Performance Evaluation 164 (2024) 102408

Contents lists available at ScienceDirect

Performance Evaluation

journal homepage: www.elsevier.com/locate/peva

The impact of load comparison errors on the power-of-d load balancing ☆

Sanidhay Bhambay [a], Arpan Mukhopadhyay [a,*], Thirupathaiah Vasantam [b]

[a] Department of Computer Science, University of Warwick, UK
[b] Department of Computer Science, Durham University, UK

ARTICLE INFO

Keywords:
Mean-field analysis
Lyapunov drift analysis
Load balancing
Power-of-d

ABSTRACT

We consider a system with n unit-rate servers where jobs arrive according a Poisson process with rate $n\lambda$ ($\lambda < 1$). In the standard *Power-of-d* or Pod scheme with $d \geq 2$, for each incoming job, a dispatcher samples d servers uniformly at random and sends the incoming job to the least loaded of the d sampled servers. However, in practice, load comparisons may not always be accurate. In this paper, we analyse the effects of noisy load comparisons on the performance of the Pod scheme. To test the robustness of the Pod scheme against load comparison errors, we assume an adversarial setting where, in the event of an error, the adversary assigns the incoming job to the worst possible server, i.e., the server with the maximum load among the d sampled servers. We consider two error models: *load-dependent* and *load-independent* errors. In the load-dependent error model, the adversary has limited power in that it is able to cause an error with probability $\epsilon \in [0,1]$ only when the difference in the minimum and the maximum queue lengths of the d sampled servers is bounded by a constant threshold $g \geq 0$. For this type of errors, we show that, in the large system limit, the benefits of the Pod scheme are retained even if g and ϵ are arbitrarily large as long as the system is heavily loaded, i.e., λ is close to 1. In the load-independent error model, the adversary is assumed to be more powerful in that it can cause an error with probability ϵ independent of the loads of the sampled servers. For this model, we show that the performance benefits of the Pod scheme are retained only if $\epsilon \leq 1/d$; for $\epsilon > 1/d$ we show that the stability region of the system reduces and the system performs poorly in comparison to the *random scheme*. Our mean-field analysis uses a new approach to characterise fixed points which neither have closed form solutions nor admit any recursion. Furthermore, we develop a generic approach to prove tightness and stability for any state-dependent load balancing scheme.

1. Introduction

Load balancing is a common technique used in many computing and communication systems to distribute workloads evenly across multiple resources or servers. A popular load balancing technique used in many systems today is the *Power-of-d choices* or the *Pod* scheme [2–5]. In the Pod scheme, an arrival is sent to the resource having the smallest load (shortest queue length) among a subset of d resources sampled uniformly at random from the set of all resources. The popularity of the Pod scheme stems primarily

☆ This is an extended version of an earlier work ([1]) which has been published in the proceedings of ACM Mobihoc 2023.

* Corresponding author.
E-mail addresses: sanidhay.bhambay@warwick.ac.uk (S. Bhambay), arpan.mukhopadhyay@warwick.ac.uk (A. Mukhopadhyay), thirupathaiah.vasantam@durham.ac.uk (T. Vasantam).

https://doi.org/10.1016/j.peva.2024.102408
Received 5 November 2023; Received in revised form 20 February 2024; Accepted 22 February 2024
Available online 28 February 2024
0166-5316/

from the relative ease with which it can be implemented over a large number of dispatchers working in parallel to distribute the workload among a large set of servers of size n [6] - indeed, each dispatcher needs to independently sample only d ($\ll n$) servers and make a local comparison to dispatch an incoming request.

In their seminal works Mitzenmacher [4] and Dobrushin and Vvedenskaya [5] independently showed that when the number of resources or servers is large and the system is operating close to its maximum capacity, the Pod scheme with $d \geq 2$ results in an exponential reduction in the mean response time of jobs compared to the *random scheme* where requests are assigned uniformly at random to resources.

However, the above result was shown to hold only when the server with the minimum load among the d sampled servers is always correctly identified. In real systems, however, it is not always possible to correctly identify the server with the minimum load, especially when the loads of the sampled servers are close to each other. For example, in data centres, a delay in obtaining the load information from servers may cause changes in the actual loads of the servers resulting in an error in comparison [7,8]. Similarly, in concurrent settings [9–11], where multiple dispatchers work in parallel, once a dispatcher decides to send its job to a given server, multiple other dispatchers can send their jobs to the same server before the first dispatcher is able to do so. This may result in sending the job to the server not having the minimum load. Load comparison results can also be affected by noise or adversarial attacks potentially causing the system to become unstable [12].

Motivated by the above scenarios, recently, in the theoretical computer science community, there has been an interest in the study of load balancing algorithms under noisy load comparisons [9–11,13]. Their focus of study is the *balls and bin process* in which m balls are sequentially placed into n ($\leq m$) bins. If each ball is placed into the least loaded of d randomly sampled bins, then the classical results of [14,15] imply that the *gap* between the maximum load and the average load of the bins is $\log\log n + O(1)$ with high probability. This gap is exponentially smaller than the gap obtained by randomly placing the balls into bins without any comparison. To establish the robustness of the above process to noisy load comparisons, recent works [10,13] considered various adversarial error models. One of their key findings was for the setting where the adversary is capable of returning the worst bin when the loads of the compared bins differ by no more than a threshold value of $g \geq 0$. They showed that even in this noisy setting the gap between the maximum and the average load of a bin remain bounded by $O\left(\frac{g}{\log g}\log\log n + g\right)$. This shows that, in the static setting, where jobs come from a finite pool and do not leave the system, the benefits of comparing a small number of bins are retained even when comparing "similarly loaded" bins results in an error.

The above observation naturally leads to the question whether similar results hold in the dynamic setting, where jobs form an infinite sequence of arrivals and each job leaves the system after being processed. To quantitatively determine the robustness of the Pod scheme against worst case load comparison errors, we assume the same adversarial setting as in the static case, i.e., we assume that in the event of an "error" an incoming request is assigned to the server having the maximum load (i.e., the worst server) within the sampled subset of d servers. To distinguish the scenario where the power of the adversary depends on the loads of the sampled servers from the scenario where it does not, we consider the following two types of error (introduced more formally in Section 2).

- *Load-dependent error model*: In this model, the adversary is limited in that it can cause an error only when the queue lengths of the sampled servers are "close" to each other, i.e., the maximum and the minimum queue lengths of the sampled servers differ by no more than a threshold value of $g \geq 0$.
- *Load-independent error model*: In this model, the adversary is not limited by the loads of sampled bins — an error can occur with a fixed probability ϵ *independently* of the loads of the sampled servers.

We study the effects of the above errors on the performance of the Pod scheme for a system composed of n unit-rate servers where jobs with exponentially distributed sizes arrive according to a Poisson process with rate $n\lambda$ ($\lambda < 1$).

1.1. Main contributions

It is natural to expect that as the threshold parameter g in the load-dependent error model increases, the performance of the Pod scheme under this error model would deteriorate, eventually resulting in a performance poorer than the random scheme. While this is true for light traffic (small values of λ), we show that, in the heavy traffic limit ($\lambda \to 1$), the performance of the system remains exponentially better than that under the random scheme for all values of g. This can be intuitively explained by the fact that queue-length differences in heavy traffic can quickly exceed the threshold parameter g due to erroneous job assignments, and, when this occurs, the Pod scheme correctly identifies the server with the minimum queue length among the sampled subset of servers. Hence, in the heavy traffic, the benefits of comparing d choices outweighs the negative impact of comparison errors. This is analogous to the behaviour of the Pod algorithm in static setting under the same error model.

For the load-independent error model, we show that the benefits of the Pod scheme are retained only if the error probability ϵ satisfies $\epsilon \leq 1/d$. For $\epsilon > 1/d$, we show that the system becomes unstable for arrival rates larger than $1/d\epsilon$ and the performance becomes worse than that under the random scheme. This can be intuitively explained by the fact that the random scheme can be thought of as a modified version of the classical Pod scheme where, after uniformly sampling the subset of d servers, the incoming job is assigned to any one of the sampled servers with probability $1/d$. Hence, under load-independent errors, the server with the maximum queue length is chosen more often in the Pod scheme than in the random scheme when $\epsilon > 1/d$.

While the above results are intuitive, proving them rigorously, is quite challenging from a technical point of view. Below we discuss these challenges in detail and our methodological contributions in relation to them:

- *Characterising the fixed point*: Characterising the performance of the Pod scheme under the error models described above requires characterising the tail behaviour of the steady-state queue length distribution which, in turn, is approximated by the fixed point of the corresponding mean-field process for large n. However, unlike the classical noiseless setting, the fixed point of the mean-field does not have any closed form solution in the noisy settings. Furthermore, in the load-dependent error model, the fixed point does not satisfy any recursion. In such cases, even the existence and uniqueness of the fixed point is not evident. The proof of global asymptotic stability of the fixed point also becomes more challenging as it relies on an inductive argument on the component indices. To overcome these challenges we employ a new approach that uses bounds on the decay rate of the mean-field process and its monotonicity. These bounds also help us compare the steady-state mean response time of jobs under the Pod scheme in the noisy settings to that under the random scheme. We believe our approach is more generally applicable to any system where the fixed point does not admit a closed form solution or a recursion.
- *Lack of a dominating system*: In the noiseless setting, a coupling can be constructed between a system operating under the Pod scheme and a system operating under the random scheme such that the later always dominates the former in terms of the empirical queue length distribution. This dominance holds for all values of the arrival rate λ and is key to establishing the superiority of the Pod scheme over the random scheme for all $\lambda < 1$. It is also implies the stability and tightness of queue lengths for the system operating under the Pod scheme. However, this dominance breaks in the noisy settings. In particular, under the load-dependent error model, the random scheme performs better than the Pod scheme for small values of λ. However, as λ increases the performance benefits of the Pod scheme becomes more prominent, eventually outperforming the random scheme when λ is close one. The opposite holds for the load-independent case in which the Pod performs better than the random scheme only for small values of λ. Due to the dichotomy mentioned above and the lack of a uniformly dominating system, proving stability and tightness results under the noisy settings becomes more challenging. The standard results of [16] and the fluid limit techniques of [17] are also not applicable since a job is not always sent to the server with the minimum load. We develop a generic framework to analyse stability and tightness for any scheme which compares the queue lengths of multiple servers to make job assignment decisions. This framework is based on analysing the drift of suitable Lyapunov functions with the help of job classes defined by the subset of servers chosen upon arrival.

1.2. Other related works

In the last two decades, the Pod scheme has emerged as a widely used load balancing scheme due to its promising gains and minimal overhead. It has been studied extensively under various scaling limits and traffic conditions. The mean-field scaling limit for this scheme was first studied for exponential service time distributions in the seminal works [4,5]. Their results were later generalised to general service time distributions in [18,19]. The heavy traffic optimality of the Pod scheme has been established in [20,21]. In [22], the analysis of the Pod scheme has been carried out for the case where the number of choices, d, is allowed to depend on the system size n and $d(n) = \omega(1)$. In this work, both the mean-field and Halfin–Whitt regimes are considered. In the mean-field limit, the Pod scheme has been shown to reach the same performance as the JSQ scheme. Recently, the Pod scheme has been analysed for different graph topologies. For example, in [23], the Pod scheme is analysed for non-bipartite graphs and sufficient conditions on the graph sequence is obtained to match the result on complete graphs in the mean-field limit. For the bipartite graphs, the Pod scheme and its variants have been analysed in [24,25]. In all cases, results similar to the complete graph setting have been obtained. For heterogeneous systems, the Pod scheme has been studied in [26,27] where speed-aware versions of the Pod scheme have been shown to yield similar performance benefits.

Recently, in the theoretical computer science community, the Pod scheme has been analysed in the context of *relaxed concurrent data structures*. Here the goal is to design data structures which allow inherently sequential computations (such as counting) to be done concurrently using multiple threads. This results in faster computations with a small loss of accuracy. In this context, a *multi-counter* data structure which uses the power-of-two choice algorithm to update multiple counters concurrently was proposed and analysed in [10]. Their analysis first incorporated the effects of load comparison errors in the adversarial setting for the balls and bins problem. This analysis was further improved in [9]. The current paper considers the effect of such errors in the dynamic setting.

1.3. Organisation

The remainder of the paper is structured as follows. In Section 2, we introduce the system model, define load-dependent and load-independent error models, and describe the system state. Our main results for the Pod-(g,ϵ) and Pod-ϵ schemes are presented in Section 3. The mean-field analysis of the Pod-(g,ϵ) scheme is conducted in Section 5, while the analysis for the Pod-ϵ scheme is provided in Section 6. Finally, we conclude the paper in Section 7.

2. System model

We consider a system consisting of n parallel servers, each with its own queue of infinite buffer size. Each server is able to process jobs at unit rate. Jobs arrive according to a Poisson process with a rate $n\lambda$ ($\lambda < 1$). Each job requires a random amount of work, independent and exponentially distributed with unit mean. The inter-arrival times and job lengths are assumed to be independent of each other. A job dispatcher assigns each incoming job to a queue of a server where jobs are served according to the First-Come-First-Server (FCFS) scheduling discipline.

Upon arrival of a job, the dispatcher samples d servers uniformly at random without replacement.[1] Under the classical Pod scheme, a job is sent to the server with the minimum queue length among the d servers selected uniformly at random. However, as explained in the introduction, the server with the minimum queue length may not always be correctly identified. Thus, to analyse the effect of inaccurate load comparisons on the performance of the power-of-d scheme, we consider the following two error models. Note that in both the models "an error" means that an arrival is sent to the server with the maximum queue length among the d sampled servers. This definition assumes the worst possible assignment of the incoming job as any other assignment would only result in a better performance of the system.

2.1. Load-dependent error model

In this model, an error occurs with probability $\epsilon \in [0, 1]$ only when the difference between the maximum and the minimum queue lengths of the sampled servers is in the range $(0, g]$ for some constant $g \geq 0$. If this difference is strictly above g, then we assume that no error is made, i.e., the job is sent to the server with the minimum queue length. In case of a tie, we assume that an arbitrary tie breaking rule based on server indices is used. In subsequent sections, it will be demonstrated that the system's performance remains unchanged regardless of the tie-breaking rule chosen. Without loss of generality (WLOG), we assume that servers are indexed from the index set $[n] = \{1, 2, \dots, n\}$, and, in case of a tie, the job is sent to the server with the smaller index among the d sampled servers. We refer to the Pod scheme under this model of error as the Pod-(g, ϵ) scheme.

2.2. Load-independent error model

In this model, an error occurs with probability $\epsilon \in [0, 1]$ independent of the current queue lengths of the sampled servers. More precisely, the incoming job is sent to the server having the maximum queue length among the sampled servers with probability ϵ and with probability $1 - \epsilon$ it is sent to the server with the minimum queue length. Ties are broken in the same way as discussed before. For simplicity, we refer to the Pod scheme under this model of error as the Pod-ϵ scheme.

