The Santa Claus Problem in a Probabilistic Setting

Stefanie Gerke  Konstantinos Panagiotou1  Justus Schwartz1  Angelika Steger1

Institute of Theoretical Computer Science
Universitätsstrasse 6, ETH Zurich
8092 Zurich, Switzerland
{sgerke|panagiok|jschwart|steger}@inf.ethz.ch

Abstract

In this paper we consider the Santa Claus problem. Santa Claus has \( n \) presents, which he wants to give to \( m \) children. Santa’s objective is to distribute the presents such that in the end the least happy child is as happy as possible. The Santa Claus problem is equivalent to the problem of scheduling a set of \( n \) jobs on \( m \) parallel machines without preemption, so as to maximize the minimum load.

Here we study a version of that problem with the additional restriction that Santa Claus has only a guess of how valuable a present is for a child, i.e., the value of a present respectively the processing time in the scheduling terminology, is a random variable. The scheduler is given the expectation \( \lambda_i \) of the processing time of the \( i \)th job for \( 1 \leq i \leq n \), but the actual duration is not known in advance. Based on this information only the jobs have to be scheduled. In this setting we evaluate the performance of an algorithm using the competitive ratio. More precisely, we determine the value of the expected competitive ratio \( E\left[\frac{\text{Opt}}{\text{Alg}}\right] \), as a function of the \( \lambda_i \), where Opt is the optimal algorithm that knows the realizations of the random variables in advance. In particular, we show that there is a critical value \( \rho \) such that if the expected values of the processing times deviate by less than a multiplicative factor of \( \rho \) from each other, then there exists an algorithm with expected competitive ratio arbitrarily close to one, i.e., Santa Claus can perform in expectation almost as good as an algorithm who knows the actual processing times in advance. On the other hand, if the expected values of the processing times deviate much from each other, then the expected performance can become arbitrarily bad for all algorithms. Our algorithm is nearly optimal also in this case.

Keywords: Scheduling, Expected Competitive Ratio, Average-case Analysis

1 Introduction

In this work we consider the following problem, known as the Santa Claus problem (see e.g. [BS06]). Santa Claus has \( n \) presents, which he wants to give to \( m \) children. Of course, not every present is equally liked by all children. To capture this, there is a valuation function that assigns to each pair consisting of a child and a subset of presents a value that represents how valuable this subset of presents is for this particular child. Santa Claus’s objective is to distribute the presents in such a way that the least happy child is as happy as possible.

1This work was partially supported by the SNF, grant number: 200021-107880/1
In the most general setting, the valuation function assigns for each child arbitrary real numbers to every possible subset of presents, and hence \( m^n \) values are needed to describe it. In a more restricted but realistic setting, the valuation function is additive, i.e., the value of a set equals the sum of the values of its items. In this paper we deal only with additive valuation functions.

The Santa Claus problem has been studied in the literature under many different names. In the remainder we shall adopt the scheduling terminology: the Santa Claus problem is equivalent to the problem of scheduling a set of \( n \) jobs on \( m \) parallel machines without preemption, so as to maximize the minimum machine completion time without introducing idle times. In other words the goal is to partition the jobs into \( m \) subsets such that the sum of the jobs in the subset with the minimum sum becomes maximum. This situation is motivated by various scenarios. The original application was the sequencing of maintenance actions for a fleet of modular gas turbine aircraft engines, see [DFL82].

1.1 Previous Work

For \( 1 \leq i \leq n \) and \( 1 \leq j \leq m \), let the processing time of the \( i \)th job on the \( j \)th machine be denoted by \( p_{ij} \). In the most simple case, when \( p_{ij} = p_i \) for all \( i \) and \( j \) (this is sometimes denoted as the identical machines case), the Santa Claus problem is well-studied. Already this case is \( \mathcal{NP} \)-hard in the strong sense [GJ79], and there is a polynomial time approximation scheme by Woeginger [Woe97]. Deuermeyer, Friesen and Langston [DFL82] proved that the well-known Longest Processing Time First (LPTF) algorithm, which sorts all jobs in a non-increasing order and assigns each job to the next available machine, has worst case approximation ratio of at most \( \frac{4}{3} \). This result was tightened by Csirik, Kellerer and Woeginger [CKW92], who showed that LPTF remains always within a factor of \( \frac{3m-2}{3m-1} \) of the optimum solution.

For the restricted assignment case, where \( p_{ij} \in \{ p_i, 0 \} \) for all admissible \( i \) and \( j \) (i.e. in the Santa Claus context present \( i \) has either value \( p_i \) or no value for kid \( j \)), Bansal and Svirdenko [BS06] gave an \( \mathcal{O} \left( \frac{\log \log m}{\log \log \log m} \right) \) approximation algorithm, which considerably improved an \( \mathcal{O}(\sqrt{m}) \) approximation algorithm designed by Golovin [Gol05]. For the general problem (i.e. \( p_{ij} \) arbitrary) Bezáková and Dani [BD05] developed two algorithms with approximation ratios \( n - m + 1 \) and \( \text{Opt} - p_{\text{max}} \), where \( p_{\text{max}} := \max_{i,j} p_{ij} \). Here \( \text{Opt} \) denotes the objective value of an optimal algorithm.

In an online setting, where jobs arrive one by one and we have to decide immediately to which machine we assign them, it is known that for the case of identical machines the greedy algorithm is \( m \)-competitive and best possible [Woe97], i.e., no deterministic online algorithm can do better if we use the competitive ratio as a performance measure. On the other hand, Azar and Epstein [AE05] designed an \( \mathcal{O}(\sqrt{m} \log m) \)-competitive randomized algorithm, and proved that every randomized algorithm has competitive ratio \( \Omega(\sqrt{m}) \). Furthermore, they give a randomized algorithm that has a competitive ratio of \( \mathcal{O}(\ln R) \), where \( R \) is the ratio between the largest and the smallest processing time. The case of uniform machines, where \( p_{ij} = p_is_j \) for given scaling parameters \( s_1, \ldots, s_m \), was studied by Tan, He and Epstein in [THE05], who developed, among other results, an optimal online algorithm for the case of two machines. More precisely, they considered the case where the jobs arrive in non-decreasing order, i.e. the absolute processing times are not known in advance.
1.2 Model & Results

Scheduling problems were among the first optimization problems to be studied. However, usually the worst-case perspective is considered, which often does not reflect that a scheduling algorithm may perform well in practice. A natural step to overcome this drawback is to consider stochastic scheduling, i.e., to interpret the problem parameters as random variables and to measure the performance by means of expected values.

In this paper we study the Santa Claus problem in a probabilistic setting. More precisely, we assume that the expected processing time for each job on each machine is known but the actual duration is unknown. The aim of the scheduler is to determine a schedule of the jobs before their actual duration becomes known, while trying to maximize the minimum resulting load. We have to fix a schedule in advance and have a guess on how long each job will take on different machines, but we cannot be sure about the actual processing time. This is relevant in situations where one does not know the actual processing time in advance, but has a good estimate from prior experience. Also, coming back to Santa Claus, everyone who has ever given a dearly wished present of the wrong colour to a child, will agree that we can never be sure with presents either.

We consider the identical machine case. Our scheduling problem is given by a set $J$ of $n$ jobs and a set $M$ of $m$ machines. The processing time $X_{ij}$ of the $i$th job on the $j$th machine equals $X_i$, where $X_i$ is an exponentially distributed random variable with mean $\lambda_i$. Let $\Lambda := (\lambda_1, \ldots, \lambda_n)$, $P(\Lambda) := (X_1, \ldots, X_n)$ and let $P := P(\Lambda)$. In this setting a scheduling algorithm $\text{ALG}$ is given $\Lambda$ and computes a partition $I_1, \ldots, I_m$ of the set $\{1, \ldots, n\}$. The value of $\text{ALG}$ is then $\text{Alg}(\Lambda, P) = \min_{1 \leq j \leq m} \sum_{i \in I_j} X_i$.

As a performance measure for a given scheduling algorithm $\text{ALG}$ we use the expected competitive ratio, which is defined as follows. Let $\text{OPT}$ be an optimal, offline algorithm that knows the actual realization of the $X_i$’s in advance. In the above described setting, $\text{Alg}(\Lambda, P)$, and $\text{OPT}(P)$ are random variables, and the expected competitive ratio of $\text{ALG}$ is defined by

$$\text{ECR}(\text{ALG}; \Lambda) := \mathbb{E}\left[\frac{\text{OPT}(P)}{\text{Alg}(\Lambda, P)}\right]. \quad (1.1)$$

This model comprises important aspects of both classical competitive analysis and stochastic scheduling models. In stochastic scheduling models, one usually tries to develop an algorithm that minimizes/maximizes the expected value of a given target function — hence an absolute value is optimized with respect to the chosen probability distribution. On the other hand, in competitive analysis the input to an algorithm is not known in advance, but becomes available gradually and can be chosen by an adversary. Hence, the competitive ratio of an algorithm is a worst-case notion, which relates the value of the algorithm to the value of an optimal, offline algorithm. Here we profit from both models: we relate on all instances the values of $\text{ALG}$ and $\text{OPT}$, and weight the resulting ratio with the probability of the occurrence of the instance. This performance measure was introduced in the mid-80’s by Coffman and Gilbert [CG85], and was frequently used to determine in a stochastic framework the performance of several scheduling algorithms and heuristics. We mention selectively [LV04], where the problem of optimizing the time for loading and unloading containers was considered, and [SSS06] and [SS06], where the authors analyzed the performance of a heuristic for the (weighted) completion time scheduling problem.

Before we proceed with presenting the results, let us start with a preliminary discussion.
of $\lambda_i = \cdots = \lambda_n$, then there is a straightforward algorithm that has the property $\text{ECR} (\text{ALG}; \Lambda) \sim 1$: As the actual processing times of the jobs are independent identically distributed random variables, partitioning the jobs in $m$ equal sized parts yields by fairly standard concentration results that the objective function of the algorithm satisfies $\text{ALG} \sim \sum_{i=1}^{\infty} \lambda_i/m$ with high probability. On the other hand we always have that $\text{OPT} \leq \sum_{i=1}^{\infty} \lambda_i/m \sim \text{ALG}$ with high probability, which implies $\text{OPT}/\text{ALG} \sim 1$.

An important ingredient of the preceding analysis is that the processing times are identically distributed. In contrast, if the $\lambda_i$ differ, then the variance of the total processing time on a machine may be dominated by only very few jobs. The question is for which “amount” of variance an algorithm can still achieve a good expected competitive ratio or if there is a critical “amount” for which no (online) algorithm behaves well. The purpose of this paper is to resolve this question.

Theorem 1.1. Let $2 \leq m = m(n) \leq \frac{n}{8 \ln n}$. There exists an $n_0$ such that for all $n > n_0$ the following is true.

1. There exists an algorithm $\text{Greedy}$ such that for all $\Lambda = (\lambda_1, \ldots, \lambda_n)$ satisfying $R(\Lambda) \ll \frac{n}{m \ln m}$

$$\text{ECR} (\text{Greedy}; \Lambda) = \mathbb{E} \left[ \frac{\text{OPT} (\mathcal{P})}{\text{Greedy} (\Lambda, \mathcal{P})} \right] = 1 + o(1) \xrightarrow{n \to \infty} 1.$$ 

2. There is a $\xi > 0$ such that for all $r \geq \xi$ there exists a $c = c(r) > 0$ and an instance $\Lambda = (\lambda_1, \ldots, \lambda_n)$ with $R(\Lambda) = r \cdot \frac{n}{m \ln m}$, such that for any algorithm $\text{Any}$ we have

$$\text{ECR} (\text{Any}; \Lambda) = \mathbb{E} \left[ \frac{\text{OPT} (\mathcal{P})}{\text{Any} (\Lambda, \mathcal{P})} \right] \geq 1 + c.$$

Furthermore we have that $c(r) \to \infty$ if $r$ tends to infinity.

