtained from the deterministic models might be near-optimal or even not valid for real-world applications with high uncertainty. This motivated us to consider both the setup times and the processing times uncertain. Since very often it is not possible to have complete information on the distribution functions of these stochastic parameters, we assume that only the first and the second moments of the probability distributions are known (Chang et al., 2019). We apply a risk-averse approach based on the idea of robust CVaR (RCVaR) to hedge against uncertainty. Even in the deterministic context, the selective structure makes the scheduling decisions challenging. If the decision-maker adopts a risk-averse

Bruni, M., Khodaparasti, S. and Beraldi, P.
A Selective Scheduling Problem with Sequence-dependent Setup Times: A Risk-averse Approach.
DOI: 10.5220/0007578001950201
In *Proceedings of the 8th International Conference on Operations Research and Enterprise Systems (ICORES 2019)*, pages 195-201
ISBN: 978-989-758-352-0

approach, the trade-off between the job profits, the expected processing and setup times, and their variations make the problem even more involved and less tractable. Differently from the existing literature which is more manufacturer-oriented and often maximizes the total revenue of the processed jobs, in the present paper, we introduce a new selective scheduling problem with profits and sequence-dependent setup times in the multi-machine environment with the objective of minimizing the sum of the total completion times. The model prioritizes the customers satisfaction by minimizing a time-related performance measure and accounts for the manufacturer's interests by enforcing the total revenue to be above a minimum threshold. This study contributes to the machine scheduling context since, to the best of our knowledge, this is the first multi-machine scheduling model with a selective structure in which the uncertainty of both the setup and processing times are taken into account. The new risk-averse model is very efficient and allows us to solve reasonable sized problem to optimality. The rest of the paper is organized as follows. Section 2 presents a review on the most related research for machine scheduling problems under uncertainty. In Section 2, the problem statement as well as the proposed mathematical formulation are provided. In Section 3, we present the extensive computational results on a set of benchmark problems.

2 LITERATURE REVIEW

There are only a few contributions on scheduling problems under uncertainty (Bruni et al., 2017; Bruni et al., 2018a; Bruni et al., 2018b), and in particular on selective problems in which the profits and/or the processing times are usually treated as random parameters (Emami et al., 2017; Xu et al., 2015). Emami et al. (Emami et al., 2017) presented a robust approach for the order acceptance scheduling problem with non-identical parallel machines, regarded as an extension of the model presented in (Oguz et al., 2010) to the multi-machine case, in which the job profits and the processing times are random parameters specified by interval data. They developed an equivalent deterministic MIP model based on the work of Bertsimas and Sim (Bertsimas and Sim, 2004) that was solved using a Benders decomposition method enhanced by valid cuts as well as a heuristic. In (Xu et al., 2015), Xu et al. addressed the dynamic order acceptance scheduling problem with sequence-dependent setup times and uncertain demands where the arrival of orders follows a Poisson distribution and the size and the type of the product required by each order is not known in advance. The objective is to maximize the expected profits of accepted orders respecting the customer's due date limitations. The authors develop an order acceptance rule approach based on the system capacity and opportunity cost where a heuristic is applied to estimate the capacity of system at the current state. Almost all the aforementioned contributions on selective scheduling problem share the same objective of maximizing the total revenue gained, which is usually a decreasing function of tardiness while the deadline and the due date constraints are met. In (Atakan et al., 2017), Atakan et al. proposed a risk-averse approach to tackle the uncertainty of processing times in a single-machine scheduling problem where a probabilistic constraint is imposed on traditional performance measures such as the total weighted completion time or the total weighted tardiness. The model is solved using a scenario decomposition approach providing promising results. Chang et al. (Chang et al., 2017) adopted the distributionally robust approach to sequence a set of jobs with uncertain processing times on a single machine where no information about the distribution function of the random parameters is available, except their moments. The model seeks to minimize the worst-case Conditional Value-at-Risk (Robust CVaR) assigned to the total completion times. They formulate the problem as an integer second-order cone programming (I-SOCP) which is solved using some approximation algorithms. In (Niu et al., 2019), Niu et al. proposed a robust distributionally approach to tackle the uncertainty of processing times in the single-machine scheduling context where the objective is expressed as the minimization of the expected worst-case total tardiness. This contribution is enhanced by a branch-and-bound algorithm as well as a beam search heuristic to solve large size instances. In (Pereira, 2016), Pereira presented a robust approach for a single-machine scheduling problem with uncertain processing times categorized by closed intervals to minimize the worst case total weighted completion time. They proposed an exact branch-and-bound methodology to solve the model. Following the risk-averse framework, Sarin et al. presented a conditional value-at-risk (CVaR) approach to tackle the uncertainty of processing times in a parallel multi-machine scheduling problem with the objective function of the total weighted tardiness (Sarin et al., 2014). They developed an integer L-shaped algorithm as well as a dynamic programming-based heuristic approach. Xu et al. proposed a min-max regret robust approach to deal with the uncertainty of the processing times for an identical parallel machine scheduling where the uncertain parameters are categorized as closed intervals and the objective function is expressed in terms of the makespan (the completion time of the last processed job) (Xu et al., 2013). As the solution approach, they designed some exact and heuristic algorithms.

