Non-clairvoyant Scheduling with Partial Predictions

Ziyad Benomar [1 2] **Vianney Perchet** [1 3]

Abstract

The non-clairvoyant scheduling problem has gained new interest within learning-augmented algorithms, where the decision-maker is equipped with predictions without any quality guarantees. In practical settings, access to predictions may be reduced to specific instances, due to cost or data limitations. Our investigation focuses on scenarios where predictions for only B job sizes out of n are available to the algorithm. We first establish near-optimal lower bounds and algorithms in the case of perfect predictions. Subsequently, we present a learning-augmented algorithm satisfying the robustness, consistency, and smoothness criteria, and revealing a novel tradeoff between consistency and smoothness inherent in the scenario with a restricted number of predictions.

1. Introduction

Optimal job scheduling is a longstanding and actively studied class of optimization problems (Panwalkar & Iskander, 1977; Lenstra & Rinnooy Kan, 1978; Graham et al., 1979; Martel, 1982; Cheng & Sin, 1990; Lawler et al., 1993; Pinedo, 2012), with applications in various domains spanning from supply chain management (Hall & Potts, 2003; Ivanov et al., 2016) to operating systems (Jensen et al., 1985; Ramamritham & Stankovic, 1994; Steiger et al., 2004). A particular setting is preemptive single-machine scheduling (Pinedo, 2012; Baker & Trietsch, 2013), where n jobs $i \in [n]$ must be executed on the same machine, with the possibility of interrupting a job and resuming it afterward, and the objective is to minimize the sum of their completion times. An algorithm is called *clairvoyant* if it has initial access to the job sizes, otherwise, it is called *non-clairvoyant* (Motwani et al., 1994). The design of non-clairvoyant scheduling algorithms is a classical problem in competitive analysis and online algorithms (Borodin & El-Yaniv, 2005). In this paradigm, decisions must be made in an environment where the parameters governing the outcome are unknown or might evolve over time.

Due to the inherent difficulty of the problems in competitive analysis, the performance of any online algorithm stays bounded away from that of the optimal offline algorithm. However, the ascent of machine learning motivated the incorporation of predictions in algorithm design, which started in works such as (Munoz & Vassilvitskii, 2017; Kraska et al., 2018), then was formalized in (Lykouris & Vassilvtiskii, 2018) and (Purohit et al., 2018). Since then, *learning-augmented algorithms* became a popular research topic and had multiple applications (Mitzenmacher & Vassilvitskii, 2022). The outcome of these algorithms depends both on the parameters of the problem and the quality of the predictions. They are required to have a performance that is near-optimal when the predictions are accurate (consistency), near the worst-case performance without advice if the predictions are arbitrarily erroneous (robustness), and that degrades smoothly as the prediction error increases (smoothness).

In practice, predictions often incur costs and, at times, are infeasible due to the lack of data. It is, therefore, crucial to understand the limitations and the feasible improvements with a restrained number of predictions in scenarios with multiple unknown variables. This question was first investigated for the caching problem (Im et al., 2022), and very recently for metrical task systems (Sadek & Elias, 2024), in settings where the algorithm is allowed to query a limited number of predictions. It was also explored for the scheduling problem (Benomar & Perchet, 2023), assuming that the decision-maker can query the true sizes of B jobs out of n. The authors present a $\left(2 - \frac{B(B-1)}{n(n-1)}\right)$-competitive algorithm, and they give a lower bound on the competitive ratio of any algorithm only when $B = o(n)$. The case of imperfect predictions, however, is not examined.

In non-clairvoyant scheduling, besides the querying model studied in the works mentioned above, predictions of the sizes of certain jobs $i \in I$ may be available, where $I \subset [n]$, perhaps derived from previous executions of similar tasks. Assuming that I is a subset of $[n]$ of size B, taken uniformly at random, we examine the limitations and possible improvements of non-clairvoyant algorithms.

[1]ENSAE, FAIRPLAY joint team, CREST, Palaiseau, France [2]Ecole polytechnique, Palaiseau, France [3]Criteo AI Lab, FAIRPLAY joint team, Paris, France. Correspondence to: Ziyad Benomar <ziyad.benomar@ensae.fr>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

1.1. Contributions

We initiate our analysis by addressing the scenario of perfect predictions. We establish a lower bound on the (n, B)-competitive ratio of any algorithm (for fixed $n \geq 2$ and $B \leq n$), which extends the lower bound of $2 - \frac{4}{n+3}$ in the non-clairvoyant case (Motwani et al., 1994). Considering that $B = wn + o(n)$ for some $w \in [0, 1]$, we derive from the prior bound that the competitive ratio of any algorithm is at least $2 - w - (\frac{4}{e} - 1)w(1 - w)$, and we show an improved bound of $2 - w - (3 - 2\sqrt{2})w(1 - w)$. Demonstrating these bounds is considerably more challenging than the case $B = 0$, due to the eventual dependency between the actions of the algorithm and the known job sizes.

In the case of perfect predictions, we show that knowing only the relative order of the B job sizes, without knowledge of their values, enables a $(2 - \frac{B}{n})$-competitive algorithm, which improves substantially upon the result of (Benomar & Perchet, 2023). We propose a second algorithm leveraging the true sizes of the B jobs, yielding an (n, B)-competitive ratio of $(2 - \frac{B}{n} - \frac{2(1-B/n)}{n+1})$, which is strictly better than the former, although both are asymptotically equivalent.

Subsequently, we adapt the latter algorithm to handle imperfect predictions. While the difficulty in most works on learning-augmented algorithms lies in ensuring robustness and consistency, smoothness in the case of scheduling with limited predictions is also not immediate. Alongside the typical consistency-robustness tradeoff, our algorithm also exhibits a consistency-smoothness tradeoff. More precisely, governed by two hyperparameters $\lambda, \rho \in [0, 1]$, the (n, B)-competitive ratio of the algorithm is at most $\min(\frac{2}{1-\lambda}, \frac{C}{\lambda} + \frac{S}{\lambda}\frac{n\mathbb{E}[\eta]}{\mathsf{OPT}})$. Here, $\mathbb{E}[\eta]$ denotes the total expected prediction error, OPT is the objective function achieved by the optimal offline algorithm, $\frac{2}{1-\lambda}$ is the algorithm's robustness, $\frac{C}{\lambda} = \frac{1}{\lambda}(2 - \frac{B}{n} + \rho\frac{B}{n}(1 - \frac{B-1}{n-1}))$ its consistency, and $\frac{S}{\lambda} = \frac{1}{\lambda}(\frac{4}{\rho}(1 - \frac{B}{n}) + \frac{B}{n})$ its smoothness factor, characterizing the sensitivity of the bound to $\mathbb{E}[\eta]$. Notably, alterations in the parameter ρ yield opposing variations on the consistency and the smoothness factor. Nonetheless, this tradeoff vanishes for B close to 0 or n, and does not appear, for instance, in (Purohit et al., 2018), (Bampis et al., 2022) or (Lindermayr & Megow, 2022).

We illustrate our results for the case of perfect predictions in Figure 1, comparing them with the competitive ratio proved in (Benomar & Perchet, 2023).

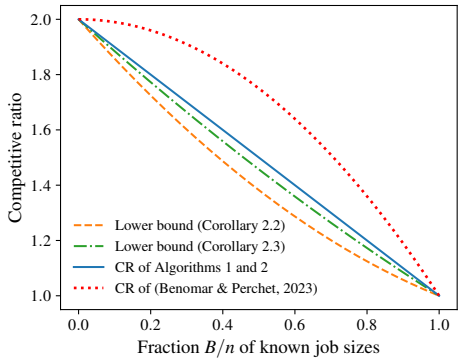

Figure 1. Lower bounds and competitive ratios for B known job sizes.

1.2. Related work

Since their introduction in (Purohit et al., 2018; Lykouris & Vassilvtiskii, 2018), learning-augmented algorithms witnessed an exponentially growing interest, as they offered a fresh perspective for revisiting online algorithms, and provided new applications for machine learning in algorithm design (Mitzenmacher & Vassilvitskii, 2022) and in the implementation of data structures (Kraska et al., 2018; Lin et al., 2022). Many fundamental problems in competitive analysis were studied in this setting, such as ski rental (Gollapudi & Panigrahi, 2019; Anand et al., 2020; Bamas et al., 2020b; Diakonikolas et al., 2021; Antoniadis et al., 2021a; Maghakian et al., 2023; Shin et al., 2023), secretary (Antoniadis et al., 2020; Dütting et al., 2021), matching (Dinitz et al., 2021; Chen et al., 2022; Sakaue & Oki, 2022; Jin & Ma, 2022), caching and metrical task systems (Lykouris & Vassilvtiskii, 2018; Chlkedowski et al., 2021; Antoniadis et al., 2023b;a; Christianson et al., 2023). In particular, scheduling is one of the problems that were studied most thoroughly. Different works cover various objective functions (Purohit et al., 2018; Lattanzi et al., 2020; Azar et al., 2021), prediction types (Antoniadis et al., 2021b; Merlis et al., 2023; Lassota et al., 2023), error metrics (Im et al., 2021; Lindermayr & Megow, 2022), and other aspects and applications (Wei & Zhang, 2020; Bamas et al., 2020a; Dinitz et al., 2022)

The setting of learning-augmented algorithms with limited predictions was initially explored by Im et al. (2022) for caching. The authors presented an algorithm using parsimonious predictions, with a competitive ratio increasing with the number of allowed queries. In another very recent paper (Sadek & Elias, 2024), a similar setting is studied for the more general problem of metrical task systems, where the algorithm is allowed to query a reduced number of *action predictions* (Antoniadis et al., 2023b), each giving the state of an optimal algorithm at the respective query step. An additional related study by Drygala et al. (2023) focuses on the ski-rental and Bahncard problems in a penalized adaptation of the setting with limited advice, where the cost of the predictions is added to the algorithm's objective function. Other works have explored related settings with different types of limited advice. For instance, the setting with a restricted number of perfect hints was examined in the con-

text of online linear optimization by (Bhaskara et al., 2021) and in the multi-color secretary problem by (Benomar et al., 2023). Another setting, where the algorithm can query two types of hints—one that is free but possibly inaccurate, and another that is expensive but accurate—has been studied in several problems, such as correlation clustering (Silwal et al., 2023), computing minimum spanning trees in a metric space (Bateni et al., 2023), sorting (Bai & Coester, 2024), and matroid optimization (Eberle et al., 2024).

In the context of scheduling, Benomar & Perchet (2023) introduced the B-clairvoyant scheduling problem, where an algorithm can query the exact sizes of B jobs at any moment during its execution. They show that the optimal strategy involves querying the sizes of B jobs selected uniformly at random at the beginning of the process. They establish that, if $B = o(n)$, then the competitive ratio of any algorithm is at least 2, then they provide a $\left(2 - \frac{B(B-1)}{n(n-1)}\right)$-competitive algorithm. The same paper also addresses the secretary problem with restricted access to binary predictions of known accuracy, and the ski-rental problem with access to an oracle whose accuracy improves progressively over time.

The limit scenario $B = 0$ corresponds to the non-clairvoyant scheduling problem, studied in-depth in (Motwani et al., 1994). In particular, the paper demonstrates that the competitive ratio of any non-clairvoyant algorithm is at least 2, and that it is achieved by the *round-robin* algorithm, executing all unfinished jobs concurrently at equal rates. On the other hand, $B = n$ corresponds to the setting presented in (Purohit et al., 2018), where the authors introduce, for all $\lambda \in (0, 1)$, a *preferential round-robin* algorithm with robustness $\frac{2}{1-\lambda}$ and consistency $\frac{1}{\lambda}$.

1.3. Problem and notations

The decision-maker is given n jobs $i \in [n]$ with unknown sizes $x_1, \dots, x_n$ to schedule on a single machine, and predictions $(y_i)_{i \in I}$ of $(x_i)_{i \in I}$, with I a uniformly random subset of $[n]$ of size B. The objective is to leverage the available predictions to minimize the sum of the completion times. We assume that preemption is allowed, i.e. it is possible to interrupt the execution of a job and resume it later, which is equivalent, by neglecting the preemption cost, to assuming that the jobs can be run in parallel at rates that sum to at most 1.

To simplify the presentation, we consider that there are predictions $y_1, \dots, y_n$ of $x_1, \dots, x_n$, but the decision-maker has only access to $y_{\sigma(1)}, \dots, y_{\sigma(B)}$, where σ is a uniformly random permutation of $[n]$. We denote by $\eta_i = |x_i - y_i|$ the error of the prediction y_i, and by $\eta^\sigma = \sum_{i=1}^{B} \eta_{\sigma(i)}$ the total error of the predictions accessed by the algorithm.

Consider an algorithm A and an instance $x = (x_1, \dots, x_n)$ of job sizes, we denote by $\mathsf{A}(x)$ the sum of the completion times of all the jobs when executed by A. Furthermore, for all $i \neq j \in [n]$ and $t > 0$, we denote by

- $S_i^{\mathsf{A}}(t)$ the processing time spent on job i until time t,
- $t_i^{\mathsf{A}} = \inf\{t \geq 0 : S_i^{\mathsf{A}}(t) = x_i\}$ its completion time,
- $D_{ij}^{\mathsf{A}} = S_i^{\mathsf{A}}(t_j^{\mathsf{A}})$ the total time spent on job i before job j terminates,
- and $P_{ij}^{\mathsf{A}} = D_{ij}^{\mathsf{A}} + D_{ji}^{\mathsf{A}}$ the mutual delay caused by i, j to each other.

When there is no ambiguity, we omit writing the dependency to A. With these notations, it holds that $t_i^{\mathsf{A}} = x_i + \sum_{j \neq i} D_{ji}^{\mathsf{A}}$ for all $i \in [n]$. Consequently, the objective function of A can be expressed as

$$\mathsf{A}(x) = \sum_{i=1}^{n} x_i + \sum_{1 \leq i < j \leq n} P_{ij}^{\mathsf{A}} \,. \tag{1}$$

Observing that, for all $i \neq j \in [n]$, if i terminates before j then $P_{ij}^{\mathsf{A}} \geq x_i$, otherwise $P_{ij}^{\mathsf{A}} \geq x_j$, we deduce that $P_{ij}^{\mathsf{A}} \geq \min(x_i, x_j)$. Equality is achieved by the clairvoyant algorithm that runs the jobs until completion in non-decreasing size order (Motwani et al., 1994), which is the optimal offline algorithm, that we denote OPT, satisfying

$$\mathsf{OPT}(x) = \sum_{i=1}^{n} x_i + \sum_{1 \leq i < j \leq n} \min(x_i, x_j) \,. \tag{2}$$

When the predictions are perfect, for all $n \geq 2$ and $B \leq n$, we define the (n, B)-competitive ratio of algorithm A as the worst-case ratio between its objective, knowing the sizes of B jobs taken uniformly at random, and that of OPT, on instances of n jobs

$$\mathsf{R}_{n,B}(\mathsf{A}) = \sup_{x \in (0,\infty)^n} \frac{\mathbb{E}[\mathsf{A}(x)]}{\mathsf{OPT}(x)} \,,$$

where the expectation $\mathbb{E}[\mathsf{A}(x)]$ is taken over the permutation σ and the actions of A if it a randomized algorithm.

If the number of predictions depends on the number of jobs, i.e. $B = (B_n)_{n \geq 1}$ defines a sequence of integers, then the competitive ratio of A is defined by

$$\mathsf{CR}_B(\mathsf{A}) = \sup_{n \geq 2} \mathsf{R}_{n,B_n}(\mathsf{A}) \,.$$

When the predictions are imperfect, the competitive ratio becomes also a function of $\mathbb{E}[\eta^\sigma]$.

2. Lower Bounds

In this section, we assume that the predictions are error-free, and we establish lower bounds on the (n, B)-competitive ratio of any algorithm, followed by a lower bound independent of n when $B_n = wn + o(n)$ for some $w \in [0, 1]$. These lower bounds are obtained by constructing random job size instances x such that, for any deterministic algorithm A, the ratio $\mathbb{E}_{\sigma,x}[\mathsf{A}(x)]/\mathbb{E}_x[\mathsf{OPT}(x)]$ is above them. The result then extends to randomized algorithms and yields bounds on their (n, B)-competitive ratio by using Lemma A.1, which is a consequence of Yao's principle (Yao, 1977).

In all this section, we consider i.i.d. job sizes. Therefore, we can assume without loss of generality that the B known job sizes are $x_1, \dots, x_B$.

In the non-clairvoyant case $B = 0$, for any algorithm A, taking i.i.d. exponentially distributed sizes gives, with easy computation, that $\mathbb{E}[P_{ij}^{\mathsf{A}}] = 1$ (Remark A.4), which yields, using Equations (1) and (2), the lower bound $2 - \frac{4}{n+3}$ on the competitive ratio (Motwani et al., 1994). However, if $B > 0$, the algorithm can act according to the information it has on $(x_i)_{i \in I}$, and the dependence between its actions and these job sizes makes the analysis more sophisticated.

For any positive and continuous function φ, and positive numbers $T \geq x > 0$ we denote

$$G_\varphi(x, T) = \int_0^{T-x} \frac{dt}{\varphi(t)} + \frac{x}{\varphi(T-x)} \ . \tag{3}$$

We prove in the following theorem a generic lower bound, using job sizes sampled independently from the distribution $\Pr(x_i \leq t) = 1 - \frac{\varphi(0)}{\varphi(t)}$.

Theorem 2.1. *Let $\varphi : [0, \infty) \to [0, \infty)$ be a continuously differentiable and increasing function satisfying $\varphi(0) > 0$, φ'/φ is non-increasing and $\int_0^\infty \frac{dt}{\varphi(t)^2} < \infty$, and let α_φ a non-negative constant satisfying*

$$\int_0^\infty \left\{ \inf_{T \geq x} G_\varphi(x, T) \right\} \frac{\varphi'(x)}{\varphi(x)^2} dx \geq \alpha_\varphi \int_0^\infty \frac{dt}{\varphi(t)^2} \ . \tag{4}$$

If $B = wn + o(n)$ for some $w \in [0, 1]$, then it holds for any randomized algorithm A that

$$\mathsf{CR}_B(\mathsf{A}) \geq 2 - 2(2 - \alpha_\varphi)w + (3 - 2\alpha_\varphi)w^2 \ .$$

Moreover, if $\int_0^\infty \frac{dt}{\varphi(t)} < \infty$, then for all $n \geq 2$ and $B \leq n$

$$\mathsf{R}_{n,B}(\mathsf{A}) \geq C_{\varphi,n,B} - \frac{C_{\varphi,n,B} - 1}{1 + \frac{n-1}{2} \frac{\int_0^\infty \frac{dt}{\varphi(t)^2}}{\int_0^\infty \frac{dt}{\varphi(t)}}} \ ,$$

where $C_{\varphi,n,B} = (2 - \frac{B}{n}) - (3 - 2\alpha_\varphi)\frac{B}{n}\big(1 - \frac{B-1}{n-1}\big)$.

To establish this theorem, we analyze i.i.d. job sizes sampled from the distribution $\Pr(x_i \leq t) = 1 - \frac{\varphi(0)}{\varphi(t)}$. We derive in Lemma A.5 a lower bound on the mutual delays incurred by these jobs during the run of any algorithm A. This involves solving a functional minimization problem, whose solution is expressed using the function G_φ defined in (3). The left term in Inequality (4) is proportional to the obtained lower bound, while the right term is proportional to $\mathbb{E}[\min(x_i, x_j)]$, which is the mutual delay caused in a run of OPT. Finally, using the identity (1), this inequality, which relates the mutual delays caused respectively by executing A and OPT on the chosen job sizes, can be extended to an inequality involving the objectives of both algorithms, giving a lower bound on the competitive ratio.

If $\int_0^\infty \frac{dt}{\varphi(t)} = \infty$, the expectation of the job sizes is infinite. In this case, we consider a truncated distribution with a maximum $a > 0$. After completing the analysis, we derive a lower bound that depends on a and w, by considering $B = wn + o(n)$ and $n \to \infty$, then, we conclude by taking the limit $a \to \infty$.

