Mean Field Methods for Heterogeneous and Coupled Systems

Sebastian Allmeier

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et technologies de l'information, Informatique
Spécialité : Mathématiques Appliquées
Unité de recherche : Centre de recherche Inria de l'Université Grenoble Alpes

Méthodes de Champ Moyen pour Systèmes Hétérogènes et Couplés

Mean Field Methods for Heterogeneous and Coupled Systems

Présentée par :

Sebastian ALLMEIER

Direction de thèse :

Nicolas GAST CHARGE DE RECHERCHE HDR, CENTRE INRIA UNIVERSITE GRENOBLE ALPES	Directeur de thèse

Rapporteurs :

CHRISTINE FRICKER
CHARGEE DE RECHERCHE HDR, CENTRE INRIA DE PARIS
YING LEI
FULL PROFESSOR, UNIVERSITY OF MICHIGAN

Thèse soutenue publiquement le **15 avril 2024**, devant le jury composé de :

JERÔME LELONG, PROFESSEUR DES UNIVERSITES, GRENOBLE INP	Président
NICOLAS GAST, CHARGE DE RECHERCHE HDR, CENTRE INRIA UNIVERSITE GRENOBLE ALPES	Directeur de thèse
CHRISTINE FRICKER, CHARGEE DE RECHERCHE HDR, CENTRE INRIA DE PARIS	Rapporteure
YING LEI, FULL PROFESSOR, UNIVERSITY OF MICHIGAN	Rapporteur
BENNY VAN HOUDT, FULL PROFESSOR, UNIVERSITEIT ANTWERPEN	Examinateur
ERIC LUÇON, MAITRE DE CONFERENCES, UNIVERSITE PARIS CITE	Examinateur

Acknowledgments

First and foremost, I want to express my sincere gratitude to Nicolas Gast. Without you, this work would not have been possible. The scientific guidance and opportunities provided made my Ph.D. a memorable experience, which I will always recall with great pleasure.

I would also like to thank the Ph.D. defense committee: Jerôme Lelong, Benny Van Houdt, and Eric Luçon with special thanks to Christine Fricker and Lei Ying for their valuable feedback.

A heartfelt appreciation to the POLARIS team, which made the atmosphere in the lab pleasant and stimulating. In particular, I want to thank my colleagues Romain, Louis-Sébastien, Victor, Davide, Dheeraj, Kimang, Chen, Vitalii, and Till for all the enjoyable moments we shared.

I am immensely grateful to my family, especially my parents, Sigird and Helmut, my grandparents Emilia and Artur, and my sister Lara, for their support, advice, and raising me with such love and care. At last, I want to thank Viktoria for all the memorable moments we shared.

Abstract

Mean Field methods have consistently been of interest to the scientific community due to their capacity to approximate a wide array of stochastic population systems. This methodology has proven to be a cornerstone in understanding and predicting the behavior of large-scale, complex systems. The key idea of the mean field approximation is to replace the complex stochastic behavior of systems with a simpler deterministic counterpart. The approximation therefore assumes that individuals become increasingly independent for large system sizes. The behavior of the system is thus obtained by replacing individual interactions with the average state of individuals. Despite its longstanding application and the advancements made in various fields, numerous questions and challenges remain open.

In the scope of this thesis, we present our contributions and advancements for two general types of population models (1) heterogeneous mean field models and (2) mean field models with a fast-changing environment.

In the first part of the thesis, we focus on stochastic systems with heterogeneous components. We consider two types of heterogeneity, individual-level diversity as well as graph-structured interactions. For both cases, we provide accuracy bounds for the expected state of finite-sized systems. The results are supported through practical examples, including cache replacement policies, supermarket models, load balancing, and bike-sharing systems, highlighting their computational tractability and precision. In the case of individual-level diversity, we further adapt the refined mean field idea and show that the refined approximation significantly reduces the error and provides accurate predictions for small to moderate-sized systems.

In the second part of the thesis, we turn our focus to mean field approximation techniques for stochastic systems with a coupled, fast-changing environment. By studying the 'average' mean field approximation, we obtain accuracy bounds for the expected system state. Furthermore, we derive a bias refinement term, which increases the accuracy of the approximation. Expanding on these results, we extend the methodology to stochastic approximation with constant step size and state-dependent Markovian noise. We show how to adapt the ideas to obtain accuracy results and a computable bias extension.

Résumé

Les méthodes de champ moyen permettent de construire des approximations de modèles aléatoires d'évolution de populations. Ces méthodes sont très utilisées pour comprendre et prédire le comportement de systèmes complexes. L'idée clé de l'approximation de champ moyen est de remplacer le comportement stochastique d'un système par un équivalent déterministe plus simple. L'approximation consiste à considérer que, dans un système de grande taille, les dépendances directes entre individus sont faibles. On obtient donc un modèle plus simple en considérant qu'un individu interagît non pas avec d'autres individus mais avec une masse d'individus qui représente leur état moyen. Dans de nombreuses applications, l'erreur de l'approximation champ moyen est en générale très faible lorsque la population est grande. Bien qu'il existe de nombreux travaux théoriques fournissant des bornes d'erreurs, de nombreuses questions et défis restent ouverts.

Dans cette thèse, nous présentons nos contributions et avancées pour deux types de modèles de populations: (1) les modèles de population hétérogènes et (2) les populations qui évoluent dans un environnement qui change rapidement.

Dans la première partie de la thèse, nous nous concentrons sur les populations hétérogènes. Nous considérons deux types d'hétérogénéité : la diversité au niveau des individus ainsi que les interactions contraintes par des graphes. Pour les deux cas, nous fournissons des bornes d'erreur pour les populations de taille finie. Ces résultats sont étayés par des exemples pratiques, comme des politiques de remplacement de cache, d'équilibrage de charge et des systèmes de partage de vélos. Ceci montre la faisabilité pratique du calcul des approximations ainsi que leurs précisions. Dans le cas de la diversité au niveau des individus, nous adaptons l'idée de champ moyen raffiné et démontrons que l'approximation raffinée réduit considérablement l'erreur et fournit des prédictions précises pour des systèmes de petite à moyenne taille.

Dans la seconde partie de la thèse, nous orientons notre attention vers les techniques d'approximation champ moyen pour les systèmes stochastiques qui évoluent dans un environnement à changement rapide. En étudiant l'approximation de champ moyen 'moyenne', nous obtenons des bornes précises sur la qualité de l'approximation. De plus, nous dérivons un terme de raffinement de biais, qui augmente la précision de l'approximation. Ces résultats sont ensuite étendus pour étudier des algorithmes d'approximation stochastique avec un pas constant et un bruit Markovien dépendant de l'état. Nous montrons comment adapter les idées pour obtenir des bornes d'erreur précises et calculer le biais asymptotique de l'approximation.

Contents

Part I

Introduction & Methodology

1 Introduction

Contents

1.1 Motivation

The study and understanding of stochastic systems which model real world phenomena is invaluable to predict, optimize and analyze environmental, public and economic effects. Examples include the spread of diseases, the transmission of information in computer systems, the reaction of chemical components, the processing of work in server centers or the behavior of bike sharing systems. Take the case of a data center which provides services to customers, understanding the performance of the system in terms of responsiveness and power consumption can have a critical impact on its environmental and economic sustainability. In order to better understand and capture the behavior of systems consisting of large numbers of interacting individuals (e.g. machines, objects, agents) stochastic population models are often used to describe their dynamics. The core unit of the population model is the individual. Each individual is described by a state, which it changes either independently or by interaction with other individuals. While population models can be suitably applied in many cases, their analysis often suffers from what is called the curse of dimensionality. The curse of dimensionality is characterized by an exponentially increasing complexity cause by the large number of interacting individuals. This can make its direct study time and resource consuming, if not prohibitive. Hence, it creates the need for approximation methods which strike a balance between the reduction of complexity and the preservation of accuracy. The mean field approximation is one such approach that has been proven universally useful to study the behavior of stochastic population models. Its fundamental idea is to replace interactions between the individuals of the stochastic system by interactions with the average state of individuals to obtain a deterministic description of the dynamics. This replacement is justified since individuals become increasingly uncorrelated in large system.

One of the striking aspects of the mean field approach is its capability to be adapted to different contexts and models. Its general idea has been picked up and developed in a broad range of domains such as game theory [80], biological neural networks [9, 98], artificial neural networks [84], multi-agent reinforcement learning [112, 115], load balancing [45, 89], wireless network analysis [47] to name a few. In this work, our focus will be on the application to pure jump Markov processes

which model interacting individuals such as density dependent population processes (DDPP) as in the work of Kurtz [76]. A classical result for DDPPs is that, for an increasing number of individuals, the trajectories of the stochastic system converge almost surely to the mean field approximation [17, 76]. To explain the accuracy of the approximation, the same studies show that the expected distance between the stochastic system and the approximation is inversely proportional to the square root of the number of individuals. Yet, this does not fully characterize the often faster diminishing error of the approximation, particularly when interested in expected state of the system. To study the accuracy of the mean field approximation and the expected state of a stochastic population system, we rely on a set of tools with Stein's method being one of the most prominent. Stein's method is used to derive bounds on the distance between distributions such as for stationary distributions of Markov processes [12, 102]. Lately, its application in the stochastic system context has seen a resurgence, initiated by the works [29, 31]. In [55, 74, 113, 114], the authors successfully apply the method to bound the difference between the generators the stochastic population model and its mean field approximation. A second building block of our results is the bias extension proposed by the authors of [54, 59] from which the 'refined' mean field approximation is obtained. Loosely speaking, the refined approximation is obtained by studying an expansion of the difference between the generators of the stochastic process and the mean field approximation. Under sufficient regularity, this allows to deduce computable correction terms which are system-size independent and significantly increase the accuracy of the approximation. A similar approach has been studied in mathematical biology which is know as system size expansion [62, 63, 109]. For the analysis of the mean field accuracy, it is often assumed [17, 55, 88, 113] that the population models fulfill two critical criteria: (1) individuals behave homogeneously, i.e., any permutation of individuals will not change the properties of the system. This implies statistically identical behavior of the individuals and dense, symmetric connectivity within the system. (2) There is no interference or coupling with other systems or environments impacting the dynamics.

In real-world applications, these assumptions often fail and the mean field approximation becomes inaccurate up to a point where it might be rendered useless to provide any valuable insights. We illustrate this using the well known Join-the-Shortest-Queue(d) (JSQ(d)) model for $d = 2$ [89, 110] as visualized in Figure 1.1. Consider customers arriving as a Poisson stream at a collection of servers. Each customer considers a subset of two servers uniformly at random upon arrival and proceeds to the server with the fewer number of customers in their queue with ties broken at random. At the queue, the customers are served in the first-come-first-serve manner. Such load balancing systems can be found in a broad range of applications and become increasingly important not least by their application to cloud networks and data centers. In order to predict and optimize their performance, the mean field approximation is a common tool used to gain valuable insight into the behavior. However, in application many systems are neither homogeneous nor act in isolation from other systems which can lead to imprecise performance estimations. In Figure 1.2, we illustrate a load balancing system following the same procedure as before but with queues having different service speeds. Not considering the heterogeneity at hand e.g., using uniform queue speeds to construct the mean field approximation, leads to under- or over-estimation of the performance which subsequently creates idle or overloaded queues. In Figure 1.3, we illustrate the impact of interfering systems. The load balancing example consists of three systems, which we call classes. Each class has an independent decision process for arriving customers and its own separated set of queues with

respective speed distinguished in the figure by color. To leave the system, customers needs to be processed by a medium that is shared among the three classes. The case that different queues simultaneously send customers to the medium gives rise to interference and consequently the loss of throughput. In the figure, the interference structure is illustrated by the arrows connecting the classed to the shared medium. Ignoring the impact of interference between classes on the dynamics leads to over-estimation of the systems performance.

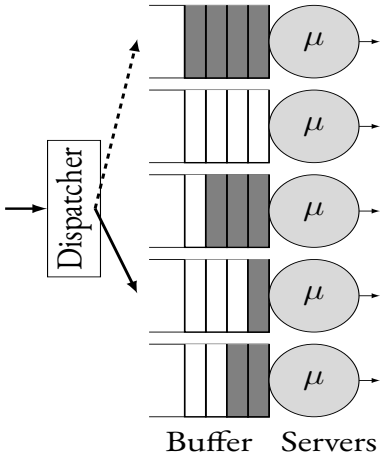


Figure 1.1: Homogeneous System

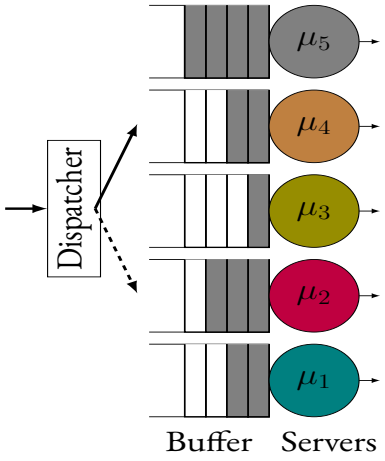


Figure 1.2: Heterogeneous System

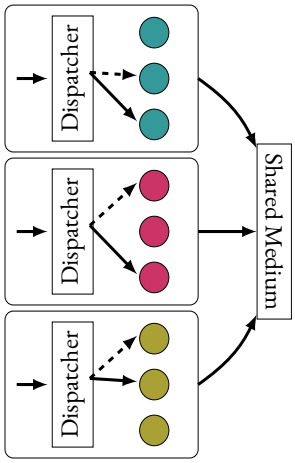


Figure 1.3: Interfering Systems

In this thesis, we present our work on the study of heterogeneous systems and systems with rapidly changing environments which contribute towards answering the two following grand questions:

- To which population models can we adapt mean field approximation and what accuracy can it provide?
- Can we increase the accuracy of the approximation by developing bias correction terms?

1.2 Overview

We present our results in three segments also visually presented in Figure 1.4. The introduction and methodology span the first part of this thesis. In the latter chapter, we present the general methodology which is consistently used to obtain accuracy bounds between the expected state of the considered stochastic systems and the deterministic counterpart, their mean field approximations. In the second part of the thesis, we study heterogeneous models. Chapter three presents a framework to study individual-level heterogeneity and provide accuracy bounds for the mean field approximation and its refinement. The next chapter presents the RMF Tool, a numerical toolbox designed to support the implementation of mean field models and their refinements. This includes the implementation of heterogeneous population models as studied in Chapter 3. The last chapter of the first

part provides new and unpublished results on graph based population models and their 'graphon' mean field approximation.

In the second part of the thesis, we shift the focus to systems with rapidly changing environments. In Chapter 6 we show that the mean field approximation obtained by an averaging principle has similar accuracy as in the homogeneous case. We further derive a new bias correction term which decodes the error made by averaging. In the last chapter, we explore how the ideas of the aforementioned paper can be transferred to stochastic approximation with state-dependent Markovian noise and constant stepsize. We prove new and currently unpublished accuracy results for the stochastic approximation setting.

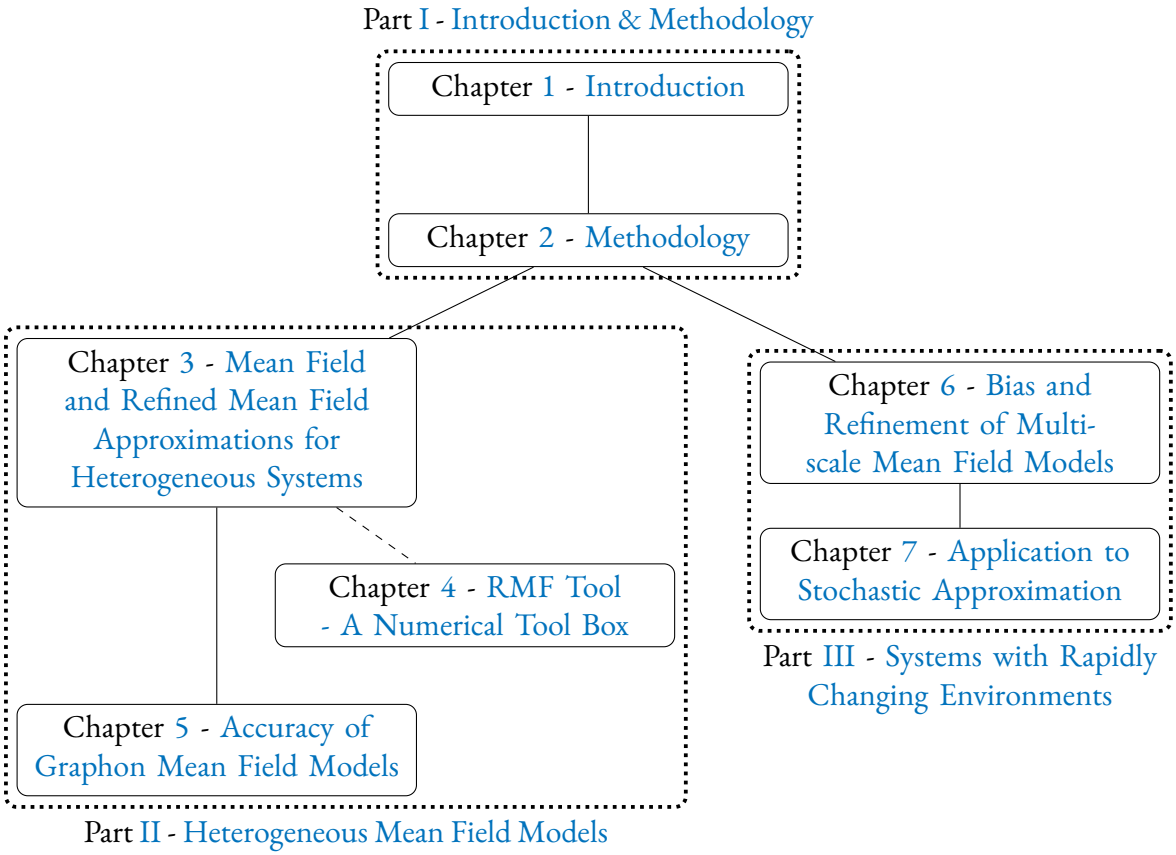


Figure 1.4: Structure of the Thesis

1.3 Detailed Contributions

Part I consists of the introduction and methodology. In this part of the document, we give an overview of the results presented in the thesis as well as a general introduction to the methodology used in the subsequent chapters.

Chapter 2 presents the general methodology to obtain bounds for the distance between the expectation of the stochastic system and its mean field approximation. This methodology is a cor-

nerstone for the results in the following chapters. The first part of the chapter is held general and introduces necessary tools such as generators, Poisson equations and points out properties of the dynamic systems needed to obtain the results. We then show how accuracy bounds for general population systems can be obtained. In the second part, we show how the first order refinement term for population processes can be derived. The refinement is an analytical expression used to increase the accuracy of the mean field approximation. In this part of the Chapter, the analysis is restricted to density dependent population processes.

PART II of this thesis presents our results for systems with heterogeneous components, in particular we focus on two distinct types of heterogeneity: In Chapter 3, we develop a framework to handle individual-level diversity in population systems and derive accuracy bounds for the corresponding mean field and refined mean field approximations. In the following Chapter 4, we present the numerical toolbox 'RMF Tool' which is designed to support the implementation of (heterogeneous) mean field models and their refinements. The second type of heterogeneity, given by graph-structured interactions of individuals is discussed in Chapter 5. In both settings, our motivation stems from understanding and bounding the difference between the stochastic system and a suitable mean field approximation. We further reflect on the applicability of the approximation and numerical complexity, which we underline by including examples for cache replacement policies, load balancing and bike-sharing systems.

CHAPTER 3 develops a framework to handle individual-level heterogeneity in population systems. Based on this framework, mean field and refined mean field approximations are derived which show comparable accuracy results as for homogeneous scenarios. The main differences in the heterogeneous setup is that the ODEs approximate the behavior of single items instead aggregate quantities as the empirical measure process for the states. Our main results show that in this setting the mean field approximations accuracy remains inversely proportional to the object count N, of order $O(1/N)$. We adapt the refinement idea to obtain computable expressions which increase the accuracy of the approximation. The results are supported by two examples, a heterogeneous load balancing system and a RANDOM(m) caching system. Both examples show the computational tractability of our approach and its accuracy in application.

CHAPTER 4 introduces the RMF Tool, a numerical toolbox designed to ease the computation of mean field approximations and their refinements. This chapter gives an overview of the python toolbox which was developed to overcome the computational challenges posed by the more intricate analytical expressions of the refined mean field approximation. The goal of the toolbox is that users input the description of a system and retrieve the approximations and additional refinements by using the functions provided by the toolbox. The RMF Tool provides this functionality for three kinds of models: (1) homogeneous population process; (2) density dependent processes; (3) heterogeneous population models. Here, the last type of models is the same as the ones discussed in the previous chapter.

CHAPTER 5 shifts the focus to graphon mean field models, where we consider systems of N particles interconnected through a dense, possibly weighted graph G^N. Each individual is a node

of the graph with internal state. The state of a particle changes according to its dependency on the states of its neighbors which is specified through the graph structure. We show that if the number of neighbors of the node grows with N, the behavior of the system converges to a deterministic limit, the graphon mean field approximation. In our results, we state precise convergence bounds that depend on the cut-norm distance between G^N and G. To illustrate that the convergence strongly depends on the sampling methods of G^N, we consider two cases: First, when G^N is a random graph obtained though sampling from the graphon G. Second, when G^N is a discretization of the graph G with individually weighted edges. To illustrate the applicability and numerical tractability of the results, we give two examples: A load balancing model where each individual describes a server-dispatcher pair with connections based on a sampled graph and a heterogeneous bike sharing system where the graph specifies the varying popularity of the stations.

PART III focuses on mean field techniques for stochastic systems with rapidly changing environment. In Chapter 6, we look at coupled population systems of 'slow-moving' populations and rapidly changing environments. In this setting, we utilize the 'averaged' mean field approximation and derive a refined 'average' mean field approximation to study the behavior of the stochastic system and characterize the accuracy of the approximation. In the second Chapter 7, we expand on these results, extending the methodology to stochastic approximation with constant step size and state-dependent Markovian noise. We show how to adapt the previous ideas to this setting to obtain accuracy results and a computable bias extension.