3 MATHEMATICAL FORMULATION

Let $G = (V,E)$ be a directed graph defined over the set of potential jobs in $V = \{0,1,\ldots,i,\ldots,j,\ldots,,n\}$, to be processed on k identical machines, and the set of arcs $E = \{(i,j) \in V \times \bar{V} | i \neq j\}$ ($\bar{V} = V - \{0\}$) represents the precedence relationships among the jobs. To each arc $(i,j) \in E$, we assign a setup time s_{ij} which is equivalent to the minimum required time to setup the machine after processing job i and before starting job j. Each potential job i requires a processing time p_i and benefits a profit value ψ_i. Note that $0 \in V$ is a dummy node representing the start of action over each machine, obviously, $p_i = \psi_i = 0$. We need to select a sub set of jobs in $\bar{V}$ such that the total benefit gained by the chosen jobs, in percentage, is above a minimum required level Γ determined by the manufacturer and the total completion time, including the setup times and the processing times is minimized. To complete the notation, we define a level set $L = \{1,\ldots,r,\ldots,N\}$, ($N = |\bar{V}| - k + 1$) where each level r denotes the order of processing a job over each machine. Since all machines are required to be used, each machine can process up to N jobs. Considering the above discussion, we define two binary decisions variables as follows. Variable x_i^r takes value 1 *iff* job i is processed at level r and otherwise is set to 0; variable y_{ij}^r takes value 1 *iff* job j is processed at level r after job i (which was processed at level $r+1$) and otherwise is set to 0.

$$\min \sum_{r=1}^{N} \sum_{j\in\bar{V}} r\,(s_{0j} + p_j)\,y_{0j}^r + \sum_{r=1}^{N-1} \sum_{i\in\bar{V}} \sum_{\substack{j\in\bar{V}\\ j\neq i}} r\,(s_{ij} + p_j)\,y_{ij}^r \quad (1)$$

$$\sum_{r=1}^{N} x_i^r \leq 1 \quad i \in \bar{V} \quad (2)$$

$$\sum_{r=1}^{N} \sum_{j\in\bar{V}} y_{0j}^r = k \quad (3)$$

$$\sum_{i\in\bar{V}} x_i^1 = k \quad (4)$$

$$\sum_{\substack{j\in\bar{V}\\ j\neq i}} y_{ij}^r = x_i^{r+1} \quad i \in \bar{V},\ r = 1,2,\ldots,N-1 \quad (5)$$

$$y_{0j}^r + \sum_{\substack{i\in\bar{V}\\ i\neq j}} y_{ij}^r = x_j^r \quad j \in \bar{V},\ r = 1,2,\ldots,N-1 \quad (6)$$

$$y_{0j}^N = x_j^N \quad j \in \bar{V} \quad (7)$$

$$\sum_{i\in\bar{V}} \sum_{r=1}^{N} \psi_i x_i^r \geq \Gamma \sum_{i\in\bar{V}} \psi_i \quad (8)$$