For any value α_φ in Theorem 2.1, observe that $C_{\varphi,n,0} = 2$ and $C_{\varphi,n,n} = 1$, which means that the lower bound interpolates properly the non-clairvoyant and clairvoyant settings. The remaining task is to choose an adequate function φ satisfying the conditions of the theorem with α_φ as large as possible. We first consider exponentially distributed job sizes, often used to prove lower bounds in scheduling problems. This corresponds to $\varphi(t) = e^t$.

Corollary 2.2. *For any algorithm A, it holds that*

$$\mathsf{R}_{n,B}(\mathsf{A}) \geq C_{n,B} - \frac{4(C_{n,B} - 1)}{n + 3} \ ,$$

with $C_{n,B} = 2 - \frac{B}{n} - (\frac{4}{e} - 1)\frac{B}{n}\big(1 - \frac{B-1}{n-1}\big)$. In particular, if $B = wn + o(n)$ then

$$\mathsf{CR}_B(\mathsf{A}) \geq (2 - w) - (\tfrac{4}{e} - 1)w(1 - w) \ .$$

Corollary 2.2 gives, in particular, that $\mathsf{R}_{n,0}(\mathsf{A}) \geq 2 - \frac{4}{n+3}$, which corresponds exactly to the tight lower bound for the non-clairvoyant scheduling problem (Motwani et al., 1994). However, the bound is not tight for all values of $B \leq n$. To refine it, we consider distributions that would be more difficult to process by the algorithm. One idea is to sample a different parameter $\lambda_i \sim \mathcal{E}(1)$ independently for each $i \in [n]$, then sample $x_i \sim \mathcal{E}(\lambda_i)$. The distribution of x_i in this case is given by

$$\begin{aligned} \Pr(x_i \geq t) &= \int_0^\infty \Pr(x_i \geq t \mid \lambda_i = \lambda) e^{-\lambda} d\lambda \\ &= \int_0^\infty e^{-(1+t)\lambda} d\lambda = \frac{1}{1+t} \ , \end{aligned}$$

which corresponds to $\varphi(t) = 1 + t$. More generally, we consider $\varphi(t) = (1 + t)^r$ for $r \in (\frac{1}{2}, 1]$. Such functions

φ correspond to distributions with a heavy tale and with infinite expectation (i.e. $\int_0^\infty \frac{dt}{\varphi(t)} = \infty$). Using Theorem 2.1, they only yield lower bounds on the competitive ratio but not on $\mathsf{R}_{n,B}$. Corollary 2.3 shows the bound obtained for $r \to \frac{1}{2}$.

Corollary 2.3. *Let $w \in [0,1]$. If $B = wn + o(n)$, then it holds for any algorithm A that*

$$\mathsf{CR}_B(\mathsf{A}) \geq (2-w) - (3-2\sqrt{2})w(1-w) \ .$$

Remark 2.4. If x_i is sampled from the distribution induced by $\varphi(t) = (1+t)^r$, then $x_i + 1$ follows a Pareto distribution with scale 1 and shape r (Arnold, 2014), which is commonly used to model the distribution of job sizes in the context of the scheduling problem.

3. Known Partial Order

Before investigating the problem within the learning-augmented framework, we introduce an algorithm exclusively for the scenario with perfect information. Subsequently, in Section 4, we present a second algorithm, that we adapt to handle possibly erroneous predictions.

The optimal algorithm OPT does not necessitate precise knowledge of job sizes. Instead, it relies solely on their ordering. This observation suggests that it might be possible to improve the competitive ratio of the non-clairvoyant case by only knowing the relative order of a subset of the job sizes. Therefore, rather than having access to the values $x_{\sigma(1)}, \ldots, x_{\sigma(B)}$, we assume that the decision-maker is only given a priority ordering π of them, i.e. a bijection $\pi : [B] \to \sigma([B])$ satisfying $x_{\pi(1)} \leq \ldots \leq x_{\pi(B)}$.

Algorithm 1 Catch-up and Resume Round-Robin (CRRR)

```
Input: Ordering π of x_{σ(1)}, ..., x_{σ(1)}
Set x_{π(0)} = 0
for i = 1 to B do
    Run job π(i) for x_{π(i-1)} units of time
    while job π(i) is not finished do
        Run round-robin on {σ(j)}_{j>B} ∪ {π(i)}
    end while
end for
Run round-robin on {σ(j)}_{j=B+1}^{n} until completion
```

In Algorithm 1 (CRRR), for all $i \in [B]$, the execution of job $\pi(i)$ starts only upon the completion of job $\pi(i-1)$. At this moment, all jobs $\sigma(j)$ for $j > B$ are either completed or have undergone execution for $x_{\pi(i-1)}$ units of time. CRRR then runs job $\pi(i)$ for a period of length $x_{\pi(i-1)}$ to catch up with the progress of the jobs $\{\sigma(j)\}_{j>B}$. Following this synchronization phase, it runs round-robin on the set of jobs $\{\sigma(j)\}_{j>B} \cup \{\pi(i)\}$ until $\pi(i)$ terminates. The same process iterates with $\pi(i+1)$ afterward. Once all the jobs $\sigma(i)$ for $i \in [B]$ are completed, the algorithm runs round robin on the unfinished jobs in $\{\sigma(j)\}_{j>B}$.

Leveraging the ordering π, the algorithm aims to minimize the delays caused by longer jobs to shorter ones. In the ideal scenario where $B = n$, each job begins execution only after all shorter ones have been completed. When $B < n$, it is evident that the jobs $\{\sigma(i)\}_{i\in[B]}$ should be executed in the order specified by π. However, CRRR takes advantage of this ordering even further, ensuring that job $x_{\pi(i)}$ not only avoids delaying $x_{\pi(i-1)}$ but also does not delay any job $\sigma(j)$ with $j > B$ that has a size at most $x_{\pi(i-1)}$.

Theorem 3.1. *Algorithm CRRR satisfies*

$$2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{(n+1)(B+1)} \leq \mathsf{R}_{n,B}(\mathsf{CRRR}) \leq 2 - \frac{B}{n} \ .$$

Moreover, if $B = \lfloor wn \rfloor$ for some $w \in [0,1]$, then $\mathsf{CR}(\mathsf{CRRR}) = 2 - w$.

Theorem 3.1 shows a substantially stronger result than the one presented in (Benomar & Perchet, 2023), where the algorithm leveraging the values of the job sizes $x_{\sigma(1)}, \ldots, x_{\sigma(B)}$ is only $\left(2 - \frac{B(B-1)}{n(n-1)}\right)$-competitive.

Action predictions The information provided to CRRR is the order in which the jobs $\{\sigma(i)\}_{i\in[B]}$ would be executed by OPT. This corresponds to the error-free scenario of *action predictions* (Antoniadis et al., 2023b; Lindermayr & Megow, 2022; Lassota et al., 2023; Sadek & Elias, 2024), where the decision-maker receives predictions regarding the actions taken by the optimal offline algorithm, rather than numeric predictions of unknown parameters. In the context of the scheduling problem, utilizing the ℓ_1 norm to measure the error is not ideal for analyzing the action prediction setting (Im et al., 2021). Alternative error metrics, which account for the number of inversions in the predicted permutation in comparison to the true one (Lindermayr & Megow, 2022), would be more suitable. Therefore, adapting CRRR to imperfect action predictions is left for future research as it requires different considerations. For now, we shift our focus to introducing another algorithm that utilizes not only the priority order induced by the job sizes, but the size values themselves.

4. Predictions of the Job Sizes

We propose in this section a generic algorithm Switch, which we will adapt in the cases of perfect and imperfect predictions. The algorithm takes as input n jobs with unknown sizes and breakpoints $z_{\sigma(1)}, \ldots, z_{\sigma(B)}$ that depend on the predictions of $x_{\sigma(1)}, \ldots, x_{\sigma(B)}$, then it alternates running round-robin on the jobs $\{\sigma(j)\}_{j>B}$ and *Shortest Predicted Job First* (SPJF), introduced in (Purohit et al., 2018), on the

jobs $\{\sigma(i)\}_{i\in[B]}$, where the moment of switching from an algorithm to another is determined by the breakpoints.

As in Section 3, we call *ordering* of $z_{\sigma(1)}, \dots, z_{\sigma(B)}$ any bijective application $\pi : [B] \to \sigma([B])$ satisfying $z_{\pi(1)} \leq \dots \leq z_{\pi(B)}$. Note that, if the breakpoints are not pairwise distinct, then the ordering is not unique. In that case, Switch chooses an ordering π uniformly at random. We assume furthermore that the breakpoints induce the same order as the predictions, i.e. $z_{\sigma(i)} < z_{\sigma(j)} \iff y_{\sigma(i)} < y_{\sigma(j)}$ for all $i, j \in [B]$.

Algorithm 2 Switch algorithm Switch(z^σ, x)

Input: Breakpoints $z^\sigma = (z_{\sigma(i)})_{i\in[B]}$
$\pi \leftarrow$ ordering of z^σ chosen uniformly at random
for $i = 1$ **to** B **do**
  while $\min_{j>B} \frac{S_{\sigma(j)}(t)}{x_{\sigma(j)}} < 1$ and $\max_{j>B} S_{\sigma(j)}(t) < z_{\pi(i)}$ **do**
    Run round-robin on $\{\sigma(j)\}_{j=B+1}^n$
  end while
  Run job $\pi(i)$ until completion
end for
Run round-robin on $\{\sigma(j)\}_{j=B+1}^n$ until completion

Consider a run of Switch, and let $i \in [B]$. The first condition for entering the while loop is the existence of $j > B$ such that $S_{\sigma(j)}(t) < x_{\sigma(j)}$. This signifies that the jobs $\{\sigma(j)\}_{j>B}$ are not all completed, which is a verification feasible for the decision-maker without knowledge of the sizes $\{x_{\sigma(j)}\}_{j>B}$. The second condition means that no job $\sigma(j)$ for $j > B$ has been in execution for more than $z_{\pi(i)}$ units of time. Given that round-robin allocates equal importance to all jobs $\{\sigma(j)\}_{j>B}$, upon exiting the while loop, each job $\sigma(j)$ is either completed or has been in execution for precisely $z_{\pi(i)}$ units of time. Following this step, job $\pi(i)$ is executed until completion, and the same process recurs for $i + 1$.

This algorithm ensures that any job $x_{\sigma(i)}$ with $i \leq B$ does not delay any other job j whose size is at most $x_j \leq z_{\sigma(i)}$, and the delay it causes to jobs not satisfying this condition is exactly $x_{\sigma(i)}$. This allows efficient control of the mutual delays between the jobs by conveniently choosing the breakpoints.

4.1. Perfect Predictions

Assuming that the predictions are perfect, i.e. the decision-maker knows the exact sizes of jobs $\sigma(1), \dots, \sigma(B)$, it is possible to set $z_{\sigma(i)} = x_{\sigma(i)}$ for all $i \in [B]$.

Theorem 4.1. *Algorithm Switch with breakpoints $z_{\sigma(i)} = x_{\sigma(i)}$ for all $i \in [B]$ satisfies*

$$R_{n,B}(\textit{Switch}) = 2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{n+1} .$$

In particular, if $B = \lfloor wn \rfloor$ for some $w \in [0,1]$ then $CR_B(\textit{Switch}) = 2 - w$.

Note that the (n, B)-competitive ratio above is strictly better than that of CRRR, presented in Theorem 3.1. However, both algorithms have equivalent performance when n is large. In particular, their competitive ratios coincide when $B = \lfloor wn \rfloor$.

A slight improvement on the (n, B)-competitive ratio can be obtained by introducing randomness into Switch. Indeed, consider the *Run To Completion* algorithm (RTC) defined in (Motwani et al., 1994), executing all the jobs until completion in a uniformly random order. Then we have the following result.

Proposition 4.2. *The algorithm that runs RTC with probability $\frac{2(n-B)}{n(n+3)-2B}$, and runs Switch with breakpoints $z_{\sigma(i)} = x_{\sigma(i)}$ for all $i \in [B]$ with the remaining probability, has an (n, B)-competitive ratio of*

$$2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})(2-\frac{B}{n})}{n+3-\frac{2B}{n}} .$$

For $B = 0$, the ratio above becomes $2 - \frac{4}{n+3}$, which is the best possible in the non-clairvoyant setting.

4.2. Imperfect Predictions

We assume in this section that no quality guarantees are given on the predictions $\{y_{\sigma(i)}\}_{i\in[B]}$. Recall that the total error $\eta^\sigma = \sum_{i=1}^B |x_{\sigma(i)} - y_{\sigma(i)}|$ is a random variable because σ is a uniformly random permutation of $[n]$, hence our results will depend on $\mathbb{E}[\eta^\sigma]$.

The goal is to design an algorithm that is consistent, robust, and with a competitive ratio having a smooth dependency to $\mathbb{E}[\eta^\sigma]$. We first study the consistency and smoothness of Switch with well-chosen breakpoints, then we show that combining it with round-robin as in (Purohit et al., 2018; Lassota et al., 2023) gives robustness guarantees.

Using the trivial breakpoints $z_{\sigma(i)} = y_{\sigma(i)}$ as in the previous section is not enough to guarantee smoothness. Consider, for example, job sizes all equal to 1, and B predictions $y_{\sigma(i)} = 1 - \epsilon$ for an arbitrarily small ϵ. Blindly following these predictions, taking $z_{\sigma(i)} = y_{\sigma(i)}$ for all $i \in [B]$, results in delaying all jobs with unknown sizes by B time units compared to the case of perfect predictions. This creates a discontinuity in the competitive ratio when ϵ becomes positive, proving non-smoothness. Hence, we consider instead randomized breakpoints.

Algorithm 3 simply runs Switch with random breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$. The following lemma gives an upper bound on the algorithm's objective function depending on the distribution F of ξ.

Algorithm 3 imperfect predictions $\mathsf{Switch}(\xi y^\sigma, x)$

Input: predictions $(y_{\sigma(i)})_{i\in[B]}$, distribution F
Sample $\xi \sim F$
Run Switch with breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$

Lemma 4.3. *Let F be a probability distribution on $(0,\infty)$, and consider the mappings $h_F : (0,\infty)^2 \to \mathbb{R}$ and $g_F : (0,\infty) \to \mathbb{R}$ defined by*

$$h_F(s,t) = t \cdot \Pr_{\xi\sim F}(\xi < \tfrac{s}{t})$$
$$g_F(s) = (1-s)\Pr_{\xi\sim F}(\xi < s) + \mathbb{E}_{\xi\sim F}[\xi \mathbb{1}_{\xi<s}] \ .$$

Let $\beta_F = \sup_{s\in(0,1]} \frac{g_F(s)}{s}$ and $\gamma_F = \sup_{s\geq 1}(g_F(s)+s)$. If $\beta_F, \gamma_F < \infty$ and h_F is L_F-Lipschitz w.r.t. the second variable t, then for any job sizes $x_1,\dots,x_n$ and $B \leq n$, the expected sum of the completion times achieved by Switch with breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$ is at most

$$\sum_{i=1}^n x_i + C^1_{n,B,F}\sum_{i<j}\min(x_i,x_j) + C^2_{n,B,F}\mathbb{E}[\eta^\sigma] \ ,$$

with $C^1_{n,B,F} = 2 - \frac{B}{n} - \big(2-\beta_F-\gamma_F\big)\frac{B}{n}\big(1-\frac{B-1}{n-1}\big)$ and $C^2_{n,B,F} = (1+L_F+\mathbb{E}[\xi])(n-B)+B-1$.

A trivial choice of ξ is the constant random variable equal to 1 a.s., but this is not enough to guarantee smoothness, as it corresponds to the Dirac distribution $F = \delta_1$, for which h_F is not continuous w.r.t. to t. In the next lemma, we provide a specific choice of distribution F depending on a single parameter ρ, and we express the upper bound from the previous lemma using this parameter.

Lemma 4.4. *Let $\rho \in (0,1]$ and*

$$F : s \mapsto (1-e^{-(s-1)/\rho})\mathbb{1}_{s>1}$$

a shifted exponential distribution with parameter $1/\rho$, i.e. $\xi \sim 1+\mathcal{E}(1/\rho)$, then Switch with breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$ for all $i \in [B]$ has an (n,B)-competitive ratio of at most

$$\Big(2-\tfrac{B}{n}+\rho\tfrac{B}{n}(1-\tfrac{B-1}{n-1})\Big)+\Big(\tfrac{4}{\rho}(1-\tfrac{B}{n})+\tfrac{B}{n}\Big)\frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} \ .$$

The previous lemma highlights a tradeoff between the smoothness and the consistency of the algorithm. Indeed, as ρ decreases, the algorithm gains in consistency, but the term $\Big(\frac{4}{\rho}(1-\frac{B}{n})+\frac{B}{n}\Big)$ multiplying $\mathbb{E}[\eta^\sigma]$ becomes larger. However, while setting ρ close to zero results in an arbitrarily high sensitivity to the error, setting it close to 1 gives consistency of at most $\big(2-\frac{B(B-1)}{n(n-1)}\big)$, which is still a decreasing function of B, interpolating the values 2 and 1 in the clairvoyant and non-clairvoyant cases. This implies that sacrificing a small amount of consistency significantly improves smoothness.

For $B = n$, assuming that all the job sizes are at least 1, it holds that $\mathsf{OPT}(x) \geq n(n+1)/2$ and the lemma gives $\mathsf{R}_{n,B}(\eta;\mathsf{Switch}) \leq 1+\frac{2\eta}{n}$, matching the bound proved on (SPJF) in (Purohit et al., 2018). On the other hand, if $\eta = 0$, setting $\rho = 0$ results in $\mathsf{R}_{n,B}(0;\mathsf{Switch}) \leq 2-\frac{B}{n}$. Using tighter inequalities in the proof, it is possible to retrieve the bound established in Theorem 4.1 (See Inequality (36) in Appendix C).

Preferential algorithm Now we need to adapt the algorithm to guarantee robustness in the face of arbitrarily erroneous predictions. We use the same approach as Lemma 3.1 of (Purohit et al., 2018), which consists of running concurrently a consistent algorithm and round-robin at respective rates λ, $1-\lambda$ for some $\lambda \in [0,1]$. However, their result only applies for deterministic algorithms A satisfying for any instances $x = (x_1,\dots,x_n)$ and $x' = (x'_1,\dots,x'_n)$ that

$$\big(\forall i \in [n] : x_i \leq x'_i\big) \implies \mathsf{A}(x) \leq \mathsf{A}(x') \ .$$

Such algorithms are called *monotonic*. Switch with breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$ is not deterministic since its objective function depends both on σ and ξ. Nonetheless, we overcome this difficulty by proving that, conditionally to σ and ξ, its outcome is deterministic and monotonic, then we establish the following theorem.

Theorem 4.5. *Let $\rho \in (0,1]$ and $F = 1+\mathcal{E}(1/\rho)$. Then the preferential algorithm ALG_λ which runs Algorithm 3 at rate λ and round-robin at rate $1-\lambda$ has an (n,B)-competitive ratio of at most*

$$\min\left(\frac{2}{1-\lambda}\ ,\ \frac{C_{\rho,n,B}}{\lambda}+\frac{S_{\rho,n,B}}{\lambda}\cdot\frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)}\right) \ ,$$

with

$$C_{\rho,n,B} = \big(2-\tfrac{B}{n}\big)+\rho\tfrac{B}{n}\big(1-\tfrac{B-1}{n-1}\big)$$
$$S_{\rho,n,B} = \tfrac{4}{\rho}\big(1-\tfrac{B}{n}\big)+\tfrac{B}{n} \ .$$

This upper bound generalizes that of (Purohit et al., 2018). It presents a consistency-robustness tradeoff that can be tuned by adjusting the parameter λ, and a consistency-smoothness tradeoff controlled by the parameter ρ, which vanishes for B close to 0 or n, as the terms multiplying ρ and $1/\rho$ respectively in $C_{\rho,n,B}$ and $S_{\rho,n,B}$ become zero.

