Robust Gittins for Stochastic Scheduling

Benjamin Moseley
moseleyb@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

Heather Newman
hanewman@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

Kirk Pruhs
kirk@cs.pitt.edu
University of Pittsburgh
Pittsburgh, Pennsylvania, USA

Rudy Zhou
rudyzhou@microsoft.com
Microsoft, Supply Chain Optimization Technologies
Washington DC, USA

Abstract

A common theme in stochastic optimization problems is that, theoretically, stochastic algorithms need to "know" relatively rich information about the underlying distributions. This is at odds with most applications, where distributions are rough predictions based on historical data. Thus, commonly, stochastic algorithms are making decisions using imperfect predicted distributions, while trying to optimize over some unknown true distributions.

We consider the fundamental problem of scheduling stochastic jobs preemptively on a single machine to minimize expected mean completion time in the setting where *the scheduler is only given imperfect predicted job size distributions*. If the predicted distributions are perfect, then it is known that this problem can be solved optimally by the Gittins index policy.

The goal of our work is to design a scheduling policy that is robust in the sense that it produces nearly optimal schedules even if there are modest discrepancies between the predicted distributions and the underlying real distributions. Our main contributions are:

- We show that the standard Gittins index policy is *not robust* in this sense. If the true distributions are perturbed by even an arbitrarily small amount, then running the Gittins index policy using the perturbed distributions can lead to an unbounded increase in mean completion time.
- We explain how to modify the Gittins index policy to make it robust, that is, to produce nearly optimal schedules, where the approximation depends on a new measure of error between the true and predicted distributions that we define.

Looking forward, the approach we develop here can be applied more broadly to many other stochastic optimization problems to better understand the impact of mispredictions, and lead to the development of new algorithms that are robust against such mispredictions.

SIGMETRICS Abstracts '25, Stony Brook, NY, USA

ACM ISBN 979-8-4007-1593-8/2025/06
https://doi.org/10.1145/3726854.3727315

CCS Concepts

• **Theory of computation** → **Scheduling algorithms**; • **Mathematics of computing** → **Combinatorial optimization**; *Queueing theory*.

Keywords

Stochastic scheduling; Approximation algorithms; Queueing theory

ACM Reference Format:
Benjamin Moseley, Heather Newman, Kirk Pruhs, and Rudy Zhou. 2025. Robust Gittins for Stochastic Scheduling. In *Abstracts of the 2025 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS Abstracts '25), June 9–13, 2025, Stony Brook, NY, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3726854.3727315

Extended Abstract

In this paper,[1] we study how to make stochastic scheduling – and stochastic optimization more generally – more robust against mispredictions in the given job size distributions. In particular, we address a variant of an open problem[2] stated by Scully, Grosof, and Mitzenmacher [9], that is, whether one can design and analyze a nonanticipatory scheduling policy that is robust with respect to minor errors in the reported probability distributions on the job sizes for the following classical stochastic scheduling problem:

> **Scheduling Problem Definition:** The input consists of non-negative probability distributions $\mathcal{D}_j$ for $j \in [n]$, where the jth job has size $P_j \sim \mathcal{D}_j$. We assume that the P_j's are independent.
> Our goal is to construct a *nonanticipatory* scheduling policy that *preemptively* schedules all n jobs to completion. That is, at any time $t \geq 0$, the scheduling policy selects the job to run at time t. A job completes if it has been run for P_j units of time (not necessarily consecutively). If a job j has run for q units of time before time t without completing, the scheduling policy knows that the probability distribution on the size of job j is now $\mathcal{D}_j$ conditioned on $P_j > q$. Since preemption is allowed, the policy can switch between jobs at any time.

[1] The full version can be found at: https://hanewman.github.io/_pages/gittins.pdf.
[2] The setting in [9] is the M/G/1 queue, which is a queue with Poisson arrivals and i.i.d. job sizes. On the other hand, in our work, we consider the no-arrival settings (finitely many jobs all available at time 0) with independent (but not necessarily identically distributed) job sizes.

