# Approximation results for makespan minimization with budgeted uncertainty

Marin Bougeret, Klaus Jansen, Michael Poss, Lars Rohwedder

# Approximation results for makespan minimization with budgeted uncertainty ⋆

Marin Bougeret[2], Klaus Jansen[1], Michael Poss[2], and Lars Rohwedder[1]

[1]Department of Computer Science, Kiel University, 24098 Kiel, Germany
[2]LIRMM, University of Montpellier, CNRS, France

{marin.bougeret,michael.poss}@lirmm.fr
{kj,lro}@informatik.uni-kiel.de

**Abstract.** We study approximation algorithms for the problem of minimizing the makespan on a set of machines with uncertainty on the processing times of jobs. In the model we consider, which goes back to Bertsimas *et al.* [3], once the schedule is defined an adversary can pick a scenario where deviation is added to some of the jobs' processing times. Given only the maximal cardinality of these jobs, and the magnitude of potential deviation for each job, the goal is to optimize the worst-case scenario. We consider both the cases of identical and unrelated machines. Our main result is an EPTAS for the case of identical machines. We also provide a 3-approximation algorithm and an inapproximability ratio of $2 - \epsilon$ for the case of unrelated machines.
**Keywords:** Makespan minimization, robust optimization, approximation algorithms, EPTAS, parallel machines, unrelated machines

## 1 Introduction

Classical optimization models suppose perfect information over all parameters. This can lead to optimal solutions having poor performance when the actual parameters deviate, even by a small amount, from the predictions used in the optimization model. Different frameworks have been proposed to overcome this issue, among which Robust Optimization which tackles the uncertainty by providing a set of possible values for these parameters, and considering the worst outcome over that set. In this paper, we consider the problem of scheduling a set of jobs $J$ on the set of machines $M$, so as to minimize the makespan, and considering that the processing times are uncertain. What is more, we consider the budgeted uncertainty model introduced by Bertsimas *et al.* [3] where each processing time varies between its nominal value and the latter plus some deviation. Further, in any scenario, at most $\Gamma$ of the uncertain parameters take the higher values, the other being at their nominal values.

Let us now formally define the Robust Scheduling on Unrelated Machines ($R|\mathcal{U}^\Gamma|C_{\max}$) problem. For any job $j \in J$ and machine $i \in M$, we denote by $\overline{p}_{ij} \geq 0$ the nominal processing time of $j$ on $i$, and by $\hat{p}_{ij} \geq 0$ the (potential) deviation of $j$ on $i$. We point

out that nominal processing times are non negative as it is also the case in the classical $R||C_{\max}$ problem, and deviations are also non negative as in the uncertainty model $\mathcal{U}^\Gamma$ defined below, any negative $\hat{p}_{ij}$ can be equivalently set to 0. A schedule $\sigma$ is a function from $J \to M$. We write $\sigma_i$ for the subset of jobs scheduled on machine $i$. Let $\mathcal{U}^\Gamma = \{\xi \in \{0,1\}^{|J|} : \|\xi\|_1 \leq \Gamma\}$ be the set of all possible scenarios where at most $\Gamma$ jobs deviate. For any $\xi \in \mathcal{U}^\Gamma$, we set $p_{ij}^\xi = \overline{p}_{ij} + \xi_j \hat{p}_{ij}$ to be the actual processing time of $j$ on $i$ in scenario $\xi$.

Let us now formalize some common terms, but with dependence on scenario $\xi$. The load of machine $i$ in scenario $\xi$ is calculated as $\sum_{j \in \sigma_i} p_{ij}^\xi$. The makespan in scenario $\xi$ is the maximum load in scenario $\xi$, i.e., $C_{\max}^\xi(\sigma) = \max_{i \in M} \sum_{j \in \sigma_i} p_{ij}^\xi$. Finally, $C_{\max}^\Gamma(\sigma) = \max_{\xi \in \mathcal{U}^\Gamma} C_{\max}^\xi(\sigma)$ denotes the objective function we consider in Robust Scheduling, where the adversary takes the worst scenario among $\mathcal{U}^\Gamma$.

Next, we will state important observations about the objective function. We first need to introduce the following notations. Given a set of jobs $X_i$ scheduled on machine $i$, we define $\overline{p}(X_i) = \sum_{j \in X_i} \overline{p}_{ij}$, $\hat{p}(X_i) = \sum_{j \in X_i} \hat{p}_{ij}$, $\Gamma(X_i)$ as the set of the $\Gamma$ jobs of $X_i$ with the largest $\hat{p}_{ij}$ values (or $\Gamma(X_i) = \sigma_i$ when $|X_i| < \Gamma$) with ties broken arbitrarily. Finally, set $\hat{p}_\Gamma(X_i) = \hat{p}(\Gamma(X_i))$.

By definition we have $C_\Gamma(\sigma) = \max_{\xi \in \mathcal{U}^\Gamma} \max_{i \in M} \sum_{j \in \sigma_i} C_\xi(\sigma)$, and thus we can rewrite $C_\Gamma(\sigma) = \max_{i \in M} \max_{\xi \in \mathcal{U}^\Gamma} \sum_{j \in \sigma_i} C_\xi(\sigma) = \max_{i \in M} C_\Gamma(\sigma_i)$, where $C_\Gamma(\sigma_i) = \max_{\xi \in \mathcal{U}^\Gamma} \sum_{j \in \sigma_i} C_\xi(\sigma)$ is the worst-case makespan on machine $i$. The benefit of rewriting $C_\Gamma(\sigma)$ in this form is that it is now clear that $C_\Gamma(\sigma_i) = \overline{p}(\sigma_i) + \hat{p}_\Gamma(\sigma_i)$ as the worst scenario $\xi$ for a fixed $\sigma_i$ is obtained by picking the $\Gamma$ jobs with highest $\hat{p}_{ij}$ and make them deviate. Thus, $R|\mathcal{U}^\Gamma|C_{\max}$ can also be thought as a "classical" scheduling problem (without adversary) where the makespan on a machine $C_\Gamma(\sigma_i)$ is simply the sum of all the nominal processing time of jobs of $\sigma_i$, plus only the $\Gamma$ largest deviating values of jobs of $\sigma_i$. We are now ready to define $R|\mathcal{U}^\Gamma|C_{\max}$.

