# A Gittins Policy for Optimizing Tail Latency

Amit Harlev*
Cornell University
Center for Applied Mathematics
Ithaca, NY, USA

George Yu*
Cornell University
School of Operations Research and Information Engineering
Ithaca, NY, USA

Ziv Scully
Cornell University
School of Operations Research and Information Engineering
Ithaca, NY, USA

## Abstract

We consider the problem of scheduling to minimize asymptotic tail latency in an M/G/1 queue with unknown job sizes. When the job size distribution is heavy-tailed, numerous policies that do not require job size information (e.g. Processor Sharing, Least Attained Service) are known to be *strongly tail optimal*, meaning that their response time tail has the fastest possible asymptotic decay. In contrast, for light-tailed size distributions, only in the last few years have policies been developed that outperform simple First-Come First-Served (FCFS). The most recent of these is $\gamma$-Boost, which achieves strong tail optimality in the light-tailed setting. But thus far, all policies that outperform FCFS in the light-tailed setting, including $\gamma$-Boost, require known job sizes.

In this paper, we design a new scheduling policy that achieves *strong tail optimality in the light-tailed M/G/1 with unknown job sizes*. Surprisingly, the optimal policy turns out to be a variant of the Gittins policy, but with a novel and unusual feature: it uses a *negative discount rate*. Our work also applies to systems with partial information about job sizes, covering $\gamma$-Boost as an extreme case when job sizes are in fact fully known. This abstract summarizes our full paper [7].

## CCS Concepts

• **General and reference** → **Performance**; • **Mathematics of computing** → **Queueing theory**; • **Networks** → **Network performance modeling**; • **Computing methodologies** → *Model development and analysis*; • **Software and its engineering** → *Scheduling*.

## Keywords

scheduling; response time; sojourn time; tail latency; service level objective (SLO); M/G/1 queue; light-tailed distribution; Gittins index; Boost scheduling; multi-armed bandit

**ACM Reference Format:**
Amit Harlev, George Yu, and Ziv Scully. 2025. A Gittins Policy for Optimizing Tail Latency. 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/10.1145/3726854.3727267

*Authors contributed equally to this research.

*SIGMETRICS Abstracts '25, Stony Brook, NY, USA*


## 1 Motivation

Service level objectives (SLOs) for queueing systems typically relate to the tail of the system's response time distribution $T$. The tail is the function mapping a time $t$ to the probability $\mathbf{P}[T > t]$. SLOs typically ask that high percentiles of $T$ are not too large, i.e. that $\mathbf{P}[T > t]$ is small for large $t$.

Motivated by the problem of optimizing SLOs, we consider the problem of asymptotically minimizing $\mathbf{P}[T > t]$ in the $t \to \infty$ limit in an M/G/1. While SLOs often put requirements on a specific deadline $t$, it turns out that roughly, minimizing $\mathbf{P}[T > t]$ "for all large values of $t$" works well, and current state-of-the-art methods for minimizing tail latency come from minimizing this asymptotic objective [16]. For light-tailed job size distributions, this problem was open for some time [15] until recent work [4, 6, 13] culminated in an optimal policy for systems with *known* sizes [16]. However, the case of *unknown* sizes remains open.

In this work, we resolve the problem for unknown job sizes. Our job model, a discrete-time variant of the *Markov-process job model* [10, Chapter 14], actually handles a range of information models, covering: unknown sizes, where only the job size distribution is known to the scheduler; known sizes, where each job's exact size is known—for which we recover prior results [16]; and settings where the scheduler has partial information about each job's size. For concreteness, we focus our discussion on the case of unknown sizes; see the full paper [7] for a description of our full model.

## 2 Background on tail optimality

To understand what it means to optimize the response time tail, we first define the notion of asymptotic optimality. Consider an M/G/1 setting with job size distribution $S$ and arrival rate $\lambda$. Let $T_\pi$ denote the response time distribution under a scheduling policy $\pi$. We say that a policy $\pi$ is *weakly tail-optimal* if there exists a constant $c \geq 1$ such that

$$\sup_{\pi'} \limsup_{t \to \infty} \frac{\mathbf{P}[T_\pi > t]}{\mathbf{P}[T_{\pi'} > t]} \leq c. \tag{2.1}$$

We further say $\pi$ is *strongly tail-optimal* if $c = 1$. In the known-size case we take the supremum over all policies, but in the unknown-size setting, we limit to non-clairvoyant policies. We assume a preempt-resume model: the job in service may be paused and resumed at a later point without loss of progress.