> The objective is then to minimize the *expected* total completion time, $\mathbb{E}[\sum_{j\in[n]} C_j]$ (or equivalently, mean completion time, $\mathbb{E}[\frac{1}{n}\cdot\sum_{j\in[n]} C_j]$), where C_j is the time that job j completes with respect to a given scheduling policy.

We emphasize that we study this problem from the perspective of *worst-case analysis*, so we make no assumptions on the job size distributions other than independence. Further, the scheduler is initially only given the *distributions*, $\mathcal{D}_j$, and observes the realized processing times P_j over time as the jobs are scheduled.

This problem is a stochastic version of the scheduling problem $1 \mid pmtn \mid \sum_j C_j$ using the standard 3-field scheduling notation [2]. Many variations of this stochastic scheduling problem have been studied for decades (good overviews can be found in Pinedo [8] and Megow and Vredeveld [6]). For the moment, it is sufficient for our purposes to know the following facts. The nonanticipatory policy Shortest Expected Processing Time (SEPT), which always schedules the unfinished job whose expected processing time is minimum, is optimal for distributions with increasing hazard rates [10]. In this setting, optimal means optimal with respect to other nonanticipatory policies, and more precisely SEPT has the smallest expected total completion time of all nonanticipatory policies for this class of distributions. The Gittins (Index) policy is optimal (again relative to other possible nonanticipatory policies) for all distributions [1].

Many stochastic scheduling policies are quite brittle/non-robust, in that even small errors in the reported distributions can result in quite poor schedules. For many stochastic scheduling problems, the known algorithms require carefully rescaled and truncated statistics based on exponential [3, 4] and pth moments [7], which are highly sensitive to errors. As we will observe, the most natural candidate for a robust policy for the scheduling problem that we consider, namely the Gittins policy itself, is another such example.

The first step toward addressing the open question of whether there is a robust scheduling policy is to select an appropriate definition of distance between distributions so that we have a measure of the error in the reported distributions. For this purpose, we introduce the following error measure.

Definition 1. Consider two probability distributions, $\mathcal{D}$ and $\mathcal{D}'$, over $\mathbb{R}_{\geq 0}$. Let $\alpha \geq 1$ be a constant. Then the pair $(\mathcal{D}, \mathcal{D}')$ is *α-close* if for all $x \geq 0$, it is the case that

$$\frac{1}{\alpha}\cdot\mathbb{P}_{P\sim\mathcal{D}}(P > \alpha x) \leq \mathbb{P}_{P'\sim\mathcal{D}'}(P' > x) \leq \alpha\cdot\mathbb{P}_{P\sim\mathcal{D}}(P > x/\alpha)\,.$$

With this definition of closeness in hand, our first contribution is showing that the Gittins policy is brittle with respect to small errors in the predicted distributions. We then run into the issue that there are several alternative formulations of the Gittins policy, which, while all equivalent on perfectly accurate predicted distributions, differ in their executions when there are errors in the distributions. We elect to consider the formulation of the Gittins Index Priority Policy, abbreviated as GIPP, that we think is the most natural candidate for application to noisy data. The one we use is based on the partitioning of jobs into subjobs called quanta (singular: quantum). The Gittins algorithm runs these subjobs/quanta in some fixed order depending on the size distribution of each quantum. Having settled on this version of Gittins, we can then show that the Gittins policy is brittle, which requires some definitions.

Definition 2. Let $\hat{\mathcal{I}} = \{\hat{\mathcal{D}}_j\}_{j=1}^n$ and $\mathcal{I}^* = \{\mathcal{D}_j^*\}_{j=1}^n$ be collections of non-negative job size distributions. We let $\mathsf{A}(\mathcal{I}^*, \hat{\mathcal{I}})$ be a nonanticipatory policy that is given access to $\hat{\mathcal{I}}$ at the beginning of time and is unaware of $\mathcal{I}^*$. At any fixed time, the policy knows the completed job sizes and how much it has processed each incomplete job, where the job sizes are drawn from $\mathcal{I}^*$.

