Computers & Operations Research 167 (2024) 106642

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

Target-oriented robust satisficing models for the single machine scheduling problems with release time

Xun Zhang [a], Du Chen [b,*]

[a] International Institute of Finance, School of Management, University of Science and Technology of China, Hefei 230036, China
[b] Nanyang Business School, Nanyang Technological University, 52 Nanyang Avenue, 639798, Singapore

ARTICLE INFO

Keywords:
Robust machine scheduling
Random release time
Random processing time
Robust satisficing

ABSTRACT

In this paper, we investigate single machine scheduling problems with release times and random processing times, where the release times can be either deterministic or random. The objective is to determine a scheduling sequence that exhibits strong out-of-sample performance. To achieve this, we employ target-oriented robust satisficing models to obtain the scheduling sequence. For the scheduling problem with deterministic release times, we derive an equivalent mixed-integer linear program and an approximate reformulation to address cases involving a large number of jobs. For the scheduling problem with random release times, we first demonstrate the challenges associated with providing an equivalent reformulation. To overcome this intractability, we propose an approximate reformulation based on the linear decision rule. Numerical experiments are conducted to demonstrate the superiority of our solutions through comparisons with several benchmarks. Furthermore, the numerical results demonstrate that a relatively higher target should be chosen if the worst-case performance is valued, otherwise, the decision-maker should set a relatively lower target to obtain better average performance.

1. Introduction

Scheduling is one of the most critical problems in production, manufacturing, and transportation systems. To obtain a good performance measure, such as total completion time and makespan, the decision-maker needs to determine the optimal sequence of the jobs to be processed. Over the last decades, most of the existing literature studied scheduling problems with specific job parameters, such as deterministic processing time and release time. For a comprehensive and in-depth survey of this field, we refer to the books by Leung (2004) and Pinedo (2008).

Scheduling with uncertain parameters has become increasingly important in recent years. Various factors contribute to uncertainty in scheduling problems, such as processing time, which can be affected by machine or tool conditions, worker skill levels, and production environments (Yang and Yu, 2002; Xu et al., 2013). Additionally, uncertainty in release time may arise from raw material or component arrivals (Zheng et al., 2019). One common real-world case of a scheduling issue with uncertain parameters is the iron-making process. In this situation, molten iron is produced by a blast furnace, and then, the molten iron is sequentially preprocessed at a pretreatment station (Yue et al., 2018). The processing time in this scenario is typically brief, but the release time, which depends on the tapping times from the blast furnace, is uncertain.

There are two principal methods for modeling uncertainty in scheduling problems. In the literature, one approach involves treating uncertain parameters such as processing and release times as random variables. In this scenario, the distributions of these uncertain parameters are assumed to be known in advance (e.g., Pinedo, 1983; Liu et al., 2020a). A stochastic optimization model is subsequently used to solve the scheduling problem, based on specific probability distributions of job parameters. However, accurately inferring and estimating the true distribution from historical data is challenging, particularly with limited available data (Lu and Pei, 2022).

Robust optimization (RO) and distributionally robust optimization (DRO) are used to address scheduling problems when the distribution of random parameters is unknown. In robust scheduling problems, uncertain parameters are typically assumed to reside in an uncertainty set. However, the lack of distribution information may result in an overly conservative solution to the robust scheduling optimization problem. DRO can be considered a generalization of robust optimization. In the DRO model, the probability distribution of the random variables is assumed to lie in an ambiguity set, which is a set of all possible probability distributions. Although the ambiguity set is defined by certain moment information (Delage and Ye, 2010; Goh and Sim, 2011) at the early stage, there is increasing attention to probability-distance-based ambiguity sets. Specifically, given a reference distribution, the

* Corresponding author.
E-mail addresses: xunzhang2023@outlook.com (X. Zhang), chen1443@e.ntu.edu.sg (D. Chen).

https://doi.org/10.1016/j.cor.2024.106642
Received 2 June 2023; Received in revised form 19 January 2024; Accepted 1 April 2024
Available online 4 April 2024

ambiguity set contains all distributions whose distance from the reference distribution is less than a threshold. Note that the threshold is usually a hyper-parameter that needs to be determined through cross-validation.

As an alternative to the DRO model, Long et al. (2023) propose a target-oriented model for robust data-driven optimization called robust satisficing. In the robust satisficing model, the decision-maker specifies a target or an acceptable loss of optimality as a trade-off for the model's ability to withstand greater uncertainty. As documented by Simon (1955), target satisficing is prevalent in human decision-making, especially in complex situations involving risks and uncertainty. Recently, Chen and Tang (2022) conducted a survey of 74 executive MBA students, with more than 90% of the respondents reporting that their companies set targets in every accounting period. Applications based on the satisficing criteria include routing under travel time uncertainty (Zhang et al., 2021), and project selection with uncertain returns (Hall et al., 2015). In fact, the robust satisficing model conceptually shares the same spirit of globalized robust optimization (Liu et al., 2023). It is worth noting that compared to determining the threshold for distribution distance, it is more interpretable for the decision-maker to choose a target, and robust satisficing may perform well when data availability is limited for cross-validation.

In this paper, we adopt the robust satisficing model to study a single machine scheduling problem with release time and random processing time, where the release time can either be deterministic or uncertain. The aim of this problem is to provide a scheduling sequence that can perform well on both the average and worst-case total completion time in most cases.

Although there are an increasing number of papers in which scheduling problems with uncertain processing times are being investigated, most of them do not consider the existence of release time. One likely reason is the hardness of machine scheduling with release time. In fact, Lenstra et al. (1977) showed that single machine scheduling with deterministic release and processing time to minimize the total completion time is NP-hard. Therefore, investigating the single machine scheduling problem with deterministic/random release times and processing times is highly challenging.

To the best of our knowledge, the study by Lu and Pei (2022) was the first to explore distributionally robust scheduling problems that incorporate release times. However, there are several differences between Lu and Pei (2022) and this paper. First, Lu and Pei (2022) proposed a distributionally robust optimization approach that utilizes a moment-based ambiguity set, whereas our proposed robust satisficing model is closer to the distributionally robust optimization that employs a probability-distance-based ambiguity set. Moreover, the mean and mean absolute deviation in their moment-based ambiguity set are estimated from historical data, which may result in estimation errors, particularly when the historical data are limited. Our robust satisficing approach directly integrates historical data into the final scheduling optimization model, which prevents estimation errors. Second, while Lu and Pei (2022) focused solely on scheduling problems with uncertain processing times, we also investigated problems with uncertain release times. Finally, we present an approximate reformulation of the scheduling problem for both deterministic and uncertain release times; moreover, our numerical experiments demonstrate that our approach performs well.

The main contributions of this study are summarized as follows.

- To the best of our knowledge, we are the first to adopt the robust satisficing model to study single machine scheduling problems with release time.
- For the robust satisficing model with deterministic release time and random processing time, we derive an equivalent mixed-integer linear program.
- For the robust satisficing model with random release and processing time, we first show the difficulty of deriving an equivalent reformulation and then, we employ the linear decision rule to provide an approximate reformulation.
- Numerical experiments demonstrate that our robust satisficing model outperforms the benchmarks, leading to several managerial insights.

The remainder of this paper is organized as follows. After the introduction, in Section 2, we review the literature on scheduling problems with random processing and release time, as well as the literature about robust optimization and robust satisficing. In Section 3, we study the single machine scheduling problem with deterministic release time and random processing time. The single machine scheduling problem with both random release and processing time is investigated in Section 4. Numerical experiments are conducted in Section 5. We also highlight several implications in Section 6. The conclusions are summarized in Section 7. All proofs are given in the Appendix.

Notations. Throughout this paper, we use lowercase ***bold*** and uppercase ***bold*** characters to denote vectors and matrices, respectively. The corresponding normal characters denote componentwise elements. Variables with tilde symbols, such as $\tilde{\boldsymbol{r}}$, represent random variables. For any K-dimensional vector $\boldsymbol{v}$ such that $\boldsymbol{v} \succeq_\infty \boldsymbol{0}$, this implies that $\boldsymbol{v}$ resides within the ∞-norm cone, where $\max_{k=2,\ldots,K} |v_k| \le v_1$. Similarly, $\boldsymbol{v} \succeq_1 \boldsymbol{0}$ indicates that $\boldsymbol{v}$ belongs to the 1-norm cone, characterized by $\sum_{k=2,\ldots,K} |v_k| \le v_1$. $\mathbb{R}^n$ denotes an n-dimensional real space. $[n]$ represents the set $\{1,2,3,\ldots,n\}$ and $[\underline{u},\overline{u}]^n$ represents the n-dimensional real space with lower bound $\underline{u}$ and upper bound $\overline{u}$. We use $\mathscr{P}_0(\mathscr{Z})$ to represent the set of all probability distributions on $\mathscr{Z}$. We denote $\mathscr{R}^{N,M}$ as the space of all measurable functions from $\mathbb{R}^N$ to $\mathbb{R}^M$ that are bounded on compact sets.

2. Literature review

2.1. Distributionally robust optimization and robust satisficing model

Over the past two decades, DRO has emerged as a pivotal modeling framework for addressing uncertainty in decision-making problems (e.g., Wang et al., 2019; Fan and Xie, 2022). In contrast to robust optimization, in which it is assumed that the decision-maker knows the support set of the random parameters, DRO posits that the probability distribution of the random variables resides in a so-called ambiguity set. The DRO models optimize the performance measures based on the worst-case distribution within the ambiguity set.

In the early stages, research on DRO was predominantly focused on moment-based ambiguity sets, encompassing all probability distributions whose moments satisfy specific properties (e.g., El Ghaoui et al., 1998; Delage and Ye, 2010; Mak et al., 2015). Recently, researchers have shown a growing interest in DRO with distance-based ambiguity sets. These sets utilize the empirical distribution constructed from historical data and encompass distributions that closely align with the empirical distribution according to a specific distance measure. For further discussions and applications of the DRO model with the Wasserstein ambiguity set, interested readers may refer to Mohajerin Esfahani and Kuhn (2018), Zhao and Guan (2018) and Gao (2022).

Notably, in this paper, we focus on a two-stage optimization problem, and we observe that the DRO model can also be applied to investigate multistage decision problems. An investigation of robust multistage decision problems was initiated by Ben-Tal et al. (2004). Unfortunately, solving such problems is generally computationally intractable. To address this challenge, a practical and efficient approach involves employing a linear decision rule, where adjustable decisions are constrained to be linearly dependent on uncertain parameters. Linear decision rules have been widely utilized in the robust optimization literature (Chen et al., 2008; Kuhn et al., 2011; Perakis et al., 2018). In a departure from the traditional linear decision rule, Bertsimas et al. (2019) and Chen et al. (2020) extended it by incorporating an auxiliary random variable, highlighting better performance.

In this paper, we adopt a target-oriented robust satisficing model to solve our scheduling problem. In contrast to optimizing the objective under the worst-case distribution within the ambiguity set in DRO approach, Long et al. (2023) proposed a data-driven robust satisficing model that determines the best possible here-and-now decision to achieve a target expected reward under distribution ambiguity. Following this seminal work, Sim et al. (2021b) adopt robust satisficing to address robust supervised learning problems. Hu et al. (2022) showed that the robust satisficing model can outperform benchmarks in assortment problems. To address broader applications in operations management, Ramachandra et al. (2021) extended robust satisficing to conic optimization problems with recourse, while Sim et al. (2021a) incorporated prediction models and side information in the robust satisficing model. Instead of adopting the linear decision rules proposed in DRO and robust satisficing works, we are the first to derive an equivalent reformulation for robust satisficing scheduling problems. This highlights the advantages of robust satisficing in addressing scheduling problems with uncertain parameters.

2.2. Scheduling problems with uncertain processing time

To address scheduling problems with uncertain processing times, a common approach involves assuming that the distribution of processing times is known before making sequence decisions. For instance, Gourgand et al. (2003) investigated a stochastic flow shop scheduling problem where the processing time of each job follows an exponentially distributed distribution with a known rate. In a similar vein, Yue and Zhou (2021) assumed that the processing times follow a normal distribution. The objective of their study was to determine a due window for each job and a processing schedule for all jobs to minimize the total expected weighted costs, considering earliness, tardiness, and due-window assignment.

The exact distributions of the parameters are difficult to obtain, and robust optimization with support set information has been applied to solve scheduling problems with uncertainty. To the best of our knowledge, Daniels and Kouvelis (1995) was the first to study robust scheduling problems with uncertain processing times. They presented exact and heuristic solutions to minimize the expected total completion time. Subsequently, many researchers started to investigate robust scheduling problems under different assumptions and settings (see, for example Xu et al., 2013; Lu et al., 2012; Pereira, 2016; Feng et al., 2016) and Sarin et al. (2010).

However, in the papers mentioned above, the moment information in the processing time is not considered, which may present overly conservative solutions. To address this issue, DRO has been applied to solve scheduling problems. Chang et al. (2017) were the first to introduce DRO into production scheduling, where only the mean and variance of processing time were assumed to be available. Their objective was the worst-case conditional value-at-risk of the total flow time. They reformulated their problem into a second-order cone program, and they showed that their model enhance the robustness of the optimal job sequence. This line of research has attracted the attention of many scholars (Chang et al., 2019; Liu et al., 2019; Zhang et al., 2018; Liu et al., 2020b).

Recently, target-oriented robust models have been applied to address scheduling problems. For instance, Li et al. (2022) investigated a target-based parallel machine scheduling problem, by introducing an underperformance risk index to regulate the extent of the total weighted completion time that exceeds the target level. Similarly, Pei et al. (2022) explored a single machine scheduling problem, aiming to control key indices beyond the target level. Our paper is distinguished from these works in two main aspects. First, we employ a different model framework that directly utilizes data, rather than constructing a model based on the moment-based ambiguity set. Second, we focus on distinct problem settings, particularly emphasizing scheduling problems with release times.

2.3. Scheduling problems with release time

The scheduling problems for a single machine with deterministic release times have been extensively investigated with different objectives. When the total completion time is the primary goal, Smith et al. (1956) demonstrated that the shortest processing time priority rule is optimal when the release times are identical. However, the problem becomes NP-hard with varying release times (Lenstra et al., 1977). To address this challenge, Bianco and Ricciardelli (1982) derived several dominance properties and proposed a branch-and-bound algorithm. Additionally, based on the Lagrangian relaxation method, Hariri and Potts (1983) proposed a branch-and-bound algorithm to address the problem with 50 jobs. Chu (1992a) presented a necessary and sufficient condition for local optimality, aiding in the proposal of a new branch-and-bound algorithm. In addition to the branch-and-bound algorithm, much effort has been directed toward developing heuristics to solve this problem (for example, Liu and MacCarthy, 1991; Chand et al., 1996; Chu, 1992b; Della Croce and T'kindt, 2002; Jouglet et al., 2008). Generally, these heuristics take the form of dispatching rules that construct the schedule one job at a time, and they perform quite well under certain circumstances.