5. Experiments

In this section, we validate our theoretical findings by testing the algorithms we presented on various benchmark job sizes. In all the figures, each point is averaged over 10^4 independent trials.

Perfect information We test the performance of Algorithms Switch and CRRR against the hard instances used to prove the lower bounds of Section 2: we consider i.i.d. job sizes sampled from the exponential distribution with parameter 1, and job sizes drawn from the distribution $\Phi(r, a)$ with parameters $r = 0.51$ and $a = 10^4$, characterized by the tail probability

$$\Pr(x_i^a \geq t) = \frac{(1+t)^{-r} - (1+a)^{-r}}{1-(1+a)^{-r}} \mathbb{1}_{t<a} \ ,$$

This distribution is a truncated version of the one defined by $\Pr(x_i^\infty \geq t) = \frac{1}{(1+t)^r}$. The bound of Corollary 2.3 is obtained by using this distribution for $a > 0$ and $r \in (\frac{1}{2}, 1)$, and taking the limits $n \to \infty$, $a \to \infty$, and $r \to 1/2$.

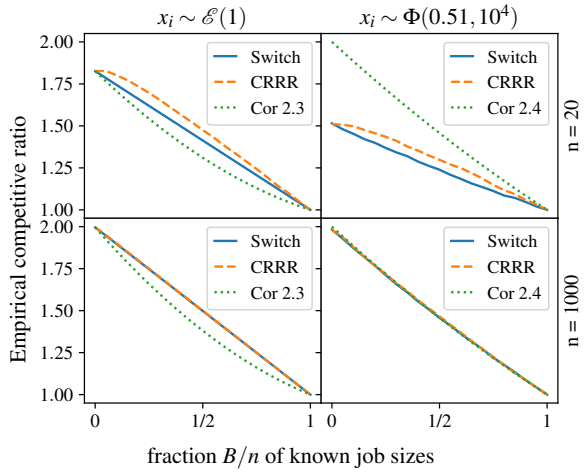

Figure 2. Lower bounds and ratios of Switch, CRRR

Figure 2 exhibits the empirical ratios achieved by both algorithms with a number $n \in \{20, 1000\}$ of jobs. For $n = 20$, Switch outperforms CRRR for the two distributions, whereas their ratios are very close for $n = 1000$. This confirms that Switch and CRRR are asymptotically equivalent, as can be deduced from Theorems 3.1 and 4.1. For the exponential distribution, as expected, both algorithms have ratios above the non-asymptotic lower bound of Corollary 2.2. Meanwhile, considering the distribution $\Phi(0.51, 10^4)$, the empirical ratios for $n = 20$ are below the lower bound of Corollary 2.3, because it is proved by taking $n \to \infty$. For $n = 1000$, the ratios match the lower bound.

Preferential algorithm In the remaining discussion, we refer to Switch with breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$ and $\xi \sim 1 + \mathcal{E}(1/\rho)$, as Switch with parameter ρ.

We generate a synthetic instance of $n = 50$ job sizes, drawn independently from the Pareto distribution with scale 1 and shape 1.1. The Pareto distribution, known for its heavy tail, is particularly suitable for modeling job sizes (Harchol-Balter & Downey, 1997; Bansal & Harchol-Balter, 2001; Arnold, 2014), and it is a commonly used benchmark for learning-augmented scheduling algorithms (Purohit et al., 2018; Lindermayr & Megow, 2022). Furthermore, we consider noisy predictions $y_i = x_i + \varepsilon_i$ for all $i \in [50]$, where ε_i is sampled independently from a normal distribution with mean 0 and standard deviation τ.

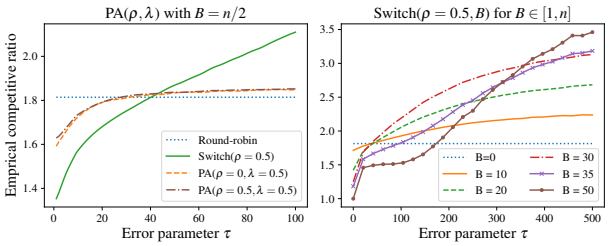

Figure 3. Preferential Algorithm (PA) with different parameters

Figure 3 illustrates the empirical ratio of the Preferential Algorithm (PA) across various parameter configurations, with varying error parameter τ.

The left plot displays the ratios for different λ and ρ values, with $B = 25 = n/2$. When $\lambda = 0$, PA becomes round-robin. For $\lambda = 1$ and $\rho = 0.5$, PA simply runs Switch($\rho = 0.5$), which gives an improved consistency ($\tau = 0$), not equal to 1 because $B < n$ and $\rho > 0$, and gives a ratio that deteriorates arbitrarily as τ increases. In contrast, PA with $\lambda = 0.5$ gives a weaker consistency but maintains bounded ratios, even with arbitrarily erroneous predictions. The choice of $\rho = 0$ exhibits a slightly better consistency compared to $\rho = 0.5$, in line with theoretical expectations, but there is no significant difference regarding sensitivity to errors. This should not be surprising since setting $\rho > 0$ ensures smoothness in the worst-case (see Figure 4), but it is not necessarily needed for all instances.

The right plot examines the influence of B on PA with parameters $\lambda = 1$ and $\rho = 0.5$, which corresponds to Switch with $\rho = 0.5$. Larger B values improve consistency and also yield a smaller sensitivity to small errors. However, for high τ values, having numerous predictions leads to faster performance deterioration compared to having fewer predictions. This shows that more predictions enhance consistency, while fewer predictions enhance robustness.

Consistency-smoothness To shed light on the tradeoff between consistency and smoothness raised in Section 4.2, we consider i.i.d. job sizes $x_1, \ldots, x_{100}$, each taking the value 1 w.p. 1/2 and 2 w.p. 1/2, and we consider noisy predictions of the form $y_i = x_i + \varepsilon_i$, where ε_i follows a uniform distribution over $[-\tau, \tau]$. Figure 4 illustrates the evolution, for τ varying in $[0, 0.15]$, of the empirical competitive ratio of Switch with parameter $\rho \in \{0, 0.1, 0.5\}$

and $B \in \{50, 95\}$,

For both values of B, the experiment reveals that larger values of ρ give bigger ratios when $\tau = 0$ (less consistency), but on the other hand they yield less sensitivity to variations of the expected prediction error (better smoothness), which confirms our theoretical findings. In particular, for $\rho = 0$, a significant discontinuity arises when τ becomes positive. Figure 4 also shows that this tradeoff is less significant as B approaches $n = 100$, with the consistency values for $\rho \in \{0, 0.1, 0.5\}$ drawing closer.

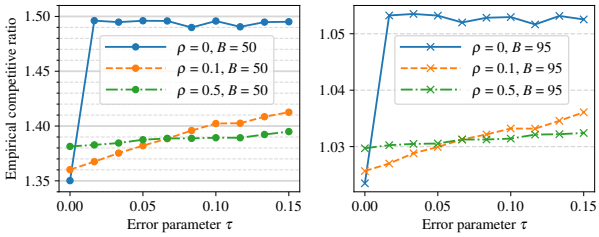

Figure 4. Tradeoff between consistency and smoothness

6. Conclusion and Future Work

This paper explores the non-clairvoyant scheduling problem with a limited number of predicted job sizes. We give near optimal lower and upper bounds in the case of perfect predictions, and we introduce a learning-augmented algorithm raising the common consistency-robustness tradeoff and an additional consistency-smoothness tradeoff, the latter vanishing when B approaches 0 or n.

Our findings join previous works in demonstrating that online algorithms can indeed achieve improved performance even when armed with a restricted set of predictions, which is an assumption more aligned with practical scenarios. Furthermore, they affirm the necessity of studying and understanding these regimes, as they may unveil unique behaviors absent in the zero- or full-information settings.

6.1. Open Questions

Tight lower bounds In the case of perfect predictions, there is a (small) gap between the lower bounds of Section 2 and the competitive ratios of Switch and CRRR. An interesting research avenue is to close this gap, either by designing better algorithms or improving the lower bound. This could involve using Theorem 2.1 with more refined distributions.

Reduced number of action predictions Algorithm Switch leverages the job sizes' predictions, not only the order they induce. Using the ℓ_1 norm to measure the error is thus a suitable choice. However, as discussed in Section 3, Algorithm CRRR only uses the priority order in which OPT runs $(x_{\sigma(i)})_{i\in[B]}$. An interesting question to explore is how to adapt it in the case of imperfect *action predictions*, using appropriate error measures.

Smooth and $(2 - \frac{B}{n})$-consistent algorithm Lemma 4.4 and Figure 4 emphasize that, to achieve smoothness, Switch with parameter ρ must exhibit a consistency exceeding $2-\frac{B}{n}$. A compelling question arises: Is it possible to devise a smooth algorithm with a consistency of at most $2 - \frac{B}{n}$?

Impact Statement

This paper presents a work whose goal is to advance the field of learning-augmented algorithms. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgements

Vianney Perchet acknowledges support from the French National Research Agency (ANR) under grant number (ANR19-CE23-0026 as well as the support grant, as well as from the grant "Investissements d'Avenir" (LabEx Ecodec/ANR11-LABX-0047).

References

Anand, K., Ge, R., and Panigrahi, D. Customizing ml predictions for online algorithms. In *International Conference on Machine Learning*, pp. 303–313. PMLR, 2020.

Antoniadis, A., Gouleakis, T., Kleer, P., and Kolev, P. Secretary and online matching problems with machine learned advice. *Advances in Neural Information Processing Systems*, 33:7933–7944, 2020.

Antoniadis, A., Coester, C., Eliás, M., Polak, A., and Simon, B. Learning-augmented dynamic power management with multiple states via new ski rental bounds. *Advances in Neural Information Processing Systems*, 34:16714–16726, 2021a.

Antoniadis, A., Ganje, P. J., and Shahkarami, G. A novel prediction setup for online speed-scaling. *arXiv preprint arXiv:2112.03082*, 2021b.

Antoniadis, A., Boyar, J., Eliás, M., Favrholdt, L. M., Hoeksma, R., Larsen, K. S., Polak, A., and Simon, B. Paging with succinct predictions. In *International Conference on Machine Learning*, pp. 952–968. PMLR, 2023a.

Antoniadis, A., Coester, C., Eliáš, M., Polak, A., and Simon, B. Online metric algorithms with untrusted predictions. *ACM Transactions on Algorithms*, 19(2):1–34, 2023b.

Arnold, B. C. Pareto distribution. *Wiley StatsRef: Statistics Reference Online*, pp. 1–10, 2014.

Azar, Y., Leonardi, S., and Touitou, N. Flow time scheduling with uncertain processing time. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1070–1080, 2021.

Bai, X. and Coester, C. Sorting with predictions. *Advances in Neural Information Processing Systems*, 36, 2024.

Baker, K. R. and Trietsch, D. *Principles of sequencing and scheduling*. John Wiley & Sons, 2013.

Bamas, É., Maggiori, A., Rohwedder, L., and Svensson, O. Learning augmented energy minimization via speed scaling. *Advances in Neural Information Processing Systems*, 33:15350–15359, 2020a.

Bamas, E., Maggiori, A., and Svensson, O. The primal-dual method for learning augmented algorithms. *Advances in Neural Information Processing Systems*, 33:20083–20094, 2020b.

Bampis, E., Dogeas, K., Kononov, A. V., Lucarelli, G., and Pascual, F. Scheduling with untrusted predictions. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI*, pp. 4581–4587, 2022.

Bansal, N. and Harchol-Balter, M. Analysis of srpt scheduling: Investigating unfairness. In *Proceedings of the 2001 ACM SIGMETRICS International conference on Measurement and modeling of computer systems*, pp. 279–290, 2001.

Bateni, M., Dharangutte, P., Jayaram, R., and Wang, C. Metric clustering and mst with strong and weak distance oracles. *arXiv preprint arXiv:2310.15863*, 2023.

Benomar, Z. and Perchet, V. Advice querying under budget constraint for online algorithms. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Benomar, Z., Chzhen, E., Schreuder, N., and Perchet, V. Addressing bias in online selection with limited budget of comparisons. *arXiv preprint arXiv:2303.09205*, 2023.

Bhaskara, A., Cutkosky, A., Kumar, R., and Purohit, M. Logarithmic regret from sublinear hints. *Advances in Neural Information Processing Systems*, 34:28222–28232, 2021.

Borodin, A. and El-Yaniv, R. *Online computation and competitive analysis*. cambridge university press, 2005.

Chen, J., Silwal, S., Vakilian, A., and Zhang, F. Faster fundamental graph algorithms via learned predictions. In *International Conference on Machine Learning*, pp. 3583–3602. PMLR, 2022.

Cheng, T. and Sin, C. A state-of-the-art review of parallel-machine scheduling research. *European Journal of Operational Research*, 47(3):271–292, 1990.

Chlkedowski, J., Polak, A., Szabucki, B., and .Zolna, K. T. Robust learning-augmented caching: An experimental study. In *International Conference on Machine Learning*, pp. 1920–1930. PMLR, 2021.

Christianson, N., Shen, J., and Wierman, A. Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems. In *International Conference on Artificial Intelligence and Statistics*, pp. 9377–9399. PMLR, 2023.

Diakonikolas, I., Kontonis, V., Tzamos, C., Vakilian, A., and Zarifis, N. Learning online algorithms with distributional advice. In *International Conference on Machine Learning*, pp. 2687–2696. PMLR, 2021.

Dinitz, M., Im, S., Lavastida, T., Moseley, B., and Vassilvitskii, S. Faster matchings via learned duals. *Advances in neural information processing systems*, 34:10393–10406, 2021.

Dinitz, M., Im, S., Lavastida, T., Moseley, B., and Vassilvitskii, S. Algorithms with prediction portfolios. *Advances in neural information processing systems*, 35: 20273–20286, 2022.

Drygala, M., Nagarajan, S. G., and Svensson, O. Online algorithms with costly predictions. In *International Conference on Artificial Intelligence and Statistics*, pp. 8078–8101. PMLR, 2023.

Dütting, P., Lattanzi, S., Paes Leme, R., and Vassilvitskii, S. Secretaries with advice. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pp. 409–429, 2021.

Eberle, F., Hommelsheim, F., Lindermayr, A., Liu, Z., Megow, N., and Schlöter, J. Accelerating matroid optimization through fast imprecise oracles. *arXiv preprint arXiv:2402.02774*, 2024.

Gollapudi, S. and Panigrahi, D. Online algorithms for rent-or-buy with expert advice. In *International Conference on Machine Learning*, pp. 2319–2327. PMLR, 2019.

Graham, R. L., Lawler, E. L., Lenstra, J. K., and Kan, A. R. Optimization and approximation in deterministic sequencing and scheduling: a survey. In *Annals of discrete mathematics*, volume 5, pp. 287–326. Elsevier, 1979.

Hall, N. G. and Potts, C. N. Supply chain scheduling: Batching and delivery. *Operations Research*, 51(4):566–584, 2003.

Harchol-Balter, M. and Downey, A. B. Exploiting process lifetime distributions for dynamic load balancing. *ACM Transactions on Computer Systems (TOCS)*, 15(3):253–285, 1997.

Im, S., Kumar, R., Montazer Qaem, M., and Purohit, M. Non-clairvoyant scheduling with predictions. In *ACM Symposium on Parallelism in Algorithms and Architectures*, 2021.

Im, S., Kumar, R., Petety, A., and Purohit, M. Parsimonious learning-augmented caching. In *International Conference on Machine Learning*, pp. 9588–9601. PMLR, 2022.

Ivanov, D., Dolgui, A., Sokolov, B., Werner, F., and Ivanova, M. A dynamic model and an algorithm for short-term supply chain scheduling in the smart factory industry 4.0. *International Journal of Production Research*, 54(2): 386–402, 2016.

Jensen, E. D., Locke, C. D., and Tokuda, H. A time-driven scheduling model for real-time operating systems. In *Rtss*, volume 85, pp. 112–122, 1985.

Jin, B. and Ma, W. Online bipartite matching with advice: Tight robustness-consistency tradeoffs for the two-stage model. *Advances in Neural Information Processing Systems*, 35:14555–14567, 2022.

Kraska, T., Beutel, A., Chi, E. H., Dean, J., and Polyzotis, N. The case for learned index structures. In *Proceedings of the 2018 international conference on management of data*, pp. 489–504, 2018.

Lassota, A. A., Lindermayr, A., Megow, N., and Schlöter, J. Minimalistic predictions to schedule jobs with online precedence constraints. In *International Conference on Machine Learning*, pp. 18563–18583. PMLR, 2023.

Lattanzi, S., Lavastida, T., Moseley, B., and Vassilvitskii, S. Online scheduling via learned weights. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1859–1877. SIAM, 2020.

Lawler, E. L., Lenstra, J. K., Kan, A. H. R., and Shmoys, D. B. Sequencing and scheduling: Algorithms and complexity. *Handbooks in operations research and management science*, 4:445–522, 1993.

Lenstra, J. K. and Rinnooy Kan, A. Complexity of scheduling under precedence constraints. *Operations Research*, 26(1):22–35, 1978.

Lin, H., Luo, T., and Woodruff, D. Learning augmented binary search trees. In *International Conference on Machine Learning*, pp. 13431–13440. PMLR, 2022.

Lindermayr, A. and Megow, N. Permutation predictions for non-clairvoyant scheduling. In *Proceedings of the 34th ACM Symposium on Parallelism in Algorithms and Architectures*, pp. 357–368, 2022.

Lykouris, T. and Vassilvtiskii, S. Competitive caching with machine learned advice. In *International Conference on Machine Learning*, pp. 3296–3305. PMLR, 2018.

Maghakian, J., Lee, R., Hajiesmaili, M., Li, J., Sitaraman, R., and Liu, Z. Applied online algorithms with heterogeneous predictors. In *International Conference on Machine Learning*, pp. 23484–23497. PMLR, 2023.

Martel, C. Preemptive scheduling with release times, deadlines, and due times. *Journal of the ACM (JACM)*, 29(3): 812–829, 1982.

Merlis, N., Richard, H., Sentenac, F., Odic, C., Molina, M., and Perchet, V. On preemption and learning in stochastic scheduling. In *International Conference on Machine Learning*, pp. 24478–24516. PMLR, 2023.

Mitzenmacher, M. and Vassilvitskii, S. Algorithms with predictions. *Communications of the ACM*, 65(7):33–35, 2022.

Motwani, R., Phillips, S., and Torng, E. Nonclairvoyant scheduling. *Theoretical computer science*, 130(1):17–47, 1994.

Munoz, A. and Vassilvitskii, S. Revenue optimization with approximate bid predictions. *Advances in Neural Information Processing Systems*, 30, 2017.

Panwalkar, S. S. and Iskander, W. A survey of scheduling rules. *Operations research*, 25(1):45–61, 1977.

Pinedo, M. L. *Scheduling*, volume 29. Springer, 2012.

Purohit, M., Svitkina, Z., and Kumar, R. Improving online algorithms via ml predictions. *Advances in Neural Information Processing Systems*, 31, 2018.

Ramamritham, K. and Stankovic, J. A. Scheduling algorithms and operating systems support for real-time systems. *Proceedings of the IEEE*, 82(1):55–67, 1994.

Sadek, K. A. A. and Elias, M. Algorithms for caching and MTS with reduced number of predictions. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=QuIiLSktO4.

Sakaue, S. and Oki, T. Discrete-convex-analysis-based framework for warm-starting algorithms with predictions. *Advances in Neural Information Processing Systems*, 35: 20988–21000, 2022.

Shin, Y., Lee, C., Lee, G., and An, H.-C. Improved learning-augmented algorithms for the multi-option ski rental problem via best-possible competitive analysis. *arXiv preprint arXiv:2302.06832*, 2023.

Silwal, S., Ahmadian, S., Nystrom, A., McCallum, A., Ramachandran, D., and Kazemi, S. M. Kwikbucks: Correlation clustering with cheap-weak and expensive-strong signals. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=p0JSSa1AuV.