The asymptotic tail behavior under a policy $\pi$ depends on whether the job's distribution is light- or heavy-tailed; Wierman and Zwart [15] showed that a policy cannot be tail-optimal for both heavy-tailed and light-tailed distributions. Recently, Yu and Scully [16, Appendix A], leveraging results of Wierman and Zwart [15], observe that for an important class of heavy-tailed distributions, many well-known policies are strongly tail-optimal. Several of these policies, such as Least Attained Service and Processor Sharing, do not

use job size information, so the problem of strong tail optimality for unknown sizes is largely solved in the heavy-tailed setting.

In the light-tailed setting, this is not the case. First-Come-First-Served (FCFS) was the best performing policy for some time in both the known and unknown size cases. In the known-size setting, FCFS was known to be weakly tail-optimal and conjectured to be strongly tail-optimal. In particular, the tail of FCFS is asymptotically exponential for light-tailed distributions, that is,

$$\mathbf{P}[T_{\text{FCFS}} > t] \sim C_{\text{FCFS}} e^{-\gamma t},$$

where $\gamma$ is called the *decay rate* and is known [8] to be the least positive real solution to

$$\gamma = \lambda(\mathbf{E}[\exp(\gamma S)] - 1), \tag{2.2}$$

and $C_{\text{FCFS}}$ is FCFS's *tail constant*. No policy has decay rate better than $\gamma$ [3, 12], so strong tail optimality amounts to minimizing

$$C_\pi = \lim_{t\to\infty} e^{\gamma t}\mathbf{P}[T_\pi > t].$$

Recently, new policies have emerged with better tail constant than FCFS, disproving the conjecture that it was strongly tail-optimal [4, 6, 13]. This line of work culminated in a strongly tail-optimal policy, *$\gamma$-Boost*, which optimizes the tail constant for class I light-tailed distributions when job sizes are known [16]. However, all of these policies make crucial use of job size information. Strong tail optimality for unknown job sizes is thus still open in the light-tailed setting,[1] so we ask:

> *In the light-tailed M/G/1 with unknown job sizes, what scheduling policy minimizes the tail constant $C_\pi$?*

## 3 A recent advance: boost policies for known job sizes

The policy that achieves strong-tail optimality in the known-size case belongs to the family of policies known as Boost policies, which are introduced and analyzed in [16]. We give a brief overview of the main ideas of [16] below, explaining how we adapt them to unknown sizes in Section 4.

Boost policies work by assigning every job a *boosted arrival time* and then serving jobs in order of increasing boosted arrival time. A job's boosted arrival time is given by

$$\text{boosted arrival time} = \text{arrival time} - \text{boost},$$

where the *boost* of a job is given by a boost function $b(s)$ that maps each job size to a non-negative boost. The strongly tail-optimal boost policy strikes the right balance between prioritizing short jobs vs. prioritizing jobs that have been in the system for a long time.

The key idea in [16] is to relate the problem of strong tail optimality in the M/G/1 queue to a deterministic batch scheduling problem. This idea follows from an alternative expression for the tail constant,

$$C_\pi = \lim_{\theta\to\gamma} \frac{\gamma - \theta}{\gamma}\mathbf{E}[\exp(\theta T_\pi)], \tag{3.1}$$

which comes from final value theorem [6, Theorem 4.3].[2] Informally, (3.1) tells us that minimizing $C_\pi$ is morally equivalent to "minimizing $\mathbf{E}[\exp(\gamma T_\pi)]$". Although this expectation is infinite for any policy in the M/G/1 queue, an analogous average is finite in the *finite batch* setting, where we start with a fixed set of jobs and there are no further arrivals. Yu and Scully [16] show that the optimal policy for the finite batch simplification of the problem also minimizes $C_\pi$ in the M/G/1 queue.

## 4 Key ideas

The optimal policy identified by Yu and Scully [16] requires knowledge of each job's size. We wish to use a similar approach when job sizes are unknown. We model jobs with unknown sizes as Markov chains with a terminating state. We then consider *state-based* scheduling policies, which, broadly speaking, alter a job's priority based on its trajectory of states. An example of an important class of policies that this captures is the class of policies that only use the amount of attained service, or *age*, of a job [11]. State-based scheduling is discussed formally in the full paper [7, Section 2]. For these policies, we ask:

(a) What is the optimal state-based scheduling policy?
(b) How do we prove its optimality?

Both of these require new ideas relative to the known-size case [16]. In brief, (a) requires a new observation but is resolved relatively easily once that observation is made, whereas (b) is more technically challenging and is where the main technical novelty of our work lies.

### 4.1 Finding the optimal scheduling policy

In the known-size case [16], the optimal policy arises by simplifying the problem from scheduling in an M/G/1 queue to scheduling a finite batch of jobs. In fact, the optimal policy for the known-size batch problem was (a minor variant of) a well-established policy in the literature [9, Section 3.1].

