A Gittins Policy for Optimizing Tail Latency

AMIT HARLEV*, Cornell University, USA
GEORGE YU*, Cornell University, USA
ZIV SCULLY, Cornell University, USA

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 γ-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 γ-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 γ-Boost as an extreme case when job sizes are in fact fully known.

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*.

Additional Key Words and Phrases: 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. *Proc. ACM Meas. Anal. Comput. Syst.* 9, 2, Article 17 (June 2025), 40 pages. https://doi.org/10.1145/3727109

1 Introduction

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 [34]. For light-tailed job size distributions, this problem was open for some time [32] until recent work [7, 10, 28] culminated in an optimal policy for systems with *known* sizes [34]. However, the case of *unknown* sizes remains open.

*Authors contributed equally to this research.

Authors' Contact Information: Amit Harlev, Center for Applied Mathematics, Cornell University, Ithaca, NY, USA; George Yu, School of Operations Research and Information Engineering, Cornell University, Ithaca, NY, USA; Ziv Scully, School of Operations Research and Information Engineering, Cornell University, Ithaca, NY, USA.

This paper resolves the problem for unknown job sizes. Our job model, a discrete-time variant of the *Markov-process job model* [22, 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 [34]; and settings where the scheduler has partial information about each job's size. For concreteness, in the rest of this section, we focus our discussion on the case of unknown sizes; see Section 2 for a description of our full model.

1.1 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. Let T_π denote the response time distribution under a scheduling policy π. We say that a policy π is *weakly tail-optimal* if there exists a constant $c \geq 1$ such that

$$\sup_{\pi'} \limsup_{t\to\infty} \frac{\mathbf{P}[T_\pi > t]}{\mathbf{P}[T_{\pi'} > t]} \leq c. \tag{1.1}$$

We further say π 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 π depends on whether the job's distribution is light- or heavy-tailed; Wierman and Zwart [32] showed that a policy cannot be tail-optimal for both heavy-tailed and light-tailed distributions. Recently, Yu and Scully [34, Appendix A], leveraging results of Wierman and Zwart [32], 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 γ is called the *decay rate* and C_{FCFS} is FCFS's *tail constant*. No policy has decay rate better than γ [5, 27], 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 [7, 10, 28]. This line of work culminated in a strongly tail-optimal policy, *γ-Boost*, which optimizes the tail constant for class I light-tailed distributions when job sizes are known [34]. 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_π?*

[1]Throughout our introduction when we refer to light-tailed distributions, we mean specifically *class I* light-tailed distributions (Definition 2.1). 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 [3, 20]).

1.2 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 [34]. We give a brief overview of the main ideas of [34] below, explaining how we adapt them to unknown sizes in Section 1.3.

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 [34] 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,

$$C_\pi = \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[\exp(\theta T_\pi)], \tag{1.2}$$

which comes from final value theorem [10, Theorem 4.3].[2] Informally, (1.2) tells us that minimizing C_π 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 [34] show that the optimal policy for the finite batch simplification of the problem also minimizes C_π in the M/G/1 queue.

1.3 Key ideas

The optimal policy identified by Yu and Scully [34] 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 [26]. State-based scheduling is discussed formally in Section 2.1. 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 [34]. 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.

1.3.1 Finding the optimal scheduling policy. In the known-size case [34], 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 [21, 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 introduction to 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

[2] While this is stated for specific policies in [10], the proof of (1.2) presented therein holds for any policy as long as the job size distribution is class I (Definition 2.1).

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 [29], we confirm that an inflation variant of Gittins is indeed optimal for the finite batch simplification of our scheduling problem.

1.3.2 Proving optimality. For (b), our approach differs significantly from that of [34]. Roughly speaking, [34] 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, with unknown sizes, setting busy periods as batches yields random instances of stochastic batch problems *with non-independent job sizes* [34, Appendix B]. Because independence is a crucial assumption for Gittins policies [9], the busy-period approach of [34] 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 [29] 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 [9]. 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 γ because $\mathbf{E}[e^{\gamma T_\pi}] = \infty$ for all policies π. Instead, we consider policies under inflation rate $\theta < \gamma$ and then let $\theta \to \gamma$. Due to the mismatch between θ and γ, we should not expect Gittins for inflation rate γ 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 [29].

1.4 Contributions

We present the *first strongly tail-optimal scheduling policy in the unknown-size setting*, γ-Gittins, for the M/G/1 queue with light-tailed job size distributions. We study our policy in both theory and simulations. We make the following contributions:

- (Section 2) We propose a generalization of the boost policies introduced by Yu and Scully [34] to policies for a flexible discrete-time, Markov-process job model (Section 2.1), and define γ-Gittins, the boost function which achieves strong tail optimality.
- (Section 3) We relate our problem to a special case of the Gittins theory where there is *inflation* instead of the standard discounting, and show that the classical Gittins theory results hold in this setting. We use this connection to construct a lower bound on the optimal tail constant and prove, assuming results in Section 4, that γ-Gittins attains this lower bound.
- (Section 4) We generalize the analysis of boost policies in [34] to the boost policies described in Section 2 and provide an explicit formula for the tail constant of any boost policy (Theorem 4.1). We then use Theorem 4.1 to close the gap in Section 3.
- (Section 5) We show in simulation that γ-Gittins improves upon FCFS's performance. See Fig. 1.1 for an initial example. We observe larger gains for higher-variance size distributions.

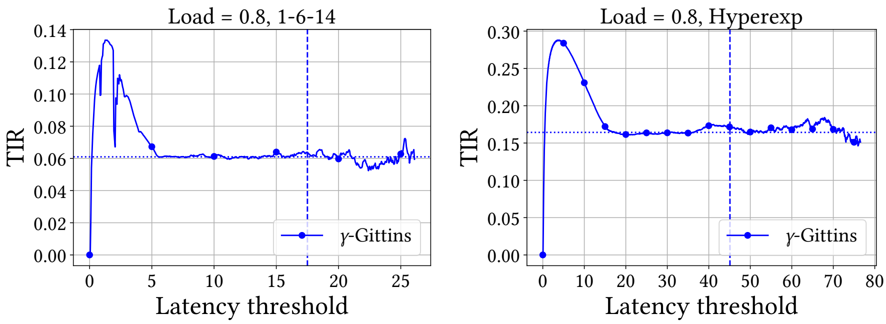

Fig. 1.1. Performance of γ-Gittins, the strongly tail-optimal policy for unknown sizes, on two different job size distributions. The plots show the tail improvement ratio (TIR), $1 - \frac{\mathbf{P}[T_{\gamma\text{-Gittins}} > t]}{\mathbf{P}[T_{\text{FCFS}} > t]}$, plotted against response time t. The 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 γ-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 $\text{Exp}(2), \text{Exp}(1/3)$ 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 Section 5 for more simulations and details on parameters.

2 System model

The queueing setting is that of [34, Section 2], but where, as mentioned in Section 1, we consider discrete-time jobs for ease of exposition.[3] We consider an M/G/1 queue with arrival rate λ, discrete-time job size distribution $S > 0$, and load $\rho = \lambda \mathbf{E}[S]$. We make the standard assumption $\rho < 1$, which implies stability and ergodicity of the system, and we assume the system is in its stationary distribution. Note that while our job sizes are discrete, arrivals are still Poisson. We further assume that S is a class I distribution, meaning it is in the following class of light-tailed distributions, which encompasses most well-behaved light-tailed distributions.[4]

Definition 2.1. A distribution S is *class I* [1] if its moment generating function's leftmost singularity

$$\theta^* = \sup\{\theta \in \mathbb{R} \mid \mathbf{E}[\exp(\theta S)] < \infty\},$$

which may be ∞, satisfies $\theta^* > 0$ and $\lim_{\theta \to \theta^*} \mathbf{E}[\exp(\theta S)] = \infty$.

The metric we care about is the *response time* of jobs, which is the amount of time between a job's arrival and the time it completes. Let the response time distribution under scheduling policy π be T_π, and the steady-state amount of *work*, the remaining time to complete all jobs currently in the system, be W. We wish to find the strongly tail-optimal policy among the set of policies which do not use job size information, where strong tail-optimality is in the sense of Boxma and Zwart [5], as stated in (1.1).

[3] If technical results about continuous-time Gittins indices with discounting [4, 15, 16, 19] can be extended to inflation, it would very likely extend our findings to continuous-time scheduling.
[4] Throughout, all limits in this paper are limits from below.

The key property of class I distributions for Boost remains important here. In particular, class I distributions ensure that W has asymptotically exponential tail [10, Equation (2)], so that

$$C_W = \lim_{t\to\infty} \exp(\gamma t)\mathbf{P}[W > t] = \lim_{\theta\to\gamma} \frac{\gamma - \theta}{\gamma}\mathbf{E}[\exp(\theta W)], \tag{2.1}$$

where γ is the *decay rate* and C_W is the *tail constant* of W, and it is known [20] that γ is the least positive real solution to

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

Similarly, we define the *tail constant* of π as

$$C_\pi = \lim_{t\to\infty} \exp(\gamma t)\mathbf{P}[T_\pi > t] = \lim_{\theta\to\gamma} \frac{\gamma - \theta}{\gamma}\mathbf{E}[\exp(\theta T_\pi)].$$

If S is class I, then π is weakly tail-optimal if and only if $C_\pi < \infty$, and π is strongly tail-optimal if and only if $C_\pi = \inf_{\pi'} C_{\pi'}$.

2.1 Job model

We use a discrete-time, countable-state version of the Markov-process job model [22–24]. That is, we model jobs as *absorbing discrete-time Markov chains with countable state space*, where the state of the job contains all information about the job relevant to the scheduler and only evolves while the job is in service. This structure allows us to work with scheduling policies that only learn information about a job as it receives service: a policy can use all trajectories of all jobs in the system up to their current states as input to decide which job to serve.

For simplicity, all jobs are assumed to share a state space $\mathbb{X} \cup \{x_{\text{done}}\}$ and follow the same stochastic Markovian dynamics, which are independent of the arrival process and evolution of other jobs. The initial state of a job is drawn from a distribution X_{new}, so different jobs may start in different initial states. The job completes and exits the system when it reaches the unique completion state $x_{\text{done}} \notin \mathbb{X}$.

We write X_u for the random state of a job at *age* u, meaning after u units of service. We denote a job's trajectory from age 0 to age u by

$$X_{0:u} = (X_0, X_1, \ldots, X_{u-1}, X_u).$$

The job size S is then the hitting time of the completion state,[5]

$$S = \min\{u \geq 0 : X_u = x_{\text{done}}\}.$$

A job's trajectory is defined only up to age S, and we always have $X_S = x_{\text{done}}$.

We emphasize that because the state space $\mathbb{X}$ can be countably infinite, any discrete job size distribution S can be realized by a Markov-process job model. Two canonical ways of doing so are the *known-size* and *age-based unknown-size* models:

- The *known-size* model lets a job's state be its remaining service time. Specifically: $\mathbb{X} = \mathbb{Z}_{>0}$, $x_{\text{done}} = 0$, X_{new} sampled from S, and transition dynamics are $X_{u+1} = X_u - 1$.
- The *age-based unknown-size* model lets a job's state be its age. Specifically: $\mathbb{X} = \mathbb{Z}_{\geq 0}$, x_{done} is some other point, $X_{\text{new}} = 0$, and the transition dynamics are

$$X_{u+1} = \begin{cases} x_{\text{done}} & \text{with probability } \mathbf{P}[S = X_u + 1 \mid S > X_u] \\ X_u + 1 & \text{otherwise} \end{cases}.$$

[5]We mildly abuse terminology by writing S for both the job size *distribution* and, when convenient, the *random variable* with that distribution corresponding to a generic job's random size. We do the same for other distributions throughout the paper without further comment.

That is, $X_u = u$ until the job completes, and each step of service, the chance of completion is the hazard probability of S.

Of course, there are many other Markov processes that induce the same distribution S, corresponding to different types of partial information the scheduler might have about jobs' sizes. See Scully et al. [23, Section 3.3] for some additional examples.

2.1.1 Relationship between discrete-time job model and continuous-time queueing model. While our job model is discrete-time, we emphasize that we still study a continuous-time M/G/1. That is, we assume service is slotted into discrete time units of length 1, and jobs cannot be preempted in the middle of a unit of service, but arrivals can happen at any time. There are two points worth clarifying about this:

- Our model is distinct from using a continuous-time Markov chain job model. With that said, our results would be straightforward to extend to such a model.
- Aside from discretization, there is essentially no restriction on the job size distribution S, because we allow countably many states.

Finally, we note that choice of time units being length 1 is arbitrary and without loss of generality. As such, we develop all of our formal definitions and theorems using this convention. However, in some examples (Section 3.1.1) we use shorter time units of length 0.1. Along the same lines, by applying our theory to arbitrarily small time units of length δ, one should be able to recover a continuous-time theory as $\delta \to 0$. However, a rigorous treatment of continuous-time Gittins indices introduces additional subtleties [4, 15, 16, 19], so we leave this to future work.

2.2 Boost policies

We introduce a new, trajectory-dependent variant of boost policies for Markov-process jobs. Recall that Boost is a family of policies where an instance is determined by a boost function that maps size to boost, $b\colon (0,\infty) \to [0,\infty)$. Trajectory-dependent boost functions map a job's trajectory through the state space to a boost, that is, $b\colon \mathbb{X}^{\text{traj}} \to [0,\infty]$, where $\mathbb{X}^{\text{traj}}$ is the set of finite-length trajectories of states. A boost policy with boost function b then operates in the same way as defined in Yu and Scully [34]: serve the jobs in order from least to greatest boosted arrival time. We require that $b(X_{\text{new}}) < \infty$ with probability 1.[6] Note that while size-based boost is fixed for a given job, trajectory-dependent boost functions vary a job's boost as it receives service. Thus, a trajectory-dependent boost policy preempts a job when its boosted arrival time exceeds that of another job.

Of course, a special case of trajectory-dependent boost functions are those that assign boosts based only on the current state. We call these *Markov* boost functions. Abusing notation slightly, if b is a Markov boost function, and its current state $x_u = x$, we write $b(x)$, suppressing the trajectory notation in $b(x_{0:u})$. In particular, for any arrival rate λ and job size distribution S satisfying our assumptions (namely, $\rho < 1$ and S is class I), the following family of Markov boost functions yield strongly tail-optimal policies:

Definition 2.2. Let γ be defined as in Equation (2.2). The *γ-Gittins boost* for a job in state x is

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

where $\Gamma_\gamma(x)$ is the *γ-Gittins index* of state x (Definition 3.3).[7]

[6] This makes ties probability-zero events: $\mathbb{X}^{\text{traj}}$ is countable, the arrivals are Poisson, and only one job at a time has boost ∞.

[7] The $\frac{1}{\gamma}\log\frac{e^\gamma}{e^\gamma-1}$ term ensures that $b_\gamma(x)$ is always nonnegative (see Corollary C.2).

One might wonder whether this policy can be efficiently computed for a given M/G/1 system, namely, whether computing $b_{\gamma\text{-Gittins}}(x)$ for all jobs is computationally feasible. Computing $b_{\gamma\text{-Gittins}}(x)$ amounts to computing the Gittins index of a Markov reward process, which is a standard problem in the literature. While the current state of the art is $O(n^{2.5286})$ in the number of states [8], because we model all jobs as sharing the same state space, computing the Gittins indices only needs to happen once—not for every job arrival. Therefore, we can treat it as an upfront, one-time cost, that is independent of the computational cost of *using* the policy in the M/G/1 queue. Additionally, for the canonical age-based unknown-size model with n possible job sizes, the Gittins index function can be computed in $O(n)$ time [25, Appendix B].

2.2.1 Preemption due to arrivals under boost policies. When using a boost policy, preemptions may occur because either:

- the boosted arrival time of the job currently in service increased, or
- a new job arrived into the system with a better boosted arrival time than the job currently in service.

A key observation is that, informally, preemptions of the latter variety have no effect on the tail constant, C_π. This is intuitively because the tail constant depends on analyzing the behavior of the system when there is $> w$ work in the system for any finite w, and when there is sufficient work in the system, an arriving job will not preempt the current job in service under any boost policy. This means that boost policies that behave "similarly", other than in how they preempt when a job arrives, will have the same tail constant. The correct formulation of "similar" turns out to rely on the following notion:

Definition 2.3. Let π be a boost policy with boost function b. The *lower envelope of* b is the boost function $\underline{b}(x_{0:t}) = \min_{0\le u\le t} b(x_{0:u})$.