Steiger, C., Walder, H., and Platzner, M. Operating systems for reconfigurable embedded platforms: Online scheduling of real-time tasks. *IEEE Transactions on computers*, 53(11):1393–1407, 2004.

Wei, A. and Zhang, F. Optimal robustness-consistency trade-offs for learning-augmented online algorithms. *Advances in Neural Information Processing Systems*, 33: 8042–8053, 2020.

Yao, A. C.-C. Probabilistic computations: Toward a unified measure of complexity. In *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*, pp. 222–227. IEEE Computer Society, 1977.

A. Lower bounds

A.1. Preliminary result

Lemma A.1. *Let F be a probability distribution on $(0, \infty)$ with finite expectation, x an array of n i.i.d. random variables sampled from F, and $\alpha_{n,B} \geq 0$ satisfying for any deterministic algorithm A with access the sizes of B jobs that*

$$\frac{\mathbb{E}_{\sigma,x}[\mathsf{A}(x)]}{\mathbb{E}_x[\mathsf{OPT}(x)]} \geq \alpha_{n,B} \ ,$$

then for any (deterministic or randomized) algorithm ALG, we have

$$\mathsf{R}_{n,B}(\mathsf{ALG}) \geq \alpha_{n,B} \ .$$

Proof. Let F and $\alpha_{n,B}$ be as stated in the lemma. Using Yao's minimax principle (Yao, 1977) we deduce that for any randomized algorithm ALG

$$\begin{aligned}\mathbb{E}_{\sigma,x}[\mathsf{ALG}(x)] &\geq \inf_{\mathsf{A}\text{ deterministic}} \mathbb{E}_{\sigma,x}[\mathsf{A}(x)] \\ &\geq \alpha_{n,B}\mathbb{E}_x[\mathsf{OPT}(x)] \ ,\end{aligned}$$

where the infimum is taken over all deterministic algorithms. The previous inequality can be written as

$$\mathbb{E}_x\big[\mathbb{E}_\sigma[\mathsf{ALG}(x)] - \alpha_{n,B}\mathsf{OPT}(x)\big] \geq 0 \ ,$$

and this implies that, necessarily, there exists a value $x^* = (x_n^*, \dots, x_n^*)$ taken by x verifying $\mathbb{E}_\sigma[\mathsf{ALG}(x^*)] - \alpha_{n,B}\mathsf{OPT}(x^*) \geq 0$, hence

$$\mathsf{R}_{n,B}(\mathsf{ALG}) \geq \frac{\mathbb{E}_\sigma[\mathsf{ALG}(x^*)]}{\mathsf{OPT}(x^*)} \geq \alpha_{n,B} \ .$$

□

A.2. Proof of Theorem 2.1

Let A be a deterministic algorithm. Using Equation 1, it suffices to bound $\mathbb{E}[P_{ij}^{\mathsf{A}}]$ for all $i \neq j$ to deduce a bound on $\mathbb{E}[\mathsf{A}(x)]$, and since P_{ij}^{A} is a non-negative random variable, the focus can be narrowed down to bounding $\Pr(P_{ij}^{\mathsf{A}} > t)$ for all $t > 0$. Following the proof scheme of (Motwani et al., 1994), we introduce the following definition.

Definition A.2. Let A be a deterministic algorithm given an instance of n jobs. For all $i \neq j \in [n]$ and $t \geq 0$, we denote by $u_{i,j}^{\mathsf{A}}(t)$ the time spent on job i when the total time spent on both jobs i and j is t, assuming neither job i nor j is completed. More precisely, $u_{i,j}^{\mathsf{A}}(t)$ is defined by

$$u_{i,j}^{\mathsf{A}}(t) = S_i^{\mathsf{A}}\big(\inf\{t' \geq 0 : S_i^{\mathsf{A}}(t') + S_j^{\mathsf{A}}(t') = t\}\big) \ .$$

$u_{i,j}^{\mathsf{A}}$ is therefore a rule defined solely by the algorithm. For an instance $x = (x_1, \dots, x_n)$ of job sizes, the real time spent on i when a total time of t has been spent on both jobs i, j is given by $\min(u_{i,j}^{\mathsf{A}}(t), x_i)$.

For all $t \geq 0$, the following Lemma allows to express the event $P_{ij}^{\mathsf{A}} > t$ using $u_{ij}^{\mathsf{A}}(t)$.

Lemma A.3. *Let $x_1, \dots, x_n$ be instance of n job sizes, then for any algorithm A, for any $i \neq j \in [n]$ and $t \geq 0$, the following equivalence holds*

$$\big(P_{ij}^{\mathsf{A}} > t\big) \iff \big(x_i > u_{i,j}^{\mathsf{A}}(t) \text{ and } x_j > t - u_{j,i}^{\mathsf{A}}(t)\big)$$

Proof. Let A be an algorithm and $i \neq j \in [n]$. We denote by $S_{ij}(t) = S_i(t) + S_j(t)$ the total processing time spent on both jobs i and j up to time t. Assuming that job i finishes first, i.e. $t_i \leq t_j$, no processing time is spent on job i after t_i, hence $S_i(t_j) = S_i(t_i) = x_i$, and $P_{ij} = S_i(t_j) + S_j(t_i) = S_{ij}(t_i)$. By symmetry, we deduce that $P_{ij} = S_{ij}(\min(t_i, t_j))$.

Therefore, using that S_i and S_j are non-decreasing and continuous, it holds for all $t \geq 0$

$$\begin{aligned} P_{ij} > t &\iff S_{ij}(\min(t_i, t_j)) > t \\ &\iff \min(t_i, t_j) > \inf\{t' : S_{ij}(t') \geq t\} \\ &\iff x_i > S_i(\inf\{t' : S_{ij}(t') \geq t\}) \text{ and } x_j > S_j(\inf\{t' : S_{ij}(t') \geq t\}) \quad (5) \\ &\iff x_i > u_{ij}(t) \text{ and } x_j > t - u_{ij}(t) \ . \quad (6) \end{aligned}$$

Equivalence (5) holds because $t_i = \inf\{t' \geq 0 : S_i(t') \geq x_i\}$, thus for any $s \geq 0$ we have $t > s \iff x_i > S_i(s)$. The same holds for j. For Equivalence (6), we simply used Definition A.2 and the observation $u_{ij}(t) + u_{ji}(t) = t$. □

Remark A.4. In the case of non-clairvoyant algorithms, the rule defining $u_{ij}^{\mathsf{A}}(\cdot)$ is dictated by the algorithm, independent of the job sizes. Thus, if the job sizes are sampled independently from the exponential distribution, Lemma A.3 gives for all $i \neq j$ that $\Pr(P_{ij}^{\mathsf{A}} > t) = \Pr(x_i > u_{i,j}^{\mathsf{A}}(t)) \Pr(x_j > t - u_{j,i}^{\mathsf{A}}(t)) = e^{-t}$, and it follows immediately that $\mathbb{E}[P_{ij}^{\mathsf{A}}] = 1$ for any deterministic algorithm. This argument, used in (Motwani et al., 1994), is not applicable in our context since the algorithm possesses access to certain job sizes, enabling the formulation of a rule for $u_{i,j}^{\mathsf{A}}$ that considers this information. Therefore, more sophisticated techniques become necessary for our analysis since the independence of the events $x_i > u_{i,j}^{\mathsf{A}}(t)$ and $x_j > t - u_{j,i}^{\mathsf{A}}(t)$ is lost.

Lemma A.5. *Let $a \in (0, \infty)$ and $\varphi : [0, \infty) \to [0, \infty)$ a continuously differentiable and increasing function satisfying that $\varphi(0) > 0$, $t \mapsto \frac{\varphi'(t)}{\varphi(t)}$ is non-increasing and $\int_0^\infty \frac{dt}{\varphi(t)^2} < \infty$. Let $x_1^a, \ldots, x_n^a$ i.i.d. random job sizes with distribution*

$$\Pr(x_1^a \leq t) = 1 - \frac{\varphi(t)^{-1} - \varphi(a)^{-1}}{\varphi(0)^{-1} - \varphi(a)^{-1}} \mathbb{1}_{t<a} \ ,$$

then for any algorithm A having access to the sizes of the first B jobs $x_1^a, \ldots, x_B^a$, it holds that

$$\begin{aligned} \mathbb{E}[P_{ij}^{\mathsf{A}}] &\geq \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt - \underset{a\to\infty}{o(1)} && \forall i \neq j \leq B \ , \\ \mathbb{E}[P_{ij}^{\mathsf{A}}] &\geq 2 \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt - \underset{a\to\infty}{o(1)} && \forall i \neq j > B \ , \\ \mathbb{E}[P_{ij}^{\mathsf{A}}] &\geq \varphi(0)^2 \int_0^\infty \left\{ \inf_{T \geq x} G_\varphi(x, T) \right\} \frac{\varphi'(x)}{\varphi(x)} dx - \underset{a\to\infty}{o(1)} && \forall i \leq B, j > B \ , \end{aligned}$$

where $G_\varphi(x, T)$ is defined in Equation (3), and the $\underset{a\to\infty}{o(1)}$ term does not depend on the algorithm A.

Although Lemma A.5 is stated with $a \in (0, \infty)$, the results also hold for random variables $x_1^\infty, \ldots, x_n^\infty$ sampled from the limit distribution $\Pr(x_1^\infty \leq t) = 1 - \frac{\varphi(0)}{\varphi(t)}$, where the $o(1)$ term becomes zero.

Before proving the lemma, let us first observe that, since φ is increasing and $\int_0^\infty \frac{dt}{\varphi(t)^2} < \infty$, then necessarily $\lim_{x\to\infty} \varphi(x) = \infty$.

Proof. Let us first observe that for all $t \geq 0$

$$\Pr(\min(x_1^\infty, x_2^\infty) \geq t) = \Pr(x_1^\infty \geq t) \Pr(x_2^\infty \geq t) = \frac{\varphi(0)^2}{\varphi(t)^2} \ ,$$

hence

$$\mathbb{E}[\min(x_1^\infty, x_2^\infty)] = \int_0^\infty \Pr(\min(x_1^\infty, x_2^\infty) \geq t) dt = \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt \ .$$

In the following, we prove separately the three claims of the Lemma: in the cases where both jobs sizes x_i, x_j are known, both are unknown, and where only x_i is known.

Both job sizes are known For any $i \neq j \in [n]$, it holds that $P_{ij} \geq \min(x_i^a, x_j^a)$, and this true in particular for $i \neq j \in [B]$. Furthermore, for any $t \geq 0$, since $\varphi(t) \geq \varphi(0)$, we have that the mapping $a \mapsto \frac{\varphi(t)^{-1}-\varphi(a)^{-1}}{\varphi(0)^{-1}-\varphi(a)^{-1}}$ is non-increasing. It follows that

$$\begin{aligned}\Pr(x_1^a \leq t) &\leq 1 - \left(\lim_{a' \to \infty} \frac{\varphi(t)^{-1} - \varphi(a')^{-1}}{\varphi(0)^{-1} - \varphi(a')^{-1}}\right) \mathbb{1}_{t<a} \\ &= 1 - \frac{\varphi(0)}{\varphi(t)} \mathbb{1}_{t<a} \ , \end{aligned} \tag{7}$$

and therefore, for any $i \neq j \leq B$

$$\begin{aligned}\mathbb{E}[P_{ij}] &\geq \mathbb{E}[\min(x_1^a, x_2^a)] \\ &= \int_0^\infty \Pr(x_1^a \geq t)^2 dt \\ &\geq \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} \mathbb{1}_{t<a} dt = \int_0^a \frac{\varphi(0)^2}{\varphi(t)^2} dt \\ &= \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt - \underset{a\to\infty}{o(1)} \ . \end{aligned}$$

Both job sizes are unknown For $i \neq j > B$, the algorithm ignores the job sizes x_i^a and x_j^a Therefore, $u_{i,j}$ is independent of x_i^a and x_j^a. Consequently, using Lemma A.3, the independence of x_i^a and x_j^a, then Inequality (7), we obtain

$$\begin{aligned}\Pr(P_{ij} > t) &= \Pr(x_i > u_{i,j}(t) \text{ and } x_j > t - u_{i,j}(t)) \\ &= \Pr(x_i > u_{i,j}(t)) \Pr(x_j > t - u_{i,j}(t)) \\ &\geq \left(\frac{\varphi(0)}{\varphi(u_{i,j}(t))} \mathbb{1}_{u_{i,j}(t)<a}\right) \left(\frac{\varphi(0)}{\varphi(t - u_{i,j}(t))} \mathbb{1}_{t-u_{i,j}(t)<a}\right) \\ &= \frac{\varphi(0)^2 \mathbb{1}_{t-a<u_{i,j}(t)<a}}{\varphi(u_{i,j}(t)\varphi(t - u_{i,j}(t))} \ . \end{aligned}$$

Using that $\frac{\varphi'}{\varphi}$ is non-increasing, we have for any $t, u \geq 0$ that

$$\begin{aligned}\frac{d}{du}\left(\varphi(u)\varphi(t-u)\right) &= \varphi'(u)\varphi(t-u) - \varphi(u)\varphi'(t-u) \\ &= \varphi(u)\varphi(t-u)\left(\frac{\varphi'(u)}{\varphi(u)} - \frac{\varphi'(t-u)}{\varphi(t-u)}\right)\end{aligned}$$

The sign of $\frac{d}{du}\left(\varphi(u)\varphi(t-u)\right)$ is the same as that of $\frac{\varphi'(u)}{\varphi(u)} - \frac{\varphi'(t-u)}{\varphi(t-u)}$, which is a non-increasing function. It is null for $u = t/2$, hence it is non-positive for $u \leq t/2$ and non-negative for $u \geq t/2$. This implies that $u \mapsto \varphi(u)\varphi(t-u)$ is minimal for $u = t/2$, i.e. $\varphi(u)\varphi(t-u) \leq \varphi(\frac{t}{2})^2$ for all $t, u \geq 0$, in particular

$$\Pr(P_{ij} > t) \geq \frac{\varphi(0)^2}{\varphi(t/2)^2} \mathbb{1}_{t-a<u_{i,j}(t)<a} \ ,$$

and it follows that

$$\mathbb{E}[P_{ij}] = \int_0^\infty \Pr(P_{ij} > t) dt \geq \int_0^\infty \frac{\varphi(0)^2}{\varphi(\frac{t}{2})^2} \mathbb{1}_{t-a<u_{i,j}(t)<a} dt \ .$$

Observing that, for any $t \geq 0$, the mapping $a \mapsto \varphi(0)^2/\varphi(\frac{t}{2})^2 \mathbb{1}_{t-a<u_{i,j}(t)<a}$ is non-decreasing, and its limit is $\varphi(0)^2/\varphi(\frac{t}{2})^2$, the monotone convergence theorem guarantees that

$$\begin{aligned}\lim_{a\to\infty} \int_0^\infty \frac{\varphi(0)^2}{\varphi(\frac{t}{2})^2} \mathbb{1}_{t-a<u_{i,j}(t)<a} dt &= \int_0^\infty \frac{\varphi(0)^2}{\varphi(\frac{t}{2})^2} dt \\ &= 2\int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt \ ,\end{aligned}$$

therefore, for any $i \neq j > B$

$$\mathbb{E}[P_{ij}] \geq 2 \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt - \underset{a\to\infty}{o(1)} \ .$$

Only the size of one job is known For $i \leq B$ and $j > B$, the algorithm knows x_i^a, therefore $u_{ij}(\cdot)$ can be a function of x_i^a. By Lemma A.3 then Inequality (7), it holds for all $t \geq 0$ that

$$\begin{aligned}
\Pr(P_{ij} > t) &= \Pr(x_i^a > u_{i,j}(t) \text{ and } x_j^a > t - u_{i,j}(t)) \\
&= \mathbb{E}[\mathbb{1}_{x_i^a > u_{i,j}(t)} \mathbb{1}_{x_j^a > t - u_{i,j}(t)}] \\
&= \mathbb{E}\big[\mathbb{1}_{x_i^a > u_{i,j}(t)} \mathbb{E}[\mathbb{1}_{x_j^a > t - u_{i,j}(t)} \mid x_i^a, u_{i,j}(t)]\big] \\
&= \mathbb{E}\big[\mathbb{1}_{x_i^a > u_{i,j}(t)} \Pr\left(x_j^a > t - u_{i,j}(t) \mid u_{i,j}(t)\right)\big] \\
&\geq \mathbb{E}\left[\mathbb{1}_{x_i^a > u_{i,j}(t)} \frac{\varphi(0)\mathbb{1}_{t - u_{i,j}(t) < a}}{\varphi(t - u_{i,j}(t))}\right] \\
&\geq \varphi(0)\mathbb{E}\left[\frac{\mathbb{1}_{t - a < u_{i,j}(t) < x_i^a}}{\varphi(t - u_{i,j}(t))}\right] \ ,
\end{aligned}$$

and we obtain by integrating over t and then using Tonelli's theorem that

$$\begin{aligned}
\mathbb{E}[P_{ij}] &= \int_0^\infty \Pr(P_{ij} > t) dt \\
&\geq \varphi(0)\mathbb{E}\left[\int_0^\infty \frac{\mathbb{1}_{t - a < u_{i,j}(t) < x_i^a}}{\varphi(t - u_{i,j}(t))} dt\right] \ .
\end{aligned}$$

Recall that $u_{i,j}(t)$ is defined as the time spent on job i when a total of t units of time have been spent on both jobs i and j, thus $u_{ij}(0) = 0$, u_{ij} is non-decreasing and it is 1-Lipschitz. Let us denote by $\mathcal{V}$ the set of functions on $[0, \infty)$ satisfying these properties

$$\mathcal{V} = \left\{ v : [0,\infty) \to [0,\infty) \ : \ v(0) = 0 \text{ and } \tfrac{v(t_2) - v(t_1)}{t_2 - t_1} \in [0,1] \ \forall t_1 < t_2 \right\} ,$$

and for all $x \geq 0$ and $T \in [0, \infty]$, we denote by $\mathcal{V}_{x,T}$ the subset of $\mathcal{V}$ defined as

$$\mathcal{V}_{x,T} = \left\{ v \in \mathcal{V} : \inf\{t \geq 0 : v(t) \geq x\} = T \right\} .$$

For any $x \geq 0$ and $v \in \mathcal{V}$, let $T_v = \inf\{t \geq 0 : v(t) \geq x\}$, then $v \in \mathcal{V}_{x,T_v}$. Since v is 1-Lipschitz and $v(0) = 0$, it holds that $T_v \geq x$. If $v(t) < x$ for all $t \geq 0$ then $T_v = \infty$. Therefore, we have for any $x \geq 0$

$$\mathcal{V} = \bigcup_{T \in [x,\infty]} \mathcal{V}_{x,T} \ .$$

Furthermore, given that v is non-decreasing, $v(t) < x$ if and only if $t < T_v$. Consequently, $\mathbb{E}[P_{ij}]$ satisfies

$$\begin{aligned}
\mathbb{E}[P_{ij}] &\geq \varphi(0)\mathbb{E}\left[\inf_{v \in \mathcal{V}} \int_0^\infty \frac{\mathbb{1}_{t - a < v(t) < x_i^a}}{\varphi(t - v(t))} dt\right] \\
&= \varphi(0)\mathbb{E}\left[\inf_{T \in [x_i^a, \infty]} \inf_{v \in \mathcal{V}_{x_i^a, T}} \int_0^\infty \frac{\mathbb{1}_{t - a < v(t) < x_i^a}}{\varphi(t - v(t))} dt\right] \\
&= \varphi(0)\mathbb{E}\left[\inf_{T \in [x_i^a, \infty]} \inf_{v \in \mathcal{V}_{x_i^a, T}} \int_0^T \frac{\mathbb{1}_{t - a < v(t)}}{\varphi(t - v(t))} dt\right] \ . \qquad (8)
\end{aligned}$$