CHAPTER 6 analyzes the approximation error of the 'average' mean field model for a two-timescale system (X, Y), where the slow component X describes a population of N interacting individuals which is fully coupled with a rapidly changing environment Y. We show that under relatively mild conditions, the 'average' mean field approximation has a bias of order $O(1/N)$ in both, the transient regime and the steady-state. To go a step further, we derive a bias correction term for the steady-state from which we define a refined 'average' mean field approximation whose bias is of order $O(1/N^2)$. We illustrate the developed framework and accuracy results through an application to a random access CSMA model.

CHAPTER 7 adapts the methodology and ideas of the previous chapter to the stochastic approximation setting. We show that for a stochastic approximation with constant step size and state-dependent Markovian noise comparable accuracy results can be achieved. In particular, the chapter we prove that given a constant step size α, the asymptotic bias of the stochastic approximation to the equilibrium the related 'averaged' deterministic system is of order α. We further show how to obtain a computable first order bias extension.

1.4 List of Publications

4. S. Allmeier and N. Gast. "Bias and Refinement of Multiscale Mean Field Models". *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 7:1, 2023, 23:1–23:29. DOI: 10.1145/3579336.
5. S. Allmeier and N. Gast. "Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works!" *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 6:1, 25, 2022, 13:1–13:43. DOI: 10.1145/3508033.
6. S. Allmeier and N. Gast. "Rmf Tool - A Library to Compute (Refined) Mean Field Approximation(s)". *ACM SIGMETRICS Performance Evaluation Review* 4, 2, 2022, pp. 35–40. ISSN: 0163-5999. DOI: 10.1145/3543146.3543156.

Above is a list of publications which I have co-authored during my PhD and can be found in the content of this thesis: Chapter 3 is based on [5], Chapter 4 on [6] and Chapter 6 on [4]. The remaining chapters contain new and currently unpublished results.

2 Methodology

Contents

2.1 Introduction

In this section we present the general methodology, which is used to obtain bounds for the bias between the expectation of the stochastic system and its mean field approximation. We aim to give an overview of the reasoning and the analytical tools developed in the papers [54, 55, 59, 114]. While the mentioned papers predominantly focus on density dependent population processes (as introduced in Section 2.4.1) the methodology is not limited to those. For example, in [4, 5], we show how the framework can be extended to different setting such as heterogeneous population models or two timescale population processes. To capture this aspect, the first part of this chapter is held fairly general and aims to be applicable to generic population processes. In the later parts, particularly to obtain accuracy bounds and refinement terms, we will restrict ourselves to density dependent population processes. We decide on this step, to keep balance between readability and generality of the later sections. To summarize, in this chapter:

1. We illustrate how to express the difference between the expected state of a stochastic population process and its mean field approximation using the generators associated to their dynamic system description in Section 2.3.
2. We utilize the specific generator expressions for density dependent population processes to:
   - Establish bounds on the difference between the two dynamical systems, validating the accuracy of the mean field approximation in Section 2.4.3.
   - Study the bias of the approximation and to develop refinement terms which increase the accuracy of the mean field approximation in Section 2.4.4.

We start by specifying the concept of population processes before progressing to the general methodology and subsequent results.

2.2 CTMCs, Semi-Groups and Generators

In the following section, we recall essential properties of continuous time Markov chains, the definition of strongly continuous semi-groups with their generators and show how such semi-groups can be defined to describe the evolution of the dynamical systems in question. We also give a short illustration of the former by applying these definition to a simple load balancing example.

2.2.1 Continuous Time Markov Chains (CTMC)

For the sake of readability and completeness, we briefly recall the quintessential properties of time homogeneous continuous time Markov chains (CTMC). For a more extensive introduction, the reader is referred to [103]. It is furthermore assumed that the reader is familiar with the Markov property for stochastic processes. In the thesis, for all encountered frameworks and population processes, we consider a population process with finite state space $\mathcal{X}$ of size $K \in \mathbb{N}$. Here and in the following, the stochastic process will be denoted by $\{\boldsymbol{X}(t), t \geq 0\}$. The Markov property implies that for any states $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and times $s \geq 0$ and $t \geq u \geq 0$

$$\mathbb{P}(\boldsymbol{X}(t+s) = \boldsymbol{x} \mid \boldsymbol{X}(t) = \boldsymbol{y}, \boldsymbol{X}(u) = \boldsymbol{z}) = \mathbb{P}(\boldsymbol{X}(t+s) = \boldsymbol{x} \mid \boldsymbol{X}(t) = \boldsymbol{y}).$$

To define the transitions of a CTMC, it is more common to transition state the transition rates between states rather than transition probabilities. In the following, we will denote by $r_{\boldsymbol{y}-\boldsymbol{x}}(\boldsymbol{X}(t))$ the rate per unit time of the CTMC to jump from state $\boldsymbol{x}$ to $\boldsymbol{y}$. We choose this notation as it will be convenient for the subsequent discussions to differ between rates based on the current state of the system $\boldsymbol{X}^{(N)}(t)$ and a set of possible jumps of the system. The relations between the transition rates and probabilities is given by

$$r_{\boldsymbol{y}-\boldsymbol{x}}(\boldsymbol{X}(t)) = \lim_{dt \downarrow 0} \frac{1}{dt} \mathbb{P}(\boldsymbol{X}(t+dt) = \boldsymbol{y} \mid \boldsymbol{X}(t) = \boldsymbol{x}) \qquad \text{for} \qquad \boldsymbol{x} \neq \boldsymbol{y}$$

and therefore

$$\mathbb{P}(\boldsymbol{X}(t+dt) = \boldsymbol{x} \mid \boldsymbol{X}(t) = \boldsymbol{y}) = r_{\boldsymbol{x}-\boldsymbol{y}}(\boldsymbol{X}(t))dt + o(dt).$$

We further define the rate for the process to remain in state $\boldsymbol{X}$ by

$$r_0(\boldsymbol{X}(t)) = -\sum_{x \neq y} r_{\boldsymbol{y}-\boldsymbol{x}}(\boldsymbol{X}(t)).$$

In Section 2.2.2 we will fall back on these properties to define the semi-group linked to the expectation of the stochastic process and its generator. This in turn will ultimately allow us to deduce the generator comparison between a deterministic differential equation, its mean field approximation, and the stochastic process.

Population Processes

As outlined before, we denote by $\{\boldsymbol{X}^{(N)}(t), t \geq 0\}$ a stochastic population process of population size $N \in \mathbb{N}$. The state space of the system $\boldsymbol{X}^{(N)}(t)$ is denoted by $\mathcal{X}^N \subset \mathbb{R}^d$. To preserve readability and explanatory intention of this chapter, we restrict the dimension of the state space to be independent of the population size. In many cases this is a reasonable assumption and holds true for a broad range of stochastic processes, e.g. density dependent population processes as defined in Section 2.4.1. In later chapters however, we will discuss settings for which the dimension can vary with N.

2.2.2 Semi-Groups and Generators for Dynamics Systems

In this subsection, we introduce the notion of semi-groups and their connected generators. We start by giving the general definition for the latter and continue by representing the stochastic and deterministic system using the semi-groups notation. Ultimately, the representation will allow us to express the difference between the two dynamical systems by their generators which is discussed in detail in Section 2.3.

Definition 1 (Definition and properties of C_0 Semi-Groups and their Generators)**.** *A family of linear operators $\{T_s : \mathcal{D} \to \mathcal{D}, s \geq 0\}$ is called a strongly continuous semi-group (or C_0-semi-group) on a Banach space $\mathcal{D}$ if it has the following properties:*

$$T_0 = Id, \tag{2.1}$$
$$T_{s+t} = T_s T_t \qquad \textit{for all} \quad s, t \geq 0, \tag{2.2}$$
$$\lim_{t \downarrow 0} T_t \boldsymbol{z} = \boldsymbol{z} \qquad \textit{for all} \quad \boldsymbol{z} \in \mathcal{D}. \tag{2.3}$$

The generator of a C_0-semi-group, if existent for $\boldsymbol{z}$, is defined by

$$A\boldsymbol{z} = \lim_{dt \downarrow 0} \frac{1}{dt}(T_{dt}\boldsymbol{z} - T_0\boldsymbol{z}). \tag{2.4}$$

A direct consequence of the definition of the generator and property (2.2), is its commutation with the defining C_0-semi-group, i.e.,

$$AT_s h(\boldsymbol{z}) = T_s A h(\boldsymbol{z}) = \frac{d}{ds} T_s h(\boldsymbol{z}). \tag{2.5}$$

This property can easily be proven by considering that by definition $T_s(T_s - Id)h(\boldsymbol{z}) = T_s(T_s - Id)h(\boldsymbol{z})$, with Id being the identity map. Dividing both sides by s and letting s go to zero shows the property. As we will be only working with C_0-semi-groups they are simply referred to as semi-groups in the following.

Semi-Group and Generator for the Stochastic System

We now want to show how to represent the stochastic system $\{\boldsymbol{X}^{(N)}(t), t \geq 0\}$ using the semi-group notation. Recall that the dynamics of the process are given by a state independent set of transition vectors $\mathcal{L}^N$ and their respective transition rates $r_\ell(\boldsymbol{x}), \ell \in \mathcal{L}^N$ for $\boldsymbol{x}$ in the domain $\mathcal{X}^N$ of $\boldsymbol{X}^{(N)}$. Let $\mathcal{C}_{\mathcal{X}^N \to \mathbb{R}}$ be the set of continuously differentiable functions from $\mathcal{X}$ to $\mathbb{R}$ where $\mathcal{X}$ is the convex hull over all $\mathcal{X}^N$. We define the stochastic semi-group mapping on $\mathcal{C}_{\mathcal{X}^N \to \mathbb{R}}$ by

$$\Psi_t^N : \mathcal{C}^1_{\mathcal{X}^N \to \mathbb{R}} \to \mathcal{C}^1_{\mathcal{X}^N \to \mathbb{R}} \qquad \text{with} \qquad \Psi_t^N h(\boldsymbol{x}) := \mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}]. \tag{2.6}$$

Strictly speaking the Ψ_t can be defined for a much broader set of functions, however, we restrict the domain of Ψ_t to $\mathcal{C}^1_{\mathcal{X} \to \mathbb{R}}$ as in the following our analysis and bounds are based on the differentiability h. It is immediate by the definition of Ψ_t and the properties of the Markov chain that the above is indeed a semi-group. By Definition 1 - Equation (2.4), the generator is defined as $\lim_{dt \downarrow 0} \frac{1}{dt}(\Psi_{dt}^N h - \Psi_0^N h)$. For $\boldsymbol{x} \in \mathcal{X}^N$ we therefore have

$$L^N h(\boldsymbol{x}) := \lim_{dt \downarrow 0} \frac{1}{dt}(\Psi_{dt}^N h - \Psi_0^N h)(\boldsymbol{x}) = \lim_{dt \downarrow 0} \frac{1}{dt} \mathbb{E}[h(\boldsymbol{X}^{(N)}(dt)) - h(\boldsymbol{x}) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}].$$

We use the general properties of the Markov chain, as recalled in Section 2.2.1, to obtain a more detailed formulation of the generator. For small times $dt > 0$ we have

$$\begin{aligned}
&\mathbb{E}[h(\boldsymbol{X}^{(N)}(dt)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] \\
&\quad = h(\boldsymbol{x}) \mathbb{P}(\boldsymbol{X}^{(N)}(dt) = \boldsymbol{x} \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}) \\
&\qquad + \sum_{\ell \in \mathcal{L}^N} h(\boldsymbol{x} + \ell) \mathbb{P}(\boldsymbol{X}^{(N)}(dt) = \boldsymbol{x} + \ell \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x})
\end{aligned}$$

By using the rate dependent description of the transition probabilities, we see that the generator of the process is equal to

$$L^N h(\boldsymbol{x}) = \sum_{\ell \in \mathcal{L}^N} \big(h(\boldsymbol{x} + \ell) - h(\boldsymbol{x})\big) r_\ell(\boldsymbol{x}). \tag{2.7}$$

In the case that the stochastic system has a unique stationary distribution, the latter is denoted by $\boldsymbol{X}^{(N)}(\infty)$. To ease the notations for the steady-state and if clear from context, we will refrain from giving the initial state of the stochastic system, i.e., for $\Psi_\infty h(\boldsymbol{x})$ we write equivalently

$$\mathbb{E}[\boldsymbol{X}^{(N)}(\infty)] \qquad \text{instead of} \qquad \mathbb{E}[\boldsymbol{X}^{(N)}(\infty) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}]. \tag{2.8}$$

Semi-Group and Generator for the Deterministic System

This subsection aims to define the semi-group and generator for the mean field approximation given by the solution to the Cauchy problem

$$\frac{d}{dt}\boldsymbol{\phi}(t, \boldsymbol{x}) = \boldsymbol{f}\big(\boldsymbol{\phi}(t, \boldsymbol{x})\big) \qquad \text{with} \qquad \boldsymbol{\phi}(0, \boldsymbol{x}) = \boldsymbol{x} \in \mathcal{X}.$$

We define the semi-group by

$$\Phi_t : \mathcal{C}^1_{\mathcal{X}\to\mathbb{R}} \to \mathcal{C}^1_{\mathcal{X}\to\mathbb{R}} \qquad \text{with} \qquad \Phi_t h(\boldsymbol{x}) = h(\boldsymbol{\phi}(t, \boldsymbol{x})).$$

In words, the mapping associates to a continuously differentiable function h its evaluation at $\boldsymbol{\phi}(t, \boldsymbol{x})$. By definition of the deterministic system, it is directly follows that Φ_t is indeed a semi-group. To derive the corresponding generator, by definition

$$\Lambda h(\boldsymbol{x}) := \lim_{dt\downarrow 0} \frac{1}{dt}(h(\boldsymbol{\phi}(dt, \boldsymbol{x})) - h(\boldsymbol{\phi}(0, \boldsymbol{x}))) = D_{\boldsymbol{x}} h(\boldsymbol{x}) \cdot \boldsymbol{f}(\boldsymbol{x}). \tag{2.9}$$

with $\cdot$ being the scalar product between $D_x h(\boldsymbol{x})$ and $\boldsymbol{f}(\boldsymbol{x})$. It is furthermore worth pointing out that the commutation of the generator with its semi-group implies

$$\begin{aligned} D_{\boldsymbol{x}} h(\boldsymbol{\phi}(t, \boldsymbol{x}))\boldsymbol{f}(\boldsymbol{\phi}(t, \boldsymbol{x})) &= \Phi_t \Lambda h(\boldsymbol{x}) && (2.10)\\ &= \Lambda \Phi_t h(\boldsymbol{x}) = D_{\boldsymbol{x}}(h \circ \boldsymbol{\phi}(t, .))(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}). && (2.11) \end{aligned}$$

This property will be especially useful in the next sections as it allows to compare two introduced generators. Similar as for the stochastic system, we denote by

$$\boldsymbol{\phi}_\infty h(\boldsymbol{x}) = h(\boldsymbol{\phi}(\infty, \boldsymbol{x})) \tag{2.12}$$

the unique equilibrium point of the deterministic system if existent. If appropriate and clear from context, we further abbreviate the equilibrium point by $\boldsymbol{\phi}_\infty$.

2.3 General Methodology

In this section, we start the discussion of the general methodology to obtain accuracy results for the difference between the stochastic and deterministic setting. In the following, we will show how to state the difference of the two systems using their generators for the transient regime and steady-state. We will (informally) show:

For the transient regime

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t, \boldsymbol{x})) = \int_0^t \Psi_{t-\tau} \Delta^N \Phi_\tau h(\boldsymbol{x}) d\tau.$$

For the steady-state

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty))] - h(\phi_\infty) = \Psi_\infty \Delta^N \int_0^\infty \Phi_\tau h(\boldsymbol{x}) d\tau$$

with $\Delta^N = L^N - \Lambda$ representing the difference between the generators of the two systems.

2.3.1 Transient Regime

We aim to reformulate the difference

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t, \boldsymbol{x})) \tag{2.13}$$

for a sufficiently continuous differentiable function $h : \mathcal{X} \to \mathbb{R}$ and $0 \leq t < \infty$. To make use of the previously introduced semi-groups, first, recall their definitions

$$\Phi_t h(\boldsymbol{x}) = h(\boldsymbol{\phi}(t, \boldsymbol{x})) \qquad \text{and} \qquad \Psi_t^N h(\boldsymbol{x}) := \mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}].$$

Based on Ψ^N and Φ we define the mapping $\nu_\tau : \mathcal{C}^1_{\mathcal{X}\to\mathbb{R}} \to \mathcal{C}^1_{\mathcal{X}\to\mathbb{R}}$ for $\tau \in [0, t]$ with

$$\boldsymbol{\nu}_\tau h(\boldsymbol{x}) := \Psi_\tau^N \Phi_{t-\tau} h(\boldsymbol{x}) = \mathbb{E}[h\big(\boldsymbol{\phi}(t-\tau, \boldsymbol{X}^{(N)}(\tau))\big) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}]. \tag{2.14}$$

The evaluation of the mapping $\nu_\tau h(\boldsymbol{x})$ for a state $\boldsymbol{x}$ and function h consists of two steps. First, the expectation of the stochastic system evaluated at time τ. Second the continuation of the stochastic trajectory by the ODE in the time frame $(\tau, t]$ with initial condition $\boldsymbol{X}^{(N)}(\tau)$. An example trajectory of $\boldsymbol{\phi}(t-\tau, \boldsymbol{X}(\tau))$ is given in Figure 2.1. The definition of $\boldsymbol{\nu}_\tau$ now allows to rewrite

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t, \boldsymbol{X})) = \boldsymbol{\nu}_t h(\boldsymbol{x}) - \boldsymbol{\nu}_0 h(\boldsymbol{x}).$$

To use the properties of the generators their respective semi-groups, we want to rewrite

$$\boldsymbol{\nu}_t h(\boldsymbol{x}) - \boldsymbol{\nu}_0 h(\boldsymbol{x}) = \int_0^t \frac{d}{ds} \boldsymbol{\nu}_s h(\boldsymbol{x}) \Big|_{s=\tau} d\tau. \tag{2.15}$$

At this point we skip over the technical explanation of the equality above. It should be noted though, that it might need some thorough investigation and justification. Looking at the right-hand side, we

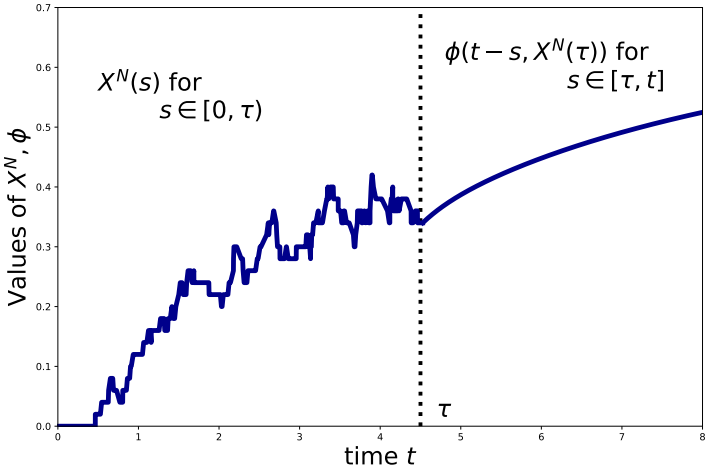


Figure 2.1: Sample trajectory for $\boldsymbol{\phi}(t-\tau, \boldsymbol{X}^{(N)}(\tau))$.

start by analyzing the derivative under the integral. We see that the difference $(\boldsymbol{\nu}_{\tau+d\tau} - \boldsymbol{\nu}_\tau)h(\boldsymbol{x})$ is equal to

$$\begin{aligned}
&\Psi^N_{\tau+d\tau}\Phi_{t-(\tau+d\tau)}h(\boldsymbol{x}) - \Psi^N_\tau\Phi_{t-\tau}h(\boldsymbol{x}) \\
&= \Psi^N_{\tau+d\tau}\Phi_{t-(\tau+d\tau)}h(\boldsymbol{x}) - \Psi^N_{\tau+d\tau}\Phi_{t-\tau}h(\boldsymbol{x}) \\
&\qquad + \Psi^N_{\tau+d\tau}\Phi_{t-\tau}h(\boldsymbol{x}) - \Psi^N_\tau\Phi_{t-\tau}h(\boldsymbol{x}).
\end{aligned}$$

Due to the assumption that $\boldsymbol{X}^{(N)}$ is a population process with finite state space, bounded transition rates and bounded jumps sizes, the above term is bounded for all $t, \tau, d\tau$. Thus, we can rewrite Equation (2.15) as

$$\lim_{d\tau \downarrow 0} \frac{1}{d\tau}(\boldsymbol{\nu}_{\tau+d\tau} - \boldsymbol{\nu}_\tau)h(\boldsymbol{x}) = L^N\Psi^N_\tau\Phi_{t-\tau}h(\boldsymbol{x}) - \Psi^N_\tau\Lambda\Phi_{t-\tau}h(\boldsymbol{x}). \tag{2.16}$$

Using the commutation of the generator for the stochastic process, i.e., $L^N\Psi^N_\tau = \Psi^N_\tau L^N$, as pointed out in the Definition 1 - Equation (2.5), and linearity, we have that Equation (2.16) is equal to $\Psi^N_\tau(L^N - \Lambda)\Phi_{t-\tau}h(\boldsymbol{x})$.

Finally, this allows us to express the difference between the expectation of the stochastic system and deterministic system by

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t, \boldsymbol{x})) = \int_0^t \Psi^N_\tau\big(L^N - \Lambda\big)\Phi_{t-\tau}h(\boldsymbol{x})d\tau. \tag{2.17}$$

This reformulation serves as basis for our further discussions on the accuracy and the bias analysis.

