The VCG Mechanism for Bayesian Scheduling*

Yiannis Giannakopoulos† Maria Kyropoulou‡

December 11, 2015

Abstract

We study the problem of scheduling m tasks to n selfish, unrelated machines in order to minimize the makespan, where the execution times are independent random variables, identical across machines. We show that the VCG mechanism, which myopically allocates each task to its best machine, achieves an approximation ratio of $O\left(\frac{\ln n}{\ln \ln n}\right)$. This improves significantly on the previously best known bound of $O\left(\frac{m}{n}\right)$ for prior-independent mechanisms, given by Chawla et al. [STOC'13] under the additional assumption of Monotone Hazard Rate (MHR) distributions. Although we demonstrate that this is in general tight, if we do maintain the MHR assumption, then we get improved, (small) constant bounds for $m \geq n \ln n$ i.i.d. tasks, while we also identify a sufficient condition on the distribution that yields a constant approximation ratio regardless of the number of tasks.

1 Introduction

We consider the problem of scheduling tasks to machines, where the processing times of the tasks are *stochastic* and the machines are *strategic*. The goal is to minimize the expected completion time (a.k.a. *makespan*) of any machine, where the expectation is taken over the randomness of the processing times and the possible randomness of the mechanism. We are interested in the performance, i.e. the expected maximum makespan, of *truthful* mechanisms compared to the optimal mechanism that is not necessarily truthful. This problem, which we call the *Bayesian scheduling* problem, was previously considered by Chawla et al. [7].

Scheduling problems constitute a very rich and intriguing area of research [18]. In one of the most fundamental cases, the goal is to schedule m tasks to n parallel machines while minimizing the makespan, when the processing times of the tasks are selected by an adversary in an arbitrary way and can depend on the machine to which they are allocated. However, the assumption that the machines will blindly follow the instructions of a central authority (scheduler) was eventually challenged, especially due to the rapid growth of the Internet and its use as a primary computing platform. This motivated a mechanism-design approach for the scheduling problem which Nisan and Ronen [26] introduced in their seminal paper: the processing times of the tasks are now private information of the machines, and each machine declares to the mechanism how much time it requires to execute each task. The mechanism then outputs the allocation of tasks to machines, as well as monetary compensations to the machines for their work, based solely on these declarations. In fact, the mechanism has to decide the output in advance, for any possible matrix of processing times the machines can report. Each machine is assumed to be rational and strategic, so, given the mechanism and the

*Supported by ERC Advanced Grant 321171 (ALGAME) and EPSRC grant EP/M008118/1.

A preliminary version of this paper appeared in WINE'15 [16].

†Department of Computer Science, University of Liverpool. Email: ygiannak@liverpool.ac.uk

A significant part of this work was carried out while the first author was a PhD student at the University of Oxford.

‡Department of Computer Science, University of Oxford. Email: kyropoul@cs.ox.ac.uk

true processing times, its declarations are chosen in order to minimize the processing time/cost it has to spend for the execution of the allocated tasks minus the payment it will receive. In this scenario, the goal is to design a *truthful* mechanism that minimizes the makespan; truthful mechanisms define the allocation and payment functions so that the machines don't have an incentive to misreport their true processing-time capabilities. We will refer to this model as the *prior free* scheduling problem, as opposed to the stochastic model we discuss next.

In the Bayesian scheduling problem [7], the time a specific machine requires in order to process a task is drawn from a distribution. We consider one of the fundamental questions posed by the algorithmic mechanism design literature, which is about quantifying the potential performance loss of a mechanism due to the requirement for truthfulness. In the Bayesian scheduling setting, this question translates to: *What is the maximum ratio (for any distribution of processing times) of the expected maximum makespan of the best truthful mechanism over the expected optimal makespan (ignoring the requirement for truthfulness)?*

In this paper we tackle this question by considering a well known and natural truthful mechanism, the *Vickrey-Clarke-Groves mechanism (VCG)*. VCG greedily and myopically allocates each task to a machine that minimizes its processing time [31, 10, 17]. It is a well known fact that it is a truthful mechanism in a very strong sense; truth-telling is a dominant strategy for the machines. Because of the notorious lack of characterization results for truthfulness for restricted domains such as scheduling, VCG is the standard and obvious choice to consider for the Bayesian scheduling problem (or slightly more generally, affine maximizers). We stress here that for the scheduling domain (and for any additive domain) the VCG allocation and payments can be computed in polynomial time. Also, it is important to note that VCG is a *prior-independent* mechanism, i.e. it does not require any knowledge of the prior distribution from which the processing times are drawn.

Prior-independence is a very strong property, and is an important feature for mechanisms used in stochastic settings. Being robust with respect to prior distributions facilitates applicability in real systems, while at the same time bypassing the pessimistic inapproximability of worst case analysis. The idea is that we would like the mechanisms we use, without relying on any knowledge of the distribution of the processing times of the tasks, to still perform well compared to the optimal mechanism that is tailored for the particular distribution.

Chawla et al. [7] were the first to examine the Bayesian scheduling problem while considering the importance for prior-independence. They study the following two mechanisms:

Bounded overload with parameter c Allocate tasks to machines such that the sum of the processing times of all tasks is minimized, subject to placing at most $c\frac{m}{n}$ tasks at any machine.

Sieve and bounded overload with parameters $c, \beta,$ **and** δ Fix a partition of the machines into two sets of sizes $(1-\delta)n$ and δn. Ignoring all processing times which exceed[1] β (i.e. setting them equal to infinity), run VCG on the first set of machines. For the tasks that remain unallocated run the bounded overload mechanism with parameter c on the second set of machines.

The above mechanisms are inspired by maximal-in-range [27] (affine maximizers) and threshold mechanisms, as these are essentially the only non-trivial truthful mechanisms we know for the scheduling domain. One would expect that the simplest of those mechanisms, which is the VCG mechanism, would be the first to be considered. Indeed, VCG is the most natural, truthful, simple, polynomial time computable, and prior-independent mechanism. Still, the authors in [7] design the above mechanisms in an attempt to prevent certain bad behaviour that

[1] Assume you run VCG on the first set of machines plus a dummy machine with processing time β on all tasks. The case where a task has processing time equal to β can be ignored without loss of generality for the case of continuous distributions.

VCG exhibits on a specific input instance and don't examine VCG beyond that point. As we demonstrate in this paper, however, this is the worst case scenario for VCG and we can identify cases where VCG performs considerably better, either by placing a restriction on the number of tasks or by making some additional distributional assumptions.

Our results. We prove an asymptotically tight bound of $\Theta\left(\frac{\ln n}{\ln \ln n}\right)$ for the approximation ratio of VCG for the Bayesian scheduling problem under the sole assumption that the machines are a priori identical. This bound is achieved by showing that the worst case input for VCG is actually one where the tasks are all of unit weight (point mass distributions). This resembles a balls-in-bins type scenario from which the bound is implied.