Let $x > 0$ and $T \in [x, \infty]$, and define $v^*_{x,T} : t \geq 0 \mapsto (t - T + x)\mathbb{1}_{t > T - x}$, then for all $v \in \mathcal{V}_{x,T}$ we have $v(t) \geq v^*_{x,T}(t)$ for all $t \in [0, T]$. Indeed, $v(t) \geq 0 = v^*_{x,T}(t)$ for $t \in [0, T - x]$, and if $v(t) < v^*_{x,T}(t)$ for some $t \in [T - x, T]$ then, because v is 1-Lipschitz, we have

$$v(T) \leq v(T - x) + x < v^*_{x,T}(T - x) + x = x \ ,$$

which contradicts $v(T) \geq x$. Finally, as φ and $v \mapsto \mathbb{1}_{t-a<v}$ are both non-decreasing, it holds for any $v \in \mathcal{V}_{x,T}$ that

$$\begin{aligned}
\int_0^T \frac{\mathbb{1}_{t-a<v(t)}}{\varphi(t-v(t))}dt &\geq \int_0^T \frac{\mathbb{1}_{t-a<v^*_{x,T}(t)}}{\varphi(t-v^*_{x,T}(t))}dt \\
&= \int_0^{T-x} \frac{\mathbb{1}_{t-a<0}}{\varphi(t)}dt + \int_{T-x}^T \frac{\mathbb{1}_{t-a<t-T+x}}{\varphi(T-x)}dt \\
&= \int_0^{\min(a,T-x)} \frac{dt}{\varphi(t)} + \frac{x\mathbb{1}_{T-x<a}}{\varphi(T-x)} \\
&= \begin{cases} \int_0^{T-x} \frac{dt}{\varphi(t)} + \frac{x}{\varphi(T-x)} & \text{if } T-x<a \\ \int_0^a \frac{dt}{\varphi(t)} & \text{if } T-x \geq a \end{cases} \qquad (9) \\
&= \begin{cases} G_\varphi(x,T) & \text{if } T-x<a \\ \int_0^a \frac{dt}{\varphi(t)} & \text{if } T-x \geq a \end{cases} . \qquad (10)
\end{aligned}$$

Taking the infimum over $v \in \mathcal{V}_{x,T}$ then over $T \in [x,\infty]$ in (10) gives for any $x \geq 0$ that

$$\begin{aligned}
\inf_{T\in[x,\infty]} \inf_{v\in\mathcal{V}_{x,T}} \int_0^T \frac{\mathbb{1}_{t-a<v(t)}}{\varphi(t-v(t))}dt &\geq \min\left(\inf_{T\in[x,a+x)} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) \\
&\geq \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) ,
\end{aligned}$$

and substituting into 8 leads to

$$\mathbb{E}[P_{ij}] \geq \varphi(0)\mathbb{E}\left[\min\left(\inf_{T\in[x_1^a,\infty]} G_\varphi(x_1^a,T), \int_0^a \frac{dt}{\varphi(t)}\right)\right] .$$

Let us denote by f_a and f_∞ respectively the density functions of x_1^a and x_1^∞. We have for any $x > 0$ that

$$f_a(x) = \frac{d}{dx}\Pr(x_1^a \leq x) = \frac{\varphi'(x)/\varphi(x)^2}{\varphi(0)^{-1}-\varphi(a)^{-1}}\mathbb{1}_{x<a} \geq \frac{\varphi(0)\varphi'(x)}{\varphi(x)^2}\mathbb{1}_{x<a} = f_\infty(x)\mathbb{1}_{x<a} , \qquad (11)$$

therefore

$$\begin{aligned}
\mathbb{E}[P_{ij}] &= \varphi(0)\int_0^\infty \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) f_a(x)dx \\
&\geq \varphi(0)\int_0^\infty \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) f_\infty(x)\mathbb{1}_{x<a}dx . \qquad (12)
\end{aligned}$$

Observing for all $x > 0$ that

$$\inf_{T\in[x,\infty]} G_\varphi(x,T) \leq \lim_{T\to\infty} G_\varphi(x,T) = \int_0^\infty \frac{dt}{\varphi(t)} \qquad (13)$$

and that the mapping $a \mapsto \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) f_\infty(x)\mathbb{1}_{x<a}$ is non-decreasing, the monotone convergence theorem, the continuity of the minimum, then Inequality 13 guarantee that

$$\begin{aligned}
\lim_{a\to\infty}\int_0^\infty \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) f_\infty(x)\mathbb{1}_{x<a}dx \\
&= \int_0^\infty \lim_{a\to\infty}\left\{\min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right) f_\infty(x)\mathbb{1}_{x<a}\right\}dx \\
&= \int_0^\infty \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^\infty \frac{dt}{\varphi(t)}\right) f_\infty(x)dx \\
&= \int_0^\infty \left(\inf_{T\in[x,\infty]} G_\varphi(x,T)\right) f_\infty(x)dx \\
&= \varphi(0)\int_0^\infty \left(\inf_{T\in[x,\infty]} G_\varphi(x,T)\right) \frac{\varphi'(x)}{\varphi(x)^2}dx .
\end{aligned}$$

Finally, the previous term is finite because

$$\begin{aligned}\int_0^\infty \left(\inf_{T\in[x,\infty]} G_\varphi(x,T)\right)\frac{\varphi'(x)}{\varphi(x)^2}dx &\le \int_0^\infty G_\varphi(x,2x)\frac{\varphi'(x)}{\varphi(x)^2}dx \\ &= \int_0^\infty \left(\int_0^x \frac{dt}{\varphi(t)} + \frac{x}{\varphi(x)}\right)\frac{\varphi'(x)}{\varphi(x)^2}dx \\ &\le \int_0^\infty \left(2\int_0^x \frac{dt}{\varphi(t)}\right)\frac{\varphi'(x)}{\varphi(x)^2}dx \\ &= 2\int_0^\infty \left(\int_t^\infty \frac{\varphi'(x)}{\varphi(t)^2}\right)\frac{dt}{\varphi(t)} \\ &= 2\int_0^\infty \frac{dt}{\varphi(t)^2} < \infty\ .\end{aligned}$$

Therefore

$$\begin{aligned}&\int_0^\infty \min\left(\inf_{T\in[x,\infty]} G_\varphi(x,T), \int_0^a \frac{dt}{\varphi(t)}\right)\frac{\varphi(0)}{\varphi(t)}\mathbb{1}_{t<a}dt \\ &\qquad = \varphi(0)\int_0^\infty \left(\inf_{T\in[x,\infty]} G_\varphi(x,T)\right)\frac{\varphi'(x)}{\varphi(x)^2}dx - o(1)\ ,\end{aligned}$$

and substituting into Inequality (12) gives the aimed result.

□

A.2.1. Proof of the theorem

Using the previous lemmas, we can now prove the theorem.

Proof. Let $x = (x_1, \dots, x_n)$ be an array of i.i.d. random job sizes with distribution $\Pr(x_i \le t) = 1 - \frac{\varphi(0)}{\varphi(t)}$. Observe that

$$\mathbb{E}[x_1] = \int_0^\infty \Pr(x_1 \ge t)dt = \varphi(0)\int_0^\infty \frac{dt}{\varphi(t)} \quad , \quad \mathbb{E}[\min(x_1,x_2)] = \int_0^\infty \Pr(x_1 \ge t)^2 dt = \varphi(0)^2\int_0^\infty \frac{dt}{\varphi(t)^2}\ .$$

Lemma A.5 with $a = \infty$ gives for any algorithm A that $\mathbb{E}[P_{ij}] \ge \mathbb{E}[\min(x_1,x_2)]$ for $i \ne j \le B$, $\mathbb{E}[P_{ij}] \ge 2\mathbb{E}[\min(x_1,x_2)]$ for $i \ne j > B$, and $\mathbb{E}[P_{ij}] \ge \alpha_\varphi \mathbb{E}[\min(x_1,x_2)]$ for $i \le B, j > B$. Indeed, the probability density function of x_1 is $t \mapsto \frac{\varphi(0)\varphi'(t)}{\varphi(t)^2}$, and we have for all $i \le B$ and $j > B$ that

$$\begin{aligned}\varphi(0)\mathbb{E}\big[\inf_{T\ge x_1} G_\varphi(x_1,T)\big] &= \varphi(0)^2\int_0^\infty \left\{\inf_{T\ge x} G_\varphi(x,T)\right\}\frac{\varphi'(x)}{\varphi(x)^2}dx \\ &\ge \alpha_\varphi\varphi(0)^2\int_0^\infty \frac{dt}{\varphi(t)^2} = \alpha_\varphi\mathbb{E}[\min(x_1,x_2)]\ .\end{aligned}$$

Assume that $\int_0^\infty \frac{dt}{\varphi(t)} < \infty$, i.e. $\mathbb{E}[x_1] < \infty$, then by Equation (1) it holds that

$$\begin{aligned}\mathbb{E}[\mathsf{A}(x)] &\ge n\mathbb{E}[x_1] + \sum_{1\le i<j\le B}\mathbb{E}[\min(x_1,x_2)] + \sum_{B<i<j\le n} 2\mathbb{E}[\min(x_1,x_2)] + \sum_{i=1}^{B}\sum_{j=B+1}^{n}\alpha_\varphi\mathbb{E}[\min(x_1,x_2)] \\ &= n\mathbb{E}[x_1] + \left(\frac{B(B-1)}{2} + (n-B)(n-B-1) + \alpha_\varphi B(n-B)\right)\mathbb{E}[\min(x_1,x_2)]\ ,\end{aligned}$$

the terms multiplying $\mathbb{E}[\min(x_1, x_2)]$ can be reordered as follows

$$\begin{aligned}
&\frac{B(B-1)}{2} + (n-B)(n-B-1) + \alpha_\varphi B(n-B) \\
&\quad = \frac{B(B-1)}{2} + (n-B)(n-1+(\alpha_\varphi-1)B) \\
&\quad = \frac{B(B-1)}{2} + n(n-1) + (\alpha_\varphi-1)nB - nB - B((\alpha_\varphi-1)B-1) \\
&\quad = \frac{B(B-1)}{2} + n(n-1) + (\alpha_\varphi-2)nB - (\alpha_\varphi-1)B(B-1) - (\alpha_\varphi-2)B \\
&\quad = n(n-1) - (2-\alpha_\varphi)(n-1)B + (\tfrac{3}{2}-\alpha_\varphi)B(B-1) && (14) \\
&\quad = \frac{n(n-1)}{2}\left(2 - (4-2\alpha_\varphi)\frac{B}{n} + (3-2\alpha_\varphi)\frac{B(B-1)}{n(n-1)}\right) && (15) \\
&\quad = \frac{n(n-1)}{2}\left(2 - \frac{B}{n} - (3-2\alpha_\varphi)\frac{B}{n}\left(1 - \frac{B(B-1)}{n(n-1)}\right)\right) , && (16)
\end{aligned}$$

hence

$$\mathbb{E}[\mathsf{A}(x)] = n\mathbb{E}[x_1] + C_{n,B}\frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)] .$$

On the other hand, by equation (2)

$$\mathbb{E}[\mathsf{OPT}(x)] = \mathbb{E}\left[\sum_{i=1}^{n} x_i + \sum_{i<j}\min(x_i, x_j)\right] = n\mathbb{E}[x_1] + \frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)] ,$$

therefore

$$\begin{aligned}
\frac{\mathbb{E}[\mathsf{A}(x)]}{\mathbb{E}[\mathsf{OPT}(x)]} &\geq \frac{n\mathbb{E}[x_1] + C_{n,B}\frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)]}{n\mathbb{E}[x_1] + \frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)]} \\
&= C_{n,B} - \frac{C_{n,B}-1}{1 + \frac{n-1}{2}\frac{\mathbb{E}[\min(x_1,x_2)]}{\mathbb{E}[x_1]}} ,
\end{aligned}$$

and using Lemma A.1, this yields

$$\mathsf{R}_{n,B}(\mathsf{A}) \geq C_{n,B} - \frac{C_{n,B}-1}{1 + \frac{n-1}{2}\frac{\mathbb{E}[\min(x_1,x_2)]}{\mathbb{E}[x_1]}} .$$

If $B_n = wn + o(n)$ for some $w \in [0, 1]$, then

$$\begin{aligned}
\mathsf{CR}_B(\mathsf{A}) &\geq \liminf_{n\to\infty} \mathsf{R}_{n,B_n}(\mathsf{A}) \\
&\geq 2 - 2(2-\alpha_\varphi)w + (3-2\alpha_\varphi)w^2 .
\end{aligned}$$

Assume now that $\int_0^\infty \frac{dt}{\varphi(t)} = \infty$, thus $\mathbb{E}[x_1] = \infty$. Let $a > 0$, and $x^a = (x_1^a, \ldots, x_n^a)$ be i.i.d. job sizes with distribution

$$\Pr(x_1^a \leq t) = 1 - \frac{\varphi(t)^{-1} - \varphi(a)^{-1}}{\varphi(0)^{-1} - \varphi(a)^{-1}}\mathbb{1}_{t<a} .$$

Therefore x_1^a has a finite expectation $\mathbb{E}[x_1^a] \leq a$. Denoting by x_1, x_2 i.i.d. random variables with distribution $\Pr(x_1 \leq t) = 1 - \frac{\varphi(0)}{\varphi(t)}$, Lemma A.5 and Equation (1) give for any algorithm A that

$$\begin{aligned}
\mathbb{E}[\mathsf{A}(x^a)] &\geq n\mathbb{E}[x_1^a] + \sum_{1\leq i<j\leq B}(\mathbb{E}[\min(x_1, x_2)] - \underset{a\to\infty}{o(1)}) + \sum_{B<i<j\leq n}(2\mathbb{E}[\min(x_1, x_2)] - \underset{a\to\infty}{o(1)}) \\
&\quad + \sum_{i=1}^{B}\sum_{j=B+1}^{n}(\alpha_\varphi\mathbb{E}[\min(x_1, x_2)] - \underset{a\to\infty}{o(1)}) \\
&= n\mathbb{E}[x_1^a] + C_{n,B}\frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)] - n^2\underset{a\to\infty}{o(1)} , && (17)
\end{aligned}$$

where the last equation is obtained with the same computation as in the case $\int_0^\infty \frac{dt}{\varphi(t)} < \infty$. Moreover, if $a \geq 1$ then

$$\mathbb{E}[\min(x_1^a, x_2^a)] = \int_0^\infty \Pr(\min(x_1^a, x_2^a) \geq t)dt = \int_0^\infty \left(\frac{\varphi(t)^{-1} - \varphi(a)^{-1}}{\varphi(0)^{-1} - \varphi(a)^{-1}}\right)^2 \mathbb{1}_{t<a} dt \ ,$$

and we have for any $t \geq 0$ that $\left(\frac{\varphi(t)^{-1}-\varphi(a)^{-1}}{\varphi(0)^{-1}-\varphi(a)^{-1}}\right)^2 \leq \frac{\varphi(t)^{-2}}{(\varphi(0)^{-1}-\varphi(1)^{-1})^2}$, which is an integrable function. Furthermore, φ is increasing and $1/\varphi^2$ is integrable, hence $\lim_{a\to\infty} \varphi(a) = \infty$. Therefore, the dominated convergence theorem gives that

$$\begin{aligned}
\lim_{a\to\infty} \mathbb{E}[\min(x_1^a, x_2^a)] &= \int_0^\infty \lim_{a\to\infty} \left\{ \left(\frac{\varphi(t)^{-1} - \varphi(a)^{-1}}{\varphi(0)^{-1} - \varphi(a)^{-1}}\right)^2 \mathbb{1}_{t<a} \right\} dt \\
&= \int_0^\infty \frac{\varphi(0)^2}{\varphi(t)^2} dt \\
&= \mathbb{E}[\min(x_1, x_2)] \ .
\end{aligned}$$

It follows that

$$\begin{aligned}
\mathsf{OPT}(x^a) &= \mathbb{E}\left[\sum_{i=1}^n x_i^a + \sum_{i<j} \min(x_i^a, x_j^a)\right] \\
&= n\mathbb{E}[x_1^a] + \frac{n(n-1)}{2}\mathbb{E}[\min(x_1^a, x_2^a)] \\
&= n\mathbb{E}[x_1^a] + \frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)] + n^2 \underset{a\to\infty}{o(1)} \ .
\end{aligned}$$

Combining this with Inequality (17), then using Lemma A.1, we obtain that

$$\mathsf{R}_{n,B} \geq \frac{n\mathbb{E}[x_1^a] + C_{n,B}\frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)] - n^2 \underset{a\to\infty}{o(1)}}{n\mathbb{E}[x_1^a] + \frac{n(n-1)}{2}\mathbb{E}[\min(x_1, x_2)] + n^2 \underset{a\to\infty}{o(1)}} \ ,$$

and if $B = wn + o(n)$ then

$$\mathsf{CR}_B(\mathsf{A}) \geq \liminf_{n\to\infty} \mathsf{R}_{n,B}(\mathsf{A}) \geq \frac{\big(2 - 2(2-\alpha_\varphi)w + (3-2\alpha_\varphi)w^2\big)\mathbb{E}[\min(x_1, x_2)] - \underset{a\to\infty}{o(1)}}{\mathbb{E}[\min(x_1, x_2)] + \underset{a\to\infty}{o(1)}}$$

and finally, taking the limit $a \to \infty$ yields

$$\mathsf{CR}_B(\mathsf{A}) \geq 2 - 2(2-\alpha_\varphi)w + (3-2\alpha_\varphi)w^2 \ .$$

□

A.3. Proof of Corollary 2.2

Proof. Let us consider i.i.d. exponentially distributed job sizes. This corresponds to $\varphi(t) = e^t$ for all $t \geq 0$. It holds that $\varphi(0) = 1 > 0$, φ is increasing, $\frac{\varphi'}{\varphi} = 1$ is non-increasing, and

$$\int_0^\infty \frac{dt}{\varphi(t)} = \int_0^\infty e^{-t}dt = 1 \ , \qquad \int_0^\infty \frac{dt}{\varphi(t)^2} = \int_0^\infty e^{-2t}dt = \frac{1}{2} \ .$$

Furthermore, it holds for all $x \geq 0$ and $T \geq x$ that

$$\begin{aligned}
G(x, T) &= \int_0^{T-x} \frac{dt}{\varphi(t)} + \frac{x}{\varphi(T-x)} \\
&= \int_0^{T-x} e^{-t}dt + xe^{-(T-x)} \\
&= 1 + (x-1)e^{-(T-x)} \ .
\end{aligned}$$

Thus, for all $x \geq 0$

$$\inf_{T \geq x} G(x,T) = \begin{cases} G(x,x) = x & \text{if } x < 1 \\ \lim_{T \to \infty} G(x,T) = 1 & \text{if } x \geq 1 \end{cases} ,$$

and it follows that

$$\begin{aligned} \varphi(0)^2 \int_0^\infty \left\{ \inf_{T \geq x} G_\varphi(x,T) \right\} \frac{\varphi'(x)}{\varphi(x)^2} dx &= \int_0^\infty \left\{ \inf_{T \geq x} G_\varphi(x,T) \right\} e^{-x} dx \\ &= \int_0^1 x e^{-x} dx + \int_1^\infty e^{-x} dx \\ &= 1 - \tfrac{1}{e} \\ &= 2(1 - \tfrac{1}{e}) \int_0^\infty \frac{dt}{\varphi(t)^2} . \end{aligned}$$

