Asynchronous Load Balancing and Auto-scaling: Mean-Field Limit and Optimal Design

Jonatha Anselmi

Abstract—We develop a Markovian framework for load balancing that combines classical algorithms such as Power-of-d with auto-scaling mechanisms that allow the net service capacity to scale up or down in response to the current load on the same timescale as job dynamics. Our framework is inspired by serverless platforms, such as Knative, where servers are software functions that can be flexibly instantiated in milliseconds according to scaling rules defined by the users of the serverless platform. The main question is how to design such scaling rules to minimize user-perceived delay performance while ensuring low energy consumption. For the first time, we investigate this problem when the auto-scaling and load balancing processes operate *asynchronously* (or *proactively*), as in Knative. In contrast to the synchronous (or reactive) paradigm, asynchronism brings the advantage that jobs do not necessarily need to wait any time a scale-up decision is taken.

In our main result, we find a general condition on the structure of scaling rules able to drive mean-field dynamics to delay and relative energy optimality, i.e., a situation where both the user-perceived delay and the relative energy waste induced by idle servers vanish in the limit where the network demand grows to infinity in proportion to the nominal service capacity. The identified condition suggests to scale up the current net capacity if and only if the mean demand exceeds the rate at which servers become idle and active. Finally, we propose a family of scaling rules that satisfy our optimality condition. Numerical simulations demonstrate that these rules provide better delay performance than existing synchronous auto-scaling schemes while inducing almost the same power consumption.

Index Terms—Load balancing, auto-scaling, serverless computing, asymptotic optimality, Knative

1 Introduction

Load balancing is the process of distributing work units (jobs) over a set of distributed computational resources (servers) for processing. In large architectures, each server has its own queue, as this enhances scalability, and jobs are irrevocably dispatched to one out of N parallel servers instantaneously upon their arrival. Given the stringent latency requirements of modern applications, breaches of which can severely impact revenue, load balancing techniques are designed to optimize user-perceived delay performance and popular examples are Power-of-d [24] and Join-the-Idle-Queue (JIQ) [21].

Closely related to load balancing, *auto-scaling* is a term often used in cloud computing to refer to the process of adjusting the current service capacity automatically in response to the current load [28]. Auto-scaling mechanisms are meant to control the current net capacity over time to avoid performance degradation, which yields unacceptably large delays, and overprovisioning of resources, which yields high infrastructure and energy costs. Google Cloud Run, Amazon Elastic Compute Cloud (EC2), Microsoft Windows Azure and Oracle Cloud Platform are examples of platforms that offer auto-scaling and load balancing features. Users of these platforms deploy their applications with some control on how the system should scale up resources in front of an increased load. Modern auto-scaling mechanisms are extremely reactive in the sense that they control the current net capacity relying on fresh observations of the system state rather than historical data. This especially holds true in *serverless computing platforms*, or Function-as-a-Service, which nowadays provide the convenient solution to deploy any type of application or backend service [22].

In this paper, we are interested in the interplay between the load balancing and auto-scaling processes. The main objective is to design a scheme that combines both to minimize delay performance while ensuring low energy consumption.

1.1 Timescale Separation

Most of the existing performance models for load balancing assume that the available service capacity remains constant over time [33], i.e., auto-scaling is not taken into account. Nonetheless, auto-scaling mechanisms are widely employed by cloud applications and affect delay performance. This does not mean that classic load balancing models are inadequate for cloud systems but simply that they assume that auto-scaling operates at a much slower *timescale* than load balancing. Essentially, this means that jobs do not see any change in the available capacity because they evolve much faster than servers. This makes sense if servers are interpreted as physical or even virtual machines because setup times are of the order of minutes if not longer [16] while in typical applications hosted in cloud networks job service times are about ten milliseconds [22]. The large body of literature on load balancing, reviewed in Section 2, is undoubtedly the proof that this timescale separation assumption is well accepted for several systems. In the context of serverless computing however, a server is interpreted as a software function that can be flexibly instantiated in milliseconds [35], [35], i.e., within a time window that is comparable with the magnitude of job inter-arrival and service times, and with negligible switching costs. Here, auto-scaling mechanisms are extremely reactive and the decisions

- *J. Anselmi is with Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, 38000 Grenoble, France. E-mail: jonatha.anselmi@inria.fr*

of turning servers on or off are based on instantaneous observations of the current system state rather than on the long-run equilibrium behavior. Therefore, the timescale separation assumption above becomes questionable, as also discussed in [22], because it would mean to assume that job dynamics achieve stochastic equilibrium between consecutive changes of the net service capacity, i.e., in milliseconds.

1.2 Getting Rid of the Timescale Separation Assumption

While a large body of the literature investigates load balancing and auto-scaling *separately* [28], [33], little has been done when both are applied jointly within the same timescale. Existing works focus on *synchronous* (i.e., both scale-up and dispatching decisions are taken at the same time) or *centralized* (i.e., all servers share a common queue) architectures [16], [22]. For scalability reasons however, no central queue is maintained (in this case, we say that the architecture is *decentralized*) and no decisions are taken synchronously in massive cloud systems; see Section 1.3 below. A decentralized but synchronous architecture where JIQ is synchronized with an ad-hoc auto-scaling strategy is considered in [12], [17], [26]. In contrast, we consider a decentralized and *asynchronous* architecture, where the term "asynchronous' means that scaling and dispatching decisions are decoupled.

1.3 Synchronous vs Asynchronous in Serverless Computing

The load balancing and auto-scaling processes of existing implementations of public serverless computing platforms are either "synchronous" or "asynchronous"; this terminology is borrowed from the cloud computing community [22], though some works use the terms "reactive" and "proactive", respectively [13]. As explained in these references,

- The auto-scaling principle underlying a synchronous architecture is that *a new server is turned on at the arrival time of a job if the job itself finds all servers busy*. The drawback of this approach is that all jobs that have triggered a scale-up signal are forced to wait before being processed. In centralized implementations, each of these jobs waits for the activation of the server that has been launched at the moment of its arrival (coldstart latency) [22], [29], [35], while in the decentralized proposals given in [12], [17], [26], each of these is sent to an already active (busy) server chosen at random, hence slowed down by the jobs ahead.
  To the best of our knowledge, no synchronous-decentralized implementations currently exist. In contrast, AWS Lambda, Azure Functions, IBM Cloud Functions and Apache OpenWhisk are examples of synchronous-centralized platforms.
- The auto-scaling principle underlying an asynchronous architecture is that *the load balancing and auto-scaling processes are decoupled*. Specifically, a job is dispatched to some running server immediately upon its arrival according to some load balancing algorithm and, independently of this, an auto-scaling mechanism decides whether the current processing capacity should change as a function of user-defined metrics that may depend on instantaneous observations of the current system state [22]. Because of this decoupling, scale-up decisions do not need to wait that all active servers are busy as in the synchronous approach. Thus, they may anticipate the arrival of a job and overcome the intrinsic drawback of the synchronous approach described above. In addition, the scale-up decision rate is fine-tuned by the platform user; in Knative, this is set via the `max-scale-up-rate` global key.
  To the best of our knowledge, no asynchronous-centralized implementations currently exist. In contrast, Google Cloud Run and Knative are examples of asynchronous-decentralized platforms [1].

In a stochastic and dynamic setting, no performance model/analysis is available in the literature for the asynchronous-decentralized approach. Our main motivation is to contribute to fill this gap.

1.4 Summary of our Contributions

We develop a Markovian framework for load balancing that includes asynchronous auto-scaling mechanisms. We refer to this framework as 'Asynchronous Load Balancing and Auto-scaling' (ALBA). Two (asynchronous) mechanisms drive dynamics in ALBA:

i) a *dispatching rule*, or load balancing rule, which defines how jobs are dispatched among the set of active servers as they join the system, and
ii) a *scaling rule*, which defines how the number of active servers scales up and down over time, possibly as a function of the current system state.

The dispatching rules included in ALBA are Join-Below-Threshold-*d* (JBT-*d*), which is a generalization of JIQ, and Power-of-*d*; in fact, these are the rules used in Knative [2]. We also assume that a server is turned off only if it remains idle during an expiration window. This scale-down rule is commonly used in practice [1], [22] and also known as "delay-off" [16]. In contrast, we do not impose any particular structure on scale-up rules because they are usually defined by the user of the serverless platform. Having fixed the scale-down rule, in the following the term "scaling rule" refers to a scale-up rule.

Our key technical contribution is a general condition on the structure of scaling rules that is able to drive the mean-field dynamics induced by ALBA to *delay and relative energy optimality*, a situation where the user-perceived delay and the relative energy wastage induced by idle servers vanish. This condition suggests to scale up capacity if and only if the mean demand exceeds the overall rate at which servers become idle and active, which can be measured.

We also propose *Rate-Idle*, see Definition 2, a scaling rule that satisfies our optimality condition. Provided that it is combined with JIQ, we show by means of numerical simulations that Rate-Idle provides a better delay performance than the synchronous schemes in [17], [26] while inducing the same energy consumption cost. We own this gain to the fact that scale up decisions may be taken before job arrivals, while in a synchronous scheme such as TABS-d, jobs are forced to wait any time a scale up decision is taken.

Our results are obtained through a rigorous analysis of the underlying Markov process in the mean-field limit. Here, we establish the convergence of the stochastic finite model to a fluid model with a discontinuous drift. Then, we leverage the fluid model to identify a condition that drives the fluid trajectories to a unique fixed point corresponding to delay and relative energy optimality.

1.5 Organization and Main Results Detailed

Section 2 reviews the existing literature and Section 3 introduces ALBA by defining a stochastic (intractable) and a deterministic (tractable) model to describe its dynamics. Section 4 presents our main results, i.e., Theorems 1, 2 and 3:

- Theorem 1 connects the stochastic and the deterministic models and justifies the use of the latter to approximate the dynamics of the former. This enables analytical tractability and allows one to study dynamics easily. We prove Theorem 1 following the framework developed in [9], [31], though we develop ad-hoc arguments to handle the discontinuities of the drift function of the underlying Markov chain.
- Theorem 2 characterizes the *fixed points* of the deterministic model in terms of a set of non-linear equations. It also provides a simple necessary and sufficient condition able to tell whether or not the nominal service capacity will be needed to handle the incoming demand. Within Power-of-d, roughly speaking, there always exists a unique fixed point if the scaling rule is "nice". Within JBT-d however, uniqueness is guaranteed only if the scaling rule has access to the number of servers containing exactly one job (see Remark 1).
- Theorem 3 investigates how to design *optimal* and globally stable scaling rules. More specifically, we identify a general condition ensuring that dynamics of the deterministic model converge to delay and relative energy optimality. We show that optimality can only be achieved within JIQ (or equivalently JBT-0), though in practice this may not be the convenient choice within architectures with several dispatchers. In this case, an exact implementation of JIQ would imply an expensive communication overhead per job and Power-of-d may be the way to go as it does not require the dispatcher(s) to store information about the server states.

Section 5 compares by simulation the asynchronous and synchronous approaches, showing that the former provides a much better delay performance. Then, Section 6 develops a tractable optimization framework to illustrate how the results presented in this paper can be applied to trade off between performance and energy consumption. Finally, Section 7 draws the conclusions. Proofs of our results are deferred to the appendix.

2 LITERATURE REVIEW

The existing literature related to load balancing and auto-scaling is huge and our goal is to provide the necessary background highlighting the difference of our work.

2.1 Load Balancing and the Zero Delay Property

Popular examples of load balancing algorithms that work well when servers are homogeneous, i.e., all servers have the same processing speed, are Random, Round-Robin (RR) [5], [20], Power-of-d [24], Join-the-Idle-Queue (JIQ) [21], Least-Left-Workload (LLW) and Size Interval Task Allocation (SITA) [7], [18], [19]. Random sends each job to random server, RR sends jobs to servers in a cyclic manner, Power-of-d sends an incoming job to the least loaded server among d selected uniformly at random. JIQ sends an incoming job to a random idle server if an idle server exists and to a random one otherwise, LLW sends an incoming jobs to the queue having the shortest workload, and SITA sends a job to a given server if its size belong to a given interval. In general, it is not possible to identify which of these algorithms is the best because the general answer depends on the underlying architecture, load conditions, service time distribution and on the amount of information available to the dispatcher [33].

Recently, a number of works attempted to understand under which conditions the mean waiting time can be driven down to zero in the limiting regime where the arrival rate grows linearly with the number of servers while keeping the average load below one. This is possible within different load balancing schemes and architectures. Examples include JIQ [30], Power-of-d with $d \to \infty$ as the network size grows to infinity [25], Power-of-d with memory [8], SITA combined with RR [6] and the pull-based policies developed in [14], [32]. To some extent, the fundamental limits of load balancing are described in [14], where the authors investigate trade-offs between performance (the zero-delay property), communication overhead and memory within a certain class of symmetric architectures and the large-system limiting regime.

2.2 Joint Load Balancing and Auto-scaling

The load balancing algorithms above have been analyzed under the assumption that the active number of servers is *constant* at all times. Few works considered a time-varying net capacity [12], [17], [26], [27]. In these references, JIQ is synchronized with a specific auto-scaling strategy as described in Section 1.3. When the traffic demand and the nominal service capacity proportionally grow to infinity, the mechanism proposed in [26] yields the zero-delay property but also deactivates any surplus idle servers, thus inducing delay and relative energy optimality. This property has been strengthened in [27], where the authors relax some finite buffer assumptions. In contrast, our work shows optimality:

- within an asynchronous (see Section 1.3) architecture; an advantage of asynchronism is that jobs do not necessarily need to wait any time a scale-up decision is taken, a fact whose performance gain is evaluated in Section 5 by simulation;
- without limiting on an ad-hoc auto-scaling strategy; rather, we identify a structural property on scaling rules that induces optimality under broader conditions (Theorem 3).

3 Asynchronous Load Balancing and Auto-scaling (ALBA)

In this section, we first describe the main principles at the basis of Asynchronous Load Balancing and Auto-scaling (ALBA). Since our aim is to develop a model tailored to serverless computing, we will make several references to Knative, a popular serverless framework for hosting Function-as-a-Service processing that is used, among others, by Google Cloud Run. Then, we propose two performance models for ALBA. The first is meant to capture the stochastic nature of the underlying dynamics while the second is deterministic and will serve to approximate the dynamics induced by the first. The advantage of the deterministic model is its tractability. Finally, we formalize the structure of the scaling rules investigated in this paper.

3.1 System Description

The proposed framework, ALBA, is composed of a system of N parallel servers, each with its own queue, that represent the nominal service capacity, i.e., the upper limit on the amount of resources that one user can have up and running at the same time[1]. In the cloud computing community, servers are also referred to as containers, cloud functions, instances or replicas. Public serverless computing platforms usually require to specify such limit in order to ensure service availability for other users. In the following, the terms servers and queues will be used interchangeably. A server is said *warm* if turned on, *cold* if turned off and *initializing* if making the transition from cold to warm. These are the possible server states [22], [23], [35]. An initializing server performs basic startup operations such as connecting to database, loading libraries, etc. This is the time to provision a new function instance. Only warm servers are allowed to receive jobs. A server is also said *idle-on* if warm but not processing any job, and *busy* if warm and processing some job. Typically, billing policies charge per number of warm and initializing servers used per time unit.

Jobs join the system from an exogenous source to receive service. Upon arrival, each job is dispatched to a warm server according to some dispatching rule. After dispatching, each job is processed by the selected server according to the presumed scheduling discipline at that server. After processing, each job leaves the system.

Assumption 1. *Jobs are dispatched to servers according to either Power-of-d or Join-Below-Threshold-d (JBT-d).*

We recall that Power-of-d sends an incoming job to the shortest among $d \geq 1$ warm servers selected at random at the moment of its arrival and JBT-d sends an incoming job to a warm server containing no more than $d \geq 0$ jobs if one exists otherwise to a warm server selected at random. In all cases, ties are broken randomly. If $d = 0$, JBT-d is also known as Join-the-Idle-Queue (JIQ) [21]. We limit our framework to these types of schemes because they involve a constant communication overhead per job (in architectures with a single dispatcher) and because they are commonly used in practice. For instance, Knative uses Power-of-2 if no limit is set on the queue length of each server and JBT-d if such limit is set to d [2].

Alongside with the above job dynamics, the pools of warm/initializing/cold servers change over time in the background and in an asynchronous manner. Precisely, the platform monitors the system state at some epochs that we refer to as *scaling times*. At such times, a cold server is selected, provided that one exists, and becomes initializing according to the outcome of some scaling rule. After some *initialization time*, or coldstart latency, an initializing server becomes idle-on. When a server becomes idle-on, it becomes cold after a scale down delay, or *expiration time*, if during such time the server received no job; this scale-down rule is used in several serverless computing platforms (including Knative) [34], [35] and also in other settings [16]. We observe that the number of warm servers fluctuates from 0 to N over time. While in practice it may be possible to set a lower limit on the number of warm servers, the scale down to zero (or one) servers configuration is usually the default choice [3].

To a great extent, the scale up rule, the expiration rate and the scaling times are under the control of the platform user, which may design them in a way to optimize a trade-off between performance and energy. On the other hand, several measurements indicate that initialization times are typically one order of magnitude higher than jobs' service times in serverless platforms [22], [35].

3.2 Notation

We introduce some notation that will be used throughout the paper. Let $B \in \mathbb{Z}_+ \cup \{+\infty\}$ be a constant that will denote the buffer size of each server. We use $\mathbb{I}_{\{A\}}$ to denote the indicator function of A. If $a \in \mathbb{R}$ and A denotes an interval, $\mathbf{1}_A^a := \mathbb{I}_{\{a \in A\}}$. We also let $(\cdot)^+ := \max\{\cdot, 0\}$ and $\|\cdot\|$ denotes the L_1 norm. Unless specified otherwise, (i,j) ranges over the set $\{0,\ldots,B\} \times \{0,1,2\}$ if $B < \infty$ and over $\mathbb{Z}_+ \times \{0,1,2\}$ otherwise. The process of interest will take values in $\mathcal{S} := \{(x_{i,j} \in \mathbb{R}_+, \forall (i,j)) : \sum_{i,j} x_{i,j} = 1\}$ and our analysis holds under the distance function d_w induced by the weighted ℓ_2 norm $\|\cdot\|_w$ on $\mathbb{R}^{\mathbb{Z}_+}$ defined by $\|x - x'\|_w^2 := \sum_{i,j} \frac{|x_{i,j} - x'_{i,j}|^2}{2^{i+j}}$. For $x \in \mathcal{S}$, let $y_i := \sum_{k \geq i} x_{i,2}$. We also let $\mathcal{S}_1 := \{x \in \mathcal{S} : \sum_{i \geq 1} i x_{i,2} < \infty\}$.