2.3. System state and notations

To analyse the system under the schemes discussed above, we first introduce Markovian state descriptors of the system. We use two Markovian state descriptors. First, we define the queue-length vector at time $t \geq 0$ as

$$\mathbf{Q}^{(n)}(t) = (Q_k^{(n)}(t), k \in [n]),$$

where $Q_k^{(n)}(t)$ denotes the queue length of the kth server. Second, we define the tail measure on the queue lengths at time t as

$$\mathbf{x}^{(n)}(t) = (x_i^{(n)}(t), i \geq 1),$$

where $x_i^{(n)}(t)$ denotes the fraction of servers with at least i jobs at time t. For completeness, we set $x_i^{(n)}(t) = 1$ for all $i \leq 0$ and all $t \geq 0$. From the Poisson arrival and the exponential job size assumption it is clear that both $\mathbf{Q}^{(n)} = (\mathbf{Q}^{(n)}(t), t \geq 0)$ and $\mathbf{x}^{(n)} = (\mathbf{x}^{(n)}(t), t \geq 0)$ are Markov processes. When the system is stable, we denote by π_n the unique invariant measure of the process $\mathbf{x}^{(n)}$ and we use $\mathbf{x}^{(n)}(\infty)$ and $\mathbf{Q}^{(n)}(\infty)$ to denote the steady-state values of the processes $\mathbf{x}^{(n)}$ and $\mathbf{Q}^{(n)}$, respectively. As the load balancing scheme does not distinguish between servers, we have

$$\mathbb{P}(Q_i^{(n)}(t) \geq k) = \mathbb{E}[\mathbb{1}\left(Q_i^{(n)}(t) \geq k\right)] = (1/n) \sum_{i \in [n]} \mathbb{E}[\mathbb{1}\left(Q_i^{(n)}(t) \geq k\right)] = \mathbb{E}[x_k^{(n)}],$$

for each $i \in [n]$ and each $t \in [0, \infty]$. To define the state space of the process $\mathbf{x}^{(n)}$, we first define the space $\bar{\mathcal{X}}$ as

$$\bar{\mathcal{X}} \triangleq \{\mathbf{x} = (x_i) : x_0 = 1, 1 \geq x_i \geq x_{i+1} \geq 0, \forall i \geq 1\}.$$

Note that the space $\bar{\mathcal{X}}$ is compact under the norm defined as

$$\|\mathbf{x}\| = \sup_{i \in \mathbb{Z}_+} \frac{|x_i|}{i+1}, \mathbf{x} \in \bar{\mathcal{X}}.$$

The process $\mathbf{Q}^{(n)}$ takes values in $\mathbb{Z}_+^n$ and the process $\mathbf{x}^{(n)}$ takes values in the space $\mathcal{X}^{(n)}$ defined as

$$\mathcal{X}^{(n)} \triangleq \{\mathbf{x} \in \bar{\mathcal{X}} : nx_i \in \mathbb{Z}_+ \ \forall i \geq 1\}.$$

We further define the space $\mathcal{X}$ as follows

$$\mathcal{X} \triangleq \{\mathbf{x} \in \bar{\mathcal{X}} : \|\mathbf{x}\|_1 < \infty\},$$

where the ℓ_1-norm, denoted by $\|\cdot\|_1$, is defined as $\|\mathbf{x}\|_1 \triangleq \sum_{i \geq 1} |x_i|$ for any $\mathbf{x} \in \mathcal{X}$.

[1] The asymptotic performance does not depend on whether servers are chosen with or without replacement. However, the stability of the system does depend on the way servers are sampled. All our results are stated for the case where servers are sampled without replacement.

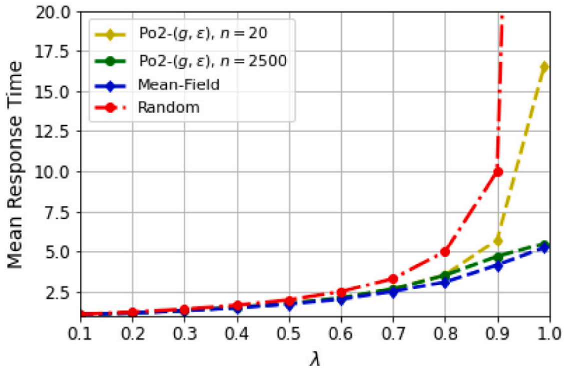

Fig. 1. Mean response time of jobs under the Po2-(g, ϵ) scheme as a function of arrival rate λ for $\epsilon = 0.4$, $g = 100$. For the simulations we set the number of arrival to be 10^7.

3. Main results and insights

In this section, we summarise our main results and discuss their consequences. Our first result characterises the stability region for each of the two schemes discussed before.

Theorem 1 (*Stability*)**.**

(i) For any $g \in \mathbb{R}_+, \epsilon \in [0,1]$, and $d \geq 2$ the system under the Pod-(g, ϵ) scheme is stable (i.e., the process $\mathbf{x}^{(n)}$ is positive recurrent) if and only if $\lambda < 1$. Furthermore, for $\lambda < 1$, the steady-state average queue length per server is bounded above as

$$\mathbb{E}_{\pi_n}[Q_i^{(n)}(\infty)] = \mathbb{E}_{\pi_n}\left[\sum_{i \geq 1} x_i^{(n)}(\infty)\right] \leq \frac{(1 + g\mathbb{1}(\epsilon > 1/d))\lambda}{1 - \lambda}. \tag{1}$$

(ii) For any $\epsilon \in [0,1]$ and $d \geq 2$, the system under the Pod-ϵ scheme is stable, if and only if $\lambda < \min\left(1, \frac{1}{d\epsilon}\right)$. Furthermore, for $\lambda < \min(1, 1/d\epsilon)$, the steady state average queue length per server is bounded above as

$$\mathbb{E}_{\pi_n}[Q_i^{(n)}(\infty)] = \mathbb{E}_{\pi_n}\left[\sum_{i \geq 1} x_i^{(n)}(\infty)\right] \leq \frac{\lambda}{1 - \max(1, d\epsilon)\lambda}. \tag{2}$$

The proof of the above theorem is given in Section 4. The theorem implies that the Pod-(g, ϵ) scheme achieves the maximal stability region for all values of g and ϵ. In contrast, for the Pod-ϵ scheme, the maximal stability region is achieved only when $\epsilon \leq 1/d$; for $\epsilon > 1/d$, the stability region shrinks to $\lambda < 1/d\epsilon$. This can be intuitively explained as follows: For the Pod-(g, ϵ) scheme, whenever the queue length difference between any two servers in the sampled subset of servers exceeds the threshold g, the incoming job is assigned to the server with the minimum queue length within the subset. This prohibits the queue length differences to grow too large, thereby stabilising the system even for large values of λ. However, for the Pod-ϵ scheme with $\epsilon > 1/d$, the probability with which the server with the maximum queue length is selected exceeds $1/d$ which is the probability of selecting the same server if the random scheme were used instead. As a result, for $\epsilon > 1/d$, the Pod-ϵ scheme performs poorly in comparison to the random scheme and looses some of the stability region.

In addition to the stability regions, the above theorem provides uniform (in the system size n) bounds on the steady-state mean queue length per server for each scheme. These uniform bounds are crucial in establishing the tightness of the stationary measures and justifying interchange of the limits in $\lim_{n\to\infty} \lim_{t\to\infty} \mathbf{x}^{(n)}(t) = \lim_{t\to\infty} \lim_{n\to\infty} \mathbf{x}^{(n)}(t)$, which shows that the mean-field approximation of the steady-state behaviour of the finite system is asymptotically exact.

The bounds in (1) and (2) also help us to compare the performance of the Pod-(g, ϵ) and the Pod-ϵ schemes to that of the random scheme. For example, when $\epsilon \leq 1/d$, both the upper bounds reduce to $\lambda/(1 - \lambda)$ which is the steady-state average queue length per server under the random scheme. This shows that, under both models of error, the Pod scheme performs better than the random scheme when the error probability $\epsilon \leq 1/d$. This is intuitive, since, for $\epsilon \leq 1/d$, the probability with which the server with the minimum queue length is chosen from the sampled subset of d servers is higher than the probability with which the same server would be chosen if the random assignment scheme were used. For $\epsilon > 1/d$, however, the bounds in (1) and (2) become higher than $\lambda/(1 - \lambda)$ indicating that the schemes may perform poorly in comparison to the random scheme.

This is numerically verified in Figs. 1 and 2 for the Po2-(g, ϵ) scheme and in Fig. 3 for the Po2-ϵ scheme. In each of these figures, we plot the steady-state mean response time of jobs as a function of the normalised arrival rate λ. From Figs. 1 and 3, we observe that both the schemes outperform the random scheme when $\epsilon \leq 1/2$. For $\epsilon > 1/2$, however, the Po2-ϵ scheme becomes unstable for

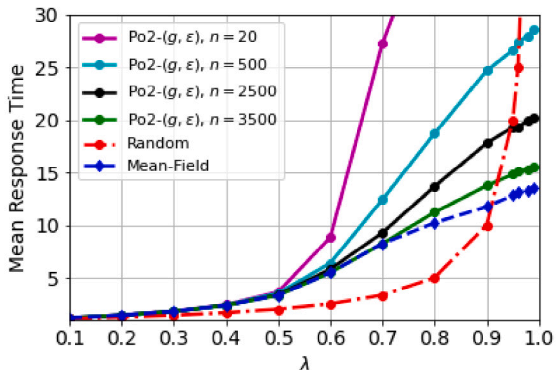

Fig. 2. Mean response time of jobs under the Po2-(g, ϵ) scheme and the random scheme as a function of arrival rate λ for $\epsilon = 0.8, g = 100$. For the simulations we set the number of arrival to be 10^7.

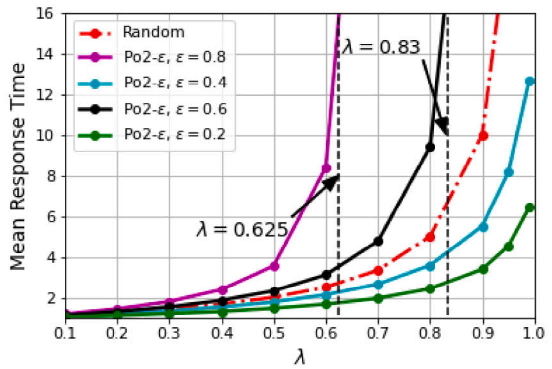

Fig. 3. Comparison of the Po2-ϵ scheme for $\epsilon \in \{0.2, 0.4, 0.6, 0.8\}$ with the random scheme. We set $n = 200$ and the number of arrival to be 10^7.

$\lambda \geq 1/2\epsilon$ and its performance becomes poorer than that of the random scheme for all $\lambda < 1$. For the Po2-(g, ϵ) scheme, we observe from Fig. 2 that the system is stable for all $\lambda < 1$ even when $\epsilon > 1/2$. However, for $\epsilon > 1/2$, we observe that the performance of the Po2-(g, ϵ) scheme is superior to that of the random scheme only when λ is close to one; for low traffic, the random scheme may outperform the Po2-(g, ϵ) scheme. The loss of uniform stochastic dominance by the random scheme means that the usual approach of proving stability and uniform bounds via coupling will not work here. Instead, we use drifts of suitable Lyapunov functions to prove our results. Our approach is generic and can be used for any scheme where queue lengths of multiple servers are compared to dispatch the incoming jobs.

Our next result characterises the behaviour of the Pod-(g, ϵ) scheme as the system size becomes large, i.e., $n \to \infty$. We show that the dynamics of the system can be characterised by a system of ordinary differential equations (ODEs) in this limit and the fixed point of this system characterises the steady-state performance of the system in the limit. Moreover, we characterise the steady-state performance of the limiting system by comparing it with that under the random scheme.

Theorem 2 (*Pod*-(g, ϵ)).

(i) (Mean-Field Limit): Let $g \in \mathbb{R}_+$, $\epsilon \in [0, 1]$, $d \geq 2$, and assume $\mathbf{x}^{(n)}(0) \in \mathcal{X}^{(n)}$ for each n and $\mathbf{x}^{(n)}(0) \stackrel{a.s}{\to} \mathbf{u} \in \mathcal{X}$ under ℓ_1 as $n \to \infty$. Then, for each $T \geq 0$, we have

$$\sup_{t \in [0,T]} \left\| \mathbf{x}^{(n)}(t) - \mathbf{x}(t) \right\|_1 \stackrel{a.s}{\to} 0$$

where $\mathbf{x} = (\mathbf{x}(t) = (x_i(t), i \geq 1), t \geq 0)$ satisfies $\mathbf{x}(0) = \mathbf{u}$ and for $t \geq 0$ and $i \geq 1$

$$\dot{x}_i(t) = G_i(\mathbf{x}(t)) \triangleq \lambda p_{i-1}(\mathbf{x}(t)) - (x_i(t) - x_{i+1}(t)). \tag{3}$$

Here for each $i \geq 1$, G_i is the ith component of the function $\mathbf{G} = (G_i, i \geq 1) : \mathcal{X} \to \mathbb{R}^\infty$ and, for $\mathbf{x} \in \mathcal{X}$, $p_{i-1}(\mathbf{x})$ is defined as

$$p_{i-1}(\mathbf{x}) = x_{i-1}^d - x_i^d + \epsilon\Big[(x_{i-g-1} - x_i)^d - (x_{i-1} - x_{i+g})^d$$

$$+(x_i - x_{i+g})^d - (x_{i-1-g} - x_{i-1})^d\Big]. \tag{4}$$

(ii) (Mean-Field Steady State Behaviour): For $g \in \mathbb{R}_+, \epsilon \in [0,1]$, $d \geq 2$, and $\lambda < 1$, there exists a unique $\mathbf{x}^* \in \mathcal{X}$ such that, $\mathbf{G}(\mathbf{x}^*) = \mathbf{0}$. In addition, $\mathbf{x}^*$ satisfies

$$x_1^* = \lambda \tag{5}$$

$$x_k^* = \lambda\Big[(x_{k-1}^*)^d + \epsilon \sum_{i=k}^{k+g-1} (x_{i-1-g}^* - x_i^*)^d - \epsilon \sum_{i=k}^{k+g} (x_{i-1-g}^* - x_{i-1}^*)^d\Big], \text{ for } k \geq 2. \tag{6}$$

Moreover, any solution $\mathbf{x}(t)$ of (3) with $\mathbf{x}(0) \in \mathcal{X}$ converges to $\mathbf{x}^*$ in ℓ_1 as $t \to \infty$. Therefore, the sequence $(\pi_n)_{n\geq 1}$ converges weakly to the Dirac measure $\delta_{\mathbf{x}^*}$ concentrated on $\mathbf{x}^*$ as $n \to \infty$.

(iii) (Heavy-Traffic Limit): Furthermore, in the heavy traffic as $\lambda \to 1$ for $d \geq 2$, we have

$$\limsup_{\lambda\to 1^-} \frac{T_d^{g,\epsilon}(\lambda)}{\log(T_1(\lambda))} \leq \frac{g+1}{\log(d)}, \tag{7}$$

where $T_d^{g,\epsilon}(\lambda) = (1/\lambda)\sum_{i\geq 1} x_i^*$ is the steady state limiting (as $n \to \infty$) average response time of jobs under the Pod-(g,ϵ) scheme and $T_1(\lambda) = 1/(1-\lambda)$ is the steady state average response time of jobs under the random scheme.