Finally, by Theorem 2.1, it holds for any algorithm A having access to the sizes of B jobs that

$$\mathsf{R}_{n,B}(\mathsf{A}) \geq C_{n,B} - \frac{C_{n,B} - 1}{1 + \frac{n-1}{4}} = C_{n,B} - \frac{4(C_{n,B} - 1)}{n+3} ,$$

with

$$C_{n,B} = 2 - (4/e)\frac{B}{n} + (4/e - 1)\frac{B(B-1)}{n(n-1)} .$$

In particular, if $B = wn + o(n)$, then again by Theorem 2.1 we have

$$\mathsf{CR}(\mathsf{A}) \geq 2 - \tfrac{4}{e} w + (\tfrac{4}{e} - 1) w^2 .$$

□

A.4. Proof of Corollary 2.3

Proof. Let $r \in (\frac{1}{2}, 1)$, and consider $\varphi : t \mapsto (1+t)^r$. It holds that $\varphi(0) = 1 > 0$, φ is increasing, $\frac{\varphi'}{\varphi} : t \mapsto \frac{r}{1+t}$ is non-increasing, and $\int_0^\infty \frac{dt}{\varphi(t)^2} = \int_0^\infty \frac{dt}{(1+t)^{2r}} = \frac{1}{2r-1} < \infty$ because $r > \frac{1}{2}$. Moreover, it holds for all $x \geq 0$ and $T \geq x$ that

$$\begin{aligned} G(x,T) &= \int_0^{T-x} \frac{dt}{\varphi(t)} + \frac{x}{\varphi(T-x)} \\ &= \int_0^{T-x} \frac{dt}{(1+t)^r} + \frac{x}{(1+T-x)^r} \\ &= -\frac{1}{1-r} + \frac{(1+T-x)^{1-r}}{1-r} + \frac{x}{(1+T-x)^r} , \end{aligned}$$

therefore, for all $x \geq 0$

$$\begin{aligned} \inf_{T \geq x} G(x,T) &= -\frac{1}{1-r} + \inf_{T \geq x} \left\{ \frac{(1+T-x)^{1-r}}{1-r} + \frac{x}{(1+T-x)^r} \right\} \\ &= -\frac{1}{1-r} + \inf_{y \in [0,1]} \left\{ \frac{y^{-(1-r)}}{1-r} + xy^r \right\} , \end{aligned}$$

where the last inequality is obtained by considering $y = \frac{1}{1+T-x}$. It holds that

$$\begin{aligned} \frac{d}{dy} \left(\frac{y^{-(1-r)}}{1-r} + xy^r \right) &= -y^{-(2-r)} + rxy^{-(1-r)} \\ &= rxy^{-(2-r)} \left(y - \frac{1}{rx} \right) , \end{aligned}$$

The mapping $y \mapsto \frac{y^{-(1-r)}}{1-r} + xy^r$ is thus minimized on $[0, \infty)$ for $y = \frac{1}{rx}$, and it is minimized on $[0, 1]$ for $y_* = \min(1, \frac{1}{rx})$. Therefore, it holds for $x \leq \frac{1}{r}$ that $y_* = 1$ and $\inf_{T \geq x} G(x, T) = -\frac{1}{1-r} + (\frac{1}{1-r} + x) = x$, and for $x > \frac{1}{r}$ we have $y_* = \frac{1}{rx}$ and

$$\begin{aligned}
\inf_{T \geq x} G(x, T) &= -\frac{1}{1-r} + \frac{(rx)^{1-r}}{1-r} + (rx)^{-r} \\
&= -\frac{1}{1-r} + \left(\frac{1}{1-r} + \frac{1}{r}\right)(rx)^{1-r} \\
&= -\frac{1}{1-r} + \frac{(rx)^{1-r}}{r(1-r)} \ .
\end{aligned}$$

We deduce that

$$\begin{aligned}
\varphi(0)^2 \int_0^\infty \left\{\inf_{T \geq x} G_\varphi(x, T)\right\} \frac{\varphi'(x)}{\varphi(x)^2} dx &= \int_0^{1/r} \left(-\frac{1}{1-r} + x\right) \frac{r}{(1+x)^{r+1}} dx \\
&\quad + \int_{1/r}^\infty \left(-\frac{1}{1-r} + \frac{(rx)^{1-r}}{r(1-r)}\right) \frac{r}{(1+x)^{r+1}} dx \\
&= -\frac{1}{1-r} + \int_0^{1/r} \frac{rx}{(1+x)^{1+r}} dx + \frac{r^{1-r}}{1-r} \int_{1/r}^\infty \frac{x^{1-r}}{(1+x)^{r+1}} dx \\
&\geq -\frac{1}{1-r} + \frac{r^{1-r}}{1-r} \int_{1/r}^\infty \frac{x^{1-r}}{(1+x)^{r+1}} dx \ . \qquad (18)
\end{aligned}$$

Let us denote $W_r = \int_{1/r}^\infty \frac{x^{1-r}}{(1+x)^{r+1}} dx$. It holds that

$$\begin{aligned}
W_r &= \int_{1/r}^\infty \frac{x^{1-r}}{(1+x)^{r+1}} dx \\
&= \int_{1/r}^\infty \left(\frac{x}{1+x}\right)^{1-r} \frac{dx}{(1+x)^{2r}} \\
&\geq \int_2^\infty \sqrt{\frac{x}{1+x}} \cdot \frac{dx}{(1+x)^{2r}} && (r > \tfrac{1}{2}) \\
&\geq \int_0^{1/3} \frac{\sqrt{1-s}}{s^{2-2r}} ds && (s \leftarrow \tfrac{1}{1+x}) \\
&= \int_0^{1/3} \sqrt{1-s} \cdot \frac{d}{ds}\left(\frac{s^{2r-1}}{2r-1}\right) ds \\
&= \left[\frac{s^{2r-1}\sqrt{1-s}}{2r-1}\right]_0^{1/3} + \frac{1}{2(2r-1)} \int_0^{1/3} \frac{s^{2r-1}}{\sqrt{1-s}} ds \\
&= \frac{3^{-(2r-1)}\sqrt{2/3}}{2r-1} + \frac{1}{2(2r-1)} \int_0^{1/3} \frac{s^{2r-1}}{\sqrt{1-s}} ds \ .
\end{aligned}$$

It holds for all $r \in (\frac{1}{2}, 1)$ that $\frac{s^{2r-1}}{\sqrt{1-s}} \leq \frac{1}{\sqrt{1-s}}$, which is an integrable function on $[0, \frac{1}{3}]$, hence the dominated convergence theorem guarantees that

$$\begin{aligned}
\lim_{r \to \frac{1}{2}} \frac{1}{2} \int_0^{1/3} \frac{s^{2r-1}}{\sqrt{1-s}} ds &= \frac{1}{2} \int_0^{1/3} \lim_{r \to \frac{1}{2}} \frac{s^{2r-1}}{\sqrt{1-s}} ds \\
&= \frac{1}{2} \int_0^{1/3} \frac{ds}{\sqrt{1-s}} \\
&= \left[-\sqrt{1-s}\right]_0^{1/3} \\
&= 1 - \sqrt{2/3} \ ,
\end{aligned}$$

we deduce that

$$\begin{aligned}(2r-1)W_r &= 3^{-(2r-1)}\sqrt{2/3} + \frac{1}{2}\int_0^{1/3} \frac{s^{2r-1}}{\sqrt{1-s}}ds \\ &= \left(\sqrt{2/3} - \underset{r\to 1/2}{o(1)}\right) + \left(1 - \sqrt{2/3} - \underset{r\to 1/2}{o(1)}\right) \\ &= 1 + \underset{r\to 1/2}{o(1)} \ .\end{aligned}$$

Substituting into Inequality 18, and recalling that $\int_0^\infty \frac{dt}{\varphi(t)^2} = \frac{1}{2r-1}$, we deduce that for r close to $\frac{1}{2}$

$$\begin{aligned}\varphi(0)^2 \int_0^\infty \left\{\inf_{T\geq x} G_\varphi(x,T)\right\} \frac{\varphi'(x)}{\varphi(x)^2}dx &\geq -\frac{1}{1-r} + \frac{r^{1-r}}{1-r}W_r \\ &= \left(-\frac{2r-1}{1-r} + \left(\frac{r^{1-r}}{1-r}\right)(2r-1)W_r\right)\int_0^\infty \frac{dt}{\varphi(t)^2} \\ &= \Big(o(1) + (\sqrt{2}+o(1))(1-o(1))\Big)\int_0^\infty \frac{dt}{\varphi(t)^2} \\ &= \Big(\sqrt{2}+o(1)\Big)\int_0^\infty \frac{dt}{\varphi(t)^2} \ .\end{aligned}$$

Consequently, by Theorem 2.1, if $B = wn + o(n)$, it holds for any algorithm A that

$$\mathsf{CR}(\mathsf{A}) \geq 2 - 2\left(2-\sqrt{2} - \underset{r\to 1/2}{o(1)}\right)w + \left(3 - 2\sqrt{2} - \underset{r\to 1/2}{o(1)}\right)w^2 \ ,$$

and taking the limit when $r \to \frac{1}{2}$ gives

$$\mathsf{CR}(\mathsf{A}) \geq 2 - 2(2-\sqrt{2})w + (3-2\sqrt{2})w^2 \ .$$

□

B. Known partial order

B.1. Preliminary results

Lemma B.1. *For any real numbers $x_1, \ldots, x_n$, if σ is a uniformly random permutation of $[n]$ then*

$$\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] = \frac{2}{n(n-1)} \sum_{1\leq i<j\leq n} \min(x_i, x_j) \ .$$

Proof. Since σ is uniformly random, we have for any $i \neq j$ that $\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] = \mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma()})]$, thus

$$\begin{aligned}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] &= \frac{2}{n(n-1)}\mathbb{E}\Big[\sum_{1\leq i<j\leq n} \min(x_{\sigma(i)}, x_{\sigma(j)})\Big] \\ &= \frac{2}{n(n-1)} \sum_{1\leq i<j\leq n} \min(x_i, x_j) \ .\end{aligned}$$

□

Lemma B.2. *For any real numbers $x_1, \ldots, x_n$, if σ is a uniformly random permutation of $[n]$ then*

$$\mathbb{E}[x_{\sigma(1)}\mathbb{1}_{x_{\sigma(1)}<x_{\sigma(2)}}] \leq \mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] = \frac{1}{n(n-1)} \sum_{1\leq i<j\leq n} \min(x_i, x_j) \ ,$$

with equality if $x_1, \ldots, x_n$ are pairwise distinct.

Proof. σ is a uniformly random permutation of $[n]$, therefore $\mathbb{E}[x_{\sigma(i)}\mathbb{1}_{x_{\sigma(i)}<x_{\sigma(j)}}] = \mathbb{E}[x_{\sigma(1)}\mathbb{1}_{x_{\sigma(1)}<x_{\sigma(2)}}]$ for all $i \neq j \in [n]$. It follows that

$$\begin{aligned}\mathbb{E}[x_{\sigma(1)}\mathbb{1}_{x_{\sigma(1)}<x_{\sigma(2)}}] &= \frac{1}{2}\left(\mathbb{E}[x_{\sigma(1)}\mathbb{1}_{x_{\sigma(1)}<x_{\sigma(2)}}] + \mathbb{E}[x_{\sigma(2)}\mathbb{1}_{x_{\sigma(2)}<x_{\sigma(1)}}]\right)\\ &\leq \frac{1}{2}\mathbb{E}[x_{\sigma(i)}\mathbb{1}_{x_{\sigma(1)}<x_{\sigma(2)}} + x_{\sigma(2)}\mathbb{1}_{x_{\sigma(2)}\leq x_{\sigma(1)}}]\\ &= \frac{1}{2}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})]\ ,\end{aligned}$$

and using Lemma B.1 concludes the proof. To have equality, it must hold that $\mathbb{1}_{x_{\sigma(2)}<x_{\sigma(1)}} = \mathbb{1}_{x_{\sigma(2)}\leq x_{\sigma(1)}}$ for any permutation, and this is true if and only if the values $x_1, \ldots, x_n$ are pairwise distinct. □

B.2. Proof of Theorem 3.1

Proof. For any $i \neq j \geq B+1$, as in round-robin, $P_{\sigma(i)\sigma(j)} = 2\min(x_{\sigma(i)}, x_{\sigma(j)})$. For $i \neq j \leq B$, since the jobs $\{x_{\sigma(k)}\}_{k=1}^{B}$ are run in non-decreasing order one after the other, it holds that $P_{\sigma(i)\sigma(j)} = \min(x_{\sigma(i)}, x_{\sigma(j)})$. Finally, for $i \leq B$ and $j \geq B+1$, the delay caused by $\sigma(j)$ to $\pi(i)$ is always $D_{\sigma(j)\pi(i)} = \min(x_{\pi(i)}, x_{\sigma(j)})$. On the other hand if $x_{\sigma(j)} \leq x_{\pi(i-1)}$ then $x_{\sigma(j)}$ terminates before phase i begins, thus job $\pi(i)$ does not delay job $\sigma(j)$: $D_{\pi(i)\sigma(j)} = 0$. If $x_{\sigma(j)} > x_{\pi(i-1)}$, then after the first step of phase i, the time spent on each of the jobs $\pi(i)$ and $\sigma(j)$ is $x_{\pi(i-1)}$. Both jobs are then executed with identical processing powers until one of them is terminated, thus

$$\begin{aligned}D_{\pi(i)\sigma(j)} &= \min(x_{\pi(i)}, x_{\sigma(j)})\mathbb{1}_{x_{\sigma(j)}>x_{\pi(i-1)}}\\ &= x_{\pi(i)}\mathbb{1}_{x_{\sigma(j)}>x_{\pi(i)}} + x_{\sigma(j)}\mathbb{1}_{x_{\pi(i)}\geq x_{\sigma(j)}>x_{\pi(i-1)}}\ .\end{aligned}$$

With the convention $x_{\pi(0)=0}$, taking the sum over i gives

$$\begin{aligned}\sum_{i=1}^{B} D_{\pi(i)\sigma(j)} &= \sum_{i=1}^{B} x_{\pi(i)}\mathbb{1}_{x_{\sigma(j)}>x_{\pi(i)}} + x_{\sigma(j)}\sum_{i=1}^{B}\mathbb{1}_{x_{\pi(i)}\geq x_{\sigma(j)}>x_{\pi(i-1)}}\\ &= \sum_{i=1}^{B} x_{\sigma(i)}\mathbb{1}_{x_{\sigma(j)}>x_{\sigma(i)}} + x_{\sigma(j)}\mathbb{1}_{x_{\sigma(j)}\leq x_{\pi(B)}}\\ &\leq \sum_{i=1}^{B} x_{\sigma(i)}\mathbb{1}_{x_{\sigma(j)}>x_{\sigma(i)}} + x_{\sigma(j)}\ .\end{aligned} \tag{19}$$

Using Lemma B.2, and recalling that $\{\pi(i)\}_{i=1}^{B} = \{\sigma(i)\}_{i=1}^{B}$, we have in expectation,

$$\begin{aligned}\mathbb{E}\Big[\sum_{i=1}^{B} D_{\sigma(i)\sigma(j)}\Big] &= \mathbb{E}\Big[\sum_{i=1}^{B} D_{\pi(i)\sigma(j)}\Big]\\ &\leq \frac{B}{2}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \frac{1}{n}\sum_{k=1}^{n} x_k\ ,\end{aligned} \tag{20}$$

and it follows that, for any $j \geq B+1$

$$\begin{aligned}\mathbb{E}\Big[\sum_{i=1}^{B} P_{\sigma(i)\sigma(j)}\Big] &= \mathbb{E}\Big[\sum_{i=1}^{B} D_{\sigma(j)\sigma(i)}\Big] + \mathbb{E}\Big[\sum_{i=1}^{B} D_{\sigma(i)\sigma(j)}\Big]\\ &\leq B\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \frac{B}{2}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \frac{1}{n}\sum_{k=1}^{n} x_k\\ &= \frac{3B}{2}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \frac{1}{n}\sum_{k=1}^{n} x_k\ ,\end{aligned}$$

hence

$$\sum_{j=B+1}^{n} \mathbb{E}\Big[\sum_{i=1}^{B} P_{\sigma(i)\sigma(j)}\Big] \leq \frac{3}{2}B(n-B)\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \left(1 - \frac{B}{n}\right)\sum_{k=1}^{n} x_k \ .$$

Finally, using Equation (1), the previous upper bounds on $P_{\sigma(i)\sigma(j)}$ for all $i \neq j$, then Equation (16) with $3/2$ instead of α_π gives that the objective of CRRR on the instance $x = \{x_1, \ldots, x_n\}$ satisfies in expectation

$$\begin{aligned}
\mathbb{E}[\mathsf{CRRR}(x)] &= \sum_{i=1}^{n} x_i + \mathbb{E}\Big[\sum_{1\leq i<j\leq n} P_{\sigma(i)\sigma(j)}\Big] \\
&= \sum_{i=1}^{n} x_i + \sum_{1\leq i<j\leq B} \mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] + \sum_{B<i<j\leq n} 2\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] \\
&\quad + \sum_{j=B+1}^{n} \mathbb{E}\Big[\sum_{i=1}^{B} P_{\sigma(i)\sigma(j)}\Big] && (21) \\
&\leq \left(2 - \frac{B}{n}\right)\sum_{i=1}^{n} x_i + \left(\frac{B(B-1)}{2} + (n-B)(n-B-1) + \frac{3}{2}B(n-B)\right)\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \\
&= \left(2 - \frac{B}{n}\right)\sum_{i=1}^{n} x_i + \left(2 - \frac{B}{n}\right)\frac{n(n-1)}{2}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \\
&= \left(2 - \frac{B}{n}\right)\sum_{i=1}^{n} x_i + \left(2 - \frac{B}{n}\right)\sum_{1\leq i<j\leq n} \min(x_i, x_j) && (22) \\
&= \left(2 - \frac{B}{n}\right)\mathsf{OPT}(x) \ , && (23)
\end{aligned}$$

where we used B.1 for (22) then Equation (2) for (23). Therefore, the (n, B)-competitive ratio of Algorithm 1 satisfies

$$\mathsf{R}_{n,B}(\mathsf{CRRR}) \leq 2 - \frac{B}{n} \ . \tag{24}$$

For proving the lower bound on $\mathsf{R}_{n,B}$, let $\varepsilon > 0$ and let us consider job sizes $x_i^\varepsilon = 1 + i\varepsilon$ for all $i \in [n]$. Observe that the only inequalities we used in the analysis are Inequalities (19) ($x_{\sigma(j)}\mathbb{1}_{x_{\sigma(j)}\leq x_{\pi(B)}} \leq x_{\sigma(j)}$), and (20) given by Lemma B.2. The second one becomes equality if the job sizes are pairwise distinct, which is satisfied by $\{x_i^\varepsilon\}_{i=1}^n$. As for Inequality (19), since the job sizes x^ε are pairwise distinct and all larger than 1, we can give instead a lower bound on $\mathbb{E}[x_{\sigma(j)}\mathbb{1}_{x_{\sigma(j)}\leq x_{\pi(B)}}]$

for $j > B$ as follows

$$\begin{aligned}
\mathbb{E}[x_{\sigma(j)}\mathbb{1}_{x_{\sigma(j)}\le x_{\pi(B)}}] &= \mathbb{E}[x_{\sigma(j)}\mathbb{1}_{\sigma(j)\le\pi(B)}] \\
&\ge \Pr(\sigma(j) < \pi(B)) = \Pr(\sigma(n) < \max_{i\le B}\sigma(i)) \\
&= \sum_{k=B+1}^{n}\sum_{m=1}^{k-1}\Pr(\sigma(n)=m, \max_{i\le B}\sigma(i)=k) \\
&= \sum_{k=B+1}^{n}\sum_{m=1}^{k-1}\sum_{\ell=1}^{B}\Pr(\sigma(n)=m, \sigma(\ell)=k, \forall i\in[B]\setminus\{\ell\} : \sigma(i)\in[k-1]\setminus\{m\}) \\
&= \sum_{k=B+1}^{n}\sum_{m=1}^{k-1}\sum_{\ell=1}^{B}\frac{1}{n!}\binom{k-2}{B-1}(B-1)!(n-B-1)! \\
&= \sum_{k=B+1}^{n}\sum_{m=1}^{k-1}\frac{B!(n-B-1)!}{n!}\binom{k-2}{B-1} \\
&= \frac{B!(n-B-1)!}{n!}\sum_{k=B}^{n}(k-1)\binom{k-2}{B-1} \\
&= \frac{B\cdot B!(n-B-1)!}{n!}\sum_{k=B}^{n}\binom{k}{B} \\
&= \frac{B\cdot B!(n-B-1)!}{n!}\binom{n}{B+1} \\
&= \frac{B}{B+1}\,.
\end{aligned}$$