In particular, if two boost policies have boost functions with the same lower envelope, then their tail constant is the same. This is proven in Section 4 as Corollary 4.2.

How does the lower envelope of the boost function relate to preemption due to arrivals? To see the connection, let π_b be a boost policy with boost function b, and then let:

- $\pi_{\underline{b}}$ be the boost policy corresponding to boost function $\underline{b}$, and
- π_{b^∞} be the boost policy corresponding to boost function,

$$b^\infty(x_{0:t}) = \begin{cases} b(x_{0:t}) & b(x_{0:t}) = \underline{b}(x_{0:t}) \\ \infty & \text{else} \end{cases}.$$

Note that the lower envelope of the boost function is the same for all three policies, so they have the same tail constant. Now observe that:

- If a batch of jobs were to simultaneously arrive to an empty system with no further arrivals, then π_b, $\pi_{\underline{b}}$, and π_{b^∞} would behave identically.
- If a job arrives to a system in a random state, $\pi_{\underline{b}}$ is the most likely to preempt the job currently in service, followed by π_b, and then by π_{b^∞}.

These observations motivate the following definition.

Definition 2.4. Let π be a boost policy with boost function b:
(a) The *maximally preemptive variant of* π is $\pi_{\underline{b}}$.
(b) The *minimally preemptive variant of* π is π_{b^∞}.

Our optimality proof in Section 3 makes crucial use of both the maximally and minimally preemptive variants of γ-Gittins.

3 The Gittins policy with inflation

In this section we identify a connection between the problem of minimizing the tail constant and the family of Gittins policies and then use that connection to construct an approach for proving the strong tail-optimality of γ-Gittins. We do this by considering two relaxations of the problem:

- The *arrival-free* batch setting, and
- the batch setting *with arrivals.*

Both relaxations are inspired by interpreting Equation (1.2) as telling us that minimizing the tail constant roughly corresponds to minimizing $\mathbf{E}[e^{\theta T_\pi}]$ for $\theta < \gamma$.

The arrival-free batch setting. Imagine that at time zero there is a *batch* of n jobs, $\mathcal{I} = \{(x_i, a_i)\}_{i=1}^n$ with states x_i and arrival times $a_i \leq 0$, in the system and that there are no future arrivals.

Definition 3.1.

(a) For any $\theta > 0$ and job $i \in \mathcal{I}$, define the *θ-cost of job i* under policy π to be,

$$\mathrm{Cost}_\pi^i(\theta, \mathcal{I}) = e^{\theta T_\pi^i},$$

where T_π^i is the response time of job i.

(b) For any $\theta > 0$, define the *θ-cost of batch $\mathcal{I}$* to be,

$$\mathrm{Cost}_\pi(\theta, \mathcal{I}) = \sum_{i=1}^{n} \mathrm{Cost}_\pi^i(\theta, \mathcal{I}) = \sum_{i=1}^{n} e^{\theta D_\pi^i} e^{-\theta a_i}, \tag{3.1}$$

where D_π^i is the departure time of job i (the time at which the job completes and leaves the system).

We aim to find the (likely θ-dependent) policy π that minimizes the mean θ-cost of batch $\mathcal{I}$.

The batch setting with arrivals. Now imagine that we would still like to minimize the expected θ-cost of the batch, but that a subset of the jobs have arrival time $a_i > 0$. In this setting, we only consider "non-anticipating" policies, i.e. the policy does not know when, or even how many, jobs will arrive in the future.

As we will see later in this section, the arrival-free batch setting is both simple enough that there exists an optimal index policy and similar enough to the M/G/1 queue setting to provide insight into the behavior of a tail-optimal policy. On the other hand, the batch setting with arrivals is closer to the M/G/1 queue setting but too complicated to admit a simple optimal policy. However, understanding what parts of the index policy's optimality proof breaks down in the presence of arrivals will inform our approach in the M/G/1 queue setting.

The key observation for the arrival-free batch setting is that the solution follows from the classical Gittins index theory for multi-armed bandits except for one twist: we must minimize inflated cost instead of maximizing discounted reward. To see this, we reinterpret (3.1) as the expected total cost of a multi-armed bandit under policy π where:

- each job is an arm of the bandit,
- we pay a cost $e^{-\theta a_i}$ when job i completes,
- costs inflate at rate θ, meaning that a cost paid at time t is multiplied by a factor of $e^{\theta t}$.

It is a standard result that the Gittins index policy maximizes rewards for multi-armed bandits with discounted rewards [9]. However, to the best of our knowledge, there is no study of the Gittins index policy for multi-armed bandits with inflated costs in the literature. We close this gap in Appendix D by adapting the optimality proof of Weber [29] to inflation. The key difference from the original proof is that the decision maker *minimizes* the expected total-inflated cost by *maximizing* the expected rate of incurred cost. This is flipped from the discounted-reward setting where the decision maker *maximizes* the expected total-discounted reward by *maximizing* the expected rate of reward.

In the remainder of this section we show how the key idea from the classic optimality proof of Weber [29] can be used in a new way to prove the tail optimality of a Gittins policy in the M/G/1 queue setting. In particular, we will:

- (Section 3.1) define the θ-Gittins family of boost policies, which contains the tail-optimal policy for the M/G/1 queue setting: γ-Gittins.
- (Section 3.2) prove that θ-Gittins is optimal in the arrival-free batch setting. We do this by using the concept of *surrogate costs* (borrowed from Weber [29]). Note that this section's presentation focuses on the specific setting of a batch of jobs, while Appendix D covers general multi-armed bandits with inflation and contains complete proofs.
- (Section 3.3) discuss the obstacles that arise in trying to use surrogate costs in the batch setting with arrivals, as well as why we expect these obstacles to go away in the M/G/1 queue setting.
- (Section 3.4) layout our approach to proving the strong tail optimality of γ-Gittins using a new, quantitative analysis of surrogate costs (presented in Section 4).

3.1 The θ-Gittins policy

In this section we define the θ-Gittins index and the θ-Gittins policy for all $\theta > 0$. We then consider an example to understand the behavior of the strongly tail-optimal policy, γ-Gittins, which is exactly the θ-Gittins policy when $\theta = \gamma$ and γ is defined as in (2.2). Since the idea behind the θ-Gittins policy is the same as the classical Gittins policy for multi-armed bandits, we keep the definition brief, and then provide examples of the policy's behavior in the M/G/1 queue setting. For more intuition on the Gittins index, see the proof in Appendix D or the classic paper by Weber [29].

Definition 3.2. For all $x \in \mathbb{X}$, $a \in \mathbb{R}$, and $\mathbb{Y} \subseteq \mathbb{X}$, define the following distributions:

$$\begin{aligned}
\mathrm{Service}(x, \mathbb{Y}) &= (\text{service needed for a job starting at state } x \text{ to exit } \mathbb{Y}),\\
\mathrm{Completed}(x, \mathbb{Y}) &= \mathbb{1}(\text{job starting at state } x \text{ is at } x_{\mathrm{done}} \text{ after exiting } \mathbb{Y}),\\
\mathrm{InflatedCost}(\theta, x, a, \mathbb{Y}) &= e^{-\theta a} \cdot e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} \cdot \mathrm{Completed}(x, \mathbb{Y}),\\
\mathrm{InflatedTime}(\theta, x, \mathbb{Y}) &= \sum_{t=1}^{\mathrm{Service}(x,\mathbb{Y})} e^{\theta t} = \frac{e^{\theta}}{e^{\theta} - 1}(e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1).
\end{aligned}$$

Definition 3.3.

(a) The *θ-Gittins index* of a job at state $x \in \mathbb{X}$ with arrival time a is

$$\Gamma_\theta(x, a) = \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}[\mathrm{InflatedCost}(\theta, x, a, \mathbb{Y})]}{\mathbf{E}[\mathrm{InflatedTime}(\theta, x, \mathbb{Y})]}.$$

For convenience, let $\Gamma_\theta(x) = \Gamma_\theta(x, 0)$ so that $\Gamma_\theta(x, a) = e^{-\theta a}\Gamma_\theta(x)$.

(b) The θ-Gittins policy is the policy that always serves the job with greatest θ-Gittins index.

Note that θ-Gittins is a Markov boost policy with boost function,

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

This can be seen by observing that $a - b_{\theta\text{-Gittins}}(x)$ is a monotonic function of $\Gamma(x, a)$, namely, it is $1/\theta$ times the negative log of $\Gamma(x, a)$ plus a constant. The constant term, $\frac{1}{\theta}\log(\frac{e^{\theta}}{e^{\theta}-1})$ is there to ensure that $b_{\theta\text{-Gittins}} \geq 0$, see Corollary C.2.

3.1.1 How does the γ-Gittins policy behave in the M/G/1 queue setting? To better understand the behavior of the γ-Gittins policy, we consider its behavior in an M/G/1 queue where:

- jobs have size distribution Unif$\{1/7, 6/7, 14/7\}$,[8]
- service is slotted into discrete units of length 0.1/7,
- the arrival rate is $\lambda = 0.8$, resulting in a load of $\rho = 0.8$.

It is straightforward to verify that for these system parameters, $\gamma \approx 0.266$, so we wish to compute the 0.266-Gittins policy.[9] However, to do so, we must specify the underlying Markov-process job model. We will compute and then compare the policy for the following job models:

- the age-based unknown-size model: the job's state contains only its age,
- the age-based unknown-size model with size information at age 1/7: the job's state always contains its age, but also contains the job's size for ages greater than or equal to 1/7.

In Fig. 3.1 we see the boosted arrival time of a job with arrival time zero under the 0.266-Gittins policy, where Fig. 3.1(a) is with the age-based unknown-size model, and Fig. 3.1(b) is with the added size information at age 1/7. Observe that for the former, the boosted arrival time strictly decreases (i.e. the job's priority increases) except for jumps at ages 1/7 and 6/7, while for the latter there is only a jump at age 1/7 since afterwards the job size is known. Fig. 3.1(c) shows how the policy interleaves service under the age-based unknown-size model when there are two jobs in the system at the same time.

3.2 Main ideas from the Gittins theory in the arrival-free batch setting

The primary challenge in minimizing the cost of an arrival-free batch is that the timing of the costs is unpredictable, as we do not know when a job will complete. The key idea of the Gittins theory is to make the costs more predictable by "smoothing them out": rather than paying a cost only when a job completes, we imagine paying a "surrogate" cost every time step.

Specifically, we define surrogate costs such that the following properties hold for all policies π:

$$\text{total surrogate cost of Gittins} \leq \text{total surrogate cost of policy } \pi$$
$$\mathbf{E}[\text{total surrogate cost of policy } \pi] \leq \mathbf{E}[\text{total real cost of policy } \pi]$$
$$\mathbf{E}[\text{total surrogate cost of Gittins}] = \mathbf{E}[\text{total real cost of Gittins}].$$

Combining these implies, as desired,

$$\mathbf{E}[\text{real cost of Gittins}] \leq \mathbf{E}[\text{real cost of policy } \pi].$$

How does one define surrogate costs with these properties?

Definition 3.4.

(a) Let the *surrogate θ-price* of a job be the minimum Gittins index it has had so far.
(b) At the end of each time step, we pay a *surrogate θ-cost* equal to the surrogate θ-price of whichever job was served (subject to inflation).[10]
(c) Let the *surrogate θ-cost of job i*, $\widehat{\mathrm{Cost}}^{i}_{\pi}(\theta, \mathcal{I})$, be the total surrogate θ-cost paid while serving job i.
(d) Let the *surrogate θ-cost of a batch $\mathcal{I}$*, $\widehat{\mathrm{Cost}}_{\pi}(\theta, \mathcal{I})$, be the total surrogate θ-cost of all jobs, i.e., the total surrogate θ-cost paid over all time steps.

Defined in this way, surrogate θ-costs have exactly the properties we wanted, as proven by the following two results.

[8] We refer to this as the 1-6-14 job size distribution but divide by 7 to normalize the mean.
[9] Note that in practice, rounding γ reduces performance, even if only slightly (see Section 5.4).
[10] We use the convention that surrogate θ-cost is paid at the end of the time step because real cost is paid when a job completes, i.e. at the end of a time step.

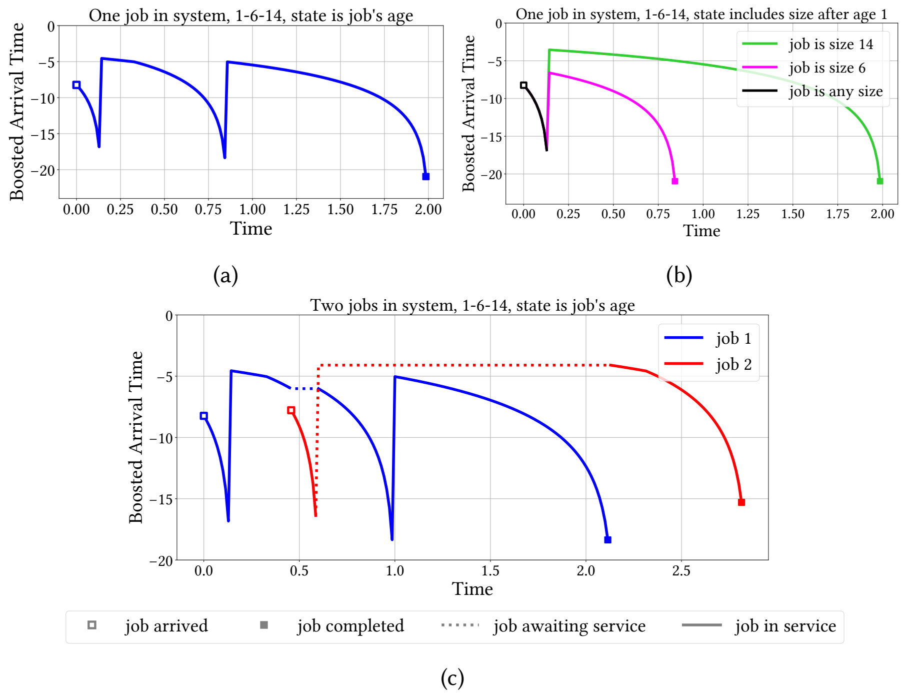

Fig. 3.1. The behavior of the 0.266-Gittins policy in an M/G/1 queue. (a) and (b) show the boosted arrival time of a job with size distribution Unif$\{1/7, 6/7, 14/7\}$ (we divide by 7 to normalize the mean) and arrival time zero as it attains service. In (a), the job's state contains only the job's age, while in (b), the job's state contains its age and, starting at age $1/7$, the job's size. Returning to the setting where the state contains only the age, (c) shows how the policy interleaves service when there are two jobs in the system: one of size $14/7$ and one of size $6/7$.

Proposition 3.5. *For any $\theta > 0$, job $i \in \mathcal{I}$, and policy π,*

$$\mathbf{E}[\mathrm{Cost}_\pi^i(\theta, \mathcal{I})] \geq \mathbf{E}[\widehat{\mathrm{Cost}}_\pi^i(\theta, \mathcal{I})],$$

*with equality if and only if π is θ-*insulated*. A policy is θ-insulated if it never preempts a job whose θ-Gittins index is different than its surrogate θ-price.*

Proof. This is a special case of Lemma D.3, which is the same statement for a single arm of a general multi-armed bandit with inflation. □

Lemma 3.6. *In the arrival-free batch setting,*
(a) θ-Gittins sample-path-wise minimizes surrogate θ-cost.
(b) θ-Gittins is θ-insulated.

Proof. Observe that for each job in the batch, the job's surrogate θ-prices form a non-increasing sequence that is independent of the policy. Thus, choosing a policy equates to picking an interleaving of these sequences. Since the sequence of surrogate θ-costs incurred will be the time-inflated interleaving of the surrogate θ-prices, a straightforward interchange argument (or equivalently, the rearrangement inequality [12, Section 10.2]) tells us that the policy which always serves the job with greatest surrogate θ-price will minimize the total surrogate θ-cost.

Now consider θ-Gittins in the arrival free batch setting. Initially, all jobs have surrogate θ-price equal to their θ-Gittins index. Since θ-Gittins serves the job with greatest θ-Gittins index, it will also initially serve the job with greatest surrogate θ-price. Now imagine θ-Gittins switches to serving a new job. This means that:

- This is the first ever time step where the original job's θ-Gittins index is less than the current θ-Gittins index of the new job, since otherwise the policy would have switched previously.
- Thus, the first job's surrogate θ-price is equal to its new θ-Gittins index, and so is less than the surrogate θ-price of the new job.
- θ-Gittins is still serving the job with greatest surrogate θ-price.