The study of scheduling problems with uncertain release times is limited. For instance, Zheng et al. (2019) investigated a single yard crane (machine) scheduling problem where the retrieval tasks' release times follow a known probability distribution. They proposed a two-stage stochastic programming model and use the SAA approach to solve the problem. Similarly, Liu et al. (2020a) studied a parallel machine scheduling problem where both the release time and processing time are stochastic and they also used a two-stage stochastic programming model with the SAA approach. Unlike these two papers, we assume that the distribution of the release time is unknown in this paper. Under the assumption that only the support set of release times is available, Yue et al. (2018) studied a single machine scheduling problem to minimize the worst-case maximum waiting time. They formulate this problem as mixed-integer programming and propose a two-stage heuristic to solve it. Additionally, Bachtler et al. (2020) proposed the concept of Gamma-robustness and addressed the robust absolute and robust regret criteria. However, both Yue et al. (2018) and Bachtler et al. (2020) solely rely on support information while disregarding distributional information. This approach may result in an overly conservative solution. To the best of our knowledge, Lu and Pei (2022) was the first to employ distributionally robust optimization to examine the scheduling problem with deterministic release time, while we use robust satisficing to examine a problem with uncertain release time.

3. Scheduling problem with deterministic release time and random processing time

In this section, we investigate the scheduling problem with deterministic release time and random processing time. Since the exact distribution of the processing time is difficult to know and the only available information is limited historical data, we construct a robust satisficing model. In Section 3.1, we describe the problem and present the model formulations. In Section 3.2, we derive an equivalent reformulation for the scheduling problem with deterministic release time and random processing time. We also present an approximate reformulation in Section 3.3.

3.1. Model formulation

We consider a single machine scheduling problem in which a set of $\mathcal{N} = \{1, \dots, n\}$ of jobs are processed on one machine. The processing time of job j is denoted as a random variable $\tilde{p}_j, \forall j \in \mathcal{N}$, which is ready to process at its release time r_j. For simplicity, we denote the processing time vector and release time vector by $\tilde{\boldsymbol{p}} = [\tilde{p}_1, \dots, \tilde{p}_n]^T$ and $\boldsymbol{r} = [r_1, \dots, r_n]^T$, respectively. We also establish the joint support set for the processing time of n jobs as $\mathscr{D}^p = \mathscr{D}_1^p \times \cdots \times \mathscr{D}_n^p$, where $\mathscr{D}_j^p = [\underline{p}_j, \overline{p}_j], \forall j \in \mathscr{N}$ represents the support set for the processing time $\tilde{p}_j$. In this context, $\underline{p}_j$ and $\overline{p}_j$ serve as the lower and upper bounds for the processing time $\tilde{p}_j$, respectively. In this paper, we are concerned with the non-preemptive scheduling discipline, which indicates that once the machine starts to execute a job, it must finish that job before executing another. Therefore, the set of feasible sequences $\mathscr{X}$ can be expressed as follows:

$$\mathscr{X} = \left\{ \boldsymbol{X} \;\middle|\; \begin{array}{l} \sum_{i=1}^n x_{ij} = 1, \forall j \in \mathscr{N}, \\ \sum_{j=1}^n x_{ij} = 1, \forall i \in \mathscr{N}, \\ x_{ij} \in \{0,1\}, \forall i,j \in \mathscr{N}. \end{array} \right\}.$$

In this expression, $x_{ij} = 1$ indicates that job j is processed at the ith position; and otherwise, $x_{ij} = 0$. The first constraint in the set $\mathscr{X}$ enforces that a given job can be processed at only one position. The second constraint requires that one position be occupied by only one job. The third constraint means that the job cannot be divided.

Given a feasible sequence $\boldsymbol{X}$, we denote the start time of the job at the ith position by t_i. Note that the start time t_i should be greater than or equal to the release time of the job at the ith position, i.e., $\sum_{j=1}^n x_{i-1,j}\tilde{p}_j$, and the completion time of job at the $(i-1)$-th position, i.e., $\sum_{j=1}^n x_{i-1,j}\tilde{p}_j$. Given the start time t_i, the completion time of the job at the ith position is $t_i + \sum_{j=1}^n x_{ij}\tilde{p}_j$. Therefore, the total completion time can be expressed as $\sum_{i=1}^n \left[t_i + \sum_{j=1}^n x_{ij}\tilde{p}_j\right]$. Our goal is to identify a sequence such that the expected total completion time is minimized.[1] In the ideal scenario where the distribution of processing time is completely known, one can solve the following stochastic scheduling problem with deterministic release time (SSD),

$$\text{(SSD)}\min \quad \mathbb{E}_{\mathbb{P}_p^*} \sum_{i=1}^n \left[t_i + \sum_{j=1}^n x_{ij}\tilde{p}_j \right] \tag{1}$$

$$\text{s.t. } t_i \ge \sum_{j=1}^n x_{ij} r_j, \qquad \forall i \in \mathscr{N}, \tag{2}$$

$$t_i - t_{i-1} \ge \sum_{j=1}^n x_{i-1,j}\tilde{p}_j, \qquad \forall i \in \mathscr{N}, \tag{3}$$

$$\boldsymbol{X} \in \mathscr{X},$$

where $t_0 = 0$, $x_{0,j} = 0, \forall j \in \mathscr{N}$ and $\mathbb{P}_p^*$ is the true distribution that characterizes processing time $\tilde{\boldsymbol{p}}$. In this formulation, the decision variables are $\boldsymbol{X} \in \mathscr{X}$ and $\boldsymbol{t}$. The objective (1) is to minimize the expected total completion time. The constraint (2) means that the start time at the ith position should be greater than or equal to the release time of the job at the ith position. The constraint (3) indicates that the start time at the ith position should be greater than or equal to the completion time of the job at the previous position.

Empirical scheduling problem with deterministic release time (ESD)

In practice, we generally do not know $\mathbb{P}_p^*$ and only limited historical data can be used. In this problem, we denote the historical processing time samples as $\hat{\boldsymbol{p}}_m, \forall m \in \mathscr{M} = \{1,\dots,M\}$. Let $\hat{\mathbb{P}}_p \in \mathscr{P}_0(\mathscr{D}^p)$ denote the empirical distribution constructed from historical data, i.e., $\hat{\mathbb{P}}_p[\tilde{\boldsymbol{p}} = \hat{\boldsymbol{p}}_m] = \frac{1}{M}$. As an approximation for the problem (SSD), we can consider the following empirical scheduling problem,

$$\text{(ESD)}\min \quad \frac{1}{M}\sum_{m=1}^M \sum_{i=1}^n \left[t_i^m + \sum_{j=1}^n x_{ij}\hat{p}_{mj} \right] \tag{4}$$

$$\text{s.t. } t_i^m \ge \sum_{j=1}^n x_{ij} r_j, \qquad \forall i \in \mathscr{N}, \forall m \in \mathscr{M}, \tag{5}$$

$$t_i^m - t_{i-1}^m \ge \sum_{j=1}^n x_{i-1,j}\hat{p}_{mj}, \qquad \forall i \in \mathscr{N}, \forall m \in \mathscr{M}, \tag{6}$$

$$\boldsymbol{X} \in \mathscr{X}.$$

In this formulation, t_i^m represents the start time of the job at the ith position under scenario m. The objective (4) is to minimize the expected total completion time under the empirical distribution $\hat{\mathbb{P}}p$. Constraints (5) and (6) correspond to the requirements of release time and start time at each position.

Robust satisficing scheduling problem with deterministic release time (RSD)

It is well-known that solving an optimization model with the empirical distribution may result in solutions that perform poorly against the true distribution (e.g., Kleywegt et al., 2002; Smith and Winkler, 2006). To address this problem, Long et al. (2023) propose a target-oriented robust satisficing model that withstands more significant distribution ambiguity. In the robust satisficing model, we use $f(\mathbf{x},\mathbf{z}) : \mathscr{X} \times \mathscr{Z} \to \mathbb{R}$ to denote as the objective function, where the first argument $\mathbf{x} \in \mathscr{X}$ is the decision variable and the second argument $\mathbf{z} \in \mathscr{Z}$ is the uncertain parameters or outcomes. Let $\hat{\mathbb{P}}_{\mathbf{z}}$ denoted the empirical distribution constructed from the historical data $\{\hat{\mathbf{z}}_m\}_{m\in\mathcal{M}}$, that is, $\hat{\mathbb{P}}_{\mathbf{z}}[\hat{\mathbf{z}} = \hat{\mathbf{z}}_m] = \frac{1}{M}$, then a general robust satisficing model under the Wasserstein metric is formulated as follows:

$$\begin{aligned} \min \quad & \kappa \\ \text{s.t.} \, & \mathbb{E}_{\mathbb{Q}_{\mathbf{z}}}[f(\mathbf{x},\mathbf{z})] - \tau \le \kappa \mathbb{E}_{\mathbb{Q}_{\mathbf{z}}}[\|\mathbf{z} - \hat{\mathbf{z}}\|], \qquad \forall \mathbb{Q}_{\mathbf{z}} \in \mathscr{F}_{\mathbf{z}}, \quad (7) \\ & \mathbf{x} \in \mathscr{X}, \\ & \kappa \ge 0. \end{aligned}$$

where the ambiguity set of the joint distribution $\mathbb{Q}_{\mathbf{z}}$ is defined as

$$\mathscr{F}_{\mathbf{z}} = \left\{ \mathbb{Q}_{\mathbf{z}} \in \mathscr{P}_0\left(\mathscr{D}^{\mathbf{z}} \times \mathscr{D}^{\mathbf{z}}\right) \,\middle|\, (\mathbf{z}, \hat{\mathbf{z}}) \sim \mathbb{Q}_{\mathbf{z}}, \hat{\mathbf{z}} \sim \hat{\mathbb{P}}_{\mathbf{z}} \right\},$$

It is crucial to note that the parameter τ represents the target that the decision-maker is willing to accept, relative to a reference such as the objective of the empirical optimization problem. For any potential distribution $\mathbb{Q}_{\mathbf{z}}$ within the support set $\mathscr{F}_{\mathbf{z}}$, the constraint (7) ensures that the gap between the objective value and the target, i.e., $\mathbb{E}_{\mathbb{Q}_{\mathbf{z}}}[f(\mathbf{x},\mathbf{z})] - \tau$, is bounded by the product of the magnitude κ and the distribution distance $\mathbb{E}_{\mathbb{Q}_{\mathbf{z}}}[\|\mathbf{z} - \hat{\mathbf{z}}\|]$. Consequently, the robust satisficing model seeks a solution that minimizes the deviation magnitude for all potential distributions.

Importantly, in this work, we present an equivalent formulation tailored to our scheduling problem with deterministic release time, rather than merely adopting the approximated reformulation employed for solving the two-stage linear optimization problem in Long et al. (2023).

For our stochastic scheduling problem, we can formulate a robust satisficing scheduling problem with deterministic release time as

[1] **Total Flow Time objective.** In addition to total completion time, total flow time (TFT) is another widely recognized metric, that is the sum of waiting time and processing time, and represents the duration from release of a job to its completion. Our robust satisficing models are applicable to the TFT objective. Given a scheduling decision $\boldsymbol{X}$ and sequences of processing times $\boldsymbol{p}$ and release times $\boldsymbol{r}$, the TFT can be modeled as

$$\begin{aligned} Q^{\mathsf{TFT}}(\boldsymbol{X},\boldsymbol{p},\boldsymbol{r}) = \min & \left\{ \sum_{i=1}^n \left[\left(t_i - r_{[i]} + p_{[i]}\right) \right] \right\} \\ \text{s.t. } & t_i \ge r_{[i]}, \qquad \forall i \in \mathscr{N}, \\ & t_i - t_{i-1} \ge p_{[i-1]}, \qquad \forall i \in \mathscr{N}. \end{aligned}$$

Following a similar analysis in Lemma 1, we can obtain a reformulation $Q^{\mathsf{TFT}}(\boldsymbol{X},\boldsymbol{p},\boldsymbol{r}) = \max_{\beta\in\mathscr{A}} \sum_{i=1}^n \left[(p_{[i-1]} - s_{[i]})\beta_i + p_{[i]}\right]$, where $\mathscr{A}$ is defined in Lemma 1. Since our robust satisficing models are built upon the $Q(\cdot)$ function, all subsequent analysis follows under $Q^{\mathsf{TFT}}(\cdot)$.

follows:

$$\begin{aligned}
\text{(RSD)}\min\quad & \kappa \\
\text{s.t.}\quad & \mathbb{E}_{\mathbb{Q}_p}[Q(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r})]-\tau \le \kappa\mathbb{E}_{\mathbb{Q}_p}[\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_1], \quad \forall \mathbb{Q}_p\in\mathscr{F}_p, && (8)\\
& \boldsymbol{X}\in\mathscr{X}, && (9)\\
& \kappa\ge 0, && (10)
\end{aligned}$$

with

$$\begin{aligned}
Q(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r}) = \min & \left\{\sum_{i=1}^{n}\left[(t_i+\tilde{p}_{[i]})\right]\right\}\\
\text{s.t. } & t_i \ge r_{[i]}, && \forall i\in\mathscr{N},\\
& t_i - t_{i-1} \ge \tilde{p}_{[i-1]}, && \forall i\in\mathscr{N},
\end{aligned}$$

where

$$\mathscr{F}_p = \left\{\mathbb{P}_p\in\mathscr{P}_0(\mathscr{D}^p\times\mathscr{D}^p)\,\middle|\,(\tilde{\boldsymbol{p}},\hat{\boldsymbol{p}})\sim\mathbb{Q}_p, \hat{\boldsymbol{p}}\sim\hat{\mathbb{P}}_p\right\},$$

and $r_{[i]} = \sum_{j=1}^n x_{ij}r_j$ and $\tilde{p}_{[i]} = \sum_{j=1}^n x_{ij}\tilde{p}_j$. In our robust satisficing scheduling problem, τ represents the target that the decision-maker is willing to accept, and $\mathbb{E}_{\mathbb{Q}_p}[\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_1]$ represents the distribution distance. Given $(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r})$, the second stage problem $Q(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r})$ is to minimize the total completion time by determining the optimal start time $\boldsymbol{t}$. Therefore, the constraint (8) ensures that the expected violation (i.e., $\mathbb{E}_{\mathbb{Q}_p}[Q(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r})]-\tau$) is restricted by the product of the statistical distance $\mathbb{E}_{\mathbb{Q}_p}[\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_1]$ and the magnitude κ for all possible distributions in the set $\mathscr{F}_p$. The constraint (9) enforces the requirement of a feasible sequence. The constraint (10) ensures that the magnitude is greater than or equal to 0. The objective of the robust satisficing model is to minimize the magnitude of expected target violation that can occur under any distribution.