3.3 Markov Model

We model the dynamics induced by ALBA in terms of a continuous time Markov chain. The exogenous arrival process of jobs is assumed to be Poisson with rate λN, with $0 < \lambda < 1$. Our analysis (Theorem 1) generalizes trivially to a time-varying arrival rate, a case that we omit for clarity of exposition. We discuss this point in the Conclusions. The processing times, or service times, of jobs are independent and exponentially distributed random variables with unit mean. Servers process jobs according to any work-conserving discipline. Upon arrival, each job is assigned to one warm server as specified in Assumption 1. In the extreme case where no warm server exists, the job is lost. We assume that each server can contain at most $B > d$ jobs and a job that is sent to a server with B jobs is rejected. If not specified otherwise, B is either finite or infinite. At each scaling time, a cold server is selected uniformly at random,

[1] In Knative, this upper limit is specified by the `max-scale-limit` global key.

provided that one exists, and becomes initializing with some probability g. This is the *scaling probability* (or rule) and will possibly depend on the system state; in the conclusion section, we will discuss how our work adapts to the case where a random number of cold servers is selected at each scaling time. Given that jobs arrive with a rate proportional to N and only one server can be added at each scaling time, we let the scaling frequency increase with N as well. As it occurs in Knative, this implies that the number of servers created in a time window of constant size is proportional to N if within such window the scaling probability is not zero. We let the inter-scaling, initialization and expiration times be independent and exponentially distributed with rate αN, β and γ, respectively.

Let $\tilde{Q}^N(t) := (\tilde{Q}^N_1(t), \dots, \tilde{Q}^N_N(t))$ be the vector of queue lengths at time t, including the jobs in service, and let $\tilde{S}^N(t) := (\tilde{S}^N_1(t), \dots, \tilde{S}^N_N(t))$ be the vector of server states. Specifically, $\tilde{S}^N_k(t) \in \{0,1,2\}$ indicates whether server k is cold ($\tilde{S}^N_k(t) = 0$), initializing ($\tilde{S}^N_k(t) = 1$) or warm ($\tilde{S}^N_k(t) = 2$) at time t. Under the above assumptions, the stochastic process $(\tilde{Q}^N(t), \tilde{S}^N(t))$ is a continuous-time Markov chain on state space $\{(n,s) \in \{0,\dots,B\}^N \times \{0,1,2\}^N : n_k > 0 \Rightarrow s_k = 2, \forall k = 1,\dots,N\}$.

It is convenient to describe dynamics in terms of the process $X^N(t) := (X^N_{0,0}(t), X^N_{0,1}(t), X^N_{i,2}(t) : i = 0,\dots,B)$ where

$$X^N_{i,j}(t) := \frac{1}{N} \sum_{k=1}^{N} \mathbb{I}_{\{\tilde{Q}^N_k(t)=i, \tilde{S}^N_k(t)=j\}} \tag{1}$$

is the proportion of servers in state j with i jobs at time t. The process $X^N(t)$ is still a Markov chain with values in some set $\mathcal{S}^{(N)}$ that is a subset of $\mathcal{S}$. Let $e_{i,j} := (\delta_{i,i'}\,\delta_{j,j'} \in \{0,1\} : i' \geq 0, j' = 0,1,2)$ where $\delta_{a,b}$ denotes the Kronecker delta and let $x := (x_{i,j}) \in \mathcal{S}^{(N)}$ denote a generic state of $X^N(t)$. For conciseness, the Markov chain $X^N(t)$ has the following transitions:

$$\begin{aligned}
&x \mapsto x' := x + \tfrac{1}{N}(e_{i,2} - e_{i-1,2}) && \text{with rate} \quad \lambda N f_{i-1}(x)\\
&x \mapsto x' := x + \tfrac{1}{N}(e_{i-1,2} - e_{i,2}) && \text{with rate} \quad x_i N\\
&x \mapsto x' := x + \tfrac{1}{N}(-e_{0,0}, e_{0,1}) && \text{with rate} \quad \alpha N g\\
&x \mapsto x' := x + \tfrac{1}{N}(-e_{0,1}, e_{0,2}) && \text{with rate} \quad \beta x_{0,1} N\\
&x \mapsto x' := x + \tfrac{1}{N}(e_{0,0} - e_{0,2}) && \text{with rate} \quad \gamma x_{0,2} N
\end{aligned}$$

for all $i = 1,\dots,B$, provided that $x, x' \in \mathcal{S}^{(N)}$. Here, $g := g(x) : \mathcal{S} \to [0,1]$ is the scaling probability, and $f_i(x)$, which depends on the dispatching rule, represents the probability of assigning an incoming job to a warm server containing exactly i jobs. If $y_0 > 0$, within Power-of-d we have (assuming that server selections are with replacement)

$$f_i(x) = \frac{y_i^d - y_{i+1}^d}{y_0^d}, \tag{2}$$

where $y_i := y_i(x) := \sum_{j \geq i} x_{j,2}$, and within JBT-$d$ we have

$$f_i(x) = \frac{x_{i,2}\,\mathbb{I}_{\{\sum_{k=0}^d x_{k,2}=0\}}}{y_0} + \frac{x_{i,2}\,\mathbb{I}_{\{\sum_{k=0}^d x_{k,2}>0\}}}{\sum_{k=0}^d x_{k,2}} \mathbb{I}_{\{i \leq d\}}, \tag{3}$$

where we have taken the convention that $0/0 = 0$, for all $i = 0,\dots,B-1$. If $y_0 = 0$, then $f_i(x) = 0$ as no warm server exists.

3.4 Deterministic Model

We introduce the deterministic (or fluid, mean-field) model for the dynamics of ALBA.

Definition 1. *A continuous function $x(t) : \mathbb{R}_+ \to \mathcal{S}$ is said to be a* fluid model *(or fluid solution) if for almost all $t \in [0,\infty)$*

$$\dot{x}_{0,0} = \gamma x_{0,2} - \alpha g \mathbb{I}_{\{x_{0,0}>0\}} - \gamma x_{0,2}\,\mathbb{I}_{\{x_{0,0}=0,\, \gamma x_{0,2} \leq \alpha g\}} \tag{4a}$$

$$\dot{x}_{0,1} = \alpha g \mathbb{I}_{\{x_{0,0}>0\}} - \beta x_{0,1} + \gamma x_{0,2}\,\mathbb{I}_{\{x_{0,0}=0,\, \gamma x_{0,2} \leq \alpha g\}} \tag{4b}$$

$$\dot{x}_{0,2} = x_{1,2} - h_0(x) + \beta x_{0,1} - \gamma x_{0,2} \tag{4c}$$

$$\dot{x}_{i,2} = x_{i+1,2}\mathbb{I}_{\{i<B\}} - x_{i,2} + h_{i-1}(x) - h_i(x)\mathbb{I}_{\{i<B\}}, \tag{4d}$$

$i = 1,\dots,B$, where $g := g(x) : \mathcal{S} \to [0,1]$, and $h_i(x) =$ $\min\{\beta x_{0,1}, \lambda\}$ *if $y_0 > 0$ and otherwise ($y_0 = 0$):*

$$h_i(x) = \lambda\,\frac{y_i^d - y_{i+1}^d}{y_0^d} \tag{5}$$

if Power-of-d is applied and

$$h_i(x) = \begin{cases} \lambda\,\frac{x_{i,2}}{\sum_{k=0}^d x_{k,2}} \mathbb{I}_{\{i \leq d\}}, \quad \text{if } \sum_{k=0}^d x_{k,2} > 0 \\[2ex] \big(\beta x_{0,1} + x_{d+1,2}\mathbb{I}_{\{i=d\}}\big)\,\mathbb{I}_{\{x_{d+1,2}+(d+1)\beta x_{0,1} \leq \lambda\}}, \\ \qquad\qquad \text{if } \sum_{k=0}^d x_{k,2} = 0,\ i \leq d, \\[2ex] \frac{x_{i,2}}{y_0}(\lambda - x_{d+1,2} - (d+1)\beta x_{0,1})^+, \\ \qquad\qquad \text{if } \sum_{k=0}^d x_{k,2} = 0,\ i > d, \end{cases} \tag{6}$$

if JBT-d is applied.

As for $X^N_{i,j}(t)$, $x_{i,j}(t)$ is interpreted as the proportion of servers in state j with i jobs at time t.

Let us provide some intuition about the fluid model. First, when a strictly positive fluid mass of warm server exists, i.e., $y_0 > 0$, the functions h_i are interpreted as the rate at which jobs are assigned to servers with exactly i jobs. When the amount of fluid of cold servers is strictly positive, i.e., $x_{0,0} > 0$, to some extent these equations may be interpreted as the conditional expected change, or *drift*, from state x of the Markov chain $X^N(t)$. In contrast, when $x_{0,0} = 0$, there exists a term, $-\mathbb{I}_{\{x_{0,0}=0,\, \gamma x_{0,2} \leq \alpha g\}} \gamma x_{0,2}$ (see (4a) and (4b)), that still drains the amount of cold servers down. This is due to warm servers that become cold but immediately turn initializing and it appears if the scaling rule is 'greedy enough', i.e., if the rate at which new initializing servers can be created is greater than or equal to the rate at which warm servers go cold. This term is due to fluctuations of order $1/N$ that appear when $X^N_{0,0}(t) = 0$, which bring discontinuities in the drift of $X^N(t)$, and will come out from the stochastic analysis developed in Appendix 1.1.3.

Now, let us focus on (5) and (6), and let us assume that $y_0 > 0$. In the case of Power-of-d, $h_i = \lambda f_i$ and $x(t)$ evolves following the natural dynamics of Power-of-d as in [24], though normalized on the variable mass of warm servers $y_0(t)$. The case of JBT-d is more delicate because of the discontinuous structure of f_i in (3). If a strictly positive fraction of warm servers with no more than d jobs exist, then $h_i = \lambda f_i$ and $x(t)$ evolves following the natural dynamics of JBT-d, though again normalized on a variable number of servers. On the other hand, when $\sum_{k=0}^d x_{k,2} = 0$, there is a flow of warm servers with at most d jobs that are created but immediately used for dispatching jobs. Specifically, there are two factors that come into play here: the first is due to initializing servers that get warm with exactly i jobs (with rate $\beta x_{0,1}$), for all $i \leq d$, and the second is due service completions from servers with exactly $d+1$ jobs (with rate $x_{d+1,2}$). The resulting rate can not be greater than λ, the rate where jobs are assigned to servers, and this justifies the $\mathbb{I}_{\{x_{d+1,2}+(d+1)\beta x_{0,1}\leq\lambda\}}$ term. Then, the excess of such rate, $(\lambda - x_{d+1,2} - (d+1)\beta x_{0,1})^+$, is distributed uniformly over servers with $i > d$ jobs. In Theorem 3, we will show that such rate is key for the design of fluid optimal scaling rules. Finally, assume that no warm server exists, i.e., $y_0 = 0$. Here, initializing servers get idle-on with rate $\beta x_{0,1}$ but all of them are immediately filled by new arrivals if $\lambda \geq \beta x_{0,1}$, and in this case the mass of idle-on servers remains zero. Otherwise, $x_{0,2}$ increases with surplus rate $\beta x_{0,1} - \lambda$.

The existence of a fluid solution started in $x^{(0)} \in \mathcal{S}_1$ will be direct from Theorem 1.

3.5 Scaling Rules

The scaling rule g gives the probability to activate a new server at each scaling time as a function of the system state. The following assumption, which will hold throughout the paper, provides the structure of the scaling rules investigated in this paper.

Assumption 2. *The scaling rule $g : \mathcal{S} \to [0,1]$ is Lipschitz continuous, and $g(x) > 0$ if $x_{0,0} = 1$.*

The last technical condition is natural and will rule out the existence of degenerate fixed points. We allow $g(x)$ to be greater than zero even when no cold server exists, i.e., $x_{0,0} = 0$. While this has no impact on the dynamics of the stochastic model, it does affect the fluid model as there may exist a flow of idle-on servers that go cold but instantly turn initializing keeping the proportion of cold servers at zero. This situation can occur if λ is large enough and not only in the transient regime; see Theorem 2.

We propose two scaling rules that satisfy Assumption 2.

Definition 2. *At each scaling time, if the system state is x,*

- Blind-θ *activates a new server with probability $g(x) = \theta$, $\theta \in (0,1]$;*
- Rate-Idle *activates a new server with probability $g(x) = \frac{1}{\lambda}(\lambda - \beta x_{0,1} - x_{1,2})^+$.*

Blind-θ is oblivious of the system state and thus highly scalable. Rate-Idle scales resources up if and only if the mean demand, λ, exceeds the rate at which servers become idle-on, $\beta x_{0,1} + x_{1,2}$. Here, the auto-scaler needs to know the amount of initializing servers, the amount of busy servers with exactly one job and both the job arrival and server initialization rates; in Knative, these variables are available to the auto-scaler. If combined with JIQ, we will show in Theorem 3 that Rate-Idle is asymptotically optimal.

4 Main Results

We now present our main results. In Theorem 1, we justify the use of the deterministic model to approximate the behavior of the stochastic model. Then, we focus on properties of the deterministic model and i) characterize its fixed points in Theorem 2 and ii) investigate the design of optimal scaling rules in Theorem 3.

4.1 Connection between the Fluid and Markov Models

The following result shows that the fluid model can be seen as a first-order approximation of the sample paths of the stochastic model.

Theorem 1. *Let $T < \infty$, $x^{(0)} \in \mathcal{S}_1$ and assume that $\|X^N(0) - x^{(0)}\|_w \to 0$ almost surely. Then, limit points of the stochastic process $(X^N(t))_{t\in[0,T]}$ exist and almost surely satisfy the conditions that define a fluid solution started at $x^{(0)}$.*

Proof. Given in Appendix 1. □

The stochastic and the deterministic models have some non-standard aspects that prevent us to prove Theorem 1 by directly applying Kurtz's theorem or similar known results. The main technical difficulty is that the trajectories of the deterministic model may cross or converge to points of discontinuity of its drift function. We handle this by following the general framework in [9], [31] and developing ad-hoc arguments specific to the structure of our problem (given in Appendix 1.1.3).

In view of Theorem 1 and since typical and default maximum scale limit values of real applications are 1000 or more [22], i.e., $N \geq 10^3$, we expect that the fluid model $x(t)$ provides an accurate approximation of the average behavior of $X^N(t)$. To support this claim, we present the results of numerical simulations; see also Section 6. Figure 1 (left) plots the trajectories of $x(t)$ and $X^N(t)$ when $N = 10^3$ and $B = 10^2$ along the coordinates of cold ($x_{0,0}$), initializing ($x_{0,1}$), idle-on ($x_{0,2}$) and busy (y_1) servers. Also, Figure 1 (right) plots the average number of jobs per warm server, which in state x is given by $Q(x) := \frac{1}{y_0}\sum_{i\geq 1} i x_{i,2}$. The fluid (stochastic) trajectories are always represented by dashed (continuous) lines and each curve is the average of ten simulations. Each simulation is based on 10^6 events. We have set $\lambda = 0.7$, $\alpha = 0.05$, $\beta = 0.1$ and $\gamma = 0.025$. As scaling rule, we have chosen Blind-θ where $\theta = \frac{0.5}{\alpha}\frac{1-\lambda}{\frac{1}{\beta}+\frac{1}{\gamma}}$; this choice will ensure that a strictly positive proportion of cold servers exists in the long run (see Theorem 2). As dispatching algorithm, we have used Power-of-2 (for JIQ, see Section 6). At time zero, we have assumed that the system is dimensioned exactly for the average demand, i.e., $(1-\lambda)N$ servers are cold and the remaining ones are idle-on. In both pictures, we observe that the fluid model captures the dynamics of $X^N(t)$ accurately.

Let us comment on the dynamics in Figure 1. Initially, the system is close to instability as capacity exactly matches demand. Here, $Q(x(t))$ increases rapidly and as soon as a warm server is created, it is filled with a job and as a result the proportion of idle-on servers decreases. These decrease also because they are not discovered fast enough upon job dispatching, thus letting them go cold even in heavy load. This explains why the number of cold servers (the blues lines) is increasing at the beginning. Then, more warm servers are created to mitigate the effect of the "close to instability" window on the accumulated overall number of jobs. Here, the mass of busy servers (y_1) becomes greater than the average demand $\lambda = 0.7$ and $Q(x(t))$ decreases. Finally, dynamics stabilize and in equilibrium there is a strictly positive fraction of servers that remain cold, initializing and idle-on. This indicates that there is a flux of idle-on servers that expires continuously even in equilibrium.

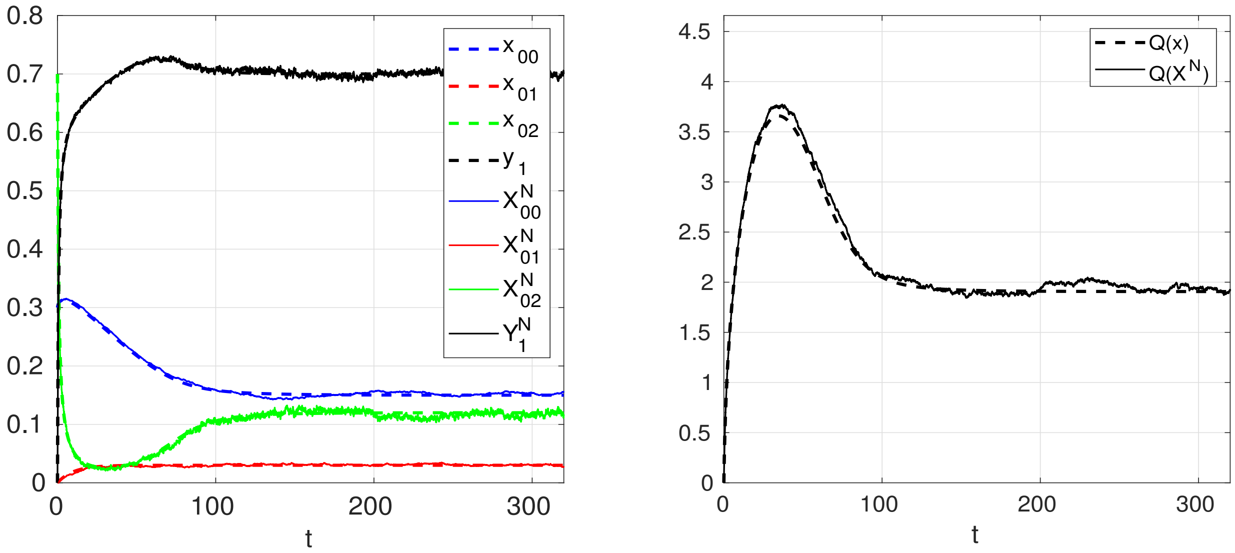


Figure 1. Numerical convergence of the stochastic model $X^N(t)$ (continuous lines), $N = 10^3$, to the fluid model $x(t)$ (dashed lines) when combining Power-of-2 and Blind-θ.

4.2 Characterization of Fixed Points

The fluid model has the form $\dot{x} = F(x)$; see Definition 1. We say that $x^* \in \mathcal{S}_1$ is a *fixed point* if $F(x^*) = 0$. We now investigate the fixed points of fluid model when buffer sizes are infinite and λ is constant and less than one (for stability).