This behavior is substantially different from the behavior of the expected competitive ratio for similar scheduling problems, as for instance for the minimum makespan scheduling problem and the weighted completion time scheduling problem. Indeed, for the minimum makespan problem, Coffman and Gilbert showed $\text{CGS5}$ that the LEPT rule (Longest Expected Processing Time First) achieves an expected competitive ratio which approaches 1 when the number of jobs increases. Souza and Steger $\text{SST06}$ showed, among other results, that the popular heuristic WSEPT (Weighted Shortest Expected Processing Time First) has expected competitive ratio at most $3 - \frac{3}{m}$, independent of the longest expected processing time that occurs. In both cases there exist algorithms with the property that their competitive ratio is bounded by some absolute constant. Therefore, from an average-case perspective, our problem behaves substantially different than related scheduling problems: the uncertainty in the input parameters may make the problem much more difficult for the scheduler.
Note that the above theorem requires the number of machines to satisfy the constraint $m \leq \frac{\ln n}{\ln n}$, which is up to the constant factor best possible due to the following reason. Let $\Lambda = (1, \ldots, 1)$ with $R(\Lambda) = 1$. If $m > \frac{2n}{\ln n}$, then in every scheduling there is a constant fraction of machines with $\frac{3\ln n}{4}$ jobs assigned to each one of them. By applying standard concentration results and the fact that the sum of independent $\text{Exp}(1)$ variables is Gamma distributed, we readily see that the minimum and the maximum load on those machines differ significantly, while an optimal algorithm can distribute the load almost evenly on all machines. Hence, the expected competitive ratio becomes arbitrarily bad.

We further study the behavior of the expected competitive ratio for $R(\Lambda) = \Omega \left( \frac{n}{m \log m} \right)$ in more detail and give almost matching lower and upper bounds. In particular, we show that the algorithm Greedy is essentially best possible in the sense that it achieves an expected competitive ratio on all possible instances that almost matches the lower bounds. In Section 5 we present two lemmas which on the one hand prove the second part of Theorem 1.1 and on the other hand give precise lower bounds on the expected competitive ratio of any algorithm depending on $R$. In Section 6 we will show that the algorithm Greedy matches these lower bounds up to logarithmic factors.

2 Preliminaries

In this section we state well-known results on random variables that we shall use later. Furthermore we establish two new bounds for sums of independent exponential variables with different means (Lemmas 2.5 and 2.6 below), which we believe are of independent interest. In the following we will write $X \sim F$ if the random variable $X$ has distribution $F$. Also we will write $\text{Exp}(\lambda)$ for the exponential distribution with mean $\lambda$ and $\text{Uni}(a, b)$ for the uniform distribution on the interval $[a, b]$.

Our first lemma is a fact about the order statistics of exponentially distributed random variables, which is a straightforward consequence of the memoryless property of the exponential distribution.

**Lemma 2.1.** Let $X_1, \ldots, X_n$ be independent exponentially distributed random variables with $E[X] = \lambda$. Then the $k$-th order statistic $X_{(k:n)}$, that is the $k$-th smallest of the $X_1, \ldots, X_n$, is distributed as the sum of independent exponentially distributed variables $\sum_{i=0}^{k-1} \text{Exp}\left(\frac{\lambda}{n-i}\right)$.

Furthermore, we shall use the following inequalities for the tails of the binomial distribution, which can for instance be found in [McD98].

**Lemma 2.2.** Let $X \sim \text{Bin}(n, p)$. For all $t \geq 0$,

$$\Pr[X \geq E[X] + t] \leq e^{-\frac{2t^2}{2E[X] + t}} \quad \text{and} \quad \Pr[X \leq E[X] - t] \leq e^{-\frac{2t^2}{2E[X]}}.$$  

As an extension of the above lemma we will use the following result, which is a corollary of the Independent Bounded Difference Inequality, see again for example [McD98].

**Lemma 2.3.** Let the random variables $X_1, \ldots, X_n$ be independent, with $0 \leq X_i \leq \lambda_i$ for each $1 \leq i \leq n$. Let $X = \sum_{1 \leq i \leq n} X_i$ and set $\lambda := E[X]$. Then for any $t \geq 0$,

$$\Pr[|X - \lambda| \geq t] \leq 2e^{-\frac{2t^2}{\sum_{i=1}^{n} \lambda_i^2}}.$$
For the uniform distribution we have the following lemma, see e.g. [Old52].

**Lemma 2.4.** Let the random variables \( Y_1, \ldots, Y_n \) be independent and uniformly distributed with \( Y_i \sim \text{Uni}(0, \lambda_i) \). Let \( Y = \sum_{1 \leq i \leq n} Y_i \). Then for any \( t \geq 0 \),
\[
\Pr[Y \leq t] \leq \frac{t^n}{n! \prod_{i=1}^n \lambda_i}.
\]

In fact, the cumulative distribution function of a sum of independent uniform random variables is known exactly, see Olds [Old52]. However, the above lemma suffices for our purposes.

In the following two lemmas we will make use of the moment generating function for the exponential distribution, which we will state here briefly. Let \( X \sim \text{Exp}(\lambda) \) be an exponentially distributed random variable with mean \( \lambda \). We consider the moment generating function
\[
E[e^{tX}] = \int_0^\infty e^{tx} \frac{1}{\lambda} e^{-x/\lambda} dx = \frac{1}{(1-t\lambda)}.
\]

For the random variable \( Y = -X \) and \( t > 0 \) the moment generating function is given by
\[
E[e^{tY}] = \int_0^\infty e^{-tx} \frac{1}{\lambda} e^{-x/\lambda} dx = \frac{1}{(1+t\lambda)}.
\]

To give bounds for the upper tail of the sum of exponential random variables we will use the moment generating function and Markov’s inequality.

**Lemma 2.5.** Let the random variables \( X_1, \ldots, X_n \) be independent and exponentially distributed with \( E[X_i] = \lambda_i \) where \( \lambda_{\max} \geq \lambda_1 \geq \cdots \geq \lambda_n \geq 1 \). Let \( X = \sum_{i=0}^n X_i \) and set \( \lambda := \sum_{i=0}^n \lambda_i \). Then we have for \( \varepsilon > 0 \)
\[
\Pr[X \geq (1 + \varepsilon)\lambda] \leq e^{-\frac{\varepsilon^2}{2(1 + \varepsilon))}\frac{\lambda}{\lambda_{\max}}.
\]

**Proof.** Using the moment generating function (2.1) and Markov’s Inequality we obtain for \( 0 < t < \frac{1}{\lambda_{\max}} \)
\[
\Pr[X \geq (1 + \varepsilon)\lambda] = \Pr\left[ e^{tX} \geq e^{t(1+\varepsilon)\lambda} \right] \\
\leq \frac{E[e^{tX}]}{e^{t(1+\varepsilon)\lambda}} = \frac{\prod_{i=0}^n E[e^{tX_i}]}{e^{t(1+\varepsilon)\lambda}} = e^{-t(1+\varepsilon)\lambda} \prod_{i=0}^n (1 - t\lambda_i)^{-1}.
\]

Basic transformations yield
\[
\Pr[X \geq (1 + \varepsilon)\lambda] \leq \exp\left( -t(1 + \varepsilon) \sum_{i=1}^n \lambda_i \right) \exp\left( -\sum_{i=1}^n \ln(1 - t\lambda_i) \right) = \exp\left( \sum_{i=1}^n (-t(1 + \varepsilon)\lambda_i - \ln(1 - t\lambda_i)) \right)
\]

6
and using the series expansion of $\ln(1 - x)$ we have

$$\Pr [X \geq (1 + \varepsilon)\lambda] \leq \exp \left( \sum_{i=1}^{n} \left( -t(1 + \varepsilon)\lambda_i + \sum_{j \geq 1} \frac{(t\lambda_j)^j}{j} \right) \right).$$

(2.3)

And further for each term $z_i, 1 \leq i \leq n,$ of the above sum we have that

$$z_i = -t(1 + \varepsilon)\lambda_i + \sum_{j \geq 1} \frac{(t\lambda_j)^j}{j} \leq -\varepsilon t\lambda_i + \sum_{j \geq 2} \frac{(t\lambda_j)^j}{2} = -\varepsilon t\lambda_i + \frac{(t\lambda_i)^2}{2} \frac{1}{1 - t\lambda_i}.$$

We now set $t = \frac{\varepsilon}{1 + \varepsilon \lambda_{\text{max}}}$ and have as required $t < \frac{1}{\lambda_{\text{max}}}$ for all $1 \leq i \leq n$. It follows

$$z_i \leq -\frac{\varepsilon^2}{1 + \varepsilon \lambda_{\text{max}}} \lambda_i + \frac{1}{2} \frac{\varepsilon^2}{(1 + \varepsilon)^2} \frac{\lambda_i^2}{\lambda_{\text{max}}} \cdot \frac{1}{1 - \frac{\varepsilon}{1 + \varepsilon} \lambda_i}.$$

Using that $\lambda_i/\lambda_{\text{max}} \leq 1$ we get

$$z_i \leq -\frac{\varepsilon^2}{1 + \varepsilon \lambda_{\text{max}}} \lambda_i + \frac{1}{2} \frac{\varepsilon^2}{(1 + \varepsilon)^2} \frac{\lambda_i}{\lambda_{\text{max}}} \cdot (1 + \varepsilon) \leq -\frac{\varepsilon^2}{2} \frac{1}{1 + \varepsilon \lambda_{\text{max}}} \lambda_i.$$

Therefore by (2.3) we obtain

$$\Pr [X \geq (1 + \varepsilon)\lambda] \leq \exp \left( -\sum_{i} \frac{\lambda_i}{\lambda_{\text{max}}} \frac{\varepsilon^2}{2(1 + \varepsilon)} \right) = e^{-\frac{\varepsilon^2}{2(1 + \varepsilon) \lambda_{\text{max}}}}.$$

For the lower tail of the sum of exponential random variables we will use the following lemma.

**Lemma 2.6.** Let $X_1, \ldots, X_n$ be independent exponentially distributed random variables with $\mathbb{E} [X_i] = \lambda_i,$ where $\lambda_{\text{max}} \geq \lambda_1 \geq \cdots \geq \lambda_n \geq 1$. Let $X = \sum_{i=1}^{n} X_i$ and set $\lambda := \sum_{i=1}^{n} \lambda_i$. Then we have for $0 < \alpha < 1$,

$$\Pr [X \leq \alpha \lambda] \leq (\alpha e^2)^{\lambda_{\text{max}}},$$

and for $0 < \varepsilon < 1$

$$\Pr [X \leq (1 - \varepsilon)\lambda] \leq e^{-\varepsilon^2 \lambda_{\text{max}}}.$$

**Proof.** With Markov’s inequality we obtain for all $t > 0$ that

$$\Pr [X \leq \alpha \lambda] = \Pr [-X \geq -\alpha \lambda] = \Pr \left[ e^{-tX} \geq e^{-\alpha t \lambda} \right] \leq \mathbb{E} \left[ e^{-tX} \right] e^{\alpha t \lambda}.$$

We have from (2.2) that for $t \geq 0$ and $1 \leq i \leq n$ the moment generating function is $\mathbb{E} \left[ e^{-tX_i} \right] = \frac{1}{1 + t \lambda_i}$. Note that contrary to the moment generating function of the exponential distribution
used in Lemma 2.5 we do not have the restriction \( t \leq \frac{1}{\lambda_i} \). Due to the independence of the \( X_i \)'s we derive

\[
\Pr [X \leq \alpha \lambda] \leq e^{t \alpha \lambda} \cdot \prod_{i=1}^{n} \mathbb{E} \left[ e^{-t X_i} \right] = e^{t \alpha \lambda} \cdot \prod_{i=1}^{n} \frac{1}{1 + t \lambda_i}.
\]  

(2.4)

To prove the first inequality of the lemma we chose \( t = \frac{1}{\alpha \lambda_{\text{max}}} \). This yields

\[
\Pr [X \leq \alpha \lambda] \leq e^{\frac{\alpha \lambda}{\lambda_{\text{max}}}} \cdot \prod_{i=1}^{n} \frac{1}{1 + \frac{\lambda_i}{\alpha \lambda_{\text{max}}}}.
\]  

(2.5)

To obtain a good upper bound for (2.5) we will use several bounds, which depend on the distribution of the values of the \( \lambda_i \)'s. For this, we define the following three sets.

\[
S_1 := \left\{ i \mid \frac{\lambda_i}{\lambda_{\text{max}}} < \alpha \right\}, \quad S_2 := \left\{ i \mid \alpha \leq \frac{\lambda_i}{\lambda_{\text{max}}} \leq \sqrt{\alpha} \right\}, \quad S_3 := \left\{ i \mid \sqrt{\alpha} < \frac{\lambda_i}{\lambda_{\text{max}}} \leq 1 \right\}.
\]

Due to \( 1 + x \geq e^x \), valid for \( 0 \leq x \leq 1 \), we have for \( \frac{\lambda_i}{\lambda_{\text{max}}} < \alpha \)

\[
\frac{1}{1 + \frac{\lambda_i}{\alpha \lambda_{\text{max}}}} \leq e^{-\frac{\lambda_i}{2 \alpha \lambda_{\text{max}}}}.
\]  

(2.6)