Hence, the process $\mathbf{x}$, defined in Theorem 2. (i), characterises the dynamics of the system in the limit as $n \to \infty$. This will be referred to as the *mean-field limit* of the system or the *mean-field process*. In Section 5, we prove the mean-field limit of the Pod-(g,ϵ) scheme and its properties. The evolution of $\mathbf{x}$, described in (3), can be explained as follows. For the nth system, the component $x_i^{(n)}(t)$ increases by $1/n$ when a job joins a server with queue length exactly $i-1$. The rate at which this happens is $n\lambda p_{i-1}(\mathbf{x})$, where $p_{i-1}(\mathbf{x})$, for $\mathbf{x} \in \mathcal{X}$, is the probability that an arrival joins a server with queue-length $i-1$ when the system is in state $\mathbf{x}$. The expression of $p_{i-1}(\mathbf{x})$ in (4) can be obtained as follows. Under the Pod-(g,ϵ) scheme, a job joins a server with queue length $i-1$ under the following scenarios:

- At least one of the d sampled servers is of queue length exactly $i-1$, at least one of the remaining sampled server is of queue length greater than or equals to $i+g$, and all other remaining sampled servers have queue lengths in the range $\{i,\dots,i+g-1\}$. In this case, with probability 1 the job joins the server with queue length $i-1$ and the overall probability of this event is
$$\sum_{k=1}^{d-1}\sum_{m=1}^{d-k}\binom{d}{k}\binom{d-k}{m}(x_{i-1}-x_i)^k x_{i+g}^m (x_i - x_{i+g})^{d-(k+m)}.$$
- At least one of the sampled servers is of queue length $i-1$ and all other remaining sampled servers have queue lengths in the range $\{i-1-g,\dots,i-2\}$. The probability of this event is $\sum_{k=1}^d \binom{d}{k}(x_{i-1}-x_i)^k(x_{i-1-g}-x_{i-1})^{d-k}$, and, in this case, the server with queue length $i-1$ is selected with probability ϵ.
- At least one of the sampled servers is of queue length $i-1$ and all other remaining sampled servers have queue lengths in the range $\{i+1,\dots,i-1+g\}$. This event occurs with probability $\sum_{k=1}^d \binom{d}{k}(x_{i-1}-x_i)^k(x_i-x_{i+g})^{d-k}$ and, in this case, the server with queue length $i-1$ is selected with probability $(1-\epsilon)$.

Combining all the above probabilities, we obtain the expression for $p_{i-1}(\mathbf{x})$. Clearly, the expression of $p_{i-1}(\mathbf{x})$ is independent of the tie-breaking rule used. Hence, the performance of the system does not depend on the choice of tie-breaking rule. Finally, the component $x_i^{(n)}(t)$ decreases by $1/n$ when a job leaves a server with queue length i and this occurs with rate $n(x_i - x_{i+1})$. Hence, the total expected rate of change (drift) in the component $x_i^{(n)}(t)$ is given by $G_i(\mathbf{x}^{(n)}(t)) = p_{i-1}(\mathbf{x}^{(n)}(t)) - (x_i^{(n)}(t) - x_{i+1}^{(n)}(t))$. In the limit as $n \to \infty$, this becomes the rate of change of $x_i(t)$.

In part (ii) of Theorem 2, we show that, as $t \to \infty$, the mean-field process $\mathbf{x}$ converges in ℓ_1 to the unique point $\mathbf{x}^* \in \mathcal{X}$ at which $\mathbf{G}(\mathbf{x}^*) = \mathbf{0}$. This point $\mathbf{x}^*$ is referred to as the *fixed point* of the mean-field since starting at this point the mean-field remains at this point at all times. Since by part (i) we have $\mathbf{x}^{(n)}(t) \to \mathbf{x}(t)$ almost surely for each $t \geq 0$, the convergence to the fixed point implies $\lim_{t\to\infty}\lim_{n\to\infty}\mathbf{x}^{(n)}(t) = \lim_{n\to\infty}\lim_{t\to\infty}\mathbf{x}^{(n)}(t) = \mathbf{x}^*$, which, in turn, means that the fixed point $\mathbf{x}^*$ characterises the steady-state behaviour of the system in the limit as $n \to \infty$. In particular, $\lim_{n\to\infty}\mathbb{P}(Q_i^{(n)}(\infty) \geq k) = x_k^*$, i.e., x_k^* denotes the limiting tail probability that a server has at least k jobs in its queue at steady-state. In Fig. 4, we plot x_k^* as a function of k for the Pod-(g,ϵ) scheme, the classical Pod scheme and the random scheme. We observe that the rate at which the x_k^* decreases with k for the Pod-(g,ϵ) scheme is slower than that for the Pod scheme, but is higher than the same for the random scheme. We later show that this rate of decay is, in fact, super-exponential. This is the reason for obtaining an exponentially smaller mean response time of jobs under the Pod-(g,ϵ) scheme compared to that under the random scheme.

In Theorem 2.(iii) we compare the steady-state mean response time of jobs under the Pod-(g,ϵ) scheme to that under the random scheme when the traffic is high (i.e., $\lambda \to 1$). Note that (7) implies that $T_d^{g,\epsilon}(\lambda) = O(\log T_1(\lambda))$ as $\lambda \to 1$. Furthermore, by the previous part of the theorem, the steady-state mean response time of the jobs under the Pod-(g,ϵ) scheme converges as $n \to \infty$ to $T_d^{g,\epsilon}(\lambda)$. Hence, this result shows that, when the system is heavily loaded, the mean response time of jobs under the Pod-(g,ϵ) scheme is exponentially smaller than that under the random scheme. This is also verified in Fig. 2 for $\epsilon = 0.8$, $d = 2$ and $g = 100$. Note that for such high error rates, the mean response time of jobs under the Pod-(g,ϵ) policy can be larger than that under the random scheme for low values of λ. However, when λ is close to its maximum value 1, the Pod-(g,ϵ) scheme performs exponentially better than

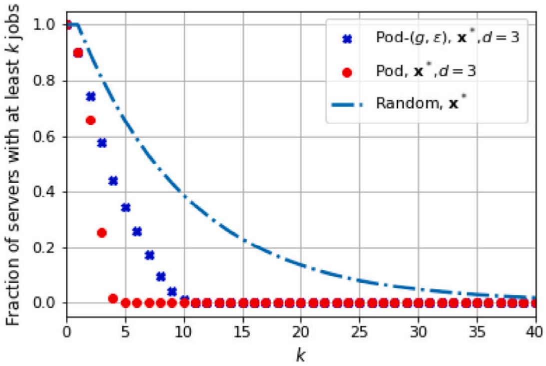

Fig. 4. The fraction of servers with at least k jobs under the Pod-(g, ϵ) scheme as a function of k. For simulations we set arrival rate $\lambda = 0.9$, $\epsilon = 0.7$, $d = 3$ and $g = 5$.

the random scheme for all values of g and ϵ. This implies that the advantage of having additional $d-1$ choices in the Pod scheme outweighs the negative impact the comparison errors when the traffic is high.

Remark 1. We observe from Figs. 1 and 2, that the accuracy of the mean-field approximation for the Pod-(g, ϵ) scheme deteriorates as λ approaches the critical value of one. As pointed out in [28], this is due to high variance of the stochastic system near the critical load. However, we note that the mean-field derived in Theorem 2 is asymptotically exact. Hence, the results of Theorem 2 are accurate for all sufficiently large n.

The main difficulty in proving Theorem 2 is that the fixed point $\mathbf{x}^*$ cannot be found in closed form. This is because each component x_k^* in (6) depends not only on the previous components but also on the next g components. This makes it hard to characterise the fixed point; indeed, even the existence of such $\mathbf{x}^*$ in S is not evident. This also makes proving the global stability difficult as it uses induction on the component index k. To overcome these difficulties, we use the monotonicity of the mean-field and uniform bounds on its tails. We believe that this new approach is generally applicable to similar systems where the fixed point cannot be found in closed form.

Similar to Theorem 2, our final result characterises the performance of the Pod-ϵ scheme in the limit as $n \to \infty$.

Theorem 3 (*Pod-ϵ*).

(i) *(Mean-Field Limit): Let $\epsilon \in [0,1]$, $d \geq 2$ and assume $\mathbf{x}^{(n)}(0) \in \mathcal{X}^{(n)}$ for each n and $\mathbf{x}^{(n)}(0) \overset{a.s}{\to} \mathbf{u} \in \mathcal{X}$ under ℓ_1 as $n \to \infty$. Then, for each $T \geq 0$, we have*

$$\sup_{t\in[0,T]} \left\| \mathbf{x}^{(n)}(t) - \mathbf{x}(t) \right\|_1 \overset{a.s}{\to} 0$$

where $\mathbf{x} = (\mathbf{x}(t) = (x_i(t), i \geq 1), t \geq 0)$ satisfies $\mathbf{x}(0) = \mathbf{u}$ and for $t \geq 0$ and $i \geq 1$

$$\dot{x}_i(t) = F_i(\mathbf{x}(t)) \triangleq \lambda p_{i-1}(\mathbf{x}(t)) - (x_i(t) - x_{i+1}(t)). \tag{8}$$

Here for each $i \geq 1$, F_i is the ith component of the function $\mathbf{F} = (F_i, i \geq 1) : \mathcal{X} \to \mathbb{R}^\infty$ and, for $\mathbf{x} \in \mathcal{X}$, $p_{i-1}(\mathbf{x})$ is defined as

$$p_{i-1}(\mathbf{x}) = (1-\epsilon)(x_{i-1}^d - x_i^d) + \epsilon((1-x_i)^d - (1-x_{i-1})^d). \tag{9}$$

We refer to the process $\mathbf{x}$ as the mean-field limit of the sequence $(\mathbf{x}^{(n)})_{n\geq 1}$.

(ii) *(Mean-Field Steady State Behaviour): For $\epsilon \in [0,1]$, $d \geq 2$, and $\lambda < \min(\frac{1}{d\epsilon}, 1)$, there exits $\mathbf{x}^* \in \mathcal{X}$ such that, if $\mathbf{x}(0) = \mathbf{x}^*$, then $\mathbf{x}(t) = \mathbf{x}^*$ for all $t \geq 0$. Furthermore, $\mathbf{x}^*$ satisfies the following recursion for all $k \geq 2$*

$$x_1^* = \lambda, \quad x_k^* = \lambda \left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon(1-(1-x_{k-1}^*)^d) \right]. \tag{10}$$

Moreover, any solution $\mathbf{x}(t)$ of (8) with $\mathbf{x}(0) \in \mathcal{X}$ converges to $\mathbf{x}^*$ in ℓ_1, i.e., $\|\mathbf{x}(t) - \mathbf{x}^*\|_1 \to 0$ as $t \to \infty$. The above results imply that the sequence $(\pi_n)_n$ converges weakly to the Dirac measure $\pi_* = \delta_{\mathbf{x}^*}$ concentrated on $\mathbf{x}^*$ as $n \to \infty$.

(iii) *(Heavy-Traffic Limit): For $\epsilon \leq 1/d$, $d \geq 2$ and $\lambda < 1$, we have*

$$\limsup_{\lambda\to 1^-} \frac{T_d^\epsilon(\lambda)}{\log(T_1(\lambda))} \leq \frac{1}{\log(d - d\epsilon(d-1))}, \tag{11}$$

where $T_d^\epsilon(\lambda) = \frac{\sum_{k=1}^\infty x_k^*}{\lambda}$ is the limiting (as $n \to \infty$) steady state average response time of jobs under the Pod-ϵ scheme.

The proof of above theorem is deferred till Section 6. In parts (i) and (ii) of Theorem 3, we characterise the mean-field limit $\mathbf{x}$ and its fixed point $\mathbf{x}^*$ under the Pod-ϵ scheme. As before, we show that the fixed point is unique and globally asymptotically stable. In the last part (part (iii)) of Theorem 3, we compare the mean response time of jobs under the Pod-ϵ scheme to that under the random scheme in the limit as $n \to \infty$. Our result indicates that when n is large and λ is close to 1, the steady state mean response time of jobs under the Pod-ϵ scheme satisfies $T_d^{\epsilon}(\lambda) = O(\log T_1(\lambda))$. This means that an exponential reduction in the steady state mean response time is achieved as long as $\epsilon \leq 1/d$. Hence, the Pod-ϵ scheme retains the benefits of the Pod scheme as long as $\epsilon \leq 1/d$.

Remark 2. It is important to note that the Pod-ϵ scheme is a special case of the Pod-(g,ϵ) scheme when $g = \infty$. Moreover, we can recover the mean-field limit of the Pod-ϵ scheme, as given in (8), by taking $g \to \infty$ in the mean-field limit of the Pod-(g,ϵ) scheme stated in (3). However, a separate analysis is required for the Pod-ϵ scheme because the results of the Pod-(g,ϵ) scheme necessitate that g be finite. In particular, the uniform bound established in Proposition 8 becomes trivial when $g = \infty$.

4. Stability and uniform bounds

In this section, we derive the stability regions and the uniform bounds stated in Theorem 1. To do so, we use drifts of appropriate Lyapunov functions. We first develop a general framework to analyse any load balancing scheme that compares the queue lengths of d uniformly sampled servers to dispatch every job. We then apply this generic framework to the schemes studied in this paper. Note that it is easy to generalise our framework further to the case where the sampling of the servers is not done uniformly.

For any function $V : \mathbb{Z}_+^n \to [0,\infty)$, the drift of $D_{\mathbf{Q}^n} V$ is defined as the expected rate of change in the value of the function along the trajectory of the process $\mathbf{Q}^n$ given the current state. More precisely,

$$D_{\mathbf{Q}^{(n)}} V(\mathbf{Q}) = \sum_{i=1}^{n} [r_i^{+,n}(\mathbf{Q})(V(\mathbf{Q} + \mathbf{e}_i^{(n)}) - V(\mathbf{Q})) \\ + r_i^{-,n}(\mathbf{Q})(V(\mathbf{Q} - \mathbf{e}_i^{(n)}) - V(\mathbf{Q}))], \tag{12}$$

where $\mathbf{e}_i^{(n)}$ denotes the n-dimensional unit vector with one in the ith position; $r_i^{\pm,n}(\mathbf{Q})$ are the transition rates from the state $\mathbf{Q}$ to the states $\mathbf{Q} \pm \mathbf{e}_i^{(n)}$. According to the Foster–Lyapunov theorem (Proposition D.3 of [29]), to prove the stability or positive recurrence of the process $\mathbf{Q}^{(n)}$, it is sufficient to show the existence of at least one function $V : \mathbb{Z}_+^n \to [0,\infty)$ such that $V(\mathbf{Q}) \to \infty$ when $\|\mathbf{Q}\|_1 \to \infty$ and $D_{\mathbf{Q}^n} V(\mathbf{Q}) < 0$ for all states $\mathbf{Q}$ lying outside a compact subset of the state-space. To further obtain uniform bounds on the stationary queue lengths, we use Proposition 1 of [30] which states that $\mathbb{E}_{\pi_n}[D_{\mathbf{Q}^{(n)}} V(\mathbf{Q}(\infty))] \geq 0$ if $D_{\mathbf{Q}^{(n)}} V(\mathbf{Q})$ is uniformly bounded for all states $\mathbf{Q} \in \mathbb{Z}_+^n$.