Let us define the following conditions:

$$x_{0,0} + x_{0,1} + x_{0,2} + \lambda = 1 \tag{7a}$$
$$\beta x_{0,1} = \gamma x_{0,2} \tag{7b}$$
$$\gamma x_{0,2} \leq \alpha g(x), \quad \text{if } x_{0,0} = 0 \tag{7c}$$
$$\gamma x_{0,2} = \alpha g(x), \quad \text{if } x_{0,0} > 0 \tag{7d}$$

and if Power-of-d is used:

$$x_{i,2} = (\lambda + x_{0,2}) \left(\left(\frac{\lambda}{\lambda+x_{0,2}}\right)^{\frac{d^i-1}{d-1}} - \left(\frac{\lambda}{\lambda+x_{0,2}}\right)^{\frac{d^{i+1}-1}{d-1}} \right), \tag{8}$$

for all $i \geq 1$, otherwise if JBT-d is used:

$$\text{if } x_{0,2} = 0 : x_{i,2} = 0, \quad 0 \leq i \leq d \tag{9a}$$
$$x_{d+i,2} = x_{d+1,2} \left(1 - \frac{x_{d+1,2}}{\lambda}\right)^{i-1}, \ i \geq 2 \tag{9b}$$
$$x_{d+1,2} \in (0, \lambda] \tag{9c}$$
$$g(x) = 0 \tag{9d}$$
$$\text{if } x_{0,2} > 0 : x_{i,2} = \left(\frac{\lambda}{z_d + x_{0,2}}\right)^i x_{0,2} \, \mathbb{I}_{\{1 \leq i \leq d+1\}}, \ i \geq 1 \tag{9e}$$

with $z_d \in [0, 1]$ being the unique solution of

$$z_d + x_{0,2} = \frac{1 - \left(\frac{\lambda}{z_d + x_{0,2}}\right)^{d+1}}{1 - \frac{\lambda}{z_d + x_{0,2}}} x_{0,2} \tag{10}$$

if $d \geq 1$ and $z_d = 0$ if $d = 0$. Here, z_d is interpreted as the proportion of busy servers with no more than d jobs.

Now, let us also introduce the following assumption, which we will only use in Theorem 2 below.

Assumption 3. *For any* $x_{0,2} \in [0, 1 - \lambda]$, (8)-(9e) *uniquely determine* $x_{i,2}$ *for all* $i \geq 1$.

Within Power-of-d, this assumption is clearly satisfied by (8). Within JBT-d, it is satisfied only if $x_{0,2} > 0$, as if $x_{0,2} = 0$, then $x_{d+1,2}$ is only required to belong to $(0, \lambda]$. Under Assumption 3, let $x^\circ = (x^\circ_{i,j})$ be the unique point in $\mathcal{S}_1$ such that $x^\circ_{0,0} = 0$, $x^\circ_{0,1} = \frac{\gamma}{\beta+\gamma}(1-\lambda)$, $x^\circ_{0,2} = \frac{\beta}{\beta+\gamma}(1-\lambda)$.

The following result characterizes fixed points.

Theorem 2. *Assume that λ is constant and less than one. If x^* satisfies the conditions in* (7)-(10), *then it is a fixed point of the fluid model with $B = +\infty$. In addition, under Assumption 3*

1) *If*
$$\alpha g(x^\circ) < \frac{1 - \lambda}{\frac{1}{\beta} + \frac{1}{\gamma}}, \tag{11}$$
then $x^*_{0,0} > 0$.
2) *If* (11) *does not hold, then $x^* = x^\circ$ is the unique fixed point.*

Proof. Given in Appendix 1. □

At the fluid scale and in a fixed point, Theorem 2 also provides the boundary scaling probability that distinguishes between a "saturated" and a non-saturated system. Specifically, if the scaling rule satisfies (11), then in a fixed point there exists a fraction of idle-on servers that go cold and instantly become initializing, provided that $g(x^\circ) > 0$. Here, the pool of cold servers remains non-empty. On the other hand, if $g(x^\circ)$ does not satisfy (11), then no cold server exists in a fixed point but we observe that (7a) and (7b) imply that a strictly positive fraction of servers remain initializing, i.e., $\frac{\gamma}{\beta+\gamma}(1 - \lambda)$. Here, the interpretation is that there still exists a mass of idle-on servers that go cold but instantly become initializing while keeping the proportion of cold servers down to zero. This corresponds to a waste of resources because initializing servers cannot process jobs. In other words, a better performance may be obtained by keeping the initializing servers warm at all times (no auto-scaling); recall also that billing policies charge warm and initializing servers.

Within Blind-θ, $g(x) = \theta$ and the conditions (7)-(10) easily identify a unique fixed point, say x^*, with $(x^*_{0,0}, x^*_{0,1}, x^*_{0,2})$ *not* depending on the choice of the load balancing algorithm.

The following remark says that uniqueness is not always guaranteed.

Remark 1 (Multiple Fixed Points)**.** *Suppose that $g(x) = 0$ whenever $y_1 = \lambda = 1 - x_{0,0}$ and that JBT-d is used. Then, Theorem 2 implies that uncountably many fixed points exist. In fact, while $x_{i,2} = 0$ for all $i = 0, \dots, d$ and $x_{i,2}$ is uniquely determined for all $i \geq d+2$* once fixed *$x_{d+1,2}$, the conditions* (7)-(10) *do not tie $x_{d+1,2} \in (0, \lambda]$ to a specific value.*

4.2.1 Blind-θ and Random Dispatching

For illustration purposes, let us consider Blind-θ with random dispatching (Power-of-1). This combination does not involve any communication overhead among the auto-scaler, dispatchers and servers, and for this reason it is well suited for large systems with vast numbers of dispatchers. Here, Theorem 2 identifies a unique fixed point, x^*. After some algebra, we obtain $x^*_{0,2} = \min\left\{\frac{\alpha\theta}{\gamma},\, x^\circ_{0,2}\right\}$ and for the mean queue length per warm server, $Q(x) = \frac{1}{y_0}\sum_i i x_{i,2}$, we obtain (using also (8))

$$Q(x^*) = \frac{\lambda}{\min\left\{\frac{\alpha\theta}{\gamma},\, x^\circ_{0,2}\right\}}. \tag{12}$$

As long as a strictly positive fraction of cold servers exists, or equivalently $\frac{\alpha\theta}{\gamma} < x^\circ_{0,2}$, we remark that $Q(x^*)$ grows *linearly* in λ.

4.3 Optimal Design

Within Blind-θ, Theorem 2 guarantees the existence of a unique fixed point and all of our numerical simulations, which we omit, indicate that it is a global attractor. Here, necessarily $x^*_{0,2} > 0$, by (7d), which means that a number of warm servers remain idle-on in equilibrium. Clearly, this is not optimal for energy consumption because idle-on servers consume energy. Our goal now is to design scaling rules ensuring that a global attractor exists *and* given by $x^\star$, where $x^\star \in \mathcal{S}$ is uniquely defined by $x^\star_{0,0} = 1 - \lambda$ and $x^\star_{1,2} = \lambda$.

Remark 2 (Fluid Optimality)**.** *In $x^\star$ dynamics have achieved* "delay and relative energy optimality" *in the sense that both the waiting time of jobs and the relative energy portion consumed by idle-on and initializing servers vanish in the limit. Here, a possible intuition is that each job is always assigned to a busy server with exactly one job but at the precise moment where it completes the processing of its previous job. Therefore, service capacity perfectly matches demand.*

A direct consequence of Theorem 2 and (7d) is that it is necessary to impose $g(x^\star) = 0$ to achieve fluid optimality. Within Power-of-d, this is impossible as this condition would imply that $x_{0,2} = 0$, and then (8) would imply $x_{i,2} = 0$ for all i, contradicting that $\|x\| = 1$. In fact, Theorem 2 implies that the unique candidate is JIQ, though it leaves open the possibility that $x(t)$ may converge to a fixed point in the sub-optimal set $\mathcal{S}_{\text{subopt}}$, see (14). Thus, it remains to understand what additional structure the scaling rule $g(x)$ should satisfy to make $x^\star$ a global attractor. Here, Remark 1 suggests that even the knowledge of the amount of busy servers is not enough. More precisely, it implies that one needs $g(x) > 0$ for all $x \in \mathcal{S}_{\text{subopt}}$ as otherwise multiple fixed points exist. Therefore, given the structure of $\mathcal{S}_{\text{subopt}}$ and $x^\star$, we have the following remark.

Remark 3. *A fluid optimal scaling rule needs the access to the amount of busy servers with exactly one job, i.e., $x_{1,2}$.*

The following result provides a general condition that yields fluid optimality.

Theorem 3 (Optimal Design)**.** *Let $\beta < 1$ and let $x(t)$, with $x(0) \in \mathcal{S}_1$, denote a fluid solution induced by JIQ and any scaling rule $g(x)$ that satisfies, beyond Assumption 2,*

$$g(x) = 0 \text{ if and only if } x_{1,2} + \beta x_{0,1} \geq \lambda. \tag{13}$$

Then, $\lim_{t\to\infty} \|x(t) - x^\star\|_w = 0$.

Proof. Given in Appendix 1. □

The interpretation is that $x_{1,2} + \beta x_{0,1}$ represents the overall rate at which servers become idle-on. Thus, our optimality condition says to scale up resources whenever the excess of the mean demand over the rate at which servers become idle-on is positive, as in this case JIQ is smart enough to fill them up immediately saturating the surplus service capacity. Otherwise, if the excess is negative, one can turn the scale-up process off ($g = 0$), and in this case the natural dynamics induced by both JIQ and the scale-down rule are enough to drive the system behavior to the desirable configuration $x^\star$.

Remark 4. Rate-Idle*, see Definition 2, satisfies* (13)*. If g denotes* Rate-Idle *and $f : [0,1] \to [0,1]$ is continuous, onto and increasing, then $f(g)$ is a scaling rule that as well satisfies* (13).

As discussed in Section 3.1, the assumption $\beta < 1$, i.e., the mean server initialization rate is smaller than the mean job service rate, is largely accepted in practice [22], [35]. From a mathematical standpoint, it is not necessary for fluid optimality but simplifies our proof.

Remark 5 (Communication Overhead)**.** *A scaling rule satisfying* (13) *requires the central controller to have access to the amount of initializing and busy servers containing exactly one job, i.e., $x_{0,1}$ and $x_{1,2}$. Since an initializing server informs the platform as soon as it becomes warm, $x_{0,1}$ is easily obtained in practice. For $x_{1,2}$, the auto-scaler can run a local memory with N slots, where the n-th slot indicates the state of server n, say 'Cold', 'Init', 'Idle-on', 'Busy$_1$' and 'Busy$_{\geq 2}$', with obvious interpretations. Then, one way to update the memory is by letting each server send a message to the auto-scaler whenever the transitions 'Busy$_{\geq 2}$' → 'Busy$_1$, 'Busy$_1$' → 'Idle-on' and 'Idle-on' → 'Busy$_1$' occur. As in standard implementations of JIQ, this involves only a constant number of messages per job to be exchanged between the auto-scaler and the servers.*

4.4 Convergence to Multiple Fixed Points

In Theorem 3, we have provided a condition ensuring that $x^\star$ is globally stable. In this section, we show that it is not always possible to have global stability. To guarantee stability, one may expect that is enough to have a strictly positive scaling probability whenever the current capacity of warm servers is less than the average demand, i.e., $g(x) > 0$ whenever $y_0 < \lambda$. The following proposition shows that this intuition is false.

Let

$$\mathcal{S}_{\text{subopt}} := \Big\{ x \in \mathcal{S} : x_{0,0} = 1-\lambda,\ x_{0,1} = x_{0,2} = 0, \\ x_{1,2} < \lambda \text{ and (9b) holds with } d = 0 \Big\} \quad (14)$$

and let $\overline{Q}(x) := \sum_{i\geq 1} i x_{i,2}$ denote the average number of jobs per server in state $x \in \mathcal{S}$; here, cold and initializing servers are included in the counting.

Proposition 1. *Assume that λ is constant and less than one. Let $g(x)$ be any scaling rule such that*

$$g(x) = \frac{1}{\lambda}(x_{0,0} - 1 + \lambda)^+, \qquad \forall x \in \mathcal{S} : y_0 < \lambda. \quad (15)$$

Let $x(t)$ denote a fluid model induced by such $g(x)$ and JIQ such that

$$x_{0,0}(0) > 1-\lambda,\ x_{0,2}(0) = 0, \\ x_{1,2}(0) + \beta x_{0,1}(0) < \lambda < \overline{Q}(x(0)) < \infty. \quad (16)$$

Suppose that $\beta < 1$, $\alpha \neq \beta$ and $B = +\infty$. Then,

$$g(x(t)) > 0, \quad \forall t \geq 0 \quad (17)$$

$$\lim_{t\to\infty} \overline{Q}(x(t)) = \overline{Q}(x(0)) + \frac{x_{0,1}(0)}{\beta} \\ + \frac{\alpha+\beta}{\alpha\beta}(x_{0,0}(0) - 1 + \lambda) > \lambda. \quad (18)$$

In addition, $y_0(t) \uparrow \lambda$, and if $x_{1,2}(t) \to x_{1,2}(\infty)$, then $x(t) \to x(\infty)$ with $x(\infty) \in \mathcal{S}_{\text{subopt}}$.

Proof. Given in Appendix 3. □

Thus, while the proportion of warm servers converges to λ, such convergence may occur *from below* even if there always exists a strictly positive probability of creating new warm servers. In this case, the average demand is greater than the current service capacity at any point in time and this makes the mean queue length converge to a limit that depends on the initial conditions.

Let us comment a little bit further and prepare the setting for our next contribution. To create the underload situation above where $y_0(t) \uparrow \lambda$, it is not necessary to assume that all warm servers are initially busy ($x_{0,2}(0) = 0$), though we have included this condition in (16) to simplify our proof. In contrast, to avoid this situation, it may be sufficient that $g(x)$ is bounded away from zero whenever $y_0 < \lambda$. By continuity, this implies that $g(x) > 0$ as well whenever $y_0 = \lambda$, but in this case the resulting scaling rule will not possess the optimality property stated in Theorem 3 below (as this will imply that $g(x^\star) > 0$). On the other hand, one may consider a scaling rule that is discontinuous on the set $\{x : y_1 = \lambda\}$, a setting that does not satisfy Assumption 2. Here, beyond revisiting Theorem 1 for justification of the fluid model, the problem is that scale-up decisions would significantly depend on small perturbations of the equilibrium system state, severely impacting robustness from a practical standpoint.

5 Empirical Comparison: Synchronous vs Asynchronous

The structural differences between the synchronous and asynchronous approaches have been described in Section 1.3. In this section, we compare both approaches by means of numerical simulations. Specifically, we compare our asynchronous combination of JIQ and Rate-Idle (see Definition 2) with a generalization of TABS, i.e., the synchronous scheme developed in [26]. For the latter, we assume that d servers are initialized at the moment of a job arrival if all active servers are busy upon arrival of that job, in which case the job is sent to a (busy) server at random. Thus, the TABS scheme in [26] is recovered when $d = 1$. Let us refer to such generalization as TABS-d. Clearly, d affects the scale-up rate and plays the same role of α in ALBA. To make the comparison fair, we will assume that α is fine-tuned such that the resulting *scale-up rate* induced by ALBA matches the scale-up rate induced by TABS-d; thus, $\alpha = \alpha(d)$. Here, the scale-up rate is defined as the number of server initialization signals divided by the time horizon.

Our comparison metrics are

- the empirical probability of waiting, that is the average fraction of jobs that are sent to a busy server. We refer to these as $p_{\text{Wait}}^{\text{ALBA}}$ and $p_{\text{Wait}}^{\text{TABS}-d}$.
- the empirical energy consumption, that is $E = N(w_{\text{init}} x_{0,1}(t) + w_{\text{idle-on}} x_{0,2}(t) + w_{\text{busy}}) y_1(t)$ averaged over time; here, we assume $w_{\text{init}} = 2$, $w_{\text{idle-on}} = 0.5$ and $w_{\text{busy}} = 1$. We refer to these as E^{ALBA} and $E^{\text{TABS}-d}$.

Then, we consider the ratios

$$\mathcal{R}_{\text{Wait}} := \frac{p_{\text{Wait}}^{\text{ALBA}}}{p_{\text{Wait}}^{\text{TABS}-d}}, \quad \mathcal{R}_{\text{Energy}} := \frac{E^{\text{ALBA}}}{E^{\text{TABS}-d}}, \quad (19)$$

and evaluate them by simulation of 10^7 events (both schemes have been tested within the same seed sequences) and when $N \in \{100, 500, 1000\}$, $\lambda \in \{0.35, 0.7\}$, $d = \{1, 5, 10\}$, $\beta = 0.1$ and $\gamma = 0.025$. If a time unit is 10 milliseconds, these parameters are realistic [11], [22], [35]. We also assume that the initial condition is $x^\star$, i.e., the global attractor of the fluid dynamics defined in Section 4.3. This choice measures the perturbations of order $1/N$ that appear around $x^\star$, which are not visible at the fluid scale. Within this setting, Figure 2 plots $\mathcal{R}_{\text{Wait}}$ (blue) and $\mathcal{R}_{\text{Energy}}$ (red) and shows that ALBA always provides a much smaller probability of waiting than TABS-d while inducing the same energy consumption cost as $\mathcal{R}_{\text{Energy}}$ is almost one; see the Appendix for a table containing numerical data. In addition, this behavior is amplified when N and d increase. As discussed in Section 1.3, we own the performance gain of ALBA to the fact that scale up decisions may be taken before job arrivals, while in a synchronous scheme such as TABS-d, jobs are forced to wait any time a scale up decision is taken. While this anticipation induces a slightly increased energy cost, it pays off because $\mathcal{R}_{\text{Energy}}$ remains very close to one.

Since $\mathcal{R}_{\text{Wait}}$ decreases with the system size N, we may postulate that it approaches zero as $N \to \infty$. This requires a second-order limit analysis of the underlying Markov chains, which we leave as future work.