2.3.2 Steady-State extension

To extend the previous discussion to the steady state, assume that the stochastic process admits a unique stationary distribution and that the ODE has a unique equilibrium state to which it converges exponentially fast, see Definition 2. As given in the discussion of the semi-groups and generators the steady-state and equilibrium point, following our definitions in Equations (2.8) and (2.12), are denoted by

$$\Psi_\infty^N h(\boldsymbol{x}) = \mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty))] \qquad \text{and} \qquad \Phi_\infty h(\boldsymbol{x}) = h(\boldsymbol{\phi}_\infty).$$

For the steady-state we are looking at the difference $\mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty))] - h(\boldsymbol{\phi}_\infty)$ which is the limit of Equation (2.17). In the next steps we will show the equality

$$\lim_{t\to\infty} \int_0^t \Psi_{t-\tau}^N \big(L^N - \Lambda\big)\Phi_\tau h(\boldsymbol{x}) d\tau = \int_0^\infty \Psi_\infty^N \big(L^N - \Lambda\big)\Phi_\tau h(\boldsymbol{x}) d\tau. \tag{2.18}$$

which then serves as a basis for the following accuracy bounds and refinement observations. To give more intuition behind why this equality holds, define for any positive and increasing sequence $(t_k)_{k\in\mathbb{N}}, t_k \to \infty$ with the function

$$q_{t_k}(\tau) := \mathbf{1}_{[0,t_k]}(s)\Psi_{t_n-\tau}^N \Delta^N \Phi_s h(\boldsymbol{x})$$

with $\Delta^N := L^N - \Lambda$. Then, by the steady-state assumption on $\boldsymbol{X}^{(N)}$, $q_{t_k}(\tau)$ converges pointwise to $q(\tau) := \Psi_\infty^N \Delta^N \Phi_\tau h(\boldsymbol{x})$ for $k \to \infty$. By the regularity assumptions on the dynamical system, we see that the quantities $q_{t_k}(\tau)$ and $q(\tau)$ are bounded by

$$g(\tau) = \sup_{\boldsymbol{x}\in\mathcal{X}} \Delta^N \Phi_\tau h(\boldsymbol{x}).$$

To apply the dominated convergence theorem justifying $\lim_{n\to\infty} \int_0^\infty q_{t_n}(\tau)d\tau = \int_0^\infty q(v)d\tau$, we now argue that $\int_0^\infty |g(\tau)|d\tau$ is finite. By assuming, without loss of generality, that $h(\boldsymbol{\phi}_\infty) = 0$[1] we see that for any $\boldsymbol{x} \in \mathcal{X}$ the integral $\int_0^\infty \Delta^N \Phi_\tau h(\boldsymbol{x})d\tau$ is finite due to the exponentially fast convergence of the ODE to its equilibrium point. This can be checked by using the bounded domain and bounds on the derivative of the ODE. We further argue that the supremum in the definition of g is attained as it is taken over the convex hull $\mathcal{X}$ of closed sets. This implies $\int_0^\infty |g(\tau)|d\tau < \infty$ and therefore q_{t_n} converges to

$$\int_0^\infty \Psi_\infty^N \Delta^N \Phi_\tau h(\boldsymbol{x}) d\tau. \tag{2.19}$$

We finish by making the following informal statement which is serves as a intuitive basis for the steady-state results of the next section. Under sufficient regularity assumptions on the expectation

[1] One can always replace h by $h^*(\boldsymbol{x}) := h(\boldsymbol{x}) - h(\boldsymbol{\phi}_\infty)$ which yields the same conclusions.

of the stochastic system as well as by the definition and properties of the generators, we can move the integral of Equation (2.19) in front of Φ_s to obtain

$$\int_0^\infty \Psi_\infty \Delta^N \Phi_\tau h(\boldsymbol{x}) d\tau = \Psi_\infty \Delta^N \int_0^\infty \Phi_\tau h(\boldsymbol{x}) d\tau \tag{2.20}$$

which will be especially useful for the forthcoming discussions on the accuracy bounds and remainder terms.

For the following, we want to point how to approach the steady-state reformulation from a different angle by utilizing Poisson equations. Therefore, the first component to bound the difference between the two systems $\mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty))] - h(\infty)$ is to set up the Poisson equation

$$\Lambda G_h(\boldsymbol{x}) = h(\boldsymbol{x}) - h(\boldsymbol{\phi}_\infty) \tag{2.21}$$

for a 'nice' enough function h. A function G_h which satisfies the above equation is called the solution to the Poisson equation. As shown in [55, 114], a functions which satisfies the above is given by

$$G_h(\boldsymbol{x}) := -\int_0^\infty h(\boldsymbol{\phi}(\tau, \boldsymbol{x})) - h(\boldsymbol{\phi}_\infty) d\tau.$$

Note that the right hand side is well-defined due to the exponential stability of the deterministic system. In Lemma 3, we give a more precise justification. To verify that G_h solves the Poisson equation, by similar steps used to obtain Equation (2.19), we can (informally) argue that

$$\begin{aligned}
\Lambda G_h(\boldsymbol{x}) &= -\Lambda \int_0^\infty (\Phi_t h - \Phi_\infty h)(\boldsymbol{x})\, dt \\
&= -\int_0^\infty \Lambda(\Phi_t h - \Phi_\infty h)(\boldsymbol{x}) dt \\
&= -\int_0^\infty \Lambda \Phi_t h(\boldsymbol{x}) dt = -\int_0^\infty \frac{d}{dt} h(\boldsymbol{\phi}(t, \boldsymbol{x})) dt \\
&= h(\boldsymbol{x}) - h(\boldsymbol{\phi}(\infty, \boldsymbol{x})).
\end{aligned}$$

For the next step, we note that given the unique stationary distribution of the stochastic system,

$$\Psi_\infty^N L^N h(\boldsymbol{x}) = \mathbb{E}[L^N h(\boldsymbol{X}^{(N)}(\infty)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] = 0. \tag{2.22}$$

Generally speaking, the above equation holds for functions of compact support. By the bounded state space of the systems, it then also holds for the solution of the Poisson equation. Next, we use G to rewrite

$$\mathbb{E}[\Lambda G_h(\boldsymbol{X}^{(N)}(\infty))] = \mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty))] - h(\boldsymbol{\phi}_\infty).$$

lightly hidden under the notation, we implicitly assumed differentiability of G_h. As the justification is a rather technical argument based on the exponential stability of the ODE, more details are given

by Lemma 3. Using the property outlined in Equation (2.22), we can add $\mathbb{E}[L^N G_h(\boldsymbol{X}^{(N)}(\infty))]$ to the equation and obtain

$$\mathbb{E}[\Lambda G_h(\boldsymbol{X}^{(N)}(\infty))] = \mathbb{E}[(\Lambda - L^N) G_h(\boldsymbol{X}^{(N)}(\infty))] \tag{2.23}$$

$$= \mathbb{E}[(L^N - \Lambda) \int_0^\infty \Big(h(\boldsymbol{\phi}(\tau, \boldsymbol{X}^{(N)}(\infty))) - h(\boldsymbol{\phi}_\infty) \Big) d\tau]. \tag{2.24}$$

Definition 2 (Global Exponential Stability)**.** *A differential equation $\frac{d}{dt}\boldsymbol{\phi}(t, \boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{\phi}(t, \boldsymbol{x}))$ is called globally exponentially stable if it has a unique attractor $\boldsymbol{\phi}_\infty$ and if there exist constants $a, b \geq 0$ such that for all $\boldsymbol{x} \in \mathcal{X}$ and $t \geq 0$*

$$\|\boldsymbol{\phi}(t, \boldsymbol{x}) - \boldsymbol{\phi}_\infty\| \leq a \exp(-bt).$$

2.4 Application to Density Dependent Population Processes

Based on the previous generator reformulation of the difference between the expected state of the stochastic and deterministic system, summarized in Equation (2.17), we will now study this difference under the consideration of density dependent population processes (DDPP). Before we state the definition of DDPPs in the next subsection, we give a briefly overview of what this section aims to achive. As in the previous Subsection 2.3.2, we use the notation

$$\Delta^N := L^N - \Lambda,$$

to denote the difference between the generators. The study of the accuracy and refinement can be broken down into two observations for this difference, namely

$$\Delta^N = O\Big(\frac{1}{N}\Big) \tag{2.25}$$ and $$\Delta^N = \frac{1}{N}\Delta + O\Big(\frac{1}{N^2}\Big). \tag{2.26}$$

The first Equation (2.25) referring to the bound on the difference of the generators and Equation (2.26) to the approximation of the generator difference by a population size independent mapping $\Delta : \mathcal{C}^2_{\mathcal{X}\to\mathbb{R}} \to \mathcal{C}^2_{\mathcal{X}\to\mathbb{R}}$ which we will call the (first order) refinement.

2.4.1 Density Dependent Population Processes

Many population processes can be model using their density such as epidemic spreading [43, 90], load balancing [87, 91] or bike sharing systems [50]. It is of little wonder that the mathematical concept of density dependent population processes (DDPPs) was developed to study these type of processes, see [76, 77]. As before, we denote by $\{\boldsymbol{X}^{(N)}(t), t \geq 0\}$ a stochastic process that represents

the state of a population of size N. It is, as the name suggests, typically the case that $\boldsymbol{X}^{(N)}(t)$ is some kind of density representation of the population. As for the generic population process, we restrict the set of states $\mathcal{X}^N$ of $\boldsymbol{X}^{(N)}$ to be a finite subset of $\mathbb{R}^d$. Furthermore, it is assumed, there exist a finite set of transitions vectors $\mathcal{L}$ and corresponding rate functions $r_\ell : \mathcal{X} \to \mathbb{R}_{\geq 0}, \ell \in \mathcal{L}$ which are both independent of the population size. Based on the jump vectors and rates, the transitions of the process are of the type

$$\boldsymbol{X}^{(N)} \qquad \text{jumps to} \qquad \boldsymbol{X}^{(N)} + \ell/N \qquad \text{at rate} \qquad N r_\ell(\boldsymbol{X}^{(N)}).$$

Intuitively, the transition structure captures that the change of a single member of the population is of order $O(1/N)$ while the actual chance of such an event is proportional to the size of the population. To catch up on the previously discussed ODE and generator definition we briefly show the typical definition of mean field approximation and the generator for this type of process. The drift corresponding to a DDPP in state $\boldsymbol{x}$ is defined as

$$\boldsymbol{f}(\boldsymbol{x}) := \sum_{\ell \in \mathcal{L}} \ell r_\ell(\boldsymbol{x}), \tag{2.27}$$

the sum over the jump vector times their rates. Under the assumption that the drift is well defined for all states in $\mathcal{X}$, we define the mean field approximation with initial condition $\boldsymbol{y} \in \mathcal{X}$ by

$$\frac{d}{dt}\boldsymbol{\phi}(t, \boldsymbol{y}) = \boldsymbol{f}(\boldsymbol{\phi}(t, \boldsymbol{y})) \qquad \text{with} \qquad \boldsymbol{\phi}(0, \boldsymbol{y}) = \boldsymbol{y}.$$

Lastly, we see that by definition, the generator is given by

$$L^N h(\boldsymbol{x}) = \sum_{\ell \in \mathcal{L}} \big(h(\boldsymbol{x} + \ell/N) - h(\boldsymbol{x})\big) N r_\ell(\boldsymbol{x}). \tag{2.28}$$

Note, by setting $h = Id$ the identity function, one obtains the drift directly from the generator definition.

2.4.2 A Simple Load Balancing Example

This subsection introduces a simple load balancing example which will be used to illustrate the subsequent methodology in Section 2.3. The system consists of N statistically identical servers which can receive and process jobs. For simplicity, we restrict the servers to have a buffer capacity of one. The dispatcher scheduling policy rules are:

- Jobs arrive at the dispatcher following a Poisson arrival stream with rate $N\lambda > 0$.
- For each arriving job, the dispatcher randomly selects two of the N servers and acts according to one of the three cases below:
  1. Both servers are idle - the job is randomly assigned to one of the servers
  2. One server is idle - the job is forwarded to the IDLE server

3. Both servers are busy - the job is discarded

- Each assigned job in the queue takes an exponentiall $\mu > 0$ distributed processing time at the server site after which it leaves the system.

We model the system with $X^{(N)}(t) \in [0,1]$ being the fraction of active servers at time $t \geq 0$. Let the system $X^{(N)}(t)$ be in state $X \in [0,1]$, then the transitions of the system are:

$$X \to X + \frac{1}{N} \qquad \text{at rate} \qquad N\lambda(1-X^2), \tag{2.29}$$

$$X \to X - \frac{1}{N} \qquad \text{at rate} \qquad N\mu X. \tag{2.30}$$

In Section 2.4.1 we will see that this load balancing example falls into a broader class of population processes, namely density dependent population processes.

Generator and ODE for the Load Balancing Example

Using the transitions of the described system, for a state $x \in \mathcal{X}^N$ the generator is given by

$$L^N h(x) = \underbrace{\big(h(x+1/N) - h(x)\big) N\lambda(1-x^2)}_{\text{Arrival}} + \underbrace{\big(h(x-1/N) - h(x)\big) N\mu x}_{\text{Departure}}.$$

The ODE associated to the process initialized at state $y \in \mathcal{X}$ is

$$\frac{d}{dt}\boldsymbol{\phi}(t,y) = \lambda(1-\boldsymbol{\phi}(t,y)^2) - \mu\boldsymbol{\phi}(t,y) \qquad \text{with} \qquad \boldsymbol{\phi}(0,y) = y.$$

2.4.3 Accuracy Results

Transient Regime

To obtain accuracy results, we pick up on Equation (2.17). Using the DDPP specific generators

$$\begin{aligned}\mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t,\boldsymbol{x})) &= \int_0^t \Psi_\tau^N \Delta^N \Phi_{t-\tau} h(\boldsymbol{x}) d\tau \\ = \int_0^t \mathbb{E}[\sum_{\ell \in \mathcal{L}^N} &\Big(h\big(\boldsymbol{\phi}(t-\tau, \boldsymbol{X}^{(N)}(\tau) + \ell/N)\big) - h\big(\boldsymbol{\phi}(t-\tau, \boldsymbol{X}^{(N)}(\tau))\big)\Big) r_\ell(\boldsymbol{X}^{(N)}(\tau)) \\ &- D_{\boldsymbol{x}}\big(h \circ \boldsymbol{\phi}(t-\tau, .)\big)(\boldsymbol{X}^{(N)}(\tau)) \boldsymbol{f}(\boldsymbol{X}^{(N)}(\tau)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] d\tau\end{aligned} \tag{2.31}$$

To obtain bounds on the distance between the generators, we have a closer look at the stochastic generator terms. To simplify notations let us define $g(\boldsymbol{x}) := h\big(\boldsymbol{\phi}(t-\tau, \boldsymbol{x})\big)$ such that the right-hand term of Equation (2.31) reduces to $\sum_{\ell \in \mathcal{L}^N} \Big(g(\boldsymbol{X}^{(N)}(\tau) + \ell/N) - g(\boldsymbol{X}^{(N)}(\tau))\Big) N r_\ell(\boldsymbol{X}^{(N)}(\tau))$.

By application of the Taylor expansion of first order on $g(\boldsymbol{X}^{(N)}(\tau)+\ell)$ around $\boldsymbol{X}^{(N)}(\tau)$ in direction ℓ/N, the expression is equal to

$$\sum_{\ell\in\mathcal{L}^N} D_x g(\boldsymbol{X}^{(N)}(\tau))\ell\, r_\ell(\boldsymbol{X}^{(N)}(\tau)) + \frac{1}{N}R_{g,1}(\boldsymbol{X}^{(N)}(\tau))$$
$$= D_x g(\boldsymbol{X}^{(N)}(\tau))\boldsymbol{f}(\boldsymbol{X}^{(N)}(\tau)) + \frac{1}{N}R_{g,1}(\boldsymbol{X}^{(N)}(\tau))$$

where by $R_{g,1}(\boldsymbol{x})$ we denote the collective remainder term arising from the expansion. Using this identity for Equation (2.31) yields

$$\int_0^t \Psi_\tau^N\big(L^N-\Lambda\big)\Phi_{t-\tau}h(\boldsymbol{x})d\tau = \frac{1}{N}\int_0^t R_{h\circ\boldsymbol{\phi}(t-\tau,.),1}(\boldsymbol{X}^{(N)}(\tau))d\tau.$$

Under the assumption that the remainder term is bounded by some $C_{R_{h\circ\boldsymbol{\phi}(t-\tau,.),1}} > 0$, it follows immediately that the accuracy of the approximation is of order $O(t/N)$.

Steady-State Accuracy

For the steady-state we continue in a similar fashion. Our starting point is the previously obtained Equation (2.24) which shows

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty)] - h(\boldsymbol{\phi}_\infty) = \mathbb{E}[(\Lambda - L^N)G_h(\boldsymbol{X}^{(N)}(\infty))]$$

with G_h being the solution to the Poisson Equation (2.21). By definition of the generators for DDPPs, we see that the right-hand side is equal to

$$\mathbb{E}[\sum_{\ell\in\mathcal{L}^N}\Big(G_h(\boldsymbol{X}^{(N)}(\infty)+\ell/N) - G_h(\boldsymbol{X}^{(N)}(\infty))\Big)Nr_\ell(\boldsymbol{X}^{(N)}(\tau)) \tag{2.32}$$
$$- D_x G_h(\boldsymbol{X}^{(N)}(\infty))\boldsymbol{f}(\boldsymbol{X}^{(N)}(\infty))]. \tag{2.33}$$

Assuming that G_h is sufficiently differentiable, which is commented on in Lemma 3, we utilize the Taylor expansion around $\boldsymbol{X}^{(N)}(\infty)$ in direction ℓ as before in the transient case. Application of Taylor then leaves us with $\mathbb{E}[\frac{1}{N}R_{G_h,1}(\boldsymbol{X}^{(N)}(\infty))]$. Here, $R_{G_h,1}(\boldsymbol{X}^{(N)}(\infty))$ arises as the sum over all Taylor remainder terms obtained from the expansion. Bounding $\mathbb{E}[R_{G_h,1}(\boldsymbol{X}^{(N)}(\infty))]$ then shows that the approximation has an accuracy of order $1/N$.

2.4.4 Refinement Terms

In this section we pick up on the bias analysis and the definition of refinement terms for the mean field approximation. The refinement concept and justification for classical mean field models was derived by Gast et al. in a recent line of work [54, 55, 57]. The following parts aim to gather the observations from the mentioned publications. While in this section we are concerned about DDPP's, similar methodological steps can be apply in other settings such as the heterogeneous case described

in [5] or two timescale setting as seen in [4]. In what follows, we restrict ourselves to the analysis of 'first order' refinement terms, i.e. computable expression which increase the accuracy of the mean field approximation by one order. In particular, this includes the derivation of the refinements terms $\boldsymbol{V} : [0, \infty] \times \mathcal{X} \to \mathbb{R}^d$ and $\boldsymbol{W} : [0, \infty] \times \mathcal{X} \to \mathbb{R}^{d \times d}$ with $t = \infty$ being the value of $\boldsymbol{V}, \boldsymbol{W}$ at the equilibrium point if existent. Under the assumption that h is a twice differentiable function and the additional assumptions outlined in Section 2.4.4, we will see that the bias between the stochastic system and the mean field approximation can be written as

$$\begin{aligned}&\mathbb{E}[h(\boldsymbol{X}^{(N)}(t)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t, \boldsymbol{x})) \\ &= \frac{1}{N}\big(D_x h(\boldsymbol{\phi}(t, \boldsymbol{x})) \cdot \boldsymbol{V}(t) + D_x^2 h(\boldsymbol{\phi}(t, \boldsymbol{x})) : \boldsymbol{W}(t)\big) + O(1/N^2)\end{aligned}$$

for $t \in [0, \infty]$ with $t = \infty$ denoting the steady-state and equilibrium respectively. Here, $\cdot$ and $:$ denote the sum over the element wise multiplication between two vectors or matrices. In the following, we will show how to obtain the above refinement terms $\boldsymbol{V}, \boldsymbol{W}$ and deduce computable expressions. Before we start, we recall and summarize the assumptions needed:

General Assumptions:

1. The set of jump vectors $\mathcal{L}$ of the stochastic process is finite and jump vectors ℓ as well as rates r_ℓ are bounded.
2. The drift $\boldsymbol{f}$ is twice continuously differentiable.

Additional for Steady-State:

1. The stochastic system admits a unique steady-state distribution.
2. The ODE $\boldsymbol{\phi}(t, \boldsymbol{x})$ has a unique attractor $\boldsymbol{\phi}_\infty$ which is exponentially stable, as defined in Definition 2.