Definition 3. Let $\mathsf{A}(\cdot,\cdot)$ be a policy as in Definition 2, and $\hat{\mathcal{I}} = \{\hat{\mathcal{D}}_j\}_{j=1}^n$ and $\mathcal{I}^* = \{\mathcal{D}_j^*\}_{j=1}^n$ be collections of job size distributions. Then we conflate $\mathsf{A}(\mathcal{I}^*, \hat{\mathcal{I}})$ to represent the policy itself and also the *expected* total completion time for policy A, using the instance $\hat{\mathcal{I}}$ as the predicted distributions that are initially provided to A, and when the instance $\mathcal{I}^*$ is the true distribution on job sizes.

Note that while $\mathsf{A}(\mathcal{I}^*, \hat{\mathcal{I}})$ may not necessarily be well-defined, it will be well-defined within the context of our use. In particular, with the notation defined above, we can express the optimality of Gittins (GIPP) as follows. We let $\mathsf{OPT}(\mathcal{I})$ be the optimal expected cost among all nonanticapatory scheduling policies with input $\mathcal{I}$.

Theorem 4 ([5, 10, 11]). *For any instance $\mathcal{I} = \{\mathcal{D}_j\}_{j\in[n]}$, it is the case that $GIPP(\mathcal{I}, \mathcal{I}) = OPT(\mathcal{I})$.*

For brevity, we also define $\mathsf{GIPP}(\mathcal{I}) = \mathsf{GIPP}(\mathcal{I}, \mathcal{I})$ (the latter in the sense of Definition 3). We are ready to state our lower bound, which states that GIPP is *not* robust to mis-specified predicted distributions – even if those predictions are arbitrarily close to the true distributions.

Theorem 5. *For all $\alpha > 1$, and for all $n \geq 2$, there exist true distributions $\mathcal{D}_j^* = \mathcal{D}_j^*(\alpha, n)$ which depend on α and n for all $j \in [n]$ and predicted distributions, $\hat{\mathcal{D}}_j = \hat{\mathcal{D}}_j(n)$ which depend on n for all $j \in [n]$. All such distributions are finitely supported, and* every *pair $(\mathcal{D}_j^*, \hat{\mathcal{D}}_j)$ is α-close. Then the instances $\mathcal{I}^* = \{\mathcal{D}_j^*\}_{j\in[n]}$ and $\hat{\mathcal{I}} = \{\hat{\mathcal{D}}_j\}_{j\in[n]}$ satisfy*

$$GIPP(\mathcal{I}^*, \hat{\mathcal{I}}) = \Omega(n)\cdot GIPP(\mathcal{I}^*, \mathcal{I}^*).$$

Intuitively, the reason for the brittleness of the Gittins policy is that both the design and analysis of Gittins depend on conditional probability distributions derived from the job size distributions, and for these instances the conditional probability distributions can be quite brittle with respect to small errors in the predicted job size distributions.

We then turn to the open question of whether there is a robust policy for this problem. Our main contribution is a positive answer to this question. Our robust policy is a modest modification of the Gittins policy that naturally arises from consideration of the lower bound instances in the proof of Theorem 5. We uncreatively call this new policy the Robust Gittins policy, or more succinctly, RG. Roughly, if the Gittins policy would preempt a job j after running it continuously for q time units, the Robust Gittins policy would run j for an additional $(\alpha-1)q$ time units. Thus in the lower bound instance for Theorem 5, the Robust Gittins policy would not make the mistake that the Gittins policy makes of preempting a job right

before it is done. We are now ready to state our bound on the performance of the Robust Gittins policy. Note that our result assumes true and predicted distributions have finite support.