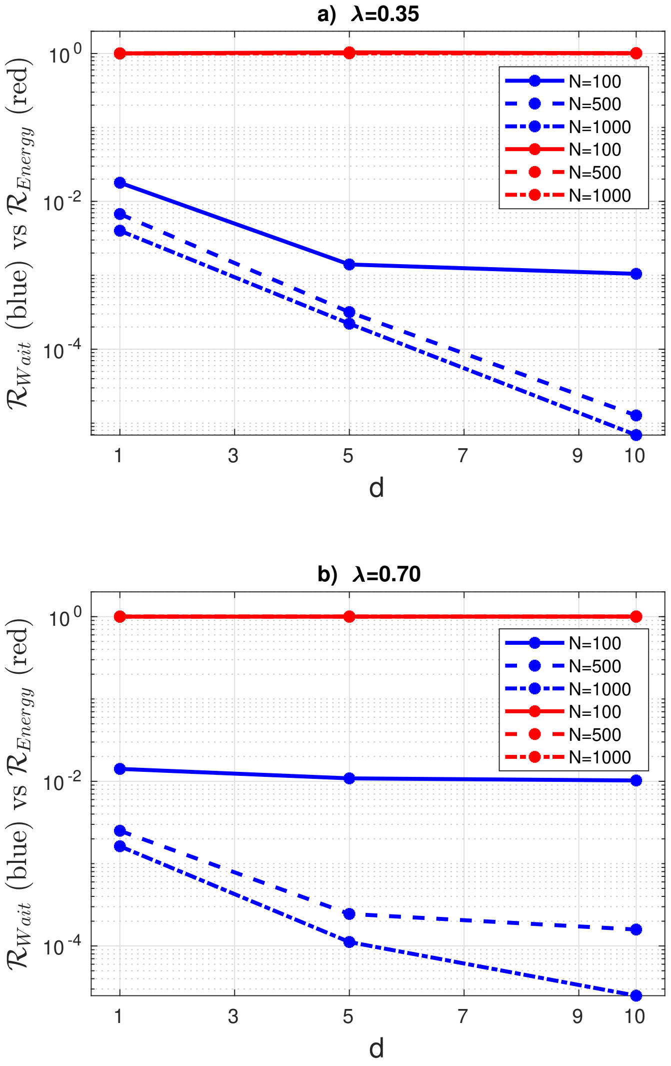


Figure 2. Ratio $\mathcal{R}^{(N)}$ of the transient probability of waiting induced by the proposed asynchronous scheme (Rate-Idle+JIQ) and the synchronous approach in [26], respectively. The initial condition is the global attractor $x^\star$ (defined in Section 4.3, see Theorem 3), which corresponds to delay and relative energy optimality.

6 Energy Optimization with Performance Guarantees

In this section, we use the fluid model in an optimization framework to trade off between performance and energy costs, and we numerically show that it accurately captures the stochastic dynamics of the finite ALBA system. Let us focus on JIQ as load balancing algorithm and on the set (say $\mathcal{G}$) of scaling rules that satisfy the assumptions in Theorem 3. Note that these imply fluid optimality in the stationary regime. As in, e.g., [4], let us define the cost function $\mathcal{J}_g$ as the long-run time average of a linear combination between the power consumption $P(x) = c_{0,1}x_{0,1} + c_{0,2}x_{0,2} + c_{1,2}y_1$, $c_{i,j} > 0$, and the average queue length per busy server $Q(x) = \frac{1}{y_1}\sum_i i x_{i,2}$ induced by the scaling rule g, i.e.,

$$\mathcal{J}_g := \lim_{T\to\infty} \frac{1}{T}\int_0^T (\kappa_1 P(x(t)) + \kappa_2 Q(x(t)))\,\mathrm{d}t \qquad (20)$$

where $\kappa_i \geq 0$, $i = 1, 2$; one can think κ_1 in terms of \$/watt and κ_2 in terms of \$/job. Then, Theorem 3 implies that

$$\inf_g \mathcal{J}_g = \mathcal{J}_{g^*} = \kappa_1 P(x^\star) + \kappa_2 Q(x^\star) = \kappa_1 c_{1,2}\lambda + \kappa_2 \qquad (21)$$

for all $g^* \in \mathcal{G}$. While all policies in $\mathcal{G}$ yield the same (optimal) cost, their behavior is clearly different trajectory-wise. Depending on the application, a platform user has several options to single out a policy in $\mathcal{G}$ that satisfies a further level of optimization. For instance, a substantial portion of the applications hosted in cloud networks have ultra-low delay requirements, as this may have important consequences on e-commerce sales. On the other hand, also energy bills are equally important from both financial and environmental standpoints. Here, a system manager may want to look for a scaling rule in $\mathcal{G}$ such that

$$Q(x(t)) \leq q, \quad \forall t \geq 0 \qquad (22)$$

where q is related to the desired user-perceived performance guarantee; by Little's law, (22) is equivalent to a constraint on the mean response time. In view of Remark 4, one may consider the parameterized subset of scaling rules

$$g(x) = \frac{1 - \exp(-\frac{\eta}{\lambda}(\lambda - x_{1,2} - \beta x_{0,1})^+)}{1 - \exp(-\eta)}, \quad \eta > 0, \qquad (23)$$

which satisfy both Assumption 2 and (13). Here, the control parameter $\eta > 0$ indicates how aggressive the scaling rule is: Rate-Idle is recovered when $\eta \downarrow 0$ and $g(x) = \mathbb{I}_{\{\lambda \geq x_{1,2} + \beta x_{0,1}\}}$ when $\eta \to \infty$. Then, one may search for the smallest (least aggressive) η such that (22) holds true.

The above problem can be easily addressed numerically within the proposed deterministic model. Assume that the system is currently in a light-load condition, say $\lambda = 0.25$, and that, as a result, it is dimensioned accordingly to save energy, say $x_{0,0} = 1 - \lambda - 0.05$, with $x_{0,2} = 0.05$ and $x_{1,2} = \lambda$; the extra 0.05 is meant to keep a reserve of idle-on servers ready to go. Then, at time zero, an unexpected workload peak occurs, and $\lambda = 0.5$. Here, the platform needs to automatically adjust the service capacity while ensuring (22). Let us assume $q = 2$, $\alpha = 0.35$, $\beta = 0.1$ and $\gamma = 0.025$. The dashed lines in Figure 3 represent the dynamics of the fluid queue lengths $Q(x(t))$ and scaling probabilities $g(x(t))$, for $\eta = 1, 10^3$. The corresponding continuous lines represent the average of ten simulations of the stochastic model $X^N(t)$ with $N = 1000$. First, let us remark that the fluid model approximation accurately captures the dynamics of $X^N(t)$, though it slightly underestimates queue lengths and scaling probabilities. Now, let us consider $\eta = 1$. Initially, queue lengths increase as expected due to the surge of demand and the scaling probability is large enough to drive the proportion of cold servers to zero. This explains the non-differentiability point of the trajectory of the scaling rule because the amount of initializing servers stops to grow. Then, the system has enough capacity to drain the load and at some point the rate at which servers become idle-on overflows the mean demand, i.e., $x_{1,2} + \beta x_{0,1} > \lambda$, so that eventually $g(x(t)) = 0$. Finally, queue lengths assess to their asymptotic value $Q(x^\star) = 1$ We conclude that $\eta = 1$ is enough to make (22) holds true. We also notice that the choice $\eta = 10^3$, which essentially means to scale up resources at the maximum available rate α whenever

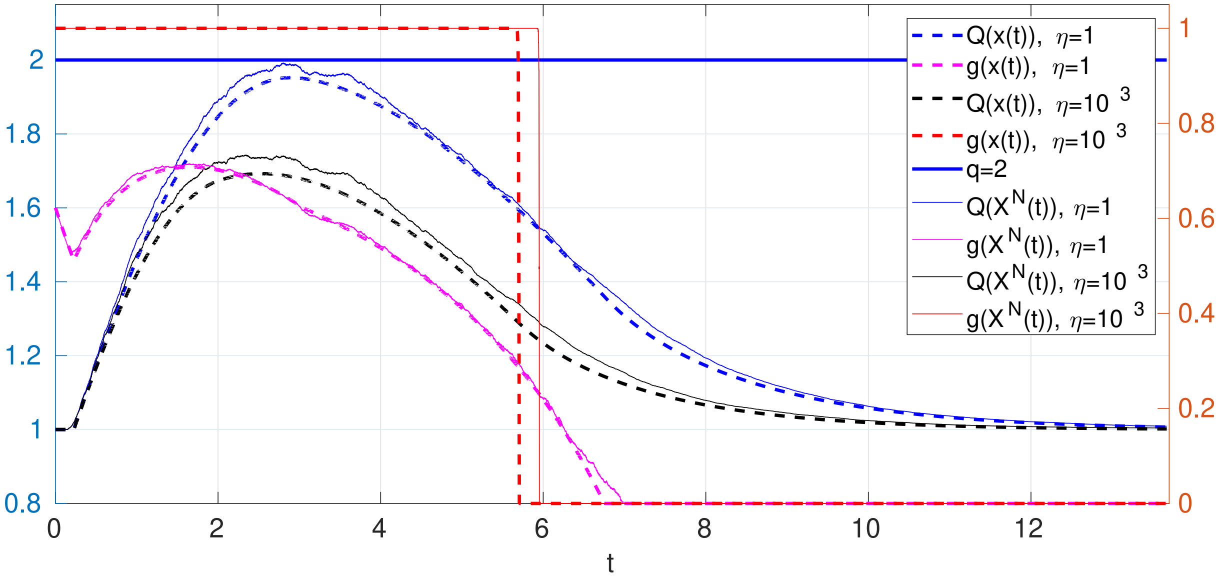


Figure 3. Transient behavior of the queue lengths (y-axis on the left) and scaling probabilities (y-axis on the right) by varying η, see (23), for both the fluid ($x(t)$) and stochastic ($X^N(t)$) models with $N = 1000$.

$x_{1,2}+\beta x_{0,1} < \lambda$, has little impact on performance. Nonetheless, it should be clear that the larger the value of η, the larger the resulting time-average power consumption.

7 Conclusion

In cloud systems, load balancing and auto-scaling are key mechanisms to optimize both delay performance and energy consumption. The focus of the existing literature has been on architectures where these mechanisms are synchronous or rely on a central queue. The novelty of our work is to consider an asynchronous and decentralized architecture. Decentralization increases scalability and asynchronism does not force jobs to wait any time a scale-up decision is taken.

Our work provides a tractable framework to evaluate the performance of auto-scaling algorithms that are up to the platform user to design. In our main result, we have identified a structural condition for asymptotic optimality that provides the platform user with some flexibility when designing an optimal scaling rule; see Remark 4. This can be exploited to develop new levels of optimization as we have shown in Section 6. By means of numerical simulations, we have show that the proposed asynchronous combination of JIQ and Rate-Idle provides a better delay performance than existing synchronous decentralized schemes while inducing almost the same energy consumption.

We discuss some generalizations and open questions:

- We have assumed that only one server at a time can be activated at each scaling time. Our approach generalizes trivially to the case where a random number C of cold servers is selected, provided that the distribution of C does not depend on N. Mutatis mutandis, it is enough to replace α by $\alpha\,\mathbb{E}[C]$.
- Theorem 1 generalizes trivially to a time-varying arrival rate setting if the arrival rate takes the form $\Lambda(t)N$ where $\Lambda(t)$ is a bounded positive real-valued function independent of N. This change only affects Lemma 1 of the supplementary material, whose proof directly generalizes by the functional strong law of large numbers for the Poisson process. The resulting deterministic model is identical to the one in Definition 1 except that λ is replaced by $\Lambda(t)$.
- From a theoretical point of view, it is interesting to prove the “interchange of limits” property. More specifically, within JIQ and the asymptotically optimal condition identified in Theorem 3, the question is whether or not the invariant distribution of the underlying Markov chain concentrates on $x^\star$ when $N \to \infty$. Numerical evidence indicates that this property holds true.
- The stability of the (finite) stochastic model is a difficult question to answer because the proposed ALBA framework is very general: the scale-up rule g satisfies mild conditions (see Assumption 2) and to come up with a stability result, one should take additional assumptions such as considering a specific scale-up policy. Even within the simplest scale-up policy, i.e., Blind-θ, and the simplest dispatching policy, i.e., where jobs are distributed to servers uniformly at random (or equivalently Power-of-d with $d = 1$), understanding whether or not the underlying Markov chain is positive recurrent is challenging. Here, one may check that (natural adaptations of) classical Lyapunov functions used in queueing theory to investigate stability via Foster-Lyapunov theorem do not work. Also, the utilization of Dai’s fluid framework [10] is again complicated by the identification of a Lyapunov function. Finally, the drift function does not preserve monotonicity and stochastic dominance arguments cannot be applied.

References

[1] Knative docs v1.3. https://knative.dev/docs/, 2022. Online; accessed: 2023-01-30.

[2] Knative Load balancing. https://knative.dev/docs/serving/load-balancing/, 2022. Online; accessed: 2023-01-30.

[3] Knative scale bounds. https://knative.dev/docs/serving/autoscaling/scale-bounds/, 2022. Online; accessed: 2023-01-30.

[4] L. L. Andrew, M. Lin, and A. Wierman. Optimality, fairness, and robustness in speed scaling designs. In *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '10, page 37–48, New York, NY, USA, 2010. Association for Computing Machinery.

[5] J. Anselmi. Asymptotically optimal open-loop load balancing. *Queueing Systems*, pages 1–23, Sept. 2017.

[6] J. Anselmi. Combining size-based load balancing with round-robin for scalable low latency. *IEEE Transactions on Parallel and Distributed Systems*, 31(4):886–896, 2020.

[7] J. Anselmi and J. Doncel. Asymptotically optimal size-interval task assignments. *IEEE Transactions on Parallel and Distributed Systems*, to appear.

[8] J. Anselmi and F. Dufour. Power-of-d-choices with memory: Fluid limit and optimality. *Math. Oper. Res.*, 45(3):862–888, 2020.

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

[10] J. G. Dai. On positive harris recurrence of multiclass queueing networks: a unified approach via fluid limit models. *The Annals of Applied Probability*, pages 49–77, 1995.

[11] J. Dean and L. A. Barroso. The tail at scale. *Commun. ACM*, 56(2):74–80, Feb. 2013.

[12] Y. Desmouceaux, M. Enguehard, and T. H. Clausen. Joint monitorless load-balancing and autoscaling for zero-wait-time in data centers. *IEEE Transactions on Network and Service Management*, 18(1):672–686, 2021.

[13] J. Dogani and F. Khunjush. Proactive auto-scaling technique for web applications in container-based edge computing using federated learning model. *Journal of Parallel and Distributed Computing*, 187:104837, 2024.

[14] D. Gamarnik, J. N. Tsitsiklis, and M. Zubeldia. Delay, memory, and messaging tradeoffs in distributed service systems. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*, SIGMETRICS '16, pages 1–12, New York, NY, USA, 2016. ACM.

[15] D. Gamarnik, J. N. Tsitsiklis, and M. Zubeldia. Delay, memory, and messaging tradeoffs in distributed service systems. *Stochastic Systems*, 8(1):45–74, 2018.

[16] A. Gandhi, S. Doroudi, M. Harchol-Balter, and A. Scheller-Wolf. Exact analysis of the m/m/k/setup class of markov chains via recursive renewal reward. In *Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '13, page 153–166, New York, NY, USA, 2013. Association for Computing Machinery.

[17] D. Goldsztajn, A. Ferragut, F. Paganini, and M. Jonckheere. Controlling the number of active instances in a cloud environment. *SIGMETRICS Perform. Eval. Rev.*, 45(3):15–20, Mar. 2018.

[18] M. Harchol-Balter, M. E. Crovella, and C. D. Murta. On choosing a task assignment policy for a distributed server system. *Journal of Parallel and Distributed Computing*, 59(2):204 – 228, 1999.

[19] M. Harchol-Balter, A. Scheller-Wolf, and A. R. Young. Surprising results on task assignment in server farms with high-variability workloads. SIGMETRICS '09, pages 287–298, New York, NY, USA, 2009. ACM.

[20] Z. Liu and R. Righter. Optimal load balancing on distributed homogeneous unreliable processors. *Operations Research*, 46(4):563–573, 1998.

[21] Y. Lu, Q. Xie, G. Kliot, A. Geller, J. R. Larus, and A. Greenberg. Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services. *Perform. Eval.*, 68(11):1056–1071, Nov. 2011.

[22] N. Mahmoudi and H. Khazaei. Performance modeling of serverless computing platforms. *IEEE Transactions on Cloud Computing*, pages 1–1, 2020.

[23] N. Mahmoudi, C. Lin, H. Khazaei, and M. Litoiu. Optimizing serverless computing: Introducing an adaptive function placement algorithm. In *Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering*, CASCON '19, page 203–213, USA, 2019. IBM Corp.

[24] M. Mitzenmacher. The power of two choices in randomized load balancing. *IEEE Trans. Parallel Distrib. Syst.*, 12(10):1094–1104, Oct. 2001.

[25] D. Mukherjee, S. C. Borst, J. S. H. van Leeuwaarden, and P. A. Whiting. Asymptotic Optimality of Power-of-d Load Balancing in Large-Scale Systems. *ArXiv e-prints*, Dec. 2016.

[26] D. Mukherjee, S. Dhara, S. C. Borst, and J. S. van Leeuwaarden. Optimal service elasticity in large-scale distributed systems. *Proc. ACM Meas. Anal. Comput. Syst.*, 1(1), June 2017.

[27] D. Mukherjee and A. Stolyar. Join idle queue with service elasticity: Large-scale asymptotics of a nonmonotone system. *Stochastic Systems*, 9(4):338–358, 2019.

[28] C. Qu, R. N. Calheiros, and R. Buyya. Auto-scaling web applications in clouds: A taxonomy and survey. *ACM Comput. Surv.*, 51(4), July 2018.

[29] M. Shahrad, R. Fonseca, I. Goiri, G. Chaudhry, P. Batum, J. Cooke, E. Laureano, C. Tresness, M. Russinovich, and R. Bianchini. Serverless in the wild: Characterizing and optimizing the serverless workload at a large cloud provider. In *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, pages 205–218. USENIX Association, July 2020.

[30] A. L. Stolyar. Pull-based load distribution in large-scale heterogeneous service systems. *Queueing Syst. Theory Appl.*, 80(4):341–361, Aug. 2015.

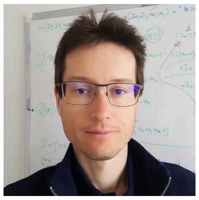
[31] J. N. Tsitsiklis and K. Xu. On the power of (even a little) resource pooling. *Stoch. Syst.*, 2(1):1–66, 2012.

[32] M. van der Boor, S. C. Borst, and J. van Leeuwaarden. Hyper-scalable JSQ with sparse feedback. *Proc. ACM Meas. Anal. Comput. Syst.*, 3(1):4:1–4:37, 2019.

[33] M. van der Boor, S. C. Borst, J. S. van Leeuwaarden, and D. Mukherjee. Scalable load balancing in networked systems: A survey of recent advances. *arXiv preprint arXiv:1806.05444*, 2018.

[34] E. van Eyk, A. Iosup, C. L. Abad, J. Grohmann, and S. Eismann. A spec rg cloud group's vision on the performance challenges of faas cloud architectures. In *Companion of the 2018 ACM/SPEC International Conference on Performance Engineering*, ICPE '18, page 21–24, New York, NY, USA, 2018. Association for Computing Machinery.

[35] L. Wang, M. Li, Y. Zhang, T. Ristenpart, and M. Swift. Peeking behind the curtains of serverless platforms. In *Proceedings of the 2018 USENIX Conference on Usenix Annual Technical Conference*, USENIX ATC '18, page 133–145, USA, 2018. USENIX Association.



Jonatha Anselmi is a tenured researcher at the French National Institute for Research in Digital Science and Technology (Inria), since 2014. Prior to this, he was a researcher at the Basque Center for Applied Mathematics and a postdoctoral researcher at Inria. He received his PhD in computer engineering at Politecnico di Milano (Italy) in 2009. His research interests are in the broad field of decision-making under uncertainty, where computer science, applied maths and engineering intersect.