2.4.5 Transient Refinement Terms

To obtain the refinement terms in the transient regime, we leverage on the previously discussed generator comparison method. Recall, in Equation (2.17) we have shown the equality

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(t) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}(t, \boldsymbol{x})) = \int_0^t \Psi_\tau \Delta^N \Phi_{t-\tau} h(\boldsymbol{x}) d\tau, \tag{2.34}$$

i.e., expressing the difference between the stochastic and deterministic system using their generators and semi-groups where $\Delta^N := L^N - \Lambda$. As we are interested in the study of the bias, which vanishes as the population size N grows, we re-scale the system by a factor N. Now, looking at $N \int_0^t \Psi_\tau \Delta^N \Phi_{t-\tau} h(\boldsymbol{x}) d\tau$, instead of using the first order Taylor expansion and bounding the remainder term, we use the a second order expansion. To start, we have a closer look at $N\Delta^N g(\boldsymbol{x})$ under the assumption that $g : \mathcal{X} \to \mathbb{R}$ is twice continuously differentiable. Under consideration

of the transitions structure for density dependent processes (Section 2.4.1) and by definition of the generators,

$$\Delta^N g(\boldsymbol{x}) = \sum_{\ell \in \mathcal{L}} \Big(g(\boldsymbol{x} + \frac{\ell}{N}) - g(\boldsymbol{x}) \Big) N r_\ell(\boldsymbol{x}) - D_{\boldsymbol{x}} g(\boldsymbol{x}) \cdot \boldsymbol{f}(\boldsymbol{x})$$

where we use the second order Taylor representation of $g(\boldsymbol{x} + \frac{\ell}{N})$ in direction ℓ with respective remainder term. This yields the equality to

$$\frac{1}{2} D_x^2 g(\boldsymbol{x}) : \boldsymbol{Q}(\boldsymbol{x}) + \frac{1}{N} R_{g,2}(\boldsymbol{x}), \qquad \text{with} \qquad \boldsymbol{Q}(\boldsymbol{x}) := \sum_{\ell \in \mathcal{L}} \ell \ell^T r_\ell(\boldsymbol{x}). \tag{2.35}$$

where $\boldsymbol{A} : \boldsymbol{B}$ is the Frobenius product ($\sum_{k,l} A_{k,s} B_{k,l}$) and $R_{g,2}(\boldsymbol{x})$ is the collective remainder term arising from the expansion. Define

$$\Delta : \mathcal{C}^2_{\mathcal{X} \to \mathbb{R}} \to \mathcal{C}^2_{\mathcal{X} \to \mathbb{R}}; \qquad \Delta g(\boldsymbol{x}) := D_x^2 g(\boldsymbol{x}) : \boldsymbol{Q}(\boldsymbol{x}). \tag{2.36}$$

With this definition, we see that for a sufficiently regular g

$$\Delta^N g(\boldsymbol{x}) = \Delta g(\boldsymbol{x}) + O\Big(\frac{1}{N}\Big)$$

where the $O(1/N)$ hides the bound on the remainder term $R_{g,2}$. To define the refinement terms $\boldsymbol{V}$ and $\boldsymbol{W}$, we start with replacing g by $h \circ \boldsymbol{\phi}(t, .)$ in the above equations. We further know that $\boldsymbol{X}^{(N)}(\tau)$ converges weakly to $\boldsymbol{\phi}(\tau, \boldsymbol{x})$ for every $\tau \in [0, t]$ as N goes to infinity in the case of DDPPs, see for example [76]. Combing these two steps, it is sensible to look at the term

$$\begin{aligned} \Psi_\tau^N \Delta \Phi_{t-\tau} h(x) &= \mathbb{E}[\frac{1}{2} D_x^2 (h \circ \boldsymbol{\phi}(t-\tau, .))(\boldsymbol{X}^{(N)}(\tau)) : \boldsymbol{Q}(\boldsymbol{X}^{(N)}(\tau)) \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] \\ \xrightarrow{N \to \infty} \Phi_\tau \Delta \Phi_{t-\tau} h(\boldsymbol{x}) &= \frac{1}{2} D_x^2 (h \circ \boldsymbol{\phi}(t-\tau, .))(\boldsymbol{\phi}(\tau, \boldsymbol{x})) : \boldsymbol{Q}(\boldsymbol{\phi}(\tau, \boldsymbol{x})) \end{aligned}$$

under the integral to obtain the deterministic refinement terms. Based on this, in Section 2.4.7 we give insight in how to obtain a set of differential equations to compute the quantity $\int_0^t \Phi_\tau \Delta \Phi_{t-\tau} h(\boldsymbol{x})$, the bias correction for $h(\boldsymbol{\phi}(t, \boldsymbol{x}))$.

2.4.6 Extension to the steady state

In this Section, we aim to obtain similar refinement terms as for the transient regime. We start by recalling the additional assumptions for the steady-state, given in 2.4.4. Next, by Equation (2.20) of Section 2.3.2 we know

$$\begin{aligned} \mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty))] - h(\boldsymbol{\phi}_\infty) &= \Psi_\infty \Delta^N \int_0^\infty \Phi_\tau h(\boldsymbol{x}) d\tau \\ &= \mathbb{E}[(L^N - \Lambda) \int_0^\infty h(\boldsymbol{\phi}(\tau, \boldsymbol{X}^{(N)}(\infty))) d\tau \mid \boldsymbol{X}^{(N)}(0) = \boldsymbol{x}], \end{aligned}$$

where we assumed that h is 'nice' enough and, w.l.o.g, that $h(\boldsymbol{\phi}_\infty) = 0$[2] such that the integral term is well defined. In fact, we will now informally show that the above term, can also be obtained by leveraging a different line of arguments based on defining a Poisson equation. To obtain computable expressions for the steady-state refinements, we extend on the analysis of the Poisson equation. First however, recall that our previous analysis Equation (2.24) showed the equality

$$\mathbb{E}[h(\boldsymbol{X}^{(N)}(\infty)) \,|\boldsymbol{X}^{(N)}(0) = \boldsymbol{x}] - h(\boldsymbol{\phi}_\infty) = \mathbb{E}[\big(L^N - \Lambda\big)G_h(\boldsymbol{X}^{(N)}(\infty))]$$

where $G_h(\boldsymbol{x})$ is given by $-\int_0^\infty h\big(\boldsymbol{\phi}(\tau, \boldsymbol{x})\big) - h\big(\boldsymbol{\phi}_\infty\big)d\tau$. To study the bias and obtain steady-state refinement terms we proceed similar as in the transient case by applying a scaling factor N to the term. Using the generator definitions, this gives

$$\begin{aligned}
&\mathbb{E}[N(\Lambda - L^N)G_h(\boldsymbol{X}^{(N)}(\infty))] \\
&\quad = \mathbb{E}[N \sum_{\ell \in \mathcal{L}^N} \left(G_h(\boldsymbol{X}^{(N)}(\infty) + \frac{\ell}{N}) - G_h(\boldsymbol{X}^{(N)}(\infty)) \right) N r_\ell(\boldsymbol{X}^{(N)}(\tau)) \\
&\qquad - N D_x G_h(\boldsymbol{X}^{(N)}(\infty)) \boldsymbol{f}(\boldsymbol{X}^{(N)}(\infty))].
\end{aligned}$$

Assuming that the drift $\boldsymbol{f}$ and ODE are sufficiently differentiable, Lemma 3 tell us that the same is true for the solution to the Poisson equation G_h. Thus, we can use a second order Taylor expression of $G_h(\boldsymbol{X}^{(N)}(\infty) + \ell)$ around $\boldsymbol{X}^{(N)}(\infty)$ which shows that the above equation is equal to

$$\mathbb{E}[D_x^2 G_h(\boldsymbol{X}^{(N)}(\infty)) : \boldsymbol{Q}(\boldsymbol{X}^{(N)}(\infty)) + \frac{1}{N} R_{G_h,2}(\boldsymbol{X}^{(N)}(\infty)).$$

Using the weak convergence of the stochastic systems, we see that, by reusing the definition of Δ (Equation (2.36)),

$$\Delta G_h(\boldsymbol{\phi}_\infty) = D_x^2 G_h(\boldsymbol{\phi}_\infty) : \boldsymbol{Q}(\boldsymbol{\phi}_\infty). \tag{2.37}$$

we obtained a similar refinement candidate as for the transient regime. This time depending on the solution to the Poisson equation G_h with respect to the function h evaluated at the equilibrium point. As in this form, $D_x^2 G_h(\boldsymbol{\phi}_\infty)$ can not be solved numerically, we discuss in Section 2.4.8 how to obtain a computable expressions of the refinement.

2.4.7 Computing the Transient Refinement Terms

In the next part, we aim to obtain a computable expression for the bias correction $\int_0^t \Phi_\tau \Delta \Phi_{t-\tau} h(\boldsymbol{x})$ of $h(\boldsymbol{\phi}(t, \boldsymbol{x}))$ with mapping $\Delta g(\boldsymbol{x}) = D_x^2 g(\boldsymbol{x}) : \boldsymbol{Q}(\boldsymbol{x})$ and $\boldsymbol{Q}(\boldsymbol{x})$ as defined in Equation (2.35). We start by looking more closely at the term $\Delta \Phi_{t-\tau} h(\boldsymbol{x}) = D_x^2(h \circ (\boldsymbol{\phi}(t-\tau, .))(\boldsymbol{x}) : \boldsymbol{Q}(\boldsymbol{x})$. To properly work with the second derivative of $h \circ \boldsymbol{\phi}(t-\tau, .)$, we take a step back and notice that for

[2] As mentioned in Section 2.3.2 if for $h(\boldsymbol{\phi}_\infty) \neq 0$, we can define $h^*(\boldsymbol{x}) := h(\boldsymbol{x}) - h(\boldsymbol{\phi}_\infty)$ which then fulfills the requirement.

twice differentiable functions $g : \mathbb{R}^d \to \mathbb{R}$ and $\boldsymbol{m} : \mathbb{R}^d \to \mathbb{R}^d$, by application of the chain and product rule,

$$D_x^2(g \circ \boldsymbol{m})_{k,l}(\boldsymbol{x}) = \sum_{i,j=1}^{d} \frac{\partial^2 g}{\partial x_i \partial x_j}(\boldsymbol{m}(\boldsymbol{x}))\frac{\partial m_i}{\partial x_k}(x)\frac{\partial m_j}{\partial x_l}(\boldsymbol{x}) + \sum_{j=1}^{d} \frac{\partial g}{\partial x_j}(\boldsymbol{m}(\boldsymbol{x}))\frac{\partial^2 m_j}{\partial x_k \partial x_l}(\boldsymbol{x})$$

Replacing the functions g and m by h and $\boldsymbol{\phi}(t-\tau,.)$ gives us for $D_x^2(h \circ \boldsymbol{\phi}(t-\tau,.))(\boldsymbol{x})$

$$\begin{aligned}\big(D_x^2 h \circ \boldsymbol{\phi}(t-\tau,.)\big)_{k,l}(\boldsymbol{x}) = &\sum_{i,j=1}^{d} \frac{\partial^2 h}{\partial x_i \partial x_j}(\boldsymbol{\phi}(t-\tau,\boldsymbol{x}))\frac{\partial \phi_i}{\partial x_k}(t-\tau,\boldsymbol{x})\frac{\partial \phi_j}{\partial x_l}(t-\tau,\boldsymbol{x}) \\ &+ \sum_{i=1}^{d} \frac{\partial h}{\partial x_i}(\boldsymbol{\phi}(t-\tau,\boldsymbol{x}))\frac{\partial^2 \phi_i}{\partial x_k \partial x_l}(t-\tau,\boldsymbol{x}).\end{aligned}$$

Under the integral and with substitution of $\boldsymbol{\phi}(\tau,\boldsymbol{x})$ for $\boldsymbol{X}^{(N)}(\tau)$ we then obtain

$$\begin{aligned}&\int_0^t \frac{1}{2} D_x^2(h \circ \boldsymbol{\phi}(t-\tau,.))(\boldsymbol{\phi}(\tau,\boldsymbol{x})) : \boldsymbol{Q}(\boldsymbol{\phi}(\tau,\boldsymbol{x})) \\ &= \int_0^t \frac{1}{2} \sum_{k,l=1}^{d} \left(\sum_{i,j=1}^{d} \frac{\partial^2 h}{\partial x_i \partial x_j}(\boldsymbol{\phi}(t,\boldsymbol{x}))\frac{\partial \phi_i}{\partial x_k}(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{x}))\frac{\partial \phi_j}{\partial x_l}(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{x})) \right) Q_{k,l}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))d\tau \\ &\quad + \int_0^t \frac{1}{2} \sum_{k,l=1}^{d} \left(\sum_{j=1}^{d} \frac{\partial h}{\partial x_j}(\boldsymbol{\phi}(t,\boldsymbol{x}))\frac{\partial^2 \phi_j}{\partial x_k \partial x_l}(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{x})) \right) Q_{k,l}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))d\tau \\ &= \sum_{i,j=1}^{d} \frac{\partial^2 h}{\partial x_i \partial x_j}(\boldsymbol{\phi}(t,\boldsymbol{x})) \underbrace{\int_0^t \frac{1}{2} \sum_{k,l=1}^{d} \left(\frac{\partial \phi_i}{\partial x_k}(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{x}))\frac{\partial \phi_j}{\partial x_l}(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{x})) \right) Q_{k,l}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))d\tau}_{=:W_{i,j}} \\ &\quad + \sum_{j=1}^{d} \frac{\partial h}{\partial x_j}(\boldsymbol{\phi}(t,\boldsymbol{x})) \underbrace{\int_0^t \frac{1}{2} \sum_{k,l=1}^{d} \frac{\partial^2 \phi_j}{\partial x_k \partial x_l}(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{x}))Q_{k,l}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))d\tau}_{=:V_j}.\end{aligned}$$

For the last equality, we used the elementary property $\boldsymbol{\phi}\big(t-\tau,\boldsymbol{\phi}(\tau,\boldsymbol{X})\big) = \boldsymbol{\phi}(t,\boldsymbol{X})$ to take the derivatives of h out of the integral. We define the refinement terms $\boldsymbol{W}, \boldsymbol{V}$ in their integral form at time t by

$$\boldsymbol{V}_i(t) := \int_0^t \frac{1}{2} D_x^2 \boldsymbol{\phi}_i(t-\tau,.))(\boldsymbol{\phi}(\tau,\boldsymbol{x})) : \boldsymbol{Q}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))d\tau, \tag{2.38}$$

$$\boldsymbol{W}_{i,j}(t) := \int_0^t \frac{1}{2} D_x \boldsymbol{\phi}_i(t-\tau,.)(\boldsymbol{\phi}(\tau,\boldsymbol{x}))\boldsymbol{Q}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))D_x \boldsymbol{\phi}_j(t-\tau,.)^T(\boldsymbol{\phi}(\tau,\boldsymbol{x}))d\tau. \tag{2.39}$$

To see that the integral forms of $\boldsymbol{V}$ and $\boldsymbol{W}$ are indeed the solutions of the differential equations (2.40) and (2.41) we recall the statement of Lemma D.3 of [5]. Alternatively, the reader is referred to [54], which also derives the differential form of the refinement terms for density dependent population processes. It holds that the refinement terms given in their integral forms as in Equations (2.38) and (2.39) are the solutions to the differential equations:

$$\frac{d}{dt}V_i(t) = \sum_{j=1}^{d} \frac{\partial f_i}{\partial x_j}(\boldsymbol{\phi}(t,\boldsymbol{x}))V_j(t) + \sum_{j,k=1}^{d} \frac{1}{2}\frac{\partial^2 f_i}{\partial x_j \partial x_k}(\boldsymbol{\phi}(t,\boldsymbol{x}))W_{j,k}(t) \tag{2.40}$$

$$\frac{d}{dt}W_{i,j}(t) = \sum_{k=1}^{d} \frac{\partial f_i}{\partial x_k}(\boldsymbol{\phi}(t,\boldsymbol{x}))W_{k,j}(t) + \sum_{k}^{d} \frac{df_j}{dx_k}(\boldsymbol{\phi}(t,\boldsymbol{x}))W_{k,i}(t) + Q_{i,j}(\boldsymbol{\phi}(t,\boldsymbol{x})) \tag{2.41}$$

$$Q_{i,j}(\boldsymbol{\phi}(t,\boldsymbol{x})) = \sum_{\ell \in \mathcal{L}} r_\ell(\boldsymbol{\phi}(t,\boldsymbol{x}))\ell_i \ell_j \tag{2.42}$$

for $i, j = 1..d$ and with initial conditions

$$\boldsymbol{V}(0) = 0 \qquad \text{and} \qquad \boldsymbol{W}(0) = 0. \tag{2.43}$$

where f refers to the drift of the mean field approximation $\boldsymbol{\phi}$.

A detailed proof can be found in [5] - Lemma D.3 p.38. For the sake of completeness we only recall the basic ideas here. To prove the connection between the integral from and differential form, note that for a sufficiently differentiable and bounded function $g(t,s) : [0,\infty) \times [0,\infty) \to \mathbb{R}$

$$\frac{d}{dt}\int_0^t g(t,\tau)d\tau = g(t,t) + \int_0^t \frac{d}{dt}g(t,\tau)d\tau.$$

To obtain the differential forms replace $g(t,\tau)$ by $D_x^2\boldsymbol{\phi}(t-\tau,.))(\boldsymbol{\phi}(\tau,\boldsymbol{x})) : \boldsymbol{Q}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))$ for $\boldsymbol{V}$ and by $D_x\boldsymbol{\phi}(t-\tau,.))(\boldsymbol{\phi}(\tau,\boldsymbol{X}))D_x\boldsymbol{\phi}(t-\tau,.))(\boldsymbol{\phi}(\tau,\boldsymbol{x})) : \boldsymbol{Q}(\boldsymbol{\phi}(\tau,\boldsymbol{x}))$ for $\boldsymbol{W}$. Careful analysis and rearranging of term then yields the desired derivatives.

2.4.8 Computing the Steady-State refinement Terms

In this subsection we aim to show that the refinement term for the steady-state as in Equation (2.37) can be efficiently computed. Thus, in the following rather technical part, we will proof two key properties:

1. The Lyapunov equation $(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty))\boldsymbol{Z} + \boldsymbol{Z}(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty))^T + \boldsymbol{Q}(\boldsymbol{\phi}_\infty) = 0$ has a unique solution $\boldsymbol{W}$ which is given by

$$\boldsymbol{W} = \int_0^\infty \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)\tau)\boldsymbol{Q}(\boldsymbol{\phi}_\infty)\exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)^T\tau)d\tau.$$

2. The second derivative of G_h satisfies

$$D_x^2 G_h(\boldsymbol{\phi}_\infty) : \boldsymbol{Q}(\boldsymbol{\phi}_\infty) = D_x^2 h(\boldsymbol{\phi}_\infty) : \boldsymbol{W} + D_x h(\boldsymbol{\phi}_\infty) \cdot \boldsymbol{V}$$

with $V := (-D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty))^{-1}(D_x^2 \boldsymbol{f}(\boldsymbol{\phi}_\infty))\boldsymbol{W}$.

The properties are originally stated and proven in Lemma 3.6 of [59]. However, as they convert the analytical expression (2.37) into computable ones, we will successively go though the reasoning behind the properties. To show (1), we first recall some properties about Lyapunov equations. For real matrices $\boldsymbol{Q}$ and $\boldsymbol{A}$, an equation of the form

$$\boldsymbol{A}\boldsymbol{Z} + \boldsymbol{Z}\boldsymbol{A}^T + \boldsymbol{Q} = 0$$

is called continuous Lyapunov equation. The equation admits a unique solution $\boldsymbol{W}$ if and only if $\boldsymbol{A}$ is Hurwitz, i.e., $\boldsymbol{A}$ has only negative eigenvalues. Symmetry of $\boldsymbol{Q}$ furthermore implies symmetry of the solution $\boldsymbol{W}$. If existent, it then is a standard property of Lyapunov equations is that its solution can be written as

$$\boldsymbol{W} = \int_0^\infty \exp(\boldsymbol{A}\tau)\boldsymbol{Q}\exp(\boldsymbol{A}^T\tau)d\tau. \tag{2.44}$$

We see that by definition $\boldsymbol{Q}(\boldsymbol{\phi}_\infty) = \sum_{\ell \in \mathcal{L}^N} \ell\ell^T r_\ell(\boldsymbol{\phi}_\infty)$ is symmetric and that by the assumption of exponential stability for the differential equation $D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty)$ is indeed Hurwitz. This shows that, by defining $\boldsymbol{A} := D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty)$, $\boldsymbol{W}$ can be computed numerically by solving the Lyapunov equation.

To show the second property (2), we split the reasoning into two parts. At first we show that the first derivative of G_{Id} is equal to $(-D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty))^{-1}$, where G_{Id} is the solution to the Poisson equation with respect to the Identity mapping.[3] In the second step, we use this property to show that $D_x^2 G_h(\boldsymbol{\phi}_\infty) : \boldsymbol{Q}(\boldsymbol{\phi}_\infty)$ can indeed be expressed using the derivatives of h and the numerically tractable expressions $\boldsymbol{W}$ and $\boldsymbol{V}$. To start, we take a step back to first look at the solution of the differential equation

$$\frac{d}{dt} D_x \boldsymbol{\phi}(t, \boldsymbol{x}) = D_x\big(\boldsymbol{f}(\boldsymbol{\phi}(t, \boldsymbol{x}))\big) = D_x \boldsymbol{f}(\boldsymbol{\phi}(t, \boldsymbol{x})) D_x \boldsymbol{\phi}(t, \boldsymbol{x})$$

with initial condition $D_x \boldsymbol{\phi}(0, \boldsymbol{x}) = I$. Starting from steady-state $\boldsymbol{\phi}_\infty$, the above differential equation has the drift $D_x \boldsymbol{f}(\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty)) D_x \boldsymbol{\phi}(t, \boldsymbol{x})$ and thus its solution at time t is $D_x \boldsymbol{\phi}(t, \boldsymbol{x}) = \exp(D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty)t)$. Returning back to $D_x G_{Id}(\boldsymbol{\phi}_\infty)$ we see, under consideration that $\boldsymbol{\phi}_t$ is exponentially stable, that

$$D_x G_{Id}(\boldsymbol{\phi}_\infty) = \int_0^\infty D_x \boldsymbol{\phi}(\tau, \boldsymbol{x}) d\tau = \int_0^\infty \exp(D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty)\tau) d\tau = (-D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty))^{-1}.$$

[3] It might appear to the reader that in the definition of the Poisson equation we restricted ourselves to function with image in $\mathbb{R}$. The extension to the multidimensional case however is straight forward by defining the Poisson equation for each component of the function.

