Approximations and Control for State-dependent Limited Processor Sharing Queues

Varun Gupta
Booth School of Business
University of Chicago
varun.gupta@chicagobooth.edu

Jiheng Zhang
Department of Industrial Engg. and Logistics Management
The Hong Kong University of Science and Technology
j.zhang@ust.hk

Abstract

The paper studies approximations and control of a processor sharing (PS) server where the service rate depends on the number of jobs occupying the server. The control of such a system is implemented by imposing a limit on the number of jobs that can share the server concurrently, with the rest of the jobs waiting in a first-in-first-out (FIFO) buffer. A desirable control scheme should strike the right balance between efficiency (operating at a high service rate) and parallelism (preventing small jobs from getting stuck behind large ones).

We employ the framework of *heavy-traffic* diffusion analysis to devise control heuristics for such a queueing system. While typical studies of diffusion control of state-dependent queueing systems begin with a given asymptotic scaling and an exogenously defined drift function, our main contribution is a method to engineer a drift function starting from the discrete (pre-limit) state-dependent PS server with the aim of obtaining a control policy for the latter. We establish steady-state distribution of the resulting diffusion, and use it to obtain insightful and closed-form approximations for the original system under a static concurrency control policy.

Finally, we extend our study to control policies that dynamically adjust the concurrency level and provide a novel numerical algorithm tailored to solve the associated diffusion control problem. Numerical experiments demonstrate the accuracy of our approximation for choosing optimal or near-optimal static and dynamic concurrency control heuristics.

1 Introduction

Consider an *emergency room* where doctors, nurses, and diagnostic equipment make up a shared resource for admitted patients. It has been empirically observed that the service rate of such service systems is state-dependent (e.g., [5]). Human operators tend to speed up service when there is congestion. As another example, consider a typical *web server or an online transaction processing system*. In such resource sharing systems, as the number of tasks (also called active threads) concurrently sharing the server increases, the server throughput initially increases due to more efficient utilization of resources. However, as the server switches from one task to another, it needs to make room for the new task's data in its cache memory by evicting an older task's data (only to fetch it again later). Without a limit on the number of concurrent tasks, this contention for the limited memory can lead to a phenomenon called *thrashing* which causes the system throughput to drop drastically (e.g., [2, 6, 11, 12, 23, 40]).

The resource sharing system examples we have described above fall into the category of the so-called *State-dependent Limited Processor Sharing* (Sd-LPS) systems. To specify an Sd-LPS system, we begin with a processor sharing (PS) server whose service rate varies as a function of the number of jobs at the server. For example,

$$\mu(1) = 1, \mu(2) = 1.5, \mu(3) = 1.25, \mu(4) = 1, \mu(5) = 0.75, \dots \tag{1}$$

When there are n jobs at the PS server, each job gets served at a rate of $\frac{\mu(n)}{n}$ jobs/second. To ensure efficient operation, we impose a limit on the maximum number of jobs that can be served in parallel. We call this the concurrency limit, K. Arriving jobs that find the server busy with K jobs wait in a first-come-first-served (FCFS) buffer. A *static concurrency control policy* is one where the concurrency limit is independent of the state. If the concurrency level can vary with the system state (e.g., the queue length of the FCFS buffer), we call it a *dynamic concurrency control policy.*

To understand the tradeoff involved in choosing the optimal concurrency level, suppose there are 3 jobs in the system described above. Even though the server is capable of serving at an aggregate rate of 1.5 jobs/second by limiting the concurrency level to 2, we may choose to increase the concurrency level to 3 and operate below peak capacity. Why might we want to do that? It is well known that if the job size distribution has high variability, then pure PS outperforms FCFS scheduling by allowing small jobs to overtake large ones. Therefore, it may be beneficial to increase the concurrency level beyond the peak efficiency even if some capacity will be lost. Similarly, for job size distributions with low variability, it may be beneficial to operate at $K = 1$. Thus Sd-LPS systems are not "work-conserving" queueing systems.

Contributions

Naturally, our goal is to choose the 'best' concurrency control policy. In this work we aim to develop a diffusion approximation framework for Sd-LPS queues, and to utilize the proposed diffusion approximation to find concurrency control policies that minimize the mean sojourn time. This immediately leads to the question: *Given that we want to control the state-dependent PS server (exemplified by* (1)*), what is a 'meaningful' asymptotic scaling to arrive at a diffusion approximation?*

While there are some works on heavy-traffic asymptotics for queues with state-dependent rates, they either (*i*) assume a sequence of systems with exogenously given limiting drift functions to be given whereas we begin with a discrete PS server of the kind shown in (1) and engineer the limiting drift function, or (*ii*) are limited to models where the server can only serve one job at a time whereas multiple jobs are processed in parallel by the PS server; or (*iii*) only analyze a Jackson network type of system and do not solve a diffusion control problem. The present paper fills these gaps in the literature.

Our main contributions are as follows:

1. We propose a method to "reverse-engineer" a sequence of Sd-LPS queueing systems starting with a discrete state-dependent PS server that yields a limiting state-dependent drift function. The crucial part is that the effect of the entire service rate curve shows up in the limiting diffusion process. All prior literaure on diffusion analysis of state-dependent queues assumes that the drift function is given exogeneously.

2. We propose an approximation for the distribution of the number of jobs in the Sd-LPS system for a static concurrency limit under a GI arrival process and GI job sizes. This

approximation is used to choose a near-optimal static concurrency limit to minimize any cost that is a function of the number of jobs in the system.

3. We extend our framework by proposing a more general scaling for developing dynamic (state-dependent) control policies and present a numerical algorithm tailored to solving the resulting diffusion control problem. Our simulation experiments show that the dynamic policies based on diffusion control perform remarkably close to the true optimal dynamic policies (for input distributions where the true optimal policy can be computed numerically).

Related work on control of LPS systems

The literature on LPS-type systems has mostly focused on the constant rate LPS queue where the server speed is independent of the state. Yamazaki and Sakasegawa [42] show qualitatively the effect of increasing the concurrency level on the mean sojourn time for NWU (New-Worse-than-Used) and Erlang job size distributions. Avi-Itzhak and Halfin [4] derive an approximation for the mean sojourn time for the constant rate LPS queue with $M/GI/$ input process, while Zhang and Zwart [45] derive one for $GI/GI/$ input. Nair et al. [34] expose the power of LPS scheduling by analyzing the tail of sojourn time under light-tailed and heavy-tailed job size distributions. They prove that with an appropriate choice of the concurrency level as a function of the load, LPS queues can achieve robustness to the distribution of job sizes (their tail to be precise).

For Sd-LPS queues, Rege and Sengupta [38] derive expressions for the moments and distribution of the sojourn time under $M/M/$ input. Gupta and Harchol-Balter [18] propose an approximation for the mean sojourn time for $GI/GI/$ input by approximating the interarrival times and job size distribution by the tractable degenerate hyperexponential distribution. They also propose heuristic dynamic admission control policies under $M/GI/$ input.

In this paper, we propose the first diffusion approximation for Sd-LPS queues with a GI/GI input and a static concurrency level. In addition, we propose the first heuristic dynamic admission control policies for Sd-LPS queues.

Related work on control of queueing systems

There is a considerable literature on the control of the arrival and service rates of queueing systems, but the majority of this work focuses on control of $M/M/1$ or $M/M/s$ systems via Markov decision process formulation, e.g., [1, 3, 16, 31]. Ward and Kumar [39] look at the diffusion control formulation for admission control in a $GI/GI/1$ with impatient customers. Our model differs significantly from those in the literature: in our model, the space of actions is the number of jobs admitted to the PS server and is therefore state-dependent.The state-dependence of the action space means that the value function may not even be monotonic in the state. We establish this result for our problem and present a simple criterion under which monotonicity holds for general control problems with state-dependent action spaces (see proof of Proposition 3). In addition, the rather arbitrary nature of the service rate curve precludes elegant structural results for the optimal value function which leads us to propose novel and efficient numerical algorithms for solving the resulting diffusion control problem.

Related work on heavy-traffic analysis of systems with state-dependent rates

Our heavy-traffic scaling is most closely related to the recent work of Lee and Puhalskii [29], who analyze a queueing network of FCFS queues in the critically loaded regime and under non-Markovian arrival and service processes. Yamada [41] also analyzes Markovian state-dependent queueing networks under a similar scaling of state-dependent service and arrival rates. Whereas [29, 41] assume an exogenously given limiting drift function, we propose a method to calculate it from the finite queueing system which is the object of the control problem. Further, the scheduling policy we consider is Processor Sharing. Other works on analysis of heavy-traffic asymptotics of state-dependent Markovian queues include Krichagina [26], Mandelbaum and Pats [32], Janssen et al. [24].

Outline

In Section 2 we present details of the Sd-LPS model, introduce the notation used in the paper, and describe our approach towards arriving at the asymptotic regime for diffusion analysis. In Section 3, we present our results on diffusion approximation for the Sd-LPS queue under a static concurrency control policy. We defer the proofs of convergence to the appendix. In Section 4 we turn to dynamic concurrency control policies for the Sd-LPS queue by setting up a diffusion control problem, and present a novel numerical algorithm to solve the diffusion control problem. We make our concluding remarks in Section 5.

2 Model and Diffusion Scaling

2.1 Stochastic model and Notation

We begin with a description of the Sd-LPS system for which we want to find the optimal control. Let $X(t)$ denote the total number of jobs in the system at time t. The control of such a system is implemented by imposing a concurrency limit K. Only $Z(t) = X(t) \wedge K$ jobs are in service and server capacity of $\mu(Z(t))$ is shared equally among the jobs. The remaining $Q(t) = (X(t) - K)^+$ jobs wait in a FCFS queue. A job, once in service, stays in service until completion. The rate of the server $\mu(Z(t))$ is understood to be the speed at which it drains the workload. So the *cumulative service amount* a job in service can receive from time s to t is

$$S(s,t) = \int_s^t \psi(Z(\tau))d\tau, \tag{2}$$

where

$$\psi(z) = \begin{cases} \frac{\mu(z)}{z}, & \text{if } z \geq 0, \\ 0, & \text{if } z = 0. \end{cases} \tag{3}$$

Without loss of generality, we assume that there is no intrinsic limit on the number of jobs the server can serve as we can set the service rate to 0 to model such a limit. Note that for the regular state-independent system whose service rate $\mu(\cdot)$ is a constant, say 1, $\frac{\mu(z)}{z}$ in the above will simply become $1/z$. The state-dependent service rate makes the system *non*-work-conserving, which brings a fundamental challenge to their study. For a single server system with constant service rate, any non-idling policy is work-conserving, meaning that workload will be drained at constant speed as long as there is any workload in the system. Since workload arrives according to a renewal process, work-conserving systems can be approximated by reflected Brownian motions with a constant drift in heavy traffic regimes, e.g, [10]. Existing studies of PS, e.g., [17] and LPS, e.g., [44] systems crucially rely on the fact that the system is work-conserving, which implies that the workload process is equivalent to that of a simple $G/G/1$ queue. However, this is not the case for our Sd-LPS model.

The number of job arrivals in time $[0, t]$ is denoted by $\Lambda(t)$. We assume that $\Lambda(\cdot)$ is a renewal process with rate λ, and c_a^2 denotes the squared coefficient of variation (SCV) for the *i.i.d.* inter-arrival times. The system is allowed to be non-empty initially. We index jobs by $i = -X(0)+1, -X(0)+2, \dots, 0, 1, \dots$. The first $X(0)$ jobs are initially in the system, with jobs $i = -X(0)+1, \dots, -Q(0)$ in service and jobs $i = -Q(0)+1, \dots, 0$ waiting in the queue. Arriving jobs are indexed by $i = 1, 2, \dots$. The size of the ith job is denoted by v_i. We assume job sizes are *i.i.d.* random variables with mean size m (in the chosen unit of measuring work) and SCV c_s^2. Jobs leave the system once the cumulative amount of service they have received from the server exceeds their job sizes.

In this study, we are interested in how the system performance (e.g., expected number of jobs in steady state) depends on the state-dependent service rate function $\mu(\cdot)$, the parameters $(\lambda, c_a^2, m, c_s^2)$ of the stochastic primitives, and the concurrency level K, which is a decision variable we can control and optimize.

Measure-valued state descriptor

Analyzing the stochastic processes underlying the Sd-LPS model with generally distributed service times requires tracking of more information about the system state than just the number of jobs. Following the framework in [43, 44], we introduce a *measure-valued* state descriptor to describe the full state of the system. At any time t and for any Borel set $A \subset (0, \infty)$, let $\mathcal{Q}(t)(A)$ denote the total number of jobs in the buffer whose job size belongs to A and $\mathcal{Z}(t)(A)$ denote the total number of jobs in service whose residual job size belongs to set A. Thus, $\mathcal{Q}(\cdot)$ and $\mathcal{Z}(\cdot)$ are measure-valued stochastic processes. Let δ_a denote the Dirac measure of point a on $\mathbb{R}$ and $A + y \doteq \{a + y : a \in A\}$. By introducing the measure-valued processes, we can characterize the evolution of the system via the following *stochastic dynamic equations*:

$$\mathcal{Q}(t)(A) = \sum_{i=B(t)+1}^{\Lambda(t)} \delta_{v_i}(A), \tag{4}$$

$$\mathcal{Z}(t)(A) = \mathcal{Z}(0)(A + S(0,t)) + \sum_{i=B(0)+1}^{B(t)} \delta_{v_i}(A + S(\tau_i, t)), \tag{5}$$

where τ_i is the time when the ith job starts to receive service and

$$B(t) = \Lambda(t) - Q(t), \tag{6}$$

which can be intuitively interpreted as the index of the last job to enter service by time t. The number of jobs in the FCFS queue, $Q(t)$, and in service, $Z(t)$, can be represented using the measure-valued descriptors as follows:

$$Q(t) = \langle 1, \mathcal{Q}(t) \rangle, \quad Z(t) = \langle 1, \mathcal{Z}(t) \rangle.$$

where $\langle f, \nu \rangle$ denotes the integral of a Borel measurable function $f : \mathbb{R}_+ \to \mathbb{R}$ with respect to a measure ν. Let $W(t)$ denote the workload of the system at time t which is defined as the sum of the sizes of all jobs in queue and the remaining sizes of all jobs in service. Due to the varying service rate of the server, the dynamics of the workload process is represented by

$$W(t) = W(0) + \sum_{i=1}^{\Lambda(t)} v_i - \int_0^t \mu(Z(s)) 1_{\{W(s)>0\}} ds. \tag{7}$$

Again, we can express the workload $W(t)$ in terms of the measure-valued descriptors:

$$W(t) = \langle \chi, \mathcal{Q}(t) + \mathcal{Z}(t) \rangle, \tag{8}$$

where χ denotes the identity function on $\mathbb{R}$.

2.2 Proposed Asymptotic Regime for Diffusion Approximation of Sd-LPS systems

We refer to the system introduced in Section 2.1 as our *original* system. We now propose an asymptotic regime where a sequence of Sd-LPS systems, parametrized by $r \in \mathbb{Z}^+$, will be studied under an appropriate scaling. The objective is to a obtain a meaningful approximation of the original system with the goal of choosing the 'best' concurrency control policy. This leads to the question:

> *What is the appropriate scaling to analyze the Sd-LPS queue? That is, what asymptotic regime captures the entire service-rate curve of the original Sd-LPS system, and thus can be used to find a near-optimal concurrency limit?*

As we mentioned earlier, the scaling we develop is very close to the scaling proposed by Yamada [41] and Lee et al. [29]. To motivate why this is the appropriate scaling for Sd-LPS systems we begin by examining two special cases of Sd-LPS systems and the motivation behind asymptotic scaling used to study them: (i) multiserver systems, and (ii) the constant rate LPS queue.

The $G/GI/k$ multiserver system A $G/GI/k$ multiserver system with a service rate of μ jobs/second per server and a central buffer can be viewed as an Sd-LPS system with $\mu(n) = n\mu$ and a concurrency limit of $K = k$. One of the most common modern asymptotic regimes in which $G/GI/k$ systems are studied is the Halfin-Whitt regime (also called square-root staffing rule, or the quality-and-efficiency-driven regime) starting with [20] and more recently [37], [14]. Here one fixes μ and creates a sequence of multiserver systems parametrized by r, where the number of servers grows according to $k^{(r)} = rk$ while the mean arrival rate $\lambda^{(r)}$ increases so that $\frac{k^{(r)}\mu - \lambda^{(r)}}{\sqrt{k^{(r)}}} \to \beta$. The key insight that motivates the Halfin-Whitt regime is that in the limiting system the probability that an arrival gets blocked converges to a non-degenerate limit (bounded away from 0 and 1). In that sense, the behavior one desires from a well-designed system that there is not too much or too little blocking survives the asymptotic scaling.

State-independent (constant rate) LPS queue In the state-independent LPS queue, the service rate of the server is a constant μ irrespective of the number of jobs at the server, and there is a fixed concurrency limit k. Recently, Zhang et al. [44] have proposed and analyzed a diffusion approximation for the LPS system where a sequence of LPS systems (parametrized by r) is devised so that the service rate remains fixed at μ, the concurrency limit increases according to $k^{(r)} = rk$ and the arrival rate increases so that $k^{(r)}(\mu - \lambda^{(r)}) \to \theta$, a constant. As in the Halfin-Whitt regime for the multiserver systems, under the proposed scaling for LPS systems the probability that an arrival finds all slots at the PS server occupied converges to a limit bounded away from 0 and 1. In addition, the queue length scaled by $\frac{1}{k^{(r)}}$ also converges to a non-degenerate distribution, unlike Halfin-Whitt where the queue lengths are smaller and must be scaled by $\frac{1}{\sqrt{k^{(r)}}}$.

It is not obvious how either of these scalings can be extended to the Sd-LPS system, but we borrow the philosophy that under a good scaling, the limiting system should in some sense be a faithful proxy for the original system. As an example of a regime that is not quite faithful enough, consider the following: We scale the concurrency limit as $k^{(r)} = rk$, leave the mean arrival rate λ unchanged, and 'stretch' the service rate curve so that for the rth Sd-LPS system, $\mu^{(r)}(rx) = \hat{\mu}(x)$ where $\hat{\mu}(\cdot)$

is a continuous interpolation of $\mu(\cdot)$. The limiting system here would correspond to a fluid limit where the steady state 'gets stuck' around x^*, where $\hat{\mu}(x^*) = \lambda$, and the rest of the service rate curve plays almost no part. This fluid regime cannot be used to devise a control policy for the original Sd-LPS system.

Instead, we propose an approach where we fix a desired limiting behavior of the sequence of Sd-LPS systems, and then reverse engineer the service rate curves which guarantee this behavior in the asymptotic limit. A suitable choice of the desired limiting behavior ensures that the effect of the entire service rate curve is preserved during the asymptotic analysis.