Whenever the processing times of the tasks are i.i.d. and drawn from an MHR continuous distribution, VCG is shown to be $2\left(1+\frac{n\ln n}{m}\right)$-approximate for the Bayesian scheduling problem. This immediately implies a constant bound at most equal to 4 when $m \geq n\ln n$. We also get an improved bound of $1+\sqrt{2}$ when $m \geq n^2$ using a different approach. For the complementary case of $m \leq n\ln n$, we identify a property of the distribution of processing times such that VCG again achieves a constant approximation. We observe that important representatives of the class of MHR distributions, that is the uniform distribution on $[0,1]$ as well as exponential distributions, do satisfy this property, so for these distributions VCG is 4-approximate regardless of the number of tasks. We note however that this is not the case for all MHR distributions.

The continuity assumption plays a fundamental role in the above results. In particular, we give a lower bound of $\Omega\left(\frac{\ln n}{\ln \ln n}\right)$ for the case of i.i.d. processing times that uses a discrete MHR distribution. Finally, we also consider the bounded overload and the sieve and bounded overload mechanisms that were studied by Chawla et al. [7], and present some instances that lower-bound their performance.

Related Work. One of the fundamental papers on the approximability of scheduling with unrelated machines is by Lenstra et al. [22] who provide a polynomial time algorithm that approximates the optimal makespan within a factor of 2. They also prove that it is NP-hard to approximate the optimal makespan within a factor of $3/2$ in this setting. In the mechanism design setting, Nisan and Ronen [26] prove that the well known VCG mechanism achieves an n-approximation of the optimal makespan, while no truthful mechanism can achieve approximation ratio better than 2. Note that the upper bound immediately carries over to the Bayesian and the prior-independent scheduling case. The lower bound has been improved by Christodoulou et al. [9] and Koutsoupias and Vidali [20] to 2.61, while Ashlagi et al. [2] prove the tightness of the upper bound for anonymous mechanisms. In contrast to the negative result on the prior free setting presented in [2], truthful mechanisms can achieve sublinear approximation when the processing times are stochastic. In fact, we prove here that VCG can achieve a sublogarithmic approximation, and even a constant one for some cases, while similar bounds for other mechanisms have also been presented by Chawla et al. [7].

For the special case of related machines, where the private information of each machine is a single value, Archer and Tardos [1] were the first to give a 3-approximation truthful in expectation mechanism, while now truthful PTAS are known by the works of Christodoulou and Kovács [8] and Dhangwatnotai et al. [13]. Putting computational considerations aside, the best truthful mechanism in this single-dimensional setting is also optimal. Lavi and Swamy [21] managed to prove constant approximation ratio for a special, yet multi-dimensional scheduling problem; they consider the case where the processing time of each task can take one of two fixed values. Yu [32] then generalized this result to two-range-values, while together with Lu and Yu [24] and Lu [23], they gave constant (better than 1.6) bounds for the case of two machines.

Daskalakis and Weinberg [11] consider computationally tractable approximations of the best *truthful* mechanism when the processing times of the tasks follow distributions (with finite support) that are known to the mechanism designer. In fact the authors provide a reduc-

tion of this problem to an algorithmic problem. Chawla et al. [6] showed that there can be no approximation-preserving reductions from mechanism design to algorithm design for the makespan objective, however the authors in [11] bypass this inapproximability by considering the design of bi-criterion approximation algorithms.

Prior-independent mechanisms have been mostly considered in the context of optimal auction design, where the goal is to design an auction mechanism that maximizes the seller's revenue. Inspired by the work of Dhangwatnotai et al. [14], Devanur et al. [12] and Roughgarden et al. [29] independently provide approximation mechanisms for multi-dimensional settings. Moreover, Dughmi et al. [15] identify conditions under which VCG obtains a constant fraction of the optimal revenue, while Hartline and Roughgarden [19] prove Bulow-Klemperer type results for VCG. Prior robust optimization is also discussed by Sivan [30].

Chawla et al. [7] are the first to consider *prior-independent* mechanisms for the (Bayesian) scheduling problem. They introduce two variants of the VCG mechanism and bound their approximation ratios. In particular, the bounded overload mechanism is prior-independent and achieves a $O(\frac{m}{n})$ approximation of the expected optimal makespan when the processing times of the tasks are drawn from machine-identical MHR distributions. For the case where the processing times of the tasks are i.i.d. from an MHR distribution, the authors prove that sieve and bounded overload mechanisms can achieve an $O(\sqrt{\ln n})$ approximation of the expected optimal makespan, as well as an approximation ratio of $O((\ln \ln n)^2)$ under the additional assumption that there are at least $n \ln n$ tasks. We note that to achieve these improved approximation ratios, a sieve and bounded overload mechanism needs to have access to a small piece of information regarding the distribution of the processing times, in particular the expectation of the minimum of a certain number of draws; nevertheless, this still breaks the prior-independence requirement.

The VCG mechanism is strongly represented in the above works. Its simplicity and amenability to practise dictate a detailed analysis of its performance for the Bayesian scheduling problem.

2 Preliminaries and Notation

Assume that we have n unrelated parallel machines and $m \geq n$ tasks that need to be scheduled to these machines. Let t_{ij} denote the processing time of task j for machine i. In the Bayesian scheduling problem, each t_{ij} is independently drawn from some probability distribution $\mathcal{D}_{ij}$. In this paper we mainly consider the machine-identical setting, that is the processing times of a specific task j are drawn from the same distribution D_j for all the machines. This is a standard assumption for the problem (see also [7]). We also consider the case where both machines and tasks are considered a priori identical, and the processing times t_{ij} are all i.i.d. drawn from the same distribution $\mathcal{D}$. The goal is to design a truthful mechanism that minimizes the expected makespan of the schedule.

We consider the VCG mechanism, the most natural and standard choice for a truthful mechanism. Thus, we henceforth assume that the machines always declare their true processing times. VCG minimizes the total workload by allocating each task to the machine that minimizes its processing time. So, if α denotes the allocation function of VCG (we omit the dependence on $\mathbf{t}$ for clarity of presentation) then, for any task j, $\alpha_{ij} = 1$ for some machine i such that $t_{ij} = \min_{i'}\{t_{i'j}\}$, otherwise $\alpha_{ij} = 0$. Without loss of generality we assume that in case of a tie, the machine is chosen uniformly at random[2]. The expected maximum makespan of VCG is then computed as

$$\mathbb{E}\left[\mathrm{VCG}(\mathbf{t})\right] = \mathbb{E}\left[\max_i \sum_{j=1}^{m} a_{ij} t_{ij}\right].$$

[2]We note here that for continuous distributions, such events of ties occur with zero probability.

In what follows, we use variable $Y_{i,j}$ to denote the processing time of task j on machine i under VCG, that is $Y_{i,j} = \alpha_{ij} t_{ij}$. We also denote by $Y_i = \sum_{j=1}^{m} Y_{i,j}$ the workload of machine i.