To compute the drift of any function V using (12), we first need the transition rates $r_i^{\pm,n}$. Clearly, the rate of departure from the ith server is given by $r_i^{-,n}(\mathbf{Q}) = \mathbb{1}(Q_i > 0)$. For any scheme which compares the states of d servers to dispatch the job to one of the servers, we define the class of an arrival as the subset of d servers sampled at the arrival instant. Let C denote the collection of all such classes. Since $|C| = \binom{n}{d}$ and a job is equally likely to belong to one of these classes, the arrival rate of any class $c \in C$ is $n\lambda / \binom{n}{d}$. Hence, we can write the rate of arrival to the ith server as

$$r_i^{+,n}(\mathbf{Q}) = \frac{n\lambda}{\binom{n}{d}} \sum_{c \in C : i \in c} p(i; c, \mathbf{Q}), \tag{13}$$

where $p(i; c, \mathbf{Q})$ is the probability that a class c arrival joins server i when the system is in state $\mathbf{Q}$. Note that the probability $p(i; c, \mathbf{Q})$ depends on the load balancing scheme used by the dispatcher. The exact expression of $p(i; c, \mathbf{Q})$ for each scheme is given later, but it is important to note that $\sum_{i \in c} p(i; c, \mathbf{Q}) = 1$ since a class c job joins one of the servers in class c with probability 1. For any class $c \in C$ and a given state $\mathbf{Q}$, we define $i_c^{\min}(\mathbf{Q})$ to be the index of the server with the minimum queue length after breaking the ties. Moreover, we define $i_c^{\max}(\mathbf{Q})$ to be the index of server with maximum queue length after resolving ties in class c. We also denote

$$m(c, \mathbf{Q}) = Q_{i_c^{\max}(\mathbf{Q})} - Q_{i_c^{\min}(\mathbf{Q})}, \tag{14}$$

for any class $c \in C$ and a given state $\mathbf{Q}$.

Now, for the Lyapunov function $V : \mathbb{Z}_+^n \to [0,\infty)$ defined as

$$V(\mathbf{Q}) = \sum_{i=1}^{n} Q_i^2,$$

the drift given in (12) simplifies to

$$D_{\mathbf{Q}^{(n)}} V(\mathbf{Q}) = \sum_{i=1}^{n} \{2[r_i^{+,n}(\mathbf{Q}) - r_k^{-,n}(\mathbf{Q})]Q_i + [r_i^{+,n}(\mathbf{Q}) + r_i^{-,n}(\mathbf{Q})]\},$$

which upon further simplification gives

$$D_{\mathbf{Q}^{(n)}} V(\mathbf{Q}) = 2 \sum_{i=1}^{n} r_i^{+,n}(\mathbf{Q}) Q_i - 2 \sum_{i \in [n]} Q_i + n\lambda + B(\mathbf{Q}), \tag{15}$$

where $B(\mathbf{Q}) = \sum_{i\in[n]} \mathbb{1}\left(Q_i > 0\right)$ represents the number of busy servers when system is in state $\mathbf{Q}$. In the above, we have used the facts $r_i^{-,n}(\mathbf{Q}) = \mathbb{1}\left(Q_i > 0\right)$ and $\sum_{i\in[n]} r_i^{+,n}(\mathbf{Q}) = n\lambda$. Moreover, using (13), the first term in the expression of the drift can be written as

$$\begin{aligned}\sum_{i=1}^{n} r_i^{+,n}(\mathbf{Q})Q_i &= \frac{n\lambda}{\binom{n}{d}} \sum_{i=1}^{n} \sum_{c\in C|i\in c} Q_i p(i;c,\mathbf{Q})\\ &= \frac{n\lambda}{\binom{n}{d}} \sum_{c\in C} \sum_{i\in c} Q_i p(i;c,\mathbf{Q}).\end{aligned} \tag{16}$$

Thus, to obtain the stability region and the uniform bound on steady-state queue length, we need to obtain upper bounds on $\sum_{i\in c} Q_i p(i;c,\mathbf{Q})$ for each scheme.

4.1. Pod-(g,ϵ) scheme

For the Pod-(g,ϵ) scheme, the probability $p(i;c,\mathbf{Q})$ for any class $c \in C$ is given by

$$\begin{aligned}p(i;c,\mathbf{Q}) = \mathbb{1}(i\in c)\Big[&\mathbb{1}\left(m(c,\mathbf{Q}) \geq g+1, i = i_c^{\min}(\mathbf{Q})\right) +\\ &\mathbb{1}\left(i = i_c^{\min}(\mathbf{Q}), m(c,\mathbf{Q}) = 0\right)\\ &+(1-\epsilon)\mathbb{1}\left(m(c,\mathbf{Q}) \in (0,g], i = i_c^{\min}(\mathbf{Q})\right)\\ &+\epsilon\mathbb{1}\left(m(c,\mathbf{Q}) \in (0,g], i = i_c^{\max}(\mathbf{Q})\right)\Big].\end{aligned} \tag{17}$$

Using the above expression, we obtain the following bound for Pod-(g,ϵ) scheme.

Lemma 4. *For $g \in \mathbb{R}_+$, $\epsilon \in [0,1]$, $d \geq 2$ and for any class $c \in C$, under the Pod-(g,ϵ) scheme, we have*

$$\sum_{i\in c} Q_i p(i;c,\mathbf{Q}) \leq \frac{\sum_{i\in c} Q_i}{d} + g\mathbb{1}(\epsilon > 1/d). \tag{18}$$

Proof. To prove the lemma, we first observe that for any $a_1 \leq a_2 \leq \cdots \leq a_d$, with $w_1 \geq 1/d$ and $w_i \leq 1/d$ for all $i \in \{2,\ldots,d\}$, $w_i \geq 0$ for all $i \in [d]$, and $\sum_{i=1}^{d} w_i = 1$ we have

$$\sum_{i=1}^{d} a_i w_i \leq \frac{\sum_{i=1}^{d} a_i}{d}. \tag{19}$$

The result of the lemma is direct when $m(c,\mathbf{Q}) = 0$. So, we consider the case $m(c,\mathbf{Q}) > 0$. For $\epsilon \leq 1/d$, using (17) and $m(c,\mathbf{Q}) > 0$, we have

$$\begin{aligned}p(i_c^{\min}(\mathbf{Q});c,\mathbf{Q}) &= \mathbb{1}(m(c,\mathbf{Q}) \geq g+1)\\ &+(1-\epsilon)\mathbb{1}(m(c,\mathbf{Q}) \in (0,g]) \geq \frac{1}{d}\mathbb{1}(m(c,\mathbf{Q}) \geq g+1)\\ &+\frac{1}{d}\mathbb{1}(m(c,\mathbf{Q}) \in (0,g]) = \frac{1}{d},\end{aligned}$$

and $p(i_c^{\max}(\mathbf{Q});c,\mathbf{Q}) = \epsilon\mathbb{1}(m(c,\mathbf{Q}) \in (0,g]) \leq 1/d$. Moreover, we have $p(i;c,\mathbf{Q}) = 0$ for all $i \in c$ and $i \notin \{i_c^{\min}(\mathbf{Q}), i_c^{\max}(\mathbf{Q})\}$. Therefore, using (19), we have

$$\sum_{i\in c} Q_i p(i;c,\mathbf{Q}) \leq \frac{\sum_{i\in c} Q_i}{d}.$$

Next, for $\epsilon > 1/d$, we note from (17) and $m(c,\mathbf{Q}) > 0$ that

$$\begin{aligned}&Q_{i_c^{\min}(\mathbf{Q})} p(i_c^{\min}(\mathbf{Q});c,\mathbf{Q}) =\\ &Q_{i_c^{\min}(\mathbf{Q})}\left(\mathbb{1}(m(c,\mathbf{Q}) \geq g+1) + (1-\epsilon)\mathbb{1}(m(c,\mathbf{Q}) \in (0,g])\right)\\ &= Q_{i_c^{\min}(\mathbf{Q})}\left((1-\epsilon) + \epsilon\mathbb{1}(m(c,\mathbf{Q}) \geq g+1)\right),\end{aligned}$$

which gives

$$\begin{aligned}&Q_{i_c^{\min}(\mathbf{Q})} p(i_c^{\min}(\mathbf{Q});c,\mathbf{Q}) + Q_{i_c^{\max}(\mathbf{Q})} p(i_c^{\max}(\mathbf{Q});c,\mathbf{Q})\\ &= Q_{i_c^{\min}(\mathbf{Q})}\left((1-\epsilon) + \epsilon\mathbb{1}(m(c,\mathbf{Q}) \geq g+1)\right)\\ &+Q_{i_c^{\max}(\mathbf{Q})}\epsilon\mathbb{1}(m(c,\mathbf{Q}) \in (0,g]).\end{aligned}$$

Therefore, using

$$\begin{aligned}&Q_{i_c^{\max}(\mathbf{Q})}\mathbb{1}(m(c,\mathbf{Q}) \in (0,g])\\ &\leq (g + Q_{i_c^{\min}(\mathbf{Q})})\mathbb{1}(m(c,\mathbf{Q}) \in (0,g])\end{aligned}$$

$$\leq (g + \sum_{i\in c} Q_i/d)\mathbb{1}(m(c,\mathbf{Q}) \in (0,g]),$$

and $Q_{i_c^{\min}(\mathbf{Q})} < \sum_{i\in c} Q_i/d + g$ in the above, we obtain

$$\sum_{i\in c} Q_i p(i;c,\mathbf{Q}) \leq g + \frac{\sum_{i\in c} Q_i}{d}.$$

This completes the proof. ■

Proof of Theorem 1.(i). Using the bound of Lemma 4, the RHS of (16) can be bounded as

$$\sum_{i=1}^{n} r_i^{+,n}(\mathbf{Q})Q_i \leq \frac{n\lambda}{\binom{n}{d}} \sum_{c\in C} \left\{ \frac{\sum_{i\in c} Q_i}{d} + g\mathbb{1}(\epsilon > 1/d) \right\}$$
$$= \frac{n\lambda}{\binom{n}{d}} \left(\frac{\binom{n-1}{d-1}}{d} \sum_{i\in[n]} Q_i + g\mathbb{1}(\epsilon > 1/d)\binom{n}{d} \right)$$
$$= \lambda \sum_{i\in[n]} Q_i + n\lambda g\mathbb{1}(\epsilon > 1/d) \tag{20}$$

Therefore, using (20) in (15), we can upper bound the drift $D_{\mathbf{Q}^{(n)}}V(\mathbf{Q})$ for the Pod-(g,ϵ) scheme as

$$D_{\mathbf{Q}^{(n)}}V(\mathbf{Q}) \leq 2(\lambda-1)\sum_{i=1}^{n} Q_i + 2n\lambda g\mathbb{1}\left(\epsilon > \frac{1}{d}\right) + n\lambda + B(\mathbf{Q}). \tag{21}$$

Now, since $B(\mathbf{Q}) \leq n$ and $\lambda < 1$, the drift $D_{\mathbf{Q}^{(n)}}V(\mathbf{Q})$ is strictly negative whenever $\sum_{i\in[n]} Q_i > n(\lambda(2\, g\mathbb{1}(\epsilon > 1/d) + 1) + 1)/2(1-\lambda)$, and is bounded above by $n(\lambda(2\, g\mathbb{1}(\epsilon > 1/d)+1)+1)$, otherwise. This shows that the system under the Pod-(g,ϵ) scheme is stable for all $\lambda < 1$. The necessity of this condition for stability can be established easily by showing that the drift of the Lyapunov function $V_1(\mathbf{Q}) = \sum_{i\in[n]} Q_i$ is always non-negative when $\lambda \geq 1$.

To prove (1), recall from the previous paragraph that $D_{\mathbf{Q}^{(n)}}V(\mathbf{Q})$ is uniformly bounded by $n(\lambda(2\, g\mathbb{1}(\epsilon > 1/d) + 1) + 1)$ for all $\mathbf{Q} \in \mathbb{Z}_+^n$. This implies that $\mathbb{E}_{\pi_n}\left[D_{\mathbf{Q}^{(n)}}V(\mathbf{Q}^{(n)}(\infty))\right] \geq 0$. Therefore, taking expectation of (21) and using the rate conservation equation $\mathbb{E}_{\pi_n}[B(\mathbf{Q})] = n\lambda$ (which holds in steady-state), we obtain

$$\frac{n\lambda(1+g\mathbb{1}(\epsilon > 1/d))}{(1-\lambda)} \geq \mathbb{E}_{\pi_n}\left[\sum_{i\in[n]} Q_i^{(n)}(\infty)\right]$$
$$= n\mathbb{E}_{\pi_n}\left[Q_i^{(n)}(\infty)\right],$$

where last equality follows due to the exchangeability of π_n. ■

4.2. Pod-ϵ scheme

For the Pod-ϵ scheme, $p(i;c,\mathbf{Q})$ for any class $c \in C$ is given by

$$p(i;c,\mathbf{Q}) = \mathbb{1}(i\in c)\Big[(1-\epsilon)\mathbb{1}\left(m(c,\mathbf{Q}) > 0, i = i_c^{\min}(\mathbf{Q})\right)$$
$$+\mathbb{1}\left(i = i_c^{\min}(\mathbf{Q}), m(c,\mathbf{Q}) = 0\right)$$
$$+\epsilon\mathbb{1}\left(m(c,\mathbf{Q}) > 0, i = i_c^{\max}(\mathbf{Q})\right)\Big]. \tag{22}$$

Using the expression above, we obtain the following bound.

Lemma 5. *For $\epsilon \in [0,1]$, $d \geq 2$ and for any class $c \in C$, under the Pod-ϵ, scheme we have*

$$\sum_{i\in c} Q_i p(i;c,\mathbf{Q}) \leq \max(1/d,\epsilon) \sum_{i\in c} Q_i. \tag{23}$$

Proof. For $m(c,\mathbf{Q}) = 0$ the above inequality follows directly. Consider $m(c,\mathbf{Q}) > 0$. For $\epsilon \leq 1/d$, using (22) we have $p(i_c^{\min}(\mathbf{Q});c,\mathbf{Q}) = (1-\epsilon) \geq 1/d$ and $p(i_c^{\max}(\mathbf{Q});c,\mathbf{Q}) = \epsilon \leq 1/d$. Therefore, from (19), it follows that $\sum_{i\in c} Q_i p(i;c,\mathbf{Q}) \leq \sum_{i\in c} Q_i/d$. When $\epsilon > 1/d$, for $m(c,\mathbf{Q}) > 0$, we can write

$$\sum_{i\in c} Q_i p(i;c,\mathbf{Q}) = Q_{i_c^{\min}(\mathbf{Q})}(1-\epsilon) + Q_{i_c^{\max}(\mathbf{Q})}\epsilon$$
$$\leq \epsilon(Q_{i_c^{\max}(\mathbf{Q})} + Q_{i_c^{\min}(\mathbf{Q})}) \leq \epsilon \sum_{i\in c} Q_i.$$

Hence, the proof is complete. ■

Proof of Theorem 1.(ii). Using the bound of Lemma 5, the RHS of (16) can be bounded as

$$\sum_{i=1}^{n} r_i^{+,n}(\mathbf{Q})Q_i \le \frac{n\lambda}{\binom{n}{d}} \sum_{c\in C}\sum_{i\in c} Q_i \max(1/d,\epsilon)$$
$$= \lambda \max(1, d\epsilon) \sum_{i\in[n]} Q_i \tag{24}$$

Therefore, using (24) in (15), we upper-bound the drift $D_{\mathbf{Q}^{(n)}}V(\mathbf{Q})$ for the Pod-ϵ scheme as

$$D_{\mathbf{Q}^{(n)}}V(\mathbf{Q}) \le 2(\lambda \max(1,d\epsilon) - 1)\sum_{i\in[n]} Q_i + n\lambda + B(\mathbf{Q}). \tag{25}$$

Since $B(\mathbf{Q}) \le n$, the above implies that, for $\lambda < \min(1, 1/d\epsilon)$, the drift is strictly negative whenever $\sum_{i\in[n]} Q_i > n(\lambda+1)/(1-\lambda\max(1,d\epsilon))$. This shows that the system under Pod-ϵ scheme is stable for all $\lambda < \min(1, 1/d\epsilon)$. Furthermore, since $\sup_{\mathbf{Q}\in\mathbb{Z}_+^n} D_{\mathbf{Q}^{(n)}}V(\mathbf{Q}) \le n(\lambda+1)$, using $\mathbb{E}_{\pi_n}\left[D_{\mathbf{Q}^{(n)}}V(\mathbf{Q}^{(n)}(\infty))\right] \ge 0$, we obtain

$$\frac{n\lambda}{(1-\lambda\max(1,d\epsilon))} \ge \mathbb{E}_{\pi_n}\left[\sum_{i\in[n]} Q_i^{(n)}(\infty)\right]$$
$$= n\mathbb{E}_{\pi_n}\left[Q_i^{(n)}(\infty)\right],$$

which proves (2).