Like in specifying the hyper-parameter in the distributionally robust optimization problem, the decision-maker must determine the target τ for the robust satisficing model. We note that τ is related to the expected total completion time and can be considered the target that the decision-maker is willing to accept. In comparison to hyper-parameters such as the radius of the Wasserstein ball for the Wasserstein ambiguity set in distributionally robust optimization, specifying the target τ is more intuitive for the decision-maker.

3.2. Exact reformulation for model (RSD)

In the following, we present an equivalent mixed-integer linear program for the model (RSD). The reformulation starts from Lemma 1, which provides an equivalent reformulation for the second stage problem $Q(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r})$.

Lemma 1. *For any given $\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r}$, the second stage problem $Q(\boldsymbol{X},\tilde{\boldsymbol{p}},\boldsymbol{r})$ equals the following problem:*

$$\max_{\boldsymbol{\beta}\in\mathscr{A}}\sum_{i=1}^{n}\left[(\tilde{p}_{[i-1]}-s_{[i]})\beta_i+\tilde{p}_{[i]}+r_{[i]}\right], \tag{11}$$

where $s_{[i]} = r_{[i]}-r_{[i-1]}$, $r_{[0]}=0$ and

$$\mathscr{A} = \left\{\begin{array}{l} 1+\beta_i-\beta_{i-1}\ge 0, \forall i=2,\dots,n,\\ \beta_n\le 1,\\ \boldsymbol{\beta}\ge 0.\end{array}\right\}.$$

According to Lemma 1, constraint (8) in model (RSD) can be reformulated as follows:

$$\mathbb{E}_{\mathbb{Q}_p}\left\{\max_{\boldsymbol{\beta}\in\mathscr{A}}\sum_{i=1}^{n}\left[(\tilde{p}_{[i-1]}-s_{[i]})\beta_i+\tilde{p}_{[i]}\right]\right\} + \sum_{i=1}^{n} r_i - \tau \le \kappa\mathbb{E}_{\mathbb{Q}_p}[\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_1], \forall\mathbb{Q}_p\in\mathscr{F}_p. \tag{12}$$

Next, we present an equivalent formulation for the inequality (12).

Proposition 1. *The inequality (12) equals the following set of inequalities:*

$$\begin{aligned}
& \frac{1}{M}\sum_{m=1}^{M}\theta_m \le \tau - \sum_{i=1}^{n} r_i, \\
& \theta_m \ge \sum_{i=1}^{n}\lambda_i^m, && \forall m\in\mathscr{M},\\
& \sum_{i=k}^{\min\{j,n\}}\lambda_i^m \ge \sum_{i=k}^{\min\{j,n\}} v_{ij}^m, && \forall 1\le k\le n, k\le j\le n+1, m\in\mathscr{M},\\
& v_{ij}^m \ge -s_{[i]}\pi_{ij} + (\pi_{ij}+1)\bar{p}_{[i-1]} \\
& \quad - \kappa(\bar{p}_{[i-1]} - \hat{p}_{m,[i-1]}), && \forall 2\le i\le n, i\le j\le n+1, m\in\mathscr{M},
\end{aligned} \tag{13}$$

$$\begin{aligned}
& v_{ij}^m \ge -s_{[i]}\pi_{ij} + (\pi_{ij}+1)\hat{p}_{m,[i-1]}, && \forall 2\le i\le n, i\le j\le n+1, m\in\mathscr{M},\\
& v_{1j}^m \ge -s_{[1]}\pi_{1j} + \bar{p}_{[n]} - \kappa(\bar{p}_{[n]} - \hat{p}_{m,[n]}), && 1\le j\le n+1, m\in\mathscr{M},
\end{aligned} \tag{14}$$

$$v_{1j}^m \ge -s_{[1]}\pi_{1j} + \hat{p}_{m,[n]}, \qquad 1\le j\le n+1, m\in\mathscr{M},$$

where $\pi_{ij} = j-i$, $\bar{p}_{[i]} = \sum_{j=1}^n x_{ij}\bar{p}_j$ and $\hat{p}_{m,[i]} = \sum_{j=1}^n x_{ij}\hat{p}_{m,j}, \forall i\in\mathscr{M}$.

Following Proposition 1, Theorem 1 easily reformulates the problem (RSD) into a mixed-integer linear program.

Theorem 1. *The robust satisficing scheduling problem* (RSD) *is equivalent to the following mixed-integer linear program:*

$$\begin{aligned}
\text{(RSDE)}\min\quad & \kappa \\
\text{s.t.}\quad & \tfrac{1}{M}\textstyle\sum_{i=1}^{n}\theta_m \le \tau - \sum_{i=1}^{n} r_i, \\
& \theta_m \ge \textstyle\sum_{i=1}^{n}\lambda_i^m, && \forall m\in\mathscr{M},\\
& \sum_{i=k}^{\min\{j,n\}}\lambda_i^m \ge \sum_{i=k}^{\min\{j,n\}} v_{ij}^m, && \forall 1\le k\le n, k\le j\le n+1, m\in\mathscr{M},\\
& v_{ij}^m \ge \sum_{q=1}^{n}\left[(\pi_{ij}+1)x_{i-1,q}\bar{p}_q \right.\\
& \quad \left. - y_{i-1,q}(\bar{p}_q - \hat{p}_{mq})\right] - s_{[i]}\pi_{ij}, && \forall 2\le i\le n, i\le j\le n+1, m\in\mathscr{M},\\
& v_{ij}^m \ge (\pi_{ij}+1)\sum_{q=1}^{n} x_{i-1,q}\hat{p}_{m,q} \\
& \quad - s_{[i]}\pi_{ij}, && \forall 2\le i\le n, i\le j\le n+1, m\in\mathscr{M},\\
& v_{1j}^m \ge \sum_{q=1}^{n} x_{n,q}\bar{p}_q \\
& \quad - \sum_{q=1}^{n} y_{n,q}(\bar{p}_q - \hat{p}_{m,q}) - s_{[i]}\pi_{ij}, && 1\le j\le n+1, m\in\mathscr{M},\\
& v_{1j}^m \ge -s_{[i]}\pi_{ij} + \sum_{q=1}^{n} x_{n,q}\hat{p}_{m,q}, && 1\le j\le n+1, m\in\mathscr{M},\\
& y_{ij}\ge 0, && \forall i\in\mathscr{N}, j\in\mathscr{N},\\
& y_{ij}\ge \kappa - B(1-x_{ij}), && \forall i\in\mathscr{N}, j\in\mathscr{N},\\
& y_{ij}\le \kappa, && \forall i\in\mathscr{N}, j\in\mathscr{N},\\
& y_{ij}\le Bx_{ij}, && \forall i\in\mathscr{N}, j\in\mathscr{N},\\
& \kappa\ge 0,
\end{aligned}$$

where B is a sufficiently large value.

Recalling the definitions of the $\bar{p}_{[i-1]} = \sum_{j=1}^n x_{i-1,j}\bar{p}_j$ and $\hat{p}_{m,[i-1]} = \sum_{j=1}^n x_{i-1,j}\hat{p}_{mj}$, there exist bilinear terms $\kappa x_{i-1,j}$ in constraints (13) and (14). These terms can be linearized by introducing the new variables y_{ij}. Then, the model (RSDE) is obtained. Since this model is a mixed-integer linear program, it can be solved by state-of-the-art commercial solvers, such as CPLEX and GUROBI.

3.3. Approximate reformulation for model (RSD)

The exact reformulation of the robust satisficing scheduling problem becomes computationally challenging when the number of jobs increases due to the integer programming nature of the problem. To expedite the problem-solving process, we propose an approximate reformulation based on the linear decision rule (LDR) as an alternative to the exact reformulation in this section. We show with numerical experiments in Section 5 that this approximation can be solved quickly while still providing a satisfactory solution.

LDR is a widely used technique in two-stage robust optimization problems (e.g., Ben-Tal et al., 2004; Chen et al., 2008; Goh and Sim, 2011). The main idea behind the LDR is to assume that the second-stage decision variable is an affine function of random variables. In our robust satisficing model, given $\tilde{\boldsymbol{p}}$, the solution to the second-stage problem $Q(\boldsymbol{X}, \tilde{\boldsymbol{p}}, \boldsymbol{r})$ is a function of $\tilde{\boldsymbol{p}}$, which is denoted as $\boldsymbol{t}(\tilde{\boldsymbol{p}}) \in \mathscr{R}^{n,n} := \{\boldsymbol{t}(\cdot) : \mathbb{R}^n \to \mathbb{R}^n\}$. Therefore, instead of solving for decisions $\mathbb{R}^n$, we can instead look for a function $\boldsymbol{t}(\cdot)$; and the robust satisficing model with deterministic release time thus becomes

$$
\begin{aligned}
\min \quad & \kappa \\
\text{s.t. } & \mathbb{E}_{\mathbb{Q}_p}\left[\sum_{i=1}^{n}\left(t_i(\tilde{\boldsymbol{p}}) + \tilde{p}_{[i]}\right)\right] - \tau \le \kappa \mathbb{E}_{\mathbb{Q}_p} \|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_1, && \forall \mathbb{Q}_p \in \mathscr{F}_p, \\
& t_i(\tilde{\boldsymbol{p}}) \ge \tilde{r}_{[i]}, && \forall i \in \mathscr{N}, \tilde{\boldsymbol{p}} \in \mathscr{D}^p, \\
& t_i(\tilde{\boldsymbol{p}}) - t_{i-1}(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{r}}) \ge \tilde{p}_{[i-1]}, && \forall i \in \mathscr{N}, \tilde{\boldsymbol{p}} \in \mathscr{D}^p, \\
& \boldsymbol{X} \in \mathscr{X}, \\
& \kappa \ge 0, \\
& t_i(\tilde{\boldsymbol{p}}) \in \mathscr{R}^{n,1}, && \forall i \in \mathscr{N}.
\end{aligned}
$$

However, the function space $\mathscr{R}^{n,n}$ is too large to manage. For tractability reasons, LDR assumes that the decision variable $\boldsymbol{t}(\tilde{\boldsymbol{p}})$ is an affine function of the random variables. Specifically, given $\tilde{\boldsymbol{p}}$, one can restrict the form of the $t_i(\tilde{\boldsymbol{p}})$ to the following set:

$$
\mathscr{L}^{n,n} = \left\{ \boldsymbol{t} \in \mathscr{R}^{n,n} \;\middle|\; \begin{array}{l} \exists \boldsymbol{t}^0, \boldsymbol{t}_i^1 \in \mathbb{R}^n, \forall i \in \mathscr{N}, \\ \boldsymbol{t}(\tilde{\boldsymbol{p}}) = \boldsymbol{t}^0 + \sum_{i=1}^{n} \boldsymbol{t}_i^1 \tilde{p}_i. \end{array} \right\}.
$$

Note that $\boldsymbol{t}^0$ and $\boldsymbol{t}_i^1, \forall i \in \mathscr{N}$ are determined by solving the optimization model. Then, we consider the following approximate robust satisficing scheduling problem:

$$
\begin{aligned}
(\overline{\text{RSD}}) \quad \min \quad & \kappa \\
\text{s.t. } & \mathbb{E}_{\mathbb{Q}_p}\left[\sum_{i=1}^{n}\left(t_i(\tilde{\boldsymbol{p}}) + \tilde{p}_{[i]}\right)\right] \\
& \quad - \tau \le \kappa \mathbb{E}_{\mathbb{Q}_p} \|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_1, && \forall \mathbb{Q}_p \in \mathscr{F}_p, && (15) \\
& t_i(\tilde{\boldsymbol{p}}) \ge \tilde{r}_{[i]}, && \forall i \in \mathscr{N}, \tilde{\boldsymbol{p}} \in \mathscr{D}^p, && (16) \\
& t_i(\tilde{\boldsymbol{p}}) - t_{i-1}(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{r}}) \ge \tilde{p}_{[i-1]}, && \forall i \in \mathscr{N}, \tilde{\boldsymbol{p}} \in \mathscr{D}^p, && (17) \\
& t_i(\tilde{\boldsymbol{p}}) = t_i^0 + \sum_{j=1}^{n} t_{ij}^1 \tilde{p}_j, && \forall i \in \mathscr{N}, && (18) \\
& \boldsymbol{x} \in \mathscr{X}, && && (19) \\
& \kappa \ge 0, && && (20) \\
& \boldsymbol{t}^0, \boldsymbol{t}_i^1 \in \mathbb{R}^n, && \forall i \in \mathscr{N}.
\end{aligned}
$$

In this formulation, the constraint (18) is the expression of the linear decision rule. The objective of the robust satisficing model is to minimize the magnitude of expected target violation that could occur under any distribution. Note that $\boldsymbol{t}^0, \boldsymbol{t}_i^1$ are determined by solving the model ($\overline{\text{RSD}}$).