Note that in the machine-identical setting $\alpha_{ij} = 1$ with probability $\frac{1}{n}$ for any task j. So, VCG exhibits a balls-in-bins type behaviour in this setting, as the machine that minimizes the processing time of each task, and hence, the machine that will be allocated the task, is chosen uniformly at random for each task. We thus know from traditional balls-in-bins analysis, that the expected maximum number of tasks that will be allocated to any machine will be $\Theta\left(\frac{\ln n}{\ln \ln n}\right)$, whenever $m = \Theta(n)$. For more precise balls-in-bins type bounds see Raab and Steger [28]. We will use the following theorem to prove in Section 3 that this is actually the worst case scenario for VCG:

Theorem 1 (Berenbrink et al. [5])**.** *Assume two vectors $\mathbf{w} \in \mathbb{R}^m$, $\mathbf{w}' \in R^{m'}$ with $m \leq m'$ and their values in non-increasing order (that is $w_1 \geq w_2 \geq \ldots \geq w_m$ and $w'_1 \geq w'_2 \geq \ldots \geq w'_{m'}$). If the following two conditions hold:*

(i) $\sum_{j=1}^{m} w_j = \sum_{j=1}^{m'} w'_j$

(ii) $\sum_{j=1}^{k} w_j \geq \sum_{j=1}^{k} w'_j \quad$ *for all* $k \in [m]$,

then the expected maximum load when allocating m balls with weights according to $\mathbf{w}$ is at least equal to the expected maximum load when allocating m' balls with weights according to $\mathbf{w}'$ to the same number of bins.

Following [5] we say that vector $\mathbf{w}$ *majorizes* $\mathbf{w}'$ whenever $\mathbf{w}$ and $\mathbf{w}'$ satisfy Conditions (i) and (ii) of Theorem 1.

Probability preliminaries. We now give some additional notation regarding properties of distributions that will be used in the analysis.

Let T be a random variable following a probability distribution $\mathcal{D}$. Assuming we perform n independent draws from $\mathcal{D}$, we use $T[r:n]$ to denote the r-th order statistic (the r-th smallest) of the resulting values, following the notation from [7]. In particular, $T[1:n]$ will denote the minimum of n draws from $\mathcal{D}$, while $T[1:n][m:m]$ denotes the maximum value of m independent experiments where each one is the minimum of n draws from $\mathcal{D}$. Note that for $t_{ij} \sim \mathcal{D}_j$, the expected processing time of machine i for task j under VCG is

$$\mathbb{E}[Y_{i,j}] = \Pr[\alpha_{ij} = 1]\, \mathbb{E}[t_{ij} \mid \alpha_{ij} = 1] = \frac{1}{n}\, \mathbb{E}[T[1:n]]. \tag{1}$$

In this work we also consider the class of probability distributions that have a *monotone hazard rate (MHR)*. A continuous distribution with pdf f and cdf F is MHR if its hazard rate $h(x) = \frac{f(x)}{1-F(x)}$ is a (weakly) increasing function. The definition of discrete MHR distributions is similar, only the hazard rate of a discrete distribution is defined as $h(x) = \frac{\Pr[X=x]}{\Pr[X \geq x]}$ (see e.g. Barlow et al. [4]). The following two technical lemmas demonstrate properties of MHR distributions. For completeness we present the proofs in Appendix A.

Lemma 1. *If T is an MHR random variable, then for every positive integer n, its first order statistic $T[1:n]$ is also MHR.*

Lemma 2. *For any* continuous *MHR random variable X and any positive integer r, $\mathbb{E}[X^r] \leq r!\, \mathbb{E}[X]^r$.*

We now introduce the notion of k-stretched distributions. The property that identifies these distributions plays an important role in the approximation ratio of VCG as we will see later in the analysis (Theorem 7).

Definition 1. Given a function k over integers, we call a distribution *k-stretched* if its order statistics satisfy

$$\mathbb{E}[T[1:n][n:n]] \geq k(n) \cdot \mathbb{E}[T[1:n]],$$

for all positive integers n.

We will use the following result by Aven to bound the expected maximum makespan of VCG.

Theorem 2 (Aven [3])**.** *If $X_1, X_2, \ldots, X_n$ are (not necessarily independent) random variables with mean μ and variance σ^2, then*

$$\mathbb{E}[\max_i X_i] \leq \mu + \sqrt{n-1}\sigma.$$

Finally, we use the notation introduced in the probability preliminaries to present some known bounds on the expected optimal makespan among all mechanisms, not necessarily truthful ones. So, if given a matrix of processing times $\mathbf{t}$ we denote its optimal makespan by $\mathrm{OPT}(\mathbf{t})$, we wish to bound $\mathbb{E}_{\mathbf{t}}[\mathrm{OPT}(\mathbf{t})]$ (we omit dependence on $\mathbf{t}$ for clarity of presentation). Part of the notorious difficulty of the scheduling problem stems exactly from the lack of general, closed-form formulas for the optimal makespan. However, the following two easy lower bounds are widely used (see e.g. [7]):

Observation 3. *If the processing times are drawn from machine-identical distributions, then the expected maximum makespan of the optimal mechanism is bounded by*

$$\mathbb{E}[\mathrm{OPT}] \geq \max\left\{\mathbb{E}\left[\max_j T_j[1:n]\right], \frac{1}{n}\sum_{j=1}^{m}\mathbb{E}\left[T_j[1:n]\right]\right\},$$

where T_j follows the distribution corresponding to task j.

3 Upper Bounds

In this section we provide results on the performance of the VCG mechanism for the Bayesian scheduling problem for different assumptions on the number of tasks (compared to the machines), and different distributional assumptions on their processing times. Our first result shows that VCG is $O\left(\frac{\ln n}{\ln \ln n}\right)$–approximate in the general case, without assuming identical tasks or even MHR distributions. We then consider some additional assumptions under which VCG achieves a constant approximation of the expected optimal makespan. In what follows, an allocation where all machines have the same workload will be called *fully balanced*.

Theorem 4. *VCG is $O\left(\frac{\ln n}{\ln \ln n}\right)$-approximate for the Bayesian scheduling problem with n identical machines.*

As we will see later in Theorem 10, this result is in general tight. In order to prove Theorem 4 we will make use of the following lemma:

Lemma 3. *If* VCG *is ρ-approximate for the prior free scheduling problem with identical machines on inputs for which the optimal allocation is fully balanced, then* VCG *is ρ-approximate for the Bayesian scheduling problem where the machines are a priori identical.*

Proof. We will show that for any instance of Bayesian scheduling with a priori identical machines, there exists a prior free scheduling instance with identical machines for which the approximation ratio of VCG is at least the same. In fact, there exists such a prior free instance, for which the optimal allocation is fully balanced.