Next we prove that for $\lambda \ge \min(1, \frac{1}{d\epsilon})$ the system is unstable. For $\epsilon \le 1/d$ and $\lambda \ge 1$, the process $\mathbf{Q}^{(n)}$ is not positive recurrent. This follows using the same argument as used in the stability proof of the Pod-(g,ϵ) scheme. Now, for $\epsilon > 1/d$ and $d\lambda\epsilon > 1$, we consider the Lyapunov function

$$V_2(\mathbf{Q}) = Q_{i^*(\mathbf{Q})},$$

where $i^*(\mathbf{Q}) = \arg\max_{i\in[n]} Q_i$ and $i^*(\mathbf{Q})$ is the minimum such index. Using (12), the drift of the function $V_2(\mathbf{Q})$ can be written as

$$D_{\mathbf{Q}^{(n)}}V_2(\mathbf{Q}) = \left(r_{i^*(\mathbf{Q})}^{+,n}(\mathbf{Q}) - \mathbb{1}\left(Q_{i^*(\mathbf{Q})} > 0\right)\right)$$
$$\ge \left(r_{i^*(\mathbf{Q})}^{+,n}(\mathbf{Q}) - 1\right).$$

From (13), we have

$$r_{i^*(\mathbf{Q})}^{+,n}(\mathbf{Q}) = \frac{n\lambda}{\binom{n}{d}} \sum_{c\in C|i^*(\mathbf{Q})\in c} p(i^*(\mathbf{Q}); c, \mathbf{Q})$$
$$\ge \frac{n\lambda}{\binom{n}{d}} \sum_{c\in C|i^*(\mathbf{Q})\in c} \epsilon = d\epsilon\lambda.$$

where the inequality follows from (22) since $Q_{i^*(\mathbf{Q})} \ge Q_j$ for any $j \in [n]$ and $i^*(\mathbf{Q})$ is the minimum such index. Hence, $D_{\mathbf{Q}^{(n)}}V_2(\mathbf{Q}) \ge d\epsilon\lambda - 1 \ge 0$ for all $\mathbf{Q} \in \mathbb{Z}_+^n$. Furthermore, since $D_{\mathbf{Q}^{(n)}}V_2(\mathbf{Q}) \le n\lambda$, the result follows from the Foster–Lyapunov criterion for transience and null recurrence (Theorem 3.3.10 of [31]). ■

5. Mean-field analysis of the Pod-(g,ϵ) scheme

In this section, we prove the main results for the Pod-(g,ϵ) scheme stated in Theorem 2.

5.1. *Mean-field limit of the Pod-(g,ϵ) scheme*

First, we establish the mean-field limit of the Pod-(g,ϵ) scheme (Theorem 2.(i)) for any $g \in \mathbb{R}_+$, $\epsilon \in [0,1]$, and $d \ge 2$. Note that under the Pod-(g,ϵ) scheme, the rate of transition of the process $\mathbf{x}^{(n)}$ from state $\mathbf{x} \in \mathcal{X}^{(n)}$ to state $\mathbf{y} \in \mathcal{X}^{(n)}$ is given by

$$r_{\mathbf{x},\mathbf{y}}^{(n)} = \begin{cases} n\lambda p_{i-1}(\mathbf{x}), & \text{if } \mathbf{y} = \mathbf{x} + \mathbf{e}_i/n \\ n(x_i - x_{i+1}), & \text{if } \mathbf{y} = \mathbf{x} - \mathbf{e}_i/n, \end{cases} \quad \forall i \ge 1, \tag{26}$$

where $p_{i-1}(\mathbf{x})$ is as defined in (4) and $\mathbf{e}_i$ is the ith unit vector in $\mathbb{R}^\infty$. Clearly, the above rates satisfy the transition structure for a *density-dependent jump Markov chain* [4,32]. Furthermore, it is easy to verify that $\sum_{y\in mathcal S} r_{\mathbf{x},\mathbf{y}}^{(n)} < n(\lambda+1)$ for all $x \in \mathcal{X}$ and the function $\mathbf{G} : \mathcal{X} \to \mathbb{R}^\infty$ is Lipschitz under the ℓ_1-norm with a Lipschitz constant of $L_\lambda^\epsilon = \lambda(8d\epsilon + 2d) + 2$ (proved in Lemma 6). Hence, using the Kurtz's theorem for density-dependent jump Markov processes [[33], Chapter 8], we obtain the desired result.

Lemma 6. *The function $\mathbf{G}(\mathbf{x})$ is Lipschitz under the ℓ_1 norm with constant $L_\lambda^\epsilon = \lambda(8d\epsilon + 2d) + 2$.*

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathcal{X}$. Then we have

$$\|\mathbf{G}(\mathbf{x}) - \mathbf{G}(\mathbf{y})\|_1$$

$$= \sum_{i\geq 1} |\lambda p_{i-1}(\mathbf{x}) - (x_i - x_{i+1}) - \lambda p_{i-1}(\mathbf{y}) + (y_i - y_{i+1})|$$
$$\leq \lambda \sum_{i\geq 1} |p_{i-1}(\mathbf{x}) - p_{i-1}(\mathbf{y})| + 2\,\|\mathbf{x}-\mathbf{y}\|_1 . \tag{27}$$

Now from (4), using the triangle inequality we can write for any $\mathbf{x},\mathbf{y} \in \mathcal{X}$

$$\begin{aligned}
|p_{i-1}(\mathbf{x}) - p_{i-1}(\mathbf{y})| &\leq \epsilon \Big\{ |(x_i - x_{i+g})^d - (y_i - y_{i+g})^d| \\
&+|(x_{i-1-g} - x_i)^d - (y_{i-1-g} - y_i)^d| + |(y_{i-1} - y_{i+g})^d - (x_{i-1} - x_{i+g})^d| \\
&+|(y_{i-1-g} - y_{i-1})^d - (x_{i-1-g} - x_{i-1})^d| \Big\} \\
&+|x_{i-1}^d - y_{i-1}^d| + |y_i^d - x_i^d|.
\end{aligned}$$

Using $a^d - c^d = (a-c)(a^{d-1} + \cdots + c^{d-1}) \leq d(a-c)$ for $0 \leq a, c \leq 1$ and summing for all $i \geq 1$ in the above expression, we can write

$$\sum_{i\geq 1} |p_{i-1}(\mathbf{x}) - p_{i-1}(\mathbf{y})| \leq (8d\epsilon + 2d)\,\|\mathbf{x}-\mathbf{y}\|_1 . \tag{28}$$

Hence, the result follows from (27) and (28). ■

5.2. Mean-field steady state behaviour for the Pod-(g,ϵ) scheme

We now turn to the proof of Theorem 2.(ii) which shows that the mean-field process $\mathbf{x}$ given by (3) has a unique fixed point $\mathbf{x}^*$ satisfying (5) and (6). Moreover, we show that the fixed point $\mathbf{x}^*$ is globally stable, i.e., all trajectories of the mean-field process $\mathbf{x}$ starting in $\mathcal{X}$ converges to $\mathbf{x}^*$.

For any $\mathbf{u} \in \mathcal{X}$, let $\mathbf{x}(t,\mathbf{u})$ denote the trajectory of the mean-field process starting at state $\mathbf{u}$. Further, define $v_k(t,\mathbf{u}) = \sum_{i\geq k} x_i(t,\mathbf{u})$ and $v_k(\mathbf{u}) = \sum_{i\geq k} u_i$ for each $k \geq 1$. When the context is clear, we shall drop the dependence of the trajectory on the initial state $\mathbf{u}$ and on the time t.

Lemma 7. *Let $g \in \mathbb{R}_+, \epsilon \in [0,1]$, $d \geq 2$. The following statements hold for the process $\mathbf{x}$ defined in Theorem 2.*

1. *If $\mathbf{u} \in \mathcal{X}$, then, for any $t \geq 0$, we have $\mathbf{x}(t,\mathbf{u}) \in \mathcal{X}$.*
2. *For any $\mathbf{u},\mathbf{u}' \in \mathcal{X}$ satisfying $\mathbf{u} \leq \mathbf{u}'$ we have $\mathbf{x}(t,\mathbf{u}) \leq \mathbf{x}(t,\mathbf{u}')$ for all $t \geq 0$, where the inequality $\leq$ is understood component-wise.*

Proof. We first note from (3) and (4) that

$$\begin{aligned}
\dot{x}_i \leq \lambda p_{i-1}(\mathbf{x}) &\leq \lambda[x_{i-1}^d + \epsilon x_{i-1-g}^d + \epsilon x_i^d] \\
&\leq \lambda(1+2\epsilon)x_{i-1-g}^d \leq M_{\lambda,\epsilon} x_{i-1-g},
\end{aligned}$$

where $M_{\lambda,\epsilon} = \lambda(1+2\epsilon)$. This implies that for each $i \geq 1$ we have

$$x_i(t) \leq x_i(0) + M_{\lambda,\epsilon} \int_0^t x_{i-1-g}(s)ds.$$

Using the above recursively for each $i \geq 1$, we obtain

$$x_i(t) \leq x_i(0) + \sum_{k=0}^{i-g-1} x_k(0) \frac{(M_{\lambda,\epsilon} t)^{i-g-k}}{(i-g-k)!}. \tag{29}$$

Summing the above for all $i \geq 1$ we obtain

$$v_1(\mathbf{x}(t)) \leq (g + 1 + v_1(\mathbf{x}(0))) \exp(M_{\lambda,\epsilon} t).$$

This shows that if $v_1(\mathbf{x}(0)) < \infty$, then $v_1(\mathbf{x}(t)) < \infty$ for all t, thus establishing the first part of the lemma.

For the second part, we note from (3) and (4) that

$$\begin{aligned}
\frac{\partial G_i(\mathbf{x})}{\partial x_{i-1}} &= \lambda\Big[d\left\{ x_{i-1}^{d-1} - \epsilon(x_{i-1} - x_{i+g})^{d-1} \right\} + \\
&\epsilon d(x_{i-1-g} - x_{i-1})^{d-1} \Big] \geq 0.
\end{aligned}$$

It is also easy to verify that $\frac{\partial G_i(\mathbf{x})}{\partial x_{i+g}}$, $\frac{\partial G_i(\mathbf{x})}{\partial x_{i-g-1}}$, and $\frac{\partial G_i(\mathbf{x})}{\partial x_{i+1}}$ are non-negative. Hence, for each $i \geq 1$, $G_i(\mathbf{x}) = \lambda p_{i-1}(\mathbf{x}) - (x_i - x_{i+1})$ is non-decreasing with respect to all x_k, $k \neq i$. Now the result follows from Theorem 5.3 of [34]. ■

The second property stated in Lemma 7 is called the *quasi-monotonicity* of the process $\mathbf{x}$. This property ensures that if the mean-field process starts from the idle initial state, i.e., if $\mathbf{x}(0) = \mathbf{e}_0 = (1,0,0,\dots)$, then it is monotonically non-decreasing in time, i.e.,

$$\mathbf{x}(t_1,\mathbf{e}_0) \leq \mathbf{x}(t_2,\mathbf{e}_0),\ \ 0 \leq t_1 \leq t_2 < \infty. \tag{30}$$

This follows because the state $\mathbf{e}_0$ is dominated by any other state in $\mathcal{X}$. In particular, $\mathbf{e}_0 \leq \mathbf{x}(t_2 - t_1, \mathbf{e}_0)$. Hence, by quasi-monotonicity, we have $\mathbf{x}(t_1, \mathbf{e}_0) \leq \mathbf{x}(t_1, \mathbf{x}(t_2 - t_1, \mathbf{e}_0)) = \mathbf{x}(t_2, \mathbf{e}_0)$.

Furthermore, Lemma 7 guarantees that if $\mathbf{x}(0) \in \mathcal{X}$ then $\mathbf{x}(t) \in \mathcal{X}$ for all $t \geq 0$. Hence, by adding (3) for all $i \geq k$ and using the fact that $\|x(t)\|_1 < \infty$ gives

$$\begin{aligned}\dot{v}_k(t) = \lambda\Big[x_{k-1}^d(t) + \epsilon \sum_{i=k}^{k+g-1} \Big[(x_{i-1-g}(t) - x_i(t))^d - \\ (x_{i-g}(t) - x_i(t))^d\Big] - \epsilon(x_{k-1-g}(t) - x_{k-1}(t))^d\Big] - x_k(t).\end{aligned} \tag{31}$$

Expanding $(x_{i-1-g}(t) - x_i(t))^d - (x_{i-g}(t) - x_i(t))^d$ in the above expression we get

$$\begin{aligned}\dot{v}_k(t) = \lambda\Big[x_{k-1}^d(t) + \epsilon \sum_{i=k}^{k+g-1} \Big[(x_{i-g-1}(t) - x_{i-g}(t)) \\ \sum_{j=1}^{d}(x_{i-1-g}(t) - x_i(t))^{d-j}(x_{i-g}(t) - x_i(t))^{j-1}\Big] \\ -\epsilon(x_{k-1-g}(t) - x_{k-1}(t))^d\Big] - x_k(t).\end{aligned}$$

Furthermore, using the fact that $x_{i-1-g}(t) - x_i(t) \leq x_{i-1-g}(t)$ and $x_{i-g}(t) - x_i(t) \leq x_{i-g}(t)$ in the above expression, we get the following bound

$$\begin{aligned}\dot{v}_k(t) &\leq \lambda\Big[x_{k-1}^d(t) + \epsilon \sum_{i=k}^{k+g-1} (x_{i-1-g}^d(t) - x_{i-g}^d(t)) - \epsilon(x_{k-g-1} - x_{k-1})^d\Big] - x_k(t), \\ &= \Big[x_{k-1}^d(t) + \epsilon(x_{k-1-g}^d(t) - x_{k-1}^d(t)) - \epsilon(x_{k-g-1}(t) - x_{k-1}(t))^d\Big] - x_k(t), \\ &\leq \lambda\Big[(1-\epsilon)x_{k-1}^d(t) + \epsilon x_{k-1-g}^d(t)\Big] - x_k(t), \\ &\leq \lambda x_{k-1-g}^d(t) - x_k(t),\end{aligned} \tag{32}$$

where the last inequality follows by using $x_{k-1-g} \geq x_{k-1}$.

Existence of the Fixed Point $\mathbf{x}^*$: To prove the existence of the fixed point $\mathbf{x}^*$ satisfying (5) and (6), we first show that $\|\mathbf{x}(t, \mathbf{e}_0)\|_1$ remains uniformly bounded for all $t \geq 0$. Note that this is a stronger result than $\|\mathbf{x}(t, \mathbf{e}_0)\|_1 < \infty$ for each $t \geq 0$ which has already been established in Lemma 7.