Theorem 2. *The robust satisficing scheduling problem with deterministic release time* ($\overline{\text{RSD}}$) *is equivalent to the following formulation:*

$$
\begin{aligned}
(\text{RSDA}) \quad \min \quad & \kappa \\
\text{s.t. } & \frac{1}{M}\sum_{m=1}^{M} \theta_m \le \tau, && && (21) \\
& \theta_m \ge \sum_{i=1}^{n} t_i^0 - \boldsymbol{\zeta}_m^T \hat{\boldsymbol{p}}_m + \underline{\boldsymbol{p}}^T \boldsymbol{u}_m^p + \bar{\boldsymbol{p}}^T \boldsymbol{v}_m^p, && \forall m \in \mathscr{M}, && (22) \\
& -\zeta_{mj} + u_{mj}^p + v_{mj}^p = \sum_{i=1}^{n} t_{ij}^1 + 1, && \forall j \in \mathscr{N}, m \in \mathscr{M}, && (23) \\
& \begin{bmatrix} \kappa \\ \boldsymbol{\zeta}_m \end{bmatrix} \ge_\infty 0, && \forall m \in \mathscr{M}, && (24) \\
& t_i^0 - r_{[i]} \ge \sum_{j=1}^{n}\left(\omega_{ij}^{mp} \underline{p}_j + \psi_{ij}^{mp} \bar{p}_j\right), && \forall i \in \mathscr{N}, m \in \mathscr{M}, && (25) \\
& \omega_{ij}^{mp} + \psi_{ij}^{mp} = -t_{ij}^1, && \forall i, j \in \mathscr{N}, m \in \mathscr{M}, && (26) \\
& t_i^0 - t_{i-1}^0 \ge \sum_{j=1}^{n}\left(\phi_{ij}^{mp} \underline{p}_j + \pi_{ij}^{mp} \bar{p}_j\right), && \forall i \in \mathscr{N}, m \in \mathscr{M}, && (27) \\
& \phi_{ij}^{mp} + \pi_{ij}^{mp} = x_{i-1,j} + t_{i-1,j}^1 - t_{ij}^1, && \forall i, j \in \mathscr{N}, m \in \mathscr{M}, && (28) \\
& \boldsymbol{u}_m^p \le 0, \boldsymbol{\omega}^{mp} \le 0, \boldsymbol{\phi}^{mp} \le 0, && \forall m \in \mathscr{M}, && (29) \\
& \boldsymbol{v}_m^p \ge 0, \boldsymbol{\psi}^{mp} \ge 0, \boldsymbol{\pi}^{mp} \ge 0, && \forall m \in \mathscr{M}, && (30) \\
& \kappa \ge 0. && && (31)
\end{aligned}
$$

Note that constraint (24) indicates that vector $[\kappa; \boldsymbol{\zeta}_m]$ is an ∞-normal cone, i.e., $\max_{j \in \mathscr{N}} |\zeta_{mj}| \le \kappa$. As a mixed-integer linear program, this model can be solved by state-of-the-art commercial solvers, such as CPLEX and GUROBI.

To further increase the model-solving speed, we observe that for any $0 \le \kappa_1 \le \kappa_2$, if κ_1 is a feasible solution to the model ($\overline{\text{RSD}}$), then κ_2 is also a feasible solution. Because model (RSDA) is an equivalent reformulation for model ($\overline{\text{RSD}}$), we can solve model (RSDA) based on the binary search algorithm. Specifically, given a $\kappa \ge 0$, if we can find a solution $(\boldsymbol{\theta}, \boldsymbol{\zeta}_m, \boldsymbol{u}_m^p, \boldsymbol{v}_m^p, \boldsymbol{\omega}^{mp}, \boldsymbol{\phi}^{mp}, \boldsymbol{\psi}^{mp}, \boldsymbol{\pi}^{mp})$ that satisfies the constraint (21)–(30), then we know κ and $(\boldsymbol{\theta}, \boldsymbol{\zeta}_m, \boldsymbol{u}_m^p, \boldsymbol{v}_m^p, \boldsymbol{\omega}^{mp}, \boldsymbol{\phi}^{mp}, \boldsymbol{\psi}^{mp}, \boldsymbol{\pi}^{mp})$ are feasible solutions to the problem (RSDA). Therefore, we can use a binary search algorithm to find a minimum value for κ such that there exists a feasible solution $(\boldsymbol{\theta}, \boldsymbol{\zeta}_m, \boldsymbol{u}_m^p, \boldsymbol{v}_m^p, \boldsymbol{\omega}^{mp}, \boldsymbol{\phi}^{mp}, \boldsymbol{\psi}^{mp}, \boldsymbol{\pi}^{mp})$ that satisfies the constraint (21)–(30). We present our binary search algorithm as follows:

Algorithm 1 Binary search algorithm

Input: Target τ, the upper bound $\bar{\kappa}$ and lower bound $\underline{\kappa}$, the tolerance Δ
while $\bar{\kappa} - \underline{\kappa} \ge \Delta$ **do**
  $\kappa = \frac{1}{2}(\bar{\kappa} + \underline{\kappa})$
  Solve model (RSI), obtain the objective value T and solution $\boldsymbol{X}'$

$$
(\text{RSI}) \quad \min \frac{1}{M}\sum_{m=1}^{M} \theta_m \quad \text{s.t } (22) - (30)
$$

  if $T \le \tau$ **then**
    Let $\bar{\kappa} \leftarrow \kappa$ and $\boldsymbol{X}^\dagger \leftarrow \boldsymbol{X}'$
  else if $T > \tau$ **then**
    Let $\underline{\kappa} \leftarrow \kappa$
  end if
end while
Output: Solution $\boldsymbol{X}^\dagger$

In this algorithm, given the input target τ, upper and lower bounds $\bar{\kappa}$ and $\underline{\kappa}$ for κ, and a sufficiently small tolerance Δ, we seek a feasible solution such that $\overline{\kappa} - \underline{\kappa} < \Delta$. Initially, we set $\kappa = \frac{1}{2}(\bar{\kappa} + \underline{\kappa})$ and solve the model (RSI). If the objective T is less than the target τ, then κ and $\boldsymbol{X}'$ represent feasible solutions for the model (RSDA). Otherwise, we update $\overline{\kappa} = \kappa$. This process is repeated until $\overline{\kappa} - \underline{\kappa} < \Delta$. Throughout our work, we choose Δ to be 0.1. Finally, we designate $\boldsymbol{X}^{\dagger}$ as the optimal scheduling sequence for the model (RSDA) with target τ.

4. Scheduling problem with random release and processing time

In this section, we study a single machine scheduling problem with both random release and processing times. In Section 4.1, we present the model formulations under random release time and processing time. In Section 4.2, we first show that the exact formulation is difficult to obtain by incorporating the random release time. Then, we present an approximate formulation for this problem. In Section 5.4, we conduct numerical experiments to show that our approximate formulation outperforms SAA and the deterministic approach.

4.1. Model formulation

In this problem, we assume that both the release time $\tilde{\boldsymbol{r}} = [\tilde{r}_1, \dots, \tilde{r}_n]^T$ and processing time $\tilde{\boldsymbol{p}} = [\tilde{p}_1, \dots, \tilde{p}_n]^T$ are random variables. The support set for each job release time and processing time are $\mathscr{D}_j^r = [\underline{r}_j, \overline{r}_j], \forall j \in \mathscr{N}$ and $\mathscr{D}_j^p = [\underline{p}_j, \overline{p}_j], \forall j \in \mathscr{N}$, respectively. The joint support set for the release time and processing time are $\mathscr{D}^r = \mathscr{D}_1^r \times \cdots \times \mathscr{D}_n^r$ and $\mathscr{D}^p = \mathscr{D}_1^p \times \cdots \times \mathscr{D}_n^p$. The objective is to minimize the expected total completion time. Based on the description in Section 3.1, the ideal stochastic scheduling problem with random release and processing times can be formulated as follows:

$$
\begin{aligned}
\text{(SSR)}\min \quad & \mathbb{E}_{\mathbb{P}^*} \sum_{i=1}^{n} \left[t_i + \sum_{j=1}^{n} x_{ij}\tilde{p}_j \right] \\
\text{s.t. } & t_i \geq \sum_{j=1}^{n} x_{ij}\tilde{r}_j, && \forall i \in \mathscr{N}, \\
& t_i - t_{i-1} \geq \sum_{j=1}^{n} x_{i-1,j}\tilde{p}_j, && \forall i \in \mathscr{N}, \\
& \boldsymbol{X} \in \mathscr{X},
\end{aligned}
$$

where $\mathbb{P}^*$ is the true distribution that characterizes release and processing time $(\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{r}})$. In this formulation, the decision variables are $\boldsymbol{X}$ and $\boldsymbol{t}$. The objective is to minimize the expected total completion time.

Empirical scheduling problem with random release time

However, the true distribution $\mathbb{P}^*$ is difficult to know in advance, and the only available information is M release and processing time samples $(\hat{\boldsymbol{r}}_m, \hat{\boldsymbol{p}}_m), \forall m \in \mathscr{M}$. The empirical distribution of the historical samples is denoted by $\hat{\mathbb{P}}$, i.e., $\hat{\mathbb{P}}[(\hat{\boldsymbol{p}}, \hat{\boldsymbol{r}}) = (\hat{\boldsymbol{p}}_m, \hat{\boldsymbol{r}}_m)] = \frac{1}{M}$. Then, one may consider the following empirical scheduling problem with random release time,

$$
\begin{aligned}
\text{(ESR)}\min \quad & \frac{1}{M} \sum_{m=1}^{M} \sum_{i=1}^{n} \left[t_i^m + \sum_{j=1}^{n} x_{ij}\hat{p}_{mj} \right] && (32)\\
\text{s.t. } & t_i^m \geq \sum_{j=1}^{n} x_{ij}\hat{r}_{mj}, && \forall i \in \mathscr{N}, \forall m \in \mathscr{M}, \quad (33)\\
& t_i^m - t_{i-1}^m \geq \sum_{j=1}^{n} x_{i-1,j}\hat{p}_{mj}, && \forall i \in \mathscr{N}, \forall m \in \mathscr{M}, \quad (34)\\
& \boldsymbol{X} \in \mathscr{X}.
\end{aligned}
$$

In this formulation, t_i^m is the start time of the job at ith position under the scenario m. The objective (32) is to minimize the expected total completion time under the empirical distribution $\hat{\mathbb{P}}$. The constraint (33) requires the start time, t_i^m, is greater than the job release time at the ith position. The constraint (34) enforces that the start time at ith position must be greater than the completion time of the job at $(i-1)$-th position.

Robust satisficing scheduling problem with random release time

As mentioned in the discussion in Section 3.1, the empirical optimization model may perform poorly in out-of-sample tests. We consider the following robust satisficing model with random release time:

$$
\begin{aligned}
\text{(RSR)}\min \quad & \kappa \\
\text{s.t. } & \mathbb{E}_{\mathbb{Q}}[Q(\boldsymbol{X}, \tilde{\boldsymbol{p}}, \boldsymbol{r})] - \tau \leq \kappa \mathbb{E}_{\mathbb{Q}}\left(\|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_1 + \|\tilde{\boldsymbol{r}} - \hat{\boldsymbol{r}}\|_1 \right), \quad \forall \mathbb{Q} \in \mathscr{F}_r, && (35)\\
& \boldsymbol{X} \in \mathscr{X}, && (36)\\
& \kappa \geq 0, && (37)
\end{aligned}
$$

where the ambiguity set $\mathscr{F}_r$ is defined as follows:

$$
\mathscr{F}_r = \left\{ \mathbb{Q} \in \mathscr{P}_0(\mathscr{D}^p \times \mathscr{D}^r \times \mathscr{D}^p \times \mathscr{D}^r) \,\middle|\, (\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{r}}, \hat{\boldsymbol{p}}, \hat{\boldsymbol{r}}) \sim \mathbb{Q}, (\hat{\boldsymbol{p}}, \hat{\boldsymbol{r}}) \sim \hat{\mathbb{P}} \right\}.
$$

In this formulation, τ represents a target total completion time. The first constraint (35) enforces that the expected violation (i.e., $\mathbb{E}_{\mathbb{Q}}[Q(\boldsymbol{X}, \tilde{\boldsymbol{p}}, \boldsymbol{r})] - \tau$) is restricted by the product of the statistical distance $\mathbb{E}_{\mathbb{Q}}\left(\|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_1 + \|\tilde{\boldsymbol{r}} - \hat{\boldsymbol{r}}\|_1 \right)$ and the magnitude κ for all possible distributions in the set $\mathscr{F}_r$. The objective of the robust satisficing model minimizes the magnitude of expected target violation that could occur under any distribution.

4.2. Approximate reformulation for model (RSR)

In this section, we first show that the model (RSR) cannot be reformulated as the equivalent mixed-integer linear program. Then, we employ the linear decision rule to provide an approximate formulation.

Claim 3. *The constraint (35) equals to the following inequalities,*

$$
\left\{
\begin{aligned}
& \frac{1}{M} \sum_{m=1}^{M} \theta_m \leq \tau, \\
& \theta_m \geq \max_{\tilde{p} \in \mathscr{D}^p, \tilde{r} \in \mathscr{D}^r} \left\{ \max_{\beta \in \mathscr{A}} \sum_{i=1}^{n} \left[(\tilde{p}_{[i-1]} - \tilde{s}_{[i]})\beta_i + \tilde{p}_{[i]} + \tilde{r}_{[i]} \right.\right. \\
& \quad \left.\left. - \kappa(\|\tilde{p}_i - \hat{p}_{m,i}\|_1 + \|\tilde{r}_i - \hat{r}_{m,i}\|_1) \right] \right\}, \forall m \in \mathscr{M}
\end{aligned}
\right\}. \quad (38)
$$

The proof is omitted since it is the same as the proof in Proposition 1. Unfortunately, we cannot provide the equivalent reformulation for this set of inequalities. To show this, we consider the constraint (38),

$$
\begin{aligned}
\theta_m & \geq \max_{\tilde{p} \in \mathscr{D}^p, \tilde{r} \in \mathscr{D}^r} \left\{ \max_{\beta \in \mathscr{A}} \sum_{i=1}^{n} \left[(\tilde{p}_{[i-1]} - \tilde{s}_{[i]})\beta_i + \tilde{p}_{[i]} \right.\right. \\
& \qquad \left.\left. + \ \tilde{r}_{[i]} - \kappa(\|\tilde{p}_i - \hat{p}_{m,i}\|_1 + \|\tilde{r}_i - \hat{r}_{m,i}\|_1) \right] \right\} \\
& = \max_{\beta \in \mathscr{A}} \max_{\tilde{p} \in \mathscr{D}^p, \tilde{r} \in \mathscr{D}^r} \left\{ \sum_{i=1}^{n} \left[(\tilde{p}_{[i-1]} - (\tilde{r}_{[i]} - \tilde{r}_{[i-1]}))\beta_i + \tilde{p}_{[i]} \right.\right. \\
& \qquad \left.\left. + \ \tilde{r}_{[i]} - \kappa(\|\tilde{p}_i - \hat{p}_{m,i}\|_1 + \|\tilde{r}_i - \hat{r}_{m,i}\|_1) \right] \right\}. && (39)
\end{aligned}
$$

Note that the order of $\max_{\tilde{p} \in \mathscr{D}^p, \tilde{r} \in \mathscr{D}^r}$ and $\sum_{i=1}^{n}$ on the right-hand side cannot be changed, because $(\tilde{r}_{[i]} - \tilde{r}_{[i-1]})$ is correlated with $(\tilde{r}_{[i+1]} - \tilde{r}_{[i]})$. Therefore, the proof technique used in Proposition 1 cannot be extended to this problem.