Consider a Bayesian scheduling instance where $t_{ij} \sim \mathcal{D}_j$ for tasks $j \in [m]$ and machines $i \in [n]$. Let $\rho \geq 1$ be such that $\mathbb{E}_{\mathbf{t}}[\mathrm{VCG}(\mathbf{t})] = \rho \cdot \mathbb{E}_{\mathbf{t}}[\mathrm{OPT}(\mathbf{t})]$. Then, conditioning on the minimum processing times of the tasks, there exists an m-dimensional vector $(t_1^\star, \ldots, t_m^\star)$ such that

$$\mathbb{E}_{\mathbf{t}}\left[\mathrm{VCG}(\mathbf{t}) \;\middle|\; \min_i t_{i1} = t_1^\star \wedge \cdots \wedge \min_i t_{im} = t_m^\star\right] \geq \rho \cdot \mathbb{E}_{\mathbf{t}}\left[\mathrm{OPT}(\mathbf{t}) \;\middle|\; \min_i t_{i1} = t_1^\star \wedge \cdots \wedge \min_i t_{im} = t_m^\star\right].$$

Notice that, once such a minimum processing time $t_j^\star$ has been fixed for all tasks, the only randomization remaining within the expected makespan of VCG is the one with respect to its internal tie breaking; thus, if we let $\mathbf{t}^*$ denote the time matrix where task j has processing time $t_{ij} = t_j^*$ for all machines i, it holds that

$$\mathbb{E}_{\mathbf{t}}\left[\mathrm{VCG}(\mathbf{t}) \;\middle|\; \min_i t_{i1} = t_1^\star \wedge \cdots \wedge \min_i t_{im} = t_m^\star\right] = \mathrm{VCG}(\mathbf{t}^\star).$$

Also, once we have fixed the smallest element in every column of an input matrix $\mathbf{t}$ (a column contains the processing times of a single task on all the machines), reducing all other values of a column j to be equal to that minimum value $t_j^\star$ can only improve the optimal makespan, so

$$\mathbb{E}_{\mathbf{t}}\left[\mathrm{OPT}(\mathbf{t}) \;\middle|\; \min_i t_{i1} = t_1^\star \wedge \cdots \wedge \min_i t_{im} = t_m^\star\right] \geq \mathrm{OPT}(\mathbf{t}^\star).$$

Combining the above, we get that indeed $\mathrm{VCG}(\mathbf{t}^\star) \geq \rho \cdot \mathrm{OPT}(\mathbf{t}^\star)$.

It remains to be shown that, without loss, $\mathbf{t}^\star$ gives rise to an optimal (prior free) allocation that is fully balanced, that is all machines have exactly the same workload (equal to the optimal makespan). Indeed, if that is not the case, then for any machine whose workload is strictly below the optimal makespan, we can slightly increase the processing time $t_j^\star$ of one of its tasks j without affecting the optimal makespan, while at the same time that increase can only make the performance of VCG worse. □

We are now ready to prove Theorem 4. Lemma 3 essentially reduces the analysis of VCG for the Bayesian scheduling problem for identical machines to that of a simple weighted balls-in-bins setting:

Proof of Theorem 4. From Lemma 3, it is enough to analyze the performance of VCG on input matrices where the processing time of each task is the same across all machines and the optimal schedule is fully balanced. Without loss (by scaling) it can be further assumed that the optimal makespan is exactly 1. Then, since VCG is breaking ties uniformly at random, the problem is reduced to analyzing the expected maximum (weighted) load when throwing m balls with weights $(w_1, \ldots, w_m) = \mathbf{w}$ (uniformly at random) into n bins, when $\sum_{j=1}^m w_j = n$. Then, by Theorem 1, that maximum load is upper bounded by the expected maximum load of throwing n (unit weight) balls into n bins, because the n-dimensional unit vector $\mathbf{1}_n$ majorizes $\mathbf{w}$: $\mathbf{1}_n$'s components sum up to n and also $w_j \leq 1$ for all $j \in [n]$ (due to the assumption that the optimal makespan is 1). By classic balls-in-bins results (see e.g. [25, 28]), the expected maximum load of any machine is upper bounded by $\Theta\left(\frac{\ln n}{\ln \ln n}\right)$. □

We now focus on the special but important case where both tasks and machines are a priori identical:

Theorem 5. *VCG is $2\left(1 + \frac{n \ln n}{m}\right)$-approximate for the Bayesian scheduling problem with i.i.d. processing times drawn from a continuous MHR distribution.*

Proof. Let T be a random variable following the distribution from which the execution times t_{ij} are drawn. Following the notation introduced in the Introduction, the workload of a machine i is given by the random variable $Y_i = \sum_{j=1}^{m} Y_{i,j}$. Then, for the expected maximum makespan $\mathbb{E}[\max_i Y_i]$ and any real $s > 0$ it holds that

$$e^{s \cdot \mathbb{E}[\max_i Y_i]} \leq \mathbb{E}[e^{s \max_i Y_i}] = \mathbb{E}[\max_i e^{sY_i}] \leq \sum_{i=1}^{n} \mathbb{E}[e^{sY_i}] = \sum_{i=1}^{n} \prod_{j=1}^{m} \mathbb{E}[e^{sY_{i,j}}] = n\,\mathbb{E}[e^{sY_{1,1}}]^m, \quad (2)$$

where we have used Jensen's inequality through the convexity of the exponential function, and the fact that for a fixed machine i the random variables $Y_{i,j}$, $j = 1, \ldots, m$, are independent (the processing times are i.i.d. and VCG allocates each task independently of the others). We now bound the term $\mathbb{E}[e^{sY_{1,1}}]$:

$$\mathbb{E}[e^{sY_{1,1}}] = \mathbb{E}\left[\sum_{r=0}^{\infty} \frac{(sY_{1,1})^r}{r!}\right] = 1 + \sum_{r=1}^{\infty} s^r \frac{\mathbb{E}[Y_{1,1}^r]}{r!} = 1 + \frac{1}{n}\sum_{r=1}^{\infty} s^r \frac{\mathbb{E}[T[1:n]^r]}{r!} \leq 1 + \frac{1}{n}\sum_{r=1}^{\infty} s^r\,\mathbb{E}[T[1:n]]^r,$$

where for the last inequality we have used the fact that the first order statistic of an MHR distribution is also MHR (Lemma 1) and Lemma 2. Then, by choosing $s = s^* \equiv \frac{1}{2 \cdot \mathbb{E}[T[1:n]]}$ we get that

$$\mathbb{E}[e^{s^* Y_{1,1}}] = 1 + \frac{1}{n}\sum_{r=1}^{\infty} \frac{1}{2^r} \leq 1 + \frac{1}{n},$$

and (2) yields

$$\begin{aligned}
\mathbb{E}[\max_i Y_i] &\leq \ln\left(n\,\mathbb{E}[e^{s^* Y_{1,1}}]^m\right)\frac{1}{s^*} \\
&\leq 2\ln\left(n\left(1+\frac{1}{n}\right)^m\right)\mathbb{E}[T[1:n]] \\
&\leq 2\ln\left(ne^{m/n}\right)\mathbb{E}[T[1:n]] \\
&= 2\left(\ln n + \frac{m}{n}\right)\mathbb{E}[T[1:n]]. \qquad (3)
\end{aligned}$$

But from Observation 3 we know that $\mathbb{E}[\mathrm{OPT}] \geq \frac{m}{n}\,\mathbb{E}[T[1:n]]$ for the case of i.i.d. execution times, and the theorem follows. □

Notice that Theorem 5 in particular implies that VCG achieves a *small, constant* approximation ratio whenever the number of tasks is slightly more than that of machines:

Corollary 6. *VCG is 4-approximate for the Bayesian scheduling problem with $m \geq n \ln n$ i.i.d. tasks drawn from a continuous MHR distribution.*

The following theorem will help us analyze the performance of VCG for the complementary case to that of Corollary 6, that is when the number of tasks is $m \leq n \ln n$. Recall the notion of k-stretched distributions introduced in Definition 1.