*Problem 1.* ROBUST SCHEDULING ON UNRELATED MACHINES ($R|\mathcal{U}^\Gamma|C_{\max}$)

- Input: $(J, M, \overline{p} \in \mathbb{Q}_+^{|M| \times |J|}, \hat{p} \in \mathbb{Q}_+^{|M| \times |J|})$ where $J$ is the set of jobs, $M$ the set of machines, $\overline{p}$ are the vectors of nominal processing times, and $\hat{p}$ the vectors of deviation
- Output: find a schedule $\sigma : J \to M$
- Objective function: $\min C_\Gamma(\sigma) = \max_{\xi \in \mathcal{U}^\Gamma} \max_{i \in M} \sum_{j \in \sigma_i} [\overline{p}_{ij} + \xi_j \hat{p}_{ij}] = \max_{i \in M} C_\Gamma(\sigma_i)$, where $C_\Gamma(\sigma_i) = \overline{p}(\sigma_i) + \hat{p}_\Gamma(\sigma_i)$.

Following the classical three field notation, we denote by $R|\mathcal{U}^\Gamma|C_{\max}$ the previous problem. Notice that when all $\hat{p}_{ij} = 0$ the problem corresponds to the classical $R||C_{\max}$, for which we denote by $C(\sigma_i) = \sum_{j \in \sigma_i} \overline{p}_{ij}$ the makespan on machine $i$. We are also interested in a simplification of the above problem. This simplification is ROBUST SCHEDULING ON IDENTICAL MACHINES ($P|\mathcal{U}^\Gamma|C_{\max}$) where each has two processing times ($\overline{p}_j$ and $\hat{p}_j$), and we have $\overline{p}_{ij} = \overline{p}_j$ and $\hat{p}_{ij} = \hat{p}_j$ for any $i$.

Robust scheduling has been considered in the past, mostly for finite uncertainty sets without particular structure [1, 6, 9, 10, 13, 11]. More recently, robust packing and

scheduling with the budgeted uncertainty model $\mathcal{U}^\Gamma$ from Bertsimas *et al.* [4] has been considered [5, 14, 2]. Specifically, Bougeret *et al.* [5] provided a 3-approximation algorithm and a $(1 + \epsilon)$-approximation (PTAS) for $P|\mathcal{U}^\Gamma|C_{\max}$ but only for a constant $\Gamma$, as well as a randomized approximation algorithm for $R|\mathcal{U}^\Gamma|C_{\max}$ having an average ratio of $O(\log(m))$, and an FPTAS for $Rm|\mathcal{U}^\Gamma|C_{\max}$ (where the number of machines is fixed). It is also known [5] that $1|\mathcal{U}^\Gamma|\sum_j w_j C_j$ is $\mathcal{NP}$-hard in the strong sense, and is polynomial when $w_j = 1$ for $j \in J$. Tadayon *et al.* [14] considered the robust one-machine problem for four commonly-used objective criteria: (weighted) total completion time, maximum lateness/tardiness, and number of late jobs. They showed that some of these problems are polynomially solvable and provide mixed-integer programming formulations for others. Their results considered $\mathcal{U}^\Gamma$ as well as two closely related uncertainty sets. In a recent work [2] (with also authors in common), it has been proved that robust bin-packing problem admits a constant-factor approximation algorithm, both for the $\mathcal{U}^\Gamma$ model, and the uncertainty sets considered by Tadayon *et al.* [14].

In this paper we improve existing results [5] for $P|\mathcal{U}^\Gamma|C_{\max}$ and $R|\mathcal{U}^\Gamma|C_{\max}$. In Section 2 we show that any $c$-approximation for the classical $R||C_{\max}$ problem leads to a $(c+1)$-approximation for $R|\mathcal{U}^\Gamma|C_{\max}$, hence obtaining a 3-approximation algorithm for the latter problem, and a $(2 + \epsilon)$-approximation for $P|\mathcal{U}^\Gamma|C_{\max}$. We point out that this result improves the previous *ad-hoc* 3-approximation for $P|\mathcal{U}^\Gamma|C_{\max}$ [5], while having a simpler proof. In Section 3, we show through a reduction from the Restricted Assignment Problem that there exists no $(2-\epsilon)$-approximation algorithm for $R|\mathcal{U}^\Gamma|C_{\max}$ unless $\mathcal{P} = \mathcal{NP}$. This implies that the best possible ratio (unless $\mathcal{P} = \mathcal{NP}$) for $R|\mathcal{U}^\Gamma|C_{\max}$ is somewhere between 2 and 3, contrasting with the classical $R||C_{\max}$ where the gap between $\frac{3}{2}$ and 2 is open since the work of Lenstra *et al.* [12].

In Section 4 we consider the $P|\mathcal{U}^\Gamma|C_{\max}$ problem and present the first step of our main result, namely a PTAS which is valid even when $\Gamma$ is part of the input, i.e., not constant. Having $\Gamma$ in the input (and not constant) requires a totally different technique from the one used in the existing PTAS [5] for fixed $\Gamma$. The algorithm is turned into an EPTAS in Section 5, i.e., a PTAS where the dependency of $\epsilon$ is not in the exponent of the encoding length.

## 2 A 3-approximation for unrelated machines

**Theorem 1.** *Any polynomial $c$-approximation for $R||C_{\max}$ implies a polynomial $(c+1)$-approximation for $R|\mathcal{U}^\Gamma|C_{\max}$.*

*Proof.* We design a dual approximation, i.e., given an instance $I$ of $R|\mathcal{U}^\Gamma|C_{\max}$ and an threshold $T$, we either give a schedule $\sigma$ of $I$ with $C_\Gamma(\sigma) \leq (c + 1)T$, or prove that $T < \mathrm{OPT}(I)$. Using a binary search on $T$ this will imply a $(c + 1)$-approximation algorithm.

For that, given an instance $I = (J, M, \overline{p}, \hat{p})$ of $R|\mathcal{U}^\Gamma|C_{\max}$, and $T$ the current threshold, our objective is to define an instance $I' = (J, M, p)$ of the classical $R||C_{\max}$ problem. The transformation of a solution for $I'$ to a solution for $I$ will be straightforward since the jobs and machines will be the same.