Desiderata for the Sd-LPS asymptotic scaling: We construct a sequence of Sd-LPS systems parametrized by $r \in \mathbb{Z}^+$ such that the rth system has a concurrency level of

$$k^{(r)} = rK. \tag{9}$$

Further, the sequence of service rate curves $\mu^{(r)}(\cdot)$ is constructed so that under a Poisson arrival process with rate λ and *i.i.d.* Exponentially distributed job sizes with mean size m (i.e., under $M/M/$ input), the distribution of the scaled number of jobs in the system (scaled by $\frac{1}{k^{(r)}}$) converges to that of the original Sd-LPS system under the same $M/M/$ input and concurrency level K.

Engineering $\mu^{(r)}$ to satisfy the desiderata: Since we start by fixing the concurrency levels for the sequence of Sd-LPS systems, the only design flexibility we have to satisfy the scaling axioms is the choice of state-dependent service rate curves. Let us denote the steady-state distribution of the number of jobs in the rth Sd-LPS system under $M/M/$ input and service rate curve $\mu^{(r)}$ by $F^{(r)}$. Our goal is to find the sequence $\mu^{(r)}(\cdot)$ so that

$$\lim_{r\to\infty} F^{(r)}(\lceil rx \rceil) = \hat{F}(x) \quad \forall x \in [0, \infty) \tag{10}$$

for some distribution function $\hat{F}(\cdot)$. This gives us our **first way** of deriving the scaling: Fix $\hat{F}(\cdot)$ to be a continuous, differentiable, strictly increasing interpolation of the steady-state distribution of the number of jobs for the original system under $M/M/$ input and reverse-engineer the sequence of service rate functions $\mu^{(r)}(i)$. The requisite service rate functions satisfy

$$\lim_{r\to\infty} r\left(\lambda m - \mu^{(r)}(\lceil rx \rceil)\right) = \lambda m \frac{d \log f(x)}{dx} \quad \forall x \in [0, \infty). \tag{11}$$

where $f(x) = \frac{d}{dx}\hat{F}(x)$. To see why, consider the rth Sd-LPS system operating under $M/M/$ input. Let $\pi^{(r)}(i)$ be the probability mass function for the steady-state number of jobs in the rth system. Flow-balance equations imply

$$\frac{\pi^{(r)}(\lceil rx+1 \rceil)}{\pi^{(r)}(\lceil rx \rceil)} = \frac{\lambda m}{\mu^{(r)}(\lceil rx \rceil)}.$$

Since, by design, we want $r\pi^{(r)}(\lceil rx \rceil)$ to converge to the density function $f(x)$, we should have:

$$\frac{\lambda m}{\mu^{(r)}(\lceil rx \rceil)} \approx \frac{f\left(x + \frac{1}{r}\right)}{f(x)} \approx 1 + \frac{1}{r}\frac{f'(x)}{f(x)}.$$

Equivalently, $r(\lambda m - \mu^{(r)}(\lceil rx \rceil) \to \lambda m \frac{d \log f(x)}{dx}$.

Of course, by reverse-engineering the scaling, we guarantee ourselves a non-degenerate limit that is interesting in that it captures the effect of the entire $\mu(\cdot)$ function. Further, it turns out we never really need to compute the service rate functions $\mu^{(r)}(\cdot)$! In Section 3, we will show that we can directly express the limiting steady-state quantities in terms of the distribution $\hat{F}(\cdot)$, which can be easily obtained from that of the original system.

Since the method described above requires the distribution $F(\cdot)$ (and its smooth interpolation $\hat{F}(\cdot)$) for the original Sd-LPS system under a *given* static concurrency limit of K, it can only used to approximate the performance for static concurrency limits, and not to design dynamic control policies. To address this, we propose a **second way** of deriving the scaling, that still guarantees (10):

Begin with $\hat{\mu}(\cdot) : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ satisfying:

1. $\hat{\mu}(\cdot)$ agrees with $\mu(\cdot)$ at integer arguments: $\hat{\mu}(i) = \mu(i)$ for $i = \{1, 2, \ldots\}$,
2. $\hat{\mu}(\cdot)$ is continuous and differentiable.

The sequence of service rate functions $\{\mu^{(r)}(\cdot)\}$ is chosen to satisfy

$$\lim_{r\to\infty} r\left(\lambda m - \mu^{(r)}(\lceil rx \rceil)\right) = \lambda m \log \frac{\lambda m}{\hat{\mu}(x)} \quad \forall x \in [0, \infty). \tag{12}$$

To motivate the second proposal, note that if $\frac{\lambda m}{\mu^{(r)}(r\cdot x)} \approx 1 - \frac{\theta(x)}{\lambda m r} \approx e^{-\frac{\theta(x)}{\lambda m r}}$, then

$$\frac{\pi^{(r)}(ry)}{\pi^{(r)}(rx)} \approx e^{-\frac{1}{\lambda m}\int_x^y \theta(u)du} \to \frac{\pi(y)}{\pi(x)} = \prod_{i=x+1}^{y} \frac{\lambda m}{\mu(i)}.$$

Or,

$$-\frac{1}{\lambda m}\int_x^y \theta(u)du \approx \log \frac{\pi^{(r)}(ry)}{\pi^{(r)}(rx)} \approx \log \frac{\pi(y)}{\pi(x)} = \sum_{i=x+1}^{y} \log \frac{\lambda m}{\mu(i)}.$$

Comparing the first and last expressions above gives us an approximation for $\theta(u)$ in terms of a continuous extension $\hat{\mu}$ of μ: $r(\lambda m - \mu^{(r)}(\lceil rx \rceil) \to -\lambda m \log \frac{\lambda m}{\hat{\mu}(x)}$.

For either way of arriving at the diffusion scaling, we see that $r(\lambda - \mu^{(r)}(\lceil rx) \rceil)$ converges to a non-degenerate *drift function* $-\theta(x)$. In the first case the $\theta(x)$ function is reverse-engineered by fixing a limiting distribution and is more appropriate for approximating performance of static control policies. In the second case it is obtained more directly using a continuous extension of $\mu(i)$, and is more appropriate for computing dynamic control policies. In both cases there is limited flexibility in extending a discrete function to a continuous smooth function.

Comparison with existing diffusion scalings It is a useful exercise to compare how our proposed scaling compares with the conventional asymptotic scalings for the two examples of Sd-LPS systems we pointed at the beginning of this section: the $G/GI/k$ queue, and the constant rate LPS queue.

The Halfin-Whitt scaling for a $G/GI/k$ queue posits constructing a sequence of systems each of which is also a homogeneous multiserver system such that under $M/M/$ input the blocking probability of the sequence converges to a non-degenerate limit (for example, to the blocking probability of the finite system being approximated). If we use our proposed scaling to approximate a multiserver system, the sequence of Sd-LPS systems would not be a homogeneous multiserver system. Indeed, $r(\lambda m - \mu^{(r)}(rx))$ grows as $\log(x)$ for small x instead of linearly in x as in Halfin-Whitt. Which is better? Our answer is that it depends on the purpose of the asymptotic analysis. If the goal is to find a staffing level for a reasonably large system (e.g., where it would take at least tens of servers to serve the demand), then the Halfin-Whitt scaling captures the essential features. However, if the goal is admission/concurrency control for a multiserver system with a given number of servers that is not too large, then by capturing the entire distribution of number of jobs, not just the blocking probability, our proposed scaling could be more useful.

For the constant rate LPS system, our asymptotic scaling matches the diffusion scaling of Zhang et al. [44], and thus can be seen as an extension of their scaling to Sd-LPS.

3 Diffusion approximation for the Sd-LPS queue with a static concurrency level

The goal of this section is to provide approximations for the steady-state performance of the Sd-LPS queue with a static concurrency level under the proposed scaling (11). In Section 3.1 we first summarize the results of this section by giving an approximation for the mean number of jobs in an Sd-LPS system under a static concurrency level (equation (13)), and providing some simulation results which show the utility of the approximation for choosing a near-optimal concurrency level. In Section 3.2, we prove process-level limits for diffusion-scaled workload and head count processes. In Section 3.3, we justify using the steady state of the limiting processes as an approximation for the limit of the steady state of the diffusion-scaled processes by establishing the required interchange of limits. We also present closed-form formulae for these steady-state distributions. All the proofs for this section can be found in the appendix.

3.1 An approximation and simulation results

Let N denote the steady-state number of jobs in the Sd-LPS system for a given static concurrency level K. Our main result of this section yields the following simple approximation formula for the expectation of N as a function of the concurrency level and other system parameters (see Proposition 2 for the formal statement)

$$\mathbf{E}[N] \approx \frac{\sum_{n=0}^{\infty}(n \wedge K)\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}}{\sum_{n=0}^{\infty}\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}} + \left(\frac{c_s^2+1}{2}\right)\frac{\sum_{n=0}^{\infty}(n-K)^{+}\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}}{\sum_{n=0}^{\infty}\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}}, \tag{13}$$

where $\pi(n)$ denotes the steady-state probability of there being n jobs in the Sd-LPS system under $M/M/$ input (that is, Poisson arrivals with mean rate λ and *i.i.d.* Exponentially distributed job sizes with mean size m).

Figure 1 shows a hypothetical service rate function for a PS server. The service rate has the functional form $\mu(i) = 1.25 - \frac{i^2}{150}$, and is monotonically decreasing in the concurrency level. Figure 2 shows the simulation results for the steady-state mean number of jobs as a function of the concurrency level K. The arrival process is Poisson with mean arrival rate shown below the figures. We simulated three distributions, each with mean $m = 1$ and SCV $c_s^2 = 19$. The solid curve shows the diffusion approximation (13) for the mean number of jobs. For each value of λ and each distribution, the optimal concurrency level obtained via approximation (13) matches the one obtained from simulating the LPS system. As expected, a higher traffic intensity shifts the optimal concurrent level towards the efficient level $K^* = 1$. Note that while the proposed diffusion approximation accurately captures the *shape* of $\mathbf{E}[N]$ versus the concurrency level curve and thus provides good

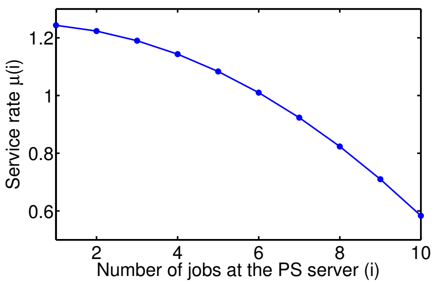
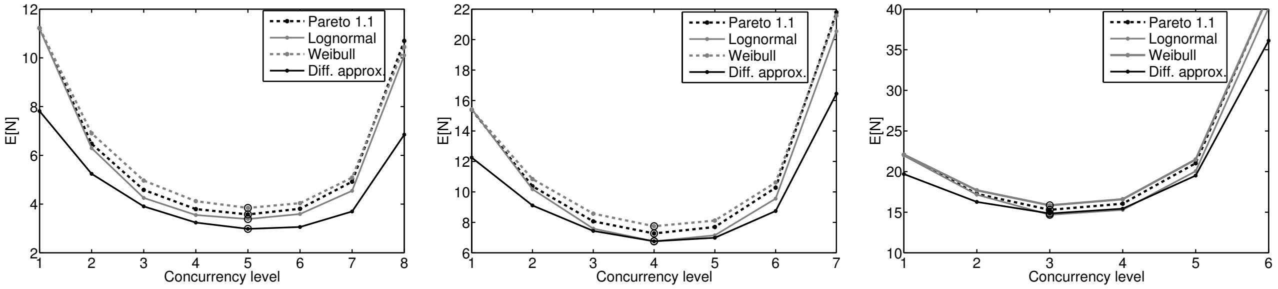


Figure 1: State-dependent service rate function used for simulation results

guidance for concurrency control, the actual numerical values for $\mathbf{E}[N]$ are not always very accurate for all values of K.



(a) $\lambda = 0.7$ (b) $\lambda = 0.8$ (c) $\lambda = 0.9$

Figure 2: Simulation results for mean number of jobs in the system versus the concurrency level for the service rate function shown in Figure 1 for various job size distributions, all with mean $m = 1$ and SCV $c_s^2 = 19$. The arrival process is Poisson with indicated mean arrival rate λ. Also shown is the diffusion approximation from equation (13). The optimal concurrency level for each curve is shown with a circle. (The confidence intervals are narrow enough that we have omitted them here, see Figure 5 in Appendix A for the plot with 95% confidence intervals.)

3.2 Diffusion analysis of Sd-LPS system

We now present the analysis of the Sd-LPS system under the asymptotic regime described in (11). For generality and notational convenience, we present all the analysis in terms of the general drift function $\theta(x)$, and then translate the result into a form involving $f(x)$ (Proposition 2) for convenience.

Consider the sequence of Sd-LPS systems indexed by r. We append a superscript (r) to all the quantities associated with the rth system. The concurrency level $k^{(r)}$ is specified as (9).

Assume that the arrival process $\Lambda^{(r)}(\cdot)$ satisfies

$$\frac{\Lambda^{(r)}(r^2t) - r^2\lambda t}{r} \Rightarrow M_a(t), \quad \text{as } r \to \infty, \tag{14}$$

where $M_a(\cdot)$ is a Brownian motion with zero drift and variance c_a^2. Further, we assume that the sizes of arriving jobs follow distribution G which satisfies

$$G \text{ is a continuous distribution function with mean } m. \tag{15}$$

Introduce the drift function

$$\theta^{(r)}(x) = \begin{cases} r\left(\mu^{(r)}(\lceil rx \rceil) - \lambda m\right) & x > 0, \\ 0 & x = 0. \end{cases}$$

The above definition is only for technical convenience, since otherwise $\mu^{(r)}(0)$ would be undefined. However, this does not matter since the server idles when there are no jobs in the system. The heavy traffic condition is specified by

$$\theta^{(r)}(x) \overset{u.o.c}{\longrightarrow} \theta(x) \quad \text{as } n \to \infty, \tag{16}$$

for some locally Lipschitz continuous function $\theta(\cdot)$ on $(0, \infty)$ satisfying

$$\theta(K) > 0. \tag{17}$$

The notation $\overset{u.o.c}{\longrightarrow}$ means uniform convergence on compact sets, which is only required for technical reasons. Condition (17) ensures that the system is stable (see the proof of Theorem 2). As a quick remark, we make a connection with the traditional single server system where the server speed is constant, say $\mu^{(r)}(\cdot) \equiv 1$, and the drift is created by constructing a sequence of $\lambda^{(r)}$ which converges to λ at the rate of $1/r$. The heavy traffic condition for this constant rate LPS system then becomes

$$r\left(1 - \lambda^{(r)} m\right) \to \theta > 0, \quad \text{as } r \to \infty.$$

We are interested in the asymptotic behavior of the diffusion-scaled processes for the rth system, defined as

$$\hat{X}^{(r)}(t) = \frac{1}{r} X^{(r)}(r^2 t), \quad \hat{W}^{(r)}(t) = \frac{1}{r} W^{(r)}(r^2 t). \tag{18}$$

The diffusion scaling for other stochastic processes $\mathcal{Q}^{(r)}$, $\mathcal{Z}^{(r)}$, $Z^{(r)}$, $Q^{(r)}$ and $B^{(r)}$ is defined in the same way. To obtain the diffusion limit of the head count process $\hat{X}^{(r)}$ and workload process $\hat{W}^{(r)}$, we need to carefully analyze the measure-valued processes introduced. The detailed analysis is presented in Appendix C.

Since we need to work with the measure-valued process, let ν denote the probability measure associated with the probability distribution function G, and ν_e denote the probability measure associated with the *equilibrium* distribution G_e of G. That is, $G_e(x) = \frac{1}{m}\int_0^x [1 - G(y)]dy$ and the mean of G_e is

$$m_e = \frac{1 + c_s^2}{2} m.$$

Let $\mathbf{M}$ denote the space of all non-negative finite Borel measures on $[0, \infty)$. We need the following regularity assumptions on the initial state to rigorously prove the diffusion approximation results. Assume there exists $(\xi^*, \mu^*) \in \mathbf{M} \times \mathbf{M}$ such that

$$(\hat{\mathcal{Q}}^{(r)}(0), \hat{\mathcal{Z}}^{(r)}(0)) \Rightarrow (\xi^*, \mu^*), \tag{19}$$

$$\langle \chi^{1+p}, \hat{\mathcal{Q}}^{(r)}(0) + \hat{\mathcal{Z}}^{(r)}(0) \rangle \Rightarrow \langle \chi^{1+p}, \xi^* + \mu^* \rangle \quad \text{for some } p > 0, \tag{20}$$

as $r \to \infty$, and

$$(\xi^*, \mu^*) = \left(\frac{w^* \wedge K m_e}{m_e} \nu, \frac{(w^* - K m_e)^+}{m} \nu_e \right), \tag{21}$$

where $w^* = \langle \chi, \xi^* + \mu^* \rangle$. The above regularity assumptions (19)–(21) basically require that the sequence of initial states is well behaved. These assumptions, together with the heavy traffic assumptions (14)–(16), are made throughout the rest of this paper.

The first result we present is an asymptotic relationship, called State Space Collapse (SSC), between the workload process and the head count process. Define a map $\Delta_K(\cdot) : \mathbb{R}_+ \to \mathbb{R}_+$ by

$$\Delta_K(w) = \frac{w \wedge K m_e}{m_e} + \frac{(w - K m_e)^+}{m}. \tag{22}$$

The SSC result states that the total number of jobs in the system $\hat{X}^{(r)}$ can be *asymptotically* represented using the workload $\hat{W}^{(r)}$ via the map Δ_K, which is a bijective map meaning that workload can also be represented using the total number of jobs. SSC is described as follows:

Proposition 1 (State Space Collapse) *For the sequence of Sd-LPS systems parametrized by $r \in \mathbb{Z}^+$ and satisfying initial conditions* (19)-(21)*, as $r \to \infty$,*

$$\sup_{t \in [0,T]} \left| (\hat{X}^{(r)}(t) \wedge K) m_e + (\hat{X}^{(r)}(t) - K)^+ m - \hat{W}^{(r)}(t) \right| \Rightarrow 0. \tag{23}$$

Note that

$$\Delta_K^{-1}(x) = (x \wedge K) m_e + (x - K)^+ m$$

is the inverse of the map $\Delta_K(\cdot)$. A full version of the SSC, which demonstrates a bijective map between the workload $\hat{W}^{(r)}$ and the measure-valued status $(\hat{\mathcal{Q}}^{(r)}, \hat{\mathcal{Z}}^{(r)})$, is presented and proved in Appendix C. Roughly speaking, SSC reveals that the residual sizes of jobs in service follow the equilibrium distribution G_e. The simpler SSC of Proposition 1 can be derived from the full version proved in Appendix C. For the purpose of performance analysis and for optimal control in this paper, we only need the simple version of SSC.

The next step is the analysis of the workload process defined in (7). The challenge here is that the evolution of workload depends on the number of jobs in service due to the state-dependent service rate. The simple SSC result allows us to overcome this difficulty. The following theorem establishes the diffusion limit of the workload process $\hat{W}^{(r)}(t)$ and the number of jobs $\hat{X}^{(r)}$ as reflected Brownian motion (RBM) with state-dependent drifts.