Theorem 7. *VCG is $4\frac{\ln n}{k(n)}$-approximate for the Bayesian scheduling problem with $m \leq n \ln n$ i.i.d. tasks drawn from a k-stretched MHR distribution.*

Proof. From the proof of Theorem 5 (and (3) in particular) we can deduce that the approximation ratio of VCG is upper bounded by

$$\begin{aligned}\left(\ln n + \frac{m}{n}\right)\frac{2\,\mathbb{E}[T[1:n]]}{\mathbb{E}[\mathrm{OPT}]} &\leq \left(\ln n + \frac{m}{n}\right)\frac{2\,\mathbb{E}[T[1:n]]}{\mathbb{E}[T[1:n][m:m]]}\\ &\leq \left(\ln n + \frac{m}{n}\right)\frac{2\,\mathbb{E}[T[1:n]]}{\mathbb{E}[T[1:n][n:n]]}\\ &\leq \left(\ln n + \frac{m}{n}\right)\frac{2}{k(n)}\\ &\leq \frac{4\ln n}{k(n)},\end{aligned}$$

where we have used Observation 3 and the fact that $n \leq m \leq n \ln n$. □

In particular, we note that Theorem 7 yields a constant approximation ratio for VCG for the important special cases where the processing times are drawn independently from the uniform distribution on $[0, 1]$ or any exponential distribution. Indeed, the uniform distribution on $[0, 1]$ as well as any exponential distribution is ln-stretched. See Appendix B for a full proof. We get the following, complementing the results in Corollary 6:

Corollary 8. *VCG is* 4*-approximate for the Bayesian scheduling problem with i.i.d. processing times drawn from the uniform distribution on* $[0, 1]$ *or an exponential distribution.*

We point out that the above corollary can not be generalized to hold for all MHR distributions, as the lower bound in Theorem 10 implies. For example, it is not very difficult to check that by taking $\varepsilon \to 0$ and considering the uniform distribution over $[1, 1 + \varepsilon]$, no stretch factor $k(n) = \Omega(\ln n)$ can be guaranteed.

For our final positive result, we present an improved constant bound on the approximation ratio of VCG when we have many tasks:

Theorem 9. *VCG is* $1 + \sqrt{2}$*-approximate for the Bayesian scheduling problem with* $m \geq n^2$ *tasks with i.i.d. processing times drawn from a continuous MHR distribution.*

Proof. We use Theorem 2 to bound the performance of VCG in this setting. In order to do so, we first bound the expectation and the variance of the makespan of a single machine. From (1), for the workload Y_i of any machine i we have:

$$\mathbb{E}[Y_i] = \sum_{j=1}^{m} \mathbb{E}[Y_{i,j}] = \frac{1}{n}\sum_{j} \mathbb{E}[T[1:n]] = \frac{m}{n}\,\mathbb{E}[T[1:n]].$$

To compute the variance of the makespan of machine i, we note that the random variables $Y_{i,j}$ are independent with respect to j, for any fixed machine i and thus we can get

$$\begin{aligned}\mathrm{Var}[Y_i] &= \sum_{j=1}^{m} \mathrm{Var}[Y_{i,j}] = \sum_{j=1}^{m} \left(\mathbb{E}[Y_{i,j}^2] - \mathbb{E}[Y_{i,j}]^2\right)\\ &\leq \sum_{j=1}^{m} \mathbb{E}[Y_{i,j}^2] = \sum_{j=1}^{m} \mathbb{E}[a_{ij}^2 t_{ij}^2] = \frac{1}{n}\sum_{j=1}^{m} \mathbb{E}[T[1:n]^2]\\ &= \frac{m}{n}\,\mathbb{E}[T[1:n]^2].\end{aligned}$$

We are now ready to use Theorem 2 and bound the expected maximum makespan:

$$\begin{aligned}
\mathbb{E}[\max_i Y_i] &\leq \mathbb{E}[Y_1] + \sqrt{n-1}\sqrt{\mathrm{Var}[Y_1]} \\
&\leq \frac{m}{n}\,\mathbb{E}[T[1:n]] + \sqrt{m}\sqrt{\mathbb{E}[T[1:n]^2]} \\
&\leq \frac{m}{n}\,\mathbb{E}[T[1:n]] + \sqrt{2}\sqrt{m}\,\mathbb{E}[T[1:n]] \\
&\leq (1+\sqrt{2})\frac{m}{n}\,\mathbb{E}[T[1:n]] \\
&\leq (1+\sqrt{2})\mathbb{E}[\mathrm{OPT}],
\end{aligned}$$

where the third inequality follows from Lemma 2 (and Lemma 1), for the fourth inequality we use the assumption that $m \geq n^2$ and to complete the proof, the last inequality uses a lower bound on $\mathbb{E}[\mathrm{OPT}]$ from Observation 3. □

4 Lower Bounds

In this section we prove some lower bounds on the performance of VCG under different distributional assumptions on the processing times. In an attempt for a clear comparison of VCG with the mechanisms that were previously considered for the Bayesian scheduling problem (in [7]), we provide instances that lower bound their performance as well.