It is straightforward to check that this argument holds every time the policy switches to serving a new job, so θ-Gittins always serves the job with greatest surrogate θ-price, and thus sample-path-wise minimizes the total surrogate θ-cost.

The above argument also shows that θ-Gittins only stops serving a job when its surrogate θ-price is equal to its θ-Gittins index, which means that θ-Gittins is θ-insulated. □

The optimality of θ-Gittins in the arrival-free batch setting follows directly from these properties.

Theorem 3.7. *For any $\theta > 0$ and batch $\mathcal{I}$, θ-Gittins minimizes the expected θ-cost of $\mathcal{I}$. That is, for all policies π,*

$$\mathbf{E}[\mathrm{Cost}_\pi(\theta, \mathcal{I})] \geq \mathbf{E}[\mathrm{Cost}_{\theta\text{-Gittins}}(\theta, \mathcal{I})].$$

Proof. Observe that since Proposition 3.5 holds for each job in the batch, by linearity of expectation,

$$\mathbf{E}[\mathrm{Cost}_\pi(\theta, \mathcal{I})] \geq \mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta, \mathcal{I})].$$

Since θ-Gittins sample-path-wise minimizes surrogate θ-cost, it must also minimize the mean surrogate θ-cost. Thus,

$$\mathbf{E}[\mathrm{Cost}_\pi(\theta, \mathcal{I})] \geq \mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta, \mathcal{I})] \geq \mathbf{E}[\widehat{\mathrm{Cost}}_{\theta\text{-Gittins}}(\theta, \mathcal{I})] = \mathbf{E}[\mathrm{Cost}_{\theta\text{-Gittins}}(\theta, \mathcal{I})],$$

where the last equality holds because θ-Gittins is θ-insulated. □

3.3 What breaks down in the batch setting with arrivals?

The objective in the batch setting with arrivals corresponds to minimizing the expected total-inflated cost of a multi-armed bandit with arms that arrive over time. The Gittins index policy is known to be optimal in the presence of arrivals *if the arrivals are time-homogeneous*, that is, if each arriving arm is drawn from the same distribution independent of arrival time [30, 31].[11] However, in the batch setting with arrivals, the arrivals are time-inhomogeneous since their costs $e^{-\theta a_i}$ are a function of their arrival times a_i, so there is no reason to think that the Gittins index policy will be optimal. Our goal in this section is not to identify the optimal policy for this setting, but rather to understand where the proof of optimality in Section 3.2 breaks down due to arrivals.

The key problem is that both parts of Lemma 3.6 no longer hold: when there are arrivals, θ-Gittins may not minimize surrogate θ-cost and may not be θ-insulated. This can be seen in the below example or by considering Fig. 3.1(c), which is effectively a two job batch with arrivals.

Example 3.8. Job (A) enters an empty system at time $t = 0$ with θ-Gittins index 1 and surrogate θ-price 1 and is served. After one unit of service, job (A) has θ-Gittins index 4 and surrogate θ-price 1. At time $t = 1$, job (B) enters the system with θ-Gittins index 3 and surrogate θ-price 3. Since

[11] Moreover, in settings with time-homogeneous arrivals and discounting, the Gittins index computation becomes more complicated, and its value depends on the arrival rate and initial state distribution of new arms. The famous case of minimizing mean weighted delay in the M/G/1 queue retains the relatively simple arrival-insensitive Gittins index due to its lack of discounting [9, 24].

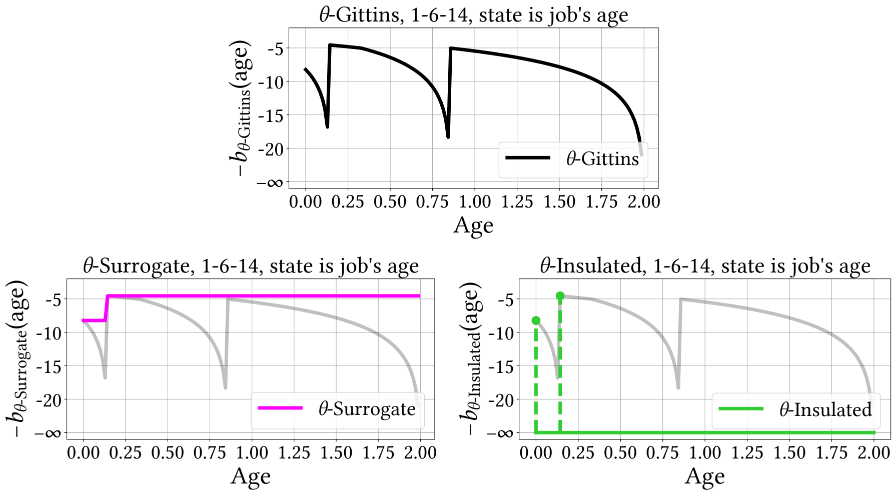

Fig. 3.2. The boost received by a job under the age-based unknown-size job model for the 0.266-Gittins, 0.266-Surrogate, and 0.266-Insulated policies. The job size distribution is Unif$\{1/7, 6/7, 14/7\}$, and for each policy we plot the boost a job would receive ($-b(x)$) against its age x.

the θ-Gittins index of job (A) is greater than the θ-Gittins index of job (B), θ-Gittins serves job (A). After one more unit of service, job (A) has θ-Gittins index 2 and surrogate θ-price 1. Now, job (B) has the greater θ-Gittins index, so θ-Gittins preempts job (A) and serves job (B).

θ-Gittins is not θ-insulated since job (A) is preempted at time $t = 3$ even though its θ-Gittins index is not equal to its surrogate θ-price. Additionally, at time $t = 2$, θ-Gittins serves job (A) even though its surrogate θ-price is below that of job (B), so θ-Gittins does not serve the job with greatest surrogate θ-price.

Thus, in settings with arrivals it is helpful to consider two new policies, which we define using our variants of boost policies from Definition 2.4:

- θ-Surrogate: the maximally preemptive variant of θ-Gittins,
- θ-Insulated: the minimally preemptive variant of θ-Gittins.

It follows directly from the definitions that θ-Surrogate always serves the job of maximum surrogate θ-price and θ-Insulated assigns the job in service a boost of infinity if its θ-Gittins index is not equal to its surrogate θ-price. Thus, θ-Insulated never preempts a job except for when its θ-Gittins index equals its surrogate θ-price and otherwise always serves the job of greatest θ-Gittins index. Thus, by reasoning similar to the proof of Lemma 3.6, we get an analogous result in the batch setting with arrivals:

Lemma 3.9. *In the batch setting with arrivals,*

- *θ-Surrogate sample-path-wise minimizes surrogate θ-cost.*
- *θ-Insulated is θ-insulated.*

Thus, the expected surrogate θ-cost of θ-Surrogate provides a lower bound on the expected real θ-cost of the optimal policy, and θ-Insulated has the same expected real θ-cost and expected surrogate θ-cost. In the arrival-free batch setting, all three of these policies are equivalent, so θ-Gittins has both properties and optimality quickly follows. On the other hand, in the batch setting with arrivals, θ-Gittins has neither property and so we should not expect it to minimize real θ-cost.

At this point, it may seem like this surrogate cost approach is doomed to fail in any setting with arrivals. However, recall our intuition from Section 2.2.1 that, *when it comes to tail behavior of response time in the M/G/1 queue setting, we can pretend that there are no preemptions due to arrivals.* Thus, the hope is that it will be sufficient for the proof in the M/G/1 queue setting that our three policies have the same tail behavior of response time, even though the policies are not themselves equivalent.

3.4 Strong tail-optimality of γ-Gittins in the M/G/1 queue setting

We now consider the problem of minimizing the tail constant, $C_\pi = \lim_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\pi}]$, in the M/G/1 queue setting. Just as when we switched from the arrival-free batch setting to the batch setting with arrivals, the new complexity added by this setting leads to new obstacles. In the batch settings we considered the performance of θ-Gittins with respect to minimizing the mean θ-cost of a batch. In the M/G/1 queue setting, we roughly want to minimize mean γ-cost,"$\mathbf{E}[e^{\gamma T_\pi}]$"—however, γ-cost is infinite under any policy. Thus, to reason about C_π, we must instead reason about θ-cost and then let $\theta \to \gamma$. This is difficult because for any fixed $\theta < \gamma$, we should not expect γ-Gittins to minimize mean θ-cost, even if we ignore the complications due to arrivals discussed in Section 3.3. We therefore need to show that the optimality gap for γ-Gittins's performance *with regards to θ-cost* becomes negligible as $\theta \to \gamma$. But this quantitative statement about θ-cost is beyond the scope of the qualitative proof using surrogate costs presented Section 3.2. Thus, our approach in the M/G/1 queue setting will rely on a quantitative analysis that characterizes the performance of arbitrary boost policies with regards to both real and surrogate θ-cost as $\theta \to \gamma$ (Section 4). To facilitate reasoning about these quantities, we introduce the following notation, which is specific to the M/G/1 queue setting.

Definition 3.10.

(a) Let $\mathrm{Cost}_\pi(\theta) = e^{\theta T_\pi}$ so that $C_\pi = \lim_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_\pi(\theta)]$.

(b) Let $\widehat{\mathrm{Cost}}_\pi(\theta)$ be a random variable distributed as total surrogate θ-cost paid while serving a job in the M/G/1 queue under policy π.

(c) The *surrogate tail constant* of policy π is $\widehat{C}_\pi = \lim_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta)]$.

(d) $C_\pi(\theta) = \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_\pi(\theta)]$.

(e) $\widehat{C}_\pi(\theta) = \frac{\gamma-\theta}{\gamma}\mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta)]$.

Using this notation, $C_\pi = \lim_{\theta\to\gamma} C_\pi(\theta)$ and $\widehat{C}_\pi = \lim_{\theta\to\gamma} \widehat{C}_\pi(\theta)$.

Using our quantitative analysis, we will prove the following result, formalizing our intuition that preemptions due to arrivals do not affect the tail behavior of response time.

Corollary 4.2. *If two boost policies π and π' have boost functions with the same lower envelope, then $C_\pi = C_{\pi'}$ and $\widehat{C}_\pi = \widehat{C}_{\pi'}$.*

Additionally, our quantitative analysis will prove the following continuity result, which lets us reason about the real and surrogate θ-cost of θ-Gittins instead of γ-Gittins as $\theta \to \gamma$.

Proposition 4.3. *For θ-Policy $\in \{\theta$-Gittins, θ-Surrogate, θ-Insulated$\}$,*

$$\lim_{\theta\to\gamma} |C_{\gamma\text{-Policy}}(\theta) - C_{\theta\text{-Policy}}(\theta)| = 0, \tag{4.2}$$

$$\lim_{\theta\to\gamma} |\widehat{C}_{\gamma\text{-Policy}}(\theta) - \widehat{C}_{\theta\text{-Policy}}(\theta)| = 0. \tag{4.3}$$

Once we have these results in hand, the optimality of γ-Gittins follows from a proof very similar to the one presented in Section 3.2. First, we need an M/G/1 queue analogue of Proposition 3.5:

Proposition 3.11. *For any policy π and $\gamma > \theta > 0$,*

$$\mathbf{E}[\mathrm{Cost}_\pi(\theta)] \geq \mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta)],$$

*with equality if and only if π is θ-*insulated. *A policy is θ-insulated if it never preempts a job whose θ-Gittins index is different than its surrogate θ-price.*

Proof. This follows from using a "tagged job" analysis [11] and then applying Lemma D.3, which is the same statement for a single arm of a general multi-armed bandit with inflation. □

We also need the below result, which follows from thinking of each busy period in the M/G/1 queue as a batch with arrivals and then applying Lemma 3.9.

Lemma 3.12. *For any policy π and $\gamma > \theta > 0$,*

$$\mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta)] \geq \mathbf{E}[\widehat{\mathrm{Cost}}_{\theta\text{-Surrogate}}(\theta)].$$

Proof. Since θ-Surrogate always serves the job with greatest surrogate θ-price, it sample-path-wise minimizes the surrogate θ-cost of every busy period. By generalized Little's law [6, 13, 18] and renewal-reward theorem [11], sample-path-wise minimizing the surrogate θ-cost of each busy period also minimizes $\mathbf{E}[\widehat{\mathrm{Cost}}_\pi(\theta)]$. □

Theorem 3.13. *For all policies π,*

$$C_\pi \geq C_{\gamma\text{-Gittins}} = C_{\gamma\text{-Surrogate}} = C_{\gamma\text{-Insulated}}.$$

In particular, γ-Gittins is a strongly tail-optimal policy in the M/G/1 queue setting.

Proof. Note that by Corollary 4.2 we need only prove this for $C_{\gamma\text{-Gittins}}$. Let $\{\pi_n^*\}_{n=1}^\infty$ be a sequence of policies such that $\lim_{n\to\infty} C_{\pi_n^*} = \liminf_\pi C_\pi$. We would like to show that

$$\lim_{\theta\to\gamma} |C_{\gamma\text{-Gittins}}(\theta) - \lim_{n\to\infty} C_{\pi_n^*}(\theta)| = 0.$$

The first step is to use the surrogate θ-cost of θ-Surrogate as a lower bound on the optimal real θ-cost:

$$C_{\pi_n^*}(\theta) \geq \widehat{C}_{\pi_n^*}(\theta) \geq \widehat{C}_{\theta\text{-Surrogate}}(\theta).$$

This follows from Proposition 3.11 and Lemma 3.12. The second step is to use the θ-Insulated policy as a "bridge" between real and surrogate θ-cost:

$$\begin{aligned}
&|C_{\gamma\text{-Gittins}}(\theta) - \widehat{C}_{\theta\text{-Surrogate}}(\theta)| \\
&\quad \leq |C_{\gamma\text{-Gittins}}(\theta) - C_{\theta\text{-Gittins}}(\theta)| + |C_{\theta\text{-Gittins}}(\theta) - C_{\theta\text{-Insulated}}(\theta)| \\
&\quad\quad + |C_{\theta\text{-Insulated}}(\theta) - \widehat{C}_{\theta\text{-Insulated}}(\theta)| + |\widehat{C}_{\theta\text{-Insulated}}(\theta) - \widehat{C}_{\theta\text{-Surrogate}}(\theta)|.
\end{aligned}$$

In particular, observe that $|C_{\theta\text{-Insulated}}(\theta) - \widehat{C}_{\theta\text{-Insulated}}(\theta)| = 0$ by Proposition 3.11 and the fact that θ-Insulated is a θ-insulated policy. Now, as $\theta \to \gamma$,

- $|C_{\gamma\text{-Gittins}}(\theta) - C_{\theta\text{-Gittins}}(\theta)| \to 0$ by Proposition 4.3,
- $|C_{\theta\text{-Gittins}}(\theta) - C_{\theta\text{-Insulated}}(\theta)| \to 0$ by Corollary 4.2,
- $|\widehat{C}_{\theta\text{-Insulated}}(\theta) - \widehat{C}_{\theta\text{-Surrogate}}(\theta)| \to 0$ by Corollary 4.2,

so

$$\lim_{\theta\to\gamma} |C_{\gamma\text{-Gittins}}(\theta) - \widehat{C}_{\theta\text{-Surrogate}}(\theta)| = 0,$$

and γ-Gittins is strongly tail-optimal in the M/G/1 queue setting. □

4 Analysis of real and surrogate tail constants for general boost policies

In this section, we provide a characterization of both the tail and surrogate tail constants of trajectory-dependent boost policies. Our approach follows the general structure of Yu and Scully [34], adapted to account for the more general class of boost policies. The main result of this section (Theorem 4.1) is a characterization of the tail and surrogate tail constants for arbitrary boost policies:

Theorem 4.1. *For any boost policy π with boost function b,*

$$C_\pi = C_W \underbrace{\mathbf{E}[e^{\gamma(S-\underline{b}(X_{0:S-1}))}]}_{\text{“size minus boost”}} \underbrace{\exp\left(\int_0^\infty \lambda \mathbf{E}[e^{\gamma S[\underline{b}>t]} - 1]\,\mathrm{d}t\right)}_{\text{“crossing work”}},$$

where $\underline{b}$ is the lower envelope of the boost function b, and $S[\underline{b} > t] = \min\{z \geq 0 : X_z = x_{\text{done}}$ or $\underline{b}(X_{0:z}) \leq t\}$ is the random amount of service a job requires to either complete or stop having (lower envelope of) boost greater than t. Similarly,

$$\widehat{C}_\pi = C_W \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_\gamma(X_{0:i-1}) e^{\gamma(i-\underline{b}(X_{0:i-1}))}\right] \exp\left(\int_0^\infty \lambda \mathbf{E}[e^{\gamma S[\underline{b}>t]} - 1]\,\mathrm{d}t\right),$$

where $\underline{\Gamma}_\gamma(x_{0:z}) = \min_{x \in x_{0:z}} \Gamma_\gamma(x)$ is the surrogate γ-price of a job with trajectory $x_{0:z}$.