Theorem 1 (Weak convergence to RBMs with state-dependent drift) *For the sequence of Sd-LPS systems parametrized by $r \in \mathbb{Z}^+$ satisfying* (19)-(21)*, as $r \to \infty$,*

$$\hat{W}^{(r)} \Rightarrow W^*, \tag{24}$$

where W^* is an RBM with initial value $W^*(0) = w^*$, drift $-\theta \left(\Delta_K(W^*(t)) \wedge K \right)$ and variance $\sigma^2 = \lambda m^2 (c_a^2 + c_s^2)$. Moreover, as $r \to \infty$,

$$\hat{X}^{(r)} \Rightarrow X^* = \Delta_K(W^*). \tag{25}$$

The proof of Theorem 1 is presented in Appendix C. Theorem 1 states that the workload process of the sequence of Sd-LPS systems converges to a reflected Brownian motion with state dependent drift and state-independent dispersion, from which the process for the number of jobs in system can be obtained using the state-space collapse map Δ_K. In the following Section 3.3, we will identify the steady-state distribution of the limiting RBM and a closed-form formula for the steady state mean number of jobs which will then lead us to the still more tractable approximation (13).

3.3 Steady State of the Diffusion Limit

The entire goal of heavy traffic analysis is to obtain a tractable process, an RBM with state-dependent drift, as an approximation of the complicated stochastic process underlying the original model. That is, the steady state of the limiting RBM can be computed. The following Proposition gives the requisite steady-state distribution of workload and number of jobs in the system (the proof appears in Appendix B):

Proposition 2 *Let W^* and X^* be the workload and number of jobs for the limiting Sd-LPS system (as defined in Theorem 1). Let the drift function $\theta(x)$ be given by $-\theta(x) = \lambda m \frac{d \log f(x)}{dx}$.*

The steady-state distributions of W^* and X^* are given by

$$\mathbf{Pr}[W^*(\infty) \le w] = \alpha \int_0^{\frac{w}{m_e}} f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx, \tag{26}$$

$$\mathbf{Pr}[X^*(\infty) \le x] = \begin{cases} \alpha \int_0^x f(u)^{\frac{c_s^2+1}{c_s^2+c_a^2}} du & x \le K, \\ \alpha \int_0^{K+(x-K)\frac{m}{m_e}} f(u)^{\frac{c_s^2+1}{c_s^2+c_a^2}} du & x > K, \end{cases} \tag{27}$$

where α is the normalization constant. The mean of the limiting scaled number of jobs is given by

$$\mathbf{E}[X^*(\infty)] = \frac{\int_{x=0}^{\infty} (x \wedge K) f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx}{\int_{x=0}^{\infty} f\,(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}}\, dx} + \frac{c_s^2+1}{2} \cdot \frac{\int_{x=0}^{\infty} (x-K)^+ f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx}{\int_{x=0}^{\infty} f\,(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}}\, dx}. \tag{28}$$

The approximation (13) at the beginning of this section is obtained from (28) by further using the probability mass function, $\pi(\cdot)$, for the number of jobs corresponding to the original Sd-LPS system in place of the density function $f(\cdot)$.

Finally, we close the loop by translating the convergence at the process level to convergence of steady-state distributions in the following theorem (proof in Appendix B). This justifies the formulae in Proposition 2 as an approximation for the steady state of the original Sd-LPS system. The quality of the approximation is demonstrated in the numerical experiment presented at the beginning of this section (see Figure 2).

Theorem 2 (Convergence of steady-state distributions) *For all large enough r, the stochastic process $\hat{X}^{(r)}$ has a steady state, denoted by $\hat{X}^{(r)}(\infty)$. Moreover,*

$$\hat{W}^{(r)}(\infty) \Rightarrow W^*(\infty),$$
$$\hat{X}^{(r)}(\infty) \Rightarrow X^*(\infty),$$

where $W^*(\infty)$ and $X^*(\infty)$ are characterized in (26) and (27).

4 Dynamic concurrency control for the Sd-LPS queue

In Section 3, we established approximations for the steady-state number of jobs and workload in an Sd-LPS system operating under a *static* concurrency level. Our numerical experiments showed that the optimal static level based on the approximations yields near-optimal performance for the original Sd-LPS system. In this section we go further by developing policies which *dynamically adjust* the concurrency level based on developing and solving an appropriate diffusion control problem.

In Section 4.1 we define a diffusion control problem and set up the HJB optimality conditions. In Section 4.2 we describe how the parameters for the diffusion control problem (the drift as a function of the control) are instantiated starting from dynamic control problem for a discrete state-dependent PS server, and how the optimal policy for the diffusion control problem is translated back into a heuristic control policy for the original discrete state space Sd-LPS server. To demonstrate the efficacy of our approach, we present numerical experiments comparing the performance of the proposed diffusion limit based control policy against the true optimal dynamic control policy for a special non-trivial input process for which the true optimal policy can be computed numerically. Finally, in Section 4.3 we describe a numerical algorithm tailored to solve the diffusion control problem posed in Section 4.1. Our algorithm iteratively refines its estimate of the average cost of the optimal policy using Newton-Raphson root finding method.

4.1 A diffusion control problem and HJB equation

Consider the problem of controlling a Reflected Brownian Motion $W(t) \in \mathbb{R}_+$ with state and control independent dispersion $\sigma^2 := \lambda m^2(c_s^2 + c_a^2)$, and control-dependent drift $-\theta(U(t))$ where $U(t) \in \mathbb{R}_+$ is the control exerted at time t. We will restrict to stationary control policies $U(t) = k(W(t))$, where the state-dependent control function $k : \mathbb{R}_+ \to \mathbb{R}_+$ is restricted to lie in the following set:

$$\mathcal{K} = \left\{ k : \mathbb{R}_+ \to \mathbb{R}_+ | k(w) \le w/m_e;\ k \text{ is Lipschitz continuous; } \int_{v=0}^{\infty} e^{-\int_0^v \theta(k(w))dw} dv < \infty \right\}. \tag{29}$$

For intuition, the state $W(t)$ maps to the workload in the limiting Sd-LPS system, and $k(W)$ maps to the concurrency level as a function of the workload. Thus, the restriction $k(w) \le w/m_e$ (or equivalently $w \ge k(w) \cdot m_e$ is reminiscent of state space collapse which states that the mean residual size of jobs at the server is m_e.

The instantaneous cost rate for policy $k \in \mathcal{K}$ and state w is given by the mapping $\Delta_k : \mathbb{R}_+ \to \mathbb{R}_+$

$$\Delta_k(w) = \frac{w \wedge k(w)m_e}{m_e} + \frac{(w - k(w)m_e)^+}{m} \tag{30}$$

which, again, reminiscent of the state space collapse result (Proposition 1) gives the number of jobs in the limiting system under control $k(w)$ and workload w.

The last condition in (29) ensures that a stationary distribution for the diffusion-scaled workload under $k(w)$ exists. Indeed, we assume that the drift function $\theta(\cdot)$ satisfies

$$\sup_{x \in [0,M]} \theta(x) > 0, \tag{31}$$

for some $M < \infty$. That is, intuitively, a service rate for the PS server strictly larger than the arrival rate is achievable at a finite concurrency limit and hence at a finite workload. In fact, we will make a stronger assumption. Define

$$\hat{\theta} \doteq \sup_{x \in \mathbb{R}_+} \theta(x) \ ; \quad \hat{k} \doteq \arg\max_k \{\theta(k)\}.$$

Here $\hat{k}$ denotes the most efficient control (concurrency level) for drift function $\theta(\cdot)$, which we will assume to be finite and unique.

Let $V_\gamma(w)$ denote the discounted total cost (with discount rate γ) for the diffusion W under a control policy $k(\cdot)$ when the workload starts in state w:

$$V_\gamma(w) = \mathbb{E}_w \Big[\int_0^\infty e^{-\gamma t} \Delta_k(W(t))dt \Big]. \tag{32}$$

Consider a small $\delta > 0$. According to Itō calculus

$$\begin{aligned}
V_\gamma(w) &= \Delta_k(w) + (1-\gamma\delta)\mathbb{E}\left[V_\gamma\left(W(\delta)\right)\right] + o(\delta) \\
&= \Delta_k(w) + (1-\gamma\delta)\mathbb{E}\left[V_\gamma(w) + V_\gamma'(w)(W(\delta)-w) + \frac{V_\gamma''(w)}{2}(W(\delta)-w)^2 + o(\delta)\right] + o(\delta) \\
&= \Delta_k(w) + (1-\gamma\delta)\left[V_\gamma(w) + V_\gamma'(w)\theta(k(w))\delta + \frac{V_\gamma''(w)}{2}\sigma^2\delta\right] + o(\delta).
\end{aligned}$$

We thus have the following relation for the discounted value function V_γ:

$$\gamma V_\gamma(w) = \Delta_k(w) - \theta(k(w))V_\gamma'(w) + \frac{\sigma^2}{2}V_\gamma''(w). \tag{33}$$

Letting $\gamma \to 0$, define

$$v = \lim_{\gamma\to 0}\gamma V_\gamma(w)\ , \quad \text{and} \quad G(w) = \lim_{\gamma\to 0}V_\gamma'(w),$$

where v is the average cost of policy $k(\cdot)$, and the value function gradient $G(w)$ solves the following ordinary differential equation (ODE):

$$v = \Delta_k(w) - \theta(k(w))G(w) + \frac{\sigma^2}{2}G'(w). \tag{34}$$

Above, we have provided a heuristic derivation to arrive at the average cost optimal control problem as a limit of the discounted cost problem. For a formal treatment of the relation between discounted and average cost problems (i.e., by defining discounted relative cost functions $h_\gamma(w) = V_\gamma(w) - V_\gamma(\tilde{w})$ for some positive recurrent state $\tilde{w}$, taking limit $h(w) = \lim_{\gamma\downarrow 0} h_\gamma(w)$ and $v = \lim_{\gamma\downarrow 0}\gamma V_\gamma(\tilde{w})$), we refer readers to [13].

The following Proposition and Remark state two useful facts about the value function gradient $G(w)$ which will be useful in the development of our numerical algorithm for solving the optimal control.

Proposition 3 *The discounted value function $V_\gamma(w)$ is non-decreasing in w for all γ, and hence $G(w) \geq 0$.*

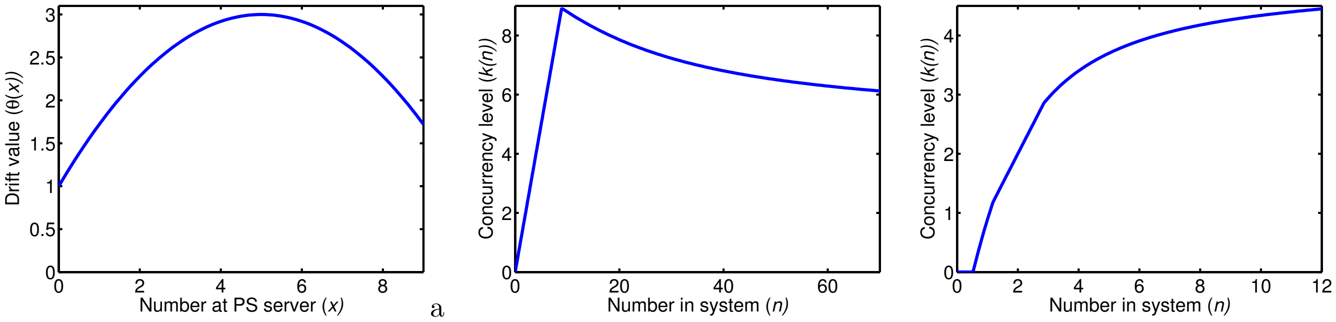
Remark 1 *For a given control policy $k(w)$, equation (34) is a first order ODE for $G(w)$. However, to solve $G(\cdot)$ we also need to know the average cost v. This is to be expected since we started from a second order ODE where we would need two boundary conditions to completely specify V_γ. In our case, one boundary condition is easy to get hold of: since we have a reflecting boundary at $w = 0$, we must have (see, for example, [33, page VIII]):*

$$V_\gamma'(0) = 0 \tag{35}$$

and therefore, also $G(0) = 0$.

Returning to equation (33), let V_γ^* denote the value function for the optimal policy. Then Bellman's principle of optimality becomes:

$$\gamma V_\gamma^*(w) = \min_{k\in[0,w/m_e]}\left\{\Delta_k(w) - \theta(k)V_\gamma^{*\prime}(w) + \frac{\sigma^2}{2}V_\gamma^{*\prime\prime}(w)\right\}. \tag{36}$$



(a) Drift function $\theta(\cdot)$

(b) $c_a^2 = c_s^2 = 10, \sigma^2 = 20$

(c) $c_a^2 = c_s^2 = 0.3, \sigma^2 = 0.6$

Figure 3: A hypothetical drift function $\theta(x)$ and the optimal diffusion control policies for two choices of workload parameters c_a^2, c_s^2.

If we let $\gamma \to 0$, then

$$v^* = \min_{k \in [0, w/m_e]} \left\{ \Delta_k(w) - \theta(k) G^*(w) + \frac{\sigma^2}{2} G^{*\prime}(w) \right\}, \quad \forall w \in \mathbb{R}_+, \tag{37}$$

where again, as remarked earlier, we have the boundary condition $G^*(0) = 0$, leaving v^* the only unknown.

For an illustration of what an optimal dynamic policy might look like, see Figure 3. The first figure shows an illustrative example of the $\theta(x)$ function for the PS server. As can be seen, the PS server is most efficient when there are $\hat{k} = 5$ jobs at the server, and the speed drops on either side of this point. The second figure shows the optimal dynamic policy (translated from $k(w)$ to $k(n)$, that is, as a function of the number of jobs in the system, for clarity) when $c_s^2 = c_a^2 = 10$. This corresponds to a workload that has significant variability, and the optimal policy increases the concurrency level to approximately 9 when the number of jobs in the system is small but scales it back when there is a long queue. The third figure shows the policy for $c_s^2 = c_a^2 = 0.3$. This is a low variability workload, and as the number of jobs in the system increases, initially the PS server acts as an FCFS server and thus compromises speed to keep the concurrency level small. At $n \approx 0.5$, the system switches to a controlled PS behavior by gradually increasing the concurrency level to increase service rate. At $n \approx 1.2$ the system switches to a pure PS behavior admitting everyone in queue, and finally at $n = 3$ it switches back to a controlled PS behavior, gradually increasing the concurrency level to $\hat{k} = 5$ as queue becomes longer.

Though many diffusion control problems addressed in the literature have a nice structure allowing a closed-form solution, e.g., [21, 22], the problem (37) is intrinsically difficult mainly due to the generality of the service rate curve. Thus we seek numerical algorithms, which presents another challenge. For diffusion control problems where a closed-form solution can be found, one of the boundary conditions is imposed by setting the coefficient of the exponential term in the solution of the second order ODE to zero. This captures the physical constraint that the optimal value function should asymptotically grow at a polynomial rate and not exponentially. However, this trick cannot be applied when searching for a numerical solution. This obstacle led us to develop the algorithm in Section 4.3. While the majority of numerical algorithms for solving diffusion control problems rely on the Markov chain method where time and space are discretized and a probability transition matrix is engineered to satisfy local consistency requirements (e.g., [28]), we directly work with the ODE in (37).

4.2 From discrete Sd-LPS server to diffusion control and back

Recall that our goal is to develop a dynamic concurrency control policy for a discrete Sd-LPS server with mean arrival rate λ, service rate function $\mu = \{\mu(1), \mu(2), \ldots\}$, i.i.d. service requirements with mean m and squared coefficient of variation c_s^2, i.i.d. interarrival times with squared coefficient of variation c_a^2. The following steps outline how we instantiate the diffusion control problem and how we translate the resulting policy $k(\cdot)$ into a heuristic dynamic control policy for the original Sd-LPS server.

1. Create a drift function $\theta(\cdot)$ from the service rate curve $\mu(\cdot)$ of the original state-dependent Processor Sharing server:
$$\theta(x) \doteq -\lambda m \log \frac{\lambda m}{\hat{\mu}(x)},$$
where $\hat{\mu}$ is any continuous and differentiable extension of μ.

2. Solve the diffusion control problem (37) to obtain the policy $k^*(w)$ which gives the concurrency level as a function of the workload.

3. Find $n(w) = k^*(w) + \frac{w - m_e \cdot k^*(w)}{m}$ as the number of jobs in the system as a function of the workload, and compute $\tilde{k}(n) = k(w^{-1}(n))$. For certain drift functions and dispersion σ, $n(w)$ is not strictly increasing and hence $w^{-1}(n)$ is not unique. In such cases we truncate $n(w)$ to the domain in which in which it is strictly increasing, and set $\tilde{k}(n) = \hat{k}$ everywhere else.

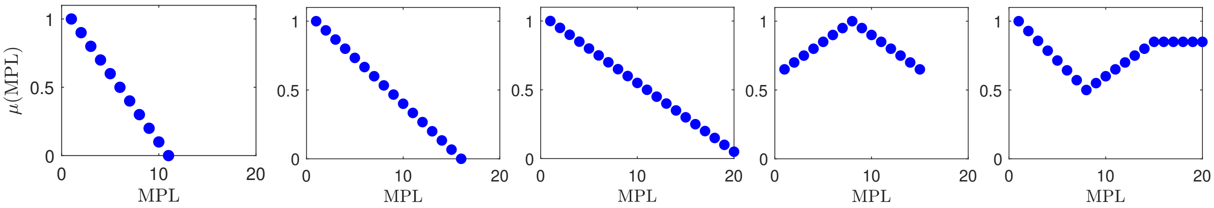
4. Let $\check{k}(n)$ be the given by rounding $\tilde{k}(n)$ to the nearest integer. The control algorithm is implemented by taking action to reach the concurrency level $\check{k}(N(t))$. In controling the original system we only take actions upon job arrivals and departures, do not preempt jobs once they enter service, and do not increase the concurrency level by more than one in any arrival/departure event. The precise policy is given as follows:

   - Define $Z(t_-)$ and $Q(t_-)$ to be the number of jobs at the server and in the buffer, respectively, before the arrival/departure event at time t.
   - **On arrival at t:** If $\check{k}(Z(t_-)+Q(t_-)+1) \geq (Z(t_-)+1)$ then admit one job to the server at t, otherwise do nothing.
   - **On departure at t:** Admit $\min\left\{\left(\check{k}(Z(t_-)+Q(t_-)-1) - Z(t_-)+1\right)^+, 2\right\}$ jobs at t.

Simulation Results

Figure 4 shows experimental results comparing the performance of the dynamic policies produced using the proposed diffusion scaling against the performance of the naive fluid heuristic that always chooses the most efficient MPL available. To gain some insights into when the dynamic heuristic is near optimal, and substantially better than the naive fluid heuristic, we simulate five different service rate curves. To be able to compare the performance of the heuristics against the optimal performance, we focus on a special class of input processes: Poisson arrivals and a degenerate Hyperexponential job size distribution (a mix of a point mass at 0 and an Exponential distribution). This allows us to compute optimal dynamic policies using the algorithm proposed by [18].