For \( \frac{\lambda_i}{\lambda_{\text{max}}} \geq \alpha \) we have

\[
\frac{1}{1 + \frac{\lambda_i}{\alpha \lambda_{\text{max}}}} \leq \frac{1}{1 + \frac{\alpha}{\alpha}} = \frac{1}{2}
\]  

(2.7)

and for \( \frac{\lambda_i}{\lambda_{\text{max}}} > \sqrt{\alpha} \) we have

\[
\frac{1}{1 + \frac{\lambda_i}{\alpha \lambda_{\text{max}}}} \leq \frac{1}{1 + \sqrt{\alpha}} \leq \sqrt{\alpha}.
\]  

(2.8)

For \( i = 1, 2, 3 \), we define \( s_i := \sum_{j \in S_i} \lambda_j \) and note that \( s_1 + s_2 + s_3 = 1 \). Furthermore, observe that by the definition of the set \( S_1 \)

\[
|S_1| \cdot \alpha \lambda_{\text{max}} \geq \sum_{j \in S_1} \lambda_i = s_1 \lambda
\]

and therefore \( |S_1| \geq \frac{s_1 \lambda}{\alpha \lambda_{\text{max}}} \). Similarly \( |S_2| \geq \frac{s_2 \lambda}{\alpha \lambda_{\text{max}}} \) and \( |S_3| \geq \frac{s_3 \lambda}{\lambda_{\text{max}}} \) by the definition of the sets \( S_i \). Now using inequalities (2.6) - (2.8) we obtain from (2.5) that

\[
\Pr [X \leq \alpha \lambda] \leq e^{\frac{\lambda}{\lambda_{\text{max}}}} \cdot \prod_{i=1}^{n} \frac{1}{1 + \frac{\lambda_i}{\alpha \lambda_{\text{max}}}} \leq e^{\frac{\lambda}{\lambda_{\text{max}}}} \left( \prod_{i \in S_1} e^{-\frac{\lambda_i}{2 \alpha \lambda_{\text{max}}}} \right) \left( \frac{1}{2} \right)^{|S_2|} \left( \sqrt{\alpha} \right)^{|S_3|} \leq e^{\frac{\lambda}{\lambda_{\text{max}}}} \left( \prod_{i \in S_1} e^{-\frac{\lambda_i}{2 \alpha \lambda_{\text{max}}}} \right) \left( \frac{1}{2} \right)^{s_2 \frac{\lambda}{\alpha \lambda_{\text{max}}} \sqrt{\alpha}} \left( \sqrt{\alpha} \right)^{s_3 \frac{\lambda}{\lambda_{\text{max}}}} = e^{\frac{\lambda}{\lambda_{\text{max}} \alpha}} \cdot e^{-s_1 \frac{\lambda}{2 \lambda_{\text{max}} \alpha}} \cdot 2^{-s_2 \frac{\lambda}{\alpha \lambda_{\text{max}} \alpha}} \cdot \alpha^{s_3 \frac{\lambda}{\lambda_{\text{max}} \alpha}}.
\]
As \( \log_b \alpha > -\frac{1}{\alpha} \) for \( 0 < \alpha < 1 \) and both \( b = e \) and \( b = 2 \) we may bound this expression from above with

\[
\Pr[X \leq \alpha \lambda] \leq e^{\frac{\lambda}{\lambda_{\text{max}}} \cdot \frac{\lambda}{\lambda_{\text{max}}} \cdot \ln \alpha \cdot \frac{\lambda}{\lambda_{\text{max}}} \cdot \log_2 \alpha \cdot \frac{\lambda}{\lambda_{\text{max}}}} = \left( \alpha^{\frac{s_1 + s_2 + s_3}{e}} \right)^{\frac{\lambda}{\lambda_{\text{max}}}} = (\alpha e^{2})^{\frac{\lambda}{\lambda_{\text{max}}}},
\]

which proves the first inequality of the lemma. For the proof of the second inequality we start from (2.4):

\[
\Pr[X \leq (1 - \varepsilon) \lambda] \leq e^{t(1 - \varepsilon) \lambda} \prod_{i=1}^{n} \frac{1}{1 + t \lambda_i} = \exp \left( \sum_{i=1}^{n} t(1 - \varepsilon) \lambda_i - \ln(1 + t \lambda_i) \right)
\]

Using that \( (1 + x) \geq e^{x - x^2/2} \) for \( 0 \leq x \leq 1 \) we have for \( 1 \leq i \leq n \) that

\[
y_i = t(1 - \varepsilon) \lambda_i - \ln(1 + t \lambda_i) \leq t(1 - \varepsilon) \lambda_i - \ln \left( e^{t \lambda_i - t^2 \lambda_i^2 / 2} \right) = t \lambda_i - t \varepsilon \lambda_i - \left( t \lambda_i - t^2 \lambda_i^2 / 2 \right).
\]

Setting \( t = \frac{\varepsilon}{\lambda_{\text{max}}} \) we arrive at

\[
y_i \leq -\varepsilon^2 \frac{\lambda_i}{\lambda_{\text{max}}} + \frac{\varepsilon^2}{2} \left( \frac{\lambda_i}{\lambda_{\text{max}}} \right)^2
\]

and since \( \frac{\lambda_i}{\lambda_{\text{max}}} \leq 1 \) we have that

\[
y_i \leq -\varepsilon^2 \frac{\lambda_i}{\lambda_{\text{max}}} + \frac{\varepsilon^2}{2} \frac{\lambda_i}{\lambda_{\text{max}}} = -\frac{\varepsilon^2 \lambda_i}{2 \lambda_{\text{max}}}.
\]

By inserting this into (2.9) the second inequality of the lemma follows:

\[
\Pr[X \leq (1 - \varepsilon) \lambda] \leq \exp \left( -\sum_{i=1}^{n} \frac{\varepsilon^2 \lambda_i}{2 \lambda_{\text{max}}} \right) = e^{-\frac{\varepsilon^2 \lambda}{2 \lambda_{\text{max}}} \text{.}}
\]

To facilitate the calculation of (bounds on) expectations of products or fractions we will use the FKG inequality in the following form. As this differs from the most commonly stated form of FKG for functions on finite lattices we will also include the proof.

**Theorem 2.7 (FKG inequality).** Let \( f : \mathbb{R}^n \mapsto \mathbb{R}^+ \) be an increasing function and \( g : \mathbb{R}^n \mapsto \mathbb{R}^+ \) a decreasing function. That is if \( x = (x_1, \ldots, x_n) \in \mathbb{R}^n \) and \( y = (y_1, \ldots, y_n) \in \mathbb{R}^n \) and \( x_i \leq y_i, \) \( 1 \leq i \leq n \) then \( f(x) \leq f(y) \) and \( g(x) \geq g(y) \). Let \( X_1, \ldots, X_n \) be independent random variables and \( X = (X_1, \ldots, X_n) \). Then

\[
\mathbb{E}[f(X)g(X)] \leq \mathbb{E}[f(X)] \mathbb{E}[g(X)].
\]
Proof. The proof is by induction over the dimension $n$. For the base of the induction where $n = 1$ we have for $x, y \in \mathbb{R}$ that $(f(x) - f(y)) (g(x) - g(y)) \leq 0$. Thus, for two independent identically distributed random variables $X, Y$ on $\mathbb{R}$, we have that

$$
0 \geq \mathbb{E} [(f(X) - f(Y)) \cdot (g(X) - g(Y))]
= \mathbb{E} [f(X)g(X) + f(Y)g(Y) - f(Y)g(X) - f(X)g(Y)]
= (\mathbb{E} [f(X)g(X)] + \mathbb{E} [f(Y)g(Y)]) - (\mathbb{E} [f(Y)g(X)] + \mathbb{E} [f(X)g(Y)]).
$$

Since $X$ and $Y$ are identically distributed and independent random variables we have

$$
\mathbb{E} [f(X)g(X)] = \mathbb{E} [f(Y)g(Y)] \quad \text{and} \quad \mathbb{E} [f(Y)g(X)] = \mathbb{E} [f(X)g(Y)] = \mathbb{E} [f(X)] \mathbb{E} [g(X)].
$$

Therefore,

$$
0 \geq 2 \mathbb{E} [f(X)g(X)] - 2 \mathbb{E} [f(X)] \mathbb{E} [g(X)],
$$

thus the inequality is true for $n = 1$.

Now we suppose that the claim is true for $n - 1 \geq 1$. If we fix $X_n$ to be $x_n$ then we can use our induction hypothesis to obtain

$$
\mathbb{E} [f(X)g(X) \mid X_n = x_n] = \mathbb{E} [f(X_1, \ldots, X_{n-1}, x_n)g(X_1, \ldots, X_{n-1}, x_n)]
\leq \mathbb{E} [f(X_1, \ldots, X_{n-1}, x_n)] \mathbb{E} [g(X_1, \ldots, X_{n-1}, x_n)]
= \mathbb{E} [f(X) \mid X_n = x_n] \mathbb{E} [g(X) \mid X_n = x_n].
$$

Observe that $\mathbb{E} [f(X) \mid X_n = x_n]$ is an increasing function in $x_n$ and $\mathbb{E} [g(X) \mid X_n = x_n]$ is a decreasing function in $x_n$ and thus we may apply the base case of the induction for $n = 1$. Now we take the expectation on both sides and apply the base case of the induction.

$$
\mathbb{E} [\mathbb{E} [f(X)g(X) \mid X_n]] \leq \mathbb{E} [\mathbb{E} [f(X) \mid X_n] \mathbb{E} [g(X) \mid X_n]]
\leq \mathbb{E} [\mathbb{E} [f(X) \mid X_n]] \mathbb{E} [\mathbb{E} [g(X) \mid X_n]]
= \mathbb{E} [f(X)] \mathbb{E} [g(X)],
$$

which proves the theorem.

Our calculations for the expectations will often lead to expressions including the exponential integral $E(x)$. There we will make use of the following approximations of the exponential integral.

**Lemma 2.8.** Let $x > 0$. The exponential integral $E(x)$

$$
E(x) := \int_x^\infty \frac{e^{-t}}{t} dt
$$

satisfies the inequalities

$$
\frac{1}{2} \ln \left(1 + \frac{2}{x}\right) \leq e^x E(x) \leq \ln \left(1 + \frac{1}{x}\right).
$$
Proof. To prove the right part of the inequality we consider the function

\[ f_1(t) = e^{-t} \left( \frac{1}{t^2 + t} + \ln(1 + \frac{1}{t}) \right). \]

The function \( f_1 \) is chosen such that

\[ \int_{x}^{\infty} f_1(t) dt = e^{-x} \ln \left( 1 + \frac{1}{x} \right). \]

We will show that \( \frac{e^{-t}}{t} \leq f_1(t) \) for all \( t > 0 \) and thus \( \int_{x}^{\infty} \frac{e^{-t}}{t} dt \leq \int_{x}^{\infty} f_1(t) dt = e^{-x} \ln(1 + \frac{1}{x}) \) which proves the right part of the inequality of the lemma. We use that for \( 0 < x < 1 \) we have \( e^{x} \leq 1 + x + x^2 \) and therefore \( e^{\frac{1}{t+1}} \leq 1 + \frac{1}{t+1} + \frac{1}{(t+1)^2} \leq 1 + \frac{1}{t} \) for all \( t > 0 \). We have for \( t > 0 \) that

\[ \frac{e^{-t}}{t} = e^{-t} \left( \frac{1}{t+1} + \frac{1}{t^2 + t} \right) = e^{-t} \left( \ln \left( \exp \left( \frac{1}{t+1} \right) + \frac{1}{t^2 + t} \right) \right) \]

and using that \( \exp \left( \frac{1}{t+1} \right) \leq 1 + \frac{1}{t} \) we obtain

\[ \frac{e^{-t}}{t} \leq e^{-t} \left( \ln \left( 1 + \frac{1}{t+1} \right) + \frac{1}{t^2 + t} \right) = f_1(t), \]

which proves the right part of the inequality. In order to prove the lower bound we consider the function \( f_2(t) = e^{-t} \left( \frac{1}{t^2 + 2t} + \frac{1}{2} \ln \left( 1 + \frac{2}{t} \right) \right) \) which is chosen such that

\[ \int_{x}^{\infty} f_2(t) dt = e^{-x} \frac{1}{2} \ln \left( 1 + \frac{2}{t} \right). \]

We show for all \( t \geq 0 \) that \( f_2(t) \leq \frac{e^{-t}}{t} \) to prove the lower bound. For all \( t > 0 \) we have

\[ \frac{e^{-t}}{t} = e^{-t} \left( \frac{1}{t} - \frac{1}{t^2 + 2t} + \frac{1}{t^2 + 2t} \right) = e^{-t} \left( \frac{2(t+1)}{t^2 + 2t} \right) \]

\[ = e^{-t} \left( \frac{2(t+1)}{t^2 + 2t} \right) \]

\[ = e^{-t} \left( \frac{1}{2} \ln \left( \frac{2(t+1)}{t^2 + 2t} \right) + \frac{1}{t^2 + 2t} \right). \]

Using the first three elements of the series expansion of the exponential function we have

\[ \exp \left( \frac{2(t+1)}{t^2 + 2t} \right) > 1 + \frac{(t+1)^2}{t(t+2)} + \frac{1}{2} \left( \frac{(t+1)^2}{t(t+2)} \right)^2 = 1 + \frac{2}{t} \cdot \frac{t^3 + 4t^2 + 4t + 1}{t^4 + 4t^3 + 4t} \geq 1 + \frac{2}{t} \]

and therefore we have the following upper bound on the expression above:

\[ \frac{e^{-t}}{t} \geq e^{-t} \left( \frac{1}{2} \ln \left( 1 + \frac{2}{t} \right) + \frac{1}{t^2 + 2t} \right) = f_2(t), \]

which proves the left part of the inequality.
3 The Algorithm

In this section we will introduce the algorithm Greedy. The algorithm Greedy has two important properties which we state in Lemma 3.1.