Theorem 10. *For any number of tasks, there exists an instance of the Bayesian scheduling problem where VCG is not better than $\Omega\left(\frac{\ln n}{\ln \ln n}\right)$-approximate and the processing times are drawn from machine-identical continuous MHR distributions.*

Proof. Consider an instance with n identical machines and m tasks where for any machine i, task j has processing time $t_{ij} = 1$ with probability 1 for $j = 1, \ldots, n-1$ and processing time $t_{ij} = \frac{1}{m-n+1}$ with probability 1 for $j = n, \ldots, m$. From classical results from balls-in-bins analysis (see also the proof of Theorem 4) we can deduce that the expected maximum number of unit-weight tasks allocated to any machine by VCG, is $\Omega\left(\frac{\ln n}{\ln \ln n}\right)$. On the other hand, there exists an allocation that achieves a makespan equal to 1, that is to allocate all of the $m - n + 1$ "small" tasks to a single machine and allocate each of the remaining unit-cost tasks to a different machine. The theorem follows by noticing that we can without loss replace these point-mass distributions on 1 and $\frac{1}{m-n+1}$ with uniform distributions over small intervals around that points. □

Notice that when the number of tasks equals that of the machines, i.e. $m = n$, then the lower bound of Theorem 10 holds for an instance of i.i.d. distributions and not just for identical machines. However, if we restrict our focus only on discrete distributions, then we can strengthen that lower bound to hold for i.i.d. distributions for essentially any number of tasks and not only for $m = n$:

Theorem 11. *For any number of $m = O(ne^n)$ tasks, there exists an instance of the Bayesian scheduling problem where VCG is not better than $\Omega\left(\frac{\ln n}{\ln \ln n}\right)$-approximate and the tasks have i.i.d. processing times drawn from a discrete MHR distribution.*

Proof. Consider an instance with n identical machines and m tasks where the processing times t_{ij} are drawn from $\{0, 1\}$ such that $t_{ij} = 1$ with probability $\left(\frac{n}{2m}\right)^{\frac{1}{n}} \equiv p$ and $t_{ij} = 0$ with probability $1 - p$. Notice that this is a well-defined distribution, since for all $m \geq n$ we have $p < 1$. Furthermore, it is easy to check that this distribution is MHR; its hazard rate at 0 is $\frac{\Pr[t_{ij}=0]}{\Pr[t_{ij}\geq 0]} = \frac{1-p}{1} = 1 - p$ and at 1 is $\frac{\Pr[t_{ij}=1]}{\Pr[t_{ij}\geq 1]} = \frac{p}{p} = 1$.

Next, let M be the random variable denoting the number of tasks whose best processing time in any machine is non-zero, that is

$$M = |\{j \mid \min_i t_{ij} = 1\}|.$$

Then M follows a binomial distribution with probability of success p^n and m trials, since the probability of a task having processing time 1 at *all* machines (success) is p^n, while there are m tasks in total. Given the definition for p, the average number of tasks that will end up requiring a processing time of 1 at any machine is $\mathbb{E}[M] = mp^n = \frac{n}{2}$. Also, we can derive that $\Pr[M > 3n] \leq e^{-n}$ by using a Chernoff bound[3]. As we have argued before, we can use classical results from balls-in-bins analysis to bound the performance of VCG. So, when $M \leq 3n$, we know that the expected maximum makespan (since each task has processing time at least 1) will be $\Omega\left(\frac{\ln n}{\ln \ln n}\right)$. That event happens almost surely, with probability $1 - e^{-n} = 1 - o(1)$.

On the other hand, we next show that the mechanism that simply balances the M "expensive" tasks across the machines (by allocating $\left\lceil \frac{M}{n} \right\rceil$ of them to every machine) achieves a constant maximum makespan, hence providing a constant upper-bound on the optimal maximum makespan:

$$\mathbb{E}[\mathrm{OPT}] \leq \Pr[M \leq 3n] \cdot \frac{3n}{n} \cdot 1 + \Pr[M > 3n] \cdot \left\lceil \frac{m}{n} \right\rceil \cdot 1 \leq 4 + \frac{m}{ne^n} = O(1).$$

□

Notice however that Theorem 11 still leaves open the possibility for *continuous* MHR distributions to perform better (see also Theorem 9 and Corollary 6).

We finally conclude with a couple of simple observations, for the sake of completeness. First, our initial requirement (see Section 2) for identical machines (which is a standard one, see [7]) is crucial for guaranteeing any non-trivial approximation ratios on the performance of VCG:

Observation 12. *There exists an instance of the Bayesian scheduling problem where VCG is not better than n-approximate even when the tasks are identically distributed according to continuous MHR distributions.*

Proof. Assume $\frac{m}{n}$ being an integer, and give as input the point-mass distributions $t_{1j} = 1 - \varepsilon$ and $t_{ij} = 1$ for all $j \in [m]$ and $i = 2, 3, \ldots, n$, where $\varepsilon \in (0, 1)$. Notice that the execution times are indeed identical across the tasks. The VCG mechanism allocates all jobs to machine 1, for a makespan of $m \cdot (1 - \varepsilon)$, while the algorithm that assigns $\frac{m}{n}$ jobs to each machine achieves a makespan of at most $\frac{m}{n} \cdot 1$, resulting to a ratio of n as $\varepsilon \to 0$. Without loss, the above analysis carries over even if we replace the point-mass distributions with uniform distributions around a small interval around the values $1 - \varepsilon$ and 1. These distributions are MHR, which concludes the proof. □

We now present some lower bounds on the performance of the mechanisms analyzed by Chawla et al. [7]. A definition of these mechanisms can be found in the Introduction. The following demonstrates that the analysis of the approximation ratio for the class of bounded overload mechanisms presented in [7] is asymptotically tight:

Observation 13. *For any number of $m \geq n$ tasks, there exists an instance of the Bayesian scheduling problem where a bounded overload mechanism with parameter c is not better than $\min\{c\frac{m}{n}, n-1\}$-approximate and the processing times are drawn from machine-identical continuous MHR distributions.*

[3]Here we use the following form, with $\beta = 1 + \sqrt{5}$: for any $\beta > 0$, $\Pr[X \geq (1+\beta)\mu] \leq e^{-\frac{\beta^2}{2+\beta}\mu}$ for any binomial random variable with mean μ.

Proof. Consider the instance of Theorem 10 and recall that the optimal makespan is equal to 1. We note that since each task has the same processing time at any machine, all possible allocations such that no machine is assigned to more than $c\frac{m}{n}$ tasks are valid outputs of bounded overload mechanisms with parameter c. Now consider the bounded overload mechanism which fixes an ordering of the machines and then breaks ties according to that ordering. This mechanism would allocate at least $\min\{c\frac{m}{n}, n-1\}$ unit-cost tasks on the first machine in its ordering. □

The same instance can be used to bound the performance of the bounded overload mechanism with parameter c that breaks ties uniformly at random as well. Having sufficiently many tasks ($m = \Omega\left(\frac{n \ln n}{\ln \ln n}\right)$) implies that the mechanism behaves almost like the VCG mechanism while allocating the unit-cost tasks, assuming they are the first to be allocated. This gives a lower bound of $\Omega\left(\frac{\ln n}{\ln \ln n}\right)$ on the approximation ratio of this mechanism as well.

Similar instances can provide lower bounds on the performance of the class of sieve and bounded overload mechanisms with parameters c, β, and δ, even for the case of i.i.d. processing times. To see this notice that if all tasks have $t_{ij} = 1$ with probability 1 on any machine ($T[1:k] = 1$ for any k), and we choose threshold $\beta < 1$ as is done in [7] for the case $m \leq n \ln n$, then a sieve and bounded overload mechanism with parameters $c, \beta \leq 1$, and δ immediately reduces to a bounded overload mechanism with parameter c on δn machines.