The dynamic policy for the diffusion control problem was computed using a Binary search variant of the Newton-Raphson method (Algorithm 1, Section 4.3) to an additive error of 10^{-3}. We used MATLAB's `ode45` function to solve the differential equations involved. The performance of the



Performance of diffusion based heuristic

$\lambda \setminus c_s^2$	1.5	9	19	1.5	9	19	1.5	9	19	1.5	9	19	1.5	9	19
0.5	1.031	1.040	1.032	1.023	1.023	1.025	1.023	1.017	1.014	1.000	1.012	1.038	1*	1.056	1.588
0.6	1*	1.032	1.020	1.020	1.034	1.030	1.021	1.018	1.016	1.000	1.013	1.006	1*	1.065	1.056
0.7	1*	1.012	1.034	1*	1.015	1.027	1.010	1.026	1.020	1.000	1.007	1.011	1.029	1.057	1.052
0.8	1*	1.001	1*	1*	1.004	1*	1*	1.005	1.008	1.000	1.001	1*	1*	1.039	1.035
0.9	1*	1.000	1*	1*	1.000	1*	1*	1.005	1.004	1*	1.001	1*	1*	1.010	1.009
0.95	1*	1.000	1.005	1*	1.000	1.005	1*	1.000	1.005	1*	1.001	1.004	1*	1.000	1.005

Performance of fluid heuristic

$\lambda \setminus c_s^2$	1.5	9	19	1.5	9	19	1.5	9	19	1.5	9	19	1.5	9	19
0.5	1.000	1.176	1.322	1.004	1.456	1.856	1.023	1.769	2.544	1.000	1.011	1.038	1.000	1.397	2.597
0.6	1.000	1.043	1.079	1.000	1.160	1.276	1.003	1.322	1.551	1.000	1.013	1.053	1.000	1.134	1.228
0.7	1.000	1.000	1.005	1.000	1.030	1.054	1.000	1.081	1.133	1.000	1.001	1.012	1.000	1.023	1.041
0.8	1	1	1	1.000	1.000	1.000	1.000	1.006	1.012	1	1	1	1	1	1
0.9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0.95	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 4: Simulation results comparing the performance of the diffusion heuristic based dynamic concurrency control policy for Poisson arrivals with rate $\lambda \in \{0.5, 0.6, 0.7, 0.8, 0.9, 0.95\}$ and degenerate hyperexponential (H^*) distribution with $m = 1$ and SCV c_s^2. Each entry in the top table is the ratio of the simulated mean sojourn time of the diffusion based heuristic policy to the mean sojourn time of a near-optimal policy computed via policy iteration (We find the optimal policy within fluid continuation policies where the MPL is chosen to be the most efficient MPL $\hat{k}$ beyond queue length $Q = 150$ and we stop policy iteration when the new average cost is at least 0.9999 times the old average cost. The 1^* indicates that the diffusion heuristic outperformed this 'optimal' policy). The bottom table shows the ratio of the policy that always chooses the most efficient feasible MPL (hence the 'fluid' heuristic) to the optimal. The 95% confidence intervals for all entries are smaller than ± 0.008.

diffusion control policy was evaluated via simulation, the performance of the fluid and the optimal policy are evaluated numerically via Matrix analytic methods.

The main takeaways are:

- From the bottom table, which compares the benefit of the optimal dynamic policy versus the fluid heuristic, we see that as in the case of the static concurrency control, the gain of the optimal dynamic policy over the fluid dynamic heuristic is larger if the arrival rate is smaller, and if c_s^2 is larger.
- The benefit of optimal dynamic control over fluid policy is more prominent when the service rate curve is decreasing, and is larger if the MPL curve falls less steeply. In the first three cases in Figure 4, the mean sojourn time under the fluid heuristic is as much as 32%, 85%, and 154% larger than the optimal.
- The top table comparing the performance of the diffusion based heuristic versus the optima dynamic control shows that the heuristic is near optimal when the service rate curve is monotonically decreasing. The mean sojourn time are at most 4% larger than the optimal.
- If the service rate curve initially increases and then decreases as in the fourth example, the diffusion based heuristic again gives near optimal policy (at most 4% loss), but the fluid heuristic is also good enough (at most 5.3% loss) for the cases shown. As mentioned earlier, for larger c_s^2, or for curves where the initial increase in service rate is less steep, and where the most efficient MPL is smaller, the gain of diffusion heuristic over fluid heuristic will be larger.

- If the service rate curve initially decreases and then increases as in the fifth case, then the diffusion policy is not always near optimal. However, we believe the loss is due to the manner in which we translate the diffusion control to a control for the original system.

Further Remarks on diffusion control formulation

The reader might wonder, why the diffusion control problem was formulated with the workload as the state variable instead of the headcount process when the workload may not be observable in the true discrete Sd-LPS system but the head count is. There are two reasons for choosing workload over the head count process: (i) the variance of head count process is state-dependent making the computation more complicated, while it is a constant for workload process, and (ii) headcount does not carry enough information since two different states (Q_1, Z_1) and (Q_2, Z_2) along the state-space collapse trajectory may have the same head count but different workloads. Therefore the control is not uniquely obtained as a function of the number of jobs in the system.

Finally, to motivate the diffusion control problem, we state the following conjecture on the process level convergence of dynamic control for Sd-LPS systems:

Conjecture 1 (Diffusion limits under a dynamic policy) *Consider a sequence of Sd-LPS servers parameterized by $r \in \mathbb{Z}^+$ (each with mean arrival rate λ) with service rate curves $\mu^{(r)}$ satisfying the limit:*

$$\lim_{r\to\infty} r\left(\lambda m - \mu^{(r)}\left(\lceil rx \rceil\right)\right) = -\theta(x).$$

Further, let the dynamic policies $k^{(r)}$ satisfy:

$$k^{(r)}(W^{(r)}(t)) = \left\lceil rk\left(W^{(r)}(t)/r\right)\right\rceil \tag{38}$$

for some $k \in \mathcal{K}$, as $r \to \infty$. Let the initial workload satisfy $\hat{W}^{(r)}(0) = 0$. Then,

$$\hat{W}^{(r)} \Rightarrow W^*, \tag{39}$$

where W^* is an RBM with initial value $W^*(0) = 0$, drift $-\theta\left(k(W^*(t))\right)$ and variance $\sigma^2 = \lambda m^2(c_a^2 + c_s^2)$. Moreover, as $r \to \infty$,

$$\hat{X}^{(r)} \Rightarrow X^* = \Delta_k(W^*). \tag{40}$$

In other words, we conjecture that the state space collapse result still holds and the state-dependent concurrency level function $k(\cdot)$ only plays a role in modifying the drift of the diffusion limit of the workload. The key to proving this conjecture is to extend the state space collapse result to allow a dynamic concurrency level and analyze the underlying fluid model (as in [44]). Due to the technical intricacies involved, proving the conjecture is beyond the scope of this paper. Instead, we focus on utilizing the conjectured diffusion limit to identify a near-optimal policy for the original LPS system.

4.3 Newton-Raphson method for solving diffusion control problem

Before giving the algorithm, we discuss the main intuition and ideas behind it. Let us assume that an oracle reveals to us the average cost v^* of the optimal policy. This, together with the boundary condition $G^*(0) = 0$, would allow us to numerically solve for the optimal control by evolving $G^*(\cdot)$ forward: Assuming we have solved $G^*(w)$ for $w \in [0, x]$, we first find

$$k^*(x) = \underset{k\in[0,x/m_e]}{\arg\min}\left\{k\left(1 - \frac{m_e}{m}\right) - \theta(k)G^*(x)\right\} \tag{41}$$

and then

$$\frac{\sigma^2}{2} G^{*\prime}(x) = v^* - \left[\frac{x}{m} + k^*(x)\left(1 - \frac{m_e}{m}\right) - \theta(k^*(x))G^*(x)\right]$$

allows us to evolve $G^*(w)$ forward in a small enough interval $(x, x + \delta x]$. The trouble is that when we do not have the correct guess v^*, we need to be able to detect whether our guess is above or below v^*. This is indeed possible (see [19]).

In this section, we will instead search for the optimal policy within a class of suboptimal policies called *fluid continuation policies.*

Definition 1 *The set of fluid continuation policies with* fluid continuation point W *is defined as*

$$\mathcal{F}_W = \left\{ k \in \mathcal{K} : k(w) = k_f(w) \doteq \arg\max_{x \leq w/m_e}\{\theta(x)\}, w \geq W \right\}. \tag{42}$$

That is, beyond the fluid continuation workload point W, the control is chosen to be the most efficient service rate available. Denote the cost of the optimal (minimum cost) policy in $\mathcal{F}_W$ by $v_f(W)$. Let $\overline{W}(v) = \min\{W \geq 0 : v = v_f(W)\}$.

All policies in $\mathcal{F}_W$ are stable due to condition (31). In fact, the policy $k_f \in \mathcal{F}_0$ is optimal when $c_s^2 = 1$ since in this case $m_e = m$, and (41) simplifies to

$$k^*(w) = \arg\min_{k \in [0, w/m_e]} \theta(k)G^*(w) = \arg\min_{k \in [0, w/m_e]} \theta(k).$$

The next proposition shows that we do not need a very large fluid continuation point for finding an approximately optimal policy.

Proposition 4 *Let $v^* \leq v \leq v_f(0)$. Then v is the average cost of an optimal fluid continuation policy k_v with continuation point $\overline{W}(v)$. That is:*

$$k_v(w) = \begin{cases} \arg\min_{k \in [0, w/m_e]} \left\{k\left(1 - \frac{m_e}{m}\right) - \theta(k)G_v(w)\right\} & w \leq \overline{W}(v), \\ k_f(w) & w > \overline{W}(v), \end{cases} \tag{43}$$

where G_v is the value function gradient for policy k_v and satisfies the ODE

$$v = \frac{w}{m} + k_v(w)\left(1 - \frac{m_e}{m}\right) - \theta(k_v(w))G_v(w) + \frac{\sigma^2}{2} G_v'(w). \tag{44}$$

Further, $\overline{W}(v) = O\left(\log \frac{1}{v - v^*}\right)$.

The advantage of the class of feasible policies described by (43) is that for any v, the function $G_v(w)$ for $w \geq \max\{\hat{k}m_e, \overline{W}(v)\}$ is easily computed. Let us call this the *fluid continuation of the value function gradient* and denote it by $\overline{G}_v(w)$. Then, it can be shown that:

$$\overline{G}_v(w) = \frac{w}{m\hat{\theta}} + \left(\hat{k}\left(1 - \frac{m_e}{m}\right) + \frac{\sigma^2}{2m\hat{\theta}} - v\right)\frac{1}{\hat{\theta}}, \quad w \geq \max\{\overline{W}(v), \hat{k}m_e\}. \tag{45}$$

To find an approximately optimal solution for the diffusion control problem (37), we first fix a large enough value of W ($W \geq \hat{k}m_e$) and seek the optimal policy and the optimal average cost in $\mathcal{F}_W$. As mentioned later, we can use a standard doubling trick to settle on a 'large enough' W. Next, we will guess an average cost value v and devise a test to compare v with $v_f(W)$. For this, we evolve ODE (37) backwards with the (terminal) boundary condition

$$G_v(W) = \frac{W}{m\hat{\theta}} + \left(\hat{k}\left(1 - \frac{m_e}{m}\right) + \frac{\sigma^2}{2m\hat{\theta}} - v\right)\frac{1}{\hat{\theta}}. \tag{46}$$

If indeed $v = v_f(W)$ then we must have $G_v(0) = 0$, and the sign of $G_v(0)$ can tell us if $v < v_f(W)$ or $v > v_f(W)$. However, since we know $G_{v_f(W)}(0) = 0$, we can go further and cast the problem of finding $v_f(W)$ as solving for the root of the equation $G_v(0) = 0$ (in v) using the Newton-Raphson method.

Let us assume that our current guess for $v_f(W)$ is v_n. To generate the next guess v_{n+1} via the Newton-Raphson method, we need the derivative of $G_v(0)$ at $v = v_n$. With some abuse of notation, define

$$g_v(w) \doteq \frac{dG_v(w)}{dv}.$$

(What we really mean by the above is that $g_v(w) \doteq \frac{\partial G(v,w)}{\partial v}$, where $G(v,w) = G_v(w)$.) As we will show in the proof of Proposition 5 (see step 2 of the proof), $G_v(w)$ is Lipschitz continuous and decreasing in v for all w and therefore $g_v(w)$ exists almost everywhere, and further it is bounded away from 0.

With $W \geq \hat{k}m_e$ representing the point at which we switch to the fluid policy k_f, we can write $G_v(w)$ as the following integral: for $w \leq W$,

$$G_{v_n}(w) = G_{v_n}(W) + \frac{2}{\sigma^2}\int_W^w \left[v_n - \min_{k\in[0,u/m_e]}\{\Delta_k(u) - \theta(k)G_{v_n}(u)\}\right] du, \tag{47}$$

where $G_{v_n}(W)$ is given by (46). Differentiating (47) with respect to v_n yields

$$g_{v_n}(w) = -\frac{1}{\hat{\theta}} + \frac{2}{\sigma^2}\int_W^w [1 + \theta(k_{v_n}(u))g_{v_n}(u)]\, du.$$

Since the policy $k_{v_n}()$ also depends on v_n, to arrive at the last equality, we have used the *envelope theorem*: If $k^*(v) = \arg\min_k \phi(k,v)$ and $\phi^*(v) = \phi(k^*(v),v)$, then $\frac{d\phi^*(v)}{dv} = \frac{\partial\phi(k^*(v),v)}{\partial v}$ (where $\frac{\partial\phi(k,v)}{\partial v}$ is the partial derivative with respect to v). Therefore, very similar to G_{v_n}, g_{v_n} satisfies the following ODE

$$1 = -\theta(k_{v_n}(w))g_{v_n}(w) + \frac{\sigma^2}{2}g'_{v_n}(w) \tag{48}$$

with the terminal condition $g_{v_n}(W) = -\frac{1}{\hat{\theta}}$. The updated guess for average cost is then

$$v_{n+1} = v_n - \frac{G_{v_n}(0)}{g_{v_n}(0)}. \tag{49}$$

It turns out that v_{n+1} is exactly the average cost of the policy, call it $k_{v_n}(w)$, that is implicitly generated when solving for G_{v_n} and g_{v_n}. This is because for a fixed policy $k(w) = k_{v_n}(w)$, G_v is linear in v from (47). Therefore the sequence of average cost iterates produced by the algorithm are in fact the average costs of a sequence of feasible policies. The next proposition formally states the result on convergence of the Newton-Raphson average cost iteration algorithm.

Proposition 5 *Let $v_1, v_2, \dots$ denote the average cost iterates generated by the Newton-Raphson method (49). Let*

$$d_\theta \doteq \sup_k \theta(k) - \inf_k \theta(k) < \infty.$$

The sequence $\{v_n\}$ monotonically decreases to $v_f(W)$, which is the average cost of the optimal diffusion control policy in the set $\mathcal{F}_W$ of fluid continuation policies with fluid continuation point W.

Since close to the root, the error roughly squares in each iteration for Newton-Raphson method, it takes $O(\log\log\frac{1}{\epsilon})$ iterations to reach an ϵ-optimal policy within $\mathcal{F}_W$. To find the ϵ-optimal policy among all policies, we can keep doubling the value of W until the error between successive iterates is sufficiently small. By our earlier result, we need a $W = O(\log\frac{1}{\epsilon})$ to arrive at an ϵ-optimal policy. Since each iteration of the Newton-Raphson method takes $O(W)$ time, the overall time complexity of the algorithm to find an ϵ-optimal policy is $O\left(\log\frac{1}{\epsilon}\log\log\frac{1}{\epsilon}\right)$. The step-by-step procedure is described in Algorithm 1. In the description, we have omitted iterating over values of W, the fluid continuation point, to focus on the core of the algorithm. The average cost of the fluid policy is chosen as the initial guess for average cost v_0 which is computed using a single step of Newton-Raphson iteration (shown in the initialize block).

Algorithm 1 Average cost iteration (Newton-Raphson method)

```
define $\hat{k} \doteq \arg\max_k \theta(k)$; $\hat{\theta} \doteq \theta(\hat{k})$
require $W \geq \hat{k}m_e$                                          ▷ (Fluid continuation point)
initialize              ▷ (Compute initial guess for average cost $v_f(0)$, see Defn. 1)
    solve functions $G_f(w)$ and $g_f(w)$ for $w \in [0, \hat{k}m_e]$:
        $G_f(\hat{k}m_e) = \left(\hat{k}(1 - m_e/m) + \frac{\sigma^2}{2m\hat{\theta}}\right)\frac{1}{\hat{\theta}} + \frac{1}{m\hat{\theta}} \cdot \hat{k}m_e$      ▷ (Terminal condition for $G_f$)
        $g_f(\hat{k}m_e) = -\frac{1}{\hat{\theta}}$                              ▷ (Terminal condition for $g_f$)
        $k_f(w) = \arg\max_{k \in [0, w/m_e]} \theta(k)$                        ▷ (Fluid optimal policy)
        $0 = \frac{w}{m} + k_f(w)(1 - \frac{m_e}{m}) - \theta(k_f(w))G_f(w) + \frac{\sigma^2}{2}G'_f(w)$      ▷ (ODE for $G_f$)
        $1 = -\theta(k_f(w))g_f(w) + \frac{\sigma^2}{2}g'_f(w)$                   ▷ (ODE for $g_f$)
    end solve
    $v_0 \leftarrow v_f(0) = -\frac{G_f(0)}{g_f(0)}$
end initialize
repeat
    solve policy $k_{v_n}(w)$, functions $G_{v_n}(w)$ and $g_{v_n}(w)$ for $w \in [0, W]$:
        $G_{v_n}(W) = \left(\hat{k}(1 - m_e/m) - v_n + \frac{\sigma^2}{2m\hat{\theta}}\right)\frac{1}{\hat{\theta}} + \frac{1}{m\hat{\theta}} \cdot W$      ▷ (Terminal condition for $G_{v_n}$)
        $g_{v_n}(W) = -\frac{1}{\hat{\theta}}$                                ▷ (Terminal condition for $g_{v_n}$)
        $k_{v_n}(w) = \arg\min_{k \in [0, w/m_e]} \left\{k\left(1 - \frac{m_e}{m}\right) - \theta(k)G_{v_n}(w)\right\}$
        $v_n = \frac{w}{m} + k_{v_n}(w)(1 - \frac{m_e}{m}) - \theta(k_{v_n}(w))G_{v_n}(w) + \frac{\sigma^2}{2}G'_{v_n}(w)$      ▷ (ODE for $G_{v_n}$)
        $1 = -\theta(k_{v_n}(w))g_{v_n}(w) + \frac{\sigma^2}{2}g'_{v_n}(w)$             ▷ (ODE for $g_{v_n}$)
    end solve
    update $v_{n+1} \leftarrow v_n - \frac{G_{v_n}(0)}{g_{v_n}(0)}$                    ▷ (Newton-Raphson update)
until $|G_{v_n}(0)| \leq \epsilon$
return Cost $v_{n+1}$; Policy $k_{v_n}(w)$
```

Comparison with the policy iteration algorithm: Puterman and Brumelle [36] prove that the policy iteration algorithm for discrete-time Markov decision processes is equivalent to the Newton-

Raphson algorithm for finding the fixed point of the dynamic programming operator performed in the value function space. Puterman [35] presents a policy iteration algorithm for control of a diffusion process in a bounded region in $\Re^n$ for finite horizon total cost optimization. In comparison with [35], one difference between our approach is that we carry out the Newton-Raphson algorithm in the space of average cost. A second difference is that the policy iteration algorithm alternates between policy evaluation and policy improvement steps while our algorithm can be viewed as one where we have folded the policy evaluation and policy improvement into one step.