Given a machine $i$, let $B_i = \{j | \hat{p}_{ij} > \frac{T}{\Gamma}\}$ and $S_i = J \setminus B_i$. Define

$$p_{ij} := \begin{cases} \overline{p}_{ij} + \hat{p}_{ij} & \text{if } j \in B_i \\ \overline{p}_{ij} & \text{otherwise.} \end{cases} \tag{1}$$

Let us now prove that (1) if $\text{OPT}(I') > T$ then we have $\text{OPT}(I) > T$, and (2) every schedule $\sigma$ with makespan $C^{I'}(\sigma)$ in $I'$ has a makespan at most $C^{I'}(\sigma) + T$ in $I$ ($C_\Gamma(\sigma) \leq C^{I'}(\sigma) + T$).

For (1), we prove that $\text{OPT}(I) \leq T$ implies that $\text{OPT}(I') \leq T$. Let $\sigma$ be an optimal solution of $I$ and $i$ a machine. $C_\Gamma(\sigma) \leq T$ implies that $C_\Gamma(\sigma_i) \leq T$ for any $i$, and thus that $\overline{p}(\sigma_i) + \hat{p}_\Gamma(\sigma_i) \leq T$. Now, observe that $(B_i \cap \sigma_i) \subseteq \Gamma(\sigma_i)$. Indeed, assume towards contradiction that there exists $j \in (B_i \cap \sigma_i) \setminus \Gamma(\sigma_i)$. This implies that $|\Gamma(\sigma_i)| = \Gamma$. As by definition, any $j' \in \Gamma(\sigma_i)$ has $\hat{p}_{ij'} \geq \hat{p}_{ij} > \frac{T}{\Gamma}$, we get that $\hat{p}_\Gamma(\sigma_i) > T$, a contradiction. This implies $C^{I'}(\sigma_i) = \overline{p}(\sigma_i) + \hat{p}(B_i \cap \sigma_i) \leq \overline{p}(\sigma_i) + \hat{p}_\Gamma(\sigma_i) \leq T$.

For (2), let $\sigma$ be a solution of $I'$. Let $i \in M$. Observe that $\hat{p}(\Gamma(\sigma_i)) \leq \hat{p}(B_i \cap \sigma_i) + T$ as $\Gamma(\sigma_i)$ contains at most $\Gamma$ jobs in $\sigma_i \setminus B_i$, and these jobs have $\hat{p}_{ij} \leq \frac{T}{\Gamma}$. Thus, $C_\Gamma(\sigma_i) = \overline{p}(\sigma_i) + \hat{p}_\Gamma(\sigma_i) \leq \overline{p}(\sigma_i) + \hat{p}(B_i \cap \sigma_i) + T = C^{I'}(\sigma_i) + T$.

Thus, given a $T$ and $I$ we create $I'$ as above and run the $c$-approximation for $R||C_{\max}$ to get a solution $\sigma$. If $C^{I'}(\sigma) > cT$ then $\text{OPT}(I') > T$, implying $\text{OPT}(I) > T$, and thus we reject $T$. Otherwise, we consider $\sigma$ as a solution for $I$, and $C_\Gamma(\sigma) \leq (c+1)T$. $\square$

Using the well-known 2-approximation algorithm from Lenstra *et al.* [12], we obtain immediately the following.

**Corollary 1.** *There is a 3-approximation for $R|\mathcal{U}^\Gamma|C_{\max}$.*

Since by this reduction identical machines stay identical we also obtain the following using the EPTAS of Jansen *et al.* [7] for the classical $P||C_{\max}$ problem.

**Corollary 2.** *For every $\epsilon > 0$ there is a $(2 + \epsilon)$-approximation for $P|\mathcal{U}^\Gamma|C_{\max}$ running in time $2^{O(1/\epsilon \log(1/\epsilon)^4)} + poly(n)$.*

We point out that even if the EPTAS of Jansen *et al.* [7] also holds for the $Q||C_{\max}$ problem (where each machine $i$ has a speed $s_i$, and $p_{ij} = \frac{p_j}{s_i}$), Corollary 2 cannot be extended to $Q|\mathcal{U}^\Gamma|C_{\max}$ using the same reduction. Indeed, starting from a $Q|\mathcal{U}^\Gamma|C_{\max}$ instance (where each job has only two processing times $\overline{p}_j$ and $\hat{p}_j$, and we have $\overline{p}_{ij} = \frac{\overline{p}_j}{s_i}$, $\hat{p}_{ij} = \frac{\hat{p}_j}{s_i}$ for any $i$), the above reduction would create an instance of $R||C_{\max}$ (and not of $Q||C_{max}$) as some job may require more than $\frac{T}{\Gamma}$ on a machine, and less that $\frac{T}{\Gamma}$ on another.

## 3  A $2 - \epsilon$ inapproximability for unrelated machines

For the classical $R||C_{\max}$ problem, when all $p_{ij} \in \{1, \infty\}$, deciding if the optimal value is at most 1 is polynomially solvable as it can be reduced to finding a matching in a bipartite graph. The result below shows that answering the same question for $R|\mathcal{U}^\Gamma|C_{\max}$ is $\mathcal{NP}$-complete.

**Theorem 2.** *Given an instance $I$ of $R|\mathcal{U}^\Gamma|C_{\max}$, it is NP-complete to decide if $\mathrm{OPT}(I) \le 1$ or $\mathrm{OPT}(I) \ge 2$, and thus for any $\epsilon > 0$ is no $(2-\epsilon)$-approximation algorithm for $R|\mathcal{U}^\Gamma|C_{\max}$ unless $\mathcal{P} = \mathcal{NP}$, even for $\Gamma = 1$ and when each job can be scheduled on at most $3$ machines.*