The second property (2) is similarly shown. We know by Lemma 3 that $D_x^2 G_h(\boldsymbol{x}) = \int_0^\infty D_x^2(h \circ \phi_\tau)(\boldsymbol{x})d\tau$, thus, we start by analyzing the time dependent term $D_x^2(h \circ \boldsymbol{\phi}(\tau, .))(\boldsymbol{x})$. By application of the chain and product rule we obtain

$$\begin{aligned}\frac{d}{dt} D_x^2 \phi(t, \boldsymbol{x})_{j,m,n} &= \left(D_x^2\big(\boldsymbol{\phi}(t, .)\big)\boldsymbol{f}(\boldsymbol{\phi}(t, .))\big)(\boldsymbol{x})\right)_{j,m,n} \\ &= \sum_k \frac{\partial f_j}{\partial x_k}(\boldsymbol{\phi}(t, \boldsymbol{x}))\frac{\partial^2 \phi_k}{\partial x_m \partial x_n}(t, \boldsymbol{x}) \\ &\quad + \sum_{k_1,k_2} \frac{\partial^2 f_j}{\partial x_{k_1}\partial x_{k_2}}(\boldsymbol{\phi}(t, \boldsymbol{x}))\frac{\partial \phi_{k_1}}{\partial x_m}(t, \boldsymbol{x})\frac{\partial \phi_{k_2}}{\partial x_n}(t, \boldsymbol{x}).\end{aligned}$$

Next, we set the initial condition to the equilibrium point $\boldsymbol{\phi}_\infty$ and keep the previously obtained exponential form for $D_x\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty) = \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)t)$ in mind. By application of the variation of constants method, it is possible to obtain the solution for the above system which is given by

$$\begin{aligned}D_x^2\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty)_{j,m,n} &= \sum_u \int_0^t \exp\big(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)(t-\tau)\big)_{j,u} \\ &\times \sum_{k_1,k_2} \frac{\partial^2 f_u}{\partial x_{k_1}\partial x_{k_2}}(\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty)) \times \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)\tau)_{k_1,m} \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)^T\tau)_{k_2,n}d\tau.\end{aligned}$$

Taking the integral and switching the order of integration, one obtains that $D_x^2 G(\boldsymbol{\phi}_\infty)$ is equal to

$$\begin{aligned}\int_0^\infty D_x^2\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty)_{j,m,n}dt &= \sum_u \big(-D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)\big)^{-1}_{j,u} \\ &\times \sum_{k_1,k_2} \frac{\partial^2 f_u}{\partial x_{k_1}\partial x_{k_2}}(\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty)) \int_0^\infty \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)\tau)_{k_1,m} \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)^T\tau)_{k_2,n}d\tau. \quad (2.45)\end{aligned}$$

Under the consideration of the results of (1), taking the Frobenius product between equation (2.45) and $\boldsymbol{Q}$ yields,

$$\sum_u \big(-D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)\big)^{-1}_{j,u} \sum_{k_1,k_2} \frac{\partial^2 f_u}{\partial x_{k_1}\partial x_{k_2}}(\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty))\boldsymbol{W}_{k_1,k_2}.$$

Lastly, it remains to apply the above deductions to $D_x^2 G_h(\boldsymbol{\phi}_\infty) : \boldsymbol{Q}(\boldsymbol{\phi}_\infty)$, i.e., the matrix $\boldsymbol{Q}(\boldsymbol{\phi}_\infty)$ multiplied by the second derivative of the solution to Poisson equation w.r.t. h. Based on Lemma 3, we see that

$$\begin{aligned}D_x^2 G_h(\boldsymbol{\phi}_\infty)_{m,n} &= \sum_i \frac{\partial h}{\partial x_i}(\boldsymbol{\phi}_\infty) \int_0^\infty D_x^2\boldsymbol{\phi}(t, \boldsymbol{\phi}_\infty)_{i,m,n}dt \\ &\quad + \sum_{i,j} \frac{\partial^2 h}{\partial x_i \partial x_j}(\boldsymbol{\phi}_\infty) \int_0^\infty D_x\boldsymbol{\phi}(\tau, \boldsymbol{x})_{i,m} D_x\boldsymbol{\phi}(\tau, \boldsymbol{x})_{j,n}dt.\end{aligned}$$

Second to last, define the vector $\boldsymbol{V}$ by

$$V_j = \frac{1}{2}\sum_u \left(-D_x \boldsymbol{f}(\boldsymbol{\phi}_\infty)\right)^{-1}_{j,u} \sum_{k_1,k_2} \frac{\partial^2 f_u}{\partial x_{k_1}\partial x_{k_2}}(\boldsymbol{\phi}(t,\boldsymbol{\phi}_\infty))W_{k_1,k_2}. \tag{2.46}$$

To see, that the initial statement indeed holds true, substitute the previously obtained, computable expressions for $\int_0^\infty D_x^2\boldsymbol{\phi}(t,\boldsymbol{\phi}_\infty)dt$ and that $D_x\boldsymbol{\phi}(t,\boldsymbol{x}) = \exp(D_x\boldsymbol{f}(\boldsymbol{\phi}_\infty)t)$ into the equation $D_x^2 G_h(\boldsymbol{\phi}_\infty) : \boldsymbol{Q}(\boldsymbol{\phi}_\infty)$. By application of the identity (2.44) and the definition of $\boldsymbol{V}$ as in Equation (2.46) property 2 is obtained.

Lemma 3 (Differentiability of the Solution to the Poisson Equation [59])**.** *Let $\boldsymbol{\phi}(t,\boldsymbol{x})$ be the solution of the Cauchy problem $\frac{d}{dt}\boldsymbol{\phi}(t,\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{\phi}(t,\boldsymbol{x}))$; $\boldsymbol{\phi}(0,\boldsymbol{x}) = \boldsymbol{x}$ with drift $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^{\dim\mathcal{X}}$. Assume that both $\boldsymbol{\phi}$ and $\boldsymbol{f}$ are k-times uniformly continuous differentiable and that $\boldsymbol{\phi}$ has a unique exponentially stable attractor. Further let $h : \mathcal{X} \to \mathbb{R}$ be a k-times differentiable function with uniform continuous k-th derivative. Then G_h is also k-times continuously differentiable with bounded derivatives equal to $\int_0^\infty D_x^k(h\circ\boldsymbol{\phi}_\tau)(\boldsymbol{x})d\tau$.*

Proof. The proof is as in [59]. □

2.4.9 Higher Order Corrections

As shown in [54, 59], it is possible to extend refinement idea, given sufficient regularity of the dynamical system ϕ and h. Loosely speaking, for a refinement of order m, the refinement expansion consists of a set of operators $\Delta_i, i = 1, 2$ such that

$$\Delta^N = \frac{1}{N}\Delta_1 + \frac{1}{N^2}\Delta_2 + o(\frac{1}{N^2}).$$

In theory and under suitable differentiability assumptions, the method can be extended to obtain even higher order expansion terms. This comes at the cost of extremely intricate expressions and implementations which result in minor accuracy gains in applications. We also point out that in application the first order correction usually strikes a good balance between the numerical complexity and accuracy gain, example Section 3.5.1 - Table 3.2.

Part II

Heterogeneous Mean Field Models

3 Mean Field and Refined Mean Field Approximations for Heterogeneous Systems

In this chapter, we define an interaction model that allows obtaining asymptotic convergence results for stochastic systems with heterogeneous object behavior, and show that the error of the mean field approximation is of order $O(1/N)$. We further show how to adapt the refined mean field approximation, developed by the authors of [54], and prove that the error of this refined approximation is reduced to $O(1/N^2)$.

This chapter is based on our publication
S. Allmeier and N. Gast. "Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works!" *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 6:1, 25, 2022, 13:1–13:43. DOI: 10.1145/3508033.

Contents

3.1 Introduction

The mean field approximation method is a widely used tool to analyze large-scale and complex stochastic models composed of interacting objects. The idea of the approximation is to assume that objects within the system evolve independently. Following this assumption, interactions of objects in the system are approximated by a "mean" behavior, which allows to model the system's evolution as a set of deterministic ordinary differential equations. Mean field approximation finds widespread use in fields such as epidemic spreading [43, 90], load balancing strategies [87, 91], caching [60] or SSDs [108]. Building on this approximation, a refined mean field approximation is introduced in [54, 59] that greatly improves the accuracy of mean field approximations for populations of $N = 10$ to $N = 100$ objects. The popularity of mean field approximation lies in the ease of defining and solving the differential equations as well as the increasingly high accuracy for large systems.

Most of the theoretical work, however, has been done for systems where the interacting objects have homogeneous transitions, as for density-dependent population processes of Kurtz [77], or can be clustered into a finite number of groups with similar statistical behavior. Yet, in many models, heterogeneity plays an important role. This is particularly relevant to model caches, where object popularities vary broadly among contents, or epidemic spreading, where variations of sensibility among agents can greatly influence the long-term dynamics and vaccination strategies [61]. Using a finite number of clusters with homogeneous behavior simplifies the underlying models and essentially ignores the actual heterogeneity. Up to now, there are virtually no fully heterogeneous models with theoretical guarantees on why mean field approximation is a valid technique.

In this chapter, we generalize the notion of mean field approximation and refined mean field approximation to stochastic systems composed of N heterogeneous objects and show that similar asymptotic results as for the homogeneous case hold. For such a system, we show that it is possible to construct a set of ordinary differential equations (ODEs) which approximate $\mathbb{P}(S_k(t) = s)$, the probability for an object k to be in a state s at time t. This can be used to approximate the expectation of a function of the state of an object (such as the average queue length in a queuing system).

To give some intuition, the way we construct our approximations is to consider a scaled model with C identical copies of each object. This allows one to define the mean field approximation $x_{(k,s)}(t)$ and a $1/C$-expansion term $v_{(k,s)}(t)$ defined in [54]. These approximations are shown in [54] to be asymptotically accurate as the number of copies C goes to infinity. As the fully heterogeneous system corresponds to having one copy, the heuristic reasoning is then to apply the approximation with $C = 1$. Up to now, there was no theoretical foundation on why this should work because all results assume that the number of copies C goes to infinity.

We provide the first rigorous justification of the validity of this approach. The main contribution of this chapter is to show that if $x_{(k,s)}(t)$ is the mean field approximation and $v_{(k,s)}(t)$ is the expansion term defined in [54], then

$$\mathbb{P}(S_k(t) = s) = x_{(k,s)}(t) + O(1/N),$$
$$\mathbb{P}(S_k(t) = s) = x_{(k,s)}(t) + v_{(k,s)}(t) + O(1/N^2).$$

As an important corollary, if the system is composed of C copies of N heterogeneous objects, we then have $\mathbb{P}(S_k(t) = s) = x_{(k,s)}(t)+(1/C)v_{(k,s)}(t)+O(1/(CN)^2)$. This shows that the accuracy of the mean field and refined mean field approximation does not depend on the level of heterogeneity of the system but only on the total number of objects (being N or NC).

To do so, we develop a heterogeneous interaction model in which each object changes state either unilaterally or by interacting with $d-1$ other objects. The main assumption that we make in our model is that the rate at which a given tuple of d individuals interact scales as $O(1/N^{d-1})$. As there are $O(N^d)$ such tuples, this guarantees a uniform bound on the interactions between tuples. This model covers an extensive range of models with pairwise interactions, such as infection models, load balancing strategies, or cache replacement policies. These approximations can be computed by solving a differential equation that can be easily integrated numerically. For the mean field approximation, the number of variables grows linearly with the number of different objects N. For the refined approximation, it grows quadratically with N. Our proposed framework naturally extends mean field models for homogeneous population processes and the results are comparable with [55, 77, 113]. Our approach does not assume any homogeneity in the system and does not cluster objects into a finite number of classes. Hence, it can be applied to interacting systems where all objects are different.

To illustrate our results, we provide two examples that show how the mean field and refined mean field approximation can be applied. They also show that the hidden constants in the $O(1/N)$ and $O(1/N^2)$ error terms given by the theorems are small in practice. Our first example is a list-based cache replacement algorithm studied in [60] consisting of N objects whose popularities follow a Zipf-like distribution. We study how the cache popularities depend on the replacement policy for the transient and steady-state regime. For transient results, we compare the mean field and refined mean field approximation with simulations, which indicates that the refined mean field provides a significant improvement of accuracy. The results are even more striking for the steady-state regime for which it is possible to compute the exact steady-state distribution if the system size is small. This allows us to compare the accuracy of the two approximations and the simulation to the exact value. We observe that, for any reasonable computational power, the confidence intervals provided by the simulation are higher than the error of the refined mean field approximation as soon as N exceeds a few tens. In a second example, we apply the approximation techniques to a heterogeneous two-choice load balancing model. The heterogeneity in the model is introduced by considering varying server rates. As for the previous examples, we give a full description justifying the use of the mean field models and show by numerical computation that the obtained results confirm the theoretical statements. Numerical calculations for values such as the average queue length and queue length tail distribution are given. We also compare the approximation results to a homogeneous variant of the system where the server rates are set to the average server rate of the heterogeneous model. This shows that, as expected, taking heterogeneity into account strikingly improves the accuracy of the

results. For both examples, we adapt numerical methods from the toolbox developed by [6], which allowed us to implement and solve the differential equations with relative ease.

3.2 Related Work

Generator and Stein's method This chapter builds on the recent line of work regarding the use of Stein's method [102]. This method allows one to estimate precisely the distance between two random variables by looking at the distance between the generators of two related stochastic systems. This method has seen a resurgence of interest in the stochastic network community since the work of [31, 33]. There is still an active development in this area. For instance, this method has been used to develop higher-order diffusion approximations [32, 33]. It is used in [66] to develop a normal approximation of a heterogeneous discrete time population process. One of the key differences between our work and theirs is that the two aforementioned papers consider one-dimensional processes (i.e., the state of each object of the system is either 0 or 1), and the extension to more complex dynamics is not direct, at least from a computational point of view. One contribution of our work is to demonstrate how to deal with multiple states, by changing the state representation.

Refined mean field methods Stein's method has been successfully used to study the accuracy of mean field methods in [55, 74, 114]. These works show that the accuracy of the mean field approximation is $O(1/N)$ for a system with N *homogeneous* objects. By using an expansion of the generator, these results have been extended in [54, 59] to propose what the authors called a *refined* mean field approximation, that is similar to the system size expansion introduced in mathematical biology [62, 63, 109].

In fact, there exists a close link between the refined approximation that we propose in the chapter and the approach of [54, 59]. Since the results of [54, 59] only apply to systems composed of homogeneous objects, let us consider a hypothetical model composed of C identical copies of each of the N object, and let us denote by $X_{(k,s)}^{[C \text{ copies}]}(t)$ the number of copies of the object k that are in state s at time t. By [54, 59], there exists a constant $v_{(k,s)}(t)$ such that

$$\mathbb{E}[X_{(k,s)}^{[C \text{ copies}]}(t)] = x_{(k,s)}(t) + \frac{1}{C} v_{(k,s)}(t) + O(1/C^2). \tag{3.1}$$

Since our original model corresponds to $C = 1$, the rationale behind our approximation is to use $x_{(k,s)}(t)$ as a first order approximation and $x_{(k,s)}(t) + v_{(k,s)}(t)$ as the refined approximation. Yet, this is no a priori guarantee of why $O(1/C^2)$ should be small for C equal to one. As a key technical contribution, this chapter gives the theoretical foundation of this method.

This requires to overcome several difficulties, that are our main contributions compared to the aforementioned papers:

1. We define an interaction model that can be dealt with by bounding the interaction rates.

2. A key idea of our work is to use the indicators $X_{k,s}(t) \in \{0, 1\}$ and not the proportion of objects in a given state. This allows us to construct the expansion terms and do the proofs.

3. Similar to previous works [55, 74, 113], we use generators to reduce the analysis of the mean field error to the study of the sensibility of an ODE with respect to its initial conditions (Section 3.6.2). The extra difficulty in our case is to carefully analyze the remainder terms: while to analyze (3.1), the error is a finite sum of $O(1/C)$ terms, here we have a sum of N (or N^2) terms, some of them being of order $O(1/N^2)$ and others being of order $O(1/N^3)$. Dealing with all these different cases requires quite some care and is the subject of Appendix 3.8.4.

Note that to study heterogeneous population models, it is quite common to assume that there are N classes with C copies of the same objects of class $k \in \{1 \dots N\}$. This has been used for instance to study load-balancing strategies [91] or cache replacement policies [65, 105]. Our approach generalizes such methods as we assume that the objects can be fully heterogeneous. This is for instance what is used in replica models [10, 86].

Heterogeneous populations and caches In the performance evaluation community, heterogeneous population models are very common when studying cache replacement policies, where the popularity of objects is typically assumed to follow a Zipf-like distribution. As the dynamics of caches are intrinsically complicated, many mean field like approximations have been proposed, such as the famous TTL-approximation of [48] (sometimes misleadingly called the Che-approximation after it was rediscovered in [40]) or fixed-point approximation like [42]. Theoretical support exists to prove that these approximations are asymptotically correct [51, 60, 71].

The generic method that we propose in this chapter has two advantages: First, we prove that the accuracy of the mean field method is $O(1/N)$ whereas the above papers only obtain bounds in $O(1/\sqrt{N})$. Second, we develop a refined approximation that can greatly improve the accuracy compared to the cited method, at the price of being computationally more expensive.

In particular, our result applies directly to the cache replacement model of [60] in which a mean field approximation for the RAND($\boldsymbol{m}$) policy is derived. This chapter also contains a theorem that shows that the mean field approximation is $O(1/\sqrt{N})$-accurate. Yet, we do not think that the proof of the main result of [60] is correct because of the use of a martingale inequality combined with the infinite-norm (and not a L_2-norm). More precisely, we believe that the problem in their proof is just below their Equation (13) when Lemma 1(ii) is used. What their Lemma 1 implies is that $M(t)$ is a Martingale such that $\mathbb{E}[\|M(t+1) - M(t)\|^2] \leq c$. This is used below their Equation (13) to imply that $\mathbb{E}[\|M(t)\|^2] \leq ct$. The problem is that this is true if the norm $\|M\|$ is a L_2 norm (or any norm that can be written as a scalar product $\|M\|^2 = \langle M, M \rangle$) but not if $\|M\|$ it is a supremum norm. The norm used in [60] is a supremum norm and we do not believe it can be derived from a scalar product. The approach that we take in this chapter is radically different as we work with a comparison of generators. This allows us to correct the proof of [60] by obtaining a tighter bound. Note that we do not claim that their result is false, but only that the proof is false. We explain in Appendix 3.8.3 that their result is a consequence of ours (and that our results give a finer bound).

3.3 The Heterogeneous Population Model

3.3.1 Interaction Model

We consider a population model composed of N interacting objects. Each object evolves in a finite[1] state space $\mathcal{S}$. The state of the k-th object at time t is denoted by $S_k^{(N)}(t) \in \mathcal{S}$ and the state of the system at time t is given by $\boldsymbol{S}^{(N)}(t) = (S_1^{(N)}(t), \dots, S_N^{(N)}(t)) \in \mathcal{S}^N$. We assume that the stochastic process $\boldsymbol{S}^{G^N} = (\boldsymbol{S}^{(N)}(t))_{t\geq 0}$ is a continuous time Markov chain (CTMC) whose transitions are the results of interactions between objects. More precisely, we assume for any tuple of d objects $\boldsymbol{k} = (k_1, .., k_d)$, that these objects jump *simultaneously*[2] from their states $\boldsymbol{s} = (s_1, \dots, s_d)$ to $\boldsymbol{s}' = (s'_1, \dots, s'_d) \neq (s_1, \dots, s_d)$ at rate $\frac{1}{dN^{d-1}} r^{(N)}_{\boldsymbol{k},\boldsymbol{s}\to\boldsymbol{s}'}$. All such interactions occur independently. We also assume that $d \leq d_{\max}$ is a constant independent of N, i.e., the maximal amount of interacting objects does not scale with the system size.

Throughout the chapter, we will refer to such a model as a *heterogeneous population model*. Note that while a transition can affect up to $d_{\max}$ objects, all the examples studied in the chapter will be with $d_{\max} = 2$ for which there are two types of transitions:

- $d = 1$: An object jumps without interacting with others. We call this a unilateral transition.
- $d = 2$: Two objects interact. We call it a pairwise interaction.

The critical assumption of our model is that the interactions between d objects scale as $O(\frac{1}{N^{d-1}})$. In particular, the rates of unilateral transitions scale like $O(1)$ and the one of pairwise interactions like $O(1/N)$. This $1/N^{d-1}$ factor is here because there are $O(N^{d-1})$ tuples of d objects. Hence, our condition implies that the total rate of transitions is $O(N)$ and that one tuple cannot have much higher rates than other tuples.

To simplify notations, we assume that for any permutation σ of the set $\{1 \dots d\}$, the rates satisfy $r^{(N)}_{k_1,\dots,k_d,(s_1,\dots,s_d)\to(s'_1,\dots,s'_d)} = r^{(N)}_{k_{\sigma(1)},\dots,k_{\sigma(d)},(s_{\sigma(1)},\dots,s_{\sigma(d)})\to(s'_{\sigma(1)},\dots,s'_{\sigma(d)})}$. This does not imply that objects are homogeneous but should be seen at a notation artifact. An alternative notation would be to consider tuples such that $k_1 < k_2 < \dots < k_d$ and to multiply all rates by $d!$. This would lead to the same model but at the price of heavier notations.

3.3.2 State Representation

The key element of our analysis is to use an alternative, binary based, representation of the state space. For an object $k \in \{1 \dots N\}$ and a state $s \in \mathcal{S}$, we define $X^{(N)}_{(k,s)}(t)$ as

$$X^{(N)}_{(k,s)}(t) = \mathbf{1}_{\{S_k^{(N)}(t)=s\}} := \begin{cases} 1 & \text{if object } k \text{ is in state } s \text{ at time } t, \\ 0 & \text{otherwise.} \end{cases}$$

[1] The fact that objects share the same state space is done without loss of generality as we do not assume that Markov chains are irreducible.

[2] Note that for the transitions caused by interactions, we do not impose that all objects jump: we may have $s_i = s'_i, i \in \{1, \dots, d\}$ in which case some objects keep their state.

The state of the system is described by $\boldsymbol{X}^{(N)}(t) = (X^{(N)}_{(k,s)}(t))_{k\in\{1\dots N\},s\in\mathcal{S}}$ with $\mathcal{X}^{(N)} \subset \{0,1\}^{N\times|\mathcal{S}|}$ being the set of attainable states for $\boldsymbol{X}^{(N)} = (\boldsymbol{X}^{(N)}(t))_{t\geq 0}$. In particular, for all $\boldsymbol{x} \in \mathcal{X}^{(N)}$, one has $\sum_{s\in\mathcal{S}} x_{(k,s)} = 1$ which follows from the fact that an object can only be in one state at any time. The notation $\boldsymbol{X}^{(N)}$ is less compact than the original representation $\boldsymbol{S}^{(N)}$ but will allow for an easier definition of the mean field and refined mean field approximation.