Proof. This follows directly from Theorems 4.11 and A.7 and Lemma 4.6, which are discussed in the rest of this section. □

Full proofs of all results are given in Appendix A. We can intuitively interpret Theorem 4.1 by noticing that the two highlighted terms capture a tradeoff that must be balanced by the boost function.

- The “size minus boost” term quantifies how much a job benefits from its own boost. For example, if the boost were always 0, this would be simply $\mathbf{E}[e^{\gamma S}]$.
- The “crossing work” term quantifies how much a job is delayed by other jobs' boosts. For example, if the boost were always 0, this would simply be 1.

The key idea behind Theorem 4.1 is that when the work in system W is very large, the response time of a job arriving into the system is roughly,

$$T \approx W + (S - B) + V, \tag{4.1}$$

where B is the *worst-ever* boost of the job, and V is the “crossing work”, i.e. all the work with priority over the job that has yet to arrive in the system. We lay out our approach to proving this idea in Section 4.1 and then provide a rigorous statement in Lemma 4.9.

Theorem 4.1 provides a direct path to proving the results we needed for γ-Gittins optimality in Section 3.4. In particular, since Theorem 4.1 characterizes C_π and $\widehat{C}_\pi$ in terms of only the lower envelope of the boost function, it follows immediately that:

Corollary 4.2. *If two boost policies π and π' have boost functions with the same lower envelope, then $C_\pi = C_{\pi'}$ and $\widehat{C}_\pi = \widehat{C}_{\pi'}$.*

Thus, all that is left is to prove:

Proposition 4.3. *For θ-Policy $\in \{\theta\text{-Gittins}, \theta\text{-Surrogate}, \theta\text{-Insulated}\}$,*

$$\lim_{\theta \to \gamma} |C_{\gamma\text{-Policy}}(\theta) - C_{\theta\text{-Policy}}(\theta)| = 0, \tag{4.2}$$

$$\lim_{\theta \to \gamma} |\widehat{C}_{\gamma\text{-Policy}}(\theta) - \widehat{C}_{\theta\text{-Policy}}(\theta)| = 0. \tag{4.3}$$

Thankfully, the proof is almost identical to that of Theorem 4.1, so we defer it to Appendix B. The main difference is that we now have θ-dependent policies and thus θ-dependent bounds on response time. This is easy to deal with using the following facts, which are also proven in the appendix:

- the "crossing work" is non-increasing in θ (Lemma C.3 and the fact that reducing the boosts of "crossing work" jobs reduces the amount of crossing work),
- $\underline{b_{\theta\text{-Gittins}}}$ (the lower envelope of $b_{\theta\text{-Gittins}}$) is left-continuous and non-increasing in θ (Lemmas C.3 and C.4).

The remainder of this section presents the main ideas needed for proving Theorem 4.1 (and thus also Proposition 4.3).

4.1 Approach: pessimistic tagged job analysis

We formalize the intuition in (4.1) using a tagged job analysis in a stationary M/G/1 system. Without loss of generality, we assume that the arrival time of the tagged job is zero. The tagged job analysis consists of three main steps:

(1) Compute the amount of work in the system at the boosted arrival time of the tagged job. All this work has priority over the tagged job.
(2) Lower bound the amount of work with priority over the tagged job that arrives after the tagged job's boosted arrival time.
(3) Upper bound the amount of work with priority over the tagged job that arrives after the tagged job's boosted arrival time.

In their analysis of known-size boost policies, Yu and Scully [34] leveraged the fact that the priority order between the tagged job and other work in the system was static: if the tagged job had priority over some work, that work could be ignored when computing the tagged job's response time. With trajectory-dependent boost functions, this is no longer the case. As the tagged job is serviced, its boost, and thus its priority, may decrease. This means that, unlike in the known-size boost setting, the priority order between the tagged job and other jobs changes with service.

Since the priority of the tagged job over other work in the system depends on its trajectory through the state space, how should we analyze it? The answer it to use the pessimism principle applied by Scully et al. [26] to analyze rank-based scheduling policies. We find that the *worst-ever boost* of the tagged job determines how much work is prioritized over it before its departure.

In the rest of this section, we combine the pessimism principle with the tagged job approach of Yu and Scully [34] to derive response time bounds for arbitrary trajectory-dependent boost policies. We then use these bounds to characterize the tail constants.

4.2 Bounding response times with crossing work

For boost policies, there are several sources of work that will be relevant to the tagged job's response time. These sources of work roughly involve (1) jobs that are already in the system at the boosted arrival time of the tagged job, and (2) work that arrives after the boosted arrival time but is boosted in front of our job. We define quantities below which will allow us to reason about work relative to the boosted arrival time of the tagged job. Throughout, we consider an arbitrary boost policy π with boost function b, which we leave implicit in our notation.

Definition 4.4.

(a) The *u-crossing work* of a job is the first age at which its boosted arrival time is after u. If this never occurs, the u-crossing work is the size of the job.
(b) The *u-non-crossing work* of a job is its size minus its u-crossing work.

(c) The *crossing work* $V(u,v)$ (or $V_\pi(u,v)$ when we wish π to be explicit) is the the sum of u-crossing work of each job that arrives in the system after time u and up to time $u+v$.
(d) The *non-crossing work*, $\overline{V}(u,v)$ is the the sum of u-non-crossing work of each job that arrives in the system after time u and up to time $u+v$. Equivalently, this is the quantity of work that arrives in the system after time u and up to time $u+v$ minus the crossing work $V(u,v)$.
(e) The *existing work* $W(u)$, is the amount of work in the system at time u, excluding any work that arrived at time u.

Observation 4.5. $V(u,v)$, $\overline{V}(u,v)$, and $W(u)$ are stationary with respect to u.

The following lemma provides an expression for the transform of crossing work:

Lemma 4.6. *For any boost policy π with boost function b, any $u \in \mathbb{R}$, and any $v \in [0,\infty]$,*

$$\mathbf{E}[e^{\gamma V(u,v)}] = \exp\left(\int_0^v \lambda \mathbf{E}[e^{\gamma S[\underline{b}>t]} - 1]\,\mathrm{d}t\right),$$

where $S[\underline{b} > t] = \min\{z \geq 0 : X_z = x_{\mathrm{done}} \text{ or } \underline{b}(X_{0:z}) \leq t\}$ is the random amount of service a job requires to either complete or stop having (lower envelope of) boost greater than t.

The full proof is given in Appendix A. For our analysis, we require the transform of the crossing work, $\mathbf{E}[e^{\gamma V(0,\infty)}]$, to be finite. We show that this transform is indeed bounded if the boost function is bounded. This is far from a necessary condition, but it holds for $b_{\gamma\text{-Gittins}}$ (Corollary C.2), so we leave the task of deriving more general sufficient conditions to future work.

Lemma 4.7. *For any boost policy with bounded boost function, $\mathbf{E}[e^{\gamma V(0,\infty)}] < \infty$.*

Proof. Suppose that the boost function is bounded by some $m > 0$. Then any jobs with arrival time after m will have no 0-crossing work. Therefore, $V(0,\infty)$ is bounded by the total amount of work that arrives within the interval $(0,m)$, and so it suffices to show that the transform of this work is bounded. Call this work R. The arrivals have distribution S and arrival rate λ, so

$$\mathbf{E}[e^{\gamma R}] = e^{\lambda m(\mathbf{E}[e^{\gamma S}]-1)} = e^{\gamma m} < \infty,$$

by a standard M/G/1 result (see, e.g. [11, Chapter 25.6]) and the definition of γ in (2.2). □

Having characterized crossing work, we now move to bounding response time in terms of crossing work. As discussed in Section 4.1, a job's response time depends critically on the worst boost it has during its time in the system.

Definition 4.8. Let $B = \underline{b}(X_{0:S-1})$ (or B_π when we wish π to be explicit) be the *worst ever boost* of the tagged job.

We now have the tools we need to formally capture the idea that $T \approx W + (S - B) + V$ when the work W is large. Specifically, we find bounds on T for any boost policy π (we write T_π when we wish π to be explicit) which are structurally similar to the known-size Boost policy bounds [34], but require reasoning about work that has priority over the tagged job with its worst-ever boost B.

Lemma 4.9. *Let $\check{B} = \min(B,m)$ for any fixed $m \geq 0$. Then for all boost policies,*

$$T \geq W(-B) + V(-B, W(-B)) + S - B,$$
$$T \leq \max\{W(-\check{B}), \check{B}\} + V(-\check{B},\infty) + \overline{V}(-\check{B},m)\mathbb{1}(W(-\check{B}) < \check{B}) + S - \check{B}.$$

We defer the proof to Appendix A, as it is very similar to an analogous proof in Yu and Scully [34, Lemma 3.3]. Choosing $m = 0$ in Lemma 4.9 gives a simple upper bound on the tagged job's response time.

Corollary 4.10. *For all boost policies, $T \leq W(0) + V(0,\infty) + S$.*

4.3 Characterizing tail constants for boost policies

THEOREM 4.11. *For any boost policy π, we have $C_\pi = C_W \mathbf{E}[e^{\gamma V_\pi(0,\infty)}]\mathbf{E}[e^{\gamma(S-B_\pi)}]$.*

PROOF SKETCH. The proof follows essentially the same steps as the Boost analysis [34, Section 3] but relies on the bounds in Lemma 4.9. A full proof is provided in Appendix A for completeness. The first step is to use Lemma 4.9 to obtain the following lower and upper bounds on $C = C_\pi$:

$$C \geq \liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}\Big[\exp\Big(\theta\big(W(-B) + V(-B, W(-B)) + S - B\big)\Big)\Big],$$
$$C \leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}\Big[\exp\Big(\theta\big(\max\{W(-\check{B}), \check{B}\} + V(-\check{B}, \infty) + \overline{V}(-\check{B}, m)\mathbb{1}(W(-\check{B}) < \check{B}) + S - \check{B}\big)\Big)\Big].$$

Roughly speaking, the reason that both right-hand sides converge to the same value as $\theta \longrightarrow \gamma$ is that the only term making the expectations infinite is the existing work. The lower bound follows quickly from this observation: $W(-B)$ is independent of the arriving job's trajectory and the arrivals after time $-B$, so we can factor a $\mathbf{E}[e^{\theta W(-B)}]$ out of the expectation. The upper bound is a bit more involved, as there is not a similarly clean factorization. But the fact that $\mathbf{E}[e^{\theta W(-\check{B})}]$ diverges as $\theta \longrightarrow \gamma$ means that we can essentially ignore terms that are only nonzero when $W(-\check{B})$ is below any finite threshold. As such, we can effectively assume that $W(-\check{B}) > m \geq \check{B}$, after which we can proceed similarly to the lower bound. □

The full proof is in Appendix A, where we also state and prove Theorem A.7, a surrogate cost analogue of Theorem 4.11. Similarly, we use the same approach with θ-dependent bounds to prove Proposition 4.3 in Appendix B. These three ingredients together complete the proof of Theorem 4.1.

5 Simulations

We have shown that γ-Gittins achieves strong tail optimality. However, there remain unanswered questions that are important to practitioners. We now explore these questions via simulations. All simulations run with service to jobs being provided in discrete time steps and jobs arriving continuously over time, as described in Section 2.1.

- Section 5.1: How well does γ-Gittins perform in practical regimes? Does it perform as well as its tail constant predicts? How much extra benefit is there to having job size information? Namely, does γ-Boost achieve significantly better performance due to having job size information?
- Section 5.2: The three variants of γ-Gittins have the same asymptotic tail constant. In practice, at what latency threshold t does the performance of these variants converge? Is it in the pre-asymptotic regime?
- Section 5.3: How does the variance of the job size distribution affect the performance of γ-Gittins? Is it affected in similar ways as γ-Boost? How does the load of the system affect the performance of γ-Gittins? Does it increase, decrease or neither?
- Section 5.4: How robust is γ-Gittins to misspecification of γ?

5.1 γ-Gittins performance

In Figs. 5.1 and 5.2, we evaluate the performance of the optimal policy across different loads for two different distributions by looking at the empirical Tail Improvement Ratio (TIR), with respect to FCFS. A policy π's empirical TIR is given by $1 - \frac{P[T_\pi > t]}{P[T_{\text{FCFS}} > t]}$. Simulations show that γ-Gittins' (and its variants') performance improves on FCFS across a variety of loads. Our empirical results show that the performance of γ-Gittins aligns closely with the theoretical asymptotic TIR of $1 - \frac{C_{\gamma\text{-Gittins}}}{C_{\text{FCFS}}}$.

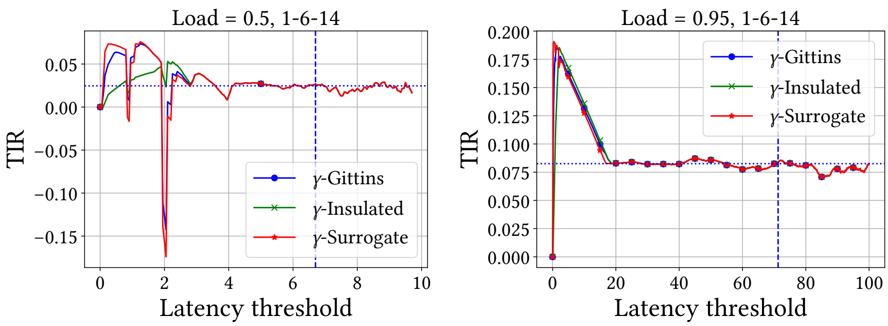

Fig. 5.1. Performance of all three variants of γ-Gittins on a discrete distribution, where job sizes can be $1/7$, $6/7$, or $14/7$, all with probability $1/3$. We refer to this as the 1-6-14 job size distribution but divide by 7 to normalize the mean. Service is provided in time steps of length $0.1/7$. 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 line denotes the 99th percentile response time for γ-Gittins. The left plot is the TIR for a load of 0.5 and the right plot is the TIR for a load of 0.95. Simulations run a total of one million busy periods.

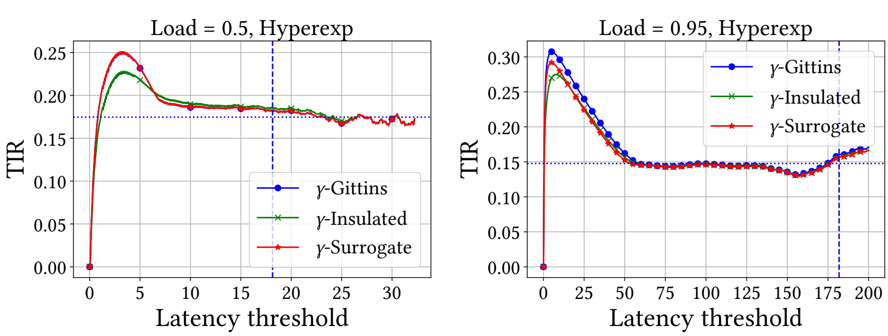

Fig. 5.2. Performance of all three variants of γ-Gittins on a discretized Hyperexponential distribution. The Hyperexponential distribution is drawn from $\text{Exp}(2)$ and $\text{Exp}(1/3)$, with first branch probability 0.8. To discretize, we consider a discrete distribution with job sizes from 0.1 to 25, in increments of 0.1. Service is provided in time steps of length 0.1. The probability for all job sizes s except 0.1 is given by $F(s) - F(s - 0.1)$, where F is the CDF of the Hyperexponential distribution described above. The probability of job size 0.1 is the difference between 1 and the sum of the other probabilities. The left plot is the TIR for a load of 0.5 and the right plot is the TIR for a load of 0.95. The dotted blue horizontal line indicates the numerical value of the theoretical asymptotic TIR $1 - C_{\gamma\text{-Gittins}}/C_{\text{FCFS}}$. The dashed vertical line denotes the 99th percentile response time for γ-Gittins. We run $\rho = 0.95$ for one million busy periods. Because the load of $\rho = 0.5$ is low, we run it for five million busy periods to ensure enough samples.