We also use a simplification to a batch problem to discover the optimal policy, but the optimal batch policy is different because it must use preemption. Our key insight is to frame the finite batch version of unknown-size scheduling as a Markovian multi-armed bandit problem, but with *inflation* instead of the usual discounting (see the full paper for more details [7, Section 3]). While this is at some level just a sign flip, i.e. inflation is simply a negative discount rate, to the best of our knowledge, the multi-armed bandit problem with inflation has never been studied in the literature. The key benefit of the multi-armed bandit framing is that the optimal policy is known, at least under discounting: it is the *Gittins index* policy. By adapting the "prevailing charge" argument of Weber [14], we confirm that an inflation variant of Gittins is indeed optimal for the finite batch simplification of our scheduling problem.

### 4.2 Proving optimality

For (b), our approach differs significantly from that of [16]. Roughly speaking, [16] proves optimality in the queueing setting by directly relating it to the batch setting. Their idea is to treat each *busy period* as a random instance of a deterministic batch problem. However,

[1] Throughout this abstract when we refer to light-tailed distributions, we mean specifically *class I* light-tailed distributions [1]. Class I distributions include many common light-tailed distributions and it is common to only consider this subset of light-tailed distributions when considering tail behavior (see for example [2, 8]).

[2] While this is stated for specific policies in [6], the proof of (3.1) presented therein holds for any policy as long as the job size distribution is class I.

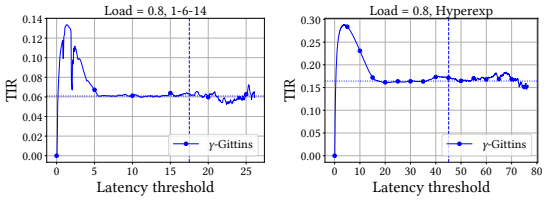


**Figure 5.1: Performance of $\gamma$-Gittins, the strongly tail-optimal policy for unknown sizes, on two different job size distributions. The plots show the tail improvement ratio (TIR), $1 - \frac{\mathbf{P}[T_{\gamma\text{-Gittins}} > t]}{\mathbf{P}[T_{\text{FCFS}} > t]}$, plotted against response time $t$. The dotted blue horizontal line indicates the numerical value of the theoretical asymptotic TIR, $1 - C_{\gamma\text{-Gittins}}/C_{\text{FCFS}}$. The vertical dashed blue line indicates the 99th percentile response time of $\gamma$-Gittins. On the left the job size distribution takes values $1/7, 6/7, 14/7$, with equal probability $1/3$. We refer to this as the 1-6-14 job size distribution, but divide everything by $7$ to normalize the mean. Service is provided in discrete time steps of length $0.1/7$. On the right is a discretized Hyperexponential distribution with two branches $\mathbf{Exp}(2), \mathbf{Exp}(1/3)$ and first branch probability $0.8$, with service provided in discrete time steps of length $0.1$. The load for both simulations is $\rho = 0.8$. Simulations run for one million busy periods. See the full paper [7, Section 5] for more simulations and details on parameters.**

with unknown sizes, setting busy periods as batches yields random instances of stochastic batch problems *with non-independent job sizes* [16, Appendix B]. Because independence is a crucial assumption for Gittins policies [5], the busy-period approach of [16] seems unlikely to work with unknown sizes.

Our main technical contribution is a new approach for (b) that proves optimality *directly in the queueing setting*, without going via the batch problem. Like our approach to the batch problem, our approach is based on Weber's proof [14] of the Gittins policy's optimality, with one key difference: our proof is "quantitative", rather than "qualitative". That is, Weber proves the Gittins policy is optimal without quantifying the performance it achieves. This qualitative approach does not work in the queueing setting for two main reasons.

The first problem is arrivals. Gittins policies are known to not be optimal in the presence of arrivals, except for in the special case of homogeneous Poisson arrivals [5]. While our arrivals are Poisson, they are *time-inhomogeneous*: the cost of a job depends directly on its arrival time.

The second problem is that we cannot reason directly about inflation rate $\gamma$ because $\mathbf{E}[e^{\gamma T_\pi}] = \infty$ for all policies $\pi$. Instead, we consider policies under inflation rate $\theta < \gamma$ and then let $\theta \to \gamma$. Due to the mismatch between $\theta$ and $\gamma$, we should not expect Gittins for inflation rate $\gamma$ to minimize $\mathbf{E}[e^{\theta T_\pi}]$ for any fixed $\theta < \gamma$.