The input for our algorithm is \( \Lambda = (\lambda_1, \ldots, \lambda_n) \in \mathbb{R}_+ \) and an integer \( m \). Without loss of generality we may assume that \( \lambda_{\text{max}} = \lambda_1 \geq \cdots \geq \lambda_n \). The output is a partition \( I_1, \ldots, I_m \) of the set \( \{1, \ldots, n\} \). The algorithm Greedy considers the values one by one in non-increasing order and assigns them always to the next set wrapping around at \( m \), see Figure 1 for an illustration. More precisely, Greedy generates the partition

\[
I_j = \{i : (i - 1) \mod m = j - 1 \text{ and } 1 \leq i \leq n\}, \quad 1 \leq j \leq m.
\]

Figure 1: A partition produced by Greedy for \( m = 4 \) and the first 9 \( \lambda_i \).

The next lemma states two important properties of the algorithm.

**Lemma 3.1.** Let \( \lambda_{\text{max}} := \lambda_1 \geq \cdots \geq \lambda_n \) and \( m \leq n \). Then Greedy computes a partition \( I_1, \ldots, I_m \) of the set \( \{1, \ldots, n\} \) such that

\[
|I_j| \geq \left\lfloor \frac{n}{m} \right\rfloor. \quad (3.1)
\]

Moreover for \( 1 \leq i, j \leq m \) let \( \mu_j := \sum_{i \in I_j} \lambda_i \) and \( \mu := \sum_{1 \leq i \leq n} \lambda_i \). Then

\[
|\mu_i - \mu_j| \leq \lambda_{\text{max}}, \quad \frac{\mu}{m} - \lambda_{\text{max}} \leq \mu_j \leq \frac{\mu}{m} + \lambda_{\text{max}}. \quad (3.2)
\]

**Proof.** By the definition of Greedy the inequality (3.1) is immediate. To simplify the proof of (3.2) we set \( \lambda_t = 0 \) for all \( t > n \). Observe that \( \mu_1 \geq \cdots \geq \mu_m \) and further that \( \lambda_{km+1} - \lambda_{km+m} \leq \lambda_{km+1} - \lambda_{km+m+1} \) for \( 0 \leq k \leq \frac{n}{m} \). Therefore we have

\[
\mu_1 - \mu_m = \sum_{i \in I_1} \lambda_i - \sum_{i \in I_m} \lambda_i = \sum_{0 \leq k \leq \frac{n}{m}} (\lambda_{km+1} - \lambda_{km+m}) \leq \sum_{0 \leq k \leq \frac{n}{m}} (\lambda_{km+1} - \lambda_{km+m+1}),
\]

which is a telescoping sum and we get

\[
\mu_1 - \mu_m \leq \lambda_1 - \lambda_{(m[\frac{n}{m}]+m+1)} \leq \lambda_{\text{max}}.
\]

12
We have $\mu_1 - \mu_m \geq \mu_i - \mu_j$ for all $1 \leq i \leq j \leq m$ which implies the first part of $(3.2)$. For $\mu$ and all $1 \leq j \leq m$ we therefore have

$$\mu = \sum_{i=1}^{m} \mu_i \leq \sum_{i=1}^{m} (\mu_j + \lambda_{max}) = m\mu_j + m\lambda_{max}$$

and also

$$\mu = \sum_{i=1}^{m} \mu_i \geq \sum_{i=1}^{m} (\mu_j - \lambda_{max}) = m\mu_j - m\lambda_{max}$$

which proves the second inequality of $(3.2)$.

4 Greedy has almost optimal expected competitive ratio for small $R$

We will prove the following lemma, which implies the first part of Theorem 1.1.

**Lemma 4.1.** Let $\omega(n)$ be an arbitrary function that tends to infinity with $n$ tending to infinity. There is a constant $n_0$ such that for all $n > n_0$, all $m = m(n) \leq \frac{n}{\omega(n) m \ln(m)}$, and every $\Lambda = (\lambda_1, \ldots, \lambda_n) \in \mathbb{R}_{+}^n$ satisfying $R(\Lambda) \leq \frac{n}{\omega(n)m \ln(m)}$, we have

$$\text{ECR(Greedy; } \Lambda) = \mathbb{E} \left[ \frac{\text{Opt}(\mathbf{P})}{\text{Greedy}(\mathbf{A}, \mathbf{P})} \right] = 1 + \mathcal{O} \left( \omega(n)^{-\frac{1}{4}} \right) \xrightarrow{n \to \infty} 1.$$

**Proof.** We will exploit the properties of algorithm Greedy from the previous section to show that it has an expected competitive ratio of at most $1 + \mathcal{O} \left( \omega(n)^{-1/4} \right)$.

From now on we shall assume without loss of generality that $\lambda_{min} = 1$, because an exponentially distributed random variable $X$ with mean $\lambda$ that is scaled by some factor $\xi$ is an exponentially distributed variable with mean $\xi\lambda$. Since we consider the ratio $\frac{\text{Opt}}{\text{Greedy}}$ scaling does not affect the results. Therefore we may assume $\lambda_{min} = 1$ and $\lambda_{max} \leq \frac{n}{\omega(n)m \ln(m)}$. Further we will assume that $\lambda_1 \geq \cdots \geq \lambda_n$. In the remainder we will write $\omega = \omega(n)$. Furthermore, we assume $n_0$ to be large enough whenever we need $\omega(n)$ to be large.

Let $I_1, \ldots, I_m$ be the partition computed by Greedy, see Section 3. To ease notation we will write for the objective values $\text{Opt} := \text{Opt}(\mathbf{P})$ and $\text{Greedy} := \text{Greedy}(\mathbf{A}, \mathbf{P})$. In order to estimate $\mathbb{E} \left[ \frac{\text{Opt}(\mathbf{P})}{\text{Greedy}(\mathbf{A}, \mathbf{P})} \right]$ we will partition the probability space into different events and compute the conditional expectations. More precisely, we will use the fact that if $A_1, A_2, \ldots$ are not necessarily disjoint events, such that the union of them is the entire probability space, then for a non-negative random variable $X$ we have

$$\mathbb{E} [X] \leq \sum_{i \geq 1} \mathbb{E} [X \mid A_i] \cdot \Pr [A_i]. \quad (4.1)$$

Let $Y_i := \sum_{j \in I_i} X_j$ and $\mu_i := \sum_{j \in I_i} \lambda_j$. Note that in the scheduling terminology $Y_i$ is the value Greedy achieves on machine $i$. Let $\mu := \sum_{i=1}^{m} \mu_i = \sum_{i=1}^{n} \lambda_i$. To partition the probability space we will define appropriate intervals $L_k$ and $U_k$, $k \geq 0$. The intervals will cover $\mathbb{R}_+$ except for a strip around the mean $\frac{\mu}{m}$, see Figure 2. The idea is that we can find
sufficient upper bounds for the ratio $\frac{\text{Opt}}{\text{Greedy}}$ if we condition on an event like $\text{Opt} \in U_i$ and $\text{Greedy} \in U_j$. We define the following intervals for all integers $k \geq 0$:

$$L_k = \left[2^{-(k+1)} \omega^{-1/4} \frac{\mu}{m^2}; 2^{-k} \omega^{-1/4} \frac{\mu}{m^2}\right] \quad \text{and} \quad U_k = \left[2^k \omega^{1/4} \mu; 2^{k+1} \omega^{1/4} \mu\right].$$

Figure 2: Intervals $L_k$ and $U_k$ in $R_+$. With $\omega_l = \omega^{-1/4}$ and $\omega_u = \omega^{1/4}$.

We will define one main event $M$ which says that all $Y_i$ are close to their respective means. Further we will define an event $S \subset M$ in such a way that $\mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid S \right] \Pr [M] = o(1)$:

1. event $M$: $\forall 1 \leq i \leq m \quad (1 - \omega^{-1/4}) \mu_i \leq Y_i \leq (1 + \omega^{-1/4}) \mu_i$.
2. event $S$: $M \land \frac{\text{Opt}}{\text{Greedy}} \leq \omega^{1/2} m^2$.

What remains to take care of is the event $\frac{\text{Opt}}{\text{Greedy}} > \omega^{1/2} m^2$. For this, consider the following definitions.

3. for all integers $k \geq 0$ the event $E_k$: $\text{Opt} \leq \omega^{1/4} \mu$ and $\text{Greedy} \in L_k$,
4. for all integers $k \geq 0$ the event $F_k$: $\text{Opt} \in U_k$ and $\text{Greedy} \geq \omega^{-1/4} \frac{\mu}{m^2}$,
5. and for all integers $k, l \geq 0$ event $G_{kl}$: $\text{Opt} \in U_k$ and $\text{Greedy} \in L_l$.

Let $\Omega$ be the probability space of all outcomes of the $X_1, \ldots, X_n$. We will show that

$$\Omega = M \cup S \cup \bigcup_{k \geq 0} E_k \cup \bigcup_{k \geq 0} F_k \cup \bigcup_{k, l \geq 0} G_{kl}. \quad (4.2)$$

Then, by using (4.1) we obtain

$$\mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \right] \leq \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid M \right] \Pr [M] + \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid S \right] \Pr [S] +
$$

$$+ \sum_{k \geq 0} \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid E_k \right] \Pr [E_k] + \sum_{k \geq 0} \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid F_k \right] \Pr [F_k] + \quad (4.3)$$

$$+ \sum_{k, l \geq 0} \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid G_{kl} \right] \Pr [G_{kl}].$$

Observe that the definitions of the events $S, E_k, F_k$ and $G_{kl}$ imply deterministic upper bounds on the conditional expectations:

$$\mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid S \right] \leq \omega^{1/2} m^2, \quad \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid E_k \right] \leq \omega^{1/2} 2^k + 1 m^2, \quad \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid G_{kl} \right] \leq \omega^{1/2} 2^k + l m^2. \quad (4.4)$$

$$\sum_{k, l \geq 0} \mathbb{E} \left[ \frac{\text{Opt}}{\text{Greedy}} \mid G_{kl} \right] \Pr [G_{kl}] = \omega^{1/2} m^2 + \omega^{1/2} 2^k + 1 m^2 + \omega^{1/2} 2^k + l m^2. \quad (4.4)$$
Furthermore, we will show the (deterministic) upper bound for the expectation of $\frac{\text{Opt}}{\text{Greedy}}$

$$
E \left[ \frac{\text{Opt}}{\text{Greedy}} \right] | M \leq 1 + 8\omega^{-1/4}, \quad (4.5)
$$

and the following upper bounds for the probabilities:

$$
\Pr [S] \leq 2m^{-2}\omega^{-1},
\forall k \geq 0 : \quad \Pr [E_k] \leq \omega^{-1/2 - 4k}m^{-2},
\forall k \geq 0 : \quad \Pr [F_k] \leq \omega^{-1/2 - 4k}m^{-2},
\forall k, l \geq 0 : \quad \Pr [G_{kl}] \leq \omega^{-1/2 - 2(k+l)}m^{-2}. (4.6) - (4.9)
$$

Substituting (4.4)-(4.9) in (4.3) we readily get

$$
E \left[ \frac{\text{Opt}}{\text{Greedy}} \right] = 1 + O\left(\omega^{-1/4}\right),
$$

which proves the theorem.