*Proof.* Let us define a reduction from 3-SAT to $R|\mathcal{U}^\Gamma|C_{\max}$ with $\Gamma = 1$. Let $I_0$ be an instance of 3-SAT with clauses $\{C_i, i \in [m_0]\}$ and variables $\{x_j, j \in [n_0]\}$. Each $C_i$ is of the form $l_i^1 \vee l_i^2 \vee l_i^3$ where $l_i^k \in \{x_j, \bar{x}_j\}$ for some $j$. We define an instance $I$ of $R|\mathcal{U}^\Gamma|C_{\max}$ with $m = 2n_0$ machines and $n = n_0 + m_0$ jobs as follows. To each variable $x_j$ we associate two machines $\{j_f, j_t\}$. We create a set of $n_0$ variable jobs where for any $j \in [n_0]$, $\overline{p}_{j_f j} = \overline{p}_{j_t j} = 1$, $\overline{p}_{i'j} = \infty$ for any other $i'$, and $\hat{p}_{ij} = 0$ for any $i \in [m]$. For any clause $C_i$, $i \in [m_0]$ we define $M_i$: the set of 3 machines corresponding to literals $\{l_i^k\}$ satisfying $C_i$. For example, if $C_7 = x_1 \vee \bar{x}_3 \vee x_5$ then $M_7 = \{1_t, 3_f, 5_t\}$. We now define a set of $m_0$ clause jobs as follows. For any $j \in [n_0 + 1, n_0 + m_0]$, job $j$ represents clause $C_{j-n_0}$ with $\hat{p}_{ij} = 1$ iff $i \in M_{j-n_0}$, $\hat{p}_{i'j} = \infty$ for any other $i'$, and $\overline{p}_{ij} = 0$ for any $i \in [m]$. For example, job $j = n_0 + 7$ is associated to $C_7$ where in particular $\hat{p}_{1_t j} = \hat{p}_{3_f j} = \hat{p}_{5_t j} = 1$. Notice that each clause job can be scheduled on at most 3 machines. Let us now verify that $I_0$ is satisfiable iff $\mathrm{OPT}(I) = 1$.

$\Rightarrow$. Suppose $I_0$ is satisfied by assignment $a$. For any $j \in [n_0]$, we schedule $j$ on $j_t$ if $x_j$ is set to false in $a$ and on $j_f$ otherwise. For any $j \in [n_0 + 1, n_0 + m_0]$, we schedule job $j$ on any machine $i \in M_{j-n_0}$ corresponding to a literal satisfying $C_i$ in assignment $a$. Notice that in this schedule, a machine either receives exactly one variable job, implying a makespan of 1, or only clause jobs, also implying a makespan of 1 as $\Gamma = 1$.

$\Leftarrow$. Suppose that $\mathrm{OPT}(I) = 1$ and let us define an assignment $a$. This implies that any variable job $j$ is either scheduled on machine $j_f$, in which case we set $x_j$ to true, or on machine $j_t$, in which case we set $x_j$ to false. As $\mathrm{OPT}(I) = 1$, and clause job $j \in [n_0 + 1, n_0 + m_0]$ is scheduled on a machine $i \in M_{j-n_0}$ that did not receive a variable job, implying that clause $j - n_0$ is satisfied by literal $i$. $\qquad\square$

## 4   A PTAS for identical machines

Recall that for the $P|\mathcal{U}^\Gamma|C_{\max}$ problem, given two $n$ dimensional vectors $\hat{p}$ and $\overline{p}$ and the number of machine $m$, the objective is to create a schedule $\sigma$ that minimizes $\max_{i \in M} C_\Gamma(\sigma_i)$. Recall also that $C_\Gamma(\sigma_i) = \overline{p}(\sigma_i) + \hat{p}_\Gamma(\sigma_i)$, where $\overline{p}(\sigma_i) = \sum_{j \in \sigma_i} \overline{p}_j$, and $\hat{p}_\Gamma(\sigma_i)$ is the sum of the $\hat{p}_j$ values of the $\Gamma$ largest jobs (w.r.t. $\hat{p}_j$) of $\sigma_i$ (or the sum of all $\hat{p}_j$ values if $|\sigma_i| \le \Gamma$). To obtain a PTAS for $P|\mathcal{U}^\Gamma|C_{\max}$, we will reduce to the following problem, which admits an EPTAS (see [8]).

*Problem 2.* Unrelated Machines with few Machine Types

– Input: $n$ jobs and a set $M$ of $m$ machines with processing times $p_{ij} \ge 0$ for job $j$ on machine $i$. Moreover, there is a constant $k$ and machine types $T_1 \dot{\cup} \cdots \dot{\cup} T_k = \{1, \ldots, m\}$, such that every machine within a type behaves the same. Formally, for every $k'$, every $i, i' \in T_{k'}$ and every $j \le n$ it holds that $p_{ij} = p_{i'j}$

- Output: find a schedule $\sigma : J \to M$
- Objective function: minimize makespan $C(\sigma) = \max_{i \in M} C(\sigma_i)$, where $C(\sigma_i) = \sum_{j \in \sigma_i} p_{ij}$

Notice that the EPTAS of Jansen and Maack [8] for this problem provides an $(1 + \epsilon)$-approximation running in time $f(|I|, \epsilon, k) = 2^{O(k \log(k) \frac{1}{\epsilon} \log^4(\frac{1}{\epsilon}))} + poly(|I|)$.

We also introduce the following decision problem.

*Problem 3.* UNRELATED MACHINES WITH FEW MACHINE TYPES AND CAPACITIES

- Input: as above, but in addition every machine $i$ has a capacity $c_i \in (0, 1]$. Moreover, capacities are the same among a type (for any $k' \in [k]$, for any $i, i' \in T_{k'}$, $c_i = c_{i'}$)
- Output: decide if there is a schedule where $C(\sigma_i) \leq c_i$ for any $i$.

Notice that the EPTAS for Problem 2 allows to approximately decide Problem 3 in the following sense.

**Lemma 1.** *There is an algorithm that for any $\epsilon > 0$, either outputs a schedule with $C(\sigma_i) \leq (1 + \epsilon) \cdot c_i$ for any $i$, or reject the instance, proving that there is no schedule with $C(\sigma_i) \leq c_i$ for any $i$. This algorithm runs in time $f(|I|, \epsilon, k)$ where $f$ is the complexity of the above EPTAS to get a $(1 + \epsilon)$-approximation.*

*Proof.* Let $A$ be the EPTAS of Jansen and Maack [8] for Problem 2. Given a input $I$ of Problem 3 we define an input $I'$ of Problem 2 in the following way. For every $j \leq n$, scale $p_{ij}$ to $p_{ij}/c_i$. Then, if $A(I') \leq (1 + \epsilon)$, we can convert the solution found by $A$ into a solution for $I$ of makespan at most $(1 + \epsilon) \cdot c_i$ for any $i$. Otherwise, as $A$ is a $(1 + \epsilon)$-approximation, it implies that $\text{OPT}(I') > 1$, and thus that no solution can have makespan at most $c_i$ for any $i$. $\qquad\square$

Let us now describe the PTAS for $P|\mathcal{U}^\Gamma|C_{\max}$. Our objective is to provide a $(1 + O(\epsilon))$ dual approximation for $P|\mathcal{U}^\Gamma|C_{\max}$. The constant multiplying $\epsilon$ can be ignored since we can divide $\epsilon$ by this constant in a preprocessing step.