5 Concluding Remarks

The primary goal of the present paper was to propose a diffusion scaling to aid the analysis and control of State-dependent Limited Processor Sharing (LPS) systems. Our philosophy while designing the scaling was to fix a limiting distribution for the steady-state number of jobs in the system, and then reverse-engineer the sequence of service rate curves that yields this limit. By choosing the limiting distribution as the one of the original state-dependent system under an $M/M/$ input, our scaling thus ensures that the effect of the entire service rate curve emerges in the diffusion approximation, which then leads to the choice of a near-optimal static and dynamic concurrency limit policies.

One task that we did not address in the paper is proving Conjecture 1 on convergence of the workload process to a limiting diffusion under dynamic control policies. However, a perhaps more important gap is that we have only shown numerical evidence of near-optimality of proposed control policies. The experiments in Figure 2 show that our approximations for the performance (steady-state mean number of jobs) are not always accurate even though they capture the shape to yield a good heuristic. Formalizing these observations would contribute to our understanding of when and what diffusion approximations are more suitable for devising control policies.

References

[1] K. M. Adusumilli and J. J. Hasenbein. Dynamic admission and service rate control of a queue. *Queueing Syst.*, 66(2):131–154, 2010.

[2] R. Agrawal, M. J. Carey, and M. Livny. Models for studying concurrency control performance: alternatives and implications. *SIGMOD Rec.*, 14(4):108–121, 1985.

[3] B. Ata and S. Shneorson. Dynamic control of an M/M/1 service system with adjustable arrival and service rates. *Management Science*, 52(11):1778–1791, 2006.

[4] B. Avi-Itzhak and S. Halfin. Expected response times in a non-symmetric time sharing queue with a limited number of service positions. *In Proceedings of ITC*, 12:5.4B.2.1–7, 1988.

[5] R. J. Batt and C. Terwiesch. Doctors under load: An empirical study of state-dependent service times. 2012.

[6] R. Blake. Optimal control of thrashing. In *Proceedings of the 1982 ACM SIGMETRICS Conference on Measurements and Modeling of Computer Systems*, 1982.

[7] M. Bramson. State space collapse with application to heavy traffic limits for multiclass queueing networks. *Queueing Syst.*, 30(1-2):89–148, 1998.

[8] A. Budhiraja and A. P. Ghosh. Diffusion approximations for controlled stochastic networks: an asymptotic bound for the value function. *Ann. Appl. Probab.*, 16(4):1962–2006, 2006.

[9] A. Budhiraja and C. Lee. Stationary distribution convergence for generalized jackson networks in heavy traffic. *Math. Oper. Res.*, 34(1):45–56, 2009.

[10] H. Chen and D. D. Yao. *Fundamentals of queueing networks*, volume 46 of *Applications of Mathematics (New York)*. Springer-Verlag, New York, 2001.

[11] P. J. Denning, K. C. Kahn, J. Leroudier, D. Potier, and R. Suri. Optimal multiprogramming. *Acta Informatica*, 7:197–216, 1976.

[12] S. Elnikety, E. Nahum, J. Tracy, and W. Zwaenepoel. A method for transparent admission control and request scheduling in e-commerce web sites. In *World-Wide-Web Conference*, 2004.

[13] W. H. Fleming and H. M. Soner. *Controlled Markov processes and viscosity solutions*, volume 25. Springer Science & Business Media, 2006.

[14] D. Gamarnik and P. Momčilović. Steady-state analysis of a multiserver queue in the Halfin-Whitt regime. *Adv. Appl. Probab.*, 40(2):548–577, 2008.

[15] D. Gamarnik and A. Zeevi. Validity of heavy traffic steady-state approximation in generalized Jackson networks. *Ann. Appl. Probab.*, 16(1):56–90, 2006.

[16] J. M. George and J. M. Harrison. Dynamic control of a queue with adjustable service rate. *Operations Research*, 49(5):pp. 720–731, 2001.

[17] H. C. Gromoll. Diffusion approximation for a processor sharing queue in heavy traffic. *Ann. Appl. Probab.*, 14(2):555–611, 2004.

[18] V. Gupta and M. Harchol-Balter. Self-adaptive admission control policies for resource-sharing systems. In *Proceedings of ACM SIGMETRICS '09*, pages 311–322, New York, NY, USA, 2009.

[19] V. Gupta and J. Zhang. Approximations and optimal control for state-dependent limited processor sharing queues. *arXiv preprint arXiv:1409.0153*, 2014.

[20] S. Halfin and W. Whitt. Heavy-traffic limits for queues with many exponential servers. *Operations research*, 29(3):567–588, 1981.

[21] J. M. Harrison, T. M. Sellke, and A. J. Taylor. Impulse control of Brownian motion. *Math. Oper. Res.*, 8(3):454–466, 1983.

[22] J. M. Harrison and M. I. Taksar. Instantaneous control of Brownian motion. *Math. Oper. Res.*, 8(3):439–453, 1983.

[23] H.-U. Heiss and R. Wagner. Adaptive load control in transaction processing systems. In *Proceedings of the 17th International Conference on Large Data Bases (VLDB)*, 1991.

[24] A. Janssen, J. v. Leeuwaarden, and J. Sanders. Scaled control in the QED regime. *Performance Evaluation*, 70(10):750–769, 2013.

[25] S. Karlin and H. M. Taylor. *A Second Course in Stochastic Processes.* Academic Press, 1981.

[26] E. Krichagina. Asymptotic analysis of queueing networks. *Stochastics and Stochastic Reports*, 40:43–76, 1992.

[27] E. V. Krichagina and A. A. Puhalskii. A heavy-traffic analysis of a closed queueing system with a GI/∞ service center. *Queueing Syst.*, 25(1-4):235–280, 1997.

[28] H. J. Kushner and P. Dupuis. *Numerical Methods for Stochastic Control Problems in Continuous Time.* Springer, 2001.

[29] C. Lee and A. A. Puhalskii. Non-Markovian state dependent networks in critical loading. arXiv preprint arXiv:1212:4078, 2012.

[30] C. Lee and A. Weerasinghe. Convergence of a queueing system in heavy traffic with general patience-time distributions. *Stochastic Processes and their Applications*, 121(11):2507–2552, 2011.

[31] N. Lee and V. Kulkarni. Optimal arrival rate and service rate control of multi-server queues. *Queueing Syst.*, 76(1):37–50, 2014.

[32] A. Mandelbaum and G. Pats. State-dependent stochastic networks. part I. approximations and applications with continuous diffusion limits. *Ann. Appl. Probab.*, 8(2):569–646, 05 1998.

[33] P. Mandl. *Analytical Treatment of One-Dimensional Markov Processes.* Academia, Springer, 1968.

[34] J. Nair, A. Wierman, and B. Zwart. Tail-robust scheduling via limited processor sharing. *Perform. Eval.*, 67(11):978–995, 2010.

[35] M. Puterman. Optimal control of diffusion processes with reflection. *Journal of Optimization Theory and Applications*, 22(1):103–116, 1977.

[36] M. L. Puterman and S. L. Brumelle. On the convergence of policy iteration in stationary dynamic programming. *Mathematics of Operations Research*, 4(1):pp. 60–69, 1979.

[37] J. E. Reed. The $G/GI/N$ queue in the Halfin-Whitt regime. *Ann. Appl. Probab.*, 19(6):2211–2269, 2009.

[38] K. Rege and M. Sengupta. Sojourn time distribution in a multiprogrammed computer system. *AT&T Tech. J.*, 64:1077–1090, 1985.

[39] A. R. Ward and S. Kumar. Asymptotically optimal admission control of a queue with impatient customers. *Mathematics of Operations Research*, 33(1):pp. 167–202, 2008.

[40] M. Welsh, D. Culler, and E. Brewer. SEDA: an architecture for well-conditioned, scalable internet services. *SIGOPS Oper. Syst. Rev.*, 35(5):230–243, 2001.

[41] K. Yamada. Diffusion approximation for open state-dependent queueing networks in the heavy traffic situation. *Ann. Appl. Probab.*, 5(4):958–982, 1995.

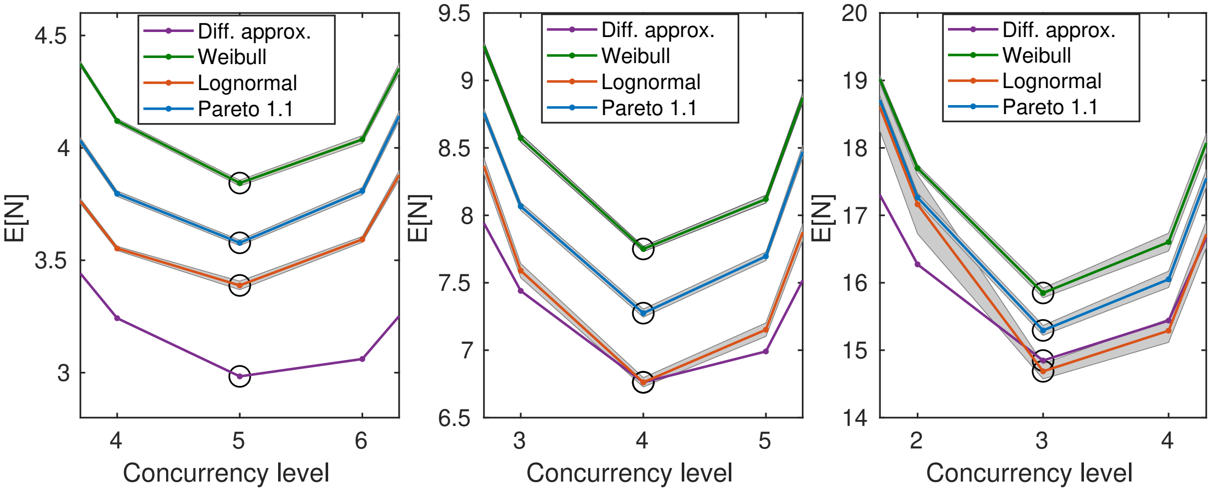
[42] G. Yamazaki and H. Sakasegawa. An optimal design problem for limited processor sharing systems. *Management Science*, 33(8):pp. 1010–1019, 1987.

[43] J. Zhang, J. G. Dai, and B. Zwart. Law of Large Number Limits of Limited Processor-Sharing Queues. *Math. Oper. Res.*, 34(4):937–970, 2009.

[44] J. Zhang, J. G. Dai, and B. Zwart. Diffusion Limits of Limited Processor-Sharing Queues. *Ann. Appl. Probab.*, 21(2):745–799, 2011.

[45] J. Zhang and B. Zwart. Steady state approximations of limited processor sharing queues in heavy traffic. *Queueing Syst.*, 60:227–246, 2008.

A Addenda to Experiments



(a) $\lambda = 0.7$ (b) $\lambda = 0.8$ (c) $\lambda = 0.9$

Figure 5: Simulation results with steady-state mean number of jobs in the system and 95% confidence intervals for Figure 2. The y-axis shows mean number of jobs in the system and the x-axis shows the concurrency level for the service rate function shown in Figure 1 for various job size distributions. Also shown is the diffusion approximation from equation (13). The confidence intervals highlight that the optimal concurrency levels (shown with circles) are indeed the unique optimal with at least 90% confidence.

B Diffusion and Steady State Analysis for the Workload Processes

Following from the dynamic equation (7), the diffusion-scaled workload is

$$\hat{W}^{(r)}(t) = \hat{W}^{(r)}(0) + \frac{1}{r}\sum_{i=1}^{\Lambda^{(r)}(r^2t)} v_i^{(r)} - \frac{1}{r}\int_0^{r^2t} \mu^{(r)}(Z^{(r)}(s))1_{\{W^{(r)}(s)>0\}}ds \tag{50}$$

Now, introduce the notations

$$\bar{K}^{(r)}(t,x) = \frac{1}{r^2}\sum_{i=1}^{\lceil r^2t\rceil} 1_{\{v_i^{(r)}\le x\}}, \tag{51}$$

$$\hat{K}^{(r)}(t,x) = r[\bar{K}^{(r)}(t,x) - tG(x)]. \tag{52}$$

The second term on the right-hand side of (50) can be written as

$$
\begin{aligned}
&r\int_0^t \int_0^\infty x d\bar{K}^{(r)}(\frac{1}{r^2}\Lambda^{(r)}(r^2 s), x) \\
&= \int_0^t \int_0^\infty x d\hat{K}^{(r)}(\frac{1}{r^2}\Lambda^{(r)}(r^2 s), x) + \frac{1}{r}\int_0^t \int_0^\infty x dG(x) d\Lambda^{(r)}(r^2 s) \\
&= \int_0^t \int_0^\infty x d\hat{K}^{(r)}(\frac{1}{r^2}\Lambda^{(r)}(r^2 s), x) + m\int_0^t d\frac{1}{r}[\Lambda^{(r)}(r^2 s) - \lambda r^2 s] + \lambda r m t.
\end{aligned}
$$

The last term on the right-hand side of (50) can be written as

$$
\begin{aligned}
&-r\int_0^t \mu^{(r)}(r\hat{Z}^{(r)}(s)) 1_{\{W^{(r)}(r^2 s)>0\}} ds \\
&= \int_0^t r[\lambda m - \mu^{(r)}(r\hat{Z}^{(r)}(s))]ds + r\int_0^t 1_{\{\hat{W}^{(r)}(s)=0\}} ds - \lambda r m t \\
&= \int_0^t r[\lambda m - \mu^{(r)}(r\Delta(\hat{W}^{(r)}(s)) \wedge k^{(r)})]ds + \int_0^t r[\mu^{(r)}(r\hat{Z}^{(r)}(s)) - \mu^{(r)}(r\Delta(\hat{W}^{(r)}(s)) \wedge k^{(r)})]ds \\
&\quad + r\int_0^t 1_{\{\hat{W}^{(r)}(s)=0\}} ds - \lambda r m t \\
&= \int_0^t \theta^{(r)}\left(\Delta(\hat{W}^{(r)}(s)) \wedge \frac{k^{(r)}}{r}\right) - \theta\left(\Delta(\hat{W}^{(r)}(s)) \wedge \frac{k^{(r)}}{r}\right) ds + r\int_0^t 1_{\{\hat{W}^{(r)}(s)=0\}} ds - \lambda r m t \\
&\quad + \int_0^t r[\mu^{(r)}(r\hat{Z}^{(r)}(s)) - \mu^{(r)}(r\Delta(\hat{W}^{(r)}(s)) \wedge k^{(r)})]ds + \int_0^t \theta\left(\Delta(\hat{W}^{(r)}(s)) \wedge \frac{k^{(r)}}{r}\right) ds
\end{aligned}
$$

In summary, we can write the workload process as

$$
\begin{aligned}
\hat{W}^{(r)}(t) = \hat{W}^{(r)}(0) + \hat{M}_s^{(r)}(t) + \hat{M}_a^{(r)}(t) + \hat{G}_1^{(r)}(t) + \hat{G}_2^{(r)}(t) \\
+ \int_0^t \theta\left(\Delta(\hat{W}^{(r)}(s)) \wedge \frac{k^{(r)}}{r}\right) ds + r\int_0^t 1_{\{\hat{W}^{(r)}(s)=0\}} ds,
\end{aligned}
\tag{53}
$$

where

$$
\hat{M}_s^{(r)}(t) = \int_0^t \int_0^\infty x d\hat{K}^{(r)}(\frac{1}{r^2}\Lambda^{(r)}(r^2 s), x), \tag{54}
$$

$$
\hat{M}_a^{(r)}(t) = m\int_0^t d\frac{1}{r}[\Lambda^{(r)}(r^2 s) - \lambda r^2 s], \tag{55}
$$

$$
\hat{G}_1^{(r)}(t) = \int_0^t r[\mu^{(r)}(r\hat{Z}^{(r)}(s)) - \mu^{(r)}(r\Delta(\hat{W}^{(r)}(s)) \wedge k^{(r)})]ds, \tag{56}
$$

$$
\hat{G}_2^{(r)}(t) = \int_0^t \theta^{(r)}\left(\Delta(\hat{W}^{(r)}(s) \wedge \frac{k^{(r)}}{r})\right) - \theta\left(\Delta(\hat{W}^{(r)}(s)) \wedge \frac{k^{(r)}}{r}\right) ds. \tag{57}
$$

The following lemma is an extension of the classical one-dimensional Skorohod problem. The proof can be found in [30].

Lemma 1 *Suppose g is a Lipschitz continuous function. For any $x \in \mathbf{D}(\mathbb{R}^+)$, there exists a unique pair $(y, z) \in \mathbf{D}^2(\mathbb{R}^+)$ satisfying*

$$z(t) = \int_0^t g(z(s))ds + x(t) + y(t), \tag{58}$$

$$z(t) \geq 0, \quad \textit{for all } t \geq 0, \tag{59}$$

$$y(0) = 0 \textit{ and } y \textit{ is non-decreasing}, \tag{60}$$

$$\int_0^t z(s)dy(s) = 0. \tag{61}$$

More over, denote $z = \psi(x)$. The mapping $\psi : \mathbf{D}(\mathbb{R}^+) \to \mathbf{D}(\mathbb{R}^+)$ is continuous in the uniform topology on compact set.

Proof of Theorem 1: We first study the first four terms on the right-hand side of equation (53). For the initial condition $\hat{W}^{(r)}(0)$, its convergence to some random variable w_0 is part of the assumption (20) on the initial state.

According to Lemma 3.8 in [27],

$$\int_0^t \int_0^\infty x d\hat{K}^{(r)}(\frac{1}{r^2}\Lambda^{(r)}(r^2 s), x) \Rightarrow \sqrt{\lambda} m c_s M_s(t), \quad \text{as } r \to \infty,$$

where $M_s(t)$ is a standard Brownian motion (with zero drift and variance 1).