Therefore, instead of (19), we have the lower bound

$$\begin{aligned}
\mathbb{E}\Big[\sum_{i=1}^{B}D_{\sigma(i)\sigma(j)}\Big] &= \mathbb{E}\Big[\sum_{i=1}^{B}D_{\pi(i)\sigma(j)}\Big] \\
&= \frac{B}{2}\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \mathbb{E}[x_{\sigma(j)}\mathbb{1}_{x_{\sigma(j)}\le x_{\pi(B)}}] \qquad (25)\\
&\ge \frac{B}{2} + \frac{B}{B+1}\,, \qquad (26)
\end{aligned}$$

and thus, since we proved earlier that $D_{\sigma(i)\sigma(j)} = \min(x^{\varepsilon}_{\sigma(i),x^{\varepsilon}_{\sigma(j)}}$ for all $i \le B$ and $j > B$, it holds that

$$\mathbb{E}\Big[\sum_{i=1}^{B}P_{\sigma(i)\sigma(j)}\Big] \ge \sum_{i=1}^{B}\mathbb{E}[\min(x_{\sigma(i),x_{\sigma(j)}}] + \frac{B}{2} + \frac{B}{B+1} \ge \frac{3B}{2} + \frac{B}{B+1}$$

and using that $x^{\varepsilon}_i \ge 1$ for all $i \in [n]$, we obtain by substituting the previous inequalities into (21) that

$$\begin{aligned}
\mathbb{E}[\mathsf{CRRR}(x^{\varepsilon})] &\ge n + \frac{B(B-1)}{2} + (n-B)(n-B-1) + (n-B)\left(\frac{B}{2}+\frac{B}{B+1}\right) \\
&= \left(n + (n-B)\frac{B}{B+1}\right) + \left(\frac{B(B-1)}{2} + (n-B)(n-B-1) + \frac{3B}{2}(n-B)\right) \\
&= \left(2n - B - \frac{n-B}{B+1}\right) + \left(n(n-1) - \frac{(n-1)B}{2}\right) \\
&= (n+1)\left(n - \frac{B}{2}\right) - \frac{n-B}{B+1}\,.
\end{aligned}$$

On the other hand, Equation (2), along with $x_i^\varepsilon \leq 1 + n\varepsilon$ for all $i \in [n]$, gives

$$\mathsf{OPT}(x^\varepsilon) \leq \sum_{i=1}^{n} (n-i+1)(1+n\varepsilon) = \frac{n(n+1)}{2}(1+n\varepsilon) \ ,$$

and it follows that

$$\begin{aligned}
\mathsf{R}_{n,B}(\mathsf{CRRR}) &\geq \frac{\mathbb{E}[\mathsf{CRRR}(x^\varepsilon)]}{\mathsf{OPT}(x^\varepsilon)} \\
&\geq \frac{(n+1)\left(n - \frac{B}{2}\right) - \frac{n-B}{B+1}}{\frac{n(n+1)}{2}(1+n\varepsilon)} \\
&= \frac{1}{1+n\varepsilon}\left(2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{(n+1)(B+1)}\right) \ .
\end{aligned}$$

Taking the limit $\varepsilon \to 0$ and using Inequality (24), we obtain that

$$2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{(n+1)(B+1)} \leq \mathsf{R}_{n,B}(\mathsf{CRRR}) \leq 2 - \frac{B}{n} \ .$$

Finally, if $B_n = \lfloor wn \rfloor$ for some $w \in [0,1]$ then $B_n \geq wn$, hence $\mathsf{CR}_B(\mathsf{CRRR}) = \sup_{n\geq 2} R_{n,B_n}(\mathsf{CRRR}) \leq 2 - w$, and this bound is reached by the lower bound on R_{n,B_n} for $n \to \infty$, which gives that $\mathsf{CR}_B(\mathsf{CRRR}) = 2 - w$. □

C. Predictions of the job sizes

C.1. Proof of Theorem 4.1

We first compute the mutual delays of the jobs in Switch with any breakpoints.

Lemma C.1. *For any job sizes $x = (x_1, \ldots, x_n)$, and for any permutation σ of $[n]$ and breakpoints $z^\sigma = (z_{\sigma(1)}, \ldots, z_{\sigma(B)})$, the Switch algorithm Switch(z^σ, x) satisfies for all $i \neq j \in [n]$ that*

$$\begin{aligned}
P_{\sigma(i)\sigma(j)} &= x_{\sigma(i)}\mathbb{1}_{z_{\sigma(i)}<z_{\sigma(j)}} + x_{\sigma(j)}\mathbb{1}_{z_{\sigma(i)}>z_{\sigma(j)}} + (\theta_{ij}x_{\sigma(i)} + (1-\theta_{ij})x_{\sigma(j)})\mathbb{1}_{z_{\sigma(i)}=z_{\sigma(j)}} && \text{if } i,j \leq B \ , \\
P_{\sigma(i)\sigma(j)} &= 2\min(x_{\sigma(i)}, x_{\sigma(j)}) && \text{if } i,j > B \ , \\
P_{\sigma(i)\sigma(j)} &= x_{\sigma(i)}\mathbb{1}_{x_{\sigma(j)}>z_{\sigma(i)}} + \min(z_{\sigma(i)}, x_{\sigma(i)}) && \text{if } i \leq B, j > B \ .
\end{aligned}$$

where θ_{ij} is a Bernoulli random variable with parameter $1/2$ independent of σ for all $i \neq j \in [B]$.

Proof. Let us first define the random variables θ_{ij}. For all $i \neq j \in [B]$, if $z_{\sigma(i)} \neq z_{\sigma(j)}$ then let θ_{ij} be an independent Bernoulli random variable with parameter $1/2$, and if $z_{\sigma(i)} = z_{\sigma(j)}$ then let θ_{ij} be the indicator that $z_{\sigma(i)}$ comes before $z_{\sigma(j)}$ with the ordering π. Since π is an ordering of the breakpoints chosen uniformly at random, then θ_{ij} is a Bernoulli random variable with parameter $1/2$ independently of σ.

For $i \neq j \in [B]$, if $z_{\sigma(i)} < z_{\sigma(j)}$ then job $\sigma(i)$ is executed until completion before job $\sigma(j)$ starts being executed, thus $D_{\sigma(i)\sigma(j)} = x_{\sigma(i)}$ and $D_{\sigma(j)\sigma(i)} = 0$ and $P_{\sigma(i)\sigma(j)} = x_{\sigma(i)}$. By symmetry, if $z_{\sigma(i)} > z_{\sigma(j)}$ then $P_{\sigma(i)\sigma(j)} = x_{\sigma(j)}$. In the case where $z_{\sigma(i)} = z_{\sigma(j)}$, since π is chosen uniformly at random among all the permutations of $[B]$ satisfying $z_{\pi(1)} \leq \ldots \leq z_{\pi(B)}$, each of the jobs $\sigma(i), \sigma(j)$ is run until completion before the other one starts with equal probability $1/2$, hence $P_{\sigma(i)\sigma(j)} = \theta_{ij}x_{\sigma(i)} + (1-\theta_{ij})x_{\sigma(j)}$. It follows that

$$P_{\sigma(i)\sigma(j)} = x_{\sigma(i)}\mathbb{1}_{z_{\sigma(i)}<z_{\sigma(j)}} + x_{\sigma(j)}\mathbb{1}_{z_{\sigma(i)}>z_{\sigma(j)}} + (\theta_{ij}x_{\sigma(i)} + (1-\theta_{ij})x_{\sigma(j)})\mathbb{1}_{z_{\sigma(i)}=z_{\sigma(j)}} \ .$$

For $i \neq j > B$, jobs $\sigma(i), \sigma(j)$ are processed symmetrically and the delays they cause to each other are the same as in round-robin. Therefore $D_{\sigma(i)\sigma(j)} = D_{\sigma(j)\sigma(i)} = \min(x_{\sigma(i)}, x_{\sigma(j)})$, and it holds that

$$P_{\sigma(i)\sigma(j)} = 2\min(x_{\sigma(i)}, x_{\sigma(j)}) \ .$$

For $i \leq B$ and $j > B$, at the time when job $\sigma(i)$ starts being executed, it holds by definition of Algorithm 2 that either job $\sigma(j)$ is completed or $S_{\sigma(j)} = z_{\sigma(i)}$, hence the delay caused by job $\sigma(j)$ to job $\sigma(i)$ is $D_{\sigma(j)\sigma(i)} = \min(z_{\sigma(i)}, x_{\sigma(j)})$. On the other hand, job $\sigma(i)$ delays job $\sigma(j)$ if and only if $x_{\sigma(j)} > z_{\sigma(i)}$, and in this case job $\sigma(i)$ runs until completion before job $\sigma(j)$ is completed, hence $D_{\sigma(i)\sigma(j)} = x_{\sigma(i)} \mathbb{1}_{x_{\sigma(j)} > z_{\sigma(i)}}$, and it follows that

$$P_{\sigma(i)\sigma(j)} = x_{\sigma(i)} \mathbb{1}_{x_{\sigma(j)} > z_{\sigma(i)}} + \min(z_{\sigma(i)}, x_{\sigma(i)})$$

□

C.1.1. Proof of the theorem

Using the previous lemma, we now prove Theorem 4.1

Proof. Using Lemma C.1, it holds for all $i \neq j \in [n]$ that $P_{\sigma(i)\sigma(j)} = \min(x_{\sigma(i)}, x_{\sigma(j)})$ if $i, j \leq B$, $P_{\sigma(i)\sigma(j)} = 2\min(x_{\sigma(i)}, x_{\sigma(j)})$ if $i, j > B$, and if $i \leq B$ and $j > B$ then

$$P_{\sigma(i)\sigma(j)} = \min(x_{\sigma(i)}, x_{\sigma(j)}) + x_{\sigma(i)} \mathbb{1}_{x_{\sigma(i)} < x_{\sigma(j)}} \ .$$

Taking the expectation over σ then using Lemma B.2 gives

$$\begin{aligned} \mathbb{E}[x_{\sigma(i)} \mathbb{1}_{x_{\sigma(i)} < x_{\sigma(j)}}] &= \mathbb{E}[x_{\sigma(1)} \mathbb{1}_{x_{\sigma(1)} < x_{\sigma(2)}}] \\ &\leq \frac{1}{2} \mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \ , \end{aligned} \tag{27}$$

hence for any $i \leq B$ and $j > B$

$$\mathbb{E}[P_{\sigma(i)\sigma(j)}] \leq \frac{3}{2} \mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \ .$$

It follows from Equation (1) that for any instance of n job sizes $x = (x_1, \ldots, x_n)$, the switch algorithm 2 with breakpoints $(z_{\sigma(i)})_{i=1}^B = (x_{\sigma(i)})_{i=1}^B$ satisfies

$$\begin{aligned} \mathbb{E}[\mathsf{Switch}(x^\sigma, x)] &\leq \sum_{i=1}^n x_i + \sum_{1 \leq i < j \leq B} \mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] + \sum_{B < i < j \leq n} 2\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] \\ &\quad + \sum_{i=1}^B \sum_{j=B+1}^n \frac{3}{2} \mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] \\ &= \sum_{i=1}^n x_i + \left(\frac{B(B-1)}{2} + (n-B)(n-B-1) + \frac{3}{2} B(n-B) \right) \mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \\ &= \sum_{i=1}^n x_i + \left(2 - \frac{B}{n}\right) \frac{n(n-1)}{2} \mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \\ &= \sum_{i=1}^n x_i + \left(2 - \frac{B}{n}\right) \sum_{1 \leq i < j \leq n} \min(x_i, x_j) \ , \end{aligned} \tag{28}$$

where we used Lemma B.1 for the last equation. We can assume without loss of generality that $x_1 \leq \ldots \leq x_n$, which yields $\sum_{1 \leq i < j \leq n} \min(x_i, x_j) = \sum_{i=1}^n (n-i) x_i$. Finally, Equation (2) gives

$$\begin{aligned} \frac{\mathbb{E}[\mathsf{Switch}(x^\sigma, x)]}{\mathsf{OPT}(x)} &\leq \frac{\sum_{i=1}^n x_i + \left(2 - \frac{B}{n}\right) \sum_{i=1}^n (n-i) x_i}{\sum_{i=1}^n x_i + \sum_{i=1}^n (n-i) x_i} \\ &= 2 - \frac{B}{n} - \frac{(1 - \frac{B}{n}) \sum_{i=1}^n x_i}{\sum_{i=1}^n (n-i+1) x_i} \\ &= 2 - \frac{B}{n} - \frac{(1 - \frac{B}{n})}{\sum_{i=1}^n (n-i+1) q_i} \ , \end{aligned} \tag{29}$$

where $q_i = (\sum_{j=1}^n x_j)^{-1} x_i$ for all $i \in [n]$. The variables $(q_i)_{i=1}^n$ satisfy $\sum_{i=1}^n q_i = 1$, and we can assume without loss of generality that $q_1 \leq \ldots \leq q_n$ (i.e. $x_1 \leq \ldots \leq x_n$). Expression (29) is maximized under these constraints for $q_1 = \ldots = q_n = \frac{1}{n}$. It follows for all job sizes $x_1, \ldots, x_n$ that

$$\frac{\mathbb{E}[\mathsf{Switch}(x^\sigma, x)]}{\mathsf{OPT}(x)} \leq 2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{n+1} ,$$

hence $\mathsf{R}_{n,B}(\mathsf{Switch}) \leq 2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{n+1}$. Let us now prove that this is exactly the competitive ratio of ALG. Observe that the only inequality we used while analyzing $\mathsf{Switch}(x^\sigma, x)$ is Inequality (27), which becomes an equality if the job sizes are pairwise distinct. Let us, therefore, consider job sizes $x_i^{(\varepsilon)} = 1 + i\varepsilon \in [1, 1 + n\varepsilon]$ for some $\varepsilon > 0$. All the inequalities becoming inequalities, it holds in particular that

$$\begin{aligned}
\mathsf{R}_{n,B}(\mathsf{Switch}) \geq \frac{\mathbb{E}[\mathsf{Switch}(x^\sigma, x^{(\varepsilon)})]}{\mathsf{OPT}(x^{(\varepsilon)})} &= 2 - \frac{B}{n} - \frac{(1-\frac{B}{n})\sum_{i=1}^n x_i^{(\varepsilon)}}{\sum_{i=1}^n (n-i+1) x_i^{(\varepsilon)}} \\
&\geq 2 - \frac{B}{n} - \frac{(1-\frac{B}{n})\sum_{i=1}^n (1+n\varepsilon)}{\sum_{i=1}^n (n-i+1)} \\
&= 2 - \frac{B}{n} - (1+n\varepsilon)\frac{2(1-\frac{B}{n})}{n+1} ,
\end{aligned}$$

and taking $\varepsilon \to 0$ yields that

$$\mathsf{R}_{n,B}(\mathsf{Switch}) = 2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})}{n+1} .$$

Let $w \in [0, 1]$. If $B = \lfloor wn \rfloor$, then $B/n \geq w$ and it holds for all $n \geq 2$ that $\mathsf{R}_{n,B}(\mathsf{Switch}) \leq 2 - w$, and this bound is reached for $n \to \infty$, thus $\mathsf{CR}_B(\mathsf{Switch}) = 2 - w$. □

C.2. Proof of Proposition 4.2

Proof. It is shown in (Motwani et al., 1994) that for any instance $x = (x_1, \ldots, x_n)$, the expected sum of completion times resulting from a run of RTC is $\mathbb{E}[\mathsf{RTC}(x)] = \frac{n+1}{2}\sum_{i=1}^n x_i$. Using Equation 28, we deduce that the algorithm ALG that runs RTC with probability $\frac{2(n-B)}{n(n+3)-2B}$ and runs Switch with breakpoints $z^\sigma = x^\sigma$ with the remaining probability satisfies

$$\begin{aligned}
\mathbb{E}[\mathsf{ALG}(x)] &= \frac{2(n-B)}{n(n+3)-2B}\mathsf{RTC}(x) + \left(1 - \frac{2(n-B)}{n(n+3)-2B}\right)\mathbb{E}[\mathsf{Switch}(x^\sigma, x)] \\
&= \frac{2(n-B)}{n(n+3)-2B} \cdot \frac{n+1}{2}\sum_{i=1}^n x_i + \left(1 - \frac{2(n-B)}{n(n+3)-2B}\right)\left(\sum_{i=1}^n x_i + \left(2 - \frac{B}{n}\right)\sum_{1\leq i<j\leq n}\min(x_i, x_j)\right) \\
&= \left(\frac{(n-1)(n-B)}{n(n+3)-2B} + 1\right)\sum_{i=1}^n x_i + \left(1 - \frac{2(n-B)}{n(n+3)-2B}\right)\left(2 - \frac{B}{n}\right)\sum_{1\leq i<j\leq n}\min(x_i, x_j) \\
&= \left(2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})(2-\frac{B}{n})}{n+3-\frac{2B}{n}}\right)\left(\sum_{i=1}^n x_i + \sum_{1\leq i<j\leq n}\min(x_i, x_j)\right) \\
&= \left(2 - \frac{B}{n} - \frac{2(1-\frac{B}{n})(2-\frac{B}{n})}{n+3-\frac{2B}{n}}\right)\mathsf{OPT}(x) .
\end{aligned}$$

□

C.3. Proof of Lemma 4.3

Proof. Let $x_1, \dots, x_n$ Let ξ be a random variable with distribution F. Let $i \neq j \leq B$, by symmetry, we can assume that $x_{\sigma(i)} \leq x_{\sigma(j)}$. Lemma C.1 gives that

$$\begin{aligned}
P_{\sigma(i)\sigma(j)} &= x_{\sigma(i)} \mathbb{1}_{y_{\sigma(i)} < y_{\sigma(j)}} + x_{\sigma(j)} \mathbb{1}_{y_{\sigma(i)} > y_{\sigma(j)}} + \tfrac{x_{\sigma(i)} + x_{\sigma(j)}}{2} \mathbb{1}_{y_{\sigma(i)} = y_{\sigma(j)}} \\
&\leq x_{\sigma(i)} \mathbb{1}_{y_{\sigma(i)} < y_{\sigma(j)}} + x_{\sigma(j)} \mathbb{1}_{y_{\sigma(i)} \geq y_{\sigma(j)}} \\
&= x_{\sigma(i)} + (x_{\sigma(j)} - x_{\sigma(i)}) \mathbb{1}_{y_{\sigma(i)} \geq y_{\sigma(j)}} \\
&\leq x_{\sigma(i)} + (x_{\sigma(j)} - y_{\sigma(j)} + y_{\sigma(i)} - x_{\sigma(i)}) \mathbb{1}_{y_{\sigma(i)} \geq y_{\sigma(j)}} \\
&\leq x_{\sigma(i)} + (\eta_{\sigma(i)} + \eta_{\sigma(j)}) \\
&= \min(x_{\sigma(i)}, x_{\sigma(j)}) + (\eta_{\sigma(i)} + \eta_{\sigma(j)}) \ ,
\end{aligned}$$

and we obtain in expectation that for all $i \neq j \leq B$

$$\mathbb{E}[P_{\sigma(i)\sigma(j)}] \leq \mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] + \frac{2}{B}\mathbb{E}[\eta^\sigma] \ . \tag{30}$$

For $i \neq j > B$, Lemma C.1 gives

$$\mathbb{E}[P_{\sigma(i)\sigma(j)}] = 2\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] \ . \tag{31}$$

For $i \leq B$ and $j > B$, we have again by Lemma C.1 that

$$\mathbb{E}[P_{\sigma(i)\sigma(j)}] = \mathbb{E}[\min(\xi y_{\sigma(i)}, x_{\sigma(j)})] + \mathbb{E}[x_{\sigma(i)} \mathbb{1}_{\xi y_{\sigma(i)} < x_{\sigma(j)}}] \ .$$