*1. Guess the makespan and scale* OPT *to 1.* Let $I$ be an input of $P|\mathcal{U}^\Gamma|C_{\max}$, and $T$ be a positive value (representing the current threshold). We start by redefining $I$ by scaling $p_j := \frac{p_j}{T}$. Our objective is now to produce a schedule $\sigma$ with $C_\Gamma(\sigma) \leq 1 + \epsilon$, or to prove that $\text{OPT}(I) > 1$.

*2. Rounding deviations.* Let us now define $I^1$ (having vectors $\bar{p}^1$ and $\hat{p}^1$) in the following way. For any $j$, if $\hat{p}_j < \epsilon/\Gamma$ then we set $\hat{p}_j^1 \leftarrow 0$. Intuitively, this will only result in an error of at most $\Gamma \cdot \epsilon/\Gamma$ on every machine. Otherwise ($\hat{p}_j \geq \epsilon/\Gamma$), we define $\hat{p}_j^1$ by rounding $\hat{p}_j$ to the closest smaller value of the form $\epsilon/\Gamma \cdot (1 + \epsilon)^i$.

**Observation 1** *In $I^1$ there are at most $O(1/\epsilon \log(\Gamma/\epsilon))$ deviation values, and at most $O(1/\epsilon \log(1/\epsilon))$ deviation values in the interval $[\epsilon/\Gamma, 1/\Gamma]$.*

In the following, we will denote by $C_\Gamma^{I'}(\sigma)$ the cost of $\sigma$ for an instance $I'$.

**Observation 2** *If* $\mathrm{OPT}(I) \leq 1$ *then* $\mathrm{OPT}(I^1) \leq 1$. *If we get solution* $\sigma^1$ *of* $I^1$, *then* $C_\Gamma^I(\sigma^1) \leq (1 + \epsilon)C_\Gamma^{I^1}(\sigma^1) + \epsilon$

It only remains now to either produce a good solution of $I^1$ (of cost at most $1 + O(\epsilon)$), or prove that $\mathrm{OPT}(I^1) > 1$.

*3. Machine thresholds.* Given any solution $\sigma$ of $I^1$ such that $C_\Gamma^{I^1}(\sigma) \leq 1$, we can associate to $\sigma$ an outline $\psi(\sigma)$ (denoted as an $m$ dimensional vector $t$) which is defined as follows. For any machine $i$ with more that $\Gamma$ jobs, the threshold value $t_i$ is such that any job on $i$ with $\hat{p}_j > t_i$ deviates (belongs to $\Gamma(\sigma_i)$) and none of the jobs with $\hat{p}_j < t_i$ deviate. Notice that among jobs with $\hat{p}_j = t_i$, some may deviate, but not necessarily all. For any machine $i$ with at most $\Gamma$ jobs, we define $t_i = 0$, implying again that any job with $\hat{p}_j > t_i$ deviates on $i$. Notice that in both cases we have $\hat{p}_\Gamma(\sigma_i) \geq \Gamma \cdot t_i$. Notice also that $C_\Gamma^{I^1}(\sigma) \leq 1$ implies $t_i \leq \frac{1}{\Gamma}$. Indeed, if we had $t_i > \frac{1}{\Gamma}$, there would be $\Gamma$ deviating jobs with $\hat{p}_j > t_i$, implying $C_\Gamma^{I^1}(\sigma_i) > 1$, a contradiction. Let us denote by $\Delta$ the set of all possible values of a $t_i$. According to Observation 1 we have $|\Delta| = O(1/\epsilon \log(1/\epsilon))$. Let $\mathcal{P} = \Delta^m$ be the set of all possible outlines (of solutions of cost at most 1).

**Lemma 2.** *Consider a solution* $\sigma^{1*}$ *of* $I^1$ *such that* $C_\Gamma(\sigma^{1*}) \leq 1$, *and let* $t^* = \psi(\sigma^{1*})$. *Then, we can guess in* $m^{O(1/\epsilon \log(1/\epsilon))}$ *time the vector* $t^*$ *(or a permutation thereof).*

*Proof.* As $t^* \in T$, all the $t_i^*$ have a value in $\{0\} \cup [\frac{\epsilon}{\Gamma}, \frac{1}{\Gamma}]$. Thus, as deviating values are rounded in $I^1$, there are only a constant number of possible threshold value and we can guess them. For every possible threshold, we guess how many machines in the optimal solution have it. $\qquad\square$

Thus, we can now assume that we know the vector $t^*$.

*4. Constructing an instance with few machine types and capacities.* To give an insight of the correct reduction defined below, let us first see what happen if we define an instance $I^2(t^*)$ of $R||C_{\max}$ as follows. For simplicity, we also assume that there are no job with $\hat{p}_j = t_i^*$ on each machine $i$ in the previously considered optimal solution of $I^1$. For any machine $i$ and job $j$, define the processing time in $I^2(t^*)$ as $p_{ij} = \bar{p}_j + \hat{p}_j$ if $\hat{p}_j \geq t_i^*$, and $p_{ij} = \bar{p}_j$ otherwise. Then, consider the following implications.

1. if $OPT(I^1) \leq 1$, then $\mathrm{OPT}(I^2(t^*)) \leq 1$
2. for any solution $\sigma'$ of $I^2(t^*)$, $C_\Gamma^{I^1}(\sigma') \leq C^{I^2}(\sigma')$ (implying that if there exists $\sigma'$ with $C^{I^2}(\sigma') \leq 1 + \epsilon$, then we will have our solution for $I^1$ of cost $1 + \epsilon$)

While Property (1) holds, this is not the case for Property (2). Indeed, suppose that in $\sigma'$ there is a machine $i$ such that for all jobs $j$ scheduled on $i$, $\hat{p}_j < t_i^*$. This implies that $C(\sigma_i) = \sum_{j \in \sigma_i} \bar{p}_j$. However, if we look now at $\sigma'$ in $I^1$, we get $C_\Gamma^{I^1}(\sigma_i) = C^{I^2}(\sigma_i) + \hat{p}(\Gamma(\sigma_i))$, which is greater than the claimed value. To solve this problem we have to remember in $R||C_{\max}$ that there will be a space of size at most $\Gamma \cdot t_i$ which will be occupied by deviations.