Motivated by the intractability of the model (38) and successful application of the LDR in the robust satisficing scheduling with deterministic release time, LDR has been adopted to provide an approximate reformulation. Because the release time is also a random variable, the decision variable $\boldsymbol{t}$ is assumed to belong to the following set,

$$
\mathscr{L}^{2n,n} = \left\{ \boldsymbol{t} \in \mathscr{R}^{2n,n} \,\middle|\, \begin{array}{l} \exists \boldsymbol{t}^0, \boldsymbol{t}_i^1, \boldsymbol{t}_i^2 \in \mathbb{R}^n, \forall i \in \mathscr{N} \\ \boldsymbol{t}(\boldsymbol{p}, \boldsymbol{r}) = \boldsymbol{t}^0 + \sum_{i=1}^{n} \left(\boldsymbol{t}_i^1 p_i + \boldsymbol{t}_i^2 r_i \right) \end{array} \right\},
$$

where $\mathscr{R}^{2n,n}$ is the space of all measurable functions from the space $\mathbb{R}^{2n}$ to the space $\mathbb{R}^n$. With the LDR, we consider the following approximate robust satisficing scheduling problem with random release time,

$$
\begin{aligned}
(\overline{\text{RSR}})\ \min\ & \kappa \\
\text{s.t. } & \mathbb{E}_{\mathbb{Q}}\left[\sum_{i=1}^{n}(t_i(\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}})+\tilde{p}_i)\right] - \tau \le \kappa \mathbb{E}_{\mathbb{Q}}\left(\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_1+\|\tilde{\boldsymbol{r}}-\hat{\boldsymbol{r}}\|_1\right), && \forall \mathbb{Q}\in\mathscr{F}_r, && (40)\\
& t_i(\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}}) \ge \tilde{r}_{[i]}, && \forall (\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}})\in\mathscr{D}^p\times\mathscr{D}^r, i\in\mathscr{N}, && (41)\\
& t_i(\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}}) - t_{i-1}(\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}}) \ge \tilde{p}_{[i-1]}, && \forall (\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}})\in\mathscr{D}^p\times\mathscr{D}^r, i\in\mathscr{N}, && (42)\\
& t_i(\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}}) = t_i^0 + \sum_{j=1}^{n}(t_{ij}^1\tilde{p}_j + t_{ij}^2\tilde{r}_j), && \forall i\in\mathscr{N}, && (43)\\
& \boldsymbol{X}\in\mathscr{X}, \\
& \kappa \ge 0, \\
& \boldsymbol{t}^0, \boldsymbol{t}_i^1, \boldsymbol{t}_i^2 \in \mathbb{R}^n, && \forall i\in\mathscr{N}.
\end{aligned}
$$

In this formulation, the constraint (40) still requires that the expected violation (i.e., $\mathbb{E}_{\mathbb{Q}}\left[\sum_{i=1}^{n}(t_i(\tilde{\boldsymbol{p}},\tilde{\boldsymbol{r}})+\tilde{p}_i)\right]-\tau$) is restricted by the product of the magnitude κ and the statistical distance $\mathbb{E}_{\mathbb{Q}}\left(\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_1+\|\tilde{\boldsymbol{r}}-\hat{\boldsymbol{r}}\|_1\right)$ for all possible distributions in the set $\mathscr{F}_r$. Constraints (41) and (42) enforce that the start time of the job at the ith position is greater than the release time of the job assigned to the ith position and the completion time of the job at the $(i-1)$-th position. The constraint (43) is the expression of the linear decision rule. The objective of the robust satisficing model minimizes the magnitude of expected target violation that could occur under any distribution. Next, we present an equivalent reformulation for the model $(\overline{\text{RSR}})$.

Theorem 4. *The robust satisficing model* $(\overline{\text{RSR}})$ *is equivalent to the following problem,*

$$
\begin{aligned}
(\text{RSRA})\ \min & \kappa \\
\text{s.t. } & \frac{1}{M}\sum_{i=1}^{M}\theta_m \le \tau, \\
& \theta_m \ge \sum_{i=1}^{n} t_i^0 - \boldsymbol{\zeta}_m^T\hat{\boldsymbol{p}}_m - \boldsymbol{\phi}_m^T\hat{\boldsymbol{r}}_m + \underline{\boldsymbol{p}}^T\boldsymbol{u}_m^p + \bar{\boldsymbol{p}}^T\boldsymbol{v}_m^p + \underline{\boldsymbol{r}}^T\boldsymbol{u}_m^r + \bar{\boldsymbol{r}}^T\boldsymbol{v}_m^r, && \forall m\in\mathscr{M}, && (44)\\
& u_{mj}^p + v_{mj}^p - \zeta_{mj} = \sum_{i=1}^{n} t_{ij}^1 + 1, && \forall j\in\mathscr{N}, m\in\mathscr{M}, && (45)\\
& u_{mj}^r + v_{mj}^r - \phi_{mj} = \sum_{i=1}^{n} t_{ij}^2, && \forall j\in\mathscr{N}, m\in\mathscr{M}, && (46)\\
& \boldsymbol{u}_m^p \le 0, \boldsymbol{u}_m^r \le 0, && \forall m\in\mathscr{M}, && (47)\\
& \boldsymbol{v}_m^p \ge 0, \boldsymbol{v}_m^r \ge 0, && \forall m\in\mathscr{M}, && (48)\\
& \begin{bmatrix}\kappa\\ \boldsymbol{\zeta}_m\\ \boldsymbol{\phi}_m\end{bmatrix} \succeq_\infty 0, && \forall m\in\mathscr{M}, && (49)\\
& t_i^0 \ge \sum_{j=1}^{n}\left(\omega_{ij}^{mr}\underline{r}_j + \psi_{ij}^{mr}\bar{r}_j + \omega_{ij}^{mp}\underline{p}_j + \psi_{ij}^{mp}\bar{p}_j\right), && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (50)\\
& \omega_{ij}^{mr} + \psi_{ij}^{mr} = x_{ij} - t_{ij}^2, && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (51)\\
& \omega_{ij}^{mp} + \psi_{ij}^{mp} = -t_{ij}^1, && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (52)\\
& \omega_{ij}^{mr} \le 0, \omega_{ij}^{mp} \le 0, \psi_{ij}^{mr} \ge 0, \psi_{ij}^{mp} \ge 0, && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (53)\\
& t_i^0 - t_{i-1}^0 \ge \sum_{j=1}^{n}\left(\phi_{ij}^{mr}\underline{r}_j + \pi_{ij}^{mr}\bar{r}_j + \phi_{ij}^{mp}\underline{p}_j + \pi_{ij}^{mp}\bar{p}_j\right), && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (54)\\
& \phi_{ij}^{mr} + \pi_{ij}^{mr} = t_{i-1,j}^2 - t_{ij}^2, && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (55)\\
& \phi_{ij}^{mp} + \pi_{ij}^{mp} = x_{i-1,j} + t_{i-1,j}^1 - t_{ij}^1, && \forall i,j\in\mathscr{N}, m\in\mathscr{M}, && (56)\\
& \phi_{ij}^{mr} \le 0, \phi_{ij}^{mp} \le 0, \pi_{ij}^{mr} \ge 0, \pi_{ij}^{mp} \ge 0, && \forall i,j\in\mathscr{N}, m\in\mathscr{M}. && (57)
\end{aligned}
$$

As a mixed-integer linear program, the model (RSRA) can be solved by state-of-the-art commercial solvers, such as CPLEX and GUROBI. Note that the model (RSRA) can also be solved based on the Algorithm 1, where the model (RSI) in Step 4 is replaced with the following model

$$
\min \frac{1}{M}\sum_{m=1}^{M}\theta_m
$$

s.t (44) – (57).

5. Numerical experiments

In this section, we compare the robust satisficing scheduling model proposed in this paper with existing approaches. We first consider the case where the release time is assumed to be deterministic in Section 5.3, and then, we consider the random release time case in Section 5.4. All the numerical experiments were conducted on a MacBook Pro with 16 GB RAM, and all models were coded in Python 3.9 and solved with the GUROBI 10.0 solver.

5.1. Benchmarks and model setup

When the release time is deterministic, we consider four models; namely: our proposed robust satisficing model (RSD), a deterministic model (D), an empirical scheduling model (ESD), and a moment-based distributionally robust scheduling model (MD) proposed by Lu and Pei (2022). The latter three models serve as benchmarks. All the models mentioned above have access to the same historical dataset $\{\hat{\boldsymbol{p}}_m\}_{m=1}^{M}$ containing data points that are i.i.d. drawn from an underlying distribution to be specified later. The Models and setups used are explained below. For our robust satisficing model, we set the target parameter τ as $\tau = \alpha Z_0$, where Z_0 is the objective value of model (ESD) and $\alpha \ge 1$ is chosen from a candidate set $\mathscr{U}_\alpha$ specified later. For more choices of the target, we refer readers to the footnote.[2] The deterministic model

[2] **Choice of target.** The target can also be set as $\tau = Z_0 + \alpha\frac{\hat{\sigma}(\text{ESD})}{\sqrt{M-1}}$, where Z_0 represents the objective value of the empirical optimization model (ESD), and $\hat{\sigma}(\text{ESD})$ is the standard deviation of the in-sample total completion time. The hyper-parameter α requires determination through a cross-validation approach. Specifically, given the solution $\boldsymbol{X}^{\text{ESD}}$ of the empirical optimization model, the total completion time is computed for each historical data point. Subsequently, the in-sample standard deviation $\hat{\sigma}(\text{ESD})$ is calculated over all obtained total completion times.

is formulated as

$$\text{(D)}\quad \min_{\boldsymbol{X},\boldsymbol{t}} \sum_{i=1}^{n}\left(t_i+\sum_{j=1}^{n}x_{ij}\hat{\mu}_j^p\right)$$
$$\text{s.t. } t_i \geq \sum_{j=1}^{n} x_{ij} r_j, \qquad \forall i \in \mathcal{N},$$
$$t_i \geq t_{i-1} + \sum_{j=1}^{n} x_{i-1,j}\hat{\mu}_j^p, \qquad \forall i \in \mathcal{N},$$
$$\boldsymbol{X} \in \mathcal{X},$$

where $\hat{\boldsymbol{\mu}}^p := \frac{1}{M}\sum_{m=1}^{M}\hat{\boldsymbol{p}}_m$ is the mean value of the historical processing time. The empirical scheduling model (ESD) is introduced in Section 3. The moment-based distributionally robust scheduling model is

$$\text{(MD)}\quad \min_{\boldsymbol{X},\boldsymbol{\eta}} \sum_{i=1}^{n+1}\eta_i + \sum_{i=1}^{n-1}\sum_{j=1}^{n}(n-i)\hat{\mu}_j^p x_{ij} + \sum_{j=1}^{n}\bar{p}_j$$
$$\text{s.t. } \sum_{i=1}^{k}\eta_i \geq \sum_{i=2}^{\min\{k,n\}}\sum_{j=1}^{n}(r_j x_{i-1,j} - r_j x_{i,j})\pi_{i,k}, \quad \forall k=1,\dots,n+1,$$
$$\sum_{i=l}^{k}\eta_i \geq \sum_{i=l}^{\min\{k,n\}}\sum_{j=1}^{n}(r_j x_{i-1,j} - r_j x_{i,j})\pi_{i,k}, \quad \forall l=2,\dots,n+1,$$
$$k=l,\dots,n+1,$$
$$\boldsymbol{X} \in \mathcal{X},$$

which can be found in Lu and Pei (2022, Theorem 4). We denote their respective solutions by $X^{\mathrm{RSDE}}(\alpha)$, X^{D}, X^{ESD} and X^{MD}.

When the release time is random, we consider three models: namely, the robust satisficing model (RSR); a deterministic model (D) whose release time is the sample mean; and an empirical scheduling model (ESR). We no longer consider (MD) because it is unsuitable for the random release time case.

5.2. Experiment setup

Data Generation. Our experimental setups are like those in Lu and Pei (2022). Specifically, we assume that the mean processing time $\mu_j^p, \forall j \in \mathcal{N}$ is drawn from $[40, 50]$. The release time $r_j, \forall j \in \mathcal{N}$ is randomly generated from $[0, \delta_r \sum_{j=1}^{n}\mu_j^p]$, where the release time range coefficient $\delta_r \in \mathcal{U}_r$ controls the range of the release time. For job $j, \forall j \in \mathcal{N}$, the mean absolute deviation σ_j^p is chosen from $[0, \delta_\sigma^p \mu_j^p]$, where the coefficient $\delta_\sigma^p \in \mathcal{U}_\sigma^p$ controls the variation of the processing time. In the numerical experiments, we assume that the underlying distribution of the processing time $\tilde{p}_j, \forall j \in \mathcal{N}$ follows a normal distribution with mean μ_j^p and mean absolute deviation σ_j^p. The lower and upper bounds of the processing time are denoted by $\underline{\boldsymbol{p}}$ and $\bar{\boldsymbol{p}}$, respectively, which correspond to 10% and 90% percentiles of the distribution. We also assume that the historical dataset $\{\hat{\boldsymbol{p}}_m\}_{m=1}^{M}$ includes $M = 10$ data points and that the test dataset $\{\hat{\boldsymbol{p}}_m\}_{m=1}^{\overline{M}}$ contains $\overline{M} = 10{,}000$ data points. All the data points are generated from the underlying normal distribution truncated to an interval $[\underline{\boldsymbol{p}}, \bar{\boldsymbol{p}}]$.

Performance Metrics. Following Liu et al. (2020b) and Lu and Pei (2022), we examine out-of-sample performances, i.e., the performances on a test dataset of the solutions that are obtained from solving models built upon a historical data set. Of particular interest are the averaged value and its 95% percentile; the former reflects the performance on average, while the latter accounts for extreme cases.