It remains to prove (4.2) and (4.5) - (4.9). We will first argue that the union of the events is the complete probability space. Note that if all values for Opt and Greedy are covered by the events then all possible outcomes of $X_1, \ldots, X_n$ have been considered. The event $M$ and $S$ together cover the possibility that $\frac{\text{Opt}}{\text{Greedy}} \leq \omega^{1/2}m^2$. So we have to ensure that the remaining events cover all possibilities where the ratio is larger than $\omega^{1/2}m^2$. If Opt $\leq \omega^{1/4}m$ then we have to consider values for Greedy which are smaller than $\omega^{-1/4}m^2$. This is covered by the events $E_k$. If on the other hand Greedy $\geq \omega^{-1/4}m$, we have to consider values for Opt that are greater than $\omega^{1/4}m$ which is covered by the events $F_k$. Finally, it remains to consider the combinations where Opt $\geq \omega^{1/4}m$ and Greedy $\leq \omega^{1/4}m$; these are covered by the events $G_{kl}$. This proves (4.2).

In order to show (4.5) - (4.9) we will make two observations. For all $1 \leq i \leq m$ we deduce from (3.2) that $|\mu_i - \mu| \leq \lambda_{\text{max}} \leq \frac{n}{\omega m \ln m}$. For large $\omega$ we then have with plenty of room to spare

$$
\frac{1}{2} \frac{\mu}{m} \leq \mu_i \leq \sqrt{2} \frac{\mu}{m}. \quad (4.10)
$$

Furthermore since $\lambda_{\text{min}} = 1$ we clearly have $\mu \geq n$. Therefore

$$
\lambda_{\text{max}} \leq \frac{n}{\omega m \ln(m)} \leq \frac{n}{m} \leq \frac{\mu}{m}. \quad (4.11)
$$

Before we proceed, observe that the best Opt can do is to distribute the jobs evenly, i.e., giving to each machine a total weight of at least $\frac{\sum_{i=1}^n p_i}{m}$. We will use this several times as an upper bound on Opt.

Proof of (4.5). If the event $M$ occurs, then by (3.2) and (4.11) we have for $1 \leq j \leq m$

$$
Y_j \geq \left(1 - \omega^{-1/4}\right) \mu_j \geq \left(1 - \omega^{-1/4}\right) \left(\frac{\mu}{m} - \lambda_{\text{max}}\right) \geq \left(1 - \omega^{-1/4}\right) \left(1 - \lambda_{\text{max}} \cdot \frac{m}{n}\right) \mu \geq \left(1 - \omega^{-1/4}\right) \left(1 - \frac{1}{\omega}\right) \frac{\mu}{m}. \quad (4.12)
$$
Thus, as \( \min \{ Y_j, 1 \leq j \leq m \} \) is the value of Greedy, this is a lower bound on the value of Greedy conditioned on \( M \). In addition, \( \sum_{i=1}^{n} X_i = \sum_{i=1}^{m} Y_j \leq \sum_{i=1}^{m} (1 + \omega^{-1/4}) \mu_j = (1 + \omega^{-1/4}) \mu \) and thus \( \text{OPT} \leq (1 + \omega^{-1/4}) \frac{\mu}{m} \). It follows that if \( \omega > 8 \)

\[
\mathbb{E} \left[ \frac{\text{OPT}}{\text{Greedy}} \middle| M \right] \leq \frac{1 + \omega^{-1/4}}{(1 - \omega^{-1/4})(1 - \frac{1}{4})} \leq 1 + 8\omega^{-1/4}.
\]

**Proof of (4.10).** Clearly \( \Pr \left[ S \right] \leq \Pr \left[ \overline{M} \right] \). In the sequel we will show that \( \Pr \left[ \overline{M} \right] \leq 2m^{-2} \omega^{-1/4} \). By Lemma 2.6 we have that

\[
\Pr \left[ Y_i \leq (1 - \omega^{-1/4}) \mu_i \right] \leq e^{-\omega^{-1/2} \frac{\mu_i}{2 \lambda_{\text{max}}}} \leq \exp \left( -\omega^{-1/2} \frac{\sqrt{2} m}{2 \omega m \ln m} \right).
\]

Thus the probability for \( \omega \), that is that there exists an \( i \) for which \( Y_i - \mu_i \geq \omega^{-1/4} \mu_i \), is at most \( 2m \left( \frac{1}{m} \right)^3 \omega^{-1} \), which completes the proof.

**Proof of (4.7).** By the definition of \( E_k \) we have \( \Pr \left[ E_k \right] \leq \Pr \left[ \text{Greedy} \in L_k \right] \). And from \( \text{Greedy} \in L_k \) it follows that there has to be an \( 1 \leq i \leq m \) such that \( Y_i \in L_k \). Therefore

\[
\Pr \left[ \text{Greedy} \in L_k \right] \leq \sum_{1 \leq i \leq m} \Pr \left[ Y_i \in L_k \right] \leq \sum_{1 \leq i \leq m} \Pr \left[ Y_i \leq \omega^{-1/4} 2^{-k} \frac{\mu_i}{m^2} \right].
\]

By Lemma 2.6 we have

\[
\Pr \left[ Y_i \leq \omega^{-1/2} 2^{-k} \frac{\mu_i}{m} \right] \leq \Pr \left[ Y_i \leq \omega^{-1/4} 2^{-k} \frac{1}{m} \frac{\mu_i}{m} \right] \leq \left( \omega^{-1/4} 2^{-k} \frac{1}{m} e^2 \right)^{m \lambda_{\text{max}}} \leq \left( \omega^{-1/4} 2^{-k} \frac{1}{m} e^2 \right)^{\frac{\mu_i}{2 \omega m \ln m}} \leq \omega^{-1/4} 2^{-4k} \left( \frac{1}{m} \right)^3.
\]

Thus, we obtain the following bound

\[
\Pr \left[ E_k \right] \leq \Pr \left[ \text{Greedy} \in L_k \right] \leq \omega^{-1/2} 2^{-4k} m^{-2}.
\]

**Proof of (4.8).** We have \( \Pr \left[ F_k \right] \leq \Pr \left[ \text{OPT} \in U_k \right] \). As we noted above \( \text{OPT} \leq \sum_{i=1}^{n} X_i \). Therefore

\[
\Pr \left[ \text{OPT} \in U_k \right] \leq \Pr \left[ \text{OPT} \geq m \omega^{-1/2} 2^{-k} \frac{\mu}{m} \right] \leq \Pr \left[ \sum_{i=1}^{n} X_i \geq m \omega^{-1/2} 2^{-k} \mu \right].
\]
By applying Lemma 2.5 we obtain with $1 + \varepsilon = m\omega^{1/4}2^k$,

$$
\Pr \left[ \sum_{i=1}^{n} X_i \geq (1 + m\omega^{1/4}2^k - 1)\mu \right] \leq e^{-\left(\frac{m\omega^{1/4}2^k - 1}{4m\omega^{1/4}2^k}\right)^2} \mu_{\max}^{-\frac{k}{m}} \leq \omega^{-1}2^{-4k} \left(\frac{1}{m}\right)^2. \tag{4.13}
$$

Proof of (4.9). As we showed in (4.13) we have $\Pr [\text{Opt} \in U_k] \leq \omega^{-1}2^{-4k} \left(\frac{1}{m}\right)^2$ and by (4.12) we have $\Pr [\text{Greedy} \in L_i] \leq \omega^{-1}2^{-4l} \left(\frac{1}{m}\right)^2$. With this we may obtain the following bound.

$$
\Pr [G_{kl}] = \Pr [\text{Opt} \in U_k \land \text{Greedy} \in L_i] \\
\leq \min \{ \Pr [\text{Opt} \in U_k], \Pr [\text{Greedy} \in L_i] \} \\
\leq \frac{\omega^{-1}}{m^2} \cdot 2^{-4 \max(k,l)} \leq \frac{\omega^{-1}}{m^2} \cdot 2^{-2(k+l)}.
$$

5 Lower Bounds for larger $R(\Lambda)$

In this section we will prove lower bounds on the expected competitive ratio for any algorithm. For each $n, m(n)$ and $R$ we will give an instance $\Lambda$ with $R = R(\Lambda)$ such that any algorithm has expected competitive ratio larger than a lower bound depending on $R(\Lambda)$ and $m$. In the first subsection we will give instances with large $R$. This bound is good as long as $R \geq \frac{n^2}{m}$. In the second part of this section we will give instances $\Lambda$ with $R(\Lambda) > \xi \frac{n}{m \log m}$ for a fixed constant $\xi$ and lower bounds on the expected competitive ratio for any algorithm on these instances. This closes the gap to Lemma 4.1 since we know that the algorithm Greedy achieves on instances $\Lambda$ with $R(\Lambda) \ll \frac{n}{m \log m}$ an expected competitive ratio of $1 + o(1)$. The ideas behind the instances for the two lower bounds are similar but part of the analysis is quite different.

5.1 Lower Bound in the case of large deviations

Lemma 5.1. There are constants $n_0, d > 0$ such that the following is true. For all $n > n_0$, $r = r(n) > 0$ and $m = m(n)$ satisfying $2 \leq m \leq \frac{n}{\ln n}$ there exists $\Lambda = (\lambda_1, \ldots, \lambda_n)$ with $R(\Lambda) = r \frac{n}{m \ln m}$ such that for any algorithm ANY

$$
\text{ECR(AANY; }\Lambda ) = \mathbb{E} \left[ \frac{\text{Opt}(P)}{\text{AANY}(\Lambda, P)} \right] \geq \text{d} \cdot m \ln \left( 1 + \frac{r}{m \ln m} \right).
$$

Proof. In order to prove the lemma we will present an instance $\Lambda$ with the property that $\text{ECR(AANY; }\Lambda ) = \Omega \left( m \log \left( 1 + \frac{r}{m \ln m} \right) \right)$ for any algorithm AANY, which schedules the given jobs onto $m$ machines according to their expected processing times. Our bad instance consists of \[ \left[ \frac{3n}{2} \right] \] “big” jobs with mean $\lambda_{\max} = r \frac{n}{m \ln m}$ and $n - \left[ \frac{3n}{2} \right]$ “small” jobs with mean 1. For the
sake of presentation and since we have \( n \geq n_0 \) we shall omit from now on \([.]\) and \([.]\). Formally we have \( \Lambda = (\lambda_1, \ldots, \lambda_n) \) with \( \lambda_1 = \cdots = \lambda_{2m} = \frac{r_m}{m} \) and \( \lambda_{2m+1} = \cdots = \lambda_n = 1 \).

The intuition behind this bad instance is that every algorithm that has to decide in advance how to schedule the jobs will schedule with constant probably the big job which ends up with the smallest processing time (among the big jobs) on a machine without another big job. On the other hand OPT can just use the \( m \) largest of the big jobs each on one machine.

We denote by \( B := \text{Exp}(r \frac{n}{m}) \) the distribution of the big jobs and by \( B_i(k) \) the distribution of the \( i \)-th smallest of \( m \) independent random variables distributed like \( B \).

Assume that we have a fixed partition \( I_1, \ldots, I_m \) computed by an algorithm Any. The “critical” parts \( I_j \) for which the sum \( \sum_{i \in I_j} X_i \) is likely to be small are the sets containing at most one big and few small jobs. We define the following sets to identify the “critical” parts:

\[
S_1 := \{ j : \text{at most one big job is in } I_j \},
\]

\[
S_2 := \{ j : \text{at most } 10n \frac{m}{m} \text{ small jobs are in } I_j \},
\]

\[
S := S_1 \cap S_2.
\]

Since there are \( m \) sets and \( \frac{3}{2}m \) big jobs, we have \( |S_1| \geq \frac{m}{2} \). There can be at most \( \frac{m}{2} \) machines with more than \( \frac{4m}{m} \) small jobs and we have that \( |S_2| \geq \frac{3m}{m} \). Therefore \( |S| = |S_1 \cap S_2| \geq \frac{m}{m} \).