Let us now turn to the correct version.

**Definition 1.** *For any $t \in \mathcal{P}$, we define the following input $I^2(t)$ of Problem 3. We set the machine capacity to*

$$c_i := 1 - \Gamma \cdot t_i + \epsilon.$$

*The addition of $\epsilon$ is only a technicality to ensure that all $c_i$ are non-zero. Note that if there are less than $\Gamma$ jobs on $i$, then $t_i$ must be $0$ and therefore $c_i = 1 + \epsilon$. For every job $j$ set*

$$p_{ij} := \begin{cases} \overline{p}_j + \hat{p}_j - t_i & \text{if } \hat{p}_j \geq t_i, \\ \overline{p}_j & \text{if } \hat{p}_j < t_i. \end{cases}$$

*Note that at $\hat{p}_j = t_i$, the values of both cases are equal. Notice also that in $I^2(t)$ there are only $|\Delta|$ different machine types.*

**Lemma 3.** *If $\text{OPT}(I^1) \leq 1$ and $t$ is the outline of an optimal solution $\sigma^2$, for any $i$, $C^{I^2(t)}(\sigma_i^2) \leq c_i$.*

*Proof.* Let us consider jobs $\sigma_i^2$ scheduled on machine $i$. If $t_i = 0$, then

$$\sum_{j \in \sigma_i^2} p_{ij} = \sum_{j \in \sigma_i^2} \overline{p}_j + \hat{p}_j \leq 1 < c_i.$$

Assume now $t_i > 0$, implying that $|\Gamma(\sigma_i^2)| \geq \Gamma$. By choice of $t_i$, every job $j \in \Gamma(\sigma_i^2)$ has $\hat{p}_j \geq t_i$ and every $j \in \sigma_i^2 \setminus \Gamma(\sigma_i^2)$ has $\hat{p}_j \leq t_i$. This implies

$$\sum_{j \in \sigma_i^2} p_{ij} = \sum_{j \in \Gamma(\sigma_i^2)} p_{ij} + \sum_{j \in \sigma_i^2 \setminus \Gamma(\sigma_i^2)} p_{ij} = \sum_{j \in \Gamma(\sigma_i^2)} [\overline{p}_j + \hat{p}_j - t_i] + \sum_{j \in \sigma_i^2 \setminus \Gamma(\sigma_i^2)} \overline{p}_j \leq 1 - \Gamma \cdot t_i < c_i.$$

$\square$

**Lemma 4.** *For any $t \in \mathcal{P}$, if there is a solution $\sigma^2$ of $I^2(t)$ such that $C^{I^2(t)}(\sigma_i^2) \leq (1 + \epsilon) \cdot c_i$ for any $i$, then $C_\Gamma^{I^1}(\sigma^2) \leq (1 + \epsilon)^2$.*

*Proof.* Let $i$ be a machine. Then for every $j \in \sigma_i^2$,

$$p_{ij} = \begin{cases} \overline{p}_j + \hat{p}_j - t_i \geq \overline{p}_j & \text{if } \hat{p}_j \geq t_i, \\ \overline{p}_j & \text{if } \hat{p}_j < t_i. \end{cases}$$

Furthermore, for every $j \in \Gamma(\sigma_i^2)$,

$$p_{ij} = \begin{cases} \overline{p}_j + \hat{p}_j - t_i & \text{if } \hat{p}_j \geq t_i, \\ \overline{p}_j > \overline{p}_j + \hat{p}_j - t_i & \text{if } \hat{p}_j < t_i. \end{cases}$$

This implies,

$$\sum_{j \in \Gamma(\sigma_i^2)} [\overline{p}_j + \hat{p}_j] + \sum_{j \in \sigma_i^2 \setminus \Gamma(\sigma_i^2)} \overline{p}_j \leq \Gamma \cdot t_i + \sum_{j \in \Gamma(\sigma_i^2)} [\overline{p}_j + \hat{p}_j - t_i] + \sum_{j \in \sigma_i^2 \setminus \Gamma(\sigma_i^2)} \overline{p}_j$$

$$\leq \Gamma \cdot t_i + \sum_{j \in \Gamma(\sigma_i^2)} p_{ij} + \sum_{j \in \sigma_i^2 \setminus \Gamma(\sigma_i^2)} p_{ij} = \Gamma \cdot t_i + \underbrace{\sum_{j \in \sigma_i^2} p_{ij}}_{\leq (1+\epsilon) \cdot c_i} \leq \Gamma \cdot t_i + (1+\epsilon) \cdot (1 - \Gamma \cdot t_i + \epsilon) \leq (1+\epsilon)^2.$$

$\square$

**Theorem 3.** *There is a $(1+\epsilon)$-approximation algorithm for $P|\mathcal{U}^\Gamma|C_{\max}$ running in time $O(m^{O(1/\epsilon \log(1/\epsilon))} \times f(|I|, \epsilon, O(1/\epsilon \log(1/\epsilon))))$ where $f$ is the function of Lemma 1. This implies a running time in $m^{O(1/\epsilon \log(1/\epsilon))}(2^{g(\epsilon)} + |I|^{O(1)})$, where $g(\epsilon) = O((1/\epsilon)^2 \log^6(1/\epsilon))$.*

*Proof.* Given $I$ input of $P|\mathcal{U}^\Gamma|C_{\max}$ and a threshold $T$, we run algorithm $A$ of Lemma 1 on $I^2(t)$ for any $t \in \mathcal{P}$ with a precision $\epsilon$. If $A$ rejects all the $I^2(t)$ then we can reject $T$ according to Observation 2 and Lemma 3. Otherwise, there exists $t_0$ such that $A(I^2(t_0))$ outputs a schedule $\sigma^2$ where $C^{I^2(t_0)}(\sigma^2) \leq (1+\epsilon) \cdot c_i$ for any $i$, implying $C_\Gamma^I(\sigma^2) \leq (1+\epsilon)C_\Gamma^{I^1}(\sigma^2) + \epsilon \leq (1+\epsilon)^3 + \epsilon \leq 1 + 5\epsilon$ according to Observation 2 and Lemma 4 (for sufficiently small $\epsilon$). Finally, the running time is as claimed due to the bound of $\mathcal{P}$ in Lemma 2. $\qquad\square$

## 5 EPTAS for identical machines

The approach for an EPTAS is similar to the PTAS above. We would like to remove the bottleneck from the previous section, which is the guessing the thresholds. In the PTAS we notice that even if the thresholds were chosen incorrectly, but we find a solution to the derived problem, we can get a good solution for the initial problem. Informally, we will now still create an instance of Problem 3, but we only guess approximately the number of machines for each threshold.