$$x_i^r \in \{0,1\} \quad i \in \bar{V},\ r = 1,2,\ldots,N \quad (9)$$

$$y_{ij}^r \geq 0 \quad i \in V,\ j \in \bar{V},\ r = 1,2,\ldots,N \quad (10)$$

The objective function (1) minimizes the total completion times for all the processed jobs, including the setup and the processing times. The set of constraints (2) ensures that each each job is processed at most once. Constraints (3) and (4) require the process of exactly k jobs at the start and at the end of process, respectively. The set of constraints in (5)-(7) are related to the connectivity of the problem. Constraints (5) require that any job i processed at the upper level $r+1$ should be followed by exactly one task (let say j) by traversing edge (i,j) at the lower level r. The set of constraints in (6) impose that any job j processed at level r should be linked to exactly one recently processed task (let say i) by traversing edge (i,j) or linked directly to the dummy job by traversing edge $(0,j)$ at the same level. Constraints (7) guarantee that the first processed job, at the highest level N, should be the processed just after the dummy task by traversing edge $(0,j)$. Constraint (8) states that the total profit gained from processing the selected jobs should be above a predefined threshold. Constraints (9)-(10) define the binary nature of variables.

3.1 Distributional Robust Formulation

In this Section, we first present some preliminaries on robust optimization (RO). The RO models do not require specification of the exact distribution of the exogenous uncertainties of the model. This is the general distinction between the approaches of robust optimization and stochastic programming toward modeling problems with uncertainties. In the framework of robust optimization, uncertainties are usually modeled as random variables with true distributions that are unknown to the modeler, but are constrained to lie within a known support. The uncertainty set can be selected as a continuous interval or a finite set of different values. In this latter setting, the problem of interest is in general the optimization of the performance in the worst case scenario. Three criteria have been introduced in the literature: absolute robustness, robust deviation and relative robust deviation. Absolute robustness considers minimizing the objective value of the worst case directly. Robust deviation (or absolute regret) minimizes the largest possible difference between the observed objective value and the optimal one, while relative robust deviation (or relative regret) deals with the ratio of the largest possible observed value to the optimal value. In the continuous case, the uncertainty sets are selected as continuous intervals. Under this assumption, the expected solution performance is typically optimized. However, this criterion assumes that the decision maker is risk-neutral and leads to solutions that may be questionable. In this case, the decision maker attitude towards a risk should be taken into account. A criterion called Conditional Value-at-Risk (CVaR), early applied to a stochastic portfolio selection problem, can be used. Using this criterion, the decision maker provides a parameter α which reflects his attitude towards a risk. When $\alpha = 0$, then CVaR becomes the expectation but for greater values, more attention is paid to the worst outcomes, which fits into the robust optimization framework. In this Section, we consider the worst-case CVaR in situation where the information on the underlying probability distribution is not exactly known. In fact, typically, the first- and second-order moments of the uncertain parameters may be known, but it is unlikely to have complete information about their distributions.

Let assume that the uncertain setup times $\tilde{s_{ij}}$ and processing times $\tilde{p_j}$ are defined by random vectors s and p, respectively, We investigate a specific case where the ambiguity set is determined by the mean and covariance and the distributional set is a semi-infinite support set. Let $\mathbb{P}_\mathbf{s}$ and $\mathbb{P}_p$ be the ambiguous distribution of random vectors **s** and **p**, respectively, which are described by their first and second moments as follows:

$$\mathbb{P}_s = \{\mathbb{P}^s | Sup(\tilde{s_{ij}}) = [0,\infty), \forall (i,j) \in V \times \bar{V}, \quad E(\tilde{s_{ij}}) = \mu_{\tilde{s_{ij}}}, Var(\tilde{s_{ij}}) = \sigma^2_{\tilde{s_{ij}}}\} \tag{11}$$

$$\mathbb{P}_p = \{\mathbb{P}^p | Sup(\tilde{p}_i) = [0,\infty), \forall i \in \bar{V}, \quad E(\tilde{p}_i) = \mu_{\tilde{p}_i}, Var(\tilde{p}_i) = \sigma^2_{\tilde{p}_i}\} \tag{12}$$

Following the risk-averse approach, we apply the *CVaR* risk measure at a given confidence level $\alpha \in (0,1)$, denoted by $CVaR_\alpha$. This risk measure quantifies the expected loss of the random variable T in the worst α% of cases described as follows:

$$CVaR_\alpha = E[T | T \geq inf\{t | P(T > t) \leq 1-\alpha\}] \tag{13}$$

where T is a vector of random variables like s or p in (11) or (12).