Specifically, for the model $\Pi \in \{\mathrm{D}, \mathrm{ESD}, \mathrm{MD}, \mathrm{RSDE}(\alpha)\}$, the average and 95% percentile of the total completion time are denoted by AVG^{Π} and PT_{95}^{Π}, respectively. We always treat performances of the deterministic model (D) as 1, and report the relatives against it, i.e.,

$$\mathrm{R_A}(\Pi) = \frac{\mathrm{AVG}^{\Pi}}{\mathrm{AVG}^{D}} \times 100\%, \qquad \forall \Pi \in \{\mathrm{D}, \mathrm{ESD}, \mathrm{MD}, \mathrm{RSDE}(\alpha)\},$$
$$\mathrm{R}^{\mathrm{PT}_{95}}(\Pi) = \frac{\mathrm{PT}_{95}^{\Pi}}{\mathrm{PT}_{95}^{\mathrm{D}}} \times 100\%, \qquad \forall \Pi \in \{\mathrm{D}, \mathrm{ESD}, \mathrm{MD}, \mathrm{RSDE}(\alpha)\}.$$

Table 1
The out-of-sample performance under different approaches.

Average gaps			95% percentile gaps		
$\mathrm{R_A}$(MD)	$\mathrm{R_A}$(ESD)	$\mathrm{R_A}$(RSDE(1.2))	$\mathrm{R_{95}^{PT}}$(MD)	$\mathrm{R_{95}^{PT}}$(ESD)	$\mathrm{R_{95}^{PT}}$(RSDE(1.2))
99.88%	99.30%	99.89%	99.63%	99.56%	93.10%

When the release time is random, we can define the performance metrics in an analogous way. All the experiments were repeated 20 times and the average of the 20 experiments was reported.

5.3. Experiments with deterministic release time

Experiments under a given set of parameters In this part, we conduct numerical experiments with a specific parameter setting. We assumed that the number of jobs is $n = 20$, the release time range coefficient $\delta_r = 0.15$, and the processing time MAD coefficient $\delta_\sigma^p = 1.5$. For the robust satisficing model, we assumed $\tau = 1.2Z_0$. The average and 95% percentile performance under different models are reported in Table 1,

Our robust satisficing model can achieve much lower 95th percentile gaps with almost the same average gaps. This observation demonstrates that the robust satisficing model may be a good choice for decision-makers who prefer a scheduling sequence that performs well in worst-case scenarios

Experiments under various $(\alpha, \delta_r, \delta_\sigma^p)$

In this part, we conduct numerical experiments to evaluate the out-of-sample performance under different combinations of the target coefficient α, the release time range coefficient δ_r, and the mean absolute deviation coefficient δ_σ^p. To vary the target, we assume that α is chosen from the set $\mathcal{U}_\alpha = \{1 + 10^{-6}, 1.02, 1.04, 1.06, 1.08, 1.10, 1.20, 1.30, 1.40, 1.50\}$. The release time range coefficient δ_r is drawn from the set $\{0.05, 0.15, 0.3\}$, and the mean absolute deviation coefficient σ^p is taken from the set $\mathcal{U}_\sigma^p = \{0.5, 1.0, 1.5\}$. The number of jobs is set to 20, i.e., $n = 20$. The average and 95% percentile gaps under various combinations of $(\alpha, \delta_r, \delta_\sigma^p)$ are reported in Tables 2 and 3. Please note that the bold numbers represent the lowest values in the same row, and the lower the value is, the better the performance.

We can make several observations from Tables 2 and 3. First, by tuning α, robust satisficing can achieve the best average and 95% percentile gaps. Additionally, we observed that the lowest average gaps are usually obtained by setting a lower target, while the lowest 95% percentile gaps are achieved by choosing a relatively higher target. This observation can help the decision-maker set appropriate targets for stochastic scheduling problems. If the decision-maker values the worst-case performance (95% percentile gaps), a relatively higher target may be a good option. In contrast, a relatively lower target is recommended if the decision-maker values the average performance.

Second, it seems that the average and 95% percentile gaps obtained from the robust satisficing, empirical, and moment-based DRO models decrease as δ_σ^p increases. This indicates that the deterministic model is not effective in providing a solution when there is a large uncertainty in the processing time. As δ_σ^p increases, the decision-maker should set a higher target to obtain the lowest 95% percentile gap. For example, with $\delta_r = 0.05$, as δ_σ^p increases from 0.5 to 1.5, the lowest 95% percentile gap obtained from the robust satisficing model decreases from 97.72% to 91.95%, and the corresponding α increases from 1.08 to 1.3.

Finally, we also observed that the lowest average gap obtained from the robust satisficing model decreases as the release time range increases. However, the lowest 95% percentile gaps obtained from robust satisficing decrease with δ_r. For example, with $\delta_\sigma^p = 1.5$, the lowest average gap obtained from robust satisficing decreases from 99.61% to 98.87%, while the lowest 95% percentile gap increases from 91.95% to 95.05% and corresponding α decreases from 1.3 to 1.2.

Table 2
The average gaps under different parameters.

Model		$R_A(\text{RSDE}(\alpha))$										$R_A(\text{ESD})$	$R_A(\text{MD})$
		$1+10^{-6}$	1.02	1.04	1.06	1.08	1.1	1.2	1.3	1.4	1.5		
$\delta_r = 0.05$	$\delta_\sigma^p = 0.5$	**99.98%**	100.37%	100.92%	101.16%	101.34%	101.39%	101.86%	103.79%	107.43%	107.43%	100.24%	100.69%
	$\delta_\sigma^p = 1.0$	**99.85%**	100.06%	100.37%	100.66%	100.85%	100.99%	101.31%	101.41%	102.00%	103.34%	99.91%	100.59%
	$\delta_\sigma^p = 1.5$	99.74%	**99.61%**	99.90%	100.02%	100.08%	100.21%	100.37%	100.34%	100.39%	100.75%	99.83%	100.75%
$\delta_r = 0.15$	$\delta_\sigma^p = 0.5$	**99.91%**	100.22%	100.55%	101.00%	101.35%	101.41%	102.43%	105.18%	113.03%	115.42%	100.14%	101.77%
	$\delta_\sigma^p = 1.0$	**99.46%**	**99.46%**	99.83%	100.10%	100.46%	100.72%	101.52%	102.17%	104.35%	107.83%	99.63%	100.81%
	$\delta_\sigma^p = 1.5$	**99.00%**	99.11%	99.30%	99.73%	99.91%	99.92%	99.89%	100.40%	101.36%	102.61%	99.30%	99.88%
$\delta_r = 0.30$	$\delta_\sigma^p = 0.5$	**99.70%**	100.12%	100.83%	101.17%	101.58%	101.65%	103.62%	105.36%	109.24%	110.37%	99.93%	101.81%
	$\delta_\sigma^p = 1.0$	**99.05%**	99.40%	99.99%	100.28%	100.61%	100.72%	101.85%	102.82%	106.23%	109.74%	99.32%	100.88%
	$\delta_\sigma^p = 1.5$	**98.87%**	98.99%	99.47%	99.98%	100.74%	101.22%	102.17%	102.72%	103.89%	105.10%	98.91%	99.82%

Table 3
The 95% percentile gaps under different parameters.

Model		$R_{95}^{PT}(\text{RSDE}(\alpha))$										$R_{95}^{PT}(\text{ESD})$	$R_{95}^{PT}(\text{MD})$
		$1+10^{-6}$	1.02	1.04	1.06	1.08	1.1	1.2	1.3	1.4	1.5		
$\delta_r = 0.05$	$\delta_\sigma^p = 0.5$	99.77%	98.61%	98.13%	97.84%	97.72%	**97.66%**	98.61%	102.09%	107.35%	107.35%	100.57%	100.63%
	$\delta_\sigma^p = 1.0$	99.65%	97.85%	96.59%	95.90%	95.29%	94.85%	**94.00%**	94.50%	96.63%	98.53%	100.00%	100.25%
	$\delta_\sigma^p = 1.5$	99.75%	97.39%	95.96%	94.97%	94.27%	93.74%	92.26%	**91.95%**	92.13%	93.29%	99.85%	100.52%
$\delta_r = 0.15$	$\delta_\sigma^p = 0.5$	99.75%	98.74%	98.20%	97.97%	97.96%	**97.89%**	99.13%	102.45%	111.06%	113.96%	100.29%	101.32%
	$\delta_\sigma^p = 1.0$	99.60%	97.92%	96.80%	96.08%	95.66%	95.35%	**95.07%**	95.61%	99.31%	102.57%	99.82%	100.45%
	$\delta_\sigma^p = 1.5$	99.24%	97.44%	96.27%	95.41%	94.80%	94.39%	**93.10%**	93.40%	94.61%	96.01%	99.56%	99.63%
$\delta_r = 0.30$	$\delta_\sigma^p = 0.5$	99.67%	98.99%	98.66%	**98.54%**	98.60%	98.60%	101.23%	104.28%	108.14%	109.35%	100.05%	102.41%
	$\delta_\sigma^p = 1.0$	99.55%	98.30%	97.25%	96.65%	96.26%	**96.00%**	96.33%	97.91%	101.72%	105.71%	99.92%	102.40%
	$\delta_\sigma^p = 1.5$	99.12%	97.95%	96.95%	96.25%	95.88%	95.65%	**95.05%**	95.43%	97.40%	99.83%	99.49%	100.90%

Comparison between the exact model (RSDE) **and the approximate model** (RSDA)

In this part, we conduct numerical experiments to compare the performance between model (RSDE) and (RSDA), an approximation model proposed in Section 3.3. In this simulation study, we randomly draw δ_r and δ_σ^p from the intervals $[0.05, 0.3]$ and $[0.2, 2.0]$, respectively. The number of jobs is chosen from the set $\{10, 20, 30\}$ and the set $\mathcal{U}_\alpha$ is the same as in Table 2. The model (RSDA) is solved by using the Algorithm 1. Solutions $\boldsymbol{X}^{\text{RSDA}}$ and $\boldsymbol{X}^{\text{RSDE}}$ are evaluated on the test data and we report the ratios $R_A(\text{RSDA})/R_A(\text{RSDE})$ and $R_{95}^{PT}(\text{RSDA})/R_{95}^{PT}(\text{RSDE})$ in Fig. 1.

We see that $R_A(\text{RSDA})/R_A(\text{RSDE})$ and $R_{95}^{PT}(\text{RSDA})/R_{95}^{PT}(\text{RSDE})$ are very close to 1, suggesting that (RSDA) is a good approximation. The expectation of a high-quality approximation is well-founded, considering the proven effectiveness of LDR in addressing operational management problems over the past decades. Theoretical foundations support this, as exemplified in special cases such as 1-dimensional decision variables, where LDR incurs no approximation error, as highlighted in Bertsimas et al. (2019, Theorem 4) and Hao et al. (2020, Proposition 2). While our models may not strictly conform to these special cases, we can reasonably anticipate a low approximation error from LDR. Empirically, the consistent ability of LDR to provide accurate approximations is evident in the literature, as demonstrated in studies such as Zhen et al. (2018), Bertsimas et al. (2022), Sim and Chen (2024). In the context of our robust scheduling problems, our numerical experiments further validate that solutions obtained from (RSDA) and (RSDE) are indeed nearly identical in most cases.

Moreover, the time to solve (RSDA) is significantly less than the time to solve (RSDE) (see Table 4), especially when the number of jobs is large. Remarkably, when $n = 30$ and $\alpha = 1.04$, solving (RSDE) needs roughly $1080s$, while it only takes $53s$ to solve (RSDA). Given the competitive performance and computational advantage of (RSDA), we thus recommend (RSDA) over (RSDE).

Experiments under various numbers of jobs

We further investigate the impact of the number of jobs by running simulations under various $n \in \{10, 20, 30, 40, 50, 60, 70, 80\}$. Other settings remain unchanged, i.e., δ_r are uniformly drawn from $[0.05, 0.3]$, δ_σ^p from $[0.2, 2.0]$. The target is set to $\tau = 1.08Z_0$ as when $\alpha = 1.08$, robust satisficing models can achieve the best average performance while also possessing a great ability to resist uncertainty (see Tables 2 and 3). The simulation results are reported in Table 5. According to the table, our approximate robust satisficing model outperforms the (MD) and (ESD) models in both average and 95% percentile gaps. We also observe that the average gaps obtained from the robust satisficing and empirical models are nearly identical, but robust satisficing can achieve a much lower 95% percentile gap. For instance, with $n = 20$, the average gaps obtained from empirical and robust satisficing are 99.40% and 98.58%, respectively. However, the 95% percentile gap acquired from robust satisficing, 93.10%, is much lower than the 99.58% acquired from the empirical model. Next, we observe that the CPU time needed to solve the robust satisficing model increases as the number of jobs increases. Although the CPU time required to solve our robust satisficing model is longer than that of the empirical model, the maximum time is still less than half an hour (1800 s).

5.4. Experiments with random release time

As release time is assumed to be random, we slightly modify the underlying distribution to reflect such a change. Specifically, for each job j, we assume that the processing time p_j and the release time r_j follow a two-dimensional joint normal distribution with mean $[\mu_j^p, \mu_j^r]^T$, and the covariance matrix is $\Sigma^j := \begin{bmatrix} \sigma_{p_j}^2 & cv{\cdot}\sigma_{p_j}\sigma_{r_j} \\ cv{\cdot}\sigma_{p_j}\sigma_{r_j} & \sigma_{r_j}^2 \end{bmatrix}$ with σ_{p_j}, σ_{p_j} being standard deviations and cv is the coefficient of correlation. With this distribution, we denote the 10% and 90% percentiles of processing times as the lower bound and the upper bound on the processing and release time, respectively.

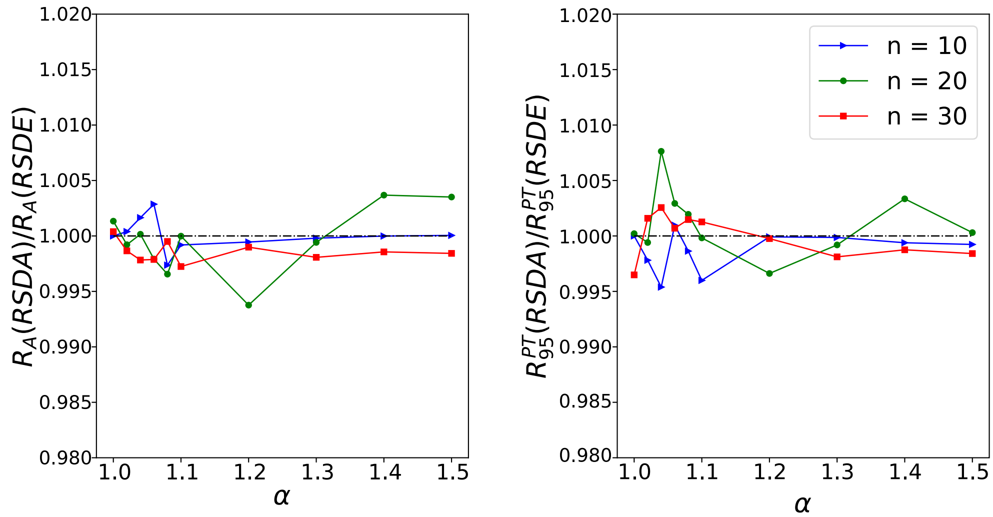

Fig. 1. Out-of-sample performance comparison.