Proposition 8. *For $g \in \mathbb{R}_+$, $\lambda < 1$, $d \geq 2$ let $\mathbf{z}^* \in \mathcal{X}$ be defined as $z_i^* = 1$ for all $i \leq 0$ and $z_i^* = \lambda(z_{i-1-g}^*)^d$ for all $i \geq 1$. Then, we have $\mathbf{x}(t, \mathbf{e}_0) \leq \mathbf{z}^*$ for all $t \geq 0$ which implies that*

$$\|\mathbf{x}(t, \mathbf{e}_0)\|_1 \leq \|\mathbf{z}^*\|_1 = (g+1)\sum_{i\geq 1} \lambda^{\frac{d^i - 1}{d-1}}, \ \forall t \geq 0. \tag{33}$$

Proof. Let $\mathbf{x}(0) = \mathbf{e}_0 \leq \mathbf{z}^*$. Then, from (30) we have $\dot{x}_k(t) \geq 0$ for all $k \geq 1$ and for all $t \geq 0$. This further implies that $\dot{v}_k(t) = \sum_{i\geq k} \dot{x}_i(t) \geq 0, \forall k \geq 1, \forall t \geq 0$. Since $\mathbf{x}(0) \leq \mathbf{z}^*$, for $\mathbf{x}(t) > \mathbf{z}^*$ for some t there must exist $t_* < t$ such that $x_l(t_*) = z_l^*$ and $\dot{x}_l(t_*) > 0$ for some $l \geq 1$. Let m be the smallest component where the above two conditions are satisfied. Then, substituting $k = m$ in (32), we obtain

$$\begin{aligned}\dot{v}_m(t_*) &\leq \lambda x_{m-1-g}^d(t_*) - x_m(t_*) = \lambda x_{m-1-g}^d(t_*) - z_m^* \\ &= \lambda(x_{m-1-g}^d(t_*) - (z_{m-1-g}^*)^d) \\ &\leq 0,\end{aligned}$$

where the last inequality follows from the definition of m. Since we already know that $\dot{v}_m(t_*) \geq 0$, the above inequality implies $\dot{v}_m(t_*) = 0$. Hence, $\dot{x}_m(t_*) = \dot{v}_m(t_*) - \dot{v}_{m+1}(t_*) = -\dot{v}_{m+1}(t_*) \leq 0$ which contradicts the fact that $\dot{x}_m(t_*) > 0$. ■

Lemma 9. *Given $g \in \mathbb{R}_+$, $\lambda < 1$, there exists $\mathbf{x}^* \in \mathcal{X}$ such that*

$$\|\mathbf{x}(t, \mathbf{e}_0) - \mathbf{x}^*\|_1 \to 0, \ t \to \infty, \tag{34}$$

and $\mathbf{G}(\mathbf{x}^*) = \mathbf{0}$. Furthermore, $\mathbf{x}^*$ satisfies (5) and (6).

Proof. Since $x_i(t, \mathbf{e}_0) \in [0, 1]$ for each i and all $t \geq 0$ and $x_i(t, \mathbf{e}_0)$ is monotonically non-decreasing in time we must have $x_i(t) \to x_i^*$ as $t \to \infty$ for each $i \geq 1$ for some $\mathbf{x}^* = (x_i^*) \in \bar{\mathcal{X}}$.

We first show that the component-wise limit x^* defined above is also the ℓ_1 limit of $\mathbf{x}(t, \mathbf{e}_0)$ which will also imply that $\mathbf{x}^* \in \mathcal{X}$. To show this, we note from Proposition 8 that the uniform bound on $\sum_{i\geq 1} x_i(t, \mathbf{e}_0)$ in (33) implies by dominated convergence theorem that

$$\lim_{t\to\infty}\|\mathbf{x}(t)\|_1=\sum_{i\geq 1}\lim_{t\to\infty}x_i(t)=\sum_{i\geq 1}x_i^*=\left\|\mathbf{x}^*\right\|_1\leq (g+1)\sum_{i\geq}\lambda^{\frac{d^i-1}{d-1}}.$$

This shows that $\left\|\mathbf{x}(t,\mathbf{e}_0)-\mathbf{x}^*\right\|_1\to 0$ as $t\to\infty$, and $\mathbf{x}^*\in\mathcal{X}$.

It now remains to show that $\mathbf{G}(\mathbf{x}^*)=0$. Note that the convergence of $\mathbf{x}(t)\to\mathbf{x}^*$ in ℓ_1 as $t\to\infty$, and the monotonicity of $\mathbf{x}(t)$ imply that for any $\delta>0$ there exists a $t_\delta>0$ such that for all $t\geq t_\delta$ we have

$$\begin{aligned}\delta &\geq \|\mathbf{x}(t+h)-\mathbf{x}(t)\|_1\geq x_i(t+h)-x_i(t)\\ &=\int_t^{t+h}G_i(\mathbf{x}(s))ds\geq hG_i(\mathbf{x}(t_h^*)),\forall i\geq 1,\forall h\geq 0,\end{aligned}$$

where $t_h^*\in[t,t+h]$ is the time at which the continuous function $G_i(\mathbf{x}(s))$ attains its minimum value in the compact interval $[t,t+h]$. Therefore, we have

$$G_i(\mathbf{x}(t_h^*))\leq\frac{\delta}{h},\ i\geq 1. \tag{35}$$

Now we can write

$$\begin{aligned}G_i(\mathbf{x}^*)&=G_i(\mathbf{x}^*)-G_i(\mathbf{x}(t_h^*))+G_i(\mathbf{x}(t_h^*))\\ &\leq\left\|\mathbf{G}(\mathbf{x}^*)-\mathbf{G}(\mathbf{x}(t_h^*))\right\|_1+G_i(\mathbf{x}(t_h^*)) &(36)\\ &\leq L_\lambda^\epsilon\delta+\frac{\delta}{h}, &(37)\end{aligned}$$

where for the second inequality we use and the fact that the function $\mathbf{G}$ is Lipschitz with constant L_λ^ϵ and (35). Note that the above inequality is true for any $\delta>0$. Therefore, by fixing $h>0$ and letting $\delta\to 0$ we have $G_i(\mathbf{x}^*)=0$ for all $i\geq 1$. Hence, $\mathbf{G}(\mathbf{x}^*)=0$. Finally, we obtain (5) by using $\sum_{i\geq 1}G_i(\mathbf{x}^*)=0$ and (6) by using $\sum_{i\geq k}G_i(\mathbf{x}^*)=0$. ■

Global Stability and Uniqueness of the Fixed Point $\mathbf{x}^*$: Now we prove that for any $\mathbf{u}\in\mathcal{X}$, $\mathbf{x}(t,\mathbf{u})$ converges to $\mathbf{x}^*$ as $t\to\infty$ in ℓ_1, where x^* is the limit of $\mathbf{x}(t,\mathbf{e}_0)$ as defined in Lemma 9. By Proposition 8 and the dominated convergence theorem, it suffices to establish this convergence component-wise. Furthermore, it is sufficient to consider initial points $\mathbf{u}\leq\mathbf{x}^*$ and $\mathbf{u}\geq x^*$ since, by the quasi-monotonicity of $\mathbf{x}$, we have $\mathbf{x}(t,\min(\mathbf{u},\mathbf{x}^*))\leq\mathbf{x}(t,\mathbf{u})\leq\mathbf{x}(t,\max(\mathbf{u},\mathbf{x}^*))$, where the min and the max are taken component-wise.

Consider the case when $\mathbf{x}(0)=\mathbf{u}\leq\mathbf{x}^*$. Since $\mathbf{u}\geq\mathbf{e}_0$, by the quasi-monotonicity of $\mathbf{x}$, we have $\mathbf{x}(t,\mathbf{e}_0)\leq\mathbf{x}(t,\mathbf{u})\leq\mathbf{x}^*,\forall t\geq 0$. Hence, by Lemma 9, we have $\mathbf{x}(t,\mathbf{u})\to\mathbf{x}^*$ since $\mathbf{x}(t,\mathbf{e}_0)\to\mathbf{x}^*$.

Next we consider the case where $\mathbf{x}(0)=\mathbf{u}\geq\mathbf{x}^*$. We first show that $v_k(t,\mathbf{u})$ remains uniformly bounded for all $t\geq 0$ and for all $k\geq 1$. From quasi-monotonicity of $\mathbf{x}$ it follows that $\mathbf{x}(t,\mathbf{u})\geq\mathbf{x}^*$ for all $t\geq 0$. This implies, in particular, that $x_1(t,\mathbf{u})\geq x_1^*=\lambda$. Hence, from (31) for $k=1$, we have $\dot{v}_1(t,\mathbf{x}(0))=\lambda-x_1(t,\mathbf{u})\leq 0$. This implies that $0\leq v_k(t,\mathbf{u})\leq v_1(t,\mathbf{u})\leq v_1(\mathbf{u})$ for all $t\geq 0$ and all $k\geq 1$.

We shall now establish the convergence $x_i(t,\mathbf{u})\to x_i^*$ for all $i\geq 1$ by showing

$$\int_0^\infty(x_i(t,\mathbf{u})-x_i^*)dt<C_i, \tag{38}$$

where $C_i>0$ is a finite constant for each $i\geq 1$. To prove (38), we use induction on i. For $i=1$, using (5) we have

$$\begin{aligned}\int_0^\tau(x_1(t,\mathbf{u})-x_1^*)dt&=\int_0^\tau(x_1(t,\mathbf{u})-\lambda)dt\\ &=v_1(\mathbf{u})-v_1(\tau,\mathbf{u})\leq v_1(\mathbf{u}),\end{aligned}$$

where the second equality follows from (31) for $k=1$ and the inequality follows as $v_1(t,\mathbf{u})$ is uniformly bounded in t. Since the RHS is independent of τ, the integral on the left hand side must be by bounded $v_1(\mathbf{u})$ as $\tau\to\infty$. This shows the base case of the induction.

Now assume that (38) is true for all $i\leq L-1$. For $i=L$, using (31) and (6), we have

$$\begin{aligned}\int_0^\tau(x_L(t)-x_L^*)dt&=v_L(\mathbf{u})-v_L(\tau,\mathbf{u})+\lambda\int_0^\tau(x_{L-1}^d(t)-(x_{L-1}^*)^d)dt\\ &+\lambda\epsilon\int_0^\tau\left[(x_{L-1-g}^*-x_{L-1}^*)^d-(x_{L-1-g}(t)-x_{L-1}(t))^d\right]dt\\ &+\epsilon\lambda\int_0^\tau\sum_{i=L}^{L+g-1}\left[(x_{i-1-g}(t)-x_i(t))^d-(x_{i-1-g}^*-x_i^*)^d\right]dt\\ &+\epsilon\lambda\int_0^\tau\sum_{i=L}^{L+g-1}\left[(x_{i-g}^*-x_i^*)^d-(x_{i-g}(t)-x_i(t))^d\right]dt.\end{aligned} \tag{39}$$

Using the uniform boundedness of $v_L(t,\mathbf{u})$ in t, we can upper bound the term $v_L(\mathbf{u})-v_L(\tau,\mathbf{u})$ by $v_L(\mathbf{u})$. To complete the proof, we shall now bound each integral term appearing on the RHS. Note that using a^d-b^d expansion for $0\leq a,b\leq 1$ and $\mathbf{x}(t)\geq\mathbf{x}^*$, we have following inequalities

$$x_{L-1}^d(t)-(x_{L-1}^*)^d=(x_{L-1}(t)-x_{L-1}^*)\sum_{j=1}^d x_{L-1}^{d-j}(t)(x_{L-1}^*)^{j-1}$$

$$\leq d(x_{L-1}(t) - x^*_{L-1}),$$

$$(x^*_{L-1-g} - x^*_{L-1})^d - (x_{L-1-g}(t) - x_{L-1}(t))^d \leq d(x_{L-1}(t) - x^*_{L-1}).$$

Therefore, using the above inequalities and by the induction hypothesis, the integrals in the second and the third terms on the RHS of (39) can be easily bounded by dC_{L-1}. It now remains to bound the integrals in the last two terms. It is important to note that the last two terms of (39) contain $x_i(t)$ and x^*_i for $i \in \{L, L+1, \dots, L+g-1\}$ for which the induction hypothesis does not apply. We note that by monotonicity we have $x_i(t) \geq x^*_i$ for all $i \geq 1$. Therefore, using $a^d - b^d$ expansion we can bound the third integral of (39) as

$$\int_0^\tau \sum_{i=L}^{L+g-1} \left[(x_{i-1-g}(t) - x_i(t))^d - (x^*_{i-1-g} - x^*_i)^d \right] dt \leq$$
$$d \int_0^\tau \sum_{i=L}^{L+g-1} (x_{i-1-g}(t) - x^*_{i-1-g}) dt - \int_0^\tau \Big[\sum_{i=L}^{L+g-1} (x_i(t) - x^*_i)$$
$$\times \sum_{j=1}^{d} (x^*_{i-1-g} - x^*_i)^{d-j} (x_{i-1-g}(t) - x_i(t))^{j-1} \Big] dt. \tag{40}$$

Moreover, using the induction hypothesis, the first term on the RHS of the above inequality is bounded by $d \sum_{i=L}^{L+g-1} C_{i-1-g}$. Similarly, we can write the last integral of (39) as

$$\int_0^\tau \sum_{i=L}^{L+g-1} \left[(x^*_{i-g} - x^*_i)^d - (x_{i-g}(t) - x_i(t))^d \right] dt =$$
$$\int_0^\tau \Big[\sum_{i=L}^{L+g-1} (x^*_{i-g} - x_{i-g}(t)) \sum_{j=1}^{d} \Big[(x^*_{i-g} - x^*_i)^{d-j}$$
$$\times (x_{i-g}(t) - x_i(t))^{j-1} \Big] \Big] dt$$
$$+ \int_0^\tau \Big[\sum_{i=L}^{L+g-1} (x_i(t) - x^*_i) \sum_{j=1}^{d} (x^*_{i-g} - x^*_i)^{d-j}$$
$$\times (x_{i-g}(t) - x_i(t))^{j-1} \Big] dt. \tag{41}$$

Note that the first term on the RHS of (41) is negative for all $i \in \{L, L+1, \dots, L+g-1\}$ due to monotonicity. Combining (40) and (41), we can bound the last two integrals of (39) as

$$d \sum_{i=L}^{L+g-1} C_{i-1-g} + \int_0^\tau \sum_{i=L}^{L+g-1} (x_i(t) - x^*_i) \sum_{j=1}^{d} \Big[(x^*_{i-g} - x^*_i)^{d-j}$$
$$\times (x_{i-g}(t) - x_i(t))^{j-1} - (x^*_{i-1-g} - x^*_i)^{d-j}$$
$$\times (x_{i-1-g}(t) - x_i(t))^{j-1} \Big] dt \leq d \sum_{i=L}^{L+g-1} C_{i-1-g},$$

where the inequality follows as $(x^*_{i-g} - x^*_i)^{d-j}(x_{i-g}(t) - x_i(t))^{j-1} - (x^*_{i-1-g} - x^*_i)^{d-j}(x_{i-1-g}(t) - x_i(t))^{j-1} \leq 0$, for all $j \in [d]$. This completes the proof of global stability of $\mathbf{x}^*$. Since all the trajectories converge to $\mathbf{x}^*$, it must be the unique solution of $\mathbf{G}(\mathbf{y}) = \mathbf{0}$ since starting from any other $\mathbf{y} \neq \mathbf{x}^*$, satisfying $\mathbf{G}(\mathbf{y}) = 0$, the trajectory remains at $\mathbf{y}$ which contradicts the global stability of $\mathbf{x}^*$.