It follows from the assumption (14) that

$$\hat{M}_a^{(r)}(t) = m\hat{\Lambda}^{(r)}(t) \Rightarrow \sqrt{\lambda} m c_a M_a(t), \quad \text{as } r \to \infty.$$

We now study the terms $\hat{G}_1^{(r)}$ and $\hat{G}_2^{(r)}$. By the stochastic bound (Lemma 2) proved in Section C, for any $\epsilon > 0$, there exists C such that $\mathbb{P}(\Omega_r) \geq 1-\epsilon$, where $\Omega_r = \left\{\sup_{t\in[0,T]} \max\left(\hat{Z}^{(r)}(s), \Delta\hat{W}^{(r)}(s)\right) \leq C\right\}$ (noting that we naturally have $\hat{Z}^{(r)}(\cdot) \leq k^{(r)}/r$). According to condition (16), for any sample path in the event Ω_r, we have

$$\hat{G}_1^{(r)}(t) \Rightarrow 0, \qquad \hat{G}_2^{(r)}(t) \Rightarrow 0, \quad \text{as } r \to \infty.$$

Let $\hat{Y}^{(r)}(t) = r\int_0^t 1_{\{\hat{W}^{(r)}(s)=0\}} ds$. It is easy to see that

$$\int_0^t \hat{W}^{(r)}(s) d\hat{Y}^{(r)}(s) = 0. \tag{62}$$

Thus $(\hat{W}^{(r)}, \hat{Y}^{(r)})$ is the solution to the reflection mapping in Lemma 1. So

$$\hat{W}^{(r)} = \psi\left(\hat{W}^{(r)}(0) + \hat{M}_s^{(r)} + \hat{M}_a^{(r)} + \hat{G}_1^{(r)} + \hat{G}_2^{(r)}\right).$$

By the continuous mapping theorem, $\hat{W}^{(r)} \Rightarrow W^*$, where $W^* = \psi(w_0 + \sqrt{\lambda} m c_s M_s(t) + \sqrt{\lambda} m c_a M_a(t))$. In other words, the limit W^* satisfies

$$W^*(t) = w_0 + \sqrt{\lambda} m c_s M_s(t) + \sqrt{\lambda} m c_a M_a(t) - \theta(\Delta(W^*))(t) + Y^*(t), \tag{63}$$

with $Y^*(0) = 0$ and being non-decreasing and

$$\int_0^t W^*(s)dY^*(s) = 0. \tag{64}$$

Thus, we have shown that the diffusion limit of the workload process is an RBM with state-dependent drift $-\theta(\Delta_K(W^*(t)) \wedge K)$ and variance $\lambda m^2(c_s^2 + c_a^2)$. The proof of (25) follows immediately from the continuous mapping theorem. ■

Proof of Proposition 2: A standard result (see [25, Chapter 15]) gives the stationary distribution of a one-dimensional RBM, W, with state-dependent drift $-\beta(\cdot)$ and state-dependent variance $s(\cdot)$ to be

$$\mathbf{Pr}[W(\infty) \le w] = \alpha \int_0^w e^{-\int_0^u \frac{\beta(v)+\frac{1}{2}s'(v)}{\frac{1}{2}s(v)}dv} du = \alpha \int_0^w \frac{1}{s(u)} e^{-\int_0^u \frac{\beta(v)}{\frac{1}{2}s(v)}dv} du, \tag{65}$$

where α is a normalization constant.

We start from (65) and substitute state-dependent variance and drift as

$$s(w) = \lambda m^2(c_a^2 + c_s^2)$$
$$\beta(w) = \begin{cases} \theta(w/m_e) = -\lambda m \left.\frac{d\log f(x)}{dx}\right|_{x=\frac{w}{m_e}} & w \le K \cdot m_e \\ \theta(K) = -\lambda m \left.\frac{d\log f(x)}{dx}\right|_{x=K} & w > K \cdot m_e \end{cases}$$

To obtain a further simplification, we use our assumption that $\frac{d\log f(x)}{dx}$ is a constant for $x \ge K$, and therefore

$$\beta(w) = -\lambda m \left.\frac{d\log f(x)}{dx}\right|_{x=\frac{w}{m_e}}, \quad \forall\ w \in [0,\infty)$$

We then get

$$\begin{aligned} \mathbf{Pr}[W^*(\infty) \le w] &= \frac{\alpha'}{\lambda m^2(c_a^2 + c_s^2)} \int_0^w e^{\frac{2}{\lambda m^2(c_a^2+c_s^2)} \int_0^u \lambda m d\log f(v/m_e)} du \\ &= \frac{\alpha'}{\lambda m^2(c_a^2 + c_s^2)} \int_0^w e^{\frac{2\lambda m m_e}{\lambda m^2(c_a^2+c_s^2)} \int_0^{u/m_e} d\log f(z)} du \\ &= \frac{\alpha''}{\lambda m^2(c_a^2 + c_s^2)} \int_0^w e^{\frac{1+c_s^2}{c_a^2+c_s^2} \log f(u/m_e)} du \\ &= \frac{\alpha''}{\lambda m^2(c_a^2 + c_s^2)} \int_0^w f\left(\frac{u}{m_e}\right)^{\frac{1+c_s^2}{c_a^2+c_s^2}} du \\ &= \alpha \int_0^{\frac{w}{m_e}} f(u)^{\frac{1+c_s^2}{c_a^2+c_s^2}} du \end{aligned}$$

which proves (26).
From (25) and the continuous mapping theorem

$$X^*(\infty) = \frac{W^*(\infty) \wedge K m_e}{m_e} + \frac{(W^*(\infty) - K m_e)^+}{m}. \tag{66}$$

It now follows that

$$\mathbf{Pr}[X^*(\infty) \le x] = \begin{cases} \mathbf{Pr}[W^*(\infty) \le x m_e] & x \le K \\ \mathbf{Pr}[W^*(\infty) \le K m_e + (x-K)m] & x > K \end{cases}$$

which, together with (26), gives (27).
To find $\mathbf{E}[X^*(\infty)]$, we will find it convenient to start with (26) and rewrite it as

$$\mathbf{Pr}\left[\frac{W^*(\infty)}{m_e} \le z\right] = \alpha \int_0^z f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx. \tag{67}$$

Therefore, $f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}}$ is the density of $\frac{W^*(\infty)}{m_e}$. Now we again use the map (66) to write

$$\begin{aligned} \mathbf{E}[X^*(\infty)] &= \mathbf{E}\left[\frac{W^*(\infty)}{m_e} \wedge K\right] + \frac{m_e}{m}\mathbf{E}\left[\left(\frac{W^*(\infty)}{m_e} - K\right)^+\right] \\ &= \frac{\int_0^\infty (x \wedge K) f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx}{\int_0^\infty f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx} + \frac{c_s^2+1}{2} \frac{\int_0^\infty (x-K)^+ f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx}{\int_0^\infty f(x)^{\frac{c_s^2+1}{c_s^2+c_a^2}} dx}, \end{aligned}$$

which proves (28). ■

Proof of Theorem 2: This theorem essentially establishes the interchange of the steady state and heavy traffic limits for the constructed sequence of Sd-LPS models. Proving such an interchange usually involves quite a complicated analysis of a well-constructed Lyapunov function (see, for example, [15] and [9]). Taking advantage of the existing studies, we use a coupling argument to prove the interchange for our model. The proofs for both the workload and queue length essentially follow the same argument. We only focus on the queue length in this proof.

For each r, we construct an auxiliary system which takes exactly the same arrival stream as the rth Sd-LPS system and the same initial condition. Denote

$$\mu_\dagger^{(r)} = \mu^{(r)}(k^{(r)}).$$

When the number of jobs in the auxiliary system is more than $k^{(r)}$, the server works at rate $\mu_\dagger^{(r)}$. When the number of jobs drops below $k^{(r)}$, the server works at speed 0 (in other words it completely shuts down). Without loss of generality, we assume that the initial number of jobs is larger than $k^{(r)}$. Let $Q^{(r)}(t)$ and $Q_\dagger^{(r)}(t)$ denote the number of jobs in the queue in the Sd-LPS and auxiliary systems, respectively. It is clear that

$$Q^{(r)}(t) < Q_\dagger^{(r)}(t). \tag{68}$$

Due to parallel processing, overtaking can happen in each system, i.e., the jth arriving job may leave the system earlier than the ith arriving job even if $j > i$. However, due to the coupling, the ith arriving job in the auxiliary system can never enter service earlier than the corresponding job in the Sd-LPS system.

By condition (17), $\mu_\dagger^{(r)} > \lambda m$ for all large enough r. So both $Q^{(r)}$ and $Q_\dagger^{(r)}$ are stationary. Let $\pi^{(r)}$ denote the stationary probability measure of the diffusion-scaled process $\hat{Q}^{(r)}$. Similarly, Let $\pi_\dagger^{(r)}$

denote the stationary probability measure of the diffusion-scaled queue length $\hat{Q}_{\dagger}^{(r)}$ in the coupled system. The key step to showing that $X^{(r)}(\infty) \Rightarrow X^*(\infty)$ as $r \to \infty$ is to show that the family of probability measures $\{\pi^{(r)}\}_{r\in\mathbb{N}}$ is tight. (Since $\hat{X}^{(r)}(t) \leq \hat{Q}^{(r)}(t) + k^{(r)}/r$, studying only the queue length suffices.) Readers can refer to the proof of Theorem 8 in [15] for a standard argument of how to prove the convergence using tightness. We now focus on proving the tightness of probability measures $\{\pi^{(r)}\}_{r\in\mathbb{N}}$.

We can model the rth auxiliary system as if it has $k^{(r)}$ identical servers. All the servers either work or stop in perfect synchronization. Denote by $S_{\dagger,i}^{(r)}(\cdot)$, $i = 1, \dots, k^{(r)}$, independent renewal processes with inter-renewal time following distribution $G(\cdot/k^{(r)})$, where G is the distribution of job sizes. In other words, the inter-renewal time has mean $mk^{(r)}$ and SCV c_s^2. The queueing dynamics of the rth auxiliary system can be written as

$$Q_{\dagger}^{(r)}(t) = Q_{\dagger}^{(r)}(0) + \Lambda^{(r)}(t) - \sum_{i=1}^{k^{(r)}} S_{\dagger,i}^{(r)}(B_{\dagger}^{(r)}(t)),$$

where $B_{\dagger}^{(r)}(t)$ is the cumulative busy time for each of the servers. Applying the diffusion scaling, we have

$$\hat{Q}_{\dagger}^{(r)}(t) = \hat{Q}_{\dagger}^{(r)}(0) + \hat{\Lambda}^{(r)}(t) - \sum_{i=1}^{k^{(r)}} \hat{S}_{\dagger,i}^{(r)}(\frac{1}{r^2}B_{\dagger}^{(r)}(r^2t)) + r(\lambda - \mu_{\dagger}^{(r)}/m)t + \frac{\mu_{\dagger}^{(r)}}{rm}(r^2t - B^{(r)}(r^2t)), \quad (69)$$

where

$$\hat{\Lambda}^{(r)}(t) = \frac{1}{r}\left(\Lambda^{(r)}(r^2t) - \lambda r^2 t\right),$$

$$\hat{S}_{\dagger,i}^{(r)}(t) = \frac{1}{r}\left(S_{\dagger,i}^{(r)}(r^2t) - \frac{\mu_{\dagger}^{(r)} r^2}{mk^{(r)}}t\right).$$

Note that $r^2t - B^{(r)}(r^2t)$ increases only when $\hat{Q}_{\dagger}^{(r)}(t) = 0$, so (69) is the same as the Skorohod mapping for the $G/G/1$ queue except that the service process is the superposition of $k^{(r)}$ renewal processes with a much lower speed (roughly $1/k^{(r)} \approx 1/r$) rather than a single renewal process. It follows from Lemma 3.5 in [8] that the centerlized renewal arrival satisfies

$$\mathbb{E}\left[\sup_{0\leq s\leq t} |\hat{\Lambda}^{(r)}(s)|^2\right] < C_a(1+t),$$

for some constants C_a and all large enough r. The analysis centerlized service processes requires slight modification of the proof of Lemma 3.5 in [8], where (3.31) and (3.32) can be enhanced as follows: The right-hand side of the second inequality in (3.20), which is $C^*(1+t)$, can be replaced by $\frac{2}{r} + \frac{C_2}{r^2} + 3C_2t$. Since our renewal process $S_{\dagger,i}^{(r)}$ has speed $1/k^{(r)}$ rather than 1, by time change, we can replace the time t by $\frac{t}{k^{(r)}}$. So

$$\mathbb{E}\left[\sup_{0\leq s\leq t} |\hat{S}_{\dagger,i}^{(r)}(s)|^2\right] < \frac{2}{r} + \frac{C_2}{r^2} + 3C_2\frac{t}{k^{(r)}} \leq \frac{3}{r} + 6KC_2\frac{t}{r}$$

for some constants C_2 and all large enough r. Consequently,

$$\mathbb{E}\sup_{0\leq s\leq t}\left(|\hat{\Lambda}^{(r)}(s)|^2 + \sum_{i=1}^{k^{(r)}} |\hat{S}_{\dagger,i}^{(r)}(s)|^2\right) < C_a(1+t) + 3 + 6KC_2t < C(1+t),$$

for some constant C and all large enough r. Thus, we have verified that the centerlized renewal arrival and service processes in the queueing dynamic equation (69) satisfies condition (A8.p) in [9] for $p = 2$. Note that the dynamic equation (69) is essentially the same as the one in [9] except that we do not have routing processes. So following exactly the same argument, in fact slightly simpler due to the missing of routing processes, Theorem 3.3 in [9] holds for our problem. This implies Theorem 3.2 in [9], i.e., $\sup_r \int_0^\infty w\pi_\dagger^{(r)}(dw) < \infty$. By the coupling construction (68), we have $\int_0^\infty x\pi^{(r)}(dx) < \int_0^\infty x\pi_\dagger^{(r)}(dx)$. This implies tightness of $\{\pi^{(r)}\}_{r\in\mathbb{N}}$. ■

C State Space Collapse for the Sd-LPS system

We introduce a strengthened version of the mapping Δ_K as the follows. Let $\Delta_{K,\nu} : \mathbb{R}_+ \to \mathbf{M}\times\mathbf{M}$ be the lifting map associated with the probability measure ν and constant K given by

$$\Delta_{K,\nu} w = \Big(\frac{(w - K\beta_e)^+}{\beta}\nu, \frac{w \wedge K\beta_e}{\beta_e}\nu_e\Big) \quad \text{for } w \in \mathbb{R}_+.$$

We aim to prove the following full version of the SSC

Theorem 3 (Full State Space Collapse) *Under the conditions (14)–(16) and (19)–(21), for any $T > 0$,*

$$\sup_{t\in[0,T]} \mathbf{d}[(\hat{\mathcal{Q}}^{(r)}(t), \hat{\mathcal{Z}}^{(r)}(t)), \Delta_{K,\nu}\hat{W}^{(r)}(t)] \Rightarrow 0 \quad \textit{as } r \to \infty.$$

It is clear that Theorem 3 implies Proposition 1. The rest of this section is devoted to the proof of the SSC.

C.1 Tightness of Shifted Fluid-Scaled Processes

The key to proving SSC, which was originally developed by [7], is to "chop" the diffusion-scaled processes into pieces.

Shifted Fluid Scaling Introduce,

$$\bar{\mathcal{Q}}^{(r,l)}(t) = \frac{1}{r}\mathcal{Q}^{(r)}(rl + rt), \quad \bar{\mathcal{Z}}^{(r,l)}(t) = \frac{1}{r}\mathcal{Z}^{(r)}(rl + rt), \tag{70}$$

for all $m \in \mathbb{N}$ and $t \geq 0$. To see the relationship between these two scalings, consider the diffusion-scaled process on the interval $[0, T]$, which corresponds to the interval $[0, r^2T]$ for the unscaled process. Fix a constant $L > 1$, the interval will be covered by $\lfloor rT\rfloor + 1$ overlapping intervals

$$[rl, rl + rL] \quad l = 0, 1, \cdots, \lfloor rT\rfloor.$$

For each $t \in [0, T]$, there exists an $l \in \{0, \cdots, \lfloor rT\rfloor\}$ and $s \in [0, L]$ (which may not be unique) such that $r^2t = rl + rs$. Thus

$$\hat{\mathcal{Q}}^{(r)}(t) = \bar{\mathcal{Q}}^{(r,l)}(s), \quad \hat{\mathcal{Z}}^{(r)}(t) = \bar{\mathcal{Z}}^{(r,l)}(s). \tag{71}$$

This will serve as a key relationship between fluid and diffusion-scaled processes.

The quantities $Q^{(r)}(\cdot)$, $Z^{(r)}(\cdot)$, $X^{(r)}(\cdot)$, $W^{(r)}(\cdot)$ are essentially functions of $(\mathcal{Q}^{(r)}(\cdot), \mathcal{Z}^{(r)}(\cdot))$, so the scaling for these quantities is defined as the functions of the corresponding scaling for $(\mathcal{Q}^{(r)}(\cdot), \mathcal{Z}^{(r)}(\cdot))$. For example

$$\bar{W}^{(r,l)}(t) = \langle\chi, \bar{\mathcal{Q}}^{(r,l)}(t) + \bar{\mathcal{Z}}^{(r,l)}(t)\rangle = \frac{1}{r}W^{(r)}(rl + rt).$$

We define the shifted fluid scaling for the arrival process as

$$\bar{\Lambda}^{(r,l)}(t) = \frac{1}{r}\Lambda^{(r)}(rl + rt),$$

for all $t \geq 0$. By (6), the shifted fluid scaling for $B^{(r)}(\cdot)$ is

$$\bar{B}^{(r,l)}(t) = \bar{E}^{(r,l)}(t) - \bar{\mathcal{Q}}^{(r,l)}(t),$$

for all $t \geq 0$. A shifted fluid-scaled version of the stochastic dynamic equations (4) and (5) can be written as, for any $A \subset (0, \infty)$, $0 \leq s \leq t$,

$$\begin{aligned} \bar{\mathcal{Q}}^{(r,l)}(t)(A) = \bar{\mathcal{Q}}^{(r,l)}(s)(A) + \frac{1}{r} \sum_{i=r\bar{E}^{(r,l)}(s)+1}^{r\bar{E}^{(r,l)}(t)} \delta_{v_i}(A) \\ - \frac{1}{r} \sum_{i=r\bar{B}^{(r,l)}(s)+1}^{r\bar{B}^{(r,l)}(t)} \delta_{v_i}(A), \end{aligned} \tag{72}$$

$$\begin{aligned} \bar{\mathcal{Z}}^{(r,l)}(t)(A) = \bar{\mathcal{Z}}^{(r,l)}(s)(A + S^{(r)}(rl + rs, rl + rt)) \\ + \frac{1}{r} \sum_{i=r\bar{B}^{(r,l)}(s)+1}^{r\bar{B}^{(r,l)}(t)} \delta_{v_i^{(r)}}(A + S^{(r)}(\tau_i^{(r)}, rl + rt)). \end{aligned} \tag{73}$$

We point out that the cumulative service process $S^{(r)}$ is never scaled because it tracks the amount of service received by each individual customer. However, via some algebra we can see that

$$S^{(r)}(rl + rs, rl + rt) = \int_{rl+rs}^{rl+rt} \frac{\mu^{(r)}(Z^{(r)}(\tau))}{Z^{(r)}(\tau)} d\tau = \int_{s}^{t} \frac{\mu^{(r)}(r\bar{Z}^{(r,l)}(\tau))}{\bar{Z}^{(r,l)}(\tau)} d\tau. \tag{74}$$

This gives two interesting observations. First, the shifted fluid scaling is essentially fluid scaling, meaning the shifted fluid-scaled processes should be close to some fluid model solutions. Second, the corresponding fluid model is essentially the same as the fluid model in [44] since by (16),

$$\mu^{(r)}(r\bar{Z}^{(r,l)}(\tau)) = 1 + O^{+}(\frac{1}{r}),$$

where $O^{+}(1/r)$ means the quantity is positive and of the same order as $1/r$ when $r \to \infty$. So

$$S^{(r)}(rl + rs, rl + rt) = \int_{s}^{t} \frac{1}{\bar{Z}^{(r,l)}(\tau)} d\tau + O^{+}(\frac{1}{r}). \tag{75}$$

Intuitively, $\bar{Z}^{(r,l)}$ is close to some fluid limit denoted by $\tilde{Z}$ as r becomes very large (in the mathematical sense of convergence in probability), then

$$S^{(r)}(rl + rs, rl + rt) \Rightarrow \int_{s}^{t} \frac{1}{\tilde{Z}(\tau)} d\tau. \tag{76}$$

So we can conclude that the underlying fluid is the same as the one for the regular LPS system. Thus, we can use existing properties developed in [44]. We hope to make the argument rigorous and concise in the follows.