Simulations show that while γ-Gittins outperforms FCFS, as our theoretical results suggest, the performance of full-information γ-Boost can be significantly better than that of γ-Gittins. In Fig. 5.3, we plot the performance of γ-Boost with full job size information for comparison, and see that there remains a noticeable gap in performance. As γ-Gittins is theoretically the best that a policy can perform without job size information, this suggests that practitioners with tail-sensitive applications can obtain significant performance gains with accurate job size information.

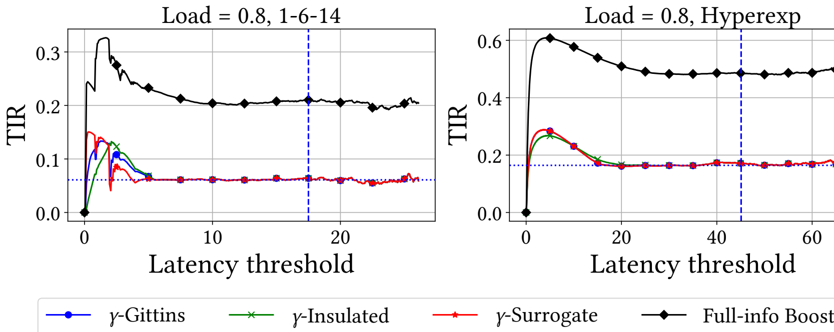

Fig. 5.3. Performance of all three variants of γ-Gittins compared to full-information γ-Boost. Performance is evaluated on the 1-6-14 job size distribution as in Fig. 5.1 and the discretized Hyperexponential distribution as in Fig. 5.2. The dotted blue horizontal line indicates the numerical value of the theoretical asymptotic TIR $1 - C_{\gamma\text{-Gittins}}/C_{\text{FCFS}}$. The dashed vertical line denotes the 99th percentile response time for γ-Gittins. Load is 0.8 for both scenarios. In both cases, full-information γ-Boost achieves roughly 3x better performance than its unknown size counterparts. Simulations run for one million busy periods.

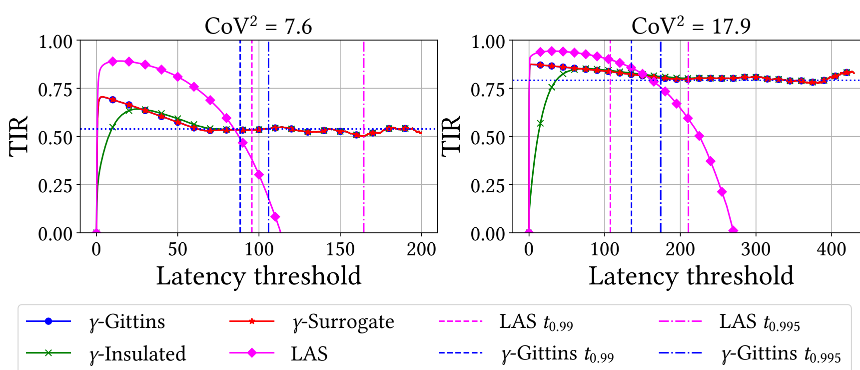

Fig. 5.4. A comparison of the Gittins variants and Least Attained Service (LAS) for different levels of variation. We use two discretized Hyperexponential distributions as in Fig. 5.2. On the left, the Hyperexponential consists of $\text{Exp}(4)$, $\text{Exp}(1/6)$ branches, with a first-branch probability of $p = 20/23$. On the right, the Hyperexponential consists of $\text{Exp}(8)$, $\text{Exp}(1/12)$, and a first-branch probability of $p = 88/95$. After discretization, these have a CoV^2 of approximately 7.6 and 17.9, respectively. We use a load of $\rho = 0.8$. Dashed vertical lines mark the $t_{0.99}$ response times of LAS and γ-Gittins, and dash-dotted vertical lines mark the $t_{0.999}$ response times of those two policies. The dotted horizontal line represents the theoretical asymptotic TIR of γ-Gittins. Observe how at higher variation, LAS has lower $t_{0.99}$ response time but still higher $t_{0.999}$ response time than γ-Gittins. Simulations run for one million busy periods.

5.2 Performance of minimally and maximally preemptive γ-Gittins

We proved in Corollary 4.2 that all three variants, γ-Gittins, γ-Surrogate, and γ-Insulated, have the same tail constant, but how do they each perform in "pre-asymptotic" regimes? As we see in Figs. 5.1 and 5.2, all three variants converge to the theoretical TIR at approximately the same rate.

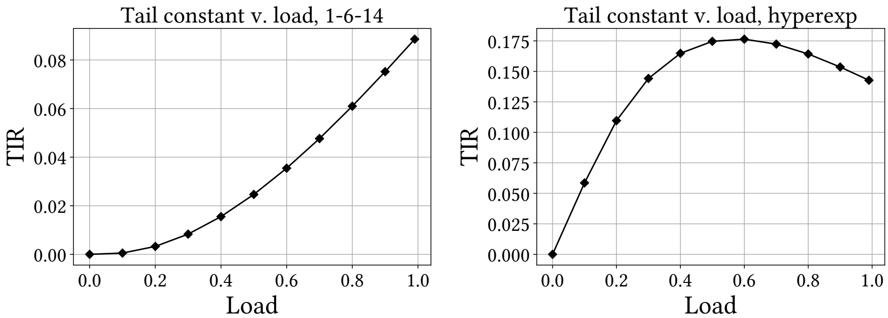

Fig. 5.5. The theoretical asymptotic TIR for γ-Gittins as a function of load. The left plot captures how the TIR evolves when the job size distribution is the 1-6-14 job size distribution as in Fig. 5.1. The right plot captures how the TIR evolves when the job size distribution is the discretized Hyperexponential job size distribution as in Fig. 5.2. Observe that the TIR is increasing for 1-6-14 as load increases, but this is not the case for the discretized Hyperexponential; as the load continues the increase, the TIR begins to decrease.

5.3 Effect of variance and load on tail improvement

In this section, we measure the tail performance against the variance of the job size distribution. The goal is to understand how variability in the job size distribution affects the tail performance of our policies. Namely, we wish to understand: when do policies optimal for class I distributions overtake suboptimal policies for tail performance, for different levels of job size distribution variability? In particular, we compare with Least Attained Service (LAS), which is strongly tail optimal for "heavy-tailed" job size distributions with unknown sizes.[12] Simulations (Fig. 5.4) suggest that as the variability of the job size distribution increases, LAS's performance improves; that is, the benefits of γ-Gittins are not seen until much higher (e.g. $t_{0.999}$) threshold response times.

Simulations show that γ-Gittins performance varies with respect to system load, depending on the actual job size distribution. We can see how the performance wrt load changes differently for two different job size distributions in Fig. 5.5. Namely, for the 1-6-14 job size distribution, the tail improvement ratio is monotonically increasing as load increases. However, for the discretized hyperexponential distribution, this is not the case. The highest tail improvement ratio is attained around a load of 0.6, and then decreases. This differs from γ-Boost, where, at least for all distributions tested in [34], full-information γ-Boost's TIR appears to strictly increase with increasing load.

5.4 Gamma sensitivity

In this section, we analyze the sensitivity of γ-Gittins to a misspecified decay rate parameter γ. As with its full-information counterpart, γ-Gittins only achieves strong tail optimality when the correct decay rate γ is used. However, in practice, one may only have access to noisy estimates of the job size distribution, in which case γ may be misspecified. We analyze the performance of γ-Gittins with various values of the decay rate parameter in Fig. 5.6. The results show that γ-Gittins is generally robust to misspecified γ.

6 Conclusion

We considered the problem of minimizing asymptotic tail latency in the light-tailed M/G/1 with unknown job sizes. Our main finding is that a novel variant of the Gittins index policy, which works

[12] Specifically, LAS is one of many policies that are strongly tail-optimal for regularly varying distributions [34, Appendix A].

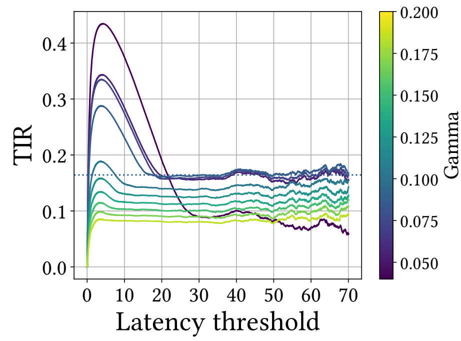

Fig. 5.6. Performance of γ-Gittins with misspecified γ. The theoretical asymptotic TIR with the correct γ is shown as a dotted horizontal line. We consider a range of $[\gamma/2, 2\gamma]$. The job size distribution is the discretized Hyperexponential distribution as in Fig. 5.2, with $\rho = 0.8$. The performance of γ-Gittins degrades for suboptimal γ but the TIR is still positive. Simulations run with one million busy periods.

with inflation instead of discounting, accomplishes this. Simulations show that the theoretical asymptotic tail improvements correspond to equal or better tail improvements at practical finite thresholds.

Our work opens up multiple open questions. Perhaps the most straightforward, albeit theoretically technical, is using excursion-theoretic or similar tools to extend the theory to continuous time [4, 15, 16, 19]. The fact that the Gittins index naturally handles weights means that it could potentially be used to minimize weighted tail objectives, which could prove useful for systems with different tiers of service. Another challenging question is whether one can emulate the Gittins index policy in settings where even the size distribution is unknown. A starting point could be to create a boost policy that resembles the Randomized Multi-Level Feedback (RMLF) policy [14], which approximately optimizes mean response time for all size distributions, in hopes of achieving similarly for asymptotic tail latency. A final direction is to explore the use of boost policies, both size-based and age-based, in network switches, which may be feasible using the Push-In First-Out (PIFO) framework [2].

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] Albert Gran Alcoz, Alexander Dietmüller, and Laurent Vanbever. 2020. SP-PIFO: Approximating Push-in First-out Behaviors Using Strict-Priority Queues. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 2020)*. USENIX Association, Santa Clara, CA, 59–76.

[3] 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

[4] Peter Bank and Christian Küchler. 2007. On Gittins' Index Theorem in Continuous Time. *Stochastic Processes and their Applications* 117, 9 (Sept. 2007), 1357–1371. doi:10.1016/j.spa.2007.01.006

[5] 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
[6] Shelby L. Brumelle. 1971. On the Relation between Customer and Time Averages in Queues. *Journal of Applied Probability* 8, 3 (1971), 508–520. doi:10.2307/3212174
[7] Nils Charlet and Benny Van Houdt. 2024. Tail Optimality and Performance Analysis of the Nudge-M Scheduling Algorithm. arXiv:2403.06588 [cs, math]
[8] Nicolas Gast, Bruno Gaujal, and Kimang Khun. 2022. Computing Whittle (and Gittins) Index in Subcubic Time. hal:hal-03602458
[9] John C. Gittins, Kevin D. Glazebrook, and Richard R. Weber. 2011. *Multi-Armed Bandit Allocation Indices* (2 ed.). Wiley, Chichester, UK.
[10] 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
[11] Mor Harchol-Balter. 2013. *Performance Modeling and Design of Computer Systems: Queueing Theory in Action*. Cambridge University Press, Cambridge, UK. doi:10.1017/CBO9781139226424
[12] Godfrey H. Hardy, John Edensor Littlewood, and George Pólya. 1952. *Inequalities* (2 ed.). Cambridge Univ. Press, Cambridge.
[13] Daniel P. Heyman and Shaler Stidham. 1980. The Relation between Customer and Time Averages in Queues. *Operations Research* 28, 4 (Aug. 1980), 983–994. doi:10.1287/opre.28.4.983
[14] Bala Kalyanasundaram and Kirk R. Pruhs. 2003. Minimizing Flow Time Nonclairvoyantly. *J. ACM* 50, 4 (July 2003), 551–567. doi:10.1145/792538.792545
[15] Nicole El Karoui and Ioannis Karatzas. 1994. Dynamic Allocation Problems in Continuous Time. *The Annals of Applied Probability* 4, 2 (May 1994), 255–286. doi:10.1214/aoap/1177005062
[16] Haya Kaspi and Avishai Mandelbaum. 1998. Multi-Armed Bandits in Discrete and Continuous Time. *The Annals of Applied Probability* 8, 4 (Nov. 1998), 1270–1290. doi:10.1214/aoap/1028903380
[17] John F. C. Kingman. 1993. *Poisson Processes*. Number 3 in Oxford Studies in Probability. Oxford University Press, Oxford.
[18] John D. C. Little. 2011. Little's Law as Viewed on Its 50th Anniversary. *Operations Research* 59, 3 (June 2011), 536–549. doi:10.1287/opre.1110.0940
[19] Avi Mandelbaum. 1987. Continuous Multi-Armed Bandits and Multiparameter Processes. *The Annals of Probability* 15, 4 (Oct. 1987), 1527–1556. doi:10.1214/aop/1176991992
[20] Michel Mandjes and Onno Boxma. 2023. *The Cramér–Lundberg model and its variants: A queueing perspective*. Springer Nature.
[21] Michael Pinedo. 2016. *Scheduling: Theory, Algorithms, and Systems* (5 ed.). Springer, Cham, Switzerland.
[22] Ziv Scully. 2022. *A New Toolbox for Scheduling Theory*. Ph. D. Dissertation. Carnegie Mellon University, Pittsburgh, PA.
[23] Ziv Scully, Isaac Grosof, and Mor Harchol-Balter. 2020. The Gittins Policy Is Nearly Optimal in the M/G/k under Extremely General Conditions. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 4, 3, Article 43 (Dec. 2020), 29 pages. doi:10.1145/3428328
[24] Ziv Scully and Mor Harchol-Balter. 2021. The Gittins Policy in the M/G/1 Queue. In *19th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt 2021)*. IFIP, Philadelphia, PA, 248–255. doi:10.23919/WiOpt52861.2021.9589051
[25] Ziv Scully, Mor Harchol-Balter, and Alan Scheller-Wolf. 2018. Optimal Scheduling and Exact Response Time Analysis for Multistage Jobs. arXiv:1805.06865 [cs, math]
[26] 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
[27] 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
[28] 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
[29] 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
[30] Gideon Weiss. 1988. Branching Bandit Processes. *Probability in the Engineering and Informational Sciences* 2, 3 (July 1988), 269–278. doi:10.1017/S0269964800000826
[31] P. Whittle. 1981. Arm-Acquiring Bandits. *The Annals of Probability* 9, 2 (April 1981). doi:10.1214/aop/1176994469

[32] 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
[33] Ronald W. Wolff. 1982. Poisson Arrivals See Time Averages. *Operations Research* 30, 2 (April 1982), 223–231. doi:10.1287/opre.30.2.223
[34] 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

A Proofs for Tail Constant of General Boost Policies

This appendix provides proofs for all the results needed to prove Theorem 4.1 and its surrogate cost analogue, Theorem A.7.

A.1 Tail Constant for Response Time

Lemma 4.6. *For any boost policy π with boost function b, any $u \in \mathbb{R}$, and any $v \in [0, \infty]$,*

$$\mathbf{E}[e^{\gamma V(u,v)}] = \exp\biggl(\int_0^v \lambda \mathbf{E}[e^{\gamma S[\underline{b} > t]} - 1] \, \mathrm{d}t\biggr),$$

where $S[\underline{b} > t] = \min\{z \geq 0 : X_z = x_{\mathrm{done}} \text{ or } \underline{b}(X_{0:z}) \leq t\}$ is the random amount of service a job requires to either complete or stop having (lower envelope of) boost greater than t.

Proof. We begin with two simplifying observations. First, because $V(u, v)$ is stationary with respect to u, it suffices to consider the $u = 0$ case. Second, we could use b instead of $\underline{b}$ in the definition of $S[\underline{b} > t]$: the first time a job's boost dips below t is also the first time the lower envelope does so.