We overcome both obstacles by using a *quantitative* approach. We quantify the performance of both Gittins and of a lower bound, and show that they match at the $\theta \to \gamma$ limit. We obtain the lower bound by quantitatively analyzing the lower bound from the qualitative proof of Weber [14].

## 5 Primary result

We present the *first strongly tail-optimal scheduling policy in the unknown-size setting*, $\gamma$-Gittins, for the M/G/1 queue with light-tailed job size distributions. In particular, we show that the boost policy with the following boost function is strongly tail optimal:

$$b_{\gamma\text{-Gittins}}(x) = \frac{1}{\gamma}\log\big(\Gamma_\gamma(x)\big) + \frac{1}{\gamma}\log\left(\frac{e^\gamma}{e^\gamma - 1}\right),$$

where $x$ is the state of the job, and $\Gamma_\gamma(x)$ is the $\gamma$-Gittins index of that state. See the full paper [7] for a description of the $\gamma$-Gittins index.

## Acknowledgments

This work was supported by the National Science Foundation (NSF) under grant no. CMMI-2307008. Amit Harlev was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate (NDSEG) Fellowship Program (https://ndseg.sysplus.com/). Code underlying plots and simulations was prepared in part using generative AI tools.

## References

[1] Joseph Abate, Gagan L. Choudhury, and Ward Whitt. 1994. Asymptotics for Steady-State Tail Probabilities in Structured Markov Queueing Models. *Communications in Statistics. Stochastic Models* 10, 1 (Jan. 1994), 99–143. doi:10.1080/15326349408807290

[2] Søren Asmussen. 2003. *Applied Probability and Queues* (2 ed.). Number 51 in Stochastic Modelling and Applied Probability. Springer, New York, NY. doi:10.1007/b97236

[3] Onno J. Boxma and Bert Zwart. 2007. Tails in Scheduling. *ACM SIGMETRICS Performance Evaluation Review* 34, 4 (March 2007), 13–20. doi:10.1145/1243401.1243406

[4] Nils Charlet and Benny Van Houdt. 2024. Tail Optimality and Performance Analysis of the Nudge-M Scheduling Algorithm. arXiv:2403.06588 [cs, math]

[5] John C. Gittins, Kevin D. Glazebrook, and Richard R. Weber. 2011. *Multi-Armed Bandit Allocation Indices* (2 ed.). Wiley, Chichester, UK.

[6] Isaac Grosof, Kunhe Yang, Ziv Scully, and Mor Harchol-Balter. 2021. Nudge: Stochastically Improving upon FCFS. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 5, 2, Article 21 (June 2021), 29 pages. doi:10.1145/3460088

[7] Amit Harlev, George Yu, and Ziv Scully. 2025. A Gittins Policy for Optimizing Tail Latency. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 9, 2, Article 17 (June 2025), 40 pages. doi:10.1145/3727109

[8] Michel Mandjes and Onno Boxma. 2023. *The Cramér–Lundberg model and its variants: A queueing perspective.* Springer Nature.

[9] Michael Pinedo. 2016. *Scheduling: Theory, Algorithms, and Systems* (5 ed.). Springer, Cham, Switzerland.

[10] Ziv Scully. 2022. *A New Toolbox for Scheduling Theory.* Ph. D. Dissertation. Carnegie Mellon University, Pittsburgh, PA.

[11] Ziv Scully, Mor Harchol-Balter, and Alan Scheller-Wolf. 2018. SOAP: One Clean Analysis of All Age-Based Scheduling Policies. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 2, 1, Article 16 (March 2018), 30 pages. doi:10.1145/3179419

[12] Alexander L. Stolyar and Kavita Ramanan. 2001. Largest Weighted Delay First Scheduling: Large Deviations and Optimality. *The Annals of Applied Probability* 11, 1 (Feb. 2001), 1–48. doi:10.1214/aoap/998926986

[13] Benny Van Houdt. 2022. On the Stochastic and Asymptotic Improvement of First-Come First-Served and Nudge Scheduling. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 6, 3 (Dec. 2022), 1–22. doi:10.1145/3570610

[14] Richard R. Weber. 1992. On the Gittins Index for Multiarmed Bandits. *The Annals of Applied Probability* 2, 4 (Nov. 1992), 1024–1033. doi:10.1214/aoap/1177005588

[15] Adam Wierman and Bert Zwart. 2012. Is Tail-Optimal Scheduling Possible? *Operations Research* 60, 5 (Oct. 2012), 1249–1257. doi:10.1287/opre.1120.1086

[16] George Yu and Ziv Scully. 2024. Strongly Tail-Optimal Scheduling in the Light-Tailed M/G/1. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 8, 2, Article 27 (June 2024), 33 pages. doi:10.1145/3656011