Stochastic Boundedness

The tightness property, which guarantees that the shifted fluid-scaled process $\{\bar{\mathcal{Q}}^{(r,l)}, \bar{\mathcal{Z}}^{(r,l)}\}$ has a convergent subsequence, can be proved in a similar way as in [44]. There are two key differences. First is the service process as pointed out before. Second is that [44] heavily relies on the known result on the diffusion of the workload (see Proposition 2.1). However, we do not have such a diffusion limit of workload a priori. Instead, we try to prove such a diffusion limit by SSC. Looking into the details of the machinery in [44], what essentially is needed for the workload process is some kind of stochastic bound, which we prove in the following lemma.

Lemma 2 (An Upper Bound of the Workload) *For any $\eta > 0$ there exists a constant M such that*

$$\mathbb{P}\left(\max_{l\leq rT}\sup_{t\in[0,L]} \bar{W}^{(r,l)}(t) < M\right) > 1-\eta. \tag{77}$$

Proof: Using the relationship between the shifted fluid scaling and diffusion scaling, we essentially need to prove that

$$\mathbb{P}\left(\sup_{t\in[0,L]} \hat{W}^{(r)}(t) < M\right) > 1-\eta.$$

Recall the representation (53) for the diffusion-scaled workload processes. Let $\underline{\theta} = \inf_{x\in[0,K]}\theta(x)$, which is finite due to condition (16), so the process $\hat{W}_1^{(r)}$ satisfying

$$\hat{W}_1^{(r)}(t) = \hat{W}^{(r)}(0) - \underline{\theta}t + \hat{M}_s^{(r)}(t) + \hat{M}_a^{(r)}(t) + \hat{G}_1^{(r)}(t) + \hat{G}_2^{(r)}(t) + r\int_0^t 1_{\{\hat{W}_1^{(r)}(s)=0\}}ds$$

is an upperbound of $\hat{W}^{(r)}$ due to the definition of $\underline{\theta}$ and condition (17). By Lemma 1, $\hat{W}_1^{(r)}$ converges to a driftless RBM, which is stochastically bounded. This implies the result. ■

Such a stochastic bound of the workload process helps to establish some useful bound estimates for the stochastic processes underlying the Sd-LPS model.

Lemma 3 (Further Bound Estimations) *For any $\eta > 0$, there exists a constant $M > 0$ and a probability event $\Omega_B^r(M)$ for each index r such that*

$$\liminf_{r\to\infty}\mathbb{P}\left(\Omega_B^r(M)\right) \geq 1-\eta, \tag{78}$$

and on the event $\Omega_B^r(M)$, we have

$$\max_{l\leq\lfloor rT\rfloor}\sup_{t\in[0,L]} \bar{Q}^{(r,l)}(t) \leq M, \tag{79}$$

$$\max_{l\leq\lfloor rT\rfloor}\sup_{t\in[0,L]} \langle\chi^{1+p}, \bar{\mathcal{Q}}^{(r,l)}(t) + \bar{\mathcal{Z}}^{(r,l)}(t)\rangle \leq M. \tag{80}$$

Proof: The result (79) holds due to Lemma 4.2 in [44], which only utilizes the regularity of the arrival process (14) and the stochastic bound (77) for the workload process proved in Lemma 2. For (80), the first half, $\max_{l\leq\lfloor rT\rfloor}\sup_{t\in[0,L]}\langle\chi^{1+p}, \bar{\mathcal{Q}}^{(r,l)}(t)\rangle \leq M$, also follows the same reasoning as Lemma 4.3 in [44]. Essentially, any results for the "queue" part follows the same argument in [44].

The challenge with the state-dependent service rate lies in the analysis of the server. It follows from the shifted fluid-scaled dynamic equation (73) that for any Borel set $A \subset (0, \infty)$,

$$\begin{aligned}\frac{1}{r}\mathcal{Z}^{(r)}(rl+rt)(A) = \frac{1}{r}\mathcal{Z}^{(r)}(0)(A + S^{(r)}(0, rl+rt))\\ + \sum_{j=0}^{m-1}\frac{1}{r}\sum_{i=B^{(r)}(r(l-j-1))+1}^{B^{(r)}(r(l-j))} \delta_{v_i}(A + S^{(r)}(\tau_i^{(r)}, rl+rt))\\ + \frac{1}{r}\sum_{i=B^{(r)}(rl)+1}^{B^{(r)}(rl+rt)} \delta_{v_i}(A + S^{(r)}(\tau_i^{(r)}, rl+rt)).\end{aligned}$$

Given $0 \le j \le m-1$, for those i's with $B^{(r)}(r(l-j-1)) < i \le B^{(r)}(r(l-j))$ we have

$$\tau_i^{(r)} \in [r(l-j-1), r(l-j)].$$

For the sake of simplicity, let us assume that $Z^{(r)}(s) > 0$ for all $s \in [0, rl+rt]$. If this does not hold, we can use a technical trick presented in the proof of Lemma 4.3 in [44] to deal with it. Here we show the main difference coming from the state-dependent service rate. By (75) and the fact that $Z^{(r)} \le k^{(r)}$, we have a lower bound on the cumulative service amount

$$S^{(r)}(rs, rt) \ge \int_{rs}^{rt} \frac{1}{Z^{(r)}(s)} ds \ge \frac{r(t-s)}{k^{(r)}}. \tag{81}$$

Thus,

$$S^{(r)}(\tau_i^{(r)}, rl+rt) \ge S^{(r)}(r(l-j), rl) \ge \frac{rj}{k^{(r)}} \ge \frac{j}{2K},$$

for all large r where the last inequality is due to (9). For those i's such that $\tau_i^{(r)}$ is larger than $B^{(r)}(rl)$, we use the trivial lower bound $S^{(r)}(\tau_i^{(r)}, rl+rt) \ge 0$. Also take the trivial lower bound that $S^{(r)}(0, rl+rt) \ge 0$. Then we have the following inequality on the $(1+p)$th moment:

$$\begin{aligned}\langle \chi^{1+p}, \frac{1}{r}\mathcal{Z}^{(r)}(rl+rt)\rangle \le \langle \chi^{1+p}, \frac{1}{r}\mathcal{Z}^{(r)}(0)\rangle\\ + \sum_{j=0}^{m-1}\langle \big((\chi - \frac{j}{2K})^+\big)^{1+p}, \frac{1}{r}\sum_{i=B^{(r)}(r(l-j-1))+1}^{B^{(r)}(r(l-j))} \delta_{v_i}\rangle\\ + \langle \chi^{1+p}, \frac{1}{r}\sum_{i=B^{(r)}(rl)+1}^{B^{(r)}(rl+rt)} \delta_{v_i}\rangle.\end{aligned} \tag{82}$$

This is the same as (4.22) in [44]. The estimation of the first term on the right-hand side in the above follows directly from the initial condition (20). The analysis of the second and third terms follows the same way as in [44]. ■

To prove that a family of measure-valued processes is tight, there are three properties to verify, namely *Compact Containment*, *Asymptotic Regularity* and *Oscillation Bound*. For brevity, we will not repeat the exact mathematical statements and their proofs. For the LPS system, these three properties were proved in Lemmas 4.4–4.6 in [44]. We just point out that the proof for the above mentioned three properties for the Sd-LPS system relies on (a) the bound estimate in Lemma 3; and (b) the fact that (75) implies the lower bound of the cumulative service process (81). The proof of Lemma 3 has demonstrated point (b) clearly, we therefore omit a repeat of the argument used in [44]. So we reach the conclusion:

Proposition 6 (Tightness of Shifted Fluid-scaled Processes) *The family of shifted fluid-scaled processes* $\{(\bar{\mathcal{Q}}^{(r,l)}, \bar{\mathcal{Z}}^{(r,l)}\}_{l\leq rT, r\in\mathbb{N}}$ *is tight.*

Loosely speaking, tightness means that any subsequence from the family of shifted fluid-scaled processes has a convergent subsequence. This is formally stated in Theorem 4.1 in [44].

C.2 Bramson's Framework for SSC

With all the above preparation, we can now prove Theorem 3. Note that Sd-LPS and LPS (studied in [44]) essentially use the same measure-valued framework. The difference lies in the cumulative service process as we explained when deriving (75) and the workload process as we studied in Lemma 2. After obtaining the tightness in Proposition 6, we can apply the framework invented by Bramson [7] in the same way as how Section 5 in [44] applies it to the LPS.

To avoid repeating all details, we only provide a sketch of the proof with the intention of making it easier to read. To prove the SSC in Theorem 3, we just need to restrict on the event $\Omega_B^r(M)$, which has probability $1-\epsilon$ by Lemma 3. The goal is to show that on this event, each $\epsilon > 0$, there exists an r_0 such that when $r > r_0$,

$$\sup_{t\in[0,T]} \mathbf{d}[(\hat{\mathcal{Q}}^{(r)}(t), \hat{\mathcal{Z}}^{(r)}(t)), \Delta_{K,\nu}\hat{W}^{(r)}(t)] < \epsilon. \tag{83}$$

We fix $r > r_0$ and a sample path in $\Omega_B^r(M)$ for the rest of the discussion.

Note that (83) is about diffusion. In order to study this, we utilize the relationship (71) between diffusion scaling and shifted fluid scaling. For any constant $L > L^* + 1$ (with L^* to be specified later), note that

$$[0, r^2T] \subset [0, rL^*] \cup \bigcup_{m=0}^{\lfloor rT\rfloor} [r(m+L^*), r(m+L)].$$

By the definition of diffusion and shifted fluid scaling, to show (83) it suffices to show

$$\max_{m\leq\lfloor rT\rfloor} \sup_{s\in[L^*,L]} \mathbf{d}[(\bar{\mathcal{Q}}^{(r,l)}(s), \bar{\mathcal{Z}}^{(r,l)}(s)), \Delta_{K,\nu}\bar{W}^{(r,l)}(s)] < \epsilon, \tag{84}$$

$$\sup_{s\in[0,L^*]} \mathbf{d}[(\bar{\mathcal{Q}}^{(r,0)}(s), \bar{\mathcal{Z}}^{(r,0)}(s)), \Delta_{K,\nu}\bar{W}^{(r,0)}(s)] < \epsilon. \tag{85}$$

The high level-logic is as the follows: the shifted fluid-scaled processes are "close" to the fluid model solution, and the fluid model solution converges to some invariant which exhibits SSC (Theorem 3.1 in [44]). Thus SSS can be proved for (84) and (85).

By Theorem 3.1 in [44], there exists an $L^* > 0$ such that when $s > L^*$,

$$\mathbf{d}[(\tilde{\mathcal{Q}}(s), \tilde{\mathcal{Z}}(s)), \Delta_{K,\nu}\tilde{W}(s)] < \epsilon/3, \tag{86}$$

where $(\tilde{\mathcal{Q}}(\cdot), \tilde{\mathcal{Z}}(\cdot))$ denote the fluid limit which exhibits SSC. Thus, (84) follows from triangular inequality by inserting the above fluid limit. Note that (85) follows from the same idea of triangular inequality, but rely on the initial conditions (19)–(21) specified in Theorem 3.

D Analysis of Algorithm for Finding Optimal Control

Recall some notation and definitions used in this section.

$$\begin{aligned}
\hat{k} &= \arg\max_{k} \theta(k) \\
\hat{\theta} &= \theta(\hat{k}) \\
\Delta_k(w) &= \frac{w}{m} + k\left(1 - \frac{m_e}{m}\right) \\
d_\theta &= \sup_k \theta(k) - \inf_k \theta(k) \\
k_f(w) &= \underset{k\in[0,w/m_e]}{\arg\max}\ \theta(k_f)
\end{aligned}$$

Throughout this section, we assume that d_θ is finite, and therefore $\theta(k)$ is bounded from above and below.

D.1 Some Auxiliary Results

We first provide some auxiliary results (Lemmas 4 and 5) which will be useful in proving the results in Section 4.

Lemma 4 *Consider the solution of the following ODE, parameterized by v and W:*

Terminal condition:

$$G_{v,W}(w) = \alpha w + \beta v + \gamma \qquad \ldots w \geq \max\{W, \hat{k}m_e\}$$

ODE:

$$\begin{aligned}
v &= \frac{w}{m} + k_f(w)\left(1 - \frac{m_e}{m}\right) - \theta(k_f(w))G_{v,W}(w) + \frac{\sigma^2}{2}G'_{v,W}(w) && \ldots w \in [W, \hat{k}m_e] \\
v &= \min_{k\in[0,w/m_e]}\left\{\frac{w}{m} + k\left(1 - \frac{m_e}{m}\right) - \theta(k)G_{v,W}(w) + \frac{\sigma^2}{2}G'_{v,W}(w)\right\} && \ldots w \in [0, W]
\end{aligned}$$

Then $G_{v,W}(w)$ is continuous in both v and W for all w.

Proof: Let (v_a, W_a) and (v_b, W_b) denote two parameter settings, and for succinctness, denote the corresponding solutions to the ODE as G_a and G_b, respectively. We will consider the case $W_a, W_b \geq \hat{k}m_e$ as other cases are analogous.

Let $W_a \leq W_b$.

At $w = W_b$, we have

$$|G_a(W_b) - G_b(W_b)| = \beta|v_a - v_b|. \tag{87}$$

For $w \in [W_a, W_b]$, we have

$$\begin{aligned}
G_a(w) &= \alpha w + \beta v_a + \gamma, && (88) \\
G'_b(w) &= \frac{2}{\sigma^2}\left(v_b + \theta(k_b(w))G_b(w) - \frac{w}{m} + k_b\left(1 - \frac{m_e}{m}\right)\right), && (89)
\end{aligned}$$

which gives

$$\frac{2}{\sigma^2}\left(v_b - \frac{W_b}{m \wedge m_e} - d_\theta G_b(w)\right) \leq G_b'(w) \leq \frac{2}{\sigma^2}\left(v_b - \frac{W_a}{m \vee m_e} + d_\theta G_b(w)\right). \quad (90)$$

Since the derivatives are bounded, $G_b(w)$ is bounded in the interval $[W_a, W_b]$. Let $D = \sup_{w \in [W_a, W_b]} |G_b(w)|$. Then,

$$\begin{aligned} |G_a(w) - G_b(w)| &\leq |G_a(w) - G_a(W_a)| + |G_a(W_b) - G_b(W_b)| + |G_b(w) - G_b(W_b)| && (91) \\ &\leq \alpha|W_a - w| + \beta|v_a - v_b| + (W_b - w)\frac{2}{\sigma^2}\left(v_b + \frac{W_b}{m \wedge m_e} + d_\theta D\right) && (92) \\ &\leq \alpha|W_b - W_a| + \beta|v_a - v_b| + (W_b - W_a)\frac{2}{\sigma^2}\left(v_b + \frac{W_b}{m \wedge m_e} + d_\theta D\right), && (93) \end{aligned}$$

which goes to 0 as $|v_a - v_b| + |W_a - W_b| \to 0$.

For $w \in [0, W_a]$, by Lemma 6,

$$\begin{aligned} |G_a'(w) - G_b'(w)| &\leq \frac{2}{\sigma^2}|v_a - v_b| + \frac{2}{\sigma^2}\Bigg| \min_{k_a \in [0, w/m_e]} (k_b(1 - m_e/m) - \theta(k_a)G_a(w)) \\ &\qquad - \min_{k_b \in [0, w/m_e]} (k_b(1 - m_e/m) - \theta(k_b)G_b(w))\Bigg| \\ &\leq \frac{2}{\sigma^2}|v_a - v_b| + \frac{2d_\theta}{\sigma^2}|G_a(w) - G_b(w)|. && (94) \end{aligned}$$

Applying Gronwall's inequality, for all $w \in [0, W_a]$

$$|G_a(w) - G_b(w)| \leq |G_a(W_a) - G_b(W_a)|e^{\frac{2d_\theta}{\sigma^2}(W_a - w)} + \frac{|v_a - v_b|}{d_\theta}\left(e^{\frac{2d_\theta}{\sigma^2}(W_a - w)} - 1\right),$$

which, together with (93), implies that for all $w \in [0, W_a]$

$$\begin{aligned} |G_a(w) - G_b(w)| \leq |v_a - v_b| &\left(|\beta|e^{\frac{2d_\theta}{\sigma^2}(W_a - w)} + \frac{e^{\frac{2d_\theta}{\sigma^2}(W_a - w)} - 1}{d_\theta}\right) \\ &+ |W_b - W_a|\left(\alpha + \frac{2}{\sigma^2}\left(v_b + \frac{W_b}{m \wedge m_e} + d_\theta D\right)\right)e^{\frac{2d_\theta}{\sigma^2}(W_a - w)}, \end{aligned}$$

which goes to 0 as $|v_a - v_b| + |W_a - W_b| \to 0$. ■

Lemma 5 *Consider $G_{v,W}$ defined in Lemma 4 for a given $W \geq \hat{k}m_e$ and $\beta \leq 0$. Then $G_{v,W}(w)$ is strictly monotonic and Lipschitz continuous in v for all w.*

Proof: Fix $W \geq \hat{k}m_e$, and consider $v_a > v_b$. Let G_a and G_b denote the solutions of the ODE defined in Lemma 4 for v_a and v_b, respectively. We will show that $G_a(w) < G_b(w)$ for all $w \geq 0$. We rely on the following two facts:

1. Terminal condition:
$$G_b(w) - G_a(w) = -\beta(v_a - v_b) \quad w \geq W$$

2. Bounds on $G'_b(w) - G'_a(w)$ for $w \in [0, W]$:

$$G'_b(w) - G'_a(w) = -\frac{2}{\sigma^2}\left[(v_a - v_b) - \min_{k\in[0,w/m_e]}(\Delta_k(w) - \theta(k)G_a(w)) - \min_{k\in[0,w/m_e]}(\Delta_k(w) - \theta(k)G_b(w))\right]$$

where recall that $\Delta_k(w) = \frac{w}{m} + k\left(1 - \frac{m_e}{m}\right)$. Under the assumption $G_a(w) \leq G_b(w)$, from Lemma 6:

$$-\frac{2}{\sigma}\left[(v_a - v_b) + d_\theta(G_b(w) - G_a(w))\right] \leq G'_b(w) - G'_a(w) \leq -\frac{2}{\sigma^2}\left[(v_a - v_b) - d_\theta\left(G_b(w) - G_a(w)\right)\right] \tag{95}$$

Combining these two facts, we get for any $w \in [0, W]$

$$(v_a - v_b)\left[-\beta + \frac{1}{d_\theta}\left(1 - e^{-\frac{2d_\theta W}{\sigma^2}}\right)\right] \leq G_b(w) - G_a(w) \leq (v_a - v_b)\left[-\beta + \frac{1}{d_\theta}\left(e^{\frac{2d_\theta W}{\sigma^2}} - 1\right)\right] \tag{96}$$

■

Lemma 6 *Let $x_1 = \arg\min_{x\in[u,v]} f_1(x)$ and $x_2 = \arg\min_{x\in[u,v]} f_2(x)$. Then,*

$$|f_1(x_1) - f_2(x_2)| \leq \sup_{x\in[u,v]} |f_1(x) - f_2(x)|$$

Proof: Proof Since $f_1(x_1) \leq f_1(x_2)$ and $f_2(x_2) \leq f_2(x_1)$,

$$f_1(x_1) - f_2(x_1) \leq f_1(x_1) - f_2(x_2) \leq f_1(x_2) - f_2(x_2)$$

and therefore, $|f_1(x_1) - f_2(x_2)| \leq \sup_{x\in[u,v]} |f_1(x) - f_2(x)|$. ■

D.2 Proofs of Results in Section 4

Proof of Proposition 3: We should point out that the monotonicity of the value function is not immediate because under the optimal policy $k^*(\cdot)$, the state-dependent cost function $\Delta_{k^*}(w)$ need not be monotonic in w. If it were, a simple sample path coupling argument could be used to deduce the monotonicity of the discounted value function by considering initial workloads $w_1 \leq w_2$.