8 Proofs of Theorems 1, 2 and 3

8.1 Theorem 1: connection between the fluid and the Markov models

To prove Theorem 1, we follow two main steps. First, we couple the processes $(X^N(t))_{t\in[0,T]}$, for all $N \in \mathbb{Z}_+$, on a common probability space and show that limit trajectories exist and are Lipschitz continuous with probability one. The arguments used in this step are routine [7], [10], [28]. Then, we prove that limit trajectories are fluid solutions, which is the main technical difficulty, and here we develop arguments specific to the model under investigation.

8.1.1 Coupled construction of sample paths

Let $\mathcal{N}_c(t)$ denote a Poisson process of rate c. We construct a probability space where the stochastic processes $\{(X^N(t))_{t\in[0,T]}\}_{N\geq 1}$ are coupled. All the processes of interest can be constructed in terms of the following mutually independent primitive processes:

- $\mathcal{N}_\phi(t)$, a Poisson process of rate $\phi := \lambda+1+\alpha+\beta+\gamma$. This process is defined on $(\Omega_E, \mathcal{A}_E, \mathbb{P}_E)$ and each jump of $\mathcal{N}_\phi(t)$ denotes the occurrence of an *event*.
- $(W_n)_n$, where the random variables W_n are $\{0,1,2,3,4\}$-valued i.i.d. and such that $\mathbb{P}(W_n = 0) = \lambda/\phi$, $\mathbb{P}(W_n = 1) = 1/\phi$, $\mathbb{P}(W_n = 2) = \alpha/\phi$ $\mathbb{P}(W_n = 3) = \beta/\phi$ and $\mathbb{P}(W_n = 4) = \gamma/\phi$. This process is defined on $(\Omega_W, \mathcal{A}_W, \mathbb{P}_W)$ and will identify the *type* of the n-th event. Specifically, $W_n = 0$ indicates a job arrival, $W_n = 1$ a potential job departure, $W_n = 2$ a scaling time, $W_n = 3$ a potential server initialization, i.e., a server completed the initialization phase, and $W_n = 4$ a potential server expiration.
- $(A_n^p)_n$, $p = 1,\ldots,d$, $(D_n)_n$, $(I_n)_n$, $(E_n)_n$ and $(R_n)_n$, where the random variables A_n^p, D_n, I_n, E_n and R_n, for all n, are all i.i.d. and uniform over the interval $[0,1]$. The rvs A_n^p, D_n, I_n, E_n will be respectively used to select a server that i) will process an arriving job, ii) fires a departure, iii) fires an initialization and iv) fires an expiration. The rv R_n is related to the scaling rule and will decide whether a new server will be activated. These processes are defined on $(\Omega_S, \mathcal{A}_S, \mathbb{P}_S)$;
- $(X^N(0))_N$, the process of the initial conditions, where each random variable $X^N(0)$ takes values in $\mathcal{S}_N$. This process is defined on $(\Omega_0, \mathcal{A}_0, \mathbb{P}_0)$.

Using that $\mathcal{N}_\phi(Nt)$ and $\mathcal{N}_{\phi N}(t)$ are equal in distribution and the well-known fact that thinnings of a Poisson process produce independent Poisson processes, each process $\{(X^N(t))_{t\in[0,T]}\}$, $N \geq 1$, can be constructed on the product space, say $(\Omega, \mathcal{A}, \mathbb{P})$.

Now, let t_n be the time of the n-th jump of $\mathcal{N}_\phi(Nt)$. Let also $X^N(t^-) := \lim_{s\uparrow t} X^N(s)$, $Y_i^N(t) := \sum_{j=0}^{i} X_{j,2}^N(t)$ for all $i \geq 0$, $Y_{-1}^N(t) = 0$ and $\mathbf{1}_A^x = 1$ if $x \in A$ and 0 otherwise. Note that in the main text, y_i is defined as a tail sum while here Y_i^N is a cumulative sum. The coordinates of $X^N(t)$ are then given by (24) for all $i \geq 1$. In (24), the H_i terms depend on the load balancing scheme used: within Power-of-d (servers are selected with replacement)

$$H_i(t_n^-) := \prod_{p=1}^{d} \mathbf{1}_{(Y_{i-1}^N(t_n^-),1]}^{A_n^p(1-X_{0,0}^N(t_n^-)-X_{0,1}^N(t_n^-))} - \prod_{p=1}^{d} \mathbf{1}_{(Y_i^N(t_n^-),1]}^{A_n^p(1-X_{0,0}^N(t_n^-)-X_{0,1}^N(t_n^-))} \in \{0,1\} \quad (25)$$

and within JBT-d

$$H_i(t_n^-) := \mathbf{1}_{(Y_{i-1}^N(t_n^-),Y_i^N(t_n^-)]}^{A_n^1(1-X_{0,0}^N(t_n^-)-X_{0,1}^N(t_n^-))} \mathbb{I}_{\{Y_d^N(t_n^-)=0\}} + \mathbf{1}_{(Y_{i-1}^N(t_n^-),Y_i^N(t_n^-)]}^{A_n^1 Y_d^N(t_n^-)} \mathbb{I}_{\{i\leq d\}} \mathbb{I}_{\{Y_d^N(t_n^-)>0\}} \in \{0,1\}. \quad (26)$$

These expressions follow by uniformization of $X^N(t)$. For instance, $X_{0,0}^N(t)$ has an upward jump of size $1/N$ at time t_n if the event occurring at that time is of type 4 (potential server expiration) and an idle-on server is actually selected at time t_n^- by the uniformized process. Analogously, $X_{0,0}^N(t)$ decreases by $1/N$ at time t_n if the event occurring at that time is of type 2, provided that at time t_n^- the cold servers pool is not empty and the scaling rule applies. Similar interpretations hold along the other coordinates of $X^N(t)$.

8.1.2 Tightness of sample paths and Lipschitz property

We now prove tightness of sample paths. The lemmas in this section are routine and equivalent to the lemmas in [11,Section 5.2].

Let us introduce the following formulas for quick reference.

Lemma 1. *Let $T > 0$. There exists $\mathcal{C} \subseteq \Omega$ such that $\mathbb{P}(\mathcal{C}) = 1$ and for all $\omega \in \mathcal{C}$:*

$$\lim_{N\to\infty} \sup_{t\in[0,T]} |\frac{1}{N}\mathcal{N}_\phi(Nt,\omega) - \phi t| = 0 \quad (27)$$

$$\lim_{N\to\infty} \sup_{t\in[0,T]} \Big|\frac{1}{N} \sum_{n=1}^{\mathcal{N}_\phi(Nt,\omega)} \mathbb{I}_{\{W_n(\omega)=k\}} - \mathbb{P}(W_1 = k)\,\phi\, t\Big| = 0 \quad (28)$$

for all $k \in \{0,\ldots,4\}$, and

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \prod_{p=1}^{d} \mathbf{1}_{(a_p,b_p]}^{c_p A_n^p} = \prod_{p=1}^{d} \frac{b_p - a_p}{c_p} \quad (29)$$

for all $a_p, b_p, c_p \in [0,1], c_p > 0, p = 1,\ldots,d$.

Proof. This lemma directly follows by applying the functional strong law of large numbers for the Poisson process (for (27)), the fact that thinnings of a Poisson process produce independent Poisson processes (for (28)) and the strong law of the large numbers (for (29)). □

We will work on a fixed ω that belongs to $\mathcal{C}$.

Let $x^0 \in [0,1]$, sequences $A_N \downarrow 0$ and $B_N \downarrow 0$ be given. Let also $D[0,T]$ denote the Skorokhod space endowed with the uniform metric $d(x,y) := \sup_{t\in[0,T]} |x(t) - y(t)|$, for all $x, y \in D[0,T]$. For $N \geq 1$, let also

$$\mathcal{E}_N(B_N, A_N, x^0) := \{x \in D[0,T] : |x(0) - x^0| \leq B_N, |x(a) - x(b)| \leq \phi|a-b| + A_N, \forall a, b \in [0,T]\}$$

$$\mathcal{E}_c(x^0) := \{x \in D[0,T] : x(0) = x^0, |x(a) - x(b)| \leq \phi|a-b|, \forall a, b \in [0,T]\}.$$

The next lemma says that the sample paths along any coordinate is approximately Lipschitz continuous. The proof is omitted because follows exactly the same standard arguments used in Lemma 5.2 of [11], which basically use the fact that the jumps of the Markov chain of interest are of the order of $1/N$ and that the evolution of such Markov chain on a given coordinate only depends on the evolution of such Markov chain on a finite number of other coordinates.

$$X_{0,0}^N(t) = X_{0,0}^N(0) + \frac{1}{N}\sum_{n=1}^{\mathcal{N}_\phi(Nt)} \left(\mathbb{I}_{\{W_n=4\}} \mathbf{1}^{E_n}_{(0,X_{0,2}^N(t_n^-)]} - \mathbb{I}_{\{W_n=2\}} \mathbb{I}_{\{X_{0,0}^N(t_n^-)>0\}} \mathbf{1}^{R_n}_{(0,g(X^N(t_n^-))]} \right) \tag{24a}$$

$$X_{0,1}^N(t) = X_{0,1}^N(0) + \frac{1}{N}\sum_{n=1}^{\mathcal{N}_\phi(Nt)} \left(\mathbb{I}_{\{W_n=2\}} \mathbb{I}_{\{X_{0,0}^N(t_n^-)>0\}} \mathbf{1}^{R_n}_{(0,g(X^N(t_n^-))]} - \mathbb{I}_{\{W_n=3\}} \mathbf{1}^{I_n}_{(0,X_{0,1}^N(t_n^-)]} \right) \tag{24b}$$

$$X_{0,2}^N(t) = X_{0,2}^N(0) + \frac{1}{N}\sum_{n=1}^{\mathcal{N}_\phi(Nt)} \left(\mathbb{I}_{\{W_n=1\}} \mathbf{1}^{D_n}_{[Y_0^N(t_n^-),Y_1^N(t_n^-)]} - \mathbb{I}_{\{W_n=0\}} H_0(t_n^-) + \mathbb{I}_{\{W_n=3\}} \mathbf{1}^{I_n}_{(0,X_{0,1}^N(t_n^-)]} - \mathbb{I}_{\{W_n=4\}} \mathbf{1}^{E_n}_{(0,X_{0,2}^N(t_n^-)]} \right) \tag{24c}$$

$$X_{i,2}^N(t) = X_{i,2}^N(0) + \frac{1}{N}\sum_{n=1}^{\mathcal{N}_\phi(Nt)} \left(\mathbb{I}_{\{W_n=0\}} \left(H_{i-1}(t_n^-) - H_i(t_n^-) \mathbb{I}_{\{i<B\}} \right) + \mathbb{I}_{\{W_n=1\}} \left(\mathbf{1}^{D_n}_{(Y_i^N(t_n^-),Y_{i+1}^N(t_n^-)]} - \mathbf{1}^{D_n}_{(Y_{i-1}^N(t_n^-),Y_i^N(t_n^-)]} \right) \right) \tag{24d}$$

Lemma 2. *Fix $T>0$, $\omega \in \mathcal{C}$, and some $x^0 \in \mathcal{S}_1$. Suppose that $\|X^N(\omega,0) - x^0\|_w \le \tilde{B}_N$, for some sequence $\tilde{B}_N \downarrow 0$. Then, there exists sequences $\left\{ B_N^{(i,j)} \downarrow 0 \right\}_{i,j}$ and $A_N \downarrow 0$ such that*

$$X_{i,j}^N(\omega,\cdot) \in \mathcal{E}_N(B_N^{(i,j)}, A_N, x^0), \quad \forall (i,j), \forall N. \tag{30}$$

The next proposition shows that any sequence of sample paths $X^N(\omega,t)$ contains a further subsequence that converges in $D^\infty[0,T]$, endowed with the metric $d^{\mathbb{Z}_+}(x,y) := \sup_{t\in[0,T]} \|x(t) - y(t)\|_w$, to a coordinate-wise Lipschitz continuous trajectory $x(t)$, as long as $\omega \in \mathcal{C}$. The proof is routine and omitted because it is a repetition of the argument used in the proof of Proposition 11 in [28] (equivalently, see also Proposition 5.3 in [11]).

Proposition 2. *Fix $T>0$, $\omega \in \mathcal{C}$, and some $x^0 \in \mathcal{S}_1$. Suppose that $\|X^N(\omega,0) - x^0\|_w \le \tilde{B}_N$, for some sequence $\tilde{B}_N \downarrow 0$. Then, every subsequence of $\{X^N(\omega,\cdot)\}_{N=1}^\infty$ contains a further subsequence $\{X^{N_k}(\omega,\cdot)\}_{k=1}^\infty$ such that*

$$\lim_{k\to\infty} d^{\mathbb{Z}_+}(X^{N_k}, x) = 0 \tag{31}$$

where $x(0) = x^0$ and $x_{i,j} \in \mathcal{E}_c(x^0)$, for all i and j.

Since Lipschitz continuity implies absolute continuity, we have obtained that limit points of $X^N(t)$ exist and are absolutely continuous. Since all sample paths of $X^N(t)$ take values in $\mathcal{S}$, these limit points must belong as well to $\mathcal{S}$ because $\mathcal{S}$ is a closed set. Therefore, to conclude the proof of Theorem 1 it remains to show that the derivative of $x_{i,j}(t)$ is as in Definition 1 for all i and j, provided that t is a regular time. This is done in the next subsection and will also prove that a fluid solution started in $x^{(0)} \in \mathcal{S}_1$ exists.

8.1.3 Limit trajectories are fluid solutions

Fix $\omega \in \mathcal{C}$ and let $\{X^{N_k}(\omega,t)\}_{k=1}^\infty$ be a subsequence that converges to $\overline{x}$ (by Proposition 2), i.e.

$$\lim_{k\to\infty} \sup_{t\in[0,T]} \|X^{N_k}(\omega,t) - \overline{x}(t)\|_w = 0. \tag{32}$$

In the remainder, we fix such $\omega \in \mathcal{C}$ such that (32) holds and for simplicity we drop the dependency on ω. Since $\overline{x}$ must be Lipschitz continuous (by Proposition 2), it is also absolutely continuous and to conclude the proof of Theorem 1, it remains to show that $\overline{x}(t)$ satisfies the conditions on the derivatives given in Definition 1 whenever $\overline{x}_{i,j}(t)$ is differentiable, for all i,j.

We say that t is a point of differentiability (of $\overline{x}$) if $x_{i,j}(t)$ is differentiable for all i,j.

We will (implicitly) use several times the following elementary lemma, which holds true because $\overline{x}$ is a nonnegative absolutely continuous function.

Lemma 3. *If $\overline{x}_{i,j}(t) = 0$ and t is a point of differentiability of $\overline{x}_{i,j}$, then $\dot{\overline{x}}_{i,j}(t) = 0$.*

Let $\epsilon > 0$. By Lemma 2, there exists a sequence $A_{N_k} \downarrow 0$ such that $X_{i,j}^{N_k}(\omega,u) \in [\overline{x}_{i,j}(t) - \epsilon\phi - A_{N_k}, \overline{x}_{i,j}(t) + \epsilon\phi + A_{N_k}]$, for all $u \in [t, t+\epsilon]$. Thus, for all k sufficiently large, $X_{i,j}^{N_k}(\omega,u) \in [\overline{x}_{i,j}(t) - 2\epsilon\phi, \overline{x}_{i,j}(t) + 2\epsilon\phi]$, for all $u \in [t, t+\epsilon]$. Thus, we have

$$|X_{i,j}^{N_k}(u) - \overline{x}_{i,j}(t)| \le 2\phi\epsilon, \quad \forall u \in [t, t+\epsilon] \tag{33}$$

for all k sufficiently large. In addition, using (33) and that g is Lipschitz, we obtain

$$|g(X^{N_k}(u)) - g(\overline{x}(u))| \le L\|X^{N_k}(u)) - \overline{x}(u)\|_w \tag{34a}$$

$$\le 2\phi\epsilon L \sqrt{\sum_{i,j} \frac{1}{2^{i+j}}} = 2\phi\epsilon L\sqrt{2}, \tag{34b}$$

for all $u \in [t, t+\epsilon]$, where L is the Lipschitz constant of the scaling rule g.

We will refer to the following lemma, which is a straightforward consequence of (33) and of the strong law of the large numbers. In points where the fluid drift function is continuous, it will provide an expression for terms related to job departures, server initializations/departures and, in some cases, dispatching decisions.

Lemma 4. *Fix $\omega \in \mathcal{C}$ and let (32) hold. Then,*

$$\lim_{\epsilon\downarrow 0} \lim_{k\to\infty} \frac{1}{\epsilon N_k} \sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \mathbb{I}_{\{W_n=1\}} \mathbf{1}^{D_n}_{(Y_{i-1}^N(t_n^-),Y_i^N(t_n^-)]} = \overline{x}_{i,2}(t)$$

$$\lim_{\epsilon\downarrow 0} \lim_{k\to\infty} \frac{1}{\epsilon N_k} \sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \mathbb{I}_{\{W_n=3\}} \mathbf{1}^{I_n}_{(0,X_{0,1}^N(t_n^-)]} = \beta\overline{x}_{0,1}(t)$$

$$\lim_{\epsilon\downarrow 0} \lim_{k\to\infty} \frac{1}{\epsilon N_k} \sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \mathbb{I}_{\{W_n=4\}} \mathbf{1}^{E_n}_{(0,X_{0,2}^N(t_n^-)]} = \gamma\overline{x}_{0,2}(t).$$

In addition,

$$\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}\mathbf{1}^{A_n^1 Y_d^N(t_n^-)}_{(Y_{i-1}^N(t_n^-),Y_i^N(t_n^-)]}\mathbb{I}_{\{i\le d\}} = \frac{\lambda\mathbb{I}_{\{i\le d\}}\overline{x}_i(t)}{\sum_{j=0}^d \overline{x}_{j,2}(t)}$$

provided that $\sum_{j=0}^d \overline{x}_{j,2} > 0$*, and*

$$\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}H_i(X^N(t_n^-)) = \lambda h_i(\overline{x})$$

provided that Power-of-d is used.

Proof. Given in Section 9. □

The next proposition proves the desired condition on the amount of fluid of cold and initializing servers.