Limit Interchange: Note that Theorem 1.(i), implies that $\pi_n(\mathcal{X}) = 1, \forall n$. Therefore, we have $\pi_n(\bar{\mathcal{X}}) = 1$ for all n. Since, the space $\bar{\mathcal{X}}$ is compact, by Prohorov's theorem the sequence $(\pi_n)_n$ must converge weakly to the limit π^* with $\pi^*(\bar{\mathcal{X}}) = 1$. Furthermore, since by (1), $\mathbb{E}_{\pi_n}\left[\sum_{i\geq 1} x_i^{(n)}(\infty)\right]$ is uniformly bounded in n, we have $\pi^*(\mathcal{X}) = 1$. Now we prove that the measure π^* is the stationary measure of the mean-field process $\mathbf{x}$ defined in (3). We know that $(\pi_n)_n \Rightarrow \pi^*$ and the space $\mathcal{X}$ is separable. Therefore, the Skorokhod's Representation Theorem implies that $\mathbf{x}^{(n)}(0) \xrightarrow{a.s} \mathbf{x}(0)$. Moreover, if we start the process $\mathbf{x}^{(n)}(0) \sim \pi_n$, then $\mathbf{x}^{(n)}(t) \sim \pi_n$ for all $t \geq 0$. Hence, from Theorem 2.(i) it follows that $\mathbf{x}(t) \sim \pi^*$ for all $t \geq 0$. This proves that π^* is indeed the stationary measure for the mean-field process $\mathbf{x}$. Now from the global stability of the fixed point $\mathbf{x}^*$, it follows immediately that the stationary measure π^* is unique and is equal to $\delta_{\mathbf{x}^*}$. This completes the proof of limit interchange.

5.3. Heavy traffic limit for the Pod-(g, ϵ) scheme

In last we prove Theorem 2.(iii). Note that from Lemma 9, we know $\mathbf{x}(t, \mathbf{e}_0)$ converges to $\mathbf{x}^*$ as $t \to \infty$ in ℓ_1 and from Proposition 8 we have

$$\|\mathbf{x}(t, \mathbf{e}_0)\|_1 \leq (g+1) \sum_{i\geq 1} \lambda^{\frac{d^i - 1}{d-1}},$$

for all $t \geq 0$. Therefore, we can write $\|\mathbf{x}^*\|_1 \leq (g+1) \sum_{i\geq 1} \lambda^{\frac{d^i-1}{d-1}}$. Hence, dividing the above inequality both side with $-\log(1-\lambda)$, taking limit $\lambda \to 1^-$, we obtain the bound given in (7).

6. Mean-field analysis of the Pod-ϵ scheme

In this section we prove the main result for the Pod-ϵ scheme stated in Theorem 3.

6.1. Mean-field limit of the Pod-ϵ scheme

We first establish the mean-field limit of the Pod-ϵ policy given in Theorem 3-(i). First note that the rate of transitions of the process $\mathbf{x}^{(n)}$ from $\mathbf{x} \in \mathcal{X}^{(n)}$ to $\mathbf{y} \in \mathcal{X}^{(n)}$ is given by

$$q^{(n)}_{\mathbf{x},\mathbf{y}} = \begin{cases} n\lambda p_{i-1}(\mathbf{x}), & \text{if } \mathbf{y} = \mathbf{x} + \mathbf{e}_i/n \\ n(x_i - x_{i+1}), & \text{if } \mathbf{y} = \mathbf{x} - \mathbf{e}_i/n, \end{cases} \quad \forall i \geq 1, \tag{42}$$

where $p_{i-1}(\mathbf{x}) = (1-\epsilon)(x_{i-1}^d - x_i^d) + \epsilon((1-x_i)^d - (1-x_{i-1})^d)$ is the probability that an arrival joins a server of queue length $i-1$. The mean-field limit of the Pod-ϵ scheme is proved using the similar argument as shown for the Pod-(g,ϵ) scheme. First we show that the function $\mathbf{F}(\mathbf{x})$ for $\mathbf{x} \in \mathcal{X}$ defined in (8) is Lipschitz under ℓ_1-norm.

Lemma 10. *The function $\mathbf{F}(\mathbf{x})$ is Lipschitz with constant $2d\lambda + 2$.*

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathcal{X}$. Then we have

$$\begin{aligned} \|\mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{y})\|_1 &= \sum_{i\geq 1} |\lambda p_{i-1}(\mathbf{x}) - (x_i - x_{i+1}) - \lambda p_{i-1}(\mathbf{y}) + (y_i - y_{i+1})| \\ &\leq \lambda \sum_{i\geq 1} |p_{i-1}(\mathbf{x}) - p_{i-1}(\mathbf{y})| + 2\,\|\mathbf{x}-\mathbf{y}\|_1 \\ &\leq 2d\lambda\,\|\mathbf{x}-\mathbf{y}\|_1 + 2\,\|\mathbf{x}-\mathbf{y}\|_1 \\ &= (2d\lambda + 2)\,\|\mathbf{x}-\mathbf{y}\|_1, \end{aligned}$$

where the second inequality follows using $a^d - c^d = (a-c)(a^{d-1} + \cdots + c^{d-1}) \leq d(a-c)$ for $0 \leq a, c \leq 1$. This completes the proof. ■

Furthermore, from (42) it is clear that the rate at which the jumps occur in $\mathbf{x}^{(n)}$ is bounded everywhere that is

$$\sum_{\mathbf{y}\in\mathcal{X}} q^{(n)}_{\mathbf{x},\mathbf{y}} < n(\lambda+1),\ \forall\ \mathbf{x} \in \mathcal{X}. \tag{43}$$

Therefore, using Lemma 10 and from (43) we conclude that the conditions of Kurtz's Theorem are satisfied. Hence, we have

$$\lim_{n\to\infty} \sup_{t\leq u} \left\| \mathbf{x}^{(n)}(u) - \mathbf{x}(u) \right\|_1 = 0,\ \ a.s.$$

This completes the proof.

6.2. Mean-field steady state behaviour for the Pod-ϵ scheme

In this section we prove Theorem 3.(ii), which shows that the differential equations defined in (8) has a unique fixed point $\mathbf{x}^*$ and it follows the recursion defined in (10). Moreover, we prove that the fixed point $\mathbf{x}^*$ is globally stable and finally establish the interchange of limits.

For $\mathbf{u} \in \mathcal{X}$, we define $v_k(t,\mathbf{u}) = \sum_{i\geq k} x_i(t,\mathbf{u})$ and $v_k(\mathbf{u}) = \sum_{i\geq k} u_i$ for each $k \geq 1$.

Lemma 11. *Let $\epsilon \in [0,1]$, $d \geq 2$. The following statements hold for the process $\mathbf{x}$ defined in Theorem 3.*

1. *If $\mathbf{u} \in \mathcal{X}$, then, for any $t \geq 0$, we have $\mathbf{x}(t,\mathbf{u}) \in \mathcal{X}$.*
2. *For any $\mathbf{u}, \mathbf{u}' \in \mathcal{X}$ satisfying $\mathbf{u} \leq \mathbf{u}'$ we have $\mathbf{x}(t,\mathbf{u}) \leq \mathbf{x}(t,\mathbf{u}')$ for all $t \geq 0$, where the inequality $\leq$ is understood component-wise.*

Proof. From (8) and (9), we can write

$$\begin{aligned} \dot{x}_i \leq \lambda p_{i-1}(\mathbf{x}) &\leq \lambda[(1-\epsilon)(x_{i-1}^d - x_i^d) + d\epsilon x_{i-1}] \\ &\leq \lambda(1 + (d-1)\epsilon)x_{i-1} = M_{\lambda,\epsilon} x_{i-1}, \end{aligned}$$

where $M_{\lambda,\epsilon} = \lambda(1 + (d-1)\epsilon)$. Therefore, for each $i \geq 1$ we have

$$x_i(t) \leq x_i(0) + M_{\lambda,\epsilon} \int_0^t x_{i-1}(s)ds.$$

Hence, using the above recursively for each $i \geq 1$, we obtain

$$x_i(t) \leq x_i(0) + \sum_{k=0}^{i-1} x_k(0) \frac{(M_{\lambda,\epsilon} t)^{i-k}}{(i-k)!}.$$

Summing the above for all $i \geq 1$ we obtain $v_1(\mathbf{x}(t)) \leq (1 + v_1(\mathbf{x}(0))) \exp(M_{\lambda,\epsilon} t)$. Therefore, if $v_1(\mathbf{x}(0)) < \infty$, then $v_1(\mathbf{x}(t)) < \infty$ for all t. This completes the proof of first part.

To prove second part we need to show that $\frac{x_i(t)}{dt}$ is non-decreasing in $x_j(t)$ for all $j \neq i$. We know from (8) that

$$\frac{dx_i(t)}{dt} = \lambda\Big[(1-\epsilon)(x_{i-1}^d(t) - x_i^d(t)) + \\ \epsilon((1-x_i(t))^d - (1-x_{i-1}(t))^d)\Big] - (x_i(t) - x_{i+1}(t)).$$

Therefore, it is clear that the above expression is non-decreasing with $x_{i+1}(t)$. Note that the partial derivative of the above expression with respect to $x_{i-1}(t)$ component is $d(1-\epsilon)x_{i-1}^{d-1}(t) + d\epsilon(1-x_{i-1}(t))^{d-1}$, which clearly positive for $\epsilon \in [0,1]$. Hence, $\frac{dx_i(t)}{dt}$ is also non-decreasing with $x_{i-1}(t)$. ■

Fixed Point: For fixed point $\mathbf{x}^*$ we need to equate (8) to 0 and get

$$\lambda\left[(1-\epsilon)((x_{i-1}^*)^d - (x_i^*)^d) + \epsilon((1-x_i^*)^d - (1-x_{i-1}^*)^d)\right] \\ = x_i^* - x_{i+1}^*,\ i \geq 1. \tag{44}$$

Summing (44) for all $i \geq 1$ we get $x_1^* = \lambda$. Moreover, summing (44) for all $i \geq k$ we obtain

$$x_k^* = \lambda \lim_{m\to\infty} \sum_{i=k}^{k+m} p_{i-1}(\mathbf{x}^*) \\ = \lambda\left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon(1-(1-x_{k-1}^*)^d)\right],\ \forall k \geq 2,$$

where the second equality follows as $\lim_{m\to\infty} x_{m+k}^* = 0$.

Global Stability: To prove global stability of the fixed point $\mathbf{x}^*$, we use the monotonicity of the mean-field process $\mathbf{x}$ shown in Lemma 11.

Note that Lemma 11 guarantees that if $\mathbf{x}(0) \in mathcal S$ then $\mathbf{x}(t) \in \mathcal{X}$ for all $t \geq 0$. Hence, by adding (8) for all $i \geq k$ and using the fact that $\|x(t)\|_1 < \infty$ gives

$$\dot{v}_k(t,\mathbf{u}) = \lambda\left[(1-\epsilon)x_{k-1}^d(t,\mathbf{u}) + \epsilon(1-(1-x_{k-1}(t,\mathbf{u}))^d)\right] - x_k(t,\mathbf{u}) \tag{45}$$

Specifically, for $k = 1$ we have

$$\dot{v}_1(t,\mathbf{u}) = \lambda - x_1(t,\mathbf{u}). \tag{46}$$

Now from the monotonicity property (Lemma 11) of the mean-field process $\mathbf{x}$ we have for any $\mathbf{x}(0) \in \mathcal{X}$ and $t \geq 0$

$$\mathbf{x}(t, \min(\mathbf{x}(0),\mathbf{x}^*)) \leq \mathbf{x}(t,\mathbf{x}(0)) \leq \mathbf{x}(t,\max(\mathbf{x}(0),\mathbf{x}^*)), \tag{47}$$

where $\min(\mathbf{u},\mathbf{v})$ with $\mathbf{u},\mathbf{v} \in \mathcal{X}$ is defined by taking the component-wise minimum. From (47) it is clear that to prove global stability it is enough to prove convergence $\mathbf{x}(t,\mathbf{x}(0)) \to \mathbf{x}^*$ holds for initial states satisfying either of the following two conditions: (i) $\mathbf{x}(0) \geq \mathbf{x}^*$ and (ii) $\mathbf{x}(0) \leq \mathbf{x}^*$.

To prove convergence holds for above two initial conditions we first show that for any solution $\mathbf{x}(\cdot,\mathbf{x}(0)) \in \mathcal{X}$, $v_k(t,\mathbf{x}(0))$ is uniformly bounded in t for all $k \geq 1$. Consider the case when $\mathbf{x}(0) \geq \mathbf{x}^*$. From Lemma 11 it follows that for $\mathbf{x}(0) \geq \mathbf{x}^*$, we have $\mathbf{x}(t,\mathbf{x}(0)) \geq \mathbf{x}^*$ for all $t \geq 0$. Therefore, we can write

$$x_1(t,\mathbf{x}(0)) \geq x_1^* = \lambda,$$

where the last equality follows from (10). Hence, from (46) we have $\frac{dv_1(t,\mathbf{x}(0))}{dt} \leq 0$ from which it follows that $0 \leq v_1(t,\mathbf{x}(0)) \leq v_1(\mathbf{x}(0))$ for all $t \geq 0$. Since the sequence $\left(v_k(t,\mathbf{x}(0))\right)_{k\geq 1}$ is non-increasing, we have $0 \leq v_k(t,\mathbf{x}(0)) \leq v_1(\mathbf{x}(0))$ for all $k \geq 1$ and for all $t \geq 0$. This proves that $v_k(t,\mathbf{x}(0))$ is uniformly bounded in t for each $k \geq 1$ if $\mathbf{x}(0) \geq \mathbf{x}^*$. Now consider the case $\mathbf{x}(0) \leq \mathbf{x}^*$. From Lemma 11 it follows that for $\mathbf{x}(0) \leq \mathbf{x}^*$, we have $\mathbf{x}(t,\mathbf{x}(0)) \leq \mathbf{x}^*$ for all $t \geq 0$. Therefore, we have $v_1(t,\mathbf{x}(0)) \leq v_1(\mathbf{x}^*)$ for all $t \geq 0$. This shows that the component $v_k(t,\mathbf{x}(0))$ is uniformly bounded in t for each $k \geq 1$ for $\mathbf{x}(0) \leq \mathbf{x}^*$.

Since $v_k(t,\mathbf{x}(0))$ is uniformly bounded in t, the convergence $x_i(t,\mathbf{x}(0)) \to x_i^*$ for all $i \geq 1$ will follow from

$$\int_0^\infty (x_i(t,\mathbf{x}(0)) - x_i^*)dt < \infty,\ \forall i \geq 1, \tag{48}$$

for the case $\mathbf{x}(0) \geq \mathbf{x}^*$ and from

$$\int_0^\infty (x_i^* - x_i(t,\mathbf{x}(0)))dt < \infty,\ i \geq 1, \tag{49}$$

for the case $\mathbf{x}(0) \leq \mathbf{x}^*$. We now prove (48) to show convergence for the case $\mathbf{x}(0) \geq \mathbf{x}^*$; the proof of other case follows similarly.