References

[1] A. Archer and É. Tardos. Truthful mechanisms for one-parameter agents. In *FOCS*, pages 482–491, 2001.

[2] I. Ashlagi, S. Dobzinski, and R. Lavi. Optimal lower bounds for anonymous scheduling mechanisms. *Math. Oper. Res.*, 37(2):244–258, 2012.

[3] T. Aven. Upper (lower) bounds on the mean of the maximum (minimum) of a number of random variables. *Journal of Applied Probability*, 22(3):pp. 723–728, 1985.

[4] R. E. Barlow, A. W. Marshall, and F. Proschan. Properties of probability distributions with monotone hazard rate. *Ann. Math. Statist.*, 34(2):375–389, 06 1963.

[5] P. Berenbrink, T. Friedetzky, Z. Hu, and R. Martin. On weighted balls-into-bins games. *Theoretical Computer Science*, 409(3):511 – 520, 2008.

[6] S. Chawla, N. Immorlica, and B. Lucier. On the limits of black-box reductions in mechanism design. In *STOC*, pages 435–448, 2012.

[7] S. Chawla, J. D. Hartline, D. Malec, and B. Sivan. Prior-independent mechanisms for scheduling. In *STOC*, pages 51–60, 2013.

[8] G. Christodoulou and A. Kovács. A deterministic truthful PTAS for scheduling related machines. *SIAM J. Comput.*, 42(4):1572–1595, 2013.

[9] G. Christodoulou, E. Koutsoupias, and A. Vidali. A lower bound for scheduling mechanisms. *Algorithmica*, 55(4):729–740, 2009.

[10] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11(1):17–33, 1971.

[11] C. Daskalakis and S. M. Weinberg. Bayesian truthful mechanisms for job scheduling from bi-criterion approximation algorithms. In *SODA*, pages 1934–1952, 2015.

[12] N. R. Devanur, J. D. Hartline, A. R. Karlin, and C. T. Nguyen. Prior-independent multi-parameter mechanism design. In *WINE*, pages 122–133, 2011.

[13] P. Dhangwatnotai, S. Dobzinski, S. Dughmi, and T. Roughgarden. Truthful approximation schemes for single-parameter agents. *SIAM J. Comput.*, 40(3):915–933, 2011.

[14] P. Dhangwatnotai, T. Roughgarden, and Q. Yan. Revenue maximization with a single sample. *Games and Economic Behavior*, 91:318–333, 2015.

[15] S. Dughmi, T. Roughgarden, and M. Sundararajan. Revenue submodularity. *Theory of Computing*, 8(1):95–119, 2012.

[16] Y. Giannakopoulos and M. Kyropoulou. The VCG mechanism for bayesian scheduling. In *Web and Internet Economics*, WINE '15, pages 343–356, 2015. URL `http://dx.doi.org/10.1007/978-3-662-48995-6_25`.

[17] T. Groves. Incentives in Teams. *Econometrica*, 41(4):617–31, July 1973.

[18] L. A. Hall. Approximation algorithms for scheduling. In D. S. Hochbaum, editor, *Approximation Algorithms for NP-hard Problems*, pages 1–45. PWS, Boston, 1997.

[19] J. D. Hartline and T. Roughgarden. Simple versus optimal mechanisms. In *EC*, pages 225–234, 2009.

[20] E. Koutsoupias and A. Vidali. A lower bound of $1+\varphi$ for truthful scheduling mechanisms. *Algorithmica*, 66(1):211–223, 2013.

[21] R. Lavi and C. Swamy. Truthful mechanism design for multidimensional scheduling via cycle monotonicity. *Games and Economic Behavior*, 67(1):99–124, 2009.

[22] J. K. Lenstra, D. B. Shmoys, and É. Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Math. Program.*, 46:259–271, 1990.

[23] P. Lu. On 2-player randomized mechanisms for scheduling. In *WINE*, pages 30–41, 2009.

[24] P. Lu and C. Yu. An improved randomized truthful mechanism for scheduling unrelated machines. In *STACS*, pages 527–538, 2008.

[25] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[26] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1/2):166–196, 2001.

[27] N. Nisan and A. Ronen. Computationally feasible VCG mechanisms. *J. Artif. Int. Res.*, 29(1):19–47, 2007.

[28] M. Raab and A. Steger. "Balls into bins" - A simple and tight analysis. In *RANDOM*, pages 159–170, 1998.

[29] T. Roughgarden, I. Talgam-Cohen, and Q. Yan. Supply-limiting mechanisms. In *EC*, pages 844–861, 2012.

[30] B. Sivan. *Prior Robust Optimization*. PhD thesis, University of Wisconsin-Madison, 2013.

[31] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16(1):8–37, March 1961.

[32] C. Yu. Truthful mechanisms for two-range-values variant of unrelated scheduling. *Theor. Comput. Sci.*, 410(21-23):2196–2206, May 2009.

A Omitted Proofs from Section 2

Lemma 1. If T is an MHR random variable then for any positive integer n, its first order statistic $T[1:n]$ is also MHR.

Proof. If T is a continuous real random variable with cdf F and pdf f, then the cdf and pdf of $T[1:n]$ are $F^{(1)}(x) = 1-(1-F(x))^n$ and $f^{(1)}(x) = nf(x)(1-F(x))^{n-1}$, respectively. So, the hazard rate of $T[1:n]$ is

$$\frac{f^{(1)}(x)}{1-F^{(1)}(x)} = \frac{n(1-F(x))^{n-1}f(x)}{(1-F(x))^n} = n\frac{f(x)}{1-F(x)},$$

which is increasing since $\frac{f(x)}{1-F(x)}$ is increasing. For the case where T is a discrete random variable, the proof is similar. In particular, the hazard rate of $T[1:n]$ is

$$\frac{\Pr[T[1:n]=x]}{\Pr[T[1:n]\geq x]} = \frac{n\Pr[T\geq x]^{n-1}\Pr[T=x]}{\Pr[T\geq x]^n} = n\frac{\Pr[T=x]}{\Pr[T\geq x]},$$

which is increasing since $\frac{\Pr[T=x]}{\Pr[T\geq x]}$ is increasing. □

Lemma 2. For any *continuous* MHR random variable X and any positive integer r, $\mathbb{E}[X^r] \leq r!\,\mathbb{E}[X]^r$.