The transitions of the model can all be expressed in terms of $\boldsymbol{X}^{(N)}$. Let $\boldsymbol{e}^{(N)}_{(k,s)}$ denote a matrix of size $N \times |S|$ whose (k, s) component is equal to 1, all others being equal to 0. If object k transitions from state s to state s', $\boldsymbol{X}^{(N)}$changes into $\boldsymbol{X}^{(N)} + \boldsymbol{e}^{(N)}_{(k,s')} - \boldsymbol{e}^{(N)}_{(k,s)}$. Hence, expressed as a function of $\boldsymbol{X}^{(N)}$, the process $\boldsymbol{X}^{(N)}$ jumps to (for $k_1, k_2, \dots \in \{1\dots N\}$ and $s_1, s'_1, s_2, s'_2, \dots \in \mathcal{S}$):

$$\boldsymbol{X}^{(N)} + \boldsymbol{e}^{(N)}_{(k_1,s'_1)} - \boldsymbol{e}^{(N)}_{(k_1,s_1)} \quad \text{at rate} \quad r^{(N)}_{k_1,(s_1)\to(s'_1)} X^{(N)}_{(k_1,s_1)} \tag{3.2a}$$

$$\boldsymbol{X}^{(N)} + \boldsymbol{e}^{(N)}_{(k_1,s'_1)} - \boldsymbol{e}^{(N)}_{(k_1,s_1)} + \boldsymbol{e}^{(N)}_{(k_2,s'_2)} - \boldsymbol{e}^{(N)}_{(k_2,s_2)}$$
$$\text{at rate} \frac{1}{2N} r^{(N)}_{k_1,k_2,(s_1,s_2)\to(s'_1,s'_2)} X^{(N)}_{(k_1,s_1)} X^{(N)}_{(k_1,s_1)} \tag{3.2b}$$

$$\boldsymbol{X}^{(N)} + \boldsymbol{e}^{(N)}_{(k_1,s'_1)} - \boldsymbol{e}^{(N)}_{(k_1,s_1)} + \dots + \boldsymbol{e}^{(N)}_{(k_d,s'_d)} - \boldsymbol{e}^{(N)}_{(k_d,s_d)}$$
$$\text{at rate } \frac{1}{dN^{d-1}} r^{(N)}_{k_1,\dots,k_d,\boldsymbol{s}\to\boldsymbol{s}'} X^{(N)}_{(k_1,s_1)} \dots X^{(N)}_{(k_d,s_d)} \tag{3.2c}$$

In the above equations (3.2a) corresponds to a unilateral transition of object k_1 from s_1 to s'_1, (3.2b) corresponds to transitions caused by a pairwise interaction between object k_1 and k_2 and (3.2c) describes the general form of the transitions of d interacting objects. Recall that we assume that $(s_1, s_2, \dots) \neq (s'_1, s'_2, \dots)$ but we do not necessarily assume all states change, i.e., a pairwise interaction might result in either one object changing state or two objects that change state simultaneously.

3.3.3 Main Notations

Throughout the chapter, we use bold letters (like $\boldsymbol{X}^{(N)}, \boldsymbol{x}, \dots$) to denote matrices and regular letters (like $X^{(N)}_{(k,s)}, x_{(k,s)}, N, \dots$) to denote scalars. Capital letters (like $\boldsymbol{X}^{(N)}, \boldsymbol{S}^{(N)}$) denote random variables whereas lower case letters ($\boldsymbol{x}, v^{(N)}_{(k,s)}, \dots$) are for deterministic values. The indices $k, k_1, k', \dots$ are reserved for objects while $s, s_1, s' \dots$ are reserved for the states.

In the results below, when we write that a quantity h satisfies $h = O(1)$ or $h = O(1/N)$, this means that there exists a constant C independent of N such that $h \leq C$ or $h \leq C/N$. In general, these constants do depend on other parameters of the problem (like $|\mathcal{S}|$, $\bar{r}$, or t).

3.4 Main Results

In this section, we define the mean field and refined mean field approximation for the heterogeneous system and formulate the corresponding theorems.

3.4.1 Drift and Mean Field Approximation

For a given state $\boldsymbol{x} \in \mathcal{X} \subset \{0,1\}^{N\times|\mathcal{S}|}$, we define the drift of the system in $\boldsymbol{x}$ as the expected variation of the stochastic process $\boldsymbol{X}^{(N)}$ at time t:

$$f^{(N)}(\boldsymbol{X}^{(N)}(t)) = \lim_{dt\to 0} \frac{1}{dt}\mathbb{E}[\boldsymbol{X}^{(N)}(t+dt) - \boldsymbol{X}^{(N)}(t) \mid \boldsymbol{X}^{(N)}(t) = \boldsymbol{x}].$$

Based on the transitions (3.2), if $d \leq 2$ (*i.e.*, only unilateral and pairwise interactions)[3], the (k,s) component of the drift can be expressed as:

$$\sum_{s'\neq s}(r^{(N)}_{k,(s')\to(s)}x_{(k,s')} - r^{(N)}_{k,(s)\to(s')}x_{(k,s)}) \tag{3.3}$$
$$+\frac{1}{N}\sum_{s',k_1,s_1,s_1'}(r^{(N)}_{k,k_1,(s',s_1')\to(s,s_1)}x_{(k,s')}x_{(k_1,s_1)} - r^{(N)}_{k,k_1,(s,s_1)\to(s',s_1')}x_{(k,s)}x_{(k_1,s_1')})$$

The first term corresponds to unilateral transitions while the second term corresponds to transitions caused by pairwise interactions. Note that compared to (3.2b), there seems to be an extra factor 2 in front of the pairwise interactions. This is not an error and it is due to the fact that we fixed the position of s in the above equation.

Note that if the conditional expectation is only defined for $\boldsymbol{x} \in \mathcal{X}^{(N)} \subset \{0,1\}^{N\times|\mathcal{S}|}$, the above expression (3.3) can be extended to a function $f^{(N)} : \mathrm{Conv}(\mathcal{X}^{(N)}) \subset [0,1]^{N\times|\mathcal{S}|} \to \mathbb{R}^{N\times|\mathcal{S}|}$, where $\mathrm{Conv}(\mathcal{X}^{(N)})$ denotes the convex hull of $\mathcal{X}$. For a given initial condition $\boldsymbol{x} \in \mathcal{X}^{(N)}$, we define the mean field approximation of the heterogeneous population model as the solution of the ODE $\frac{d}{dt}\boldsymbol{\phi}^{(N)}(\boldsymbol{x},t) = f^{(N)}(\boldsymbol{\phi}^{(N)}(\boldsymbol{x},t))$ that starts in $\boldsymbol{x} = \boldsymbol{X}^{(N)}(0)$, and we denote by $\boldsymbol{\phi}^{(N)}(\boldsymbol{x},t)$ the value of this solution at time t. The solution is unique as $f^{(N)}$ is Lipschitz-continuous (all elements of $f^{(N)}$ are polynomials) and $\boldsymbol{\phi}(\boldsymbol{x},t)$ takes values in a bounded set. Notice, from the definition of the ODE, that $\phi^{(N)}_{(k,s)} \in [0,1]$, and $\sum_s \phi^{(N)}_{(k,s)}(\boldsymbol{x},t) = 1$ for all t and $\boldsymbol{x}$ in $\mathrm{Conv}(\mathcal{X}^{(N)})$.

3.4.2 Accuracy of the Mean Field Approximation

To obtain asymptotic properties we require that there exists a uniform bound $\bar{r}$, independent of N, such that for all $s_1, s_1', s_2, s_2', \ldots \in \mathcal{S}$ and $k_1, k_2, \ldots \in \{1,\ldots,N\}$ we have:

$$r^{(N)}_{k_1,\ldots,k_d,(s_1,\ldots,s_d)\to(s_1',\ldots,s_d')} \leq \bar{r}. \tag{3.4}$$

Theorem 4. *Assume that the model, defined in Section 3.3.2, satisfies (3.4). Let $\boldsymbol{\phi}^{(N)}(\boldsymbol{x},t)$ be the solution of the ODE introduced in Section 3.4.1 with initial condition $\boldsymbol{X}^{(N)}(0) = \boldsymbol{x} \in \mathcal{X}^{(N)}$ and drift $f^{(N)}$. Then, for $(k,s) \in \{1,\ldots,N\}\times\mathcal{S}$ and $t < \infty$,*

$$\mathbb{P}(S_k(t) = s) = \mathbb{E}[X^{(N)}_{(k,s)}(t)] = \phi^{(N)}_{(k,s)}(\boldsymbol{x},t) + O(1/N). \tag{3.5}$$

[3] We give the general drift definition in Appendix 3.8.2

Key ideas of the proof. A complete proof is provided in Section 3.6.3. Here we only give a brief overview. Note that by definition of the initial value problem and the drift being a polynomial, the differentiability of the solution $\boldsymbol{\phi}$ w.r.t. the initial condition and time is given. In the first part of the proof, we borrow ideas from [55, 74] and use a Taylor's expansion of $\boldsymbol{\phi}^{(N)}$ to show that $\mathbb{P}(S_k(t) = s) - \phi^{(N)}_{(k,s)}(\boldsymbol{x}, t)$ can be bounded by a weighted sum of $\frac{\partial^2 \phi_{(k,s)}(\boldsymbol{x},t)}{\partial x_{(k',s')} \partial x_{(k'',s'')}}$. The key technical difficulty of the proof is then to show that these terms are small when the number of objects is large. To do so, we distinguish the cases where k' and k'' refer to the same object as k or not. □

The statement of the theorem can be interpreted as saying that the probability of an object k in the Markov chain $\boldsymbol{X}$ to be in state s is approximated by $\boldsymbol{\phi}_{(k,s)}$ with an accuracy of $O(1/N)$. Indeed, with this result we can obtain similar statements as for the homogeneous case where in many cases asymptotic results are proven for $Z^{(N)}_s(t) = \frac{1}{N}\sum_{k=1}^{N} X_{(k,s)}(t)$, the stochastic process describing the fraction of objects which are in state s. It should be noted that the solution of the ODE $\boldsymbol{\phi}_{(k,s)}$, taking values in $[0, 1]$, is not close to the value of $\boldsymbol{X}_{(k,s)}$, which indicates if object k is in state s and takes the values zero or one. Hence, single trajectories of the stochastic process are not comparable to the approximation.

To illustrate this result, let us consider a cache model with a total of $N = 4$ objects and a cache that can store 2 objects. We assume that the popularities follow a Zipf distribution of parameter 0.8, meaning that object k is requested at rate $\lambda_k = 1/k^{0.8}$ and use the RANDOM replacement policy. The policy exchanges objects the following way: When an object is requested and inserted in the cache, we evict another object picked uniformly at random among the two objects in the cache. Initially, the cache contains the objects 3 and 4.

In Figure 3.1, we plot the behavior of the cache as a function of time. Each plot corresponds to a different object and contains three curves: In gray we plot one stochastic trajectory of the cache, $X_{(k,\text{in})}(t)$, where $X_{(k,\text{in})} = 1$ means that the object k is in the cache and 0 that it is not. In blue, we plot the probability for object k to be in the cache at time t, $\mathbb{P}(\text{object } k \text{ in cache at time } t) = \mathbb{E}[X_{(k,\text{in})}(t)]$, which is computed by averaging over 1000 trajectories. In green, we plot the solution of the mean field approximation, $\phi^{(N)}_{(k,\text{in})}(\boldsymbol{x}, t)$. We emphasize that $X_{(k,\text{in})}(t)$ is never close to $\phi^{(N)}_{(k,\text{in})}(\boldsymbol{x}, t)$, because the former can only take the values 0 and 1 whereas the latter takes values between 0 and 1. Moreover, the latter, which is the mean field approximation, seems to provide a very good approximation for the object to be in the cache.

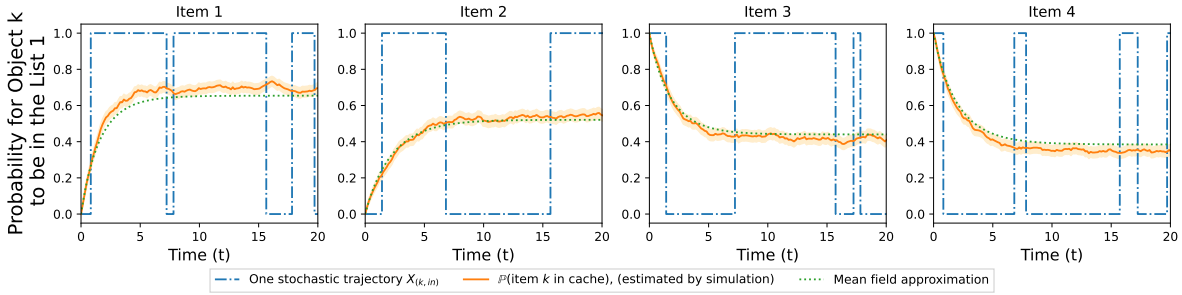


Figure 3.1: Behavior of the RANDOM policy for a cache of size two and a total of four objects. For each of the four objects, we compare the stochastic system with the mean field approximations.

3.4.3 Accuracy of the Refined Mean Field Approximation

In [54, 59], Gast et al. introduced a refined mean field approximation which provides significantly more accurate approximations in the homogeneous case. The idea is to study higher moments of $X_{(k,s)} - \phi_{(k,s)}$ and to derive refinement terms. Taking the refinements into account is especially important for small to moderate system sizes, i.e. $N \approx 10-50$, where the mean field approximation does not capture the dynamics of the stochastic system well. In this section, we show how to derive the refinement for our heterogeneous model.

To construct the refinement term, we consider an (imaginary) system in which there are C replicas of the same object. Let us denote $X_{(k,s)}^{[C \text{ copies}]}(t)$ the fraction of replicas of type k that are in state s at time t. The process $\mathbf{X}^{[C \text{ copies}]}$ is a density-dependent population process and our original process is given by $\boldsymbol{X} = \boldsymbol{X}^{[1 \text{ copy}]}$. The mean field approximation of $\boldsymbol{X}^{[C \text{ copies}]}$ is also $\boldsymbol{\phi}^{(N)}$. The idea of [54] is to study the stochastic fluctuation of $X_{(k,s)}^{[C \text{ copies}]}(t)$ around its mean field approximation. The authors show that there exists a set of deterministic values $v_{(k,s)}^{(N)}(t)$ such that

$$\mathbb{E}[\mathbf{X}_{(k,s)}^{[C \text{ copies}]}(t)] = \phi_{(k,s)}^{(N)}(\boldsymbol{x}, t) + \frac{1}{C} v_{(k,s)}^{(N)}(\boldsymbol{x}, t) + O(1/C^2). \tag{3.6}$$

The values $v_{(k,s)}^{(N)}(t)$ are shown in [54] to satisfy a system of linear ordinary differential equations whose solution can be expressed in integral form as:

$$v_{(k,s)}(\boldsymbol{x}, t) = \frac{1}{2} \int_0^t \sum_{\substack{(k_1,s_1),(k_2,s_2) \\ \in \{1,\dots,N\} \times \mathcal{S}}} Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{\phi}(\boldsymbol{x}, \tau)) \frac{\partial^2 \phi_{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)}}(\boldsymbol{\phi}(\boldsymbol{x}, \tau), t - \tau) d\tau$$

where $Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{x})$ corresponds to the expected change of the covariance between the values of $X_{(k_1,s_1)}$ and $X_{(k_2,s_2)}$ of the stochastic system at some given point $\boldsymbol{X} = \boldsymbol{x}$. We formally introduce and elaborate more on the refinement terms in Appendix 3.8.2.

In our heterogeneous population model, we have no replica which corresponds to setting $C = 1$. Hence, the above bound does not guarantee that the $O(1/C^2)$ should be small for $C = 1$. The next theorem shows that, surprisingly, using the refined approximation (3.6) with $C = 1$ copy leads to an approximation that is an order of magnitude more accurate than the mean field approximation provided before. When comparing Equation (3.5) and (3.7), what this theorem shows is that the correction $v_{(k,s)}^{(N)}(x, t)$ is of order $O(1/N)$ and is the leading term of the $O(1/N)$-term of Equation (3.5).

Note that to obtain the accuracy bound, No further assumption is needed compared to the case of the mean field approximation. That is, we assume that the parameters r are uniformly bounded.

Theorem 5. *Assume that the model, defined in Section 3.3.2, satisfies (3.4). Let $\boldsymbol{\phi}^{(N)}(\boldsymbol{x}, t)$ be the solution of the ODE introduced in Section 3.4.1 with initial condition $\boldsymbol{X}^{(N)}(0) = \boldsymbol{x} \in \mathcal{X}^{(N)}$ and drift $f^{(N)}$. Let $\boldsymbol{v}^{(N)}(\boldsymbol{x}, t)$ be the solution of the refinement term explicitly defined in Appendix 3.8.2. Then, for $(k, s) \in \{1, \dots, N\} \times \mathcal{S}$ and $t < \infty$,*

$$\mathbb{P}(S_k(t) = s) = \phi_{(k,s)}^{(N)}(\boldsymbol{x}, t) + v_{(k,s)}^{(N)}(\boldsymbol{x}, t) + O(1/N^2). \tag{3.7}$$

Key ideas of the proof. A complete proof is provided in Section 3.6.4. The proof of Theorem 5 uses the same methodology as the proof of Theorem 4 but refines the analysis to extract the term v. First, we use a second-order Taylor expansion instead of the first order expansion used in the proof of Theorem 5, this allows us to derive an expansion term in integral form. Second, we show that this expansion term is essentially equal to the refinement term v. Last, we show that the remainder terms are of order $O(1/N^2)$ by carefully studying how small the third and fourth derivatives of ϕ with respect to its initial condition are. □

The theorem shows that, by adding the refinement term, the error of the refined approximation is of order $O(1/N^2)$, which is an order of magnitude better than the $O(1/N)$ of the classical mean field approximation. This implies that both equations are asymptotically exact as the number of interacting objects goes to infinity. Hence, the refinement is especially interesting to approximate systems with few interacting objects. Note that in theory, it is possible to obtain a refined-refined approximation that has an accuracy of $O(1/N^3)$. For that, one can adapt the $1/N^2$-expansion of [54] to compute a second expansion term. This expansion depends on up to the fourth derivative of ϕ. Yet, proving carefully that this expansion is $O(1/N^3)$-accurate seems difficult as it requires obtaining precise estimates of up to the sixth derivative of ϕ. Also, from a practical point of view, computing such an expansion involves solving an ODE with $O((NS)^4)$ variables which seems difficult as soon as N grows. Hence, in this chapter, we restrict our attention to the first expansion term.

To illustrate how this refinement improves the accuracy compared to the classical mean field, we consider the same cache replacement policy as the one studied in Figure 3.1, with four objects and a cache of size 2. Compared to the previous figure, we now added an orange curve that corresponds to the refined mean field approximation. We observe that, if the mean field approximation was good, the refined mean field approximation seems almost exact.

In fact, the refined mean field approximation lies within the confidence interval of the sample mean which is calculated from 1000 sample trajectories of the underlying system. It is noticeable that computing the mean field and refinement term takes about 150ms whereas simulating 1000 sample paths and calculating the sample mean takes several seconds. This suggests that for the same computational budget, the refined mean field approximation is more accurate than the simulation. We will elaborate more on that in Section 3.5.1.

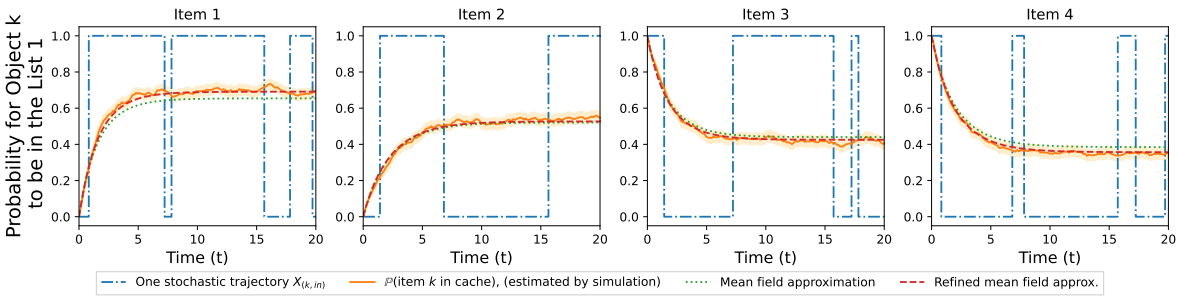


Figure 3.2: Behavior of the RANDOM policy for a cache of size 2 and a total of four objects. For each of the four objects, we compare the stochastic system with the mean field and refined mean field approximations.

On the applicability to partially heterogeneous systems

Consider the case that the stochastic system is not fully heterogeneous, i.e. there exists a finite number of classes (say N), each class having C objects that have the same behavior. Such a model corresponds to our model of C copies. As this model is a density dependent process, the results from [54, 59] show that the mean field approximation is $O(1/C)$-accurate and that the refined mean field approximation is $O(1/C^2)$ accurate for the C-copy model. We now show that Theorems 4 and 5 can be used to obtain the following much sharper bound:

$$\mathbb{E}[\mathbf{X}_{(k,s)}^{[C \text{ copies}]}(t)] = \phi_{(k,s)}^{(N)}(\boldsymbol{x}, t) + \frac{1}{C} v_{(k,s)}^{(N)}(\boldsymbol{x}, t) + O\left(\frac{1}{(CN)^2}\right). \tag{3.8}$$

The (only) difference between this equation and (3.6) is that the term $O(1/C^2)$ of (3.6) is replaced by the much smaller term $O(1/(CN)^2)$. This implies that the accuracy of the mean field interaction model does not depend on the number of homogeneous copies but only on the total number of objects.

To see why (3.8) works, we remark that our model with C copies of N classes can be represented by a fully heterogeneous model with $N' = CN$ objects (one just have to use equal rates for objects that are similar). The result of Theorem 4 and 5 imply that the mean field $\phi^{(N')}$ and refined mean field $\phi^{(N')} + v^{(N')}$ approximations are $O(1/N')$ and $O(1/(N')^2)$ accurate. By replacing n' by nC and summing over identical objects, one obtains (3.8).