We start by defining $I^1$ as in the previous section. Given any solution $\sigma^1$ of $I^1$ such that $C_\Gamma^{I^1}(\sigma^1) \leq 1$, we can associate to $\sigma$ a restricted outline $\overline{m} = \overline{\psi}(\sigma)$ where $\overline{m}$ is defined as follows. Let $t = \psi(\sigma)$. For any threshold value $l \in \Delta$, let $m_l = |\{i | t_i = l\}|$ be the number of machines with threshold $l$ in $\sigma^1$. We define $\overline{m}_l \in \{0, 1, 2, 4, 8, \ldots, 2^{\lfloor \log(m) \rfloor}\}$ such that $\overline{m}_l \leq m_l < 2\overline{m}_l$. Let $\overline{\mathcal{P}} = \{\overline{m} \in \{0, 1, 2, 4, 8, \ldots, 2^{\lfloor \log(m) \rfloor}\}^\Delta$ such that $\frac{m}{2} \leq \sum_l \overline{m}_l \leq m\}$ be the set of possible restricted outlines (of solutions of cost at most 1).

**Lemma 5.** *Consider a solution $\sigma^{1*}$ of $I^1$ such that $C_\Gamma(\sigma^{1*}) \leq 1$, and let $\overline{m}^* = \overline{\psi}(\sigma^{1*})$. Then, we can guess in time $2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}$ the vector $\overline{m}^*$.*

*Proof.* Clearly it suffices to iterate over all values $\overline{m}_i \in \{0, 1, 2, 4, 8, \ldots, 2^{\lfloor \log(m) \rfloor}\}$, i.e., $O(\log(m))$ many. Guessing this number for every threshold value in $\Delta$ takes $\log^{O(1/\epsilon \log(1/\epsilon))}(m)$ time. Consider first the case when $\log(m)/\log\log(m) \leq 1/\epsilon \log(1/\epsilon)$. Observe that for any $m \geq 2$, it holds that $\log^{1/2}(m) \leq \log(m)/\log\log(m)$. This implies that $\log^{1/2}(m) \leq 1/\epsilon \log(1/\epsilon)$. Hence,

$$\log^{O(1/\epsilon \log(1/\epsilon))}(m) = (\log^{1/2}(m))^{2 \cdot O(1/\epsilon \log(1/\epsilon))} \leq (1/\epsilon \log(1/\epsilon))^{O(1/\epsilon \log(1/\epsilon))} \leq 2^{O(1/\epsilon \log^2(1/\epsilon))}.$$

If on the other hand, if $\log(m)/\log\log(m) \geq 1/\epsilon \log(1/\epsilon)$, then

$$\log^{O(1/\epsilon \log(1/\epsilon))}(m) \leq \log^{O(\log(m)/\log\log(m))}(m) = 2^{O(\log(m)/\log\log(m) \cdot \log\log(m))} = m^{O(1)}.$$

We conclude,
$$\log^{O(1/\epsilon \log(1/\epsilon))}(m) \leq 2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}.$$

From all the guesses, we report fail whenever $\sum_i m_i < m/2$ or $\sum_i m_i > m$. $\qquad\square$

For any $\overline{m} \in \overline{\mathcal{P}}$, we define the following input $I^2(\overline{m})$ of Problem 3. We first create for any $l$ a set $M_l$ of $\overline{m}_l$ machines where for each machine $i \in M_l$ the capacity and the $p_{ij}$ are defined as in Definition 1 for threshold $t_i = l$. Then, we create another set $M'_l$ of $\overline{m}_l$ machines (that we call cloned machines) with the same capacity and the same $p_{ij}$ values. Let $m' = \sum \overline{m}_l$. Notice that the total number of machines is $2m'$, with $m \leq 2m' < 2m$. Thus, we have to ensure that not too many machines are used in total. For that purpose we add a set of $2m' - m$ dummy jobs $D$, where all $j \in D$ have $p_{ij} = \infty$ on the original machines $i \in M_l$ and $p_{ij} = c_i$ on every cloned machine $i \in M'_l$. Notice that the number of types is now $2|\Delta|$, which is still small enough to get an EPTAS. Let us call the non-dummy jobs *regular* jobs.

**Lemma 6.** *If* $\mathrm{OPT}(I^1) \leq 1$ *and* $\overline{m}$ *is the restricted outline of an optimal solution, then there exists a solution* $\sigma^2$ *of* $I^2(\overline{m})$ *such that for any* $i$, $C^{I^2(\overline{m})}(\sigma_i^2) \leq c_i$.

*Proof.* Let $m_l^* = |\{i | t_i = l\}|$ be the number of machines with threshold $l$ in the considered optimal solution of $I^1$. Let $l \in \Delta$ be a threshold value. We first schedule $2\overline{m}_l - m_l^*$ many dummy jobs on cloned machines of $M'_l$. This will cover all dummy jobs, since

$$\sum_l [2\overline{m}_l - m_l^*] = 2 \sum_l \overline{m}_l - \sum_l m_l^* = 2m' - m.$$

We will now schedule all remaining jobs on the empty machines. For every threshold value $l$ we have $2\overline{m}_l - (2\overline{m}_l - m_l^*) = m_l^*$ many empty machines. In other words, we are left with an instance with the exact same number of machines for each threshold as in the optimal solution and with the original jobs. As argued in Lemma 3, we get the desired claim. $\square$

**Lemma 7.** *For any* $\overline{m} \in \overline{\mathcal{P}}$, *if there is a solution* $\sigma^2$ *of* $I^2(\overline{m})$ *such that* $C^{I^2(\overline{m})}(\sigma_i^2) \leq c_i + \epsilon$ *for any* $i$, *then we can deduce a solution* $\sigma^3$ *for* $I^1$ *with* $C_\Gamma^{I^1}(\sigma^3) \leq (1 + 2\epsilon)^2$.