Proposition 3. *Fix* $\omega\in\mathcal{C}$*, let* (32) *hold and assume that* t *is a point of differentiability. Then,*

$$\dot{\overline{x}}_{0,0} = \gamma\overline{x}_{0,2}(t) - \alpha\mathbb{I}_{\{\overline{x}_{0,0}(t)>0\}}g(\overline{x}(t)) - \gamma\overline{x}_{0,2}(t)\,\mathbb{I}_{\{\overline{x}_{0,0}(t)=0,\ \gamma\overline{x}_{0,2}(t)\le\alpha g(\overline{x}(t))\}} \tag{35}$$

$$\dot{\overline{x}}_{0,1} = \alpha g(\overline{x}(t))\mathbb{I}_{\{\overline{x}_{0,0}(t)>0\}} - \beta\overline{x}_{0,1}(t) + \gamma\overline{x}_{0,2}(t)\,\mathbb{I}_{\{\overline{x}_{0,0}(t)=0,\ \gamma\overline{x}_{0,2}(t)\le\alpha g(\overline{x}(t))\}}. \tag{36}$$

Proof. Assume that $\overline{x}_{0,0}(t)>0$ and let $\epsilon\in(0,\frac{\overline{x}_{0,0}(t)}{2\phi})$. Given that

$$t_n\in(t,t+\epsilon] \text{ if } n\in\{\mathcal{N}_\phi(N_k t)+1,\dots,\mathcal{N}_\phi(N_k(t+\epsilon))\}, \tag{37}$$

(33) implies that for all k sufficiently large, $|X_{0,0}^{N_k}(t_n^-)-\overline{x}_{0,0}(t)|\le 2\phi\epsilon<\overline{x}_{0,0}(t)$ and thus $X_{0,0}^{N_k}(t_n^-)>0$. We have shown that

$$\mathbb{I}_{\{X_{0,0}^{N_k}(t_n^-)>0\}} = 1,\quad \forall n\in\{\mathcal{N}_\phi(N_k t)+1,\dots,\mathcal{N}_\phi(N_k(t+\epsilon))\} \tag{38}$$

for all k sufficiently large. Using (24), Lemma 4 and (38), we have

$$\begin{aligned}\dot{\overline{x}}_{0,0}(t) &= \lim_{\epsilon\downarrow 0}\frac{1}{\epsilon}\lim_{k\to\infty}\left(X_{0,0}^{N_k}(t+\epsilon)-X_{0,0}^{N_k}(t)\right)\\ &= \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\Bigg(\mathbb{I}_{\{W_n=4\}}\mathbf{1}^{E_n}_{(0,X_{0,2}^{N_k}(t_n^-)]}\\ &\qquad - \mathbb{I}_{\{W_n=2\}}\mathbb{I}_{\{X_{0,0}^{N_k}(t_n^-)>0\}}\mathbf{1}^{R_n}_{(0,g(X^{N_k}(t_n^-))]}\Bigg) \qquad (39a)\\ &= \gamma\overline{x}_{0,2}(t) - \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=2}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbf{1}^{R_n}_{(0,g(X^{N_k}(t_n^-))]}. \qquad (39b)\end{aligned}$$

Since t is a point of differentiability, the double limit in the RHS of (39b) exists. Then, (34) implies that given $\epsilon>0$ small enough, $g(X^{N_k}(t_n^-))\in[g(\overline{x}(t))-2\phi\epsilon L\sqrt{2}, g(\overline{x}(t))+2\phi\epsilon L\sqrt{2}]$ for all k sufficiently large. Combining these bounds with Lemma 1 and letting $\epsilon\downarrow 0$ (as in the proof of Lemma 4), we obtain

$$\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=2}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbf{1}^{R_n}_{(0,g(X^{N_k}(t_n^-))]} = \alpha g(\overline{x}(t)). \tag{40}$$

Similarly, on coordinates (0,1), we obtain

$$\begin{aligned}\dot{\overline{x}}_{0,1}(t) &= \lim_{\epsilon\downarrow 0}\frac{1}{\epsilon}\lim_{k\to\infty}\left(X_{0,1}^{N_k}(t+\epsilon)-X_{0,1}^{N_k}(t)\right)\\ &= \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=2}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{X_{0,0}^{N_k}(t_n^-)>0\}}\mathbf{1}^{R_n}_{(0,g(X^{N_k}(t_n^-))]}\\ &\quad - \frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=3}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=3\}}\mathbf{1}^{I_n}_{(0,X_{0,1}^{N_k}(t_n^-)]}\\ &= \alpha g(\overline{x}) - \beta\overline{x}_{0,1}(t).\end{aligned}$$

Now, let us assume that $\overline{x}_{0,0}(t)=0$. First, we notice that

$$\dot{\overline{x}}_{0,0}(t) = \gamma\overline{x}_{0,2}(t) - \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=2, X_{0,0}^N(t_n^-)>0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\frac{\mathbf{1}^{R_n}_{(0,g(X^{N_k}(t_n^-))]}}{\epsilon N_k} \tag{41}$$

$$\begin{aligned}&\ge \gamma\overline{x}_{0,2}(t) - \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=2}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbf{1}^{R_n}_{(0,g(X^{N_k}(t_n^-))]}\\ &= \gamma\overline{x}_{0,2}(t) - \alpha g(\overline{x}) \qquad (42)\end{aligned}$$

where the first equality follows by (39a) and Lemma 4, and the last equality follows by (40). Thus, if $\overline{x}_{0,0}(t)=0$ and $\gamma\overline{x}_{0,2}(t)>\alpha g(\overline{x})$, then by the previous inequality $\dot{\overline{x}}_{0,0}(t)>0$, which is not possible because if t is a point of differentiability and $\overline{x}_{0,0}(t)=0$ then necessarily $\dot{\overline{x}}_{0,0}(t)=0$ as $\overline{x}_{0,0}$ is a non-negative absolutely continuous function. Thus, in a point of differentiability t where $\overline{x}_{0,0}(t)=0$, we must have $\gamma\overline{x}_{0,2}(t)\le\alpha g(\overline{x})$. and, necessarily, $\dot{\overline{x}}_{0,0}(t)=0$. In this case, (41) gives

$$\gamma\overline{x}_{0,2}(t) = \lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=2\}} \times\mathbb{I}_{\{X_{0,0}^N(t_n^-)>0\}}\mathbf{1}^{R_n}_{(0,g(X^N(t_n^-))]}. \tag{43}$$

This term is interpreted as the amount of idle-on servers that become cold but instantly turn initializing. Substituting (43) in the previous equalities within the conditions $\gamma\overline{x}_{0,2}(t)\le\alpha g(\overline{x})$ and $\overline{x}_{0,0}(t)=0$, we obtain (35) and (36). □

On the coordinates associated to warm servers, it remains to prove that

$$\dot{\overline{x}}_{0,2}(t) = \overline{x}_{1,2}(t) - \lambda h_0(\overline{x}(t)) + \beta\overline{x}_{0,1}(t) - \gamma\overline{x}_{0,2}(t) \tag{44}$$

$$\dot{\overline{x}}_{i,2}(t) = \overline{x}_{i+1,2}(t)\mathbb{I}_{\{i<B\}} - \overline{x}_{i,2}(t) + \lambda(h_{i-1}(\overline{x}(t)) - h_i(\overline{x}(t))\mathbb{I}_{\{i<B\}}),\ i\ge 1, \tag{45}$$

whenever t is a point of differentiability of $\overline{x}$. Let

$$\mathcal{H}_i(t) := \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}H_i(t_n^-) \geq 0, \tag{46}$$

which is interpreted as the rate at which jobs are assigned to warm servers with exactly i jobs. Using Lemma 4 and (24), we have

$$\dot{\overline{x}}_{0,2}(t) = \lim_{\epsilon\downarrow 0}\frac{1}{\epsilon}\lim_{k\to\infty}\left(X_{0,2}^{N_k}(t+\epsilon) - X_{0,2}^{N_k}(t)\right) \tag{47a}$$

$$= \overline{x}_{1,2}(t) - \mathcal{H}_0(t) + \beta\overline{x}_{0,1}(t) - \gamma\overline{x}_{0,2}(t) \tag{47b}$$

$$\dot{\overline{x}}_{i,2}(t) = \lim_{\epsilon\downarrow 0}\frac{1}{\epsilon}\lim_{k\to\infty}\left(X_{i,2}^{N_k}(t+\epsilon) - X_{i,2}^{N_k}(t)\right) \tag{47c}$$

$$= \overline{x}_{i+1,2}(t)\mathbb{I}_{\{i<B\}} - \overline{x}_{i,2}(t) + \mathcal{H}_{i-1}(t) - \mathcal{H}_i(t)\mathbb{I}_{\{i<B\}}. \tag{47d}$$

In the following, we need to show that $\mathcal{H}_i(t) = h_i(\overline{x}(t))$ where the h_i's are as in Definition 1. We treat the cases of Power-of-d and JBT-d separately.

Lemma 5. *Assume that Power-of-d is applied. Then,* (44) *and* (45) *hold true.*

Proof. If $\overline{x}_{0,0} + \overline{x}_{0,1} < 1$, then the structure of the H_i's in (25) and Lemma 4 immediately give (44) and (45). Now, let us assume that $\overline{x}_{0,0} + \overline{x}_{0,1} = 1$. On coordinate (0,2), in a point of differentiability we necessarily have $\dot{\overline{x}}_{0,2} = 0$. Using Lemma 4 and (24), we obtain

$$\dot{\overline{x}}_{0,2}(t) = \lim_{\epsilon\downarrow 0}\frac{1}{\epsilon}\lim_{k\to\infty}\left(X_{0,2}^{N_k}(t+\epsilon) - X_{0,2}^{N_k}(t)\right) \tag{48}$$

$$= \beta\overline{x}_{0,1}(t) - \mathcal{H}_0(t) = 0. \tag{49}$$

Similarly, on coordinate $(1,2)$, Lemma 4 and (48) imply that in a point of differentiability we have $\dot{\overline{x}}_{1,2}(t) = \mathcal{H}_0(t) - \mathcal{H}_1(t) = 0$ and thus $\mathcal{H}_1(t) = \mathcal{H}_0(t) = \beta\overline{x}_{0,1}(t)$. Then, on coordinate $(i,2)$ by induction we obtain $\mathcal{H}_i(t) = \mathcal{H}_{i-1}(t) = \beta\overline{x}_{0,1}(t)$. On the other hand, we also have

$$\dot{\overline{x}}_{0,2}(t) = \beta\overline{x}_{0,1}(t) - \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))} H_0(t_n^-)$$

$$\geq \beta\overline{x}_{0,1}(t) - \lambda$$

where in the last inequality we have just used that $H_0(t_n^-) \leq 1$. Thus, if $\beta\overline{x}_{0,1}(t) > \lambda$, we get a contradiction and t can not be a point of differentiability. Substituting $\mathcal{H}_i(t) = \beta\overline{x}_{0,1}(t)$ in (47) when $\beta\overline{x}_{0,1}(t) \leq \lambda$, we obtain (44)-(45). □

The case of JBT-d is more delicate than Power-of-d because of the discontinuous structure of the H_i's when $\sum_{j=0}^d X_{j,2}^N(t_n^-) = 0$, see (26). In addition to a more involved argument than the one presented in the proof of Lemma 5, which we will develop in Lemma 7 below, we need the following lemma, which we will use to determine an expression for $\mathcal{H}_i$ when $\sum_{j=0}^d \overline{x}_{j,2}(t) = 0$.

Lemma 6. *Assume that $\overline{x}(t)$ satisfies $\overline{x}_{0,0}(t) + \overline{x}_{0,1}(t) < 1$. Then,* (50) *holds true for all i.*

Proof. Given in Section 9. □

The following lemma proves the desired property in the case of JBT-d.

Lemma 7. *Assume that JBT-d is applied. Then,* (44) *and* (45) *hold true.*

Proof. We analyze $\mathcal{H}_i$ and the resulting expression will be substituted in (47). This will give (44) and (45).

First, if $\overline{x}_{0,0} + \overline{x}_{0,1} = 1$, the argument in the proof of Lemma 5 gives i) $\mathcal{H}_i(t) = \beta\overline{x}_{0,1}(t)$ when $\beta\overline{x}_{0,1}(t) \leq \lambda$ and ii) t not a point of differentiability when $\beta\overline{x}_{0,1}(t) > \lambda$. This gives (44) and (45) (when $\overline{x}_{0,0} + \overline{x}_{0,1} = 1$) and in the remainder we assume that $\overline{x}_{0,0} + \overline{x}_{0,1} < 1$.

Let us now assume that $\sum_{j=0}^d \overline{x}_{j,2}(t) > 0$ and let $\epsilon \in (0, \frac{\sum_{j=0}^d \overline{x}_{j,2}(t)}{2\phi(d+1)})$. Since $t_n \in (t, t+\epsilon]$ whenever $n \in \{\mathcal{N}_\phi(N_k t)+1, \ldots, \mathcal{N}_\phi(N_k(t+\epsilon))\}$, (33) and the triangular inequality imply that for all k sufficiently large $|\sum_{j=0}^d X_{j,2}^{N_k}(t_n) - \overline{x}_{j,2}(t)| \leq 2(d+1)\phi\epsilon < \sum_{j=0}^d \overline{x}_{j,2}(t)$ and thus $\sum_{j=0}^d X_{j,2}^{N_k}(t_n) > 0$. We have shown that

$$\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}} = 1, \forall n \in \{\mathcal{N}_\phi(N_k t)+1, \ldots, \mathcal{N}_\phi(N_k(t+\epsilon))\} \tag{51}$$

for all k sufficiently large, given $\epsilon > 0$ sufficiently small. Substituting (51) in (26) and applying Lemma 4, we obtain (44) and (45) (under the conditions $\overline{x}_{0,0} + \overline{x}_{0,1} < 1$ and $\sum_{j=0}^d \overline{x}_{j,2}(t) > 0$).

It remains to understand the terms $\mathcal{H}_i$ in the case where $\sum_{j=0}^d \overline{x}_{j,2}(t) = 0$, which we assume in the remainder of the proof.

Suppose that t is a point of differentiability. Then, by applying Lemma 4 to $X_{0,2}^N$ (see (24)), we obtain

$$\dot{\overline{x}}_{0,2}(t) = \lim_{\epsilon\downarrow 0}\frac{1}{\epsilon}\lim_{k\to\infty} X_{0,2}^{N_k}(t+\epsilon) - X_{0,2}^{N_k}(t)$$

$$= \overline{x}_{1,2}(t)\mathbb{I}_{\{d=0\}} + \beta\overline{x}_{0,1}(t) - \mathcal{H}_0(t), \tag{52}$$

and given that necessarily $\dot{\overline{x}}_{0,2}(t) = 0$, we obtain

$$\mathcal{H}_0(t) = \overline{x}_{1,2}(t)\mathbb{I}_{\{d=0\}} + \beta\overline{x}_{0,1}(t). \tag{53}$$

Similarly, on coordinate $(i,2)$, with $0 < i \leq d$, we obtain

$$\dot{\overline{x}}_{i,2}(t) = \overline{x}_{i+1,2}(t)\mathbb{I}_{\{i=d\}} + \mathcal{H}_{i-1}(t) - \mathcal{H}_i(t) = 0. \tag{54}$$

By induction, this gives $\mathcal{H}_i(t) = \mathcal{H}_0(t) = \beta\overline{x}_{0,1}(t)$ for all $i < d$ and $\mathcal{H}_d(t) = \beta\overline{x}_{0,1}(t) + \overline{x}_{d+1,2}(t)$, that is,

$$\mathcal{H}_i(t) = \beta\overline{x}_{0,1}(t) + \overline{x}_{d+1,2}(t)\mathbb{I}_{\{i=d\}}, \qquad i \leq d. \tag{55}$$

We have proven (55) under the hypothesis that t was a point of differentiability but now we show that $\overline{x}(t)$ is not differentiable if $\lambda < \overline{x}_{d+1,2} + (d+1)\beta\overline{x}_{0,1}$. Towards this purpose, first we notice that

$$\sum_{i=0}^d \mathcal{H}_i(t) = \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}\sum_{i=0}^d H_i(t_n^-)$$

$$= \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^N(t_n^-)>0\}}$$

$$\leq \lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}} = \lambda.$$

Here, the first equality follows because the limits $\mathcal{H}_i(t)$ exist and the second inequality follows by the fact that (recall the definition of $\mathcal{H}_i$ in (46))

$$\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}\mathbf{1}_{(Y_{i-1}^{N_k}(t_n^-),Y_i^{N_k}(t_n^-)]}^{A_n^1(1-X_{0,0}^{N_k}(t_n^-)-X_{0,1}^{N_k}(t_n^-))}\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}}$$
$$=\frac{\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{n=\mathcal{N}_\phi(N_k t)+1}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{W_n=0\}}\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}}. \tag{50}$$

$$\sum_{i=0}^{d} H_i(t_n^-)=\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N}(t_n^-)>0\}}$$
$$+\mathbf{1}_{(0,Y_d^N(t_n^-)]}^{A_n^1(1-X_{0,0}^{N}(t_n^-)-X_{0,1}^{N}(t_n^-))}\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N}(t_n^-)=0\}} \tag{56}$$

and by Lemma 4 because $\sum_{j=0}^d \overline{x}_{j,2}(t)=0$. Then, using (55), we necessarily have

$$\sum_{i=0}^{d}\mathcal{H}_i(t)=\overline{x}_{d+1,2}(t)+(d+1)\beta\overline{x}_{0,1}(t)\le\lambda \tag{57}$$

and, given that necessarily $\mathcal{H}_i\ge 0$, we conclude that t can not be a point of differentiability whenever (57) does not hold true.

Now, we investigate $\mathcal{H}_i$ when $i>d$ and assuming that (57) holds as otherwise $\overline{x}(t)$ would not be differentiable. We observe that

$$\mathcal{H}_i(t)=\frac{\lambda\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}$$
$$-\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbf{1}_{(Y_{i-1}^{N}(t_n^-),Y_i^{N}(t_n^-)]}^{A_n^1(1-X_{0,0}^{N}(t_n^-)-X_{0,1}^{N}(t_n^-))}$$
$$\times\,\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N}(t_n^-)>0\}}$$
$$=\frac{\lambda\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}-\frac{\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}$$
$$\times\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))}\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N}(t_n^-)>0\}}$$
$$=\frac{\lambda\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}-\frac{\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}\sum_{i=0}^{d}\mathcal{H}_i(t)$$
$$=\overline{x}_{i,2}(t)\frac{\lambda-\overline{x}_{d+1,2}(t)-(d+1)\beta\overline{x}_{0,1}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)}.$$

In the first equality, we have used (26) and applied Lemma 4 to the definition of $\mathcal{H}_i$ in (46); In the second, we have applied Lemma 6. In the third, we have used (56) and that

$$0\le\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\mathbf{1}_{(0,Y_d^N(t_n^-)]}^{A_n^1(1-X_{0,0}^{N}(t_n^-)-X_{0,1}^{N}(t_n^-))}\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N}(t_n^-)=0\}}$$
$$\le\lim_{\epsilon\downarrow 0}\lim_{k\to\infty}\frac{1}{\epsilon N_k}\mathbf{1}_{(0,Y_d^N(t_n^-)]}^{A_n^1(1-X_{0,0}^{N}(t_n^-)-X_{0,1}^{N}(t_n^-))}=0$$

with the last inequality following by Lemma 4 as $\sum_{j=0}^d\overline{x}_{j,2}(t)=0$; in the fourth, we have substituted (55). This concludes the proof. □

Thus, we have shown that $\overline{x}$ is a fluid solution.