Theorem 6. *Let $\alpha \geq 1$. Let $\mathcal{I}^* = \{\mathcal{D}_j^*\}_{j\in[n]}$ be a collection of true size distributions with finite support on n jobs, and $\hat{\mathcal{I}} = \{\hat{\mathcal{D}}_j\}_{j\in[n]}$ be a collection of predicted size distributions with finite support on n jobs. Further assume that* every *pair of distributions $(\mathcal{D}_j^*, \hat{\mathcal{D}}_j)$ is α-close. Then*

$$RG(\mathcal{I}^*, \hat{\mathcal{I}}) \leq \alpha^6 \cdot GIPP(\mathcal{I}^*, \mathcal{I}^*).$$

Note that the right-hand side is just the optimal expected cost for the *true* distributions. So in the case that $\alpha = 1 + \varepsilon$ for some small ε, Theorem 6 states that the expected total completion time for our Robust Gittins policy is roughly within a $(1 + 6\varepsilon)$ factor of the optimal expected total completion time, despite receiving the potentially erroneous predicted job size distributions as input.

In the full version of this paper, we prove both of the above theorems (Theorem 5 and Theorem 6). Along the way, we prove some natural and desirable properties of our new error metric (it is monotone, symmetric, etc.), give examples of natural distributions which are close in this metric, and connect it with other well-known measures of distance between distributions such as Wasserstein- and Lévy-distance.

Acknowledgments

B. Moseley and H. Newman were supported in part by a Google Research Award, an Infor Research Award, a Carnegie Bosch Junior Faculty Chair, NSF grants CCF-2121744 and CCF-1845146 and ONR Grant N000142212702. K. Pruhs was supported in part by NSF grant CCF-2209654.

References

[1] John C Gittins. 1979. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 41, 2 (1979), 148–164.

[2] Ronald Lewis Graham, Eugene Leighton Lawler, Jan Karel Lenstra, and AHG Rinnooy Kan. 1979. Optimization and approximation in deterministic sequencing and scheduling: a survey. In *Annals of Discrete Mathematics*. Vol. 5. Elsevier, 287–326.

[3] Anupam Gupta, Amit Kumar, Viswanath Nagarajan, and Xiangkun Shen. 2021. Stochastic load balancing on unrelated machines. *Mathematics of Operations Research* 46, 1 (2021), 115–133.

[4] Jon Kleinberg, Yuval Rabani, and Éva Tardos. 1997. Allocating bandwidth for bursty connections. In *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing*. 664–673.

[5] Alan G. Konheim. 1968. A Note on Time Sharing with Preferred Customers. *Probability Theory and Related Fields* 9 (1968), 11–18. https://doi.org/10.1007/BF00531956

[6] Nicole Megow and Tjark Vredeveld. 2014. A Tight 2-Approximation for Preemptive Stochastic Scheduling. *Mathematics of Opreations Research* 39, 4 (2014), 1297–1310.

[7] Marco Molinaro. 2019. Stochastic ℓ_p load balancing and moment problems via the L-function method. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 343–354.

[8] Michael L. Pinedo. 2022. *Scheduling: Theory, Algorithms, and Systems* (6th ed.). Springer, New York. https://doi.org/10.1007/978-3-030-98545-8

[9] Ziv Scully, Isaac Grosof, and Michael Mitzenmacher. 2022. Uniform Bounds for Scheduling with Job Size Estimates. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022) (Leibniz International Proceedings in Informatics (LIPIcs))*, Mark Braverman (Ed.), Vol. 215. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 114:1–114:30. https://doi.org/10.4230/LIPIcs.ITCS.2022.114

[10] Kenneth C. Sevcik. 1971. Scheduling for Minimum Total Loss Using Service Time Distributions. *J. ACM* 18, 4 (1971), 717–729. https://doi.org/10.1145/321662.321667

[11] Gideon Weiss. 1995. On Almost Optimal Priority Rules for Preemptive Scheduling of Stochastic Jobs on Parallel Machines. *Advances in Applied Probability* 27, 3 (1995), 821–839. http://www.jstor.org/stable/1428135