Conditional on the permutation σ, i.e. taking the expectation only over ξ, the first term can be bounded as follows

$$\begin{aligned}
\mathbb{E}[\min(\xi y_{\sigma(i)}, x_{\sigma(j)}) \mid \sigma] &\leq \mathbb{E}[\min(\xi x_{\sigma(i)}, x_{\sigma(j)}) + \xi \eta_{\sigma(i)} \mid \sigma] \\
&= \mathbb{E}[x_{\sigma(j)} \mathbb{1}_{\xi x_{\sigma(i)} \geq x_{\sigma(j)}} + \xi x_{\sigma(i)} \mathbb{1}_{\xi x_{\sigma(i)} < x_{\sigma(j)}} \mid \sigma] + \mathbb{E}[\xi]\eta_{\sigma(i)} \\
&= x_{\sigma(j)} \Pr(\xi \geq \tfrac{x_{\sigma(j)}}{x_{\sigma(i)}} \mid \sigma) + x_{\sigma(i)} \mathbb{E}\Big[\xi \mathbb{1}_{\xi < \frac{x_{\sigma(j)}}{x_{\sigma(i)}}} \mid \sigma\Big] + \mathbb{E}[\xi]\eta_{\sigma(i)} \ ,
\end{aligned} \tag{32}$$

and for the second term, the expectation conditional on σ satisfies

$$\begin{aligned}
\mathbb{E}[x_{\sigma(i)} \mathbb{1}_{\xi y_{\sigma(i)} < x_{\sigma(j)}} \mid \sigma] &\leq \mathbb{E}[y_{\sigma(i)} \mathbb{1}_{\xi y_{\sigma(i)} < x_{\sigma(j)}} + \eta_{\sigma(i)} \mid \sigma] \\
&= y_{\sigma(i)} \Pr(\xi < \tfrac{x_{\sigma(j)}}{y_{\sigma(i)}} \mid \sigma) + \eta_{\sigma(i)} \\
&= h_F(x_{\sigma(j)}, y_{\sigma(i)}) + \eta_{\sigma(i)} \\
&\leq h_F(x_{\sigma(j)}, x_{\sigma(i)}) + L_F |x_{\sigma(i)} - y_{\sigma(i)}| + \eta_{\sigma(i)} && (33) \\
&= x_{\sigma(i)} \Pr(\xi < \tfrac{x_{\sigma(j)}}{x_{\sigma(i)}} \mid \sigma) + (1 + L_F)\eta_{\sigma(i)} \ , && (34)
\end{aligned}$$

where we used for (33) that h_F is L-Lipschitz with respect to the second variable. Combining (32) and (34) yields

$$\begin{aligned}
\mathbb{E}[P_{\sigma(i)\sigma(j)} \mid \sigma] &\leq x_{\sigma(j)} + (x_{\sigma(i)} - x_{\sigma(j)}) \Pr(\xi < \tfrac{x_{\sigma(j)}}{x_{\sigma(i)}} \mid \sigma) + x_{\sigma(i)} \mathbb{E}\Big[\xi \mathbb{1}_{\xi < \frac{x_{\sigma(j)}}{x_{\sigma(i)}}} \mid \sigma\Big] + (1 + L_F + \mathbb{E}[\xi])\eta_{\sigma(i)} \\
&= x_{\sigma(j)} + x_{\sigma(i)} \left(\left(1 - \tfrac{x_{\sigma(j)}}{x_{\sigma(i)}}\right) \Pr(\xi < \tfrac{x_{\sigma(j)}}{x_{\sigma(i)}} \mid \sigma) + \mathbb{E}\Big[\xi \mathbb{1}_{\xi < \frac{x_{\sigma(j)}}{x_{\sigma(i)}}} \mid \sigma\Big] \right) + (1 + L_F + \mathbb{E}[\xi])\eta_{\sigma(i)} \\
&= x_{\sigma(j)} + x_{\sigma(i)} g_F\left(\tfrac{x_{\sigma(j)}}{x_{\sigma(i)}}\right) + (1 + L_F + \mathbb{E}[\xi])\eta_{\sigma(i)} \ .
\end{aligned} \tag{35}$$

We can assume without of generality that $x_1 \leq \ldots \leq x_n$. It holds by definition of the constants β_F and γ_F that $g_F\left(\frac{x_{\sigma(j)}}{x_{\sigma(i)}}\right) \leq \beta_F \frac{x_{\sigma(j)}}{x_{\sigma(i)}}$ if $\sigma(j) < \sigma(i)$, and $g_F\left(\frac{x_{\sigma(j)}}{x_{\sigma(i)}}\right) \leq \gamma_F - \frac{x_{\sigma(j)}}{x_{\sigma(i)}}$ if $\sigma(j) > \sigma(i)$, which gives

$$\begin{aligned}
\mathbb{E}\left[x_{\sigma(j)} + x_{\sigma(i)} g_F\left(\tfrac{x_{\sigma(j)}}{x_{\sigma(i)}}\right)\right] &= \frac{1}{2}\mathbb{E}\left[x_{\sigma(j)} + x_{\sigma(i)} g_F\left(\tfrac{x_{\sigma(j)}}{x_{\sigma(i)}}\right) \mid \sigma(j) < \sigma(i)\right] \\
&\quad + \frac{1}{2}\mathbb{E}\left[x_{\sigma(j)} + x_{\sigma(i)} g_F\left(\tfrac{x_{\sigma(j)}}{x_{\sigma(i)}}\right) \mid \sigma(j) > \sigma(i)\right] \\
&\leq \frac{1+\beta_F}{2}\mathbb{E}[x_{\sigma(j)} \mid \sigma(j) < \sigma(i)] + \frac{\gamma_F}{2}\mathbb{E}[x_\sigma(i) \mid \sigma(j) > \sigma(i)] \\
&= \frac{1+\beta_F}{2}\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)}) \mid \sigma(j) < \sigma(i)] + \frac{\gamma_F}{2}\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)}) \mid \sigma(j) > \sigma(i)] \\
&= \frac{1+\beta_F+\gamma_F}{2}\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] \ ,
\end{aligned}$$

thus we obtain by taking the expectation over σ in (35) that

$$\mathbb{E}[P_{\sigma(i)\sigma(j)}] \leq \left(\tfrac{1+\beta_F+\gamma_F}{2}\right)\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] + \left(\tfrac{1+L_F+\mathbb{E}[\xi]}{B}\right)\mathbb{E}[\eta^\sigma] \ .$$

Using Equation 1, the inequality above, Inequalities (30), (31), then Equation 16 with $\frac{1+\beta_F+\gamma_F}{2}$ instead of α_π, we deduce that Algorithm 3 satisfies for any job sizes $x_1, \ldots, x_n$ that

$$\begin{aligned}
\mathbb{E}[\mathsf{Switch}(\xi y^\sigma, x)] &= \sum_{i=1}^n x_i + \sum_{1\leq i<j\leq n} \mathbb{E}[P_{\sigma(i)\sigma(j)}] \\
&\leq \sum_{i=1}^n x_i + \sum_{1\leq i<j\leq B}\left(\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] + \frac{2}{B}\mathbb{E}[\eta^\sigma]\right) + \sum_{B<i<j\leq n} 2\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] \\
&\quad + \sum_{i=1}^B \sum_{j=B+1}^n \left(\left(\tfrac{1+\beta_F+\gamma_F}{2}\right)\mathbb{E}[\min(x_{\sigma(i)}, x_{\sigma(j)})] + \left(\tfrac{1+L_F+\mathbb{E}[\xi]}{B}\right)\mathbb{E}[\eta^\sigma]\right) \\
&= \sum_{i=1}^n x_i + \left(\frac{B(B-1)}{2} + (n-B)(n-B-1) + \left(\tfrac{1+\beta_F+\gamma_F}{2}\right)B(n-B)\right)\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \\
&\quad + \big(B - 1 + (1 + L_F + \mathbb{E}[\xi])(n-B)\big)\mathbb{E}[\eta^\sigma] \\
&= \sum_{i=1}^n x_i + \left(n(n-1) + \left(1 - \tfrac{\beta_F+\gamma_F}{2}\right)B(B-1) - \left(\tfrac{3}{2} - \tfrac{\beta_F+\gamma_F}{2}\right)B(n-1)\right)\mathbb{E}[\min(x_{\sigma(1)}, x_{\sigma(2)})] \\
&\quad + \big(B - 1 + (1 + L_F + \mathbb{E}[\xi])(n-B)\big)\,\mathbb{E}[\eta^\sigma] \\
&= \sum_{i=1}^n x_i + \left(2 - \left(3 - \beta_F - \gamma_F\right)\frac{B}{n} + \left(2 - \beta_F - \gamma_F\right)\frac{B(B-1)}{n(n-1)}\right)\sum_{i<j}\min(x_i, x_j) \\
&\quad + \big(B - 1 + (1 + L_F + \mathbb{E}[\xi])(n-B)\big)\mathbb{E}[\eta^\sigma] \ ,
\end{aligned}$$

where we used Lemma B.1 for the last equality. □

C.4. Proof of Lemma 4.4

Proof. Let $\xi \sim 1 + \mathcal{E}(1/\rho)$. The mapping h_F defined in Lemma 4.3 becomes for all $s, t > 0$

$$h_F(s,t) = tF(\tfrac{s}{t}) = t\left(1 - e^{-\frac{1}{\rho}(\frac{s}{t}-1)}\right)\mathbb{1}_{t<s} \ .$$

For all $s > 0$, it holds for all $t < s$ that

$$\frac{\partial h_F(s,t)}{\partial t} = 1 - e^{\frac{1}{\rho}(\frac{s}{t}-1)} - t \times \frac{s}{\rho t^2} e^{-\frac{1}{\rho}(\frac{s}{t}-1)} = 1 - \left(\frac{s}{\rho t} + 1\right) e^{-\frac{1}{\rho}(\frac{s}{t}-1)} \ ,$$

but the mapping $u \mapsto (\frac{u}{\rho}+1)e^{-\frac{u-1}{\rho}}$ is decreasing on $[1,\infty]$. Indeed

$$\frac{\partial}{\partial u}\left((\tfrac{u}{\rho}+1)e^{-\frac{u-1}{\rho}}\right) = \tfrac{1}{\rho}e^{-\frac{u-1}{\rho}} - \tfrac{1}{\rho}(\tfrac{u}{\rho}+1)e^{-\frac{u-1}{\rho}} = -\tfrac{u}{\rho^2}e^{-\frac{u-1}{\rho}} < 0 \ ,$$

and since $\frac{s}{t} > 1$ we deduce that

$$-\frac{1}{\rho} = 1 - \lim_{u\to 1}(\tfrac{u}{\rho}+1)e^{-\frac{u-1}{\rho}} \leq \frac{\partial h_F(s,t)}{\partial t} \leq 1 - \lim_{u\to\infty}(\tfrac{u}{\rho}+1)e^{-\frac{u-1}{\rho}} = 1 \ ,$$

thus $|\frac{\partial h_F(s,t)}{\partial t}| \leq \max(1,\frac{1}{\rho}) = \frac{1}{\rho}$ for all $t < s$. Otherwise, if $t \geq s$ then $h_F(s,t) = 0$ and $h_F(s,\cdot)$ is continuous in $t = s$, therefore $t \mapsto h_F(s,t)$ is $\frac{1}{\rho}$-Lipschitz.

On the other hand, given that $\xi > 1$ a.s., it holds for $s \leq 1$ that $\mathbb{E}[\xi \mathbb{1}_{\xi<s}] = 0$, and for $s > 1$ that

$$\begin{aligned}
\mathbb{E}[\xi \mathbb{1}_{\xi<s}] &= \Pr(\xi < s) + \mathbb{E}[(\xi-1)\mathbb{1}_{\xi-1<s-1}] \\
&= F(s) + \int_0^\infty t\mathbb{1}_{t<s-1}\frac{e^{-t/\rho}}{\rho}dt \\
&= F(s) + \int_0^{s-1}\frac{te^{-t/\rho}}{\rho}dt \\
&= F(s) + \left[-(t+\rho)e^{-t/\rho}\right]_0^{s-1} \\
&= \left(1 - e^{-(s-1)/\rho}\right) + \left(-(s-1+\rho)e^{-t/\rho} + \rho\right) \\
&= 1 + \rho + (s+\rho)e^{-(s-1)/\rho} \ .
\end{aligned}$$

If follows for all $s > 0$ that

$$\begin{aligned}
g_F(s) &= (1-s)\left(1 - e^{-(s-1)/\rho}\right)\mathbb{1}_{s>1} + \left(1+\rho+(s+\rho)e^{-(s-1)/\rho}\right)\mathbb{1}_{s>1} \\
&= \left((2+\rho) - s - (1+\rho)e^{-(s-1)/\rho}\right)\mathbb{1}_{s>1} \ ,
\end{aligned}$$

hence $\beta_F = \sup_{0<s\leq 1}\frac{g_F(s)}{s} = 0$ and

$$\gamma_F = \sup_{s\geq 1}(g_F(s)+s) = \sup_{s\geq 1}\left((2+\rho) - (1+\rho)e^{-(s-1)/\rho}\right) = 2+\rho \ .$$

Finally, with $L_F = \frac{1}{\rho}$, $\beta_F = 0$ and $\gamma_F = 2+\rho$, and observing that $\mathbb{E}[\xi] = 1+\rho$, the constants $C^1_{n,B,F}$ and $C^1_{n,B,F}$ defined in Lemma 4.3 satisfy

$$\begin{aligned}
C^1_{n,B,F} &= 2 - (1-\rho)\frac{B}{n} - \rho\frac{B(B-1)}{n(n-1)} = 2 - \frac{B}{n} + \rho\frac{B}{n}\left(1 - \frac{B-1}{n-1}\right) \ , \\
C^2_{n,B,F} &\leq (2+\rho+\tfrac{1}{\rho})(n-B) + B \leq \frac{4}{\rho}(n-B) + B \ ,
\end{aligned}$$

and the objective function of Algorithm 3 with any job sizes $x = \{x_1,\ldots,x_n\}$ can be upper-bounded as follows

$$\mathbb{E}[\mathsf{Switch}(\xi y^\sigma, x)] \leq \sum_{i=1}^n x_i + C^1_{n,B,F}\sum_{i<j}\min(x_i,x_j) + \left(\frac{4}{\rho}(n-B)+B\right)\mathbb{E}[\eta^\sigma] \ ,$$

and by Equation (2) we obtain

$$\begin{aligned}
\frac{\mathbb{E}[\mathsf{Switch}(\xi y^\sigma, x)]}{\mathsf{OPT}(x)} &\leq \frac{\sum_{i=1}^n x_i + C^1_{n,B,F}\sum_{i<j}\min(x_i,x_j)}{\sum_{i=1}^n x_i + \sum_{i<j}\min(x_i,x_j)} + \left(\frac{4}{\rho}(n-B)+B\right)\frac{\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} \\
&= C^1_{n,B,F} - \frac{(C^1_{n,B,F}-1)\sum_{i=1}^n x_i}{\sum_{i=1}^n x_i + \sum_{i<j}\min(x_i,x_j)} + \left(\frac{4}{\rho}(n-B)+B\right)\frac{\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} \ .
\end{aligned}$$

As we showed in the proof of Theorem 4.1, we have $\frac{\sum_{i=1}^n x_i}{\sum_{i=1}^n x_i + \sum_{i<j} \min(x_i, x_j)} \geq \frac{2}{n+1}$, and the minimum is reached when all the job sizes are equal.

$$\frac{\mathbb{E}[\mathsf{Switch}(\xi y^\sigma, x)]}{\mathsf{OPT}(x)} \leq C^1_{n,B,F} - \frac{2(C^1_{n,B,F} - 1)}{n+1} + \left(\tfrac{4}{\rho}(1 - \tfrac{B}{n}) + \tfrac{B}{n}\right) \frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} ,$$

which yields after taking the supremum over instances x of n jobs

$$\begin{aligned}
\mathsf{R}_{n,B}(\eta; \mathsf{Switch}) &\leq C^1_{n,B,F} - \frac{2(C^1_{n,B,F} - 1)}{n+1} + \left(\tfrac{4}{\rho}(1 - \tfrac{B}{n}) + \tfrac{B}{n}\right) \frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} \\
&\leq C^1_{n,B,F} + \left(\tfrac{4}{\rho}(1 - \tfrac{B}{n}) + \tfrac{B}{n}\right) \frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} \\
&= \left(2 - \tfrac{B}{n} + \rho\tfrac{B}{n}(1 - \tfrac{B-1}{n-1})\right) + \left(\tfrac{4}{\rho}(1 - \tfrac{B}{n}) + \tfrac{B}{n}\right) \frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)} .
\end{aligned} \tag{36}$$

□

C.5. Proof of Theorem 4.5

Proof. Let $\rho \in (0, 1]$, and $x = (x_1, \ldots, x_n)$ be an instance of job sizes all at least equal to 1. Before starting to run, the algorithm has access to predictions $y_{\sigma(1)}, \ldots, y_{\sigma(B)}$ of $x_{\sigma(1)}, \ldots, x_{\sigma(B)}$, it samples $\xi \sim 1 + \mathcal{E}(1/\rho)$ and sets the breakpoints $z_{\sigma(i)} = \xi y_{\sigma(i)}$ for all $i \in [B]$. With σ and the breakpoints fixed, the preferential algorithm runs Algorithm 2 at rate λ and round-robin at rate $1 - \lambda$. Lemma C.1 and Equation 1 guarantee that, with a fixed permutation σ and fixed breakpoints, Algorithm 2 is monotonic, as all the mutual delays $P_{\sigma(i)\sigma(j)}$ are non-decreasing functions of $x_{\sigma(i)}$ and $x_{\sigma(j)}$. Following the proof of Lemma 3.1 in (Purohit et al., 2018), when running the preferential algorithm, the completion times of Algorithm 2 increase by a factor of $1/\lambda$, while the completion times of round-robin increase by a factor $1/(1 - \lambda)$, and since both algorithms are monotonic, the fact that some jobs are run simultaneously by both of them can only improve the objective function. Denoting $\mathsf{Switch}(\xi y^\sigma, x)$ the objective function of Algorithm 2 with instance $x = (x_i)_{i=1}^n$ and breakpoints $\xi y^\sigma = (\xi y_{\sigma(i)})_{i=1}^B$, and $\mathsf{RR}(x)$ the objective function of round-robin with x, it follows that

$$\mathsf{ALG}_\lambda(x) \leq \min\left(\frac{\mathsf{Switch}(\xi y^\sigma, x)}{\lambda}, \frac{\mathsf{RR}(x)}{1 - \lambda}\right) .$$

Given that $\mathsf{RR}(x)$ is deterministic and 2-competitive, and using Lemma 4.4, we obtain by taking the expectation that

$$\begin{aligned}
\mathbb{E}[\mathsf{ALG}_\lambda(x)] &\leq \mathbb{E}\left[\min\left(\frac{\mathsf{RR}(x)}{1 - \lambda}, \frac{\mathsf{Switch}(\xi y^\sigma, x)}{\lambda}\right)\right] \\
&\leq \min\left(\frac{\mathsf{RR}(x)}{1 - \lambda}, \frac{\mathbb{E}[\mathsf{Switch}(\xi y^\sigma, x)]}{\lambda}\right) \\
&\leq \min\left(\frac{2}{1 - \lambda}, \frac{\left(2 - \frac{B}{n} + \rho\frac{B}{n}(1 - \frac{B-1}{n-1})\right) + \left(\frac{4}{\rho}(1 - \frac{B}{n}) + \frac{B}{n}\right) \frac{n\mathbb{E}[\eta^\sigma]}{\mathsf{OPT}(x)}}{\lambda}\right) \mathsf{OPT}(x),
\end{aligned}$$

which concludes the proof. □