Therefore, considering the above definitions, we define the robust risk measure $CVaR_\alpha(T)$, denoted by, $RCVaR_\alpha(T)$, as follows:

$$RCVaR_\alpha(T) = sup_{\mathbb{P}^T \in \mathbb{P}^T} CVaR_\alpha(T) \tag{14}$$

It is easy to see that the worst-case $CVaR_\alpha(T)$ is nothing but the robust $RCVaR_\alpha(T)$ which can be equivalently expressed as

$$\min_{(x,y)\in X} RCVaR_\alpha(T) = \min_{(x,y)\in X} sup_{\mathbb{P}^T \in \mathbb{P}^T} CVaR_\alpha(T) \tag{15}$$

where X is the solution space describing the set of constraints in (2)-(10).

As proposed in (Chang et al., 2017), it can be proved that the solution of the robust selective parallel scheduling model (1)- (10) can be found by applying the following Theorem.

Theorem 1. *For any random variable $T \in \mathbb{R}^+$, with a distribution function $\mathbb{P}^T$ belonging to the distributional set $\mathbb{P}_T = \{\mathbb{P}^T | Sup(\tilde{T}) = [0,\infty), E(T) = \mu_T, Var(T) = \sigma^2_T\}$, the $RCVaR_\alpha(T)$ is calculated as follows:*

$$\begin{cases} \frac{\mu_T}{1-\alpha}, & if\, 0 \leq \alpha \leq \frac{\sigma_T^2}{\sigma_T^2+\mu_T^2} \\ \mu_T + \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{\sigma_T^2}, & if\, \frac{\sigma_T^2}{\sigma_T^2+\mu_T^2} \leq \alpha \leq 1 \end{cases}$$

Proof: See (Chang et al., 2017). □

Theorem 1 provides a baseline to present an equivalent mixed integer mathematical model for the robust model that we are going to present. In fact, for any feasible solution $(x,y) \in X$ described by the set of constraints in (2)-(10) with the distributional loss function Z_t, Z_s subject to a distribution in $\mathbb{P}^z = \{\mathbb{P}^z | Sup(Z_{(x,y)}) = [0,\infty), E(Z_{(x,y)}) = \mu_z(x,y), Var(Z_{(x,y)}) = \sigma^2_z(x,y)\}$, the following result holds (Chang et al., 2017).

$$\min_{(x,y)} RCVaR^z_\alpha(Z_{(x,y)}) = \min_{(x,y)\in X}(z_1^*, z_2^*), \tag{16}$$

where z_1^* is the optimal objective function value of the following integer linear problem

$$\min \tfrac{1}{1-\alpha}(\textstyle\sum_{r=1}^N \sum_{j\in\bar{V}} r[\mu(\tilde{s_{0j}}) + \mu(\tilde{p}_j)]\, y^r_{0j} + \\ + \textstyle\sum_{r=1}^N \sum_{i\in\bar{V}} \sum_{j\in\bar{V}, j\neq i} r[\mu(\tilde{s_{ij}}) + \mu(\tilde{p}_j)]\, y^r_{ij}) \\ s.t. (2)-(10) \tag{17}$$

and z_2^* is the optimal objective function value of the following nonlinear integer problem

$$\min(\sum_{r=1}^N \sum_{j\in\bar{V}} r[\mu(\tilde{s_{0j}}) + \mu(\tilde{p}_j)]\, y^r_{0j} + \\ + \sum_{r=1}^N \sum_{i\in\bar{V}} \sum_{\substack{j\in\bar{V}\\ j\neq i}} r[\mu(\tilde{s_{ij}}) + \mu(\tilde{p}_j)]\, y^r_{is}) + \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{b} \tag{18}$$

$s.t.\;\; (2)-(10)$

where

$$b = \sum_{r=1}^N \sum_{j\in\bar{V}} r^2[\sigma^2(\tilde{s_{0j}}) + \sigma^2(\tilde{p}_j)]\, y^r_0 + \\ + \sum_{r=1}^N \sum_{i\in\bar{V}} \sum_{j\in\bar{V}\, j\neq i} r^2[\sigma^2(\tilde{s_{ij}}) + \sigma^2(\tilde{p}_j)]\, y^r_{is} \tag{19}$$

hence, the optimal solution of the distributionally robust model is obtained by solving one linear and one non-linear mixed integer mathematical model. Then, we should take the minimum between the optimal values of z_1^* and z_2^*.