The result is a relatively routine application of Campbell's formula for the Laplace functional of a Poisson point process [17, Section 3.2], similar to an analogous application for known-size boost policies [34, Lemma 3.5]. Specifically, let $\mathcal{A}$ be the random set of pairs $(x_{0:s}, t)$ corresponding to the trajectories $x_{0:s}$ and arrival times t of jobs that arrive during $(u, u + v)$. The crossing work is

$$\begin{aligned} V(0, v) &= \sum_{(x_{0:s},t) \in \mathcal{A}} (\text{number of steps of } x_{0:s} \text{ where the lower boost envelope is greater than } t) \\ &= \sum_{(x_{0:s},t) \in \mathcal{A}} \min\{u \in \{0, \dots, s\} : b(x_{0:u}) \leq t\}. \end{aligned}$$

The intensity measure μ of the $\mathcal{A}$ Poisson process is $\mu(\mathcal{X} \times \mathrm{d}t) = \mathbf{P}[X_{0:S} \in \mathcal{X}] \, \lambda \, \mathrm{d}t$, so Campbell's formula, Tonelli's theorem, and the definition of $S[\underline{b} > t]$ together imply

$$\begin{aligned} \mathbf{E}[e^{\gamma V(0,v)}] &= \exp\biggl(\int \bigl(\exp\bigl(\gamma \min\{u \in \{0, \dots, s\} : b(x_{0:u}) \leq t\}\bigr) - 1\bigr) \, \mu(\mathrm{d}(x_{0:s}, t))\biggr) \\ &= \exp\biggl(\int_0^v \mathbf{E}\bigl[\exp\bigl(\gamma \min\{u \in \{0, \dots, S\} : b(X_{0:u}) \leq t\}\bigr) - 1\bigr] \, \lambda \, \mathrm{d}t\biggr) \\ &= \exp\biggl(\int_0^v \lambda \mathbf{E}[e^{\gamma S[\underline{b} > t]} - 1] \, \mathrm{d}t\biggr). \end{aligned}$$ □

Observation A.1. Let U be the first age at which the tagged job has boost B. For the tagged job to be served at age U, there can be no work in the system with boosted arrival time before $-B$. Thus for the tagged job to have age greater than U, and in particular for the tagged job to have completed, there must have been a point in time since the arrival of the job where there was no work with boosted arrival time less than $-B$.

Observation A.2. The existing work $W(u)$ has the stationary distribution by a reasoning similar to that in [34, Section 3.1]. In particular, $W(-B)$ is distributed according to the stationary distribution, because the tagged job's boost is independent of its arrival time.

To see that this is the case, note that since $\mathbb{X}^{\text{traj}}$ is countable, the set of possible values for B, the worst-ever boost of the tagged job, is countable. Since the worst-ever boost is independent of the arrival time, we can use Poisson thinning to split the arrival process into independent Poisson processes, each corresponding to a specific worst-ever boost. Now consider a specific one of these processes corresponding to a constant worst-ever boost $\beta \geq 0$. Clearly, if we shift the process by $-\beta$ in time, it is still a Poisson process and independent of the other processes and all job trajectories. Additionally, the value of $W(u)$ at an arrival of this shifted process is independent of all future job arrivals (from all the processes), so by PASTA [33], this shifted process observes the stationary distribution. Since this holds for each of the "thinned" processes, this must also hold for the original Poisson arrival process.

Above, we have used the fact that there are countably many possible boost values so that we can use Poisson thinning, but one could reason in much the same way if there were uncountably many boost values. The key idea is to view the arrival process as a Poisson process of (arrival time, initial state) pairs over $\mathbb{R} \times \mathbb{X}$ that is time-homogeneous, i.e. the distribution is invariant to shifting the first element of the pairs. One can then check that the induced (boosted arrival time, initial state) pairs are still a time-homogeneous Poisson process over $\mathbb{R} \times \mathbb{X}$.

Lemma 4.9. *Let $\check{B} = \min(B, m)$ for any fixed $m \geq 0$. Then for all boost policies,*

$$T \geq W(-B) + V(-B, W(-B)) + S - B,$$

$$T \leq \max\{W(-\check{B}), \check{B}\} + V(-\check{B}, \infty) + \overline{V}(-\check{B}, m)\mathbb{1}(W(-\check{B}) < \check{B}) + S - \check{B}.$$

Proof. Recall we assume without loss of generality that the arrival time of the tagged job is time 0. The key idea is to analyze the maximum and minimum amounts of work that the server could have done after time $-B$ before completing the tagged job.

Lower bound: Observe that

- At time $-B$, there is $W(-B)$ work in the system with boosted arrival time before $-B$.
- At time $-B + W(-B)$, there is at least $V(-B, W(-B))$ work in the system with boosted arrival time before $-B$.

This tells us that at all times from $-B$ to $-B + W(-B) + V(-B, W(-B))$, there is work in the system with boosted arrival time less than $-B$. Therefore, Observation A.1 tells us that at least $W(-B) + V(-B, W(-B))$ other work was completed before the tagged job completed. Since the tagged job itself requires S service, we conclude that there was at least

$$W(-B) + V(-B, W(-B)) + S$$

work that needed to be completed after time $-B$ for the tagged job to complete, which yields the desired lower bound.

Upper bound: Observe that decreasing the tagged job's worst ever boost B to $\check{B}$ can only increase its response time. Thus, it is sufficient to analyze the response time as if the worst ever boost is $\check{B}$. To attain an upper bound, it is sufficient to bound all current and future work that could be served before completing a job with boosted arrival time $-\check{B}$.

Observe that if $W(-\check{B}) \geq \check{B}$, this work includes

- $W(-\check{B})$: the work in the system at time $-\check{B}$,
- $V(-\check{B}, \infty)$: all such work that arrives after time $-\check{B}$ with boosted arrival time before $-\check{B}$.

If instead $W(-\check{B}) < \check{B}$, we can upper bound the tagged job's response time by preventing it from receiving service until after its arrival time. However, at time 0 (its arrival time), there may be work belonging to $\overline{V}(-\check{B}, m)$ that cannot be preempted (see Example A.3 below). Finally, all future work that has priority over the tagged job is bounded by, $V(-\check{B}, \infty)$. Therefore, along with the S service required by the tagged job to complete, we conclude that between $-\check{B}$ and the tagged job's completion, the amount of work completed is at most

$$\max\{W(-\check{B}), \check{B}\} + V(-\check{B}, \infty) + \overline{V}(-\check{B}, m)\mathbb{1}(W(-\check{B}) < \check{B}) + S. \qquad \square$$

Example A.3. A job J arrives in $(-\check{B}, 0)$ and has boosted arrival time after $-\check{B}$. It receives service prior to the arrival of the tagged job and has its boosted arrival time improve to before $-\check{B}$. Thus, when the tagged job enters the system, J has priority over it. However, all of J's work is in $\overline{V}(-\check{B}, m)$ because its arrival time is in $(-\check{B}, -\check{B} + m)$ and it has no $(-\check{B})$-crossing work.

Observation A.4. A basic fact about Poisson arrival processes is that the arrivals in disjoint time intervals are independent of each other. Thus, for all boost policies π, and an arbitrary time u, $W(u)$, $V(u, \infty)$, and S are independent since $W(u)$ corresponds to arrivals before time u, $V(u, \infty)$ corresponds to arrivals after time u, and S is independent of all other jobs and arrival times.

Theorem 4.11. *For any boost policy π, we have $C_\pi = C_W \mathbf{E}[e^{\gamma V_\pi(0,\infty)}]\mathbf{E}[e^{\gamma(S-B_\pi)}]$.*

Proof. It suffices to show

$$\liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta T_\pi}] \geq C_W \mathbf{E}[e^{\gamma V_\pi(0,\infty)}]\mathbf{E}[e^{\gamma(S-B_\pi)}]$$

$$\limsup_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta T_\pi}] \leq C_W \mathbf{E}[e^{\gamma V_\pi(0,\infty)}]\mathbf{E}[e^{\gamma(S-B_\pi)}].$$

For the lower bound, consider an arbitrary $w > 0$. Lemma 4.9 implies that

$$\begin{aligned}
&\liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta T_\pi}] \\
&\quad \geq \liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta(W(-B_\pi) + V_\pi(-B_\pi, W(-B_\pi)) + S - B_\pi)}] \\
&\quad \geq \liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta(W(-B_\pi) + V_\pi(-B_\pi, W(-B_\pi)) + S - B_\pi)} \mathbb{1}(W(-B_\pi) > w)] \\
&\quad \geq \liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta(W(-B_\pi) + V_\pi(-B_\pi, w) + S - B_\pi)} \mathbb{1}(W(-B_\pi) > w)].
\end{aligned}$$

By independence (Observation A.4), we can separate the terms in the last line, yielding

$$\begin{aligned}
&= \liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta W(-B_\pi)} \mathbb{1}(W(-B > w))]\mathbf{E}[e^{\theta V_\pi(-B_\pi, w)}]\mathbf{E}[e^{\theta(S-B_\pi)}] \\
&= \liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta W(-B_\pi)} \mathbb{1}(W(-B > w))]\mathbf{E}[e^{\theta V_\pi(0, w)}]\mathbf{E}[e^{\theta(S-B_\pi)}] \\
&\geq C_W \mathbf{E}[e^{\gamma V_\pi(-B_\pi, w)}]\mathbf{E}[e^{\gamma(S-B_\pi)}],
\end{aligned}$$

where the second line is by stationarity of V_π, and the last line is by Fatou's lemma. Since our choice of w was arbitrary, this bound holds in the $w \to \infty$ limit. The monotone convergence theorem implies that

$$\liminf_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[e^{\theta T_\pi}] \geq C_W \mathbf{E}[e^{\gamma V_\pi(0,\infty)}]\mathbf{E}[\exp \gamma(S - B_\pi)].$$

If $\mathbf{E}[e^{\gamma V_\pi(0,\infty)}]$ is infinite, then the RHS of the above is infinite, which implies that,

$$\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\pi}],$$

is infinite as well. For the upper bound, we need only compute the case where $\mathbf{E}[e^{\gamma V_\pi(0,\infty)}]$ is finite. For any $m \geq 0$, Lemma 4.9 implies

$$\begin{aligned}
&\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\pi}] \\
&\quad\leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(\max\{W(-\check{B}_\pi),\check{B}_\pi\}+V_\pi(-\check{B}_\pi,\infty)+\overline{V}(-\check{B}_\pi,m)\mathbb{1}(W(-\check{B}_\pi)<\check{B}_\pi)+S-\check{B}_\pi)}] \\
&\quad= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(V_\pi(-\check{B}_\pi,\infty)+\overline{V}(-\check{B}_\pi,m)+S)}\mathbb{1}(W(-\check{B}_\pi)<\check{B}_\pi)]+ \\
&\qquad \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-\check{B}_\pi)+V_\pi(-\check{B}_\pi,\infty)+S-\check{B}_\pi)}\mathbb{1}(W(-\check{B}_\pi)\geq\check{B}_\pi)] \\
&\quad\leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(V_\pi(-\check{B}_\pi,\infty)+\overline{V}(-\check{B}_\pi,m)+S)}]+ \\
&\qquad \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-\check{B}_\pi)+V_\pi(-\check{B}_\pi,\infty)+S-\check{B}_\pi)}].
\end{aligned}$$

Observe that $V_\pi(-\check{B}_\pi,\infty)+\overline{V}(-\check{B}_\pi,m)$ is simply all the work that arrives in the interval $(-\check{B}_\pi,-\check{B}_\pi+m]$, call it A_1 and all the work that arrives after $-\check{B}_\pi+m$ which boosts past $-\check{B}_\pi$, call it A_2. Then

$$\begin{aligned}
&\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(V_\pi(-\check{B}_\pi,\infty)+\overline{V}(-\check{B}_\pi,m)+S)}] \\
&\quad= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(A_1+A_2+S)}] \\
&\quad= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta A_1}]\mathbf{E}[e^{\theta A_2}]\mathbf{E}[e^{\theta S}] \\
&\quad\leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta A_1}]\mathbf{E}[e^{\theta V_\pi(-\check{B}_\pi,\infty)}]\mathbf{E}[e^{\theta S}] \\
&\quad\leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\gamma A_1}]\mathbf{E}[e^{\gamma V_\pi(-\check{B}_\pi,\infty)}]\mathbf{E}[e^{\gamma S}] \\
&\quad= 0,
\end{aligned}$$

where we have used the independence of A_1 and A_2 (which follows from the independence of disjoint time intervals of arrivals), and the fact that the transform of all three factors is finite:

- $\mathbf{E}[e^{\gamma S}]$ is finite by the definition of γ.
- $\mathbf{E}[e^{\gamma V_\pi(-\check{B}_\pi,\infty)}]$ is finite by assumption.
- A_1 is simply the amount of work that arrives in an interval of length m, where arrivals have distribution S and arrival rate λ, so $\mathbf{E}[e^{\gamma A_1}] = e^{\lambda m(\mathbf{E}[e^{\gamma S}]-1)} = e^{\gamma m}$, which is finite, by a standard M/G/1 result [11, Chapter 25.6] and the definition of γ.

We now compute the limit for the second term:

$$\begin{aligned}\limsup_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\pi}] &\le \limsup_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-\check{B}_\pi)+V_\pi(-\check{B}_\pi,\infty)+S-\check{B}_\pi)}]\\ &= \limsup_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta W(-\check{B}_\pi)}]\mathbf{E}[e^{\theta V_\pi(-\check{B}_\pi,\infty)}]\mathbf{E}[e^{\theta(S-\check{B}_\pi)}]\\ &= \limsup_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta W(-\check{B}_\pi)}]\mathbf{E}[e^{\theta V_\pi(0,\infty)}]\mathbf{E}[e^{\theta(S-\check{B}_\pi)}]\\ &\le C_W\mathbf{E}[e^{\gamma V_\pi(-\check{B}_\pi,\infty)}]\mathbf{E}[e^{\gamma(S-\check{B}_\pi)}].\end{aligned}$$

We have used the stationarity of V_π in the second line and reverse Fatou's Lemma for the last line. Because this holds for any m, it holds in the $m\to\infty$ limit. Monotone convergence and the fact that $\check{B}_\pi\to B_\pi$ as $m\to\infty$ yields the desired limit. □

A.2 Tail Constant for Surrogate Cost

Lemma A.5. *For all $i\ge 1$,*

$$\lim_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\pi(i)}\mid X_{0:S}] = C_W\mathbf{E}[e^{\gamma V_\pi(0,\infty)}]\mathbf{E}[e^{\gamma(i-\underline{b}(X_{0:i-1}))}\mid X_{0:S}],$$

where $T_\pi(i)$ is the time to reach age i under policy π.

Proof. Follow the proof of Theorem 4.11 but with conditional expectations, and substituting S with i and B_π with $\underline{b}(X_{0:i-1})$, as defined in the above statement. Using the fact that the crossing work and amount of work in system before the boosted arrival of the tagged job are independent of the tagged job's service history yields the above result. □

Lemma A.6. *For any policy π such that $\mathbf{E}[e^{\gamma V_\pi(0,\infty)}]<\infty$,*

$$\lim_{\theta\to\gamma}\mathbf{E}\left[\sum_{i=1}^{S}\frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\pi(i)}\,\middle|\,X_{0:S}\right]\right] = \mathbf{E}\left[\sum_{i=1}^{S}\lim_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\pi(i)}\,\middle|\,X_{0:S}\right]\right]$$

Proof. To prove this, we need only justify the use of the dominated convergence theorem. Observe that

$$\begin{aligned}\left|\sum_{i=1}^{S}\frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\pi(i)}\,\middle|\,X_{0:S}\right]\right| &\le \sum_{i=1}^{S}\frac{\gamma-\theta}{\gamma}\mathbf{E}\left[e^{\theta T_\pi(i)}\,\middle|\,X_{0:S}\right]\\ &\le \sum_{i=1}^{S}\frac{\gamma-\theta}{\gamma}\mathbf{E}\left[e^{\theta(W(0)+V_\pi(0,\infty)+S)}\,\middle|\,X_{0:S}\right]\\ &\le \sum_{i=1}^{S}\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta W(0)}]\mathbf{E}[e^{\theta V_\pi(0,\infty)}]\mathbf{E}[e^{\theta S}\mid X_{0:S}]\\ &\le \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta W(0)}]\mathbf{E}[e^{\theta V_\pi(0,\infty)}]Se^{\theta S}\end{aligned}$$

where the first inequality is by Lemma C.1, the second inequality is by Corollary 4.10, and the third inequality is by the fact that W and V_π are both independent of the tagged job's service history Observation A.4. By (2.1), for θ sufficiently close to γ,

$$\frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta W(0)}]\mathbf{E}[e^{\theta V_\pi(0,\infty)}]Se^{\theta S}\le(C_W+1)\mathbf{E}[e^{\theta V_\pi(0,\infty)}]Se^{\theta S}\le(C_W+1)\mathbf{E}[e^{\gamma V_\pi(0,\infty)}]Se^{\gamma S}.$$

Thus, for θ sufficiently close to γ,

$$\left|\sum_{i=1}^{S} \frac{\gamma - \theta}{\gamma} \underline{\Gamma}_{\theta}(X_{0:i-1}) \mathbf{E}\left[e^{\theta T_{\pi}(i)} \,\middle|\, X_{0:S}\right]\right| \leq (C_W + 1)\mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}] S e^{\gamma S},$$

and so the last step to be able to apply the dominated convergence theorem is showing that

$$\mathbf{E}[(C_W + 1)\mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}] S e^{\gamma S}] = (C_W + 1)\mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}]\mathbf{E}[S e^{\gamma S}] < \infty,$$

for which it will be sufficient to show that $\mathbf{E}[S e^{\gamma S}] < \infty$. Observe that $\mathbf{E}[e^{(\gamma+\varepsilon)S}] < \infty$ for some $\varepsilon > 0$ since S is class I (Definition 2.1) and $\mathbf{E}[e^{\gamma S}] < \infty$. For this same ε, there exists some $M > 0$ such that $S \geq M$ implies $S e^{\gamma S} \leq e^{(\gamma+\varepsilon)S}$, and so

$$\mathbf{E}[S e^{\gamma S}] \leq R e^{\gamma R} + \mathbf{E}[e^{(\gamma+\varepsilon)S}] < \infty. \qquad \square$$

Theorem A.7 (Surrogate Cost Analogue of Theorem 4.11). *For any boost policy π where $\mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}] < \infty$,*

$$\widehat{C}_{\pi} = \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[\widehat{\mathrm{Cost}}_{\pi}(\theta)] = C_W \mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}] \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\gamma}(X_{0:i-1}) e^{\gamma(i - \underline{b}(X_{0:i-1}))}\right],$$

where $\underline{b}(X_{0:i-1})$ is the worst-ever boost we see before an age i.