We will see later that an algorithm which does not put at least one big job in each set in \( S \) performs badly. Therefore let us first assume that the partition produced by the algorithm Any is such that there is a big job in each \( I_j \) for \( j \in S \). For \( 1 \leq j \leq m \), we let \( T_j \) be the sum of the small jobs in the set \( I_j \), more precisely we define

\[
T_j := \sum_{i \geq \frac{3m}{m}, i \in I_j} X_i.
\]

In order to show that Any has large expected competitive ratio we will condition on two events. We will chose the events in such a way that their intersection has constant probability and leads to a small expectation for Any. Event \( A_1 \) will guarantee that \( T_j \) for each \( j \in S \) is small and event \( A_2 \) is the event that the smallest big job, i.e. \( \min_{1 \leq i \leq \frac{3m}{m}} (X_i) \), is in a set \( I_j \) with \( j \in S \):

\[ (A_1) : T_j \leq \frac{20n}{m} \text{ for all } j \in S. \]

\[ (A_2) : \text{the big job with minimum processing time is in some } I_j \text{ with } j \in S. \]

If \( A_1 \) and \( A_2 \) occur simultaneously, then the value of Any is at most the minimum processing time that one of the \( \frac{3m}{m} \) big jobs has, plus the processing times of some small jobs, which sum up to at most \( \frac{20m}{m} \). On the other hand, we will show that OPT will be “large” because the \( m \) largest of the big jobs are likely to have large processing times.

We first estimate the probability of event \( A_1 \). By Lemma 2.5 and as there are for each \( j \in S \) at most \( \frac{10m}{m} \) small jobs in \( I_j \), we have that

\[
\Pr \left[ T_j \geq \frac{20n}{m} \right] \leq \Pr \left[ \sum_{i=1}^{\frac{10m}{m}} \text{Exp}(1) \geq 2 \frac{10n}{m} \right] \leq \exp \left( -\frac{1}{8} \cdot \frac{10n}{m} \right) \leq \frac{1}{n},
\]

\[ 18 \]
because we assumed \( m \leq \frac{n}{8 \log n} \). Therefore we have

\[
\Pr(A_1) \geq 1 - \frac{m}{n}. \tag{5.1}
\]

Every job of the big jobs is equally likely to have the smallest processing time among them, so as there are at least \( \frac{1}{m} \) machines in \( S \), the probability for event \( A_2 \) is at least \( \frac{4/10m}{\frac{1}{2}m} = \frac{4}{15} \).

As the events \( A_1 \) and \( A_2 \) are independent we deduce \( \Pr[A_1 \land A_2] \geq (1 - \frac{m}{n}) \cdot \frac{4}{15} \geq \frac{1}{4} \) since \( m \leq \frac{n}{\ln n} \).

From now on we condition on the event \( A_1 \land A_2 \). Note that by conditioning on \( A_1 \land A_2 \) we have not gained any knowledge about the actual values of the \( X_i \) for \( 1 \leq i \leq \frac{3}{4}m \) except that the minimum value is attained by a big job in a \( I_j \) with \( j \in S \). As the minimum of the big jobs is in a \( I_j \) with only one big job, \( \text{ANY} \) has objective value of at most the minimum of the big jobs plus \( T_j \leq \frac{20m}{m} \), i.e., \( \text{ANY} \leq B_{(\frac{3}{4}m \oplus \frac{3}{2}m \oplus 1)} \). The value of \( \text{OPT} \) is at least \( B_{(\frac{1}{2}m \oplus \frac{3}{2}m \oplus 1)} \) because \( \text{OPT} \) can put the \( m \) largest of the big jobs each on one machine.

Clearly, the processing time of the \( m \)th largest (equivalently, the \((\frac{3}{4}m \oplus 1)\)st smallest) big job depends on the processing time of the smallest big job. Because of the fact that the exponential distribution is memoryless we deduce that if the processing time of the smallest of the big jobs is \( \tau \) then the \((\frac{3}{4}m \oplus 1)\)st smallest is distributed like \( B_{(\frac{3}{2}m \oplus 1)} \). By Lemma 2.4 we have that \( B_{(i,k)} \) is distributed like the sum of independent exponentially distributed random variables \( \sum_{j=1}^{i-1} \exp \left( \frac{r n}{m \ln m (k-j)} \right) \). Thus for \( D \sim B_{(\frac{3}{2}m \oplus 1)} \) we have assuming \( m \geq 2 \)

\[
\mathbb{E}[D] = r \frac{n}{m \ln m} \sum_{i=0}^{\frac{3}{4}m-1} \frac{1}{\frac{3}{4}m - 1 - i} \geq \frac{r n}{3m \ln m}. \tag{5.2}
\]

Thus, conditioned on \( B_{(\frac{1}{2}m \oplus 1)} = \tau \) and \( A_1 \land A_2 \) we have that the value of \( \text{OPT} \) is at least \( D + \tau \), and the value of \( \text{ANY} \) is at most \( \tau + \frac{20m}{m} \). Denote by \( f(\tau) := \frac{3m^2 \ln m}{2r n} e^{-\frac{2m^2 \ln m}{2r n}} \) the density of \( B_{(\frac{1}{2}m \oplus 1)} \). With the above discussion we now obtain the following lower bound on the expectation \( \mathbb{E}[\text{OPT} | \text{ANY}] \):

\[
\mathbb{E}[\text{OPT} | \text{ANY}] \geq \Pr[A_1 \land A_2] \cdot \mathbb{E}[\text{OPT} | \text{ANY} \land A_1 \land A_2] \geq \frac{1}{4} \int_0^\infty \mathbb{E} \left[ D + \tau \right] f(\tau) d\tau \geq \frac{1}{4} \int_0^\infty \mathbb{E} \left[ D \right] \frac{3m^2 \ln m}{2r n} e^{-\frac{2m^2 \ln m}{2r n}} d\tau \geq \frac{m}{8} \int_0^\infty e^{-\frac{2m^2 \ln m}{2r n}} d\tau. \tag{5.3}
\]

Recalling the definition of the exponential integral from Lemma 2.8 we obtain by perform-
ing the substitution $\tau = \frac{2rn}{3m^2 \ln m} - \frac{20n}{m}$,

$$E \left[ \frac{\text{OPT}}{\text{ANY}} \right] \geq \frac{m}{8} \int_{r}^{\infty} \frac{e^{-x+30 \ln m}}{30m \ln r} \cdot \frac{2rn}{3m^2 \ln m} \, dx$$

$$= \frac{m}{8} e^{\frac{30m \ln m}{r}} \cdot E \left( \frac{30m \ln m}{r} \right) \quad \text{(by Lemma 2.8)}$$

$$\geq \frac{m}{16} \ln \left( 1 + \frac{r}{15m \ln m} \right). \quad (5.4)$$

We have setting $d = \frac{1}{16-15} = \frac{1}{240}$ that

$$\frac{m}{16} \ln \left( 1 + \frac{r}{15m \ln m} \right) \geq d \cdot m \ln \left( 1 + \frac{r}{m \ln m} \right).$$

This is true for the case that ANY assigns at least one big job to each $I_j$ for $j \in S$. Now suppose there is an $I_j$ with $j \in S$ without a big job. In this case we just need $A_1$. We deduce that

$$E \left[ \frac{\text{OPT}}{\text{ANY}} \right] \geq \Pr [A_1] \cdot E \left[ \frac{\text{OPT}}{\text{ANY}} \bigg| A_1 \right] \geq \frac{1}{4} \cdot \frac{r}{20n} \geq \frac{r}{240 \ln m} = d \cdot \frac{r}{\ln m}.$$  

Basic transformations yield

$$E \left[ \frac{\text{OPT}}{\text{ANY}} \right] \geq d \cdot m \frac{r}{m \ln m} \geq d \cdot m \ln \left( 1 + \frac{r}{m \ln m} \right).$$

This proves the lemma.

We further note that if we would use the Longest Expected Processing Time First (LEPT) algorithm on the instance $\lambda_1 = \lambda_{\max}, \lambda_2 = \cdots = \lambda_n = 1$ the expected competitive ratio is unbounded, if $\lambda_{\max} > 2n$. Note that LPT would schedule the first job alone on a machine. If we condition on the event that some other $m$ jobs have processing times at least an (arbitrary) constant $C$ then we have as above

$$E \left[ \frac{\text{OPT}}{\text{LPT}} \right] \geq \int_{0}^{\infty} C \frac{1}{r} \frac{e^{-\tau}}{\lambda_1} d\tau = \infty.$$  

In other words, every algorithm that schedules the jobs in such a way that one machine has only one job, has unbounded expected competitive ratio on certain instances.

### 5.2 Lower Bound in the case of small deviations

**Lemma 5.2.** There are constants $n_0$, and $\xi, d > 0$ such that the following is true. For all $n > n_0$, $m = m(n) \leq \frac{n}{8 \ln n}$ and $\xi < r < m^{1/4}$ there exists $\Lambda = (\lambda_1, \ldots, \lambda_n)$ with $R(\Lambda) = r \frac{n}{m \ln m}$ such that for any algorithm ANY

$$\text{ECR}(\text{ALG} ; \Lambda) = E \left[ \frac{\text{OPT}(P)}{\text{ALG}(\Lambda, P)} \right] \geq d \cdot \frac{r}{\ln r}.$$
Proof. To obtain a lower bound for $\xi \leq r \leq m^{1/4}$ we use the instance $A$ consisting of $k \cdot m$ big jobs $\lambda_1 = \cdots = \lambda_{km} = \frac{mn}{m \ln m}$, where $k$ will be specified later, and $\lambda_{km+1} = \cdots = \lambda_n = 1$ small jobs. Let $I_1, \ldots, I_m$ be the partition computed by $\text{ANY}$, which corresponds to a schedule on the $m$ machines.

The intuition behind this bad instance is similar as in Lemma 5.1: every algorithm that has to decide in advance how to schedule the jobs will do this with constant probability in such a way that there is a machine, where the sums of the processing times of the big jobs is not too “large”. To formalize this, consider the following three sets of machines.

$$S_1 := \{ j : \text{at most } \frac{3k}{2} \text{ big jobs are in } I_j \},$$
$$S_2 := \{ j : \text{at most } \frac{10n}{m} \text{ small jobs are in } I_j \},$$
$$S := S_1 \cap S_2.$$

By the pigeonhole principle we have $|S_1| > \frac{m}{\lambda}$ and $|S_2| > \frac{9m}{\lambda}$. Therefore the cardinality of $S$ is at least $|S_1| \cap |S_2| \geq \frac{7m}{\lambda} \geq \frac{m}{\lambda}$. Let us denote by $F_j$ the sum of the processing times of the big jobs on machine $j$, i.e. $F_j := \sum_{i \leq \lambda m} X_i$. Further we denote by $T_j$ the sum of the processing times of the small jobs, i.e. $T_j := \sum_{i \leq \lambda n} X_i$. We define three events in such a way that the expected competitive ratio is large if we condition on them:

$$A_1 : \exists j \in S : F_j \leq \frac{20n}{m},$$
$$A_2 : \forall j \in S : T_j \leq \frac{20n}{m},$$
$$A_3 : \text{OPT} \geq \frac{k \lambda_{\text{max}}}{40}.$$

The event $A_1 \cap A_2 \cap A_3$ implies a deterministic upper bound on $\frac{\text{OPT}}{\text{ALG}}$ and we have

$$\mathbb{E} \left[ \frac{\text{OPT}}{\text{ALG}} \bigg| A_1 \cap A_2 \cap A_3 \right] \geq \frac{k \lambda_{\text{max}}}{40} \frac{m}{40n} = \frac{kr}{1600 \ln m}. \quad (5.5)$$

We now set $k = \left\lceil \frac{4 \ln m}{m \ln r} \right\rceil$. In the remainder we will show that then $\Pr [A_1 \cap A_2 \cap A_3] \geq \frac{1}{4}$. This concludes the proof of the lemma.

We first give a lower bound for $\Pr [A_1]$. We have

$$\Pr [A_1] = \Pr \left[ \exists j \in S : F_j \leq \frac{20n}{m} \right] = 1 - \prod_{j \in S} \left( 1 - \Pr \left[ F_j \leq \frac{20n}{m} \right] \right).$$

As there are at most $\frac{3k}{2}$ big jobs in $I_j$ for $j \in S$ we have $\Pr [F_j \leq t] \geq \Pr \left[ \sum_{i=1}^{3k} \text{Exp}(\lambda_{\text{max}}) \leq t \right]$ for all $j \in S$ and $t \geq 0$. Moreover, observe that trivially $\Pr \left[ \sum_{i=1}^{3k} \text{Exp}(\lambda_{\text{max}}) \leq t \right] \geq \Pr \left[ \text{Exp}(\lambda_{\text{max}}) \leq \frac{20n}{3km} \right]$. We obtain

$$\Pr [A_1] \geq 1 - \left( 1 - \Pr \left[ \text{Exp}(\lambda_{\text{max}}) \leq \frac{20n}{3km} \right] \right)^{|S|}.$$
Since \(1 - e^{-x} \geq \frac{x}{2}\) for \(0 \leq x \leq 1\) we have

\[
\Pr[A_1] \geq 1 - \left(1 - \left(\frac{20m}{3km \lambda_{\max}}\right) \frac{\mu_B}{2}\right)^{|S|} \geq 1 - \exp\left(-\left(\frac{20 \ln m}{3kr}\right) \frac{\mu_B}{6}\right).
\]

Straightforward but tedious calculations show that there is a \(\xi'\) such that whenever \(r \geq \xi'\) we have \(\Pr[A_1] \geq \frac{1}{2}\).