4 COMPUTATIONAL EXPERIMENTS

To test the model, we have chosen some instances from the benchmark on order acceptance in single scheduling context (Oguz et al., 2010). The setup times, processing times, and the profits are the same as those reported in the benchmark and the setup time variance $\sigma^2(\tilde{s_{ij}})$ as well as the processing time variances $\sigma^2(\tilde{s_i})$ were set to $\sigma^2(\tilde{t_{ij}}) = \lceil \zeta_t^2 \rceil$ and $\sigma^2(\tilde{s_i}) = \lceil \zeta_s^2 \rceil$ where ζ_t and ζ_s are random numbers uniformly distributed in intervals $[1, \frac{1}{2}(\min_{i \in V, j \in \bar{V}} \mu(\tilde{t_{ij}}) + \max_{i \in V, j \in \bar{V}} \mu(\tilde{t_{ij}}))]$ and $[1, \frac{1}{2}(\min_{i \in \bar{V}} \mu(\tilde{s_i}) + \max_{i \in \bar{V}} \mu(\tilde{s_i}))]$, respectively. The value of Γ is set to 0.6 and the number of machines (k) varies from one to six depending on the size of instance. The proposed model was implemented in C++. The experiments were executed on an Intel® Core™ i7 2.90 GHz, with 8.0 GB of RAM memory.

To show the efficiency of the proposed model, we perform a set of experiments on 20 instances selected from the benchmark and compare the deterministic counterpart of our model with the order acceptance scheduling model presented in (Oguz et al., 2010), which shares the same idea of job acceptance or rejection with our model. For more information on the deterministic order acceptance scheduling problems, see for example (Geramipour et al., 2017; Nguyen, 2016).

To make the comparison in a fair way, we replace those constraints related to the tardiness and deadlines in the order acceptance model with the service level constraint (8) and set the objective function equal to the total completion time. Since the order acceptance model in (Oguz et al., 2010) is, indeed, designed for the single machine case, we set $k = 1$ in our proposed model. Both models were solved by CPLEX considering a time limit of 1000 seconds. Table 1 summarizes the obtained results where Column 1 and 2 represent the instance name and the number of jobs, respectively; Columns 3 and 4 exhibit, the best objective value and the computational time (in seconds) for the proposed model followed by its relative gap (in percentage) with respect to the CPLEX linear relaxation lower bound. In a similar way, Columns 6-8 present the same information exhibited in Columns 3-5 for the other model. Column 9 shows the optimality gap for the order acceptance model and, finally, Column 10 indicates the speed up in solution time calculated as $\Delta = \frac{CPU_{\text{Proposed Model}}}{CPU_{\text{Order acceptance Model}}} \times 100$. In terms of the solution quality, the proposed model was able to find the optimal solution, verified by the zero gap values in Column 5, in a solution time limited to 132 seconds.

Table 1: Comparing the results of two models.

Instance	#Jobs	Proposed Model			Traditional Model				
		Obj Val.	CPU(s)	Gap_{LB}(%)	Obj Val.	CPU(s)	Gap_{LB}(%)	Gap_{Opt}(%)	Δ(%)
10Tao-R1-1	10	131	0.15	0	131	4.1	0	0	3.66
10Tao-R3-1	10	150	0.05	0	150	9.36	0	0	0.53
10Tao-R5-1	10	106	0.05	0	106	1.46	0	0	3.42
10Tao-R7-1	10	155	0.09	0	155	5.94	0	0	1.52
10Tao-R9-1	10	192	0.09	0	192	11.85	0	0	0.76
15Tao-R1-1	15	175	0.1	0	175	1000	29.7	0	0.01
15Tao-R3-1	15	287	0.4	0	287	1000	44.9	0	0.04
15Tao-R5-1	15	311	0.24	0	311	1000	46.3	0	0.02
15Tao-R7-1	15	244	0.23	0	282	1000	74	13.48	0.01
15Tao-R9-1	15	269	0.13	0	269	1000	40	0	0.01
25Tao-R1-1	25	646	2.24	0	648	1000	79.3	0.31	0.21
25Tao-R3-1	25	566	1.56	0	569	1000	77.9	0.53	0.16
25Tao-R5-1	25	772	2.24	0	843	1000	82.2	8.42	0.22
25Tao-R7-1	25	555	1.43	0	557	1000	78.9	0.36	0.14
25Tao-R9-1	25	620	2.43	0	649	1000	77.2	4.47	0.24
50Tao-R1-1	50	1501	43.7	0	1720	1000	91.2	12.73	4.34
50Tao-R3-1	50	1977	24.24	0	-	1000	∞	∞	2.33
50Tao-R5-1	50	2300	131.24	0	-	1000	∞	∞	13.12
50Tao-R7-1	50	1854	23.01	0	-	1000	∞	∞	2.3
50Tao-R9-1	50	2216	49.39	0	-	1000	∞	∞	4.94
Avg			14.15						1.9