Proof. For any positive integer s, denote the normalized moments $\lambda_s \equiv \frac{\mathbb{E}[X^s]}{s!}$. Then from [4] we know that for all real $t > s > 0$ and integer i,

$$\left(\frac{\lambda_{i+t}}{\lambda_i}\right)^s \leq \left(\frac{\lambda_{i+s}}{\lambda_i}\right)^t.$$

By selecting $t = r$, $s = 1$ and $i = 0$, this inequality gives $\lambda_r\lambda_0^{r-1} \leq \lambda_1^r$. We get the desired inequality by noticing that $\lambda_r = \mathbb{E}[X^r]/r!$, $\lambda_1 = \mathbb{E}[X]$ and $\lambda_0 = \mathbb{E}[1] = 1$. □

The continuity assumption in Lemma 2 is essential, as it is demonstrated by the following example: consider a discrete random variable X over $\{0,1\}$ with $\Pr[X=0] = \frac{1}{2}+\varepsilon$ and $\Pr[X=1] = \frac{1}{2}-\varepsilon$, for some small $\varepsilon > 0$. This distribution is MHR since its hazard rate at 0 and 1 respectively is $h(0) = \frac{\Pr[X=0]}{\Pr[X\geq 0]} = \frac{1}{2}+\varepsilon$ and $h(1) = \frac{\Pr[X=1]}{\Pr[X\geq 1]} = 1$. However, it is easy to see that $\mathbb{E}[X^2] = \mathbb{E}[X] = \Pr[X=1] = \frac{1}{2}-\varepsilon$ and thus $\frac{\mathbb{E}[X]^2}{\mathbb{E}[X^2]} = \mathbb{E}[X] < \frac{1}{2}$.

B Proof of Corollary 8

Throughout this section we will use the fact that if T is a random variable with cdf F then for any positive integer n, the cdf's of the first and and last order statistics $T[1:n]$ and $T[n:n]$ are given by

$$F^{(1)}(x) = 1-(1-F(x))^n \quad \text{and} \quad F^{(n)}(x) = F^n(x), \tag{4}$$

respectively.

Lemma 4. *If T is a uniform random variable over $[0,1]$, then for all positive integers n, m*

$$\mathbb{E}[T[1:n]] = \frac{1}{n+1} \quad \textit{and} \quad \mathbb{E}[T[1:n][m:m]] = 1 - mB\left(m, 1+\frac{1}{n}\right),$$

where $B(x,y) \equiv \int_0^1 t^{x-1}(1-t)^{y-1}\,dt$ is the beta function.

Proof. If T is a uniformly distributed random variable over $[0,1]$ then its cdf is given by $F(x) = x$, $x \in [0,1]$. The first equality is very easy, since from (4) the cdf of $T[1:n]$ is $1-(1-x)^n$, thus its expectation is $\int_0^1 (1-x)^n\,dx = \frac{1}{n+1}$. For the second one, again from (4) it is straightforward to see that the cdf of $T[1:n][m:m]$ is $[1-(1-x)^n]^m$ so its expectation is $\int_0^1 1-[1-(1-x)^n]^m\,dx = 1-\int_0^1[1-(1-x)^n]^m\,dx$. Next we compute the value of this integral

$$I(m) \equiv \int_0^1 [1-(1-x)^n]^m\,dx.$$

We have:

$$\begin{aligned}
I(m) &= \int_0^1 [1-(1-x)^n]^{m-1}(1-(1-x)^n)\,dx \\
&= I(m-1) - \int_0^1 [1-(1-x)^n]^{m-1}(1-x)^n\,dx \\
&= I(m-1) - \frac{1}{n}\int_0^1 [1-(1-x)^n]^{m-1}\,(1-(1-x)^n)'\,(1-x)\,dx \\
&= I(m-1) - \frac{1}{nm}\int_0^1 ([1-(1-x)^n]^m)'\,(1-x)\,dx \\
&= I(m-1) - \frac{1}{nm}\left[(1-(1-x)^n)^m\,(1-x)\right]_{x=0}^{x=1} + \frac{1}{nm}\int_0^1 [1-(1-x)^n]^m(1-x)'\,dx \\
&= I(m-1) - \frac{1}{nm}\int_0^1 [1-(1-x)^n]^m\,dx \\
&= I(m-1) - \frac{1}{nm}I(m),
\end{aligned}$$

meaning that

$$I(m) = \frac{1}{1+\frac{1}{nm}}I(m-1) \quad \text{with} \quad I(1) = \int_0^1 1-(1-x)^n\,dx = 1-\frac{1}{n+1}.$$

Solving the above recurrence gives

$$I(m) = \frac{2\cdot 3\cdot\cdots\cdot m}{\left(1+\frac{1}{n}\right)\cdot\left(2+\frac{1}{n}\right)\cdot\cdots\cdot\left(m+\frac{1}{n}\right)} = \frac{\Gamma(m)\Gamma\left(1+\frac{1}{n}\right)}{\Gamma\left(m+1+\frac{1}{n}\right)} = mB\left(m,1+\frac{1}{n}\right),$$

where Γ denotes the (complete) gamma function. □

Lemma 5. *If T is an exponentially distributed random variable with parameter λ, then for all positive integers n, m*

$$\mathbb{E}[T[1:n]] = \frac{1}{\lambda n} \quad \textit{and} \quad \mathbb{E}[T[1:n][m:m]] = \frac{H_m}{\lambda n},$$

where $H_m = 1+\frac{1}{2}+\cdots+\frac{1}{m}$ is the harmonic function.

Proof. If T is exponentially distributed, then its cdf is given by $F(x) = 1-e^{-\lambda x}$, $x \in [0,\infty)$, where λ is a positive real parameter. The first equality is again easy, since from (4) the cdf of $T[1:n]$ is $1-(e^{-\lambda x})^n$, thus its expectation is $\int_0^\infty e^{-n\lambda\cdot x}\,dx = \frac{1}{\lambda n}$. For the second one, From (4)

it is straightforward to see that the cdf of $T[1:n][m:m]$ is $(1-e^{-\lambda n x})^m$ so its expectation is

$$\begin{aligned}
\int_0^\infty 1-(1-e^{-\lambda n x})^m\,dx &= \frac{1}{\lambda n}\int_0^\infty 1-(1-e^{-y})^m\,dy, && \text{by changing } y=\lambda n x,\\
&= \frac{1}{\lambda n}\int_0^1 \frac{1-(1-z)^m}{z}\,dz, && \text{by changing } z=e^{-y},\\
&= \frac{1}{\lambda n}\int_0^1 \frac{1-w^m}{1-w}\,dw, && \text{by changing } w=1-z,\\
&= \frac{1}{\lambda n}\int_0^1 \sum_{k=0}^{m-1} w^k\,dw\\
&= \frac{1}{\lambda n}\sum_{k=0}^{m-1}\frac{1}{k+1}\\
&= \frac{H_m}{\lambda n}.
\end{aligned}$$

□

To conclude the proof of Corollary 8, from Lemma 4 we deduce that the stretch factor of the uniform distribution is

$$\frac{\mathbb{E}[T[1:n][n:n]]}{\mathbb{E}[T[1:n]]} = (n+1)\left[1-nB\left(n,1+\frac{1}{n}\right)\right]$$

and from Lemma 5 the stretch factor for the exponential distribution with parameter λ is

$$\frac{\mathbb{E}[T[1:n][n:n]]}{\mathbb{E}[T[1:n]]} = H_n.$$

It can be verified that both the above quantities are lower-bounded by $\ln n$.