3.4.4 Numerical Complexity

From a computational point of view, the mean field and refined mean field approximations greatly fasten the estimation of transient or steady-state values compared to a direct study of the original Markov process $\boldsymbol{S}^{(N)}$. Indeed, the continuous-time Markov chain $\boldsymbol{S}^{(N)}$ has up to $|\mathcal{S}|^N$ states where the mean field approximation can be computed by solving a non-linear ODE with $N|\mathcal{S}|$ variables. As shown in Appendix 3.8.2, the refinement term $\boldsymbol{v}^{(N)}$ is the solution of a linear ODE with $N|\mathcal{S}|+(N|\mathcal{S}|)^2$ variables. This means that both approximations can be solved by using standard numerical integrators. Moreover, the number of dimensions of these ODEs grow linearly (for the mean field) or quadratically (for the refinement) in the number of objects whereas an exact analysis grows exponentially with the number of objects. In some cases, computing the drifts or its derivative can be costly as it grows linearly with the total number of possible transitions at a given state $\boldsymbol{x} \in \mathcal{X}^{(N)}$: the number of transitions can grow exponentially with the maximal number of interacting objects $d_{\max}$. Yet, it is often the case that closed form expression or simplifications are obtainable. It is also noticeable that for $d_{\max} = 2$ the second derivative of the drift becomes constant, which can be used to speed up computations.

To study the time taken to compute the mean field and refined mean field approximation in more detail, we consider the RANDOM model already presented in Figure 3.2 and we vary the number of objects N from 10 to 1000. For each system size N, we measure the time to compute the four values described next and we report them in Table 3.1.

• (Transient) The first two columns correspond to the computation of the mean field approximation $x_{(k,\cdot)}^{(N)}(t)$, and the refined approximation $(x+v)_{(k,\cdot)}^{(N)}(t)$ for $t \in [0, 1000]$. The computation

	Transient up to $T = 1000$		Steady-state (=Fixed point)	
N	mean field	refined m.f.	mean field	refined m.f.
10	30ms	180ms	2ms	2ms
30	30ms	370ms	3ms	5ms
50	35ms	1s	2ms	6ms
100	60ms	14s	3ms	30ms
300	170ms	–*	9ms	200ms
500	300ms	–*	10ms	700ms
1000	970ms	–*	30ms	9s

Table 3.1: Computation time of the mean field and refined mean field approximation for the RANDOM model as a function of the number of objects N.
* "–" means that the ODE solver did not finish before 30 seconds.

is done by using a straightforward implementation of the ODEs given in Appendix 3.8.2, which is solved by using the function `solve_ivp` of `scipy`. There is no particular optimization of the code to use that a large number of terms are 0. We observe that computing the mean field approximation seems to scale linearly with N and can be done for a system of more than a $N = 1000$ objects. For the refined mean field approximation, the computation cost grows quickly when N exceeds 100 (it takes several minutes for $N = 200$ objects).

• (Steady-state). The last two columns correspond to the computation of the limiting value as t goes to infinity: $x^{(N)}_{(k,\cdot)}(\infty) = \lim_{t\to\infty} x^{(N)}_{(k,\cdot)}(t)$ and $v^{(N)}_{(k,\cdot)}(\infty) = \lim_{t\to\infty} v^{(N)}_{(k,\cdot)}(t)$. We observe that the computation of these values is much faster: the computation of the mean field approximation is essentially instantaneous whereas the computation of the refined mean field is doable for $N = 1000$. This is since the computation of the steady-state values corresponds to finding the fixed point of a linear system of ODEs, which is done by solving a linear system.

The choice between the mean field and refined mean field approximation can certainly depend on the system size as the complexity of the former grows linearly with N and quadratically for the latter. Hence, for system sizes larger than $N \geq 100$, the computation time of the refined mean field increases rapidly. As shown in Theorem 4, for large N the mean field approximation already gives a good estimate of the true values. This makes the refined approximation more interesting for reasonable system sizes (say $N \leq 100$). Note that for a homogeneous system, the complexity of computing the refined approximation does not depend on the number C of replicas. For a fully heterogeneous system, this is no longer the case because the $v^{(N)}_{(k,\cdot)}$ depends on the object's identity.

3.5 Numerical Experiments

In this section, we illustrate our main results with two examples, a cache replacement model and a two-choice load balancing model. We will see that both models fulfill the requirements for Theorems 4 and 5 and that the hidden constant given in the theorems is small. The two examples are chosen to illustrate models for which the classical, homogeneous mean field approximation cannot be used but our heterogeneous framework applies.

3.5.1 Application to a Cache Replacement Algorithm: the RANDOM($\boldsymbol{m}$) Model

As a first example we consider the list-based RANDOM($\boldsymbol{m}$) cache replacement policy which was introduced in [60, 64]. List-based cache replacement policies are used to order content in a cache separated into lists. The cache is separated into $\ell := |\mathcal{S}| - 1$ lists with sizes $m_1^{(N)}, \dots, m_\ell^{(N)}$. When an object is requested, if it is not in the cache, it is inserted in the first list and replaces a randomly chosen object from it. If the object is in a list s, it is promoted to list $s+1$ and a randomly chosen object from list $s+1$ is moved to list s. If an object that is not in the cache is requested we call it a 'miss' otherwise a 'hit'. It is shown in [60] that list-based cache replacement policies can greatly improve the hit rate compared to the classical RANDOM or LRU policies, at the price of being less responsive. The authors of [60] used a mean field approximation for which some theoretical support was given (essentially by showing that the error of the mean field approximation is $O(1/\sqrt{N})$). In this section, we push this analysis further in two directions. First, we show that our framework improves on the bound of [60] by showing that the error of the mean field approximation is $O(1/N)$ and not $O(1/\sqrt{N})$. Second, we show that our refined approximation provides an extremely accurate approximation (essentially more accurate than simulation). While this last fact was empirically observed in [36], Theorem 5 provides theoretical support by showing that the error of the refined approximation is $O(1/N^2)$.

Model and Approximations

We consider that there are N objects with identical sizes. Requests for an object k arrive according to a Poisson process of intensity λ_k. In our framework, the state of object k at time t is $S_k^{(N)}(t) \in \{0, 1 \dots \ell\}$, that represents the list in which object k is ("0" means that the object is not stored in the cache). Following our framework, we denote by $X_{(k,s)}(t)$ the random variable that equals 1 if object k is in list s and 0 otherwise. According to the RANDOM($\boldsymbol{m}$) policy, if object k is in list $s \in \{0, 1, \dots, \ell - 1\}$ and gets requested, then it is moved into list $s+1$ and a randomly selected object (say k_1) from list $s+1$ moves into list s. The corresponding transitions for the Markov chain $\boldsymbol{X}$ are:

$$\boldsymbol{X} \mapsto \boldsymbol{X} + \boldsymbol{e}_{(k,s+1)} - \boldsymbol{e}_{(k,s)} + \boldsymbol{e}_{(k_1,s)} - \boldsymbol{e}_{(k_1,s+1)} \qquad \text{at rate } \frac{\lambda_k}{m_{s+1}^{(N)}} X_{(k,s)} X_{(k_1,s+1)}. \tag{3.9}$$

Here, $\lambda_k X_{(k,s)}$ is the rate at which object k is requested while being in list s, and $X_{(k_1,s+1)}/m_s^{(N)}$ is the probability that object k_1 is in list $s+1$ and is chosen to be exchanged.

The drift of the stochastic system is the vector $(f_{(k,s)}(\boldsymbol{x}))_{(k,s)}$. By definition, for object k in list s, the (k,s) component of the drift is

$$f_{(k,s)}(\boldsymbol{x}) = \lambda_k x_{(k,s-1)} - \sum_{k_1=1}^{N} \frac{\lambda_{k_1}}{m_s} x_{(k_1,s-1)} x_{(k,s)} + \Big(\sum_{k_1=1}^{N} \frac{\lambda_{k_1}}{m_{s+1}} x_{(k_1,s)} x_{(k,s+1)} - \lambda_k x_{(k,s)}\Big) \mathbf{1}_{\{s<\ell\}}.$$

The mean field approximation is the solution of the ODE $\dot{\boldsymbol{x}} = f(\boldsymbol{x})$. Note that this ODE is the same as the one given in [60].

The transitions of this model are pairwise interactions with $b^{(N)}_{k,k_1,s,s+1,s+1,s} = \frac{\lambda_k}{m^{(N)}_{s+1}/N}$ as defined in (3.2b). There is no unilateral transition. To apply Theorems 4 and 5, we assume that the list sizes $m_i^{(N)}$ grow linearly with N which guarantees that b remains bounded by $\bar{b} = \frac{\max_k \lambda_k}{\min_s(m_s/N)}$. Therefore, the assumptions of the theorems are satisfied as soon as the values λ_k are bounded and the list sizes grow linearly with the number of objects. This guarantees that the mean field approximation is $O(1/N)$ accurate whereas the refined mean field approximation is $O(1/N^2)$ accurate.

Transient Analysis

We calculate the solution of the ODEs for the mean field and refined mean field approximation and compute the simulations by adapting the toolbox [6]. We implement the Markov chain, the drift f, the drift derivatives and $\boldsymbol{Q}$ for the approximations based on the transitions (3.9). For our numerical example, we consider a cache with $N = 20$ objects for which the request rates follows a Zipf distribution with parameter $\alpha = 0.8$, that is, $\lambda_k = A/k^\alpha$ with A being a normalizing constant. We consider a cache with three lists of sizes $m_1 = 5$, $m_2 = 3$, and $m_3 = 2$.

In Figure 3.3 we compare the mean field and refined mean field approximations of the cache popularities, *i.e.*, $\sum_{k=1}^{N} \lambda_k x_{(k,s)}$ and $\sum_{k=1}^{N} \lambda_k (x_{(k,s)} + v_{(k,s)})$ for $s = 0, 1, \ldots, 3$, against the "true" value $\sum_{k=1}^{N} \lambda_k \mathbb{E}[X_{(k,s)}]$, $s = 0, 1, \ldots, 3$ that is estimated by simulation. We compute the sample mean and the 95-percent confidence interval of the cache popularities by running 2000 Markov chain simulations. This figure shows that the mean field approximation captures the qualitative behavior of the stochastic process very well. Quantitatively, the mean field provides a good approximation but does not accurately capture the behavior of the system, especially for the third list. The values of the refined mean field approximation give a considerably better approximation. It lies within the 95-percent confidence interval of the sample mean and seems to be almost exact.

Yet, evaluating how precise the refined approximation is difficult since it lies within the confidence interval of the simulation. To study this error in more detail, Next we study the steady-state behavior of the cache, for which an exact analysis is doable when $\boldsymbol{m}$ is small enough.

Steady-state Analysis

The previous results show that the mean field and refined mean field can accurately approximate the transient behavior of the RANDOM($\boldsymbol{m}$) policy. In Figure 3.4, we compare the steady-state values

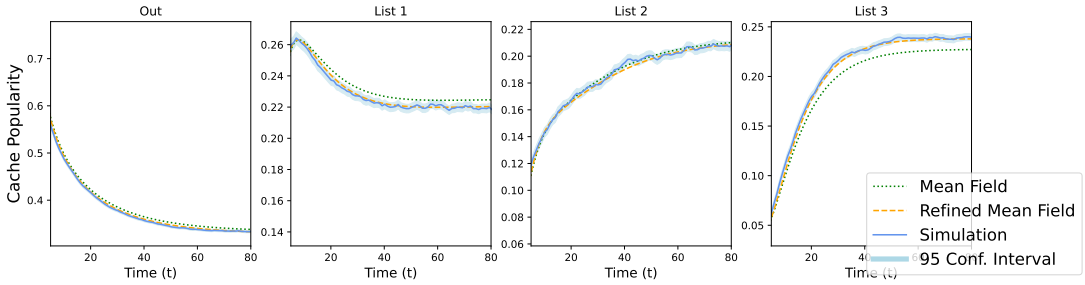


Figure 3.3: Transient state comparison of cache popularities.

of the simulation, mean field and refined mean field approximation against an exact solution. To make the figure visible, we consider a case with 8 objects having a Zipf popularity with parameter 0.5 and three lists of sizes 2. This figure shows that even for the steady-state, the mean field and refined mean field approximation are very good estimates of the true mean. As for the transient regime, this figure shows that the refined mean field approximation captures the cache popularities more closely than the mean field approximation: the curve provided by simulation and by refined mean field approximation are almost indistinguishable. Note that the bound obtained in Theorem 4 and 5 are only for the transient regime. We believe that obtaining a similar bound for the steady-state is possible but requires to precisely control how fast the mean field approximation converges to its fixed point. We leave this for future work.

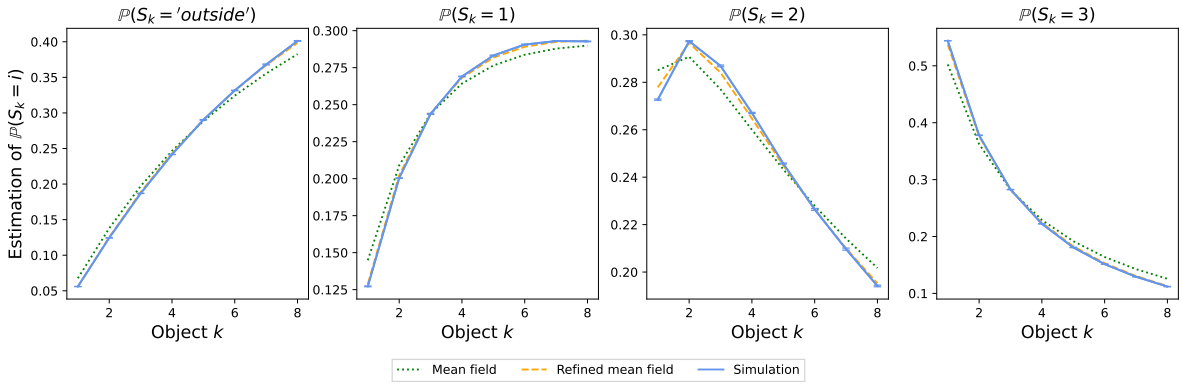


Figure 3.4: Steady-state probabilities estimated by simulation, mean field and refined mean field approximation.

While the previous figure suggests that the refined mean field is extremely accurate, it does not give a precise idea of how accurate the approximation is. To go one step further, we consider a cache model with N different objects following a Zipf popularity with parameter $\alpha = 0.5$, and a cache with two lists of size $m_1 = m_2 = \lfloor 0.3N \rfloor$. We study the accuracy of the mean field and refined mean field approximation as N grows. One difficulty to do so is that when the number of objects N is large, obtaining an accurate simulated estimation of $\mathbb{P}(S_k = s)$ for all $(k, s) \in \{1 \dots N\} \times \mathcal{S}$ is difficult. As we show below, the refined mean field seems more accurate than the simulation as soon as N is more than 20.

	Mean field			Refined mean field			Simulation	
N	Error	$N \times$ Error	(time)	Error	$N^2 \times$ Error	(time)	Error	(time)
10	0.0142	0.142	(10ms)	0.00197	0.197	(10ms)	0.00026	(4.3s)
20	0.0074	0.149	(11ms)	0.00049	0.197	(13ms)	0.00043	(4.6s)
30	0.0050	0.151	(14ms)	0.00022	0.196	(17ms)	0.00047	(4.9s)
40	0.0038	0.153	(13ms)	0.00012	0.195	(22ms)	0.00055	(6.1s)
50	0.0031	0.154	(17ms)	0.00008	0.193	(30ms)	0.00055	(5.7s)

Table 3.2: Average per-object error of three estimation methods: mean field, refined mean field and simulation. We also indicate in parentheses the time taken to compute these numbers.

We show in Appendix 3.8.3 that one can use the product form of the steady-state distribution to obtain a recurrence equation for the steady-state probability of $\mathbb{P}(S_k = s)$. While the complexity of computing this is quite large for large caches (our implementation does not allow us to compute it for more than 3 lists of size 10), it is possible to compute the exact steady-state distribution for relatively small values of $\boldsymbol{m}$. We call this value $\pi^{\text{exact}}_{(k,s)}$. We also compute an estimation $\pi^{\text{method}}_{(k,s)}$ for each method $\in \{\text{mean field, refined mean field, simulation}\}$. For the estimate computed by using simulations, we simulated 10^8 requests and estimate the steady-state probability after a warp-up period of 10^7 requests. The average error of a method is defined as

$$\text{Error}(\text{method}) = \frac{1}{N} \sum_{k,s} \left| \pi^{\text{method}}_{(k,s)} - \pi^{\text{exact}}_{(k,s)} \right|. \tag{3.10}$$

We report in Table 3.2 the error of the three estimation methods (mean field, refined mean field and simulation). By Theorem 4 and 5, we expect the average error of the mean field to be of order $O(1/N)$ and the error of the refined mean field to be of order $O(1/N^2)$. This is what we observe in Table 3.2, in which we also show the error multiplied by N or N^2 (depending on the method), to emphasize the convergence rate. We also observe that when N is larger than 20, the simulation makes more errors than the refined mean field. Note that, the value obtained by simulation is an unbiased estimator of the true value, and the error that we report arises because we can only simulate a finite number of requests. For our simulation, we choose 10^8 requests to have a reasonably fast method (it takes between 5 and 10 seconds to simulate the 10^8 requests by using an optimized C++ simulator). As a matter of comparison, we also indicate in parentheses the time taken by our implementation to compute the mean field and refined mean field approximation. Since we consider a relatively small system (at most $N = 50$ heterogeneous objects), the computation of the refined mean field approximation is fast (less than 30ms). This shows for $N \geq 30$, the refined mean field is much more accurate than the simulation, while being much faster to compute. The time taken to simulate a system of N objects grows with the number of heterogeneous objects N because the complexity of sampling from a Zipf distribution with N object grows with N. Yet, this additional computation cost is low (sampling from N objects can be done in $O(\log N)$). For the refined mean field, the situation is different and the computation time might be large for high values of N. Our experiment suggests that it is possible to compute a refined mean field approximation for a few hundred objects in a relatively fast time (less than 5 seconds). Note that for $N = 500$ objects, the error of the refined

mean field approximation is in theory 100 times smaller than the error reported in Table 3.2 for $N = 50$ (*i.e.*, extremely small).

3.5.2 Application to a Load Balancing Algorithm: The two-choice Model

Model and Approximations

In our second example, we consider a variation of the well studied two-choice model [87]. In contrast to the homogeneous case, where all servers have equal service rate parameters, we consider a heterogeneous setup in which processors can have different speeds. We study the impact of the heterogeneity of servers on their performance. Note that a similar analysis was done in [91] by using two classes of servers. The purpose of this example is twofold. It illustrates that our framework can incorporate load balancing models with heterogeneous servers. It also shows that taking heterogeneity into account in such systems is important if one wants to characterize the performance precisely.

The model consists of N servers with heterogeneous service rate parameters $\mu_k, k = 1, \dots, N$ and a finite buffer of size b, including the job in service. Jobs arrive according to a Poisson process of rate λn, we call λ the arrival rate. For each incoming job, we randomly pick two servers. The job is then assigned to the server which has the least number of unfinished jobs. If both servers have the same queue length and a full buffer, the job is discarded. Otherwise, at equal queue length, the assignment between the two servers is done at random. The service time of a job in the queue of server i is exponentially distributed with mean μ_i. The state of a server k at time t is its queue length $S_k(t) \in \mathcal{S} = \{0, 1, \dots, b\}$, state 0 is referring to the idle state.

We denote by $X_{(k,s)}$ the random variable that equals 1 if server k has s jobs. The process $\boldsymbol{X} = (X_{(k,s)})_{(k,s)}$ is a Markov chain whose transitions are (for all $k, k_1 \in \{1 \dots N\}$):

$$\boldsymbol{X} \mapsto X - e_{(k,s)} + e_{(k,s-1)} \quad \text{at rate } \mu_k X_{(k,s)}, \tag{3.11a}$$

$$\boldsymbol{X} \mapsto X + e_{(k,s+1)} - e_{(k,s)} \quad \text{at rate } (2\lambda N \mathbf{1}_{\{s_1 \geq s+1\}} + \lambda N \mathbf{1}_{\{s_1 = s\}}) \frac{X_{(k,s)}}{N} \frac{X_{(k_1,s_1)}}{N}. \tag{3.11b}$$

In the above equation, the first type of transition (3.11a) corresponds to the completion of a job by server k when the queue is of size $1 \leq s \leq b$. It reduces the queue length from s to $s-1$ which sets $X_{(k,s)}$ to 0 and $X_{(k,s-1)}$ to 1. The second type of transitions, equation (3.11b), corresponds to adding a job to a server k having $0 \leq s \leq b-1$ jobs in the buffer. In this case, the queue size is increased by one from s to $s+1$. To explain the transition rate we see that the servers k, k_1 can be selected in two ways, by selecting k or k_1 first and the other second. In the case that both queues have equal length, the chance to add the job to server k is $1/2$. If both buffers are full, the job is discarded.

This model has both, unilateral transitions with $r^{(N)}_{k,(s)\to(s-1)} = \mu_k$ and pairwise interactions with $r^{(N)}_{k,k_1,(s,s_1)\to(s+1,s_1)} = (2\lambda \mathbf{1}_{\{s_1 \geq s+1\}} + \lambda \mathbf{1}_{\{s_1 = s\}})/N$. The bound (3.4), required to apply Theorems 4 and 5, is verified when the values of λ and μ_k are bounded independently of N.