*Proof.* We will first normalize $\sigma^2$. Since dummy jobs have $p_{ij} = c_i$ on cloned machines, in a $(1 + \epsilon)$-approximation there can only be one per machine (assuming that $\epsilon < 1$). Indeed, there may still be a load of $\epsilon \cdot c_i$ from other jobs on the same machine. We want to ensure that every machine either has a dummy job or some regular load, but not both. For every threshold value $l \in \Delta$, there can be at most $\overline{m}_l$ machines in $M'_l$ that have a dummy job. For any such machine in $M'_l$, we remove all the regular jobs (of total load of at most $\epsilon \cdot c_i$) from it and move them to one of the original machines in $M_l$, without using the same machine in $M_l$ twice. Since for any $i \in M_l$ we had $C^{I^2(\overline{m})}(\sigma_i^2) \leq (1+\epsilon)c_i$ before moving the jobs, and since regular jobs have the same processing time on machines $M_l$ and $M'_l$, after moving the jobs we get $C^{I^2(\overline{m})}(\sigma_i^2) \leq (1 + 2\epsilon)c_i$ for any $i \in M_l$. We now have a solution violating the capacities by at most $2\epsilon \cdot c_i$ such that a machine with a dummy job has no other jobs.

We now forget about all dummy jobs and the machines they are on. What we are left with is a set of $m$ machines (with some thresholds $t$) such that for any $i$ we have $C^{I^2(\overline{m})}(\sigma_i^2) \leq (1 + 2\epsilon)c_i$. By Lemma 4 we get the desired result. $\square$

All in all, we were able to reduce the number of instances created to only $2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}$ many and removed the bottleneck from the PTAS this way. This leads to the following result.

**Theorem 4.** *There is a $(1+\epsilon)$-approximation algorithm for $P|\mathcal{U}^\Gamma|C_{\max}$ running in time $O(2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}) \times f(|I|, \epsilon, O(1/\epsilon \log(1/\epsilon)))$ where $f$ is the function of Lemma 1. This implies a running time in $2^{g(\epsilon)}|I|^{O(1)}$, where $g(\epsilon) = O((1/\epsilon)^2 \log^6(1/\epsilon))$.*

*Proof.* As in Theorem 3, given an instance of $P|\mathcal{U}^\Gamma|C_{\max}$ we will use the algorithm of Lemma 1 on $I^2(m)$ for any $m \in \overline{\mathcal{P}}$. As according to Lemma 5 we have $|\overline{\mathcal{P}}| \leq 2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}$, we get a running time in $O(2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}) \times f(|I|, \epsilon, O(1/\epsilon \log(1/\epsilon)))$ where $f$ is the function of Lemma 1. This running time can be rewritten $(2^{O(1/\epsilon \log^2(1/\epsilon))} + m^{O(1)}) \times (2^{O((1/\epsilon)^2 \log^6(1/\epsilon))} + |\mathcal{I}|^{O(1)})$, implying the claimed bound. $\square$

## 6 Conclusion

In this article we have provided approximation results for $P|\mathcal{U}^\Gamma|C_{\max}$, $Q|\mathcal{U}^\Gamma|C_{\max}$ and $R|\mathcal{U}^\Gamma|C_{\max}$, improving the results obtained by Bougeret *et al.* [5]. The complexity status of $P|\mathcal{U}^\Gamma|C_{\max}$ is somehow closed as we cannot hope for an FPTAS. Yet this paper leaves open two interesting questions: closing the $[2 - \epsilon, 3]$ gap for the approximability of $R|\mathcal{U}^\Gamma|C_{\max}$, and determining whether $Q|\mathcal{U}^\Gamma|C_{\max}$ admits a PTAS.

## References

1. M. A. Aloulou and F. D. Croce. Complexity of single machine scheduling problems under scenario-based uncertainty. *Operations Research Letters*, 36(3):338 – 342, 2008.
2. A. Basu Roy, M. Bougeret, N. Goldberg, and M. Poss. Approximating robust bin-packing with budgeted uncertainty. In *Algorithms and Data Structures Symposium (WADS) 2019, August 5-7, 2019, Edmonton, Canada*, 2019.
3. D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Math. Program.*, 98(1-3):49–71, 2003.
4. D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.
5. M. Bougeret, A. A. Pessoa, and M. Poss. Robust scheduling with budgeted uncertainty. *Discrete Applied Mathematics*, 261:93–107, 2019.
6. R. L. Daniels and P. Kouvelis. Robust scheduling to hedge against processing time uncertainty in single-stage production. *Management Science*, 41(2):pp. 363–376, 1995.
7. K. Jansen, K.-M. Klein, and J. Verschae. Closing the gap for makespan scheduling via sparsification techniques. *Mathematics of Operations Research*, 45(4):1371–1392, 2020.
8. K. Jansen and M. Maack. An eptas for scheduling on unrelated machines of few different types. *Algorithmica*, 81(10):4134–4164, Oct 2019.
9. A. Kasperski, A. Kurpisz, and P. Zielinski. Approximating a two-machine flow shop scheduling under discrete scenario uncertainty. *European Journal of Operational Research*, 217(1):36–43, 2012.
10. A. Kasperski, A. Kurpisz, and P. Zielinski. Parallel machine scheduling under uncertainty. In *Advances in Computational Intelligence - 14th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2012, Catania, Italy, July 9-13, 2012, Proceedings, Part IV*, pages 74–83, 2012.

11. A. Kurpisz. Approximation schemes for robust makespan scheduling problems. In M. Lübbecke, A. Koster, P. Letmathe, R. Madlener, B. Peis, and G. Walther, editors, *Operations Research Proceedings 2014*, pages 327–333, Cham, 2016. Springer International Publishing.

12. J. K. Lenstra, D. B. Shmoys, and E. Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Mathematical programming*, 46(1-3):259–271, 1990.

13. M. Mastrolilli, N. Mutsanas, and O. Svensson. Single machine scheduling with scenarios. *Theoretical Computer Science*, 477:57 – 66, 2013.

14. B. Tadayon and J. C. Smith. Algorithms and complexity analysis for robust single-machine scheduling problems. *J. Scheduling*, 18(6):575–592, 2015.