As in (5.1) we have \(\Pr[A_2] \geq 1 - \frac{m}{n}\). To estimate \(\Pr[A_3]\) observe that \(\text{Opt}\) achieves at least a value of \(\frac{1}{2} \cdot B\left(\frac{km}{2}, km\right)\), as it can assign \(\frac{km}{2}\) of the larger big jobs to each machine. We will show using Chebychev’s inequality that \(B\left(\frac{km}{2}, km\right)\) is unlikely to be small. As \(B\left(\frac{km}{2}, km\right)\) is distributed like \(\sum_{i=0}^{\frac{km}{2}-1} \exp\left(\frac{\lambda_{\max}}{km - i}\right)\) we have

\[
\mu_B := \mathbb{E}\left[B\left(\frac{km}{2}, km\right)\right] \geq \frac{\lambda_{\max}}{2}
\]

and

\[
\text{Var}\left[B\left(\frac{km}{2}, km\right)\right] = \sum_{i=0}^{\frac{km}{2}-1} \left(\frac{\lambda_{\max}}{km - i}\right)^2 \leq \frac{km}{2} \left(\frac{\lambda_{\max}}{2}\right)^2 = \frac{2\lambda_{\max}^2}{km}.
\]

Therefore we have by Chebychev’s inequality

\[
\Pr\left[B\left(\frac{km}{2}, km\right) < \frac{1}{10} \frac{\lambda_{\max}}{2}\right] \leq \Pr\left[B\left(\frac{km}{2}, km\right) < \frac{\mu_B}{10}\right]
\]

\[
\leq \Pr\left[|B\left(\frac{km}{2}, km\right) - \mu_B| > \frac{\mu_B}{10}\right]
\]

\[
\leq \frac{2\lambda_{\max}^2}{km} \left(\frac{10}{9\mu_B}\right) \leq \frac{\lambda_{\max}^2}{m} \left(\frac{20}{9\lambda_{\max}}\right) = \left(\frac{20}{9}\right)^2 \frac{1}{m}.
\]

Therefore we have \(\Pr[A_3] = 1 - (20/9)^2 \cdot \frac{1}{m}\) and as we have that \(m\) is larger than \(\xi^4\) we conclude that \(\Pr[A_1 \cap A_2 \cap A_3] \geq \frac{1}{4}\).

6 Upper bounds on the expected competitive ratio

In this section we will show that the algorithm \textsc{Greedy} performs well even if the \(\lambda_i\) deviate much from each other. We will prove the following lemma.

\textbf{Lemma 6.1.} There are constants \(n_0\) and \(d > 1\) such that for all \(n > n_0\) and \(2 \leq m = m(n) \leq \frac{n \ln n}{8}\) the following is true. For all \(\Lambda = (\lambda_1, \ldots, \lambda_n)\) with \(R(\Lambda) = r \frac{n}{m \ln m}\) we have for all \(r \geq 1\)

\[
\text{ECR(\textsc{Greedy}; \Lambda)} = \mathbb{E}\left[\frac{\text{OPT}(P)}{\text{\textsc{Greedy}(\Lambda, P)}}\right] \leq d \cdot r,
\]  \hspace{1cm} (6.1)

and for \(r \geq \ln m\) we have

\[
\text{ECR(\textsc{Greedy}; \Lambda)} = \mathbb{E}\left[\frac{\text{OPT}(P)}{\text{\textsc{Greedy}(\Lambda, P)}}\right] \leq d \cdot m \ln m \cdot \ln \left(1 + \frac{r}{\ln m}\right).
\] \hspace{1cm} (6.2)
Without loss of generality we assume in this section again that $\lambda_{\text{max}} = r_m^\frac{n}{m \ln m} \geq \lambda_1 \geq \cdots \geq \lambda_n = 1$.

Our main tool to compute an upper bound for the expected competitive ratio is the FKG-inequality as stated in Theorem 2.7. Clearly OPT as well as Greedy are increasing functions that map from $\mathbb{R}^n_+$ to $\mathbb{R}_+$ in the sense of Theorem 2.7. Since Greedy is increasing, $\frac{1}{\text{Greedy}}$ is a decreasing function and therefore by Theorem 2.7

$$E \left[\frac{\text{OPT}}{\text{Greedy}}\right] \leq E[\text{OPT}] E \left[\frac{1}{\text{Greedy}}\right].$$

(6.3)

In the remainder of this section we will consider the partition $I_1, \ldots, I_m$ computed by Greedy as described in Section 3. Before we proceed we will define some random variables. For $1 \leq j \leq m$ let the random variable $T_j$ be the sum of the jobs on machine $j$ except the one with largest expectation, which is $X_j$. More precisely we define

$$T_j := \sum_{i \in I_j, i > m} X_i, \quad \nu_j := E[T_j], \quad \nu := \sum_{1 \leq j \leq m} \nu_j.$$

(6.4)

We observe that the $T_j$ correspond to the partition we would obtain if Greedy is given $(\lambda_{m+1}, \ldots, \lambda_n)$ as input, where now the maximum expectation is $\lambda_{m+1}$. Therefore, by Lemma 3.1 we have for $1 \leq i, j \leq m$

$$|\nu_j - \nu_i| \leq \lambda_{m+1} \quad \text{and} \quad \frac{\nu}{m} - \lambda_{m+1} \leq \nu_j \leq \frac{\nu}{m} + \lambda_{m+1}$$

(6.5)

### 6.1 Proof of Inequality (6.1)

In order to prove (6.1) we distinguish two cases depending on the value of $\nu$. First we consider the case where $\nu \geq 100rn$ and show that in this case the expected competitive ratio is bounded by a constant:

$$E \left[\frac{\text{OPT}}{\text{Greedy}}\right] \leq 66.$$

On the other hand if $\nu \leq 100rn$ we will prove that

$$E \left[\frac{\text{OPT}}{\text{Greedy}}\right] \leq 3333 \cdot r.$$

Putting the above facts together proves the first inequality of Lemma 6.1.

For the case where $\nu \geq 100rn$ we have for all $1 \leq i \leq m$ by (6.3) that

$$\nu_j \geq \frac{\nu}{m} - \lambda_{\text{max}} \geq \frac{100rn}{m} - \frac{rn}{m \ln m} \geq \frac{98rn}{m}.$$

(6.6)

We obtain for $E \left[\frac{1}{\text{Greedy}}\right]$ that

$$E \left[\frac{1}{\text{Greedy}}\right] \leq \Pr \left[\text{Greedy} \geq \frac{\nu}{32m}\right] E \left[\frac{1}{\text{Greedy}} \mid \text{Greedy} \geq \frac{\nu}{32m}\right] + \sum_{i \geq 5} \Pr \left[\frac{2^{-i-1} \nu}{m} \leq \text{Greedy} \leq 2^{-i} \frac{\nu}{m}\right] E \left[\frac{1}{\text{Greedy}} \mid 2^{-i-1} \frac{\nu}{m} \leq \text{Greedy} \leq 2^{-i} \frac{\nu}{m}\right]$$

$$\leq \frac{32m}{\nu} + \sum_{i \geq 5} \Pr \left[\text{Greedy} \leq 2^{-i} \frac{\nu}{m}\right] 2^{i+1} \frac{m}{\nu}.$$

(6.7)
To give a bound on \( \Pr \left[ \text{GREEDY} \leq 2^{-i} \frac{\nu}{m} \right] \) we first estimate the probability that the value on a fixed machine is smaller than \( 2^{-i} \frac{\nu}{m} \). Since \( \nu \geq 100rn \) we have that \( \nu_j \geq \frac{\nu}{m} - \lambda_{\text{max}} \geq \frac{\nu_j}{m} \).

For \( 1 \leq j \leq m \) we have by Lemma 2.6 and that

\[
\Pr \left[ T_j \leq 2^{-i} \frac{\nu}{m} \right] \leq \Pr \left[ T_j \leq 2^{-i} 2^\nu \right] \leq \left( e^2 2^{-i+1} \right) ^{\frac{\nu}{3\lambda_{\text{max}}} \leq 2^{-(i-4)40 \ln m}.
\]

If GREEDY is smaller than \( 2^{-i} \frac{\nu}{m} \) then at least one \( T_j \) is smaller than \( 2^{-i} \frac{\nu}{m} \). Thus, for \( i \geq 5 \) we have

\[
\Pr \left[ \text{GREEDY} \leq 2^{-i} \frac{\nu}{m} \right] \leq \sum_{1 \leq j \leq m} \Pr \left[ T_j \leq 2^{-i} \frac{\nu}{m} \right] \leq m 2^{-(i-4)40 \ln m} \leq 2^{-4i}.
\]

Thus, we get from (6.7) that

\[
\mathbb{E} \left[ \frac{1}{\text{GREEDY}} \right] \leq \frac{32m}{\nu} + \sum_{i \geq 4} 2^{-4i} 2^{i+1} \frac{m}{\nu} \leq \frac{32m}{\nu} + \frac{m}{\nu} = \frac{33m}{\nu}.
\]

On the other hand, using the upper bound on OPT we obtain

\[
\mathbb{E} [\text{OPT}] \leq \mathbb{E} \left[ \sum_{i=1}^{m} X_i \right] = \frac{\nu}{m} \frac{\sum_{i=1}^{m} \lambda_i}{m} \leq \frac{\nu}{m} + \lambda_{\text{max}} \leq \frac{2\nu}{m}.
\]

Therefore we conclude that

\[
\mathbb{E} \left[ \frac{\text{OPT}}{\text{GREEDY}} \right] \leq \mathbb{E} \left[ \text{OPT} \right] \mathbb{E} \left[ \frac{1}{\text{GREEDY}} \right] \leq \frac{2\nu}{m} \cdot \frac{33m}{\nu} = 66.
\]

This completes the proof for the case \( \nu \geq 100rn \). If on the other hand \( \nu < 100rn \) then we use Lemma 2.6 as in (6.16) and have that

\[
\Pr \left[ T_j \leq 2^{-i} \frac{n}{2m} \right] \leq 2^{-2(i-3) \ln m}.
\]

As the event \( \text{"GREEDY} \leq 2^{-i} \frac{n}{2m} \text{"} \) implies that at least one \( T_j \) is smaller than \( 2^{-i} \frac{n}{2m} \) we get

\[
\Pr \left[ \text{GREEDY} \leq 2^{-i} \frac{n}{2m} \right] \leq \sum_{1 \leq j \leq m} \Pr \left[ T_j \leq 2^{-i} \frac{n}{2m} \right] \leq m \cdot 2^{-2(i-3) \ln m}.
\]

We obtain

\[
\mathbb{E} \left[ \frac{1}{\text{GREEDY}} \right] \leq \Pr \left[ \text{GREEDY} \geq \frac{n}{32m} \right] \mathbb{E} \left[ \frac{1}{\text{GREEDY}} \right] \text{GREEDY} \geq \frac{n}{32m} + \\
+ \sum_{i \geq 4} \Pr \left[ 2^{-i-1} \frac{n}{2m} \leq \text{GREEDY} \leq 2^{-i} \frac{n}{2m} \right] \mathbb{E} \left[ \frac{1}{\text{GREEDY}} \right] 2^{-i-1} \frac{n}{2m} \leq \text{GREEDY} \leq 2^{-i} \frac{n}{2m}
\]

\[
\leq \frac{32m}{n} + \sum_{i \geq 4} \Pr \left[ \text{GREEDY} \leq 2^{-i} \frac{n}{2m} \right] 2^{i+1} \frac{2m}{n}
\]

\[
\leq \frac{32m}{n} + \sum_{i \geq 0} m 2^{-(i+1) \ln m} 2^{i+5} \frac{2m}{n} \leq \frac{33m}{n}.
\]
To bound \( E[\text{Opt}] \) note that

\[
E[\text{Opt}] \leq \frac{1}{m} \left( \nu + m \cdot \lambda_{\text{max}} \right) \leq \frac{1}{m} \left( 100rn + m \cdot r \frac{n}{m \ln m} \right) \leq \frac{101rn}{m}.
\]

In combination with the above bound on \( E[\text{Greedy}] \) and using (6.3) we arrive at the upper bound

\[
E \left[ \frac{\text{Opt}}{\text{Greedy}} \right] \leq \frac{101rn}{m} \cdot \frac{33m}{n} = 3333 \cdot r,
\]

which proves (6.1).