8.2 Proof of Theorem 2: fixed points

We now prove Theorem 2. By definition, $x\in\mathcal{S}_1$ is a fixed point if and only if

$$0=\gamma x_{0,2}-\alpha g\mathbb{I}_{\{x_{0,0}>0\}}-\gamma x_{0,2}\,\mathbb{I}_{\{x_{0,0}=0,\,\gamma x_{0,2}\le\alpha g\}} \tag{58a}$$
$$0=\alpha g\mathbb{I}_{\{x_{0,0}>0\}}-\beta x_{0,1}+\gamma x_{0,2}\,\mathbb{I}_{\{x_{0,0}=0,\,\gamma x_{0,2}\le\alpha g\}} \tag{58b}$$
$$0=x_{1,2}-h_0(x)+\beta x_{0,1}-\gamma x_{0,2} \tag{58c}$$
$$0=x_{i+1,2}-x_{i,2}+h_{i-1}(x)-h_i(x),\quad i\ge 1. \tag{58d}$$

Together with $\|x\|=1$, we now show that these conditions coincide with (7)-(9).

If i) $x_{0,0}=0$ and $\gamma x_{0,2}>\alpha g$, or if ii) $x_{0,0}+x_{0,1}=1$, then we easily observe that x cannot be a fixed point. Therefore, in the following we exclude these conditions. Now, summing (58a) and (58b), we obtain

$$\beta x_{0,1}=\gamma x_{0,2} \tag{59}$$

which gives (7b). Then, (7c) and (7d) directly follow from (58a) and (58b).

Substituting (59) in (58c), the conditions (58c)-(58d) become

$$0=x_{1,2}-h_0(x) \tag{60a}$$
$$0=x_{i+1,2}-x_{i,2}+h_{i-1}(x)-h_i(x),\quad i\ge 1, \tag{60b}$$

and taking summations

$$x_{i,2}=h_{i-1}(x),\quad i\ge 1. \tag{61}$$

The equations in (60) are interpreted as the mean-field fixed-point equations associated to Power-of-d and JBT-d when the number of servers is Ny_0 instead of N; we recall that $y_i=\sum_{i\ge 0}x_{i,2}$ is the proportion of warm servers with at least i jobs. Within Power-of-d, one can directly check that for any given $x_{0,2}$, (61) holds if and only if $x_{i,2}$ is given by (8) and that, after a substitution, this gives $\sum_{i\ge 1}x_{i,2}=\lambda$ so that (7a) must hold true. The following lemma, given in Section 9, handles the more delicate case of JBT-d.

Lemma 8. *Within JBT-d, for any given $x_{0,2}$,* (60) *holds if and only if $x_{i,2}$ satisfies (9a)-(9e). In addition, (7a) holds true.*

Therefore, the conditions in (58) are equivalent to (7)-(9). This proves the first statement of Theorem 2.

Now, under Assumption 3, $x_{i,2}$ is a function of $x_{0,2}$, for all $i\ge 1$, and we write $x_{i,2}$ as a shorthand notation for $x_{i,2}(x_{0,2})$. Using (59), we can then focus only on the following conditions:

$$x_{0,0}+\left(\frac{\gamma}{\beta}+1\right)x_{0,2}=1-\lambda \tag{62a}$$
$$\gamma x_{0,2}\le\alpha g,\quad \text{if } x_{0,0}=0 \tag{62b}$$
$$\gamma x_{0,2}=\alpha g,\quad \text{if } x_{0,0}>0. \tag{62c}$$

Here, we notice that $(x^\circ_{0,0}, x^\circ_{0,2}) = \big(0, \frac{\beta}{\beta+\gamma}(1-\lambda)\big)$ uniquely solves (62) if

$$\left(\frac{1}{\beta}+\frac{1}{\gamma}\right)\alpha g(x^\circ) \ge 1-\lambda \tag{63}$$

where x° is uniquely determined by $(x^\circ_{0,0}, x^\circ_{0,2})$. So, let us assume that (63) does *not* hold true. Then, if a point $(x_{0,0}, x_{0,2}) \in [0,1)^2$ that solves (62) exists, then necessarily $x_{0,0} > 0$ as otherwise $x_{0,2} = x^\circ_{0,2}$ (by (62a)) and (63) would hold, contradicting the hypothesis. This proves the second part of Theorem 2.

8.3 Proof of Theorem 3: fluid optimality

The non-linear structure taken by the h_i's when $x_{0,2} = 0$, see (6), complicates the analysis and the identification of a Lyapunov function. For this reason, our strategy is based on a divide-and-conquer approach. This will actually provide insights about the dynamics followed by fluid solutions. For simplicity, we provide a proof assuming that $B < \infty$, which is essentially equivalent to assume that $x_{i,2}(0) = 0$ for all i large enough; this is not critical as $x^\star_{i,2} = 0$ for all $i \ge 2$.

Let $\overline{Q}(x) := \sum_{i=1}^{B} i x_{i,2}$, i.e., the overall number of jobs in the system in state x. The following lemma gives a property on the time derivative of $\overline{Q}(x(t))$.

Lemma 9. *Let $x(t)$ be a fluid solution induced by JIQ such that $x(0) \in \mathcal{S}_1$ and $B < \infty$. If t is a point of differentiability, then*

$$\dot{\overline{Q}}(x(t)) = \lambda - y_1(t). \tag{64}$$

Proof. First, we notice that

$$\dot{\overline{Q}}(x(t)) = \sum_{i\ge 1} i\dot{x}_{i,2}(t) = -y_1 + \sum_{i=0}^{B-1} h_i(x(t)) \tag{65}$$

where the second equality follows by applying Definition 1. Now, we treat the cases $x_{0,2}(t) > 0$ and $x_{0,2}(t) = 0$ separately. Suppose that $x_{0,2}(t) > 0$. Then, $h_i(x(t)) = \lambda \mathbb{I}_{\{i=0\}}$ (by (6)) and substituting in (65) we immediately get $\dot{\overline{Q}}(x(t)) = \lambda - y_1(t)$ as desired. Thus, suppose in the remainder that $x_{0,2}(t) = 0$. Now, assume that $y_0(t) > 0$. Then, using again (6),

$$\begin{aligned}\dot{\overline{Q}}(x(t)) =& -y_1 + (\beta x_{0,1} + x_{1,2})\,\mathbb{I}_{\{x_{1,2}+\beta x_{0,1}\le\lambda\}}\\ &+ \frac{y_1}{y_0}(\lambda - x_{1,2} - \beta x_{0,1})^+\\ =& -y_1 + (\beta x_{0,1} + x_{1,2})\,\mathbb{I}_{\{x_{1,2}+\beta x_{0,1}\le\lambda\}}\\ &+ (\lambda - x_{1,2} - \beta x_{0,1})^+\end{aligned}$$

and the statement follows immediately if $x_{1,2}(t) + \beta x_{0,1}(t) \le \lambda$. On the other hand, if $x_{1,2}(t) + \beta x_{0,1}(t) > \lambda$, then, since $x_{0,2}(t) = 0$ and t is supposed to be a point of differentiability, we get (by (4c)) the contradiction that $0 = \dot{x}_{0,2}(t) = x_{1,2}(t) - h_0(x(t)) + \beta x_{0,1}(t) - \gamma x_{0,2}(t) = x_{1,2}(t) + \beta x_{0,1}(t) > \lambda$; the first equality holds because $x_{0,2}(t)$ is a non-negative absolutely continuous function. This shows that t cannot be a point of differentiability. Finally, if $y_0(t) = 0$, then the differentiability at t and the normalizing condition $\|x\| = 1$ give $0 = \dot{y}_0(t) = -\dot{x}_{0,0}(t) - \dot{x}_{0.1}(t) = \beta x_{0,1}$ and thus $x_{0,0}(t) = 1$. Assumption 2 requires that $g(x) > 0$ when $x_{0,0} = 1$, so (4a) implies that $\dot{x}_{0,0} < 0$. This contradicts that t is a point of differentiability because $x_{0,0}(t)$ is uniformly bounded by one and absolutely continuous. □

We now prove Theorem 3 by showing that $\|x(t) - x^\star\| \to 0$ in each of the following complete and mutually exclusive cases. For each case, we show that $x(t)$ follows a unique trajectory that stays in $\mathcal{S}_1$.

Case i). Suppose that $x_{0,2}(t) = 0$ for all $t \ge 0$. This rules out the possibility that $x_{0,0}(t)$ stays on zero for all t large enough because (4a) and (4b), together with the normalizing condition $\|x\| = 1$, would imply that $y_1(t) \to 1$ as $t \to \infty$, and in this case Lemma 9 yields the contradiction that $\overline{Q}(x(t))$ is eventually negative. Thus, without loss of generality, let us assume that $x_{0,0}(0) > 0$. Then, using (4), $x(t)$ satisfies

$$\dot{x}_{0,0} = -\alpha g(x) \tag{66a}$$

$$\dot{x}_{0,1} = \alpha g(x) - \beta x_{0,1} \tag{66b}$$

$$\dot{x}_{0,2} = 0, \quad x_{1,2} + \beta x_{0,1} \le \lambda. \tag{66c}$$

Note that $\lim_{t\to\infty} x_{0,0}(t)$ exists, say $x_{0,0}(\infty)$, because $\dot{x}_{0,0}(t) \le 0$ and $x_{0,0}(t)$ is uniformly bounded. Thus, as $t \to \infty$, $\dot{x}_{0,0}(t) = -\alpha g(x(t)) \to 0$. Given the assumptions on g, $(\lambda - x_{0,1}(t) - \beta x_{1,2}(t))^+ \to 0$ and since $x_{1,2}(t) + \beta x_{0,1}(t) \le \lambda$ for all t, by (66c), we obtain that $x_{1,2}(t) + \beta x_{0,1}(t) \to \lambda$. Then, (66b) and $g(x(t)) \to 0$ imply that $x_{0,1}(t) \to 0$ and thus $x_{1,2}(t) \to \lambda$. In turn, (4d) gives $x_{i,2}(t) \to 0$ for all $i \ge 2$, and the normalizing condition $\|x\| = 1$ implies that necessarily $x_{0,0}(t) \to 1 - \lambda$. Thus, $\|x(t) - x^\star\| \to 0$.

Case ii). Suppose that $x_{0,2}(t) > 0$ for all t. Then, $x(t)$ satisfies the following conditions (using Definition 1)

$$\dot{x}_{0,0} = \gamma x_{0,2} - \alpha g \mathbb{I}_{\{x_{0,0}>0\}} - \gamma x_{0,2}\, \mathbb{I}_{\{x_{0,0}=0,\, \gamma x_{0,2}\le\alpha g\}} \tag{67a}$$

$$\dot{x}_{0,1} = \alpha g \mathbb{I}_{\{x_{0,0}>0\}} - \beta x_{0,1} + \gamma x_{0,2}\, \mathbb{I}_{\{x_{0,0}=0,\, \gamma x_{0,2}\le\alpha g\}} \tag{67b}$$

$$\dot{x}_{0,2} = x_{1,2} - \lambda + \beta x_{0,1} - \gamma x_{0,2} \tag{67c}$$

$$\dot{x}_{1,2} = x_{2,2} - x_{1,2} + \lambda \tag{67d}$$

$$\dot{x}_{i,2} = x_{i+1,2}\mathbb{I}_{\{i<B\}} - x_{i,2}, \quad i \ge 2. \tag{67e}$$

The ODE system (67d)-(67e) is an autonomous linear ODE system with constant coefficients and, developing the matrix-exponential general solution of such ODE system, for all $i \ge 1$ we obtain

$$x_{i,2}(t) = \lambda \mathbb{I}_{\{i=1\}} + e^{-t} \sum_{k=i}^{B} \frac{t^{k-i}}{(k-i)!}(x_{k,2}(0) - \lambda \mathbb{I}_{\{k=1\}}) \tag{68}$$

and thus $x_{i,2}(t) \to \lambda \mathbb{I}_{\{i=1\}}$ as $t \to \infty$. In turn, $\lim_{t\to\infty}(\lambda - x_{1,2}(t) - \beta x_{0,1}(t))^+ = \lim_{t\to\infty}(-\beta x_{0,1}(t))^+ = 0$ and therefore $g(x(t)) \to 0$. Since $g(x(t)) \to 0$, $x_{0,1}(t) \to 0$ necessarily by (67b), and using this in (67c) we obtain $x_{0,2}(t) \to 0$. Since $\|x\| = 1$, necessarily $x_{0,0}(t) \to 1 - \lambda$ and we have shown that $\|x(t) - x^\star\| \to 0$.

Case iii). If the conditions in cases *i)* and *ii)* are not met, then there exists t_0, t_1, with $t_0 \le t_1 < \infty$, and $\delta > 0$ such that

1) $x_{0,2}(t) = 0$ for all $t \in [t_0, t_1]$

2) $x_{0,2}(t) > 0$ and $\dot{x}_{0,2}(t) < 0$ for all $t \in [t_0 - \delta, t_0)$, and
3) $x_{0,2}(t) > 0$ and $\dot{x}_{0,2}(t) > 0$ for all $t \in (t_1, t_1 + \delta]$.

On $[t_0 - \delta, t_0)$, $h_0(x(t)) = \lambda$ (by (6)) and using (4c), we obtain $\dot{x}_{0,2}(t) = x_{1,2}(t) - \lambda + \beta x_{0,1}(t) - \gamma x_{0,2}(t) < 0$ and thus by continuity

$$\begin{aligned} 0 \geq & \lim_{t \uparrow t_0} x_{1,2}(t) - \lambda + \beta x_{0,1}(t) - \gamma x_{0,2}(t) \\ = & x_{1,2}(t_0) - \lambda + \beta x_{0,1}(t_0). \end{aligned} \tag{69}$$

Since $x_{0,2}(t_0) = 0$ on $[t_0, t_1]$, (4c) implies that (69) holds as well on $[t_0, t_1]$. On $(t_1, t_1 + \delta]$, $h_0(x(t)) = \lambda$ (by (6)) and using again (4c), we obtain

$$\begin{aligned} 0 < \dot{x}_{0,2}(t) &= x_{1,2}(t) - h_0(x(t)) + \beta x_{0,1}(t) - \gamma x_{0,2}(t) \\ &< x_{1,2}(t) - \lambda + \beta x_{0,1}(t) \end{aligned}$$

and therefore $g(x(t)) = 0$. By continuity of fluid solutions, $x_{1,2}(t_1) + \beta x_{0,1}(t_1) = \lambda$. In addition, on $(t_1, t_1 + \delta]$, $x(t)$ is uniquely defined by

$$\dot{x}_{0,0} = \gamma x_{0,2} \tag{70a}$$

$$\dot{x}_{0,1} = -\beta x_{0,1} \tag{70b}$$

$$\dot{x}_{0,2} = x_{1,2} - \lambda + \beta x_{0,1} - \gamma x_{0,2} \tag{70c}$$

$$\dot{x}_{1,2} = x_{2,2} - x_{1,2} + \lambda \tag{70d}$$

$$\dot{x}_{i,2} = x_{i+1,2} \mathbb{I}_{\{i<B\}} - x_{i,2}, \quad i \geq 2, \tag{70e}$$

and we also know that $\dot{x}_{0,2}(t) > 0$. As long as *a)* $g(x(t)) = 0$ and *b)* $x_{0,2}(t) > 0$, on $[t_1, \infty)$ the fluid solution under investigation $x(t)$ is indeed uniquely given by the trajectory induced by (70) on $[t_1, \infty)$. In the remainder, we show that both *a)* and *b)* hold true for all t. This will conclude the proof because $x^\star$ is the unique fixed point of (70) and because (70) is a linear ODE system with constant coefficients. For simplicity of notation, let us shift time and assume that $t_1 = 0$. Now, since $x_{0,1}(t) = x_{0,1}(0)e^{-\beta t}$ (by (70b)) and since $x_{1,2}(t)$ takes the form given in (68), substituting in (70c) we obtain

$$\begin{aligned} \dot{x}_{0,2}(t) &= \beta x_{0,1}(0)e^{-\beta t} - \gamma x_{0,2}(t) + e^{-t}(x_{1,2}(0) - \lambda) \\ &\quad + e^{-t} \sum_{k=2}^{B} \frac{t^{k-1}}{(k-1)!} x_{k,2}(0) \\ &= \beta x_{0,1}(0)e^{-\beta t} - \gamma x_{0,2}(t) - \beta x_{0,1}(0)e^{-t} \\ &\quad + e^{-t} \sum_{k=2}^{B} \frac{t^{k-1}}{(k-1)!} x_{k,2}(0) \\ &\geq \beta x_{0,1}(0)(e^{-\beta t} - e^{-t}) - \gamma x_{0,2}(t). \end{aligned}$$

Thus, $x_{0,2}(t) \geq z(t)$ where $z(t)$ is uniquely defined by $\dot{z}(t) = \beta x_{0,1}(0)(e^{-\beta t} - e^{-t}) - \gamma z(t)$ with $z(0) = x_{0,2}(0)$. The solution of this differential equation is

$$z(t) = \beta x_{0,1}(0)e^{-\gamma t} \left(\frac{1 - e^{-t(\beta - \gamma)}}{\beta - \gamma} - \frac{1 - e^{-t(1-\gamma)}}{1 - \gamma} \right)$$

and now we notice that $z(t) > 0$ if $\beta > 1$, for all t. This proves property *b)*. To prove property *a)*, we use again (68) and $x_{1,2}(t_1) + \beta x_{0,1}(t_1) = \lambda$ to obtain

$$\begin{aligned} x_{1,2}(t) + \beta x_{0,1}(t) - \lambda = \beta x_{0,1}(0) \left(e^{-\beta t} - e^{-t} \right) \\ + e^{-t} \sum_{k=2}^{B} \frac{t^{k-1}}{(k-1)!} x_{k,2}(0) > 0, \end{aligned} \tag{71}$$

where the last inequality follows because $\beta < 1$. Given (18), (71) implies $g(x(t)) = 0$.

9 Proofs of technical lemmas

9.1 Proof of Lemma 4

We give a proof for the first limit because the argument used for the others is identical.

Since $t_n \in (t, t + \epsilon]$ whenever $n \in \{\mathcal{N}_\phi(N_k t) + 1, \ldots, \mathcal{N}_\phi(N_k(t + \epsilon))\}$, (33) implies that for all k sufficiently large $|Y_i^{N_k}(t_n) - \sum_{j=0}^{i} \overline{x}_{j,2}(t)| \leq C\epsilon$, for some constant C, i.e.,

$$\begin{aligned} \mathbf{1}^{D_n}_{(\sum_{j=0}^{i} \overline{x}_{j,2}(t) + C\epsilon, \sum_{j=0}^{i} \overline{x}_{j,2}(t) - C\epsilon]} &\leq \mathbf{1}^{D_n}_{(Y_{i-1}^N(t_n^-), Y_i^N(t_n^-)]} \\ &\leq \mathbf{1}^{D_n}_{(\sum_{j=0}^{i} \overline{x}_{j,2}(t) - C\epsilon, \sum_{j=0}^{i} \overline{x}_{j,2}(t) + C\epsilon]} \end{aligned} \tag{72}$$

Let Γ denote the LHS of the first equation in Lemma 4. Applying Lemma 1, we obtain

$$\begin{aligned} \Gamma \leq & \lim_{\epsilon \downarrow 0} \lim_{k \to \infty} \sum_{\substack{n = \mathcal{N}_\phi(N_k t) + 1: \\ W_n = 1}}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \frac{\mathbf{1}^{D_n}_{(\sum_{j=0}^{i} \overline{x}_{j,2}(t) - C\epsilon, \sum_{j=0}^{i} \overline{x}_{j,2}(t) + C\epsilon]}}{\epsilon N_k} \\ = & \overline{x}_{i,2}(t) \end{aligned}$$

and using (72) in the other direction we obtain $\Gamma = \overline{x}_{i,2}(t)$ as desired.