Proof of Theorem A.7. Observe that

$$\begin{aligned}
\widehat{C}_{\pi} &= \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}[\widehat{\mathrm{Cost}}_{\pi}(\theta)] \\
&= \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\theta}(X_{0:i-1}) e^{\theta T_{\pi}(i)}\right] \\
&= \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}\left[\mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\theta}(X_{0:i-1}) e^{\theta T_{\pi}(i)} \,\middle|\, X_{0:S}\right]\right] && \text{(Tower rule)} \\
&= \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\theta}(X_{0:i-1}) \mathbf{E}\left[e^{\theta T_{\pi}(i)} \,\middle|\, X_{0:S}\right]\right] \\
&= \mathbf{E}\left[\sum_{i=1}^{S} \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \underline{\Gamma}_{\theta}(X_{0:i-1}) \mathbf{E}\left[e^{\theta T_{\pi}(i)} \,\middle|\, X_{0:S}\right]\right] && \text{(Lemma A.6)} \\
&= \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\gamma}(X_{0:i-1}) \lim_{\theta \to \gamma} \frac{\gamma - \theta}{\gamma} \mathbf{E}\left[e^{\theta T_{\pi}(i)} \,\middle|\, X_{0:S}\right]\right] && \text{(Lemma C.4)} \\
&= C_W \mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}] \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\gamma}(X_{0:i-1}) \mathbf{E}[e^{\gamma(i - \underline{b}(X_{0:i-1}))} \mid X_{0:S}]\right] && \text{(Lemma A.5)} \\
&= C_W \mathbf{E}[e^{\gamma V_{\pi}(0,\infty)}] \mathbf{E}\left[\sum_{i=1}^{S} \underline{\Gamma}_{\gamma}(X_{0:i-1}) e^{\gamma(i - \underline{b}(X_{0:i-1}))}\right].
\end{aligned}$$

Lemma A.6 uses the rough bound provided by Corollary 4.10 to justify moving the limit into the expectation by the dominated convergence theorem. Lemma C.4 shows that the surrogate θ-price of a job is left-continuous on the interval $\gamma \geq \theta > 0$. $\square$

B Costs in $\theta \to \gamma$ limit results

This appendix provides proofs for all results needed to prove Proposition 4.3. Throughout, we subscript by θ to denote θ-Gittins. For example, B_θ is equivalent to $B_{\theta\text{-Gittins}}$, the worst-ever boost under the policy θ-Gittins.

Lemma B.1. $C_{\gamma\text{-Insulated}} \leq \liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_{\theta\text{-Insulated}}(\theta)]$.

Proof. We start the proof by following the steps of the proof of Theorem 4.11. Applying the lower bound in Lemma 4.9, we get that for all $w \geq 0$,

$$\begin{aligned}
&\liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_{\theta\text{-Insulated}}(\theta)] \\
&\quad\geq \liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-B_\theta)+V_\theta(-B_\theta,W(-B_\theta))+S-B_\theta)}] \\
&\quad\geq \liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-B_\theta)+V_\theta(-B_\theta,w)+S-B_\theta)}\mathbb{1}(W(-B_\theta) > w)] \\
&\quad= \liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-B_\theta))}\mathbb{1}(W(-B_\theta) > w)]\mathbf{E}[e^{\theta V_\theta(-B_\theta,w)}]\mathbf{E}[e^{\theta(S-B_\theta)}]
\end{aligned}$$

where the last equality follows from the independence of $W(-B_\theta)$, $V_\theta(-B_\theta, w)$, and $(S-B_\theta)$. Observe that since $W(\cdot)$ is identically distributed at all times,

$$= C_W \liminf_{\theta\to\gamma} \mathbf{E}[e^{\theta V_\theta(-B_\theta,w)}] \liminf_{\theta\to\gamma} \mathbf{E}[e^{\theta(S-B_\theta)}].$$

By the same reasoning, $V_\theta(\cdot, w)$ is identically distributed at all times and we may simplify to

$$= C_W \liminf_{\theta\to\gamma} \mathbf{E}[e^{\theta V_\theta(0,w)}] \liminf_{\theta\to\gamma} \mathbf{E}[e^{\theta(S-B_\theta)}].$$

$V_\theta(0, w)$ is non-increasing in θ because increasing θ means decreasing these jobs' boosts (Lemma C.3) which only decreases $V_\theta(0, w)$.

$$\geq C_W \liminf_{\theta\to\gamma} \mathbf{E}[e^{\theta V_\gamma(0,w)}] \liminf_{\theta\to\gamma} \mathbf{E}[e^{\theta(S-B_\theta)}].$$

Finally, B_θ is left-continuous and non-increasing in θ (Lemmas C.3 and C.4) so we may apply the monotone convergence theorem,

$$\geq C_W \mathbf{E}[e^{\gamma V_\gamma(0,w)}]\mathbf{E}[e^{\gamma(S-B_\gamma)}].$$

Since this holds for all $w \geq 0$, it must also hold in the $w \to \infty$ limit. Once again applying monotone convergence theorem,

$$\liminf_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_{\theta\text{-Insulated}}(\theta)] \geq C_W \mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]\mathbf{E}[e^{\gamma(S-B_\gamma)}] = C_{\gamma\text{-Insulated}}$$

where the last equality follows from Theorem 4.11. □

Lemma B.2. $C_{\gamma\text{-Insulated}} \geq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_{\theta\text{-Insulated}}(\theta)]$.

Proof. Following the initial steps in the upper bound in the proof of Theorem 4.11, we get that for any $m \geq 0$,

$$\begin{aligned}\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[\mathrm{Cost}_{\theta\text{-Insulated}}(\theta)] &= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\theta}] \\ &\leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(V_\theta(-\check{B}_\theta,\infty)+\overline{V}_\theta(-\check{B}_\theta,m)+S)}]+ \\ &\quad \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-\check{B}_\theta)+V_\theta(-\check{B}_\theta,\infty)+S-\check{B}_\theta)}].\end{aligned}$$

Observe that $V_\theta(-\check{B}_\theta,\infty)+\overline{V}_\theta(-\check{B}_\theta,m)$ is simply all the work that arrives in the interval $[-\check{B}_\theta,-\check{B}_\theta+m]$, call it $A_{1,\theta}$, and all the work that arrives after $-\check{B}_\theta+m$ which boosts past $-\check{B}_\theta$ with boost function $b_{\theta\text{-Insulated}}$, call it $A_{2,\theta}$. Note that since the system has Poisson arrivals, $A_{1,\theta}$ is stationary in θ and furthermore, $A_{1,\theta}$, $A_{2,\theta}$, and S are independent, so

$$\begin{aligned}&\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(V_\theta(-\check{B}_\theta,\infty)+\overline{V}_\theta(-\check{B}_\theta,m)+S)}] \\ &\quad= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(A_{1,\theta}+A_{2,\theta}+S)}] \\ &\quad= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta A_{1,\theta}}]\mathbf{E}[e^{\theta A_{2,\theta}}]\mathbf{E}[e^{\theta S}] \\ &\quad= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta A_{1,\gamma}}]\mathbf{E}[e^{\theta A_{2,\theta}}]\mathbf{E}[e^{\theta S}].\end{aligned}$$

Now note that $A_{2,\theta} < V_\theta(-\check{B}_\theta,\infty)$ by definition and $V_\theta(\cdot,\infty)$ is stationary in θ. Thus,

$$\limsup_{\theta\to\gamma}\mathbf{E}[e^{\theta A_{2,\theta}}] \leq \limsup_{\theta\to\gamma}\mathbf{E}[e^{\theta V_\theta(0,\infty)}] = \mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]$$

Putting these together,

$$\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(V_\theta(-\check{B}_\theta,\infty)+\overline{V}_\theta(-\check{B}_\theta,m)+S)}] \leq \mathbf{E}[e^{\gamma A_{1,\gamma}}]\mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]\mathbf{E}[e^{\gamma S}]\limsup_{\theta\to\gamma}\frac{\gamma-\theta}{\gamma} = 0$$

where the last equality follows from the fact that the three expectation terms are all finite:

- $\mathbf{E}[e^{\gamma S}]$ is finite by the definition of γ.
- $\mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]$ is finite by assumption.
- A_1 is simply the amount of work that arrives in an interval of length m, where arrivals have distribution S and arrival rate λ, so $\mathbf{E}[e^{\gamma A_1}] = e^{\lambda m(\mathbf{E}[e^{\gamma S}]-1)} = e^{\gamma m}$, which is finite, by a standard M/G/1 result [11, Chapter 25.6] and the definition of γ.

We are left with

$$\begin{aligned}\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\theta}] &\leq \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta(W(-\check{B}_\theta)+V_\theta(-\check{B}_\theta,\infty)+S-\check{B}_\theta)}] \\ &= \limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta W(-\check{B}_\theta)}]\mathbf{E}[e^{\theta V_\theta(-\check{B}_\theta,\infty)}]\mathbf{E}[e^{\theta(S-\check{B}_\theta)}] \\ &= C_W\mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]\mathbf{E}[e^{\gamma(S-\check{B}_\gamma)}].\end{aligned}$$

Because this holds for any m, it holds in the $m \to \infty$ limit,

$$\begin{aligned}\limsup_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\theta}] &\le \lim_{m\to\infty} C_W \mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]\mathbf{E}[e^{\gamma(S-\check{B}_\gamma)}] \\ &= C_W \mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]\mathbf{E}[e^{\gamma(S-B_\gamma)}] \\ &= C_{\gamma\text{-Insulated}},\end{aligned}$$

where the first equality follows by the dominated convergence theorem and the fact that $\check{B}_\theta \to B_\theta$ as $m \to \infty$, and the second equality follows from Theorem 4.11. □

Lemma B.3. *For both θ-Insulated and θ-Surrogate:*

$$\lim_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\mathbf{E}[e^{\theta T_\theta(i)} \mid X_{0:S}] = C_W \mathbf{E}[e^{\gamma V_\gamma(0,\infty)}]\mathbf{E}[e^{\gamma(i-\underline{b}_\gamma(X_{0:i-1}))} \mid X_{0:S}].$$

Proof. Look at the proof of Theorem 4.11 and note that changing the stopping time from S to i does not change the proof. Similarly, conditioning on the trajectory does not change the proof. □

Lemma B.4 (Analogue of Lemma A.6).

$$\lim_{\theta\to\gamma} \mathbf{E}\left[\sum_{i=1}^{S} \frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\theta(i)} \,\middle|\, X_{0:S}\right]\right] = \mathbf{E}\left[\sum_{i=1}^{S} \lim_{\theta\to\gamma} \frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\theta(i)} \,\middle|\, X_{0:S}\right]\right].$$

Proof. The proof is essentially the same as that of Lemma A.6, but with slightly different justification for applying dominated convergence theorem, so we give only this different justification below.

Imitate the proof of Lemma A.6 up to the conclusion that for θ sufficiently close to γ,

$$\left|\sum_{i=1}^{S} \frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\theta(i)} \,\middle|\, X_{0:S}\right]\right| \le (C_W+1)\mathbf{E}[e^{\theta V_\theta(0,\infty)}]Se^{\theta S}.$$

Recall from Lemma C.3 that under θ-Gittins, the boost of all states is decreasing as a function of θ. This means that $V_\theta(0,\infty)$ is decreasing as a function of θ, because strictly fewer states "boost past" time 0. Therefore, for any fixed $\eta < \gamma$, we have

$$\limsup_{\theta\to\gamma} \mathbf{E}[e^{\theta V_\theta(0,\infty)}] \le \mathbf{E}[e^{\gamma V_\eta(0,\infty)}],$$

which is finite since all states have bounded boost (Corollary C.2). So for θ sufficiently close to γ,

$$\mathbf{E}[e^{\theta V_\theta(0,\infty)}] < \mathbf{E}[e^{\gamma V_\eta(0,\infty)}].$$

Thus, for θ sufficiently close to γ,

$$\left|\sum_{i=1}^{S} \frac{\gamma-\theta}{\gamma}\underline{\Gamma}_\theta(X_{0:i-1})\mathbf{E}\left[e^{\theta T_\theta(i)} \,\middle|\, X_{0:S}\right]\right| \le (C_W+1)\mathbf{E}[e^{\gamma V_\eta(0,\infty)}]Se^{\gamma S}.$$

The right-hand side does not depend on θ, so it is suitable as a bound for applying dominated convergence theorem. The remainder of the proof follows as in Lemma A.6. □

Proposition 4.3. *For* θ-Policy $\in$ {θ-Gittins, θ-Surrogate, θ-Insulated},

$$\lim_{\theta\to\gamma} |C_{\gamma\text{-Policy}}(\theta) - C_{\theta\text{-Policy}}(\theta)| = 0, \tag{4.2}$$

$$\lim_{\theta\to\gamma} |\widehat{C}_{\gamma\text{-Policy}}(\theta) - \widehat{C}_{\theta\text{-Policy}}(\theta)| = 0. \tag{4.3}$$

Proof. The first equality follows from Lemmas B.1 and B.2 and then observing that the proof is unchanged if we replace Insulated with Policy everywhere. This is because all the terms only depend on the lower envelope of the boost function, which is identical for all three policies.

The second equality follows by imitating the Theorem A.7 except at the step that uses Lemma A.6 use Lemma B.4 and at the step that uses Theorem 4.11 use Lemma B.3. □

C Properties of the θ-Gittins Index and θ-Gittins Boost Function

This appendix proves basic properties of the θ-Gittins index and θ-Gittins boost function that we need.