For the order acceptance model, CPLEX found the optimal solutions only in 9 instances, including 4 cases for which the optimality did not proved (verified by the non-zero gaps in Column 8). In 7 cases, CPLEX provided only near-optimal solutions with different optimality gap varying from 0.31% to 12.73% and for the 4 last instances with the largest size, CPLEX did not find any feasible solution (specified by ”-”). Also, with respect to the solution time, the time limit for all cases but those with 10 jobs was reached. Apart from that, the considerable difference between Gap_{Opt} and Gap_{LB} in Columns 8 and 9 is an informative insight showing that the lower bound resulted from the linear relaxation of the order acceptance model is not tight at all even for moderate instances with 25 jobs. In summary, the proposed model outperforms the other model in terms of both the solution quality and the computational time. In terms of the solution quality, the proposed model was able to find the optimal solution, verified by the zero gap values in Column 5, in a solution time limited to 132 seconds. Regarding the order acceptance model, CPLEX was able to find the optimal solutions only in 9 cases including 5 cases for which the optimality was not verified. For 7 cases, CPLEX provided only near-optimal solutions with different optimality gaps varying from 0.31% to 12.73% and for the 4 last instances with the largest size, CPLEX did not find any feasible solution (those specified by ”-”). Also, with respect to the solution time, only for the smallest instances with size 10 the time limit did not reached. We should mention that the considerable difference between Gap_{Opt} and Gap_{LB} is an informative insight showing the weak performance of lower bounds provided by the linear relaxation of the order acceptance model even for moderate instances with 25 jobs. In summary, the proposed model outperforms the order

Table 2: CPU time for 10 jobs, one machine.

	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$
Instance	Avg. CPU(s)	Avg. CPU(s)	Avg. CPU(s)
Tao1R1	7.75	7.52	7.98
Tao1R3	8.33	8.13	9.01
Tao1R5	8.82	8.18	9.05
Tao1R7	8.15	7.91	8.76
Tao1R9	8.54	8.56	9.33
Tao3R1	8.52	7.51	8.04
Tao3R3	8.43	7.7	7.98
Tao3R5	8.97	8.55	9.4
Tao3R7	8.55	7.86	8.38
Tao3R9	8.98	8.17	8.93
Tao5R1	7.79	7.39	8.29
Tao5R3	8.16	8.74	9.52
Tao5R5	8.61	8.52	8.59
Tao5R7	8.07	7.61	8.6
Tao5R9	7.16	7.47	8.01
Tao7R1	8.18	8.32	9.53
Tao7R3	7.64	7.12	7.5
Tao7R5	8.04	7.55	8.52
Tao7R7	8.09	8.11	8.9
Tao7R9	8.19	8.26	9.2
Tao9R1	8.65	7.96	8.75
Tao9R3	8.39	6.96	7.8
Tao9R5	9.05	8.6	9.28
Tao9R7	9.75	9.09	9.89
Tao9R9	8.77	8.5	8.9
Avg.	8.38	8.01	8.73

Table 3: CPU time for 15 jobs, one machine.