9.2 Proof of Lemma 6

We recall that we have analyzed $\overline{x}$ along a fixed $\omega \in \mathcal{C}$, where $\mathbb{P}(\mathcal{C}) = 1$. We now explicit the dependence on ω and treat quantities $\overline{x}(t)$ and $X^N(t)$ as random variables. Let

$$Z_n^N := \mathbf{1}^{A_n^1(1 - X_{0,0}^N(t_n^-) - X_{0,1}^{N_k}(t_n^-))}_{(Y_{i-1}^N(t_n^-), Y_i^N(t_n^-)]} \mathbb{I}_{\{\sum_{j=0}^{d} X_{j,2}^N(t_n^-) > 0\}}. \tag{73}$$

For all n, the random variable Z_n^N is $\mathcal{F}_n$-measurable where $\mathcal{F}_n := \{X^N(t_n^{N,\lambda-}), A_n^1, W_n\}$, and

$$\mathbb{E}[Z_n^N | \mathcal{F}_n \setminus A_n^1] = \frac{X_{i,2}^N(t_n^-)}{1 - X_{0,0}^N(t_n^-) - X_{0,1}^N(t_n^-)} \mathbb{I}_{\{\sum_{j=0}^{d} X_{j,2}^N(t_n^-) > 0\}} \tag{74}$$

where the set $\mathcal{F}_n \setminus W_n$ denotes the set $\mathcal{F}_n$ with A_n^1 removed. Now, let $\Delta_n^N := Z_n^N - \mathbb{E}[Z_n^N | \mathcal{F}_n \setminus W_n]$. Then, $\mathbb{E}[\Delta_n^N | \mathcal{F}_n \setminus W_n] = 0$ and $|\Delta_n^N| \leq 2$, and applying the Azuma–Hoeffding inequality, we get

$$\mathbb{P}\left(\frac{1}{N} \left| \sum_{n=1}^{N} \Delta_n^N \right| > \delta \right) \leq 2 \exp\left(-\frac{(N\delta)^2}{8N} \right) \tag{75}$$

for any $\delta > 0$. Since $\sum_N \exp\left(-N\delta^2/8\right) < \infty$, an application of the Borel–Cantelli lemma shows that $\frac{1}{N} \sum_{n=1}^{N} \Delta_n^N \to 0$ almost surely. In particular,

$$\begin{aligned} \lim_{N \to \infty} \frac{1}{\epsilon N} \sum_{\substack{n = \mathcal{N}_\phi(Nt) + 1: \\ W_n = 0}}^{\mathcal{N}_\phi(N(t+\epsilon))} \mathbf{1}^{A_n^1(1 - X_{0,0}^N(t_n^-) - X_{0,1}^N(t_n^-))}_{(Y_{i-1}^N(t_n^-), Y_i^N(t_n^-)]} \\ \times \left(\mathbb{I}_{\{\sum_{j=0}^{d} X_{j,2}^N(t_n^-) > 0\}} - \mathbb{E}[Z_n^N | \mathcal{F}_n \setminus W_n] \right) = 0 \end{aligned} \tag{76}$$

almost surely. We now come back to work on a given trajectory ω. In view of the previous equality, we may redefine $\mathcal{C}$ in Lemma 1 to be a subset of $\mathcal{C}'$ where $\mathbb{P}(\mathcal{C}' = 1)$ and (76) holds for all $\omega \in \mathcal{C}'$. Therefore, we fix $\omega \in \mathcal{C}$ and use (33) and (74) to obtain that

$$\lim_{k\to\infty} \frac{1}{\epsilon N_k} \sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \mathbb{E}[Z_n^{N_k} | \mathcal{F}_n \setminus W_n] \mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}}$$
$$\leq \lim_{k\to\infty} \frac{1}{\epsilon N_k} \sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \frac{\overline{x}_{i,2}(t)+\delta}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)-\delta}$$
$$\times \mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}}$$

for any $\delta > 0$ sufficiently small. Replacing δ by $-\delta$ in the last fraction term, the previous inequality can be reversed and letting $\delta \downarrow 0$, we obtain

$$\lim_{k\to\infty} \frac{1}{\epsilon N_k} \sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \mathbb{E}[Z_n^{N_k} | \mathcal{F}_n \setminus W_n] \mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}}$$
$$= \frac{\overline{x}_{i,2}(t)}{1-\overline{x}_{0,0}(t)-\overline{x}_{0,1}(t)} \lim_{k\to\infty} \sum_{\substack{n=\mathcal{N}_\phi(N_k t)+1:\\ W_n=0}}^{\mathcal{N}_\phi(N_k(t+\epsilon))} \frac{\mathbb{I}_{\{\sum_{j=0}^d X_{j,2}^{N_k}(t_n^-)>0\}}}{\epsilon N_k} \tag{77}$$

Finally, (77) and (76) give (50).

9.3 Proof of Lemma 8

Let $w_d := \sum_{j=0}^d x_{j,2}$. For now, let us assume that $x_{0,2} > 0$. In this case, $w_d > 0$ and (60) boils down to (by (6))

$$x_{1,2} = \tfrac{\lambda}{w_d} x_{0,2} \tag{78a}$$
$$x_{i+1,2} = x_{i,2} + \tfrac{\lambda}{w_d}(x_{i,2} - x_{i-1,2}), \quad i = 1, \dots, d \tag{78b}$$
$$x_{d+2,2} = x_{d+1,2} - \tfrac{\lambda}{w_d} x_{d,2} \tag{78c}$$
$$x_{i+1,2} = x_{i,2}, \qquad i \geq d+2. \tag{78d}$$

Since $\|x\| = 1$, (78) holds if and only if $x_{i,2} = 0$ for all $i \geq d+2$ and

$$x_{i,2} = \left(\tfrac{\lambda}{w_d}\right)^i x_{0,2}, \qquad i = 0, \dots, d+1. \tag{79}$$

If $d = 0$, then $w_d = x_{0,2}$ and $x_{1,2} = \lambda$, and the lemma is proven. Thus, let $d \geq 1$. Summing (79) over $i = 0, \dots, d$, we obtain

$$w_d = \frac{1 - \left(\frac{\lambda}{w_d}\right)^{d+1}}{1 - \frac{\lambda}{w_d}} x_{0,2} \tag{80}$$

and letting $z_d := \sum_{j=1}^d x_{j,2}$ we obtain (10) as desired and it remains to prove (7a). Using (79), we notice that (10) holds if and only if

$$z_d + x_{0,2} = \frac{x_{0,2} - x_{d+1,2}}{1 - \frac{\lambda}{z_d + x_{0,2}}} \tag{81}$$

and rearranging terms we obtain $z_d + x_{0,2} - \lambda = x_{0,2} - x_{d+1,2}$. Then, (7a) follows by using the normalizing condition $\|x\| = 1$ as $x_i = 0$ for all $i \geq d+2$.

It remains to consider the case where $x_{0,2} = 0$. Here, $x_{0,2} = 0$ if and only if $x_{0,1} = 0$, by (59), which implies

$$g \mathbb{I}_{\{x_{0,0}>0\}} = 0, \tag{82}$$

by (58b). In addition, if $w_d > 0$, then (60) boils down again to (78) and $x_{0,2} = 0$ would imply that $x_{i,2} = 0$ for all i. This is not possible in view of $\|x\| = 1$ and, therefore, we must have $w_d = 0$. Since necessarily $x_{0,0} < 1$, (6) simplifies to

$$h_i(x) = \begin{cases} x_{d+1,2} \mathbb{I}_{\{i=d\}} \mathbb{I}_{\{x_{d+1,2} \leq \lambda\}} & \text{if } i \leq d, \\ \frac{x_{i,2}}{1-x_{0,0}} (\lambda - x_{d+1,2})^+ & \text{if } i > d \end{cases} \tag{83}$$

and substituting in (61) we get

$$x_{i,2} = 0, \quad i \leq d \tag{84a}$$
$$x_{d+1,2} = x_{d+1,2} \mathbb{I}_{\{x_{d+1,2} \leq \lambda\}} \tag{84b}$$
$$x_{i,2} = \frac{x_{i-1,2}}{1 - x_{0,0}} (\lambda - x_{d+1,2})^+, \quad i \geq d+2. \tag{84c}$$

This gives (9a). Now, if $x_{d+1,2} > \lambda$, then (84b) is violated, and if $x_{d+1,2} = 0$, then (84c) and $\|x\| = 1$ give the contradiction that $1 = \lambda$. So, necessarily $x_{d+1,2} \in (0, \lambda]$, i.e., (9c). Here, we notice that $x_{d+1,2}$ is not tied to a specific value. Then, summing (84c) we obtain

$$\sum_{i \geq d+2} x_{i,2} = \sum_{i \geq d+2} \frac{x_{i-1,2}}{1 - x_{0,0}} (\lambda - x_{d+1,2}), \tag{85}$$

which, using $\|x\| = 1$ and (84), holds if and only if

$$1 - x_{d+1,2} - x_{0,0} = \frac{\lambda - x_{d+1,2}}{1 - x_{0,0}} (1 - x_{0,0}) \tag{86}$$

i.e., if and only if $x_{0,0} = 1 - \lambda$; note that $x_{0,0} = 0$ is not possible as otherwise (84c) and $\|x\| = 1$ give the contradiction that $1 < \lambda$. Since $x_{0,0} > 0$, necessarily $g = 0$ by (82), which gives (9d). Using $x_{d+1,2} \leq \lambda$ and $x_{0,0} = 1 - \lambda$ in (84c), we obtain $x_{d+2,2} = x_{d+1,2}\,(1 - x_{d+1,2}/\lambda)$ and applying inductively (84c), we obtain (9b). This concludes the proof.

10 Proof of Proposition 1

The fact that $x^\star$ is a fixed point is trivial. Suppose that there exists $\delta > 0$ such that $x_{0,2}(t) > 0$ on $(0, \delta]$. Then, there exists $\delta' > 0$ such that $\dot{x}_{0,2}(t) > 0$ on $(0, \delta']$. Using (4c), which gives $\dot{x}_{0,2} = x_{1,2} - \lambda + \beta x_{0,1} - \gamma x_{0,2}$, we obtain

$$x_{1,2}(t) + \beta x_{0,1}(t) > \lambda + \gamma x_{0,2}(t), \quad \forall t \in (0, \delta'] \tag{87}$$

and thus $x_{1,2}(0) + \beta x_{0,1}(0) = \lim_{t\downarrow 0} x_{1,2}(t) + \beta x_{0,1}(t) \geq \lambda$, by continuity of the fluid model. This contradicts the last condition in (15) and thus $x_{0,2}(t) = 0$ on a right neighborhood of zero, say $[0, \delta]$. Since $g(x) = \lambda - 1 + x_{0,0}$, on $[0, \delta]$ we obtain (using (4))

$$\dot{x}_{0,0} = -\alpha(\lambda - 1 + x_{0,0}) \tag{88a}$$
$$\dot{x}_{0,1} = \alpha(\lambda - 1 + x_{0,0}) - \beta x_{0,1} \tag{88b}$$
$$\dot{x}_{0,2} = 0 \tag{88c}$$
$$\dot{x}_{i,2} = x_{i+1,2} - x_{i,2} + h_{i-1}(x) - h_i(x), \quad i \geq 1 \tag{88d}$$

where

$$h_i(x) = \begin{cases} \beta x_{0,1} + x_{1,2} & \text{if } i = 0, \\ \frac{x_{i,2}}{y_1} (\lambda - x_{1,2} - \beta x_{0,1})^+ & \text{if } i > 0. \end{cases} \tag{89}$$

We observe that (88a)-(88b) form an autonomous linear ODE system. By continuity of $x(t)$, (15) holds as well on a right neighborhood of zero. Now, we actually show that (15) holds on $[0, \infty)$, i.e., $\delta = +\infty$. Towards this purpose, let us analyze the system (88a)-(88b) in isolation. After some algebra, we obtain

$$x_{0,0}(t) = 1 - \lambda + (x_{0,0} - 1 + \lambda)e^{-\alpha t} \tag{90a}$$

$$x_{0,1}(t) = \frac{\alpha(x_{0,0} - 1 + \lambda)}{\beta - \alpha}(e^{-\alpha t} - e^{-\beta t}) + x_{0,1}(0)e^{-\beta t}. \tag{90b}$$

Thus,

i) $x_{0,0}(t)$ monotonically decreases to zero as $t \to \infty$, and

ii) $y_0(t) = y_1(t) < 1$ with both $y_0(t)$ and $y_1(t)$ monotonically increasing to λ because $\dot{x}_{0,0} + \dot{x}_{0,1}$ is always non-increasing and $x_{0,2}$ stays on zero.

To prove that (15) holds on $[0, \infty)$, it remains to show that $x_{1,2}(t) + \beta x_{0,1}(t) < \lambda$ for all $t \geq 0$. This property is true because $x_{1,2} + \beta x_{0,1} \leq y_1 + x_{0,1} = 1 - x_{0,0} = \lambda - (x_{0,0}(0) - 1 + \lambda)e^{-\alpha t} < \lambda$. Thus, $x(t)$ satisfies (88) on $[0, \infty)$. In addition, since $x_{0,0}(0) + x_{0,1}(0) < 1$ and $\dot{x}_{0,0}(t) + \dot{x}_{0,1}(t) = -\beta x_{0,1}(t) \leq 0$ for all t, the drift function of (88) is Lipschitz and therefore it induces a unique flow [11,page 56]. Since $x_{0,0}(t) \downarrow 1 - \lambda$ as $t \to \infty$, for all $t \geq 0$

$$\begin{aligned}\dot{\overline{Q}}(x(t)) = \lambda - y_1(t) &= x_{0,0}(t) + x_{0,1}(t) + \lambda - 1 \\ &\geq x_{0,0}(t) + \lambda - 1 > 0,\end{aligned} \tag{91}$$

where the first equality follows by Lemma 9. In particular, $\lim_{t\to\infty} Q(x(t))$ exists and must be greater than λ because $\lambda < \overline{Q}(x(0)) < \infty$. Combining (90) and (91), we obtain

$$\begin{aligned}\dot{\overline{Q}}(x(t)) &= (x_{0,0}(0) - 1 + \lambda)e^{-\alpha t} + x_{0,1}(0)e^{-\beta t} \\ &\quad + \frac{\alpha(x_{0,0}(0) - 1 + \lambda)}{\beta - \alpha}(e^{-\alpha t} - e^{-\beta t}) \\ &= \underbrace{\frac{\beta(x_{0,0}(0) - 1 + \lambda)}{\beta - \alpha}}_{:=C_1} e^{-\alpha t} \\ &\quad + \underbrace{\left(x_{0,1}(0) - \frac{\alpha(x_{0,0}(0) - 1 + \lambda)}{\beta - \alpha}\right)}_{:=C_2} e^{-\beta t}.\end{aligned}$$

Integrating,

$$\begin{aligned}\overline{Q}(x(t)) =& \overline{Q}(x(0)) + \frac{C_1}{\alpha}(1 - e^{-\alpha t}) + \frac{C_2}{\beta}(1 - e^{-\beta t}) \\ \xrightarrow[t\to\infty]{}& \overline{Q}(x(0)) + \frac{\alpha + \beta}{\alpha\beta}(x_{0,0}(0) - 1 + \lambda) + \frac{1}{\beta}x_{0,1}(0)\end{aligned}$$

which proves (17). Finally, suppose that $\lim_{t\to\infty} x_{1,2}(t)$ exists, say $x_{1,2}(\infty)$. Then, necessarily $x_{1,2}(\infty) < \lambda$ because $y_1(t) \to \lambda$ and $\lim_{t\to\infty} Q(x(t)) > \lambda$ excludes that $x_{1,2}(t) \to \lambda$. Then, using (88d) when $i = 1$ and that $x_{1,2}(t)$ is Lipschitz continuous,

$$\begin{aligned}0 &= \lim_{t\to\infty} \dot{x}_{1,2}(t) \\ &= \lim_{t\to\infty} x_{2,2} + \beta x_{0,1} - \frac{x_{1,2}}{1 - x_{0,0} - x_{0,1}}(\lambda - x_{1,2} - \beta x_{0,1}) \\ &= \lim_{t\to\infty} \left(x_{2,2} - \frac{x_{1,2}}{\lambda}(\lambda - x_{1,2})\right) \\ &= -x_{1,2}(\infty)\left(1 - \frac{x_{1,2}(\infty)}{\lambda}\right) + \lim_{t\to\infty} x_{2,2},\end{aligned}$$

which shows that $\lim_{t\to\infty} x_{2,2}$ must exists as well and be equal to $x_{1,2}(\infty)\left(1 - \frac{x_{1,2}(\infty)}{\lambda}\right)$. By induction, $\lim_{t\to\infty} x_{i,2}$ exists and is equal to $x_{i,2}(\infty)\left(1 - \frac{x_{1,2}(\infty)}{\lambda}\right)^{i-1}$. Thus, $x(\infty) \in \mathcal{S}_{\text{subopt}}$.

11 Additional material supporting numerical simulations

Table 1 reports the numerical values of $\mathcal{R}_{\text{Wait}}$ and $\mathcal{R}_{\text{Energy}}$ plotted in Figure 2.

$\lambda = 0.35$

	$d = 1$	$d = 5$	$d = 10$
$N = 100$	$0.01786, 1.00296$	$0.00140, 1.03773$	$0.001046, 1.00762$
$N = 500$	$0.00674, 1.00271$	$0.00031, 1.01113$	$0.000013, 1.00956$
$N = 1000$	$0.00400, 1.00387$	$0.00022, 1.00492$	$0.000007, 1.00554$

$\lambda = 0.7$

	$d = 1$	$d = 5$	$d = 10$
$N = 100$	$0.01414, 1.00091$	$0.01086, 1.00127$	$0.010230, 1.00284$
$N = 500$	$0.00250, 1.00200$	$0.00024, 1.00081$	$0.000158, 1.00153$
$N = 1000$	$0.00162, 1.00234$	$0.00011, 1.00355$	$0.000025, 1.00285$

Table 1

Numerical values of $(\mathcal{R}_{\text{Wait}}, \mathcal{R}_{\text{Energy}})$ in Figure 2.