Lemma C.1. *For any $x \in \mathbb{X}$ and all $\gamma \geq \theta > 0$,*

$$1 \geq \Gamma_\theta(x) \geq \frac{e^\theta - 1}{e^\theta}.$$

Proof. For the upper bound observe that,

$$\begin{aligned}\Gamma_\theta(x) &= \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}[\text{InflatedCost}(\theta, x, 0, \mathbb{Y})]}{\mathbf{E}[\text{InflatedTime}(\theta, x, \mathbb{Y})]} \\ &\leq \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}[e^{\theta\,\text{Service}(x,\mathbb{Y})}]}{\mathbf{E}\Big[\sum_{t=1}^{\text{Service}(x,\mathbb{Y})} e^{\theta t}\Big]} \\ &\leq \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}[e^{\theta\,\text{Service}(x,\mathbb{Y})}]}{\mathbf{E}[e^{\theta\,\text{Service}(x,\mathbb{Y})}]} = 1,\end{aligned}$$

where the last inequality follows by dropping all but the last term of the sum in the denominator. For the lower bound observe that,

$$\begin{aligned}\Gamma_\theta(x) &= \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}[\text{InflatedCost}(\theta, x, 0, \mathbb{Y})]}{\mathbf{E}[\text{InflatedTime}(\theta, x, \mathbb{Y})]} \\ &= \frac{e^\theta - 1}{e^\theta} \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}[\text{InflatedCost}(\theta, x, 0, \mathbb{Y})]}{\mathbf{E}[e^{\theta\,\text{Service}(x,\mathbb{Y})} - 1]} \\ &\geq \frac{e^\theta - 1}{e^\theta} \frac{\mathbf{E}[e^{\theta\,\text{Service}(x,\mathbb{X})}]}{\mathbf{E}[e^{\theta\,\text{Service}(x,\mathbb{X})} - 1]} \\ &\geq \frac{e^\theta - 1}{e^\theta}.\end{aligned}$$

□

Corollary C.2. *For any $x \in \mathbb{X}$ and all $\gamma \geq \theta > 0$,*

$$\frac{1}{\theta} \log\left(\frac{e^\theta}{e^\theta - 1}\right) \geq b_{\theta\text{-Gittins}}(x) \geq 0.$$

Proof. Observe that by Lemma C.1,

$$\begin{aligned}b_{\theta\text{-Gittins}}(x) &= \frac{1}{\theta} \log(\Gamma_\theta(x)) + \frac{1}{\theta} \log\left(\frac{e^\theta}{e^\theta - 1}\right) \\ &\geq \frac{1}{\theta} \log\left(\frac{e^\theta - 1}{e^\theta}\right) + \frac{1}{\theta} \log\left(\frac{e^\theta}{e^\theta - 1}\right) = 0,\end{aligned}$$

and,

$$\begin{aligned} b_{\theta\text{-Gittins}}(x) &= \frac{1}{\theta}\log(\Gamma_\theta(x)) + \frac{1}{\theta}\log\left(\frac{e^\theta}{e^\theta - 1}\right) \\ &\leq \frac{1}{\theta}\log(1) + \frac{1}{\theta}\log\left(\frac{e^\theta}{e^\theta - 1}\right) \\ &= \frac{1}{\theta}\log\left(\frac{e^\theta}{e^\theta - 1}\right). \end{aligned}$$

□

Lemma C.3. *For all $x \in \mathbb{X}$, $b_{\theta\text{-Gittins}}(x)$ is non-increasing in θ.*

Proof. Observe that,

$$\begin{aligned} b_{\theta\text{-Gittins}}(x) &= \frac{1}{\theta}\log(\Gamma_\theta(x)) + \frac{1}{\theta}\log\left(\frac{e^\theta}{e^\theta - 1}\right) \\ &= \frac{1}{\theta}\log\left(\frac{e^\theta - 1}{e^\theta}\sup_{\mathbb{Y}\subseteq\mathbb{X}}\frac{\mathbf{E}[\mathrm{InflatedCost}(\theta, x, 0, \mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]}\right) + \frac{1}{\theta}\log\left(\frac{e^\theta}{e^\theta - 1}\right) \\ &= \frac{1}{\theta}\log\left(\sup_{\mathbb{Y}\subseteq\mathbb{X}}\frac{\mathbf{E}[\mathrm{InflatedCost}(\theta, x, 0, \mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]}\right) \\ &= \frac{1}{\theta}\log\left(\sup_{\mathbb{Y}\subseteq\mathbb{X}}\frac{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x, \mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]}\right). \end{aligned}$$

So we will be done if we can show that

$$\frac{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x, \mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]}$$

is non-increasing in θ. Observe that,

$$\begin{aligned} &\frac{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x, \mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]} \\ &\quad= \frac{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1 + 1 - e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}(1 - \mathrm{Completed}(x, \mathbb{Y}))]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]} \\ &\quad= 1 + \frac{\mathbf{E}[1 - e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}(1 - \mathrm{Completed}(x, \mathbb{Y}))]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})} - 1]}, \end{aligned}$$

which is clearly non-increasing in θ since the numerator is non-increasing in θ and the denominator is non-decreasing in θ. □

Lemma C.4. *For all trajectories $x_{0:u} \in \mathbb{X}^{\mathrm{traj}}$, $\underline{\Gamma}_\theta(x_{0:u})$ and $\underline{b_{\theta\text{-Gittins}}}(x_{0:u})$ are left-continuous in θ on the interval $\gamma \geq \theta > 0$.*

Proof. Fix an arbitrary $x \in \mathbb{X}$. We first prove that $\Gamma_\theta(x)$ and $b_{\theta\text{-Gittins}}(x)$ are left-continuous in θ. Observe that for any $\gamma \geq \theta' > 0$,

$$\lim_{\theta\to\theta'}\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x, \mathbb{Y})] = \mathbf{E}[e^{\theta'\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x, \mathbb{Y})]$$

by the dominated convergence theorem and the fact that $\mathbf{E}[e^{\gamma S}] < \infty$. Similarly,

$$\lim_{\theta\to\theta'}\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}] = \mathbf{E}[e^{\theta'\,\mathrm{Service}(x,\mathbb{Y})}].$$

This means that

$$\frac{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x,\mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}-1]}$$

is continuous (and thus lower semicontinuous) since the denominator is positive for all $\theta > 0$. A standard fact about lower semicontinuous functions is that the supremum of an arbitrary family of lower semicontinuous functions is lower semicontinuous. Thus,

$$\sup_{\mathbb{Y}\subseteq\mathbb{X}} \frac{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}\,\mathrm{Completed}(x,\mathbb{Y})]}{\mathbf{E}[e^{\theta\,\mathrm{Service}(x,\mathbb{Y})}-1]},$$

and therefore also $\Gamma_\theta(x)$, are lower semicontinuous in θ on the interval $\gamma \geq \theta > 0$. Since $\Gamma_\theta(x)$ is non-decreasing in θ (Lemma D.7), this means that $\Gamma_\theta(x)$, and therefore, $b_{\theta\text{-Gittins}}(x)$, are left-continuous in θ.

Now fix a trajectory $x_{0:u} \in \mathbb{X}^{\mathrm{traj}}$ and recall that,

$$\underline{\Gamma}_\theta(x_{0:u}) = \min_{x\in x_{0:u}} \Gamma_\theta(x).$$

A minimum over a finite set of left-continuous functions is left-continuous, so $\underline{\Gamma}_\theta(x_{0:u})$, and thus also $\underline{b_{\theta\text{-Gittins}}}(x_{0:u})$, are left-continuous on the interval $\gamma \geq \theta > 0$. □

D Multi-armed Bandits with Costs and Inflation

In this section we prove that a Gittins policy is optimal for a modification of the classical discounted-reward Markovian multi-armed bandit (MAB) problem that generalizes the arrival-free batch problem described in Section 3. In particular, we consider the inflated-cost Markovian MAB with jobs problem. This is the same as the classical MAB problem except for the following three modifications:

(1) Rather than there being n arms, there are n Markov jobs (Section 2.1).
(2) Serving a job incurs a random cost *at the end of the time step* instead of earning a random reward *at the start of the time step.*
(3) Costs incurred in the future are inflated rather than discounted. That is, a cost incurred k time steps in the future would cost β^k times as much as if that same cost were incurred at the current time step, where now $\beta > 1$.

Our proof builds on the classical economic argument introduced by Weber [29]. Thus, our exposition highlights the differences while succinctly presenting the well-established aspects of the proof. We encourage readers unfamiliar with the economic argument, or those who would like a refresher, to review Weber [29] before proceeding.

D.1 The Model

The decision maker (DM) is presented with n independent Markov jobs (Section 2.1) that must be served to completion. Let $x_i(t)$ denote the state of job i at time $t = 0, 1, \ldots$. Each job i is associated with a cost function, $W_i(x, x')$, that maps each ordered pair of states to a non-negative distribution. If the DM serves job i at time t:

(a) the DM may not serve any other job at time t,
(b) the DM incurs a cost sampled from $W_i(x_i(t), x_i(t+1))$ at the end of the time step, i.e. at time $t + 1$,
(c) the state of job i updates according to the known Markovian dynamics,
(d) the state of all other jobs remains unchanged.

All costs are inflating over time at rate $\beta > 1$, so the expected total-inflated cost incurred under policy π is

$$\mathbf{E}\left[\sum_{t=1}^{T} \beta^t w_{\pi(t)}\left(x_{\pi(t)}(t)\right)\right] \tag{D.1}$$

where T is the sum of the sizes of the jobs, $\pi(t)$ denotes the job served at time step t under policy π, and $w_i(x_i(t)) \sim W_i(x_i(t), x_i(t+1))$. We assume that (D.1) is finite for all policies π that serve a job every time step. The goal is to determine the policy π that minimizes (D.1).

D.2 Differences From the Classical Economic Argument

In the economic argument, the first step is to consider only a single arm and modify the game so that playing the arm forever results in zero expected total-discounted reward. Implicit in that construction is the fact that the DM's local goal (when considering only one arm) is to maximize discounted reward since their global goal (when considering all arms) is also to maximize discounted reward. However, when considering inflated costs, the DM wishes to locally maximize inflated costs so as to globally minimize inflated costs. This is because with inflated costs (under the assumption that all costs must be paid eventually, as is the case here), minimizing costs involves paying the largest costs as early as possible, thus minimizing the amount of time they inflate. Therefore, we construct a modified single job game assuming that the DM's goal is to incur as much cost as possible.

D.3 Optimality of the Gittins Index for Multi-Armed Bandits with Inflation

Consider a modified situation where:

- There is only a single job.
- At each time step, the DM may choose to serve the job or stop forever.
- Each time the DM serves the job, they earn a fixed reward ξ, which we refer to as the *prevailing reward.*
- The DM wishes to incur as much cost as possible.

Clearly, since the DM wishes to maximize incurred cost, the DM will only agree to continue playing at a given state x if the expected future-inflated cost is non-negative for some stopping policy:

$$\sup_{\tau \geq 1} \mathbf{E}\left[\sum_{t=t}^{\tau} \beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] \geq 0.$$

Define the *maximal acceptable reward at state* x, denoted $\Gamma(x)$, to be the maximal prevailing reward such that the DM agrees to continue serving the job, that is,

$$\Gamma(x) = \sup\left\{\xi \in \mathbb{R} : \sup_{\tau \geq 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] \geq 0\right\}.$$

Remark D.1. An equivalent formulation (that we use in Section 3.1),

$$\Gamma(x) = \sup_{\mathbb{Y} \subseteq \mathbb{X}} \frac{\mathbf{E}\left[\sum_{t=1}^{S(\mathbb{Y})} \beta^t (w(x(t))) \,\middle|\, x(0) = x\right]}{\mathbf{E}\left[\sum_{t=1}^{S(\mathbb{Y})} \beta^t \,\middle|\, x(0) = x\right]}$$

where $\mathbb{X}$ is the state space of the job and $S(\mathbb{Y})$ is the first time the bandit's state exits $\mathbb{Y}$. This expression follows from straightforward manipulations of the above definition and the fact that stopping times on Markov chains are defined by a continuation set $\mathbb{Y} \subseteq \mathbb{X}$.

Now imagine that we set the initial prevailing reward to be $\Gamma(x(0))$ and that each time the DM reaches a state such that they prefer to stop, we decrease the prevailing reward to the maximal acceptable reward of the current state. In doing so, we ensure that the DM is willing to continue serving the job until completion. Furthermore, by the definition of the maximal acceptable reward, the DM would experience a fair game,

$$\mathbf{E}\left[\sum_{t=1}^{T} \beta^t (w(x(t)) - \underline{\Gamma}(t))\right] = 0 \tag{D.2}$$

where $\underline{\Gamma}(t) = \min_{0 \le u \le t} \Gamma(x(u))$ and T is the size of the job.

We now wish to consider the situation where the DM may not stop forever, but may instead stop serving the job for any finite number of time steps. These pauses will later correspond to the DM serving other jobs. We will show that when pauses are allowed, the expected total-inflated cost is greater than or equal to zero (in contrast, it equals zero when pausing isn't allowed, as expressed by (D.2)).

To prove this, we consider two cases:

(1) The DM pauses at a time t when $\underline{\Gamma}(t) = \Gamma(x(t))$.
(2) The DM pauses at a time t when $\underline{\Gamma}(t) < \Gamma(x(t))$.

In the first case, we know that if the DM does not pause at time t and instead continues serving the job to completion, the expected inflated-cost incurred from that point onwards is zero (by the definition of $\Gamma(x(t))$). Clearly, the extra inflation caused by pausing at time t does not change this and so the expected total-inflated cost incurred is zero as it is unaffected by the pause.

In the second case, if the DM does not pause at time t and instead continues serving the job to completion, the expected inflated cost incurred from that point onwards is positive. Thus, the extra inflation caused by pausing at time t increases the expected total-inflated cost incurred, and so pausing at such a time leads to a positive expected total-inflated cost.

The above argument only considers a single pause, but it is straightforward to extend it to any number of pauses by induction. Putting everything together we conclude that:

(1) The expected total-inflated cost is greater than or equal to zero with or without pauses.
(2) The expected total-inflated cost is equal to zero if and only if pauses are only taken when the prevailing reward equals the maximal acceptable reward of the current state.

This motivates the following definition.

Definition D.2. We say that a policy π is *insulated* if the DM only pauses at times when the prevailing reward equals the maximal acceptable reward of the current state. Equivalently, in the inflated-cost MAB with jobs problem, we say that a policy is insulated if the DM only switches the job they are serving in states where the prevailing reward of the previously served job is equal to the maximal acceptable reward of its current state.

We can now formalize our results for the single-job game as the following lemma:

LEMMA D.3. *For a single job and all policies π,*

$$\mathbf{E}_\pi\left[\sum_{u=1}^{S} \beta^{T(u)} w(x(u))\right] \ge \mathbf{E}_\pi\left[\sum_{u=1}^{S} \beta^{T(u)} \underline{\Gamma}(u)\right]$$

where $T(u)$ is the time at which the job is served for the u-th time following policy π. Furthermore, equality holds if and only if π is insulated.

To complete the proof, we now return to the inflated-cost MAB with jobs problem. Set the prevailing reward of each job to be the maximal acceptable reward of its initial state. We now make 3 key observations.

Observation D.4. The policy which always serves the job with greatest prevailing reward is equivalent to the policy which always serves the job with greatest maximal acceptable reward. Furthermore, both of these policies are insulated.

Observation D.5. Since the policy which always serves the job with greatest prevailing reward is insulated, the expected total-inflated reward earned by the policy is equal to the expected total-inflated cost of the policy.

Lastly, since the prevailing rewards are non-increasing, there is inflation, and every job must be served to completion,

Observation D.6. The policy which always serves the job with greatest prevailing reward minimizes the total-inflated reward earned.

Using these three observations together, it is straightforward to show the desired result. First, note that by applying Lemma D.3 to each job individually, we get that the minimal possible expected total-inflated reward earned is a lower bound on the expected total-inflated cost of any policy. The three observations above together show that the expected total-inflated cost earned by the policy that always serves the job with greatest maximal acceptable reward attains this lower bound. Noting that the maximal acceptable reward is a variant of the Gittins index, we conclude that a variant of the Gittins policy minimizes expected total-inflated cost.

D.4 Properties of the Gittins Index

The following lemma is a inflation version of a classical result from the Gittins index literature [9]. The proof is included here for completeness.

Lemma D.7. *For all states $x \in \mathbb{X}$, $\Gamma(x)$ is non-decreasing in the inflation factor β.*

Proof. Let $\Gamma_\beta(x)$ be the Gittins index with inflation factor β. We wish to prove that for all $\alpha < \beta$ and $x \in \mathbb{X}$, $\Gamma_\alpha(x) \le \Gamma_\beta(x)$. Recall that,

$$\Gamma_\beta(x) = \sup\left\{\xi \in \mathbb{R} : \sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] \ge 0\right\}$$

so it will be sufficient to prove that

$$\sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] \ge \sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \alpha^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right].$$

For any stopping time τ we can define a geometrically distributed stopping time σ that is independent of τ such that $\mathbf{P}[\sigma \ge t] = (\alpha/\beta)^t$. Thus,

$$\begin{aligned}\sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] &\ge \sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau\wedge\sigma} \beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] \\ &= \sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \mathbf{P}[\sigma \ge t]\beta^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right] \\ &= \sup_{\tau \ge 1} \mathbf{E}\left[\sum_{t=1}^{\tau} \alpha^t (w(x(t)) - \xi) \,\middle|\, x(0) = x\right].\end{aligned}$$ □

Received January 2025; revised April 2025; accepted April 2025