	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$
Instance	Avg. CPU(s)	Avg. CPU(s)	Avg. CPU(s)
Tao1R1	36.69	38.6	40.06
Tao1R3	37.17	39.66	42.05
Tao1R5	39.36	40.99	41.71
Tao1R7	39.76	46.09	43.54
Tao1R9	38.66	40.31	41.53
Tao3R1	37.79	40.41	40.29
Tao3R3	38.23	39.38	43.55
Tao3R5	39.94	39.82	43.27
Tao3R7	41.68	46.59	45.33
Tao3R9	36	40.66	40.62
Tao5R1	39.2	42.34	43.64
Tao5R3	40.57	40.18	39.94
Tao5R5	37.67	41.1	43.3
Tao5R7	38.64	41.65	43.48
Tao5R9	34.46	37.74	38.01
Tao7R1	39.28	40.21	41.84
Tao7R3	35.88	39.11	41.88
Tao7R5	41.32	41.91	45.64
Tao7R7	35.99	36.73	39.23
Tao7R9	39.61	41.68	43.52
Tao9R1	38.93	42.09	43.04
Tao9R3	34.23	35.46	36.17
Tao9R5	36.87	39.39	42.31
Tao9R7	39.67	42.86	43.41
Tao9R9	39.18	41.33	41.53
Avg.	38.27	40.65	41.96

Table 4: CPU time for 25 jobs, one machine.

	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$
Instance	Avg. CPU(s)	Avg. CPU(s)	Avg. CPU(s)
Tao1R1	295.95	380.85	696.31
Tao1R3	284.4	330.59	538.97
Tao1R5	383.52	449.52	621.13
Tao1R7	292.87	395.81	622.81
Tao1R9	329.85	492.69	948.22
Tao3R1	303.51	457.73	779.48
Tao3R3	324.32	367.85	630.11
Tao3R5	388.36	452.19	558.76
Tao3R7	395.02	411.51	714.42
Tao3R9	349.44	440.82	748.21
Tao5R1	340.54	424.23	810.03
Tao5R3	296.48	323.62	499.65
Tao5R5	272.08	350.51	614.55
Tao5R7	288.31	318.23	626.53
Tao5R9	344.98	377.1	775.88
Tao7R1	312.19	342.35	572.17
Tao7R3	292.1	339.05	518.48
Tao7R5	270.79	335.57	587.55
Tao7R7	275.4	354.04	454.84
Tao7R9	287.97	345.61	563.71
Tao9R1	229.7	370.06	417.83
Tao9R3	259.06	340.6	529.14
Tao9R5	280.73	345	455.18
Tao9R7	233.08	281.73	565.33
Tao9R9	259.3	292.25	430.72
Avg.	303.6	372.78	611.2

acceptance model in terms of both the solution quality and the computational time.

The second set of the experiments was devoted to the assessment of the computational tractability of the proposed distributionally robust model. To this aim, the non-linear MIP model was solved using the open source SCIP library, released 3.2.0.

Tables 2 and 3 show the average CPU time in seconds, provided by SCIP for each class of instances. For all the instances with 10 and 15 nodes and one machine, SCIP was able to find the optimal solution within short computational times. The CPU time does not show a significant variation with the increase of α. For the biggest instances with 25 nodes (Tables 4 and 5 for one and two machines, respectively), with the increase of α from 0.5 to 0.9, the CPU time drastically increases (64% and 104%).

5 CONCLUSION

In this paper, we introduced the selective job scheduling problem with sequence dependent setup times in a multi-machine context where the setup times as well as the job processing times are uncertain. The problem seeks the minimization of the total completion time such that a minimum service level in terms of the profits of the selected jobs is met. We adopted a distributionally robust approach and formulated the problem as a non-linear MIP risk-averse model. We tested the efficiency of the proposed model on a large

Table 5: CPU time for 25 jobs, two machines.

	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$
Instance	Avg. CPU(s)	Avg. CPU(s)	Avg. CPU(s)
Tao1R1	130.69	89.78	122.66
Tao1R3	120.01	146.64	148.64
Tao1R5	94.19	123.81	259.42
Tao1R7	114.16	174.75	568.06
Tao1R9	134.41	172.25	461.29
Tao3R1	112.58	160.54	358.55
Tao3R3	97.51	142.8	492.49
Tao3R5	141.19	163.15	330.12
Tao3R7	117.34	160.33	189.83
Tao3R9	100.38	68.21	78.84
Tao5R1	89.53	88.82	281.69
Tao5R3	93.56	86.06	259.5
Tao5R5	70.34	112.47	158.87
Tao5R7	89.2	89.77	188.48
Tao5R9	125.8	115.44	440.91
Tao7R1	73.28	91.76	159.64
Tao7R3	80.9	85.39	322.09
Tao7R5	116.16	123.9	243.05
Tao7R7	110.08	193.51	322.81
Tao7R9	88.09	158.97	281.59
Tao9R1	101.87	138.27	255.47
Tao9R3	147.58	185.53	255.07
Tao9R5	112.69	88.8	74.32
Tao9R7	100.9	135.96	279.13
Tao9R9	77.43	148.42	84.86
Avg.	105.59	129.81	264.7

set of scheduling benchmark instances providing the optimal solution within short computational time for the set of small and moderate sized instances. For the biggest instances the computational effort increases, calling for the development of a tailored heuristic approach, that could be a promising avenue for future research.