We will use induction starting with $i = 1$. We can write (48) for $i = 1$ as

$$\begin{aligned}\int_0^\tau (x_1(t,\mathbf{x}(0)) - x_1^*)dt &= \int_0^\tau (x_1(t,\mathbf{x}(0)) - \lambda)\\ &= -\int_0^\tau \frac{dv_1(t,\mathbf{x}(0))}{dt}dt\\ &= v_1(\mathbf{x}(0)) - v_1(\tau,\mathbf{x}(0))\\ &\le v_1(\mathbf{x}(0)),\end{aligned}$$

where the second equality follows from (46) and the inequality follows as $v_1(t,\mathbf{x}(0))$ is uniformly bounded in t. Observe that the right hand side is bounded by a constant for all τ, the integral on the left hand side must converge as $\tau \to \infty$. This shows that $x_1(t,\mathbf{x}(0)) \to x_1^*$ as $t \to \infty$. Now assume that (48) is true for all $i \le L-1$. For $i = L$ we can write (48) as

$$\begin{aligned}&\int_0^\tau (x_L(t,\mathbf{x}(0)) - x_L^*)dt = \int_0^\tau \Big[\frac{-dv_L(t,\mathbf{x}(0))}{dt}\\ &+ \lambda\left\{(1-\epsilon)x_{L-1}^d(t,\mathbf{x}(0)) + \epsilon(1-(1-x_{L-1}(t,\mathbf{x}(0)))^d)\right\}\\ &- \lambda(1-\epsilon)(x_{L-1}^*)^d - \lambda\epsilon(1-(1-x_{L-1}^*)^d)\Big]dt,\\ &= v_L(\mathbf{x}(0)) - v_L(\tau,\mathbf{x}(0)) + \int_0^\tau \Big[\lambda(1-\epsilon)(x_{L-1}^d(t,\mathbf{x}(0))\\ &- (x_{L-1}^*)^d) + \lambda\epsilon\left\{(1-x_{L-1}(t,\mathbf{x}(0)))^d - (1-x_{L-1}^*)^d\right\}\Big]dt,\\ &\le v_L(\mathbf{x}(0)) + \int_0^\tau \lambda\Big[d(1-\epsilon)(x_{L-1}(t,\mathbf{x}(0)) - (x_{L-1}^*))\Big]dt\\ &+ \int_0^\tau \lambda\Big[d\epsilon(x_{L-1}(t,\mathbf{x}(0)) - x_{L-1}^*)\Big]dt,\end{aligned}$$

where the first equality follows from (45) for $k = L$ and from (10) for $k = L$. Moreover, the inequality follows as $v_L(t,\mathbf{x}(0))$ is uniformly bounded in t and using $a^d - b^d$ expansion for $0 \le a, b \le 1$. Furthermore, by the induction hypothesis, the last two integral on the right hand side of above inequality converges as $\tau \to \infty$. Hence, the integral on the left hand side also must converge as required.

Limit Interchange: We know from Theorem 1.(ii) that for $\epsilon \in [0,1]$, $d \ge 2$ and $\lambda < \min(1, 1/d\epsilon)$ we have

$$\mathbb{E}_{\pi_n}\left[\sum_{i\ge1} x_i^n(\infty)\right] \le \frac{\lambda}{1-\max(1,d\epsilon)\lambda}.$$

Therefore, using the global stability result and the process convergence result of the Pod-ϵ scheme, the limit interchange follows using similar argument as proved for the Pod-(g,ϵ) scheme.

6.3. Heavy-traffic limit of the Pod-ϵ scheme

In this section we prove Theorem 3.(iii), which computes the ratio of the average response time of jobs under the Pod-ϵ scheme with the logarithmic of average response time of jobs under the random scheme as $\lambda \to 1$. We first bound the recursion given in (10) for $k \ge 1$ as

$$\begin{aligned}x_k^* &= \lambda\left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon(1-(1-x_{k-1}^*)^d)\right]\\ &= \lambda\left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon\left\{x_{k-1}^*\left(1+\sum_{j=2}^d(1-x_{k-1}^*)^{j-1}\right)\right\}\right]\\ &\le \lambda\left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon\left\{x_{k-1}^*(1+(d-1)(1-x_{k-1}^*))\right\}\right]\\ &= \lambda\left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon d x_{k-1}^* - (d-1)\epsilon(x_{k-1}^*)^2\right]\\ &\le \lambda\left[(1-\epsilon)(x_{k-1}^*)^d + \epsilon d x_{k-1}^* - (d-1)\epsilon(x_{k-1}^*)^d\right]\\ &= \lambda\left[(1-d\epsilon)(x_{k-1}^*)^d + \epsilon d x_{k-1}^*\right],\end{aligned}$$

where the first inequality follows as $(1-x_{k-1}^*)^{j-1} \le (1-x_{k-1}^*)$ and the second inequality follows as $(x_{k-1}^*)^2 \ge (x_{k-1}^*)^d$. Therefore, we can write

$$x_k^* \le T_{\lambda,\epsilon}(x_{k-1}^*) = \lambda\sum_{i=1}^2 a_i(x_{k-1}^*)^{d_i},\ \forall k \ge 1, \tag{50}$$

where $a_1 = (1-d\epsilon)$, $a_2 = d\epsilon$, $d_1 = d$, and $d_2 = 1$. Hence, the result now follows from Theorem 4.5 of [35].

7. Conclusion and future works

In this paper, we analysed the effects of load comparison errors on the performance of the Pod scheme. We considered two models of error. For the load-dependent error model, we showed that the Pod scheme retains its benefits over the random scheme in the heavy traffic limit $\lambda \to 1$ for all values of g and ϵ. For the load-independent error model, we have shown that the Pod scheme retains its benefits over the random scheme only if the probability of error $\epsilon \leq 1/d$. We introduced a general framework using Lyapunov functions to prove stability and uniform bounds for our schemes. We also use a new approach to establish the mean-field limit results as the fixed point does not admit a recursive solution.

There are many interesting directions for further research. We have analysed the performance of the Pod-(g, ϵ) scheme assuming g to be constant independent of n. It will be interesting to see the effect of varying g as a function of n. Another direction is to study the effects of delay in receiving the queue length information at the dispatcher. A more explicit delay dependent error model can be considered. Here, the challenge will be to analyse the effect of the delay on the performance of the Pod scheme. Establishing the validity of refined mean-field approximations for the error models is also a challenging research direction as it requires the exponential stability of the fixed points.

CRediT authorship contribution statement

Sanidhay Bhambay: Writing – review & editing, Writing – original draft, Methodology, Investigation. **Arpan Mukhopadhyay:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Conceptualization. **Thirupathaiah Vasantam:** Writing – review & editing, Writing – original draft, Methodology, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Sanidhay Bhambay, Arpan Mukhopadhyay, Thirupathaiah Vasantam, The power of two choices with load comparison errors, in: Proceedings of the Twenty-Fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, MobiHoc '23, Association for Computing Machinery, New York, NY, USA, ISBN: 9781450399265, 2023, p. 121–130, http://dx.doi.org/10.1145/3565287.3610259, https://doi.org/10.1145/3565287.3610259.

[2] W. Tarreau, Test driving power of two random choices load balancing, 2019, URL https://www.haproxy.com/blog/power-of-two-load-balancing/.

[3] O. Garrett, Nginx and the 'power of two choices' load-balancing algorithm, 2018, URL https://www.nginx.com/blog/.

[4] M. Mitzenmacher, The Power of Two Choices in Randomized Load Balancing (Ph.D. thesis), University of California at Berkeley, 1996.

[5] N.D. Vvedenskaya, R.L. Dobrushin, F.I. Karpelevich, Queueing system with selection of the shortest of two queues: An asymptotic approach, Probl. Pereda. Inf. 32 (1) (1996) 20–34.

[6] Y. Lu, Q. Xie, G. Kliot, A. Geller, J.R. Larus, A. Greenberg, Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services, Perform. Eval. 68 (11) (2011) 1056–1071, http://dx.doi.org/10.1016/j.peva.2011.07.015.

[7] M. Smith, Rethinking netflix's edge load balancing, 2018, URL https://netflixtechblog.com/netflix-edge-load-balancing-695308b5548c.

[8] X. Zhou, N. Shroff, A. Wierman, Asymptotically optimal load balancing in large-scale heterogeneous systems with multiple dispatchers, Perform. Eval. 145 (2021) Art–No.

[9] D. Los, T. Sauerwald, Balanced allocations with the choice of noise, in: Proceedings of the 2022 ACM Symposium on Principles of Distributed Computing, 2022, pp. 164–175.

[10] D. Alistarh, T. Brown, J. Kopinsky, J.Z. Li, G. Nadiradze, Distributionally linearizable data structures, in: Proceedings of the 30th on Symposium on Parallelism in Algorithms and Architectures, SPAA '18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 133–142, http://dx.doi.org/10.1145/3210377.3210411.

[11] G. Nadiradze, On Achieving Scalability Through Relaxation (Ph.D. thesis), Institute of Science and Technology Austria, 2021.

[12] Q. Burke, P. McDaniel, T. La Porta, M. Yu, T. He, Misreporting attacks against load balancers in software-defined networking, Mob. Netw. Appl. (2024) 1–16.

[13] D. Los, T. Sauerwald, Balanced allocations with incomplete information: The power of two queries, in: M. Braverman (Ed.), 13th Innovations in Theoretical Computer Science Conference, ITCS 2022, in: Leibniz International Proceedings in Informatics (LIPIcs), vol. 215, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2022, pp. 103:1–103:23, http://dx.doi.org/10.4230/LIPIcs.ITCS.2022.103, URL https://drops.dagstuhl.de/opus/volltexte/2022/15699.

[14] Y. Azar, A.Z. Broder, A.R. Karlin, E. Upfal, Balanced allocations, in: Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 1994, pp. 593–602.

[15] P. Berenbrink, A. Czumaj, A. Steger, B. Vöcking, Balanced allocations: The heavily loaded case, SIAM J. Comput. 35 (6) (2006) 1350–1385, http://dx.doi.org/10.1137/S009753970444435X, arXiv:https://doi.org/10.1137/S009753970444435X.

[16] M. Bramson, Stability of join the shortest queue networks, Ann. Appl. Probab. 21 (4) (2011) 1568–1625.

[17] S. Foss, N. Chernova, On the stability of a partially accessible multi-station queue with state-dependent routing, Queueing Syst. 29 (1998) 55–73.

[18] M. Bramson, Y. Lu, B. Prabhakar, Asymptotic independence of queues under randomized load balancing, Queueing Syst. 71 (3) (2012) 247–292.

[19] M. Bramson, Y. Lu, B. Prabhakar, Randomized load balancing with general service time distributions, ACM SIGMETRICS Perform. Eval. Rev. 38 (1) (2010) 275–286.

[20] H. Chen, H.-Q. Ye, Asymptotic optimality of balanced routing, Oper. Res. 60 (1) (2012) 163–179.

[21] S.T. Maguluri, R. Srikant, L. Ying, Heavy traffic optimal resource allocation algorithms for cloud computing clusters, Perform. Eval. 81 (2014) 20–39.

[22] D. Mukherjee, S.C. Borst, J.S. Van Leeuwaarden, P.A. Whiting, Asymptotic optimality of power-of-d load balancing in large-scale systems, Math. Oper. Res. 45 (4) (2020) 1535–1571.

[23] A. Budhiraja, D. Mukherjee, R. Wu, Supermarket model on graphs, Ann. Appl. Probab. 29 (3) (2019) 1740–1777.

[24] D. Rutten, D. Mukherjee, Load Balancing under Strict Compatibility Constraints, SIGMETRICS '21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 51–52, http://dx.doi.org/10.1145/3410220.3456275.
[25] Z. Zhao, D. Mukherjee, R. Wu, Exploiting data locality to improve performance of heterogeneous server clusters, arXiv preprint arXiv:2211.16416.
[26] A. Mukhopadhyay, R.R. Mazumdar, Analysis of randomized join-the-shortest-queue (jsq) schemes in large heterogeneous processor-sharing systems, IEEE Trans. Control Netw. Syst. 3 (2) (2015) 116–126.
[27] A. Mukhopadhyay, A. Karthik, R.R. Mazumdar, Randomized assignment of jobs to servers in heterogeneous clusters of shared servers for low delay, Stoch. Syst. 6 (1) (2016) 90–131, http://dx.doi.org/10.1287/15-SSY179, arXiv:https://doi.org/10.1287/15-SSY179.
[28] N. Gast, B. Van Houdt, A refined mean field approximation, Proc. ACM Meas. Anal. Comput. Syst. 1 (2) (2017) 1–28.
[29] F. Kelly, E. Yudovina, Stochastic Networks, Cambridge University Press, 2014, http://dx.doi.org/10.1017/CBO9781139565363.
[30] P.W. Glynn, A. Zeevi, Bounding stationary expectations of markov processes, in: Markov Processes and Related Topics: A Festschrift for Thomas G. Kurtz, Institute of Mathematical Statistics, 2008, pp. 195–214.
[31] R. Srikant, L. Ying, Communication Networks: An Optimization, Control, and Stochastic Networks Perspective, Cambridge University Press, 2013.
[32] S.N. Ethier, T.G. Kurtz, Markov Processes: Characterization and Convergence, Vol. 282, John Wiley & Sons, 2009.
[33] T.G. Kurtz, Approximation of Population Processes, SIAM, 1981.
[34] K. Deimling, Ordinary Differential Equations in Banach Spaces, Vol. 596, Springer, 2006.
[35] T. Hellemans, B. Van Houdt, Heavy traffic analysis of the mean response time for load balancing policies in the mean field regime, arXiv preprint arXiv:2004.00876.

Sanidhay Bhambay is currently a Ph.D student at University of Warwick, UK. He received the B.Tech. degree in electrical engineering from the J.S.S. Academy of Technical Education, Noida, India, in 2014, and the M.Tech. (Research) degree in electrical engineering from the Indian Institute of Science, Bengaluru, India, in 2018. His research interests include queuing theory, performance analysis of large complex networks, applied proability, and optimization.

Arpan Mukhopadhyay received the B.E. degree in electronics and telecommunication engineering from Jadavpur University, Calcutta, India, in 2009, the M.E. degree in telecommunications from the Indian Institute of Science, Bengaluru, India, in 2011, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Canada, in 2016. He is currently an Assistant Professor with the Department of Computer Science, University of Warwick, U.K. Before joining the University of Warwick, he spent two years with EPFL, Switzerland, and one year with INRIA, Paris, as a Post-Doctoral Researcher. His research interests include applied probability, stochastic processes, algorithm design, and optimization with applications to cloud computing systems, caching systems, wireless networks, social networks, and smart grids. He was a recipient of Best Paper Awards from the IFIP Performance 2015 Conference and the International Teletraffic Congress (ITC) 2015, and he received the Rising Scholar Award at the International Teletraffic Congress 2018 for his contributions to mean field analysis of large heterogeneous networks.

Thirupathaiah Vasantam is an Assistant Professor of Computer Science at Durham University, UK. Before that, he has worked as a Postdoc in the College of Information and Computer Sciences at the University of Massachusetts, Amherst, USA. He received his PhD from the department of Electrical and Computer Engineering at the University of Waterloo, Canada, in November 2019. He obtained his Bachelors and Masters degrees in Engineering from Osmania University and Indian Institute of Science, Bangalore, respectively. His research focuses on performance modeling and analysis of communication networks with applications to Cloud computing, Internet of Things, Datacenters, Sensor networks, and Quantum networking.