Let $k^*_\gamma(\cdot)$ be the optimal policy minimizing expected discounted cost, and $V_\gamma(w)$ be the corresponding value function. Consider $w_1 \leq w_2$. We will create an alternate control policy π_1 when the initial workload is w_1, and denote the corresponding expected discounted cost by $\tilde{v}_1$. We will then show that $\tilde{v}_1 \leq V_\gamma(w_2)$ (in fact, our construction involves stochastic coupling and implies that the discounted reward starting with w_1 and using π_1 is stochastically smaller than the discounted reward starting with w_2 and using $k^*_\gamma(\cdot)$).

Construction of π_1: We simulate two independent systems in parallel: system 1 with initial workload $W_1(0) = w_1$ under control policy π_1 (which we will describe shortly); and system 2 with initial workload $W_2(0) = w_2$ under the optimal control policy $k^*_\gamma(w)$. The control at time t under π_1 is chosen to be

$$k_{\pi_1}(t) = \arg\min_{k\in[0,W_1(t)/m_e]} \frac{W_1(t)}{m} + k(1 - m_e/m)$$

for $t \in [0, \tau]$, where $\tau \doteq \min\{s \geq 0 : W_1(s) = W_2(s)\}$ is the coupling time of the two systems. That is, τ is the first time the workloads of the two coupled processes W_1 and W_2 coincide. For $t \geq \tau$, $k_{\pi_1}(t) = k^*_\gamma(W_1(t))$.

It is easy to see that since W_1 and W_2 have continuous sample paths, $W_1(t) \leq W_2(t)$ for $t \leq \tau$. Due to the choice of k_{π_1}, this further implies that

$$\begin{aligned}
\min_{k \in [0, W_1(t)/m_e]} \left(\frac{W_1(t)}{m} + k\left(1 - \frac{m_e}{m}\right) \right) &= \min\left\{ \frac{W_1(t)}{m}, \frac{W_1(t)}{m_e} \right\} \\
&\leq \min\left\{ \frac{W_2(t)}{m}, \frac{W_2(t)}{m_e} \right\} \\
&\leq \frac{W_2(t)}{m} + k^*_\gamma(W_2(t))\left(1 - \frac{m_e}{m}\right).
\end{aligned}$$

For $t \geq \tau$, $W_1(t)$ is stochastically equal to $W_2(t)$. Therefore, the discounted cost of π_1 (with initial workload w_1) is stochastically smaller than the discounted cost of k^*_γ (with initial workload w_2). This implies $\tilde{v}_1 \leq V_\gamma(w_2)$, but $V_\gamma(w_1) \leq \tilde{v}_1$ (since $V_\gamma(w_1)$ is the optimal expected discounted cost). Therefore, $V_\gamma(w_1) \leq V_\gamma(w_2)$ when $w_1 \leq w_2$.

Since $(V_\gamma(w_2) - V_\gamma(w_1)) \geq 0$ for all γ, this also holds as $\gamma \downarrow 0$.

Note : The only facts we relied on to argue monotonicity were (i) continuity of sample paths, and (ii) the cost of the cheapest action available in each state is monotonic in w. These appear to be weaker than the conditions typically used in the literature where the set of available actions is assumed to be independent of the state. Further, the cost is assumed to be non-decreasing in the state variable for each action. ■

We now provide the proofs of Proposition 4 and 5 for the analysis of our algorithms.

Proof of Proposition 4: It is easy to see that if the average cost of the optimal diffusion control formulation for the Sd-LPS system with a fluid continuation workload W is v and the value function gradient is $G_v(0)$, then (43) defines the optimal control, and (44) defines the ODE for G_v with initial condition $G_v(0) = 0$ (since the process reflects at $w = 0$, see [33]). We now show that for any v satisfyin $v^* < v \leq v_f(0)$, it is the average cost of some optimal fluid continuation policy.

We first observe that for each $0 \leq W < \infty$, there is a unique v such that $v = v_f(W)$. Further, $v_f(W)$ is continuous in W. To see this consider an arbitrary pair W, v and solve the following ODE

$$\begin{cases} v = \left\{\frac{w}{m} + k\left(1 - \frac{m_e}{m}\right) - \theta(k_f(w))G_{v,W}(w)\right\} + \frac{\sigma^2}{2}G'_{v,W}(w) & w \in [W, \max\{W, \hat{k}m_e\}] \\ v = \min_{k\in[0,w/m_e]}\left\{\frac{w}{m} + k\left(1 - \frac{m_e}{m}\right) - \theta(k)G_{v,W}(w)\right\} + \frac{\sigma^2}{2}G'_{v,W}(w) & w \leq \min\{W, \hat{k}m_e\} \end{cases} \tag{97}$$

backwards with terminal condition

$$G_{v,W}(\max\{W, \hat{k}m_e\}) = \left(\hat{k}(1 - m_e/m) + \frac{\sigma^2}{2m\hat{\theta}}\right)\frac{1}{\hat{\theta}} + \frac{\max\{W, \hat{k}m_e\}}{m\hat{\theta}}.$$

Note that this is the same ODE as (44) but we may not have $G_{v,W}(0) = 0$). Lemma 5 then shows that $G_{v,W}(0)$ is strictly monotonic and continuous in v. Therefore, for each W, there exists a unique $v^*(W)$ such that $G_{v^*(W),W}(0) = 0$ for the ODE and terminal conditions above. Further, Lipschitz continuity and Lemma 4 imply that the map $v^*(W)$ is continuous. From the foregoing discussion, we see that $v^*(W)$ denotes the cost of the optimal finite buffer policy with finite buffer W, which we call $v_f(W)$.

Next, it is easy to see that for $W_1 \leq W_2$, $v_f(W_1) \geq v_f(W_2)$. Combined with $v^* = v_f(\infty)$, this gives us that for each $v^* < v \leq v_f(0)$, there is a unique W such that $v = v_f(W)$.

we will next argue that $\overline{W}(v) = O\left(\log \frac{1}{v^* - v}\right)$ (and hence also finite). We will instead prove the following equivalent result: let v^*_W denote the average cost of the optimal fluid continuation policy with fluid continuation point W. Then $(v^* - v^*_W) = O(e^{-\beta W})$ as $W \to \infty$ for some constant $\beta > 0$.

Intuitively, the service rate of the optimal control must asymptotically approach $\hat{\theta}$ as the backlog builds up, and hence the distribution of the workload (and therefore number of jobs in the system) should decay at an exponential rate. Therefore, the loss from truncating the optimal control at workload W and using the fluid continuation policy should also be $O(e^{-\beta W})$ for some constant $\beta > 0$.

The proof will proceed in a few steps:
Step 1: Define

$$T(W) \doteq \int_0^W \theta(k^*(w))dw,$$

where $k^*(\cdot)$ is the optimal control. Then as $W \to \infty$, $T(W) = \Theta(W)$. That is, the integral of the drift over the interval $[0, W]$ for the optimal control $k^*(\cdot)$ must asymptotically grow linearly in W.

Proof: We begin by rewriting the HJB equation for $G^*(w)$

$$\begin{aligned}\frac{\sigma^2}{2}(G^*)'(w) &= v^* - \min_{k \in [0, w/m_e]} \left(\Delta_k(w) - \theta(k)G^*(w)\right) \\ &\leq v^* - \frac{w}{m \vee m_e} + \theta(k^*(w))G^*(w).\end{aligned}$$

Take the integration

$$\int_{w=0}^W \frac{\sigma^2}{2}(G^*)'(w)dw \leq W \cdot v^* - \frac{W^2}{2(m \vee m_e)} + \left(\alpha + \frac{W}{m\hat{\theta}}\right)\int_0^W \theta(k^*(w))dw.$$

This implies

$$\frac{\sigma^2}{2}\left(G^*(W) - G^*(0)\right) \leq W \cdot v^* - \frac{W^2}{2(m \vee m_e)} + \left(\alpha + \frac{W}{m\hat{\theta}}\right)T(W).$$

The left-hand side of the above inequality is at least 0 since $G^*(W) \geq 0$ and $G^*(0) = 0$. The first term on the right-hand side grows linearly in W. The second term grows as $\Theta(W^2)$. If $T(W) = o(W)$ then the right-hand side becomes negative for W large enough – a contradiction. Therefore, $T(W)$ must grow at least linearly. By our assumptions, $\theta(w) \leq \hat{\theta} < \infty$. Therefore, $T(W) = \Theta(W)$.
Step 2: Denote the distribution function of the workload under control k^* by F, the preceding step implies

$$\overline{F}(W) = O(e^{-\beta W})$$

for some positive constant $\beta > 0$. That is, the density of workload under the optimal control falls exponentially.

Proof: The density function is given by

$$\begin{aligned}f(W) &= \kappa e^{-\int_0^W \frac{\theta(k^*(w))}{\sigma^2/2}}dw \\ &= \kappa e^{-\frac{T(W)}{\sigma^2/2}},\end{aligned}$$

where κ is the normalization constant. Since $T(W) = \Theta(W)$ by the preceding step, $\overline{F}(W) = O(e^{-\beta W})$ for some $\beta > 0$.

Step 3: Consider the following control policy, parameterized by continuattion point W where we will assume $W \geq \hat{k}m_e$:

$$\tilde{k}_W(w) = \begin{cases} k^*(w) & w \leq W, \\ k_f(w) = \hat{k} & w > W. \end{cases}$$

That is, we create a fluid continuation control with prefix $k^*(w)$ for $w \leq W$. This results in a suboptimal control in the set $\mathcal{F}_W$. If we denote the average cost of this control as $\tilde{v}_W$, and the average cost of the optimal fluid continuation policy in $\mathcal{F}_W$ as $v_f(W)$, then

$$v^* \leq v_f(W) \leq \tilde{v}_W.$$

However, since the workload density decays exponentially under control $k^*(\cdot)$ and $\theta(\hat{k}) > 0$, $(\tilde{v}_W - v^*) = O(e^{-\beta W})$, and hence $(v_f(W) - v^*) = O(e^{-\beta W})$. ■

Proof of Proposition 5: Recall the Newton-Raphson algorithm from Algorithm 1: We first pick a large enough value of workload $W \geq \hat{k}m_e$ (which is not changed during subsequent iterations). The goal of the Newton-Raphson algorithm then is to find the average cost of the optimal dynamic policy under the restriction that the control for $w \geq W$ is the fluid control $\hat{k}$. With v_n as our guess in the nth iteration, we backwards evolve the ODEs:

$$v_n = \min_{k \in [0, w/m_e]} \left[\frac{w}{m} + k\left(1 - \frac{m_e}{m}\right) - \theta(k) G_{v_n}(w) \right] + \frac{\sigma^2}{2} G'_{v_n}(w) \tag{98}$$

$$1 = -\theta(k_{v_n}(w)) g_{v_n}(w) + \frac{\sigma^2}{2} g'_{v_n}(w) \tag{99}$$

for $w \in [0, W]$ with (terminal) boundary conditions:

$$G_{v_n}(W) = \left(\hat{k}\left(1 - \frac{m_e}{m}\right) - v_n + \frac{\sigma^2}{2\hat{\theta}} \right) \frac{1}{\hat{\theta}} + \frac{1}{m\hat{\theta}} W \tag{100}$$

$$g_{v_n}(W) = -\frac{1}{\hat{\theta}} \tag{101}$$

Here $k_{v_n}(w)$ denotes the policy obtained while solving the ODE for G_{v_n}.

The updated guess for the $(n+1)$st iteration is

$$v_{n+1} = v_n - \frac{G_{v_n}(0)}{g_{v_n}(0)}.$$

We develop our proof of the proposition in several steps.

Step 1 : $v_n \geq v_f(W)$ for $n \geq 1$

Proof: Let $\widetilde{G}_v(w)$ (parameterized by v, w) be given by the ODE

$$v = \frac{w}{m} + k_{v_n}(w)\left(1 - \frac{m_e}{m}\right) - \theta(k_{v_n}(w)) \widetilde{G}_v(w) + \frac{\sigma^2}{2} \widetilde{G}'_v(w)$$

for $w \in [0, W]$ with boundary condition

$$\widetilde{G}_v(W) = \left(\hat{k}\left(1 - \frac{m_e}{m}\right) - v + \frac{\sigma^2}{2\hat{\theta}} \right) \frac{1}{\hat{\theta}} + \frac{1}{m\hat{\theta}} W$$

This is essentially the same ODE as (98) but with the min operator replaced by the fixed policy k_{v_n}. The first observation is that $\widetilde{G}_v(0)$ is a linear function of v, and $\widetilde{G}_{v_n}(w) = G_{v_n}(w)$ for all $w \in [0, W]$. Further, denoting

$$\widetilde{g}_v(w) = \frac{d}{dv}\widetilde{G}_v(w)$$

it is easy to see that $\widetilde{g}_v(w) = g_{v_n}(w)$. Therefore,

$$v_{n+1} = v_n - \frac{G_{v_n}(0)}{g_{v_n}(0)} = v_n - \frac{\widetilde{G}_{v_n}(0)}{\widetilde{g}_{v_n}(0)}$$

Since $\widetilde{G}_v(w)$ is a linear function in v for all w, the Newton-Raphson update for $\widetilde{G}_v(0)$ directly yields that value of v for which $\widetilde{G}_v(0) = 0$. But this must be the average cost of policy k_{v_n}. Therefore, v_{n+1} is in fact the average cost of policy k_{v_n}. Since k_{v_n} is a feasible policy in the set $\mathcal{F}_W$, its average cost must be no less than $v_f(W)$ and hence all the iterates $\{v_1, v_2, \ldots\}$ produced are larger than $v_f(W)$.

Step 2: The iterates for average cost $\{v_1, v_2, \ldots\}$ form a strictly decreasing sequence.

Proof: For this, we will show that $G_v(0)$ is monotonically decreasing and Lipschitz continuous in v with derivative bounded away from 0. This would imply that for $v > v_f(W)$, $G_v(0) < 0$, as well as $g_v(0) < 0$, and hence $v_1 > v_2 > \ldots > v_f(W)$.

Consider $v_a > v_b$, and let G_a, g_a, k_a and G_b, g_b, k_b represent the solution of (98)-(101) and the optimal controls for v_a and v_b, respectively. Our goal is to show $G_a(w) < G_b(w)$ for all $w \geq 0$. We rely on the following two facts:

1. Terminal condition:

$$G_b(w) - G_a(w) = \frac{a-b}{\hat{\theta}} \quad w \geq W$$

2. Bounds on $G'_b(w) - G'_a(w)$ for $w \in [0, W]$: [1]

$$G'_b(w) - G'_a(w) = -\frac{2}{\sigma^2}\left[(v_a - v_b) - \min_{k\in[0,w/m_e]}\left(\Delta_k(w) - \theta(k)G_a(w)\right) - \min_{k\in[0,w/m_e]}\left(\Delta_k(w) - \theta(k)G_b(w)\right)\right]$$

where recall that $\Delta_k(w) = \frac{w}{m} + k\left(1 - \frac{m_e}{m}\right)$. By the assumption $G_a(w) \leq G_b(w)$ and Lemma 6,

$$-\frac{2}{\sigma}\left[(v_a - v_b) + d_\theta(G_b(w) - G_a(w))\right] \leq G'_b(w) - G'_a(w) \leq -\frac{2}{\sigma^2}\left[(v_a - v_b) - d_\theta\left(G_b(w) - G_a(w)\right)\right].$$

Combining these two facts, we get

$$(v_a - v_b)\left[\frac{1}{\hat{\theta}} + \frac{1}{d_\theta}\left(1 - e^{-\frac{2d_\theta W}{\sigma^2}}\right)\right] \leq G_b(0) - G_a(0) \leq (v_a - v_b)\left[\frac{1}{\hat{\theta}} + \frac{1}{d_\theta}\left(e^{\frac{2d_\theta W}{\sigma^2}} - 1\right)\right]. \quad (102)$$

Thus we have proved that $G_v(0)$ is monotonically decreasing in v and therefore the Newton-Raphson iterates will form a monotonically decreasing sequence. Further, $G_v(w)$ is Lipschitz continuous in v for all w, and therefore its derivative with respect to v exists almost everywhere, and according to the first inequality of (102) this derivative is bounded away from 0.

[1] We use primes to denote derivatives with respect to w. Derivatives with respect to v are denoted with $\frac{\partial}{\partial v}$ notation.

The properties proved so far are sufficient to prove that the Newton-Raphson algorithm converges to the optimal $v_f(W)$, and the convergence rate is at least linear. It is also true that the second derivative of $G_v(0)$ with respect to v is finite in some neighborhood of $v_f(W)$, and hence the Newton-Raphson iterates converge quadratically to $v_f(W)$ – but we omit this argument.

■