REFERENCES

Atakan, S., Bülbül, K., and Noyan, N. (2017). Minimizing value-at-risk in single-machine scheduling. *Annals of Operations Research*, 248(1-2):25–73.

Bertsimas, D. and Sim, M. (2004). The price of robustness. *Operations research*, 52(1):35–53.

Blazewicz, J., Dror, M., and Weglarz, J. (1991). Mathematical programming formulations for machine scheduling: A survey. *European Journal of Operational Research*, 51(3):283–300.

Bougeret, M., Pessoa, A. A., and Poss, M. (2018). Robust scheduling with budgeted uncertainty. *Discrete Applied Mathematics*.

Bruni, M.E., Di Puglia Pugliese, L., Beraldi, P., and Guerriero, F. (2018a). A two-stage stochastic programming model for the resource constrained project scheduling problem under uncertainty. pages 194–200. SciTePress.

Bruni, M.E., Pugliese, L. D. P., Beraldi, P., and Guerriero, F. (2017). An adjustable robust optimization model for the resource-constrained project scheduling problem with uncertain activity durations. *Omega*, 71:66 – 84.

Bruni, M.E., Pugliese, L. D. P., Beraldi, P., and Guerriero, F. (2018b). A computational study of exact approaches for the adjustable robust resource-constrained project scheduling problem. *Computers and Operations Research*, 99:178 – 190.

Chang, Z., Ding, J.-Y., and Song, S. (2019). Distributionally robust scheduling on parallel machines under moment uncertainty. *European Journal of Operational Research*, 272(3):832–846.

Chang, Z., Song, S., Zhang, Y., Ding, J.-Y., Zhang, R., and Chiong, R. (2017). Distributionally robust single machine scheduling with risk aversion. *European Journal of Operational Research*, 256(1):261–274.

Emami, S., Moslehi, G., and Sabbagh, M. (2017). A benders decomposition approach for order acceptance and scheduling problem: a robust optimization approach. *Computational and Applied Mathematics*, 36(4):1471–1515.

Geramipour, S., Moslehi, G., and Reisi-Nafchi, M. (2017). Maximizing the profit in customers order acceptance and scheduling problem with weighted tardiness penalty. *Journal of the Operational Research Society*, 68(1):89–101.

Nguyen, S. (2016). A learning and optimizing system for order acceptance and scheduling. *The International Journal of Advanced Manufacturing Technology*, 86(5-8):2021–2036.

Niu, S., Song, S., Ding, J.-Y., Zhang, Y., and Chiong, R. (2019). Distributionally robust single machine scheduling with the total tardiness criterion. *Computers & Operations Research*, 101:13–28.

Oguz, C., Salman, F. S., Yalçın, Z. B., et al. (2010). Order acceptance and scheduling decisions in make-to-order systems. *International Journal of Production Economics*, 125(1):200–211.

Pereira, J. (2016). The robust (minmax regret) single machine scheduling with interval processing times and total weighted completion time objective. *Computers & Operations Research*, 66:141–152.

Sarin, S. C., Sherali, H. D., and Liao, L. (2014). Minimizing conditional-value-at-risk for stochastic scheduling problems. *Journal of Scheduling*, 17(1):5–15.

Suwa, H. and Sandoh, H. (2012). *Online scheduling in manufacturing: A cumulative delay approach*. Springer Science & Business Media.

Xu, L., Wang, Q., and Huang, S. (2015). Dynamic order acceptance and scheduling problem with sequence-dependent setup time. *International Journal of Production Research*, 53(19):5797–5808.

Xu, X., Cui, W., Lin, J., and Qian, Y. (2013). Robust makespan minimisation in identical parallel machine scheduling problem with interval data. *International Journal of Production Research*, 51(12):3532–3548.