Table 4
CPU time: s.

Model		α									
		$1+10^{-6}$	1.02	1.04	1.06	1.08	1.10	1.20	1.30	1.40	1.50
$n = 10$	RSDA(α)	0.27	0.29	0.3	0.33	0.36	0.4	0.48	0.44	0.34	0.3
	RSDE(α)	0.46	1.88	2.02	2.17	2.2	2.3	1.47	1.4	1.44	1.19
$n = 20$	RSDA(α)	3.84	4.27	6.2	8.42	9.36	10.18	9.87	7.17	5.03	4.46
	RSDE(α)	13.66	38.27	49.19	57.95	50.19	54.15	35.5	25.3	19.32	16.88
$n = 30$	RSDA(α)	14.79	23.62	52.93	62.98	64.41	58.59	43.76	35.4	23.33	17.6
	RSDE(α)	312.21	821.75	1081.4	795.69	932.86	1007.66	592.55	381.89	311.19	285.59

Table 5
The gaps under different the number of jobs.

Model	(MD)			(ESD)			(RSDA)		
	R_A(MD)	R_{95}^{PT}(MD)	Time	R_A(ESD)	R_{95}^{PT}(ESD)	Time	R_A(RSDA(1.08))	R_{95}^{PT}(RSDA(1.08))	Time
$n = 10$	100.40%	101.32%	0.07	99.64%	100.03%	0.01	99.25%	94.68%	0.36
$n = 20$	99.81%	99.46%	9.83	99.40%	99.58%	0.05	98.58%	93.10%	7.63
$n = 30$	99.79%	99.94%	1319.22	99.30%	99.54%	0.13	98.69%	94.44%	40.05
$n = 40$	100.05%	100.09%	1588.50	99.24%	99.41%	0.30	98.51%	95.40%	153.64
$n = 50$	–	–	–	99.42%	99.52%	0.51	98.81%	95.45%	311.71
$n = 60$	–	–	–	99.48%	99.76%	0.86	99.13%	96.06%	613.30
$n = 70$	–	–	–	99.53%	99.74%	1.27	99.13%	96.13%	1263.27
$n = 80$	–	–	–	99.35%	99.57%	1.95	98.76%	95.83%	1742.88

Note. "-" represents that the corresponding model cannot be solved to optimal within 1800 s.

Experiments under various $(\alpha, \delta_r, \delta_\sigma^r, \delta_\sigma^p)$

In this part, we conduct numerical experiments under various parameter combinations that are the same as in Table 2. Additionally, δ_σ^r is chosen from the set $\{0.5, 1.0, 1.5\}$. The average and 95% percentile gaps are reported in Figs. 2 and 3. We can draw several observations from the figures. First, the robust satisficing model usually obtains the lowest average performance by setting a relatively lower target, while the best worst-case performance (95% percentile gaps) is achieved by choosing a relatively higher target. A similar observation can be found in .

Second, as δ_r increases, $\mathrm{R_A}(\Pi)$ and $\mathrm{R}_{95}^{\mathrm{PT}}(\Pi)$, $\forall \Pi \in \{\mathrm{ESR}, \mathrm{RSRA}(\alpha)\}$ decline in most subfigures. This observation demonstrates that the deterministic model performs poorly if the jobs' release times are generated over a larger range. Additionally, as the uncertainty of the release time increases, the deterministic model performs much worse with a wider release time range. For example, given $\delta_\sigma^p = 0.5$, if $\delta_\sigma^r = 0.5$ and the release time range coefficient δ_r increases from 0.05 to 0.3, the average empirical gaps $\mathrm{R_A}$(ESR) decrease only from 99.9% to 98.1%. However, if $\delta_\sigma^r = 1.5$, $\mathrm{R_A}$(ESR) decreases from 99.7% to 88.9%, which represents a substantial decrease.

Third, we observe that both the empirical model and the robust satisficing model can achieve better performances as the uncertainty of the release time increases. For instance, given $\delta_r = 0.3$ and $\delta_\sigma^p = 1.5$, by tuning δ_σ^r from 0.5 to 1.5, we can see that $\{\mathrm{R_A}(\mathrm{ESR}), \mathrm{R}_{95}^{\mathrm{PT}}(\mathrm{ESR})\}$ decreases from $\{97.3\%, 98.4\%\}$ to $\{89.7\%, 85.3\%\}$, and $\{\mathrm{R_A}(\mathrm{RSRA}(1.06)), \mathrm{R}_{95}^{\mathrm{PT}}(\mathrm{RSRA}(1.06))\}$ declines from $\{97.7\%, 95.9\%\}$ to $\{89.1\%, 84.6\%\}$. Furthermore, as δ_σ^r increases, the optimal α that helps the robust satisficing to obtain the lowest average and 95% percentile gaps does not change significantly. It is useful for the decision-maker to set the target in the stochastic scheduling problem with random release time.

Finally, as δ_σ^p increases (i.e., as the uncertainty of the processing time increases), it is evident that the robust satisficing model achieve a lower 95% percentile gap. For example, given $\delta_r = 0.05$ and $\delta_\sigma^r = 0.5$, the average and 95% percentile gaps obtained from the empirical model are close to 1 when δ_σ^p is changed from 0.5 to 1.5. In contrast, although the lowest average gap obtained from robust satisficing is close to 1,

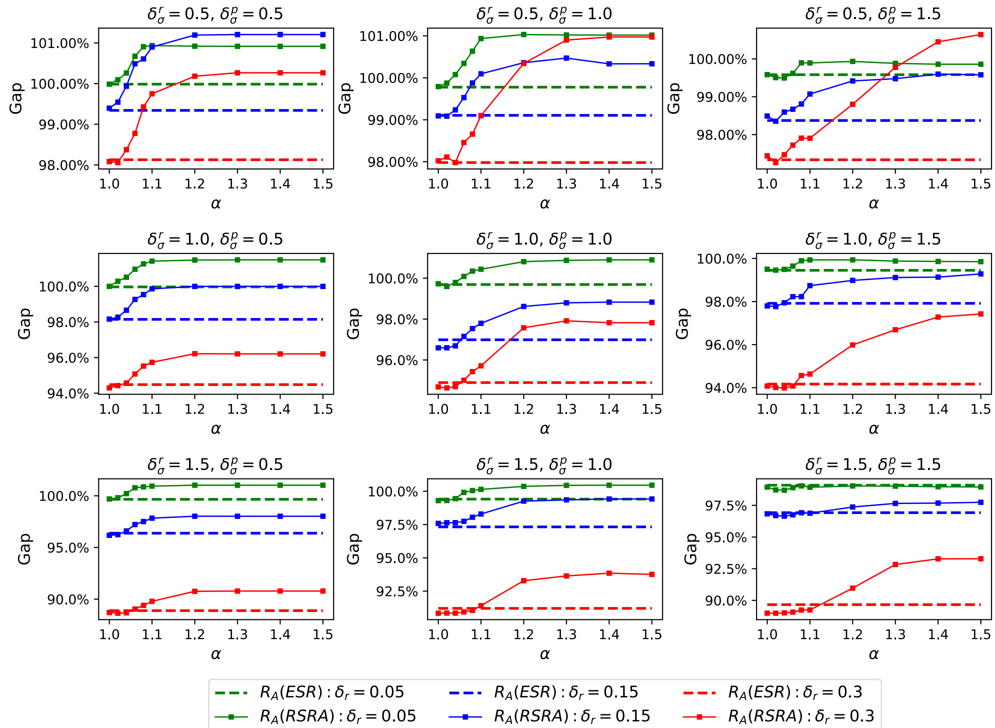

Fig. 2. The average gaps under different combinations of $(\alpha, \delta_r, \delta_\sigma^r, \delta_\sigma^p)$.

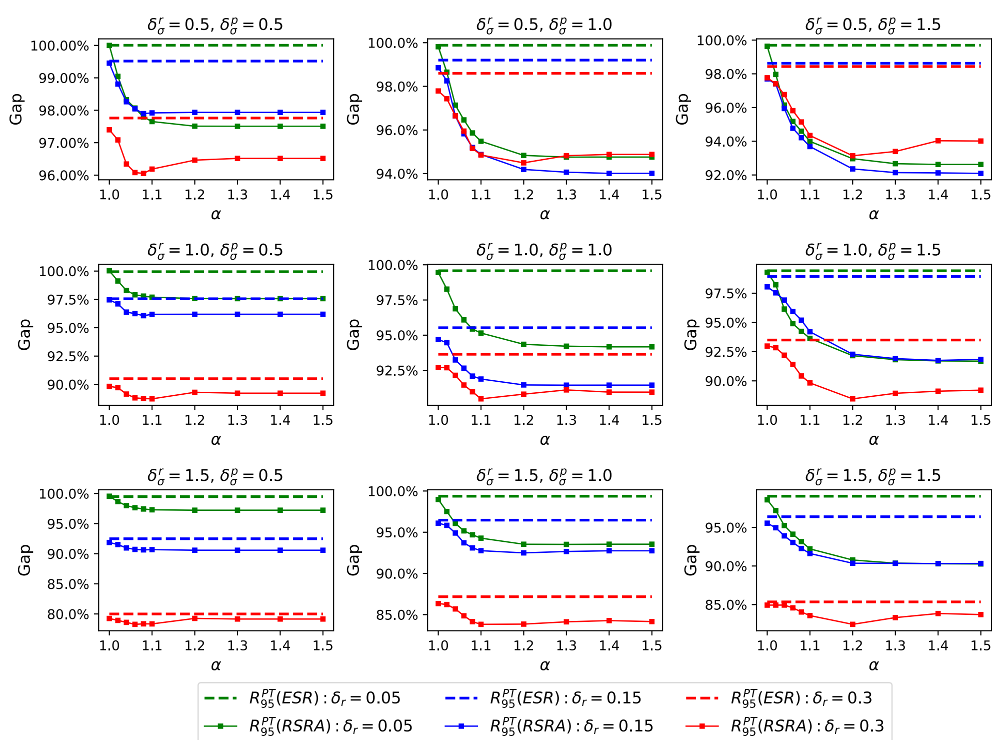

Fig. 3. The 95% percentile gaps under different combinations of $(\alpha, \delta_r, \delta_\sigma^r, \delta_\sigma^p)$.

the lowest 95% percentile gap is reduced from 97.5% to 92.6%, and the corresponding α increases.

Experiments under different correlation conditions between processing and release time

In this part, we examine the performances of the models by varying the correlation between the release and the processing time. All the experimental settings remain unchanged for the coefficient of correlation $cv \in \{-0.6, -0.4, -0.2, 0.0, 0.2, 0.4, 0.6\}$. Tables 6 and 7 summarize the average and 95% percentile gaps.

As we can see, the robust satisficing and empirical model can obtain better out-of-sample performance than that obtained from the deterministic model as cv increases if there exists a positive correlation between release and processing time (i.e., $cv > 0$). In contrast, the advantage of the robust satisficing and empirical model decreases as

Table 6
The average gaps under different correlations.

Model		R_A(RSRA(α))										R_A(ESR)
		$1+10^{-6}$	1.02	1.04	1.06	1.08	1.1	1.2	1.3	1.4	1.5	
	−0.6	98.74%	**98.70%**	98.90%	99.02%	99.48%	99.63%	100.24%	100.66%	100.66%	100.66%	98.38%
	−0.4	98.30%	**98.22%**	98.35%	98.66%	98.93%	99.14%	99.56%	99.80%	99.92%	99.92%	98.17%
	−0.2	**97.69%**	97.75%	97.95%	98.42%	98.74%	98.95%	99.60%	99.65%	99.86%	99.86%	97.81%
cv	0.0	98.32%	**98.28%**	98.39%	98.70%	99.07%	99.31%	100.15%	100.35%	100.29%	100.29%	98.41%
	0.2	**98.45%**	98.55%	98.57%	98.88%	99.28%	99.47%	99.97%	100.18%	100.37%	100.38%	98.63%
	0.4	**97.77%**	97.78%	98.07%	98.34%	98.49%	98.90%	99.38%	99.41%	99.38%	99.38%	97.94%
	0.6	**97.21%**	97.27%	97.35%	97.66%	98.06%	98.23%	99.17%	99.33%	99.18%	99.18%	97.47%

Table 7
The 95% percentile gaps under different correlations.

Model		R_{95}^{PT}(RSRA(α))										R_{95}^{PT}(ESR)
		$1+10^{-6}$	1.02	1.04	1.06	1.08	1.1	1.2	1.3	1.4	1.5	
	−0.6	98.60%	98.84%	97.50%	96.41%	95.93%	95.45%	**94.69%**	94.85%	94.83%	94.83%	99.09%
	−0.4	98.12%	97.78%	96.68%	95.66%	95.29%	94.68%	94.06%	**94.04%**	94.24%	94.24%	98.68%
	−0.2	96.88%	96.40%	95.31%	94.48%	94.04%	93.68%	93.20%	**93.13%**	93.26%	93.26%	97.64%
cv	0.0	98.00%	97.51%	96.51%	95.48%	94.93%	94.42%	93.88%	93.96%	**93.84%**	93.84%	98.74%
	0.2	97.77%	97.38%	96.01%	95.23%	94.78%	94.33%	**93.57%**	93.65%	93.85%	93.84%	98.51%
	0.4	97.43%	97.03%	95.66%	94.94%	94.27%	94.05%	93.37%	93.24%	**93.21%**	93.21%	98.03%
	0.6	96.47%	96.08%	94.54%	93.56%	92.90%	92.48%	91.95%	92.02%	**91.90%**	91.90%	97.25%

Table 8
The numerical results under different numbers of jobs.