6.2 Proof of (6.2)

The proof of (6.2) proceeds in the following way. First we will establish a new upper bound for \( E[\text{Opt}] \). Note that \( \text{Greedy} \) achieves on machine \( j \) a value of \( X_j + T_j \), and hence the value of \( \text{Greedy} \) is

\[
\text{Greedy}(\lambda, P) = \min_{1 \leq j \leq m} (X_j + T_j).
\]

We will prove for all \( 1 \leq j \leq m \) that

\[
E[\text{Opt}] \leq \frac{256m}{m - j + 1} \ln \left( 1 + \frac{32m\lambda_{\text{max}}}{n} \right). \tag{6.8}
\]

Furthermore we have by the FKG inequality (6.3) that

\[
E \left[ \frac{\text{Opt}}{\text{Greedy}} \right] \leq E[\text{Opt}] \leq \sum_{1 \leq j \leq m} E[\text{Opt}] \left[ \frac{1}{X_j + T_j} \right]. \tag{6.9}
\]

Combining these two inequalities we obtain

\[
E \left[ \frac{\text{Opt}}{\text{Greedy}} \right] \leq \sum_{1 \leq j \leq m} \frac{256m}{m - j + 1} \ln \left( 1 + \frac{32m\lambda_{\text{max}}}{n} \right) \leq 256m \ln m \cdot \ln \left( 1 + \frac{m\lambda_{\text{max}}}{n} \right),
\]

which proves inequality (6.2).

Before proving (6.8) we will establish a new upper bound on \( \text{Opt} \). For each \( 1 \leq j \leq m \) we will prove that

\[
E[\text{Opt}] \leq \lambda_j + \frac{1}{m - j + 1} \sum_{1 \leq i \leq m} \nu_i.
\]

We shall first show that the value of \( \text{Opt} \) is always less then the sum of all but any \( k < m \) jobs divided by \( m - k \). To be more precise we fix some arbitrary set \( F \subset \{1 \ldots n\} \) with \( |F| = k \) and show that \( \text{Opt} \leq \frac{1}{m - k} \sum_{1 \leq i \leq n, i \notin F} X_i \).

Consider the partition \( J_1, \ldots, J_m \) computed by \( \text{Opt} \), after seeing the realizations of the \( X_i \). By the pigeonhole principle there are at least \( m - k \) indices \( j \) with \( J_j \cap F = \emptyset \) and thus \( \text{Opt} \leq \frac{1}{m - k} \sum_{1 \leq i \leq n, i \notin F} X_i \). We could think of this bound as satisfying \( k \) machines with the jobs in \( F \) and then distributing the remaining sum evenly to the other machines.
Most importantly we have the following upper bound for $1 \leq j \leq m$ by chosing $F = \{1, \ldots, j - 1\}$:

$$
\mathbb{E}[\text{Opt}] \leq \mathbb{E}\left[\frac{1}{m - j + 1} \sum_{j \leq i \leq n} X_i\right] = \mathbb{E}\left[\frac{1}{m - j + 1} \left( \sum_{j \leq i \leq m} X_i + \sum_{1 \leq i \leq m} T_i \right)\right]
= \frac{1}{m - j + 1} \left( \sum_{j \leq i \leq m} \lambda_i + \sum_{1 \leq i \leq m} \nu_i \right). \quad (6.10)
$$

Since $\lambda_j \geq \cdots \geq \lambda_m$ we have $
\frac{1}{m - j + 1} \sum_{j \leq i \leq m} \lambda_i \leq \lambda_j$
and therefore

$$
\mathbb{E}[\text{Opt}] \leq \lambda_j + \frac{1}{m - j + 1} \cdot \nu. \quad (6.11)
$$

Observe that this bound reduces to the bound used in Section 4 for $j = 1$.

In order to prove (6.8) we distinguish two cases depending on the value of $\nu$. Let $1 \leq j \leq m$. The idea is that if $\nu$ is large compared to $\lambda_j$, then this implies that $\nu_j = \mathbb{E}[T_j]$ is large and we will bound $\mathbb{E}\left[\frac{1}{X_j + T_j}\right]$ by $\mathbb{E}\left[\frac{1}{T_j}\right]$. On the other hand, if $\nu$ is small compared to $\lambda_j$, we will use $X_j$ to give an upper bound on $\mathbb{E}\left[\frac{1}{X_j + T_j}\right]$. More precisely we will first prove that if $\nu \geq 5m\lambda_j$, then

$$
\mathbb{E}[\text{Opt}] \mathbb{E}\left[\frac{1}{X_j + T_j}\right] \leq \frac{256m}{m - j + 1} \cdot \nu. \quad (6.12)
$$

On the other hand, if $\nu \leq 5m\lambda_j$ then we will show that

$$
\mathbb{E}[\text{Opt}] \mathbb{E}\left[\frac{1}{X_j + T_j}\right] \leq \frac{12m}{m - j + 1} \cdot \ln \left(1 + \frac{32m\lambda_{\text{max}}}{n}\right). \quad (6.13)
$$

Note that we have that $\lambda_{\text{max}} \geq n/m$ and therefore $\ln(1 + 32m\lambda_{\text{max}}/n) \geq 1$. Thus, combining the two inequalities (6.12) and (6.13) we obtain (6.8).

**Proof of Inequality (6.12).** Furthermore, using the assumption $\nu \geq 5m\lambda_j$ we readily obtain

$$5m\lambda_j \leq \nu = \sum_{1 \leq i \leq m} \nu_i \leq m(\nu_j + \lambda_j)
$$

and, therefore, $\nu_j \geq 4\lambda_j$ and $\nu \leq 2m\nu_j$. Next we will give an upper bound on $\mathbb{E}\left[\frac{1}{X_j + T_j}\right]$.

$$
\mathbb{E}\left[\frac{1}{X_j + T_j}\right] \leq \mathbb{E}\left[\frac{1}{T_j}\right] \leq \Pr\left[T_j \geq \frac{\nu_j}{32}\right] \cdot \frac{32}{\nu_j} + \sum_{i \geq 5} \Pr\left[2^{-i-1}\nu_j \leq T_j \leq 2^{-i}\nu_j\right] \cdot \frac{1}{2^{-i-1}\nu_j}. \quad (6.14)
$$

For these probabilities we have by Lemma 2.6 for $i \geq 5$ that

$$
\Pr\left[2^{-i-1}\nu_j \leq T_j \leq 2^{-i}\nu_j\right] \leq \Pr\left[T_j \leq 2^{-i}\nu_j\right] \leq (e^22^{-i})^{\frac{\nu_j}{2\lambda_j}}
\leq 2^{-2(i-3)}.
$$
Using this we obtain from (6.14) the following bound:

\[
E \left[ \frac{1}{X_j + T_j} \right] \leq \frac{1}{\nu_j} \left( 32 + \sum_{i \geq 5} 2^{i+1-2(i-3)} \right) \leq \frac{1}{\nu_j} \left( 32 + \sum_{i \geq 5} 2^{-i+7} \right) \leq \frac{64}{\nu_j}.
\]

From the upper bound \(6.11\) on \(E[\text{OPT}]\) we have, recalling that \(\nu_j \geq 4\lambda_j\) and \(\nu \leq 2m\nu_j\),

\[
E[\text{OPT}] \leq \lambda_j + \frac{1}{m-j+1} \nu + \frac{1}{m-j+1} 2m\nu_j \leq \frac{3m}{m-j+1} \nu.
\]

Combining the last two inequalities yields immediately (6.12).

\[\blacksquare\]

**Proof of Inequality (6.13).** In the case where \(\nu \leq 5m\lambda_j\) \(6.15\)

the expectation of \(T_j\) is too small to give a good upper bound on \(E \left[ \frac{1}{X_j + T_j} \right] \) and we have to use \(X_j\). We have

\[
E \left[ \frac{1}{X_j + T_j} \right] \leq E \left[ \frac{1}{X_j + T_j} \mid T_j \geq \frac{n}{32m} \right] + \sum_{i \geq 5} E \left[ \frac{1}{X_j + T_j} \mid 2^{-i-1} \frac{n}{m} \leq T_j \leq 2^{-i} \frac{n}{m} \right] \Pr \left[ T_j \leq 2^{-i+1} \frac{n}{2m} \right]
\]

By Lemma 3.1 each \(T_j\) is the sum of at least \(\frac{n}{m}\) exponentially distributed random variables with mean at least 1. Therefore we have by Lemma 2.6 that for \(i \geq 5\)

\[
\Pr \left[ T_j \leq 2^{-i} \frac{n}{2m} \right] \leq \Pr \left[ \sum_{i=1}^{\frac{n}{m}} \text{Exp}(1) \leq 2^{-i} \frac{n}{2m} \right] \leq (e^{2-2^{-i}})^{\frac{n}{2m}} \leq 2^{-2(i-3)\ln n} \leq 2^{-10(i-3)},
\]

as we assume \(m \leq \frac{n}{8\ln n}\) and \(n \geq n_0\) for sufficiently large \(n_0\). Thus, we obtain

\[
E \left[ \frac{1}{X_j + T_j} \right] \leq E \left[ \frac{1}{X_j + \frac{n}{32m}} \right] + \sum_{i \geq 5} E \left[ \frac{1}{X_j + 2^{-i-1} \frac{n}{m}} \right] 2^{-10(i-4)}
\]

\[
= E \left[ \frac{1}{X_j + \frac{n}{32m}} \right] + \sum_{i \geq 5} E \left[ \frac{1}{2^{-i+1}X_j + 2^{-5} \frac{n}{m}} \right] 2^{-9(i-4)}
\]

and since for \(i \geq 5\) clearly \(E \left[ \frac{1}{X_j + \frac{n}{32m}} \right] \geq E \left[ \frac{1}{2^{-i+1}X_j + \frac{n}{32m}} \right]\) we arrive at

\[
E \left[ \frac{1}{X_j + T_j} \right] \leq 2 \cdot E \left[ \frac{1}{X_j + \frac{n}{32m}} \right].
\]

The last term is now easily bounded similarly as in Section 5.1

\[
2E \left[ \frac{1}{X_j + \frac{n}{32m}} \right] = 2 \int_0^\infty \frac{1}{\tau + \frac{n}{32m}} \frac{1}{\lambda_j} e^{-\frac{\tau}{\lambda_j}} d\tau = e^{\frac{32m}{32m}} E \left( \frac{n}{32m\lambda_j} \right) \frac{2}{\lambda_j}
\]

27
which is by Lemma 2.8 smaller than
\[ \leq \frac{2}{\lambda_j} \ln \left( 1 + \frac{32m\lambda_j}{n} \right). \]

By (6.11) and since we have (6.15) we readily get
\[ \mathbb{E}[\text{Opt}] \leq \lambda_j + \frac{5m}{m - j + 1} \lambda_j \leq \frac{6m}{m - j + 1} \lambda_j. \]

In combination with the above inequality we have Inequality (6.13):
\[
\mathbb{E}[\text{Opt}] \mathbb{E} \left[ \frac{1}{X_j + T_j} \right] \leq \frac{6m}{m - j + 1} \lambda_j \cdot \frac{2}{\lambda_j} \ln \left( 1 + \frac{32m\lambda_{\text{max}}}{n} \right)
= \frac{12m}{m - j + 1} \ln \left( 1 + \frac{32m\lambda_{\text{max}}}{n} \right).
\]

7 Conclusions

We showed that as long as the instances \( \Lambda \) to the Santa Claus problem job satisfy
\[ R(\Lambda) \ll \frac{n}{m \ln m}, \]
where \( R(\Lambda) \) is the ratio between the largest and the smallest expected running time of a job in \( \Lambda \), the algorithm Greedy has almost optimal expected competitive ratio. For larger \( R(\Lambda) \) we gave instances for which any algorithm will perform badly. Furthermore we showed that for all instances \( \Lambda \) with \( R(\Lambda) \geq \frac{n}{m \ln m} \) algorithm Greedy matches these lower bounds up to a logarithmic factor. The main remaining challenge is to close this gap between the upper and lower bound.

In this work we considered the case where the jobs have identical processing times on all machines. The next natural step would be to look at the uniform machine case, where a scaling factor \( s_j \) is associated with the \( j \)-th machine and the processing time of job \( i \) on machine \( j \) is \( P_{ij} = P_i s_j \). Finally, it would be interesting to investigate to which distributions other than the exponential distribution our result generalizes to.

References