Model	(ESR)			(RSRA)		
	R_A(ESR)	R_{95}^{PT}(ESR)	Time	R_A(RSRA(1.08))	R_{95}^{PT}(RSRA(1.08))	Time
$n=10$	100.03%	99.98%	0.01	100.26%	94.01%	0.54
$n=20$	98.84%	99.32%	0.04	99.51%	95.18%	8.05
$n=30$	97.15%	96.08%	0.26	97.59%	92.78%	54.34
$n=40$	98.91%	99.29%	0.55	99.65%	95.77%	172.12
$n=50$	97.48%	97.03%	1.23	98.24%	94.69%	605.57
$n=60$	95.86%	95.21%	2.56	96.05%	91.97%	1232.84
$n=70$	96.12%	94.91%	4.52	96.84%	92.82%	2370.40
$n=80$	97.10%	95.97%	6.88	97.66%	93.64%	3478.68

cv increases if there exists a negative correlation between release and processing time. Nevertheless, our robust satisficing can achieve the best performance under all correlation scenarios.

Experiments under different numbers of jobs

In this part, we present the numerical results for different numbers of jobs in Table 8. For this simulation study, we set the target in a robust satisficing model as $\tau = 1.08Z_0$. From Table 8, we can observe that our approximate robust satisficing model (RSRA) achieves a better 95% percentile gap compared to the empirical model. Moreover, the average performance obtained from our robust satisficing model is nearly identical to that of the empirical model. We note that the CPU time required to solve both the empirical and robust satisficing models increases with the number of jobs. Additionally, it is noteworthy that the longest CPU time required to solve the robust satisficing model is less than one hour.

6. Theoretical and managerial implications

In this section, we present several theoretical and managerial implications derived from this work. In terms of theoretical insights, we first demonstrate that the robust satisficing scheduling problem with the deterministic model can be reformulated as an equivalent mixed-integer linear program. We also highlight the difficulty of reformulating robust satisficing scheduling problems with random release times. Furthermore, in this work, we have verified that the LDR technique, when utilized, enables an approximate reformulation that performs very well within a significantly shorter CPU time. This observation suggests that the LDR technique has the potential to address more complex scheduling problems.

According to the numerical experiments, several managerial implications can be highlighted in the context of the single machine scheduling problem with random processing and release time:

- If the decision-maker prioritizes worst-case performance (95% percentile gaps), it may be advisable to set a relatively higher target. Conversely, if the decision-maker values average performance, a relatively lower target should be chosen.
- As the uncertainty in processing time increases, the decision-maker should maintain a lower target if he or she prioritizes average performance. However, if they value the worst-case performance, a higher target should be selected.
- For the single machine scheduling problem with random release time, given the same uncertainty level in the processing time, the decision-maker does not need to make significant changes to the target, regardless of whether they prioritize average performance or worst-case performance.

7. Conclusion

In this paper, we adopt target-oriented robust satisficing models to address single machine scheduling problems with release time. For the problem with deterministic release time, we derive an equivalent mixed-integer linear program and an approximate reformulation based on the linear decision rule. For the problem with random release time, we first show that it is difficult to obtain an equivalent reformulation. Then, we provide an approximate reformulation. Numerical experiments are conducted to demonstrate that our proposed robust satisficing model performs well in terms of out-of-sample performance.

We also derive several managerial implications from the numerical experiments.

Future research can be pursued in the following directions. Extending the investigation to parallel machine scheduling problems with uncertain release and processing times may be a challenging yet important research direction. Additionally, exploring different criteria for scheduling, such as maximum lateness or tardiness, may provide insights into other aspects of scheduling problems.

CRediT authorship contribution statement

Xun Zhang: Conceptualization, Formal analysis, Funding acquisition, Methodology, Visualization, Writing – original draft. **Du Chen:** Conceptualization, Formal analysis, Methodology, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgment

The authors sincerely thank the department editor, the associate editor, and anonymous referees for their valuable comments that improved the work significantly. Xun Zhang's work was supported by the National Natural Science Foundation of China [Grants 72025201 and 72331006].

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cor.2024.106642.

References

Bachtler, O., Krumke, S.O., Le, H.M., 2020. Robust single machine makespan scheduling with release date uncertainty. Oper. Res. Lett.

Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A., 2004. Adjustable robust solutions of uncertain linear programs. Math. Program. 99 (2), 351–376.

Bertsimas, D., Shtern, S., Sturt, B., 2022. Technical note—Two-stage sample robust optimization. Oper. Res. 70 (1), 624–640. http://dx.doi.org/10.1287/opre.2020.2096.

Bertsimas, D., Sim, M., Zhang, M., 2019. Adaptive distributionally robust optimization. Manage. Sci. 65 (2), 604–618.

Bianco, L., Ricciardelli, S., 1982. Scheduling of a single machine to minimize total weighted completion time subject to release dates. Nav. Res. Logist. Q. 29 (1), 151–167.

Chand, S., Traub, R., Uzsoy, R., 1996. An iterative heuristic for the single machine dynamic total completion time scheduling problem. Comput. Oper. Res. 23 (7), 641–651.

Chang, Z., Ding, J.Y., Song, S., 2019. Distributionally robust scheduling on parallel machines under moment uncertainty. European J. Oper. Res. 272 (3), 832–846.

Chang, Z., Song, S., Zhang, Y., Ding, J.Y., Zhang, R., Chiong, R., 2017. Distributionally robust single machine scheduling with risk aversion. European J. Oper. Res. 256 (1), 261–274.

Chen, X., Sim, M., Sun, P., Zhang, J., 2008. A linear decision-based approximation approach to stochastic programming. Oper. Res. 56 (2), 344–357.

Chen, Z., Sim, M., Xiong, P., 2020. Robust stochastic optimization made easy with RSOME. Manage. Sci. 66 (8), 3329–3339.

Chen, L.G., Tang, Q., 2022. Supply chain performance with target-oriented firms. Manuf. Serv. Oper. Manage. 24 (3), 1714–1732.

Chu, C., 1992a. A branch-and-bound algorithm to minimize total flow time with unequal release dates. Nav. Res. Logist. 39 (6), 859–875.

Chu, C., 1992b. Efficient heuristics to minimize total flow time with release dates. Oper. Res. Lett. 12 (5), 321–330.

Daniels, R.L., Kouvelis, P., 1995. Robust scheduling to hedge against processing time uncertainty in single-stage production. Manage. Sci. 41 (2), 363–376.

Delage, E., Ye, Y., 2010. Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper. Res. 58 (3), 595–612.

Della Croce, F., T'kindt, V., 2002. A recovering beam search algorithm for the one-machine dynamic total completion time scheduling problem. J. Oper. Res. Soc. 53 (11), 1275–1280.

El Ghaoui, L., Oustry, F., Lebret, H., 1998. Robust solutions to uncertain semidefinite programs. SIAM J. Optim. 9 (1), 33–52.

Fan, Z., Xie, X., 2022. A distributionally robust optimisation for COVID-19 testing facility territory design and capacity planning. Int. J. Prod. Res. 1–24.

Feng, X., Zheng, F., Xu, Y., 2016. Robust scheduling of a two-stage hybrid flow shop with uncertain interval processing times. Int. J. Prod. Res. 54 (12), 3706–3717.

Gao, R., 2022. Finite-sample guarantees for wasserstein distributionally robust optimization: Breaking the curse of dimensionality. Oper. Res.

Goh, J., Sim, M., 2011. Robust optimization made easy with ROME. Oper. Res. 59 (4), 973–985.

Gourgand, M., Grangeon, N., Norre, S., 2003. A contribution to the stochastic flow shop scheduling problem. European J. Oper. Res. 151 (2), 415–433.

Hall, N.G., Long, D.Z., Qi, J., Sim, M., 2015. Managing underperformance risk in project portfolio selection. Oper. Res. 63 (3), 660–675.

Hao, Z., He, L., Hu, Z., Jiang, J., 2020. Robust vehicle pre-allocation with uncertain covariates. Prod. Oper. Manage. 29 (4), 955–972.

Hariri, A., Potts, C.N., 1983. An algorithm for single machine sequencing with release dates to minimize total weighted completion time. Discrete Appl. Math. 5 (1), 99–109.

Hu, B., Jin, Q., Long, D.Z., 2022. Robust assortment revenue optimization and satisficing. Available at SSRN 4268706.

Jouglet, A., Savourey, D., Carlier, J., Baptiste, P., 2008. Dominance-based heuristics for one-machine total cost scheduling problems. European J. Oper. Res. 184 (3), 879–899.

Kleywegt, A.J., Shapiro, A., Homem-de Mello, T., 2002. The sample average approximation method for stochastic discrete optimization. SIAM J. Optim. 12 (2), 479–502.

Kuhn, D., Wiesemann, W., Georghiou, A., 2011. Primal and dual linear decision rules in stochastic and robust optimization. Math. Program. 130 (1), 177–209.

Lenstra, J.K., Kan, A.R., Brucker, P., 1977. Complexity of machine scheduling problems. In: Annals of Discrete Mathematics, vol. 1, Elsevier, pp. 343–362.

Leung, J.Y., 2004. Handbook of Scheduling: Algorithms, Models, and Performance Analysis. CRC Press.

Li, Y., Kuo, Y.H., Li, R., Shen, H., Zhang, L., 2022. A target-based distributionally robust model for the parallel machine scheduling problem. Int. J. Prod. Res. 1–22.

Liu, F., Chen, Z., Wang, S., 2023. Globalized distributionally robust counterpart. INFORMS J. Comput.

Liu, X., Chu, F., Zheng, F., Chu, C., Liu, M., 2020a. Parallel machine scheduling with stochastic release times and processing times. Int. J. Prod. Res. 1–20.

Liu, M., Liu, X., Chu, F., Zheng, F., Chu, C., 2019. Service-oriented robust parallel machine scheduling. Int. J. Prod. Res. 57 (12), 3814–3830.

Liu, M., Liu, X., Chu, F., Zheng, F., Chu, C., 2020b. Profit-oriented distributionally robust chance constrained flowshop scheduling considering credit risk. Int. J. Prod. Res. 58 (8), 2527–2549.

Liu, J., MacCarthy, B., 1991. Effective heuristics for the single machine sequencing problem with ready times. Int. J. Prod. Res. 29 (8), 1521–1533.

Long, D.Z., Sim, M., Zhou, M., 2023. Robust satisficing. Oper. Res. 71 (1), 61–82.

Lu, C.C., Lin, S.W., Ying, K.C., 2012. Robust scheduling on a single machine to minimize total flow time. Comput. Oper. Res. 39 (7), 1682–1691.

Lu, H., Pei, Z., 2022. Single machine scheduling with release dates: A distributionally robust approach. European J. Oper. Res.

Mak, H.Y., Rong, Y., Zhang, J., 2015. Appointment scheduling with limited distributional information. Manage. Sci. 61 (2), 316–334.

Mohajerin Esfahani, P., Kuhn, D., 2018. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. Math. Program. 171 (1), 115–166.

Pei, Z., Lu, H., Jin, Q., Zhang, L., 2022. Target-based distributionally robust optimization for single machine scheduling. European J. Oper. Res. 299 (2), 420–431.

Perakis, G., Sim, M., Tang, Q., Xiong, P., 2018. Robust pricing and production with information partitioning and adaptation. Available at SSRN 3305039.

Pereira, J., 2016. The robust (minmax regret) single machine scheduling with interval processing times and total weighted completion time objective. Comput. Oper. Res. 66, 141–152.

Pinedo, M., 1983. Stochastic scheduling with release dates and due dates. Oper. Res. 31 (3), 559–572.

Pinedo, M., 2008. Scheduling: Theory, algorithms, and systems.

Ramachandra, A., Rujeerapaiboon, N., Sim, M., 2021. Robust conic satisficing. arXiv preprint arXiv:2107.06714.

Sarin, S.C., Nagarajan, B., Liao, L., 2010. Stochastic Scheduling: Expectation-Variance Analysis of a Schedule. Cambridge University Press.

Sim, M., Chen, L., 2024. Robust CARA optimization. Oper. Res.

Sim, M., Tang, Q., Zhou, M., Zhu, T., 2021a. The analytics of robust satisficing. Available at SSRN 3829562.
Sim, M., Zhao, L., Zhou, M., 2021b. Tractable robust supervised learning models. Available at SSRN 3981205.
Simon, H.A., 1955. A behavioral model of rational choice. Q. J. Econ. 99–118.
Smith, J.E., Winkler, R.L., 2006. The optimizer's curse: Skepticism and postdecision surprise in decision analysis. Manage. Sci. 52 (3), 311–322.
Smith, W.E., et al., 1956. Various optimizers for single-stage production. Nav. Res. Logist. Q. 3 (1–2), 59–66.
Wang, Y., Zhang, Y., Tang, J., 2019. A distributionally robust optimization approach for surgery block allocation. European J. Oper. Res. 273 (2), 740–753.
Xu, X., Cui, W., Lin, J., Qian, Y., 2013. Robust makespan minimisation in identical parallel machine scheduling problem with interval data. Int. J. Prod. Res. 51 (12), 3532–3548.
Yang, J., Yu, G., 2002. On the robust single machine scheduling problem. J. Comb. Optim. 6 (1), 17–33.
Yue, F., Song, S., Zhang, Y., Gupta, J.N., Chiong, R., 2018. Robust single machine scheduling with uncertain release times for minimising the maximum waiting time. Int. J. Prod. Res. 56 (16), 5576–5592.
Yue, Q., Zhou, S., 2021. Due-window assignment scheduling problem with stochastic processing times. European J. Oper. Res. 290 (2), 453–468.
Zhang, Y., Shen, Z.J.M., Song, S., 2018. Exact algorithms for distributionally β-robust machine scheduling with uncertain processing times. INFORMS J. Comput. 30 (4), 662–676.
Zhang, Y., Zhang, Z., Lim, A., Sim, M., 2021. Robust data-driven vehicle routing with time windows. Oper. Res. 69 (2), 469–485.
Zhao, C., Guan, Y., 2018. Data-driven risk-averse stochastic optimization with wasserstein metric. Oper. Res. Lett. 46 (2), 262–267.
Zhen, J., den Hertog, D., Sim, M., 2018. Adjustable robust optimization via Fourier–Motzkin elimination. Oper. Res. 66 (4), 1086–1100. http://dx.doi.org/10.1287/opre.2017.1714.
Zheng, F., Man, X., Chu, F., Liu, M., Chu, C., 2019. A two-stage stochastic programming for single yard crane scheduling with uncertain release times of retrieval tasks. Int. J. Prod. Res. 57 (13), 4132–4147.

Further reading

Zangwill, W.I., 1966. A deterministic multi-period production scheduling model with backlogging. Manage. Sci. 13 (1), 105–119.
Zangwill, W.I., 1969. A backlogging model and a multi-echelon model of a dynamic economic lot size production system—a network approach. Manage. Sci. 15 (9), 506–527.