To simplify notations, let $g_s(t) = \sum_{k=1}^{N} \sum_{\tilde{s} \geq s} X_{(k,\tilde{s})}(t)/N$ be the fraction of servers at time t with queue length at least s. By summing over all possible values of k_1 and s_1, the transition (3.11b) can then be rewritten as

$$X \mapsto X + e_{(k,s+1)} - e_{(k,s)} \qquad \text{at rate } \lambda\, X_{(k,s)}(g_s + g_{s+1}).$$

By using this notation, the drift for index (k, s) is

$$f_{(k,s)}(\boldsymbol{x}) = \mu_k x_{(k,s+1)} - \lambda x_{(k,s)}(g_s + g_{s+1}) - \big(\mu_k x_{(k,s)} - \lambda x_{(k,s-1)}(g_{s-1} + g_s)\big)\mathbf{1}_{\{s \geq 1\}}.$$

Numerical Comparison

As for the caching example, we adapt methods of the toolbox [6] to perform a numerical comparison of mean field and refined mean field approximation against an estimation of the expected value of the system. Following the equations (3.11a) and (3.11b) we implement the Markov chain and define the drift f, the derivatives of f and the tensor $\boldsymbol{Q}$. To obtain the plots, we consider models with systems sizes of $N = 10, 20, 30, 40$ and an arrival rate $\lambda = 1$. The heterogeneity is introduced by the consideration of differing server rates. For every system size, we consider a model having service rates as follows. One fifth of the server rates is equal to 2.0, one fifth is equal to 0.5 and the remaining rates are sampled uniformly between 1.0 and 1.4. In transient state we calculate the sample mean for the system sizes by averaging over 2000 simulations for $N = 10, 20$, and over 3000 simulations for $N = 30, 40$. For the steady-state, the estimations are computed by calculating the independent time-average of 19×10^6 events of the Markov chain after a warp-up of 5×10^5 events. To compute the mean field and refined mean field approximation faster, we restricted the queue size of the system to a maximum of 12 (for simulation, we assume unbounded queue lengths). This is justified by two facts: First, the refined mean field seems to be very accurate even with this bounded queue size. Second, we also show in Figure 3.7 that the queue length distribution vanishes very quickly for high queue sizes. We collect all simulation results in Figures 3.5, 3.6 and 3.7.

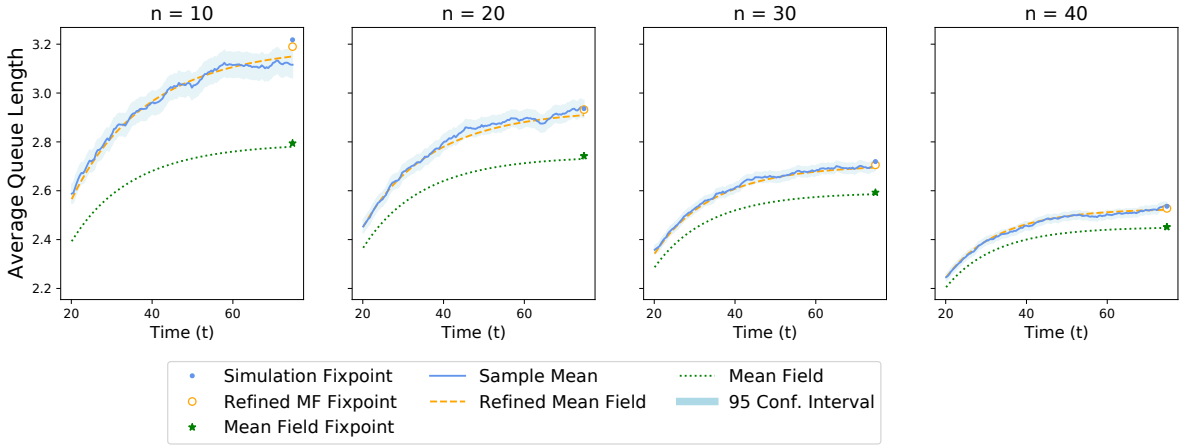
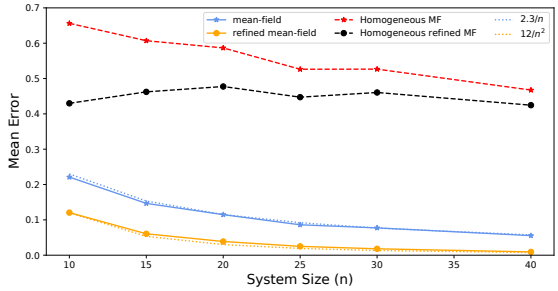


Figure 3.5: Average Queue Size of simulation mean vs. mean field vs. refined mean field approximation.

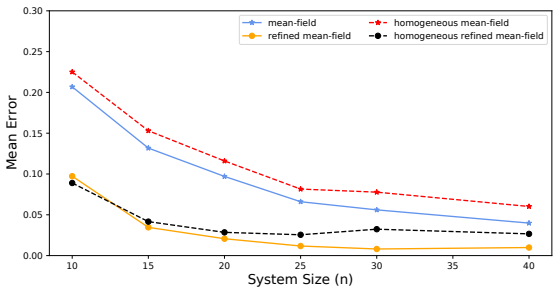
Figure 3.5 shows the average queue size of the system. We plot the sample mean of the average queue size with a 95-percent confidence interval against the average queue size calculated from the

mean field and refined mean field approximation. We observe that as N grows, both the mean field and the refined mean field approximations seem to be asymptotically exact. Note that the mean field approximation depends on N because it depends on the exact server speeds. Also, in all cases, the mean field approximation underestimates the average queue length, whereas the refined mean field approximation lies within the confidence interval. On each plot, we also show the steady-states estimates (as a single point on the right of each panel). The observation is the same as for the transient regime: the refined mean field approximation is extremely accurate also for the steady-state regime.

To demonstrate the impact of heterogeneity, we also consider an approximation (that we call the "homogeneou" approximation) in which there are N servers with speed $\bar{\mu} = (\sum_k \mu_k)/N$. We consider the corresponding mean field and refined mean field approximation. For these four approximation methods, we denote by $\text{Error}(\text{method}) = \frac{1}{N}\sum_{(k,s)} \left| \mathbb{E}[X_{(k,s)}(\infty)] - \pi^{\text{method}}_{(k,s)} \right|$ the mean error, where $\mathbb{E}[X_{(k,s)}(\infty)]$ is the steady-state of the stochastic system, approximated by simulation, and $\pi^{\text{method}}_{(k,s)}$ is the estimation of the steady-state probability for the given method. We plot these four errors as a function of N in the Figures 3.6a and 3.6b. The setup of the first figure is as described before, one fifth of the servers are of speed 2.0, one fifth are of speed 0.5 and the remaining are uniformly chosen between 1.0 and 1.4. For the second figure, all server speeds are uniformly chosen between 1.0 and 1.4. We observe that, as expected, the error of the heterogeneous mean field and refined mean field approximation decrease with N (at rate $O(1/N)$ and $O(1/N^2)$) while the error of the homogeneous mean field or refined mean field does not improve much with N. This indicates that taking heterogeneity into account is necessary to obtain accurate performance metrics in any case. We also see that for larger variance in the server rates, i.e., stronger heterogeneity, the error of the homogeneous approximation increases whereas our heterogeneous approach gives good estimates.



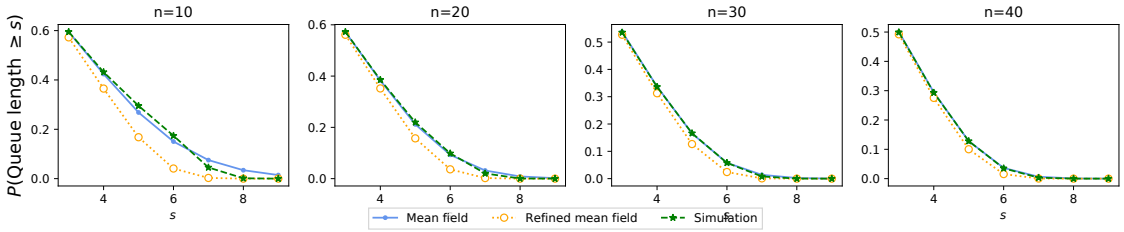
(a) Mean error for strongly varying server rates.



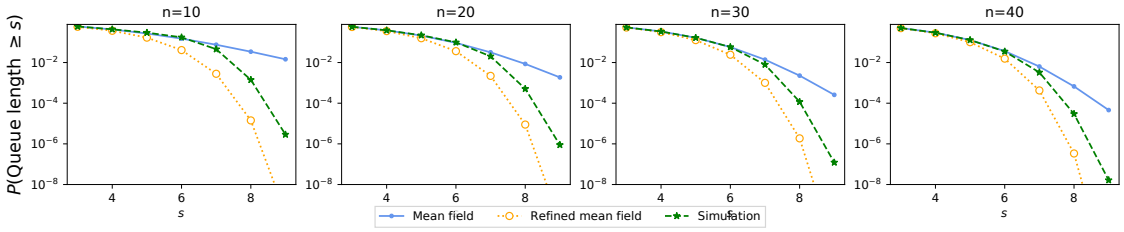
(b) Mean error for lightly varying server rates.

Figure 3.6: Steady-state mean error comparison of heterogeneous and homogeneous models.

Last, in Figure 3.7 we plot the queue length distribution tail. We plot $\frac{1}{N}\sum_k \mathbb{P}(S_k \geq s)$, the probability that a server picked at random has a queue length larger than s as a function of s. The top panel is in normal scale whereas the bottom figure is in log-scale, to zoom on the tail. We observe that for all system sizes the mean field and the refined mean field predict the shape of the distribution well. Yet, they both underestimate the actual tail distribution. The refined approximation improves notably upon the mean field method for "small" s. It does not fully correct the tail distribution for large s. Note that a similar observation was made in [59] for the refined mean field for homogeneous systems.



(a) Tail-distribution of queue length (normal-scale)



(b) Same figure with log-scale

Figure 3.7: Queue length distribution of simulated expectation, mean field approximation and refined mean field approximation in steady-state for system sizes $N = 10, 20, 30, 40$.

3.6 Proofs

This section contains the proofs of the main theorems. After recalling some notations in Section 3.6.1, we start with a first technical lemma in Section 3.6.2 in which we show that the difference between the stochastic and deterministic systems depends on the difference between the generator of the stochastic systems and the one of the ODE. Then, we prove Theorem 4 in Section 3.6.3 and Theorem 5 in Section 3.6.4. To ease the reading, some technical lemmas – whose proof are not complicated but long and technical – are postponed to the appendix.

3.6.1 Notation

In all the proofs, to ease the reading, we drop the superscript N. It should be kept in mind that all quantities $\boldsymbol{X}$, f,... depend on N. Also, instead of indexing the vectors by a pair (k, s), we will use an index $i \in \mathcal{I}$, where $\mathcal{I} = \{1 \dots N\} \times \mathcal{S}$ is the set of object-state pairs. For a function $h : \mathcal{X} \times \mathbb{R}^+ \to \mathbb{R}$, we denote by $D_{\boldsymbol{x}}h$ the derivative of h with respect to the first coordinate $\boldsymbol{x}$ and by $D_t h$ the derivative with respect to the second coordinate. This means that for a given pair $(\boldsymbol{y}, s) \in \mathcal{X} \times \mathbb{R}^+$, the quantity $D_{\boldsymbol{x}}h(\boldsymbol{y}, s)$ and $D_t h(\boldsymbol{y}, s)$ are the derivatives of h with respect to $\boldsymbol{x}$ and t evaluated at the point $(\boldsymbol{y}, s)$.

For convenience, we will denote by $K^{(N)}_{\boldsymbol{x},\boldsymbol{x}'}$ the rate at which the Markov chain $\boldsymbol{X}$ jumps from $\boldsymbol{x}$ to $\boldsymbol{x}'$ for $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$. With this notation, for intuition, the transitions (3.2a) and (3.2b) correspond to (for $\boldsymbol{x} \in \mathcal{X}$ and $k, k_1 \in \{1, \dots, N\}$, $s, s', s_1, s_1' \in \mathcal{S}$ with $(s, s_1) \neq (s', s_1')$)

$$
\begin{aligned}
K^{(N)}_{\boldsymbol{x},\boldsymbol{x}+\boldsymbol{e}_{(k,s')}-\boldsymbol{e}_{(k,s)}} &= r_{k,(s)\to(s')} X^{(N)}_{(k,s)}, \\
K^{(N)}_{\boldsymbol{x},\boldsymbol{x}+\boldsymbol{e}_{(k,s')}-\boldsymbol{e}_{(k,s)}+\boldsymbol{e}_{(k_1,s_1')}-\boldsymbol{e}_{(k_1,s_1)}} &= \frac{1}{N} r_{k,k_1,(s,s_1)\to(s',s_1')} X_{(k,s)} X_{(k_1,s_1)}.
\end{aligned}
$$

3.6.2 Comparison of the Generators

Lemma 6. *Let $\boldsymbol{X}(t)$ be the continuous time Markov chain defined in Section 3.3.2. Let $\boldsymbol{\phi}(\boldsymbol{x}, t)$ be the value at time t of the solution of the ODE $\frac{d}{dt}\boldsymbol{\phi}(\boldsymbol{x}, t) = f(\boldsymbol{\phi}(\boldsymbol{x}, t))$ with initial condition $\boldsymbol{x} \in \mathcal{X}$. We have*

$$\mathbb{E}[\boldsymbol{X}(t) - \boldsymbol{\phi}(\boldsymbol{X}(0), t))] = \int_0^t \mathbb{E}\Big[\sum_{x' \in \mathcal{X}} K_{\boldsymbol{X}(\tau), x'} \big(\boldsymbol{\phi}(\boldsymbol{x}', t-\tau) - \boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau)\big) - D_{\boldsymbol{x}}\boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau) f(\boldsymbol{X}(\tau))\Big] d\tau.$$

Proof. The following calculations are based on the ideas used in the proofs of [74, Theorem 1] and [55, Theorem 3.1]. By defining $\boldsymbol{\psi}(\tau) = \mathbb{E}[\boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau)]$ we can rewrite $\mathbb{E}[\boldsymbol{X}(t) - \boldsymbol{\phi}(\boldsymbol{X}(0), t)] = \boldsymbol{\psi}(t) - \boldsymbol{\psi}(0)$. At first, we derive the time derivative of $\boldsymbol{\psi}$. We start by looking at the expected change at a given time τ, $\frac{d}{ds}\mathbb{E}[\boldsymbol{\phi}(\boldsymbol{X}(\tau+s), t-(\tau+s)) \mid X(\tau)]\big|_{s=0}$, which can be written as

$$\lim_{ds \downarrow 0} \frac{1}{ds} \big(\mathbb{E}[\boldsymbol{\phi}(\boldsymbol{X}(\tau+ds), t-(\tau+ds)) \mid X(\tau)] - \boldsymbol{\phi}(\boldsymbol{X}(\tau), t-(\tau+ds)) + \boldsymbol{\phi}(\boldsymbol{X}(\tau), t-(\tau+ds)) - \boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau)\big).$$

In the limit, the first difference corresponds to the generator of $\boldsymbol{X}$ at τ and the second difference to the change of $\boldsymbol{\phi}$ due to the decrease of $t-\tau$. By taking the expectation and explicitly writing the limit terms, the derivative of $\boldsymbol{\psi}$ is

$$\frac{d}{d\tau}\boldsymbol{\psi}(\tau) = \mathbb{E}\Big[\sum_{x' \in \mathcal{X}} K_{\boldsymbol{X}(t), \boldsymbol{x}'}(\boldsymbol{\phi}(\boldsymbol{x}', t-\tau) - \boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau)) - D_t\boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau)\Big].$$

Note that by definition of the stochastic process the derivative with respect to time and the expectation are interchangeable. i.e., $\frac{d}{ds}\mathbb{E}[\mathbb{E}[\boldsymbol{\phi}(\boldsymbol{X}(\tau+s), t-(\tau+s)) \mid \boldsymbol{X}(\tau)]\big|_{s=0}]$ is equal to $\mathbb{E}[\frac{d}{ds}\mathbb{E}[\boldsymbol{\phi}(\boldsymbol{X}(\tau+s), t-(\tau+s)) \mid \boldsymbol{X}(\tau)]\big|_{s=0}]$. As $\boldsymbol{\phi}(\boldsymbol{x}, \cdot)$ is the solution of the ODE starting in $\boldsymbol{x}$ at time 0, we use[4] that $D_t\boldsymbol{\phi}(\boldsymbol{x}, t) = D_{\boldsymbol{x}}\boldsymbol{\phi}(\boldsymbol{x}, t) f(\boldsymbol{x})$. The proof is concluded by rewriting $\mathbb{E}[\boldsymbol{X}(t) - \boldsymbol{\phi}(\boldsymbol{X}(0), t)] = \boldsymbol{\psi}(t) - \boldsymbol{\psi}(0) = \int_0^t \frac{d}{d\tau}\boldsymbol{\psi}(\tau) d\tau$. □

3.6.3 Proof of Theorem 4 (Mean Field Approximation)

By Lemma 6, we have

$$\mathbb{E}[\boldsymbol{X}(t) - \boldsymbol{\phi}(\boldsymbol{X}(0), t)] = \int_0^t \mathbb{E}[\sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{X}(\tau), \boldsymbol{x}'}(\boldsymbol{\phi}(\boldsymbol{x}', t-\tau) - \boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau)) - D_x\boldsymbol{\phi}(\boldsymbol{X}(\tau), t-\tau) f(\boldsymbol{X}(\tau))] d\tau. \tag{3.12}$$

[4] To see why, for $t, s \geq 0$, the solution of the ODE satisfies $\frac{d}{ds}\boldsymbol{\phi}(\boldsymbol{x}, t+s) = \frac{d}{ds}\boldsymbol{\phi}(\boldsymbol{\phi}(\boldsymbol{x}, s), t)$. This shows that $\frac{d}{ds}\boldsymbol{\phi}(\boldsymbol{x}, t+s) = \frac{d}{ds}\boldsymbol{\phi}(\boldsymbol{\phi}(\boldsymbol{x}, s), t) = D_{\boldsymbol{x}}\boldsymbol{\phi}(\boldsymbol{\phi}(\boldsymbol{x}, s), t) f(\boldsymbol{\phi}(\boldsymbol{x}, s), t)$. Evaluating this expression at time $s = 0$ gives the result. Note that by definition, one also has $D_t\boldsymbol{\phi}(\boldsymbol{x}, t) = f(\boldsymbol{\phi}(x, t))$ but the latter is hard to use in the analysis.

The above expression involves terms of the form $\boldsymbol{\phi}(\boldsymbol{x}', \tau) - \boldsymbol{\phi}(\boldsymbol{x}, \tau)$. By using a first order Taylor's expansion, we have:

$$\boldsymbol{\phi}(\boldsymbol{x}', \tau) - \boldsymbol{\phi}(\boldsymbol{x}, \tau) = D_{\boldsymbol{x}}\boldsymbol{\phi}(\boldsymbol{x}, \tau)(\boldsymbol{x}' - \boldsymbol{x}) + R_1(\boldsymbol{x}, \boldsymbol{x}', \tau), \tag{3.13}$$

where $R_1(\boldsymbol{x}, \boldsymbol{x}', \tau)$ is a remainder term that can be expressed in integral form as

$$R_1(\boldsymbol{x}, \boldsymbol{x}', \tau) = \int_0^1 (1-\nu) \sum_{i,j \in \mathcal{I}} \frac{\partial^2 \boldsymbol{\phi}}{\partial x_i \partial x_j}(\boldsymbol{x} + \nu(\boldsymbol{x}' - \boldsymbol{x}), \tau)(\boldsymbol{x}'_i - \boldsymbol{x}_i)(\boldsymbol{x}'_j - \boldsymbol{x}_j) d\nu.$$

Moreover, by definition of the drift, one has $\sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}(\boldsymbol{x}' - \boldsymbol{x}) = f(\boldsymbol{x})$. Combining this with (3.13) and plugging this into equation (3.12) shows that

$$\mathbb{E}[\boldsymbol{X}(t) - \boldsymbol{\phi}(\boldsymbol{X}(0), t)] = \int_0^t \mathbb{E}[\sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{X}(\tau),\boldsymbol{x}'} R_1(\boldsymbol{X}(\tau), \boldsymbol{x}', t - \tau)] d\tau. \tag{3.14}$$

To conclude the proof, we show in Lemma 8 that $\sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'} R_1(\boldsymbol{x}, \boldsymbol{x}', \tau)$ is of order $O(1/N)$. Note that obtaining this bound is the most technical step of the proof as it requires bounding the second derivative of $\boldsymbol{\phi}$ as a function of the initial condition. This is where we use the assumptions on the rates r.

3.6.4 Proof of Theorem 5 (Refined mean field approximation)

The proof of Theorem 5 uses the same methodology as the proof of Theorem 4 with two additional ideas: The first is to use a second-order Taylor expansion instead of the first order expansion used in (3.13). The second is to express the refinement term $\boldsymbol{v}$ as an integral of quantities that depend on the second derivative of $\boldsymbol{\phi}$.

By using a second order Taylor expansion of $\boldsymbol{\phi}$, it holds that

$$\boldsymbol{\phi}(\boldsymbol{x}', \tau) - \boldsymbol{\phi}(\boldsymbol{x}, \tau) = D_{\boldsymbol{x}}\boldsymbol{\phi}(\boldsymbol{x}, \tau)(\boldsymbol{x}' - \boldsymbol{x}) + \sum_{i,j \in \mathcal{I}} Q_{i,j}(\boldsymbol{x}) \frac{\partial^2 \boldsymbol{\phi}}{\partial x_i \partial x_j}(\boldsymbol{x}, \tau) + R_2(\boldsymbol{x}, \boldsymbol{x}', \tau), \tag{3.15}$$

where the remainder term R_2 is equal to

$$R_2(\boldsymbol{x}, \boldsymbol{x}', \tau) = \frac{1}{2} \int_0^1 (1-\nu)^2 \sum_{i,j,u \in \mathcal{I}} \frac{\partial^3 \boldsymbol{\phi}}{\partial x_i \partial x_j \partial x_u}(\boldsymbol{x} + \nu(\boldsymbol{x}' - \boldsymbol{x}), \tau) \times (\boldsymbol{x}'_i - \boldsymbol{x}_i)(\boldsymbol{x}'_j - \boldsymbol{x}_j)(\boldsymbol{x}'_u - \boldsymbol{x}_u) d\nu$$

and $Q_{i,j}(\boldsymbol{x})$ is given by $\sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}(x'_i - x_i)(x'_j - x_j)$ which we formally introduce in Appendix 3.8.2. To simplify notations, let $\boldsymbol{g}(\boldsymbol{x}, \tau) := \sum_{i,j \in \mathcal{I}} Q_{i,j}(\boldsymbol{x}) \frac{\partial^2 \boldsymbol{\phi}}{\partial x_i \partial x_j}(\boldsymbol{x}, \tau)$. Similarly to (3.14), we have

$$\mathbb{E}[\boldsymbol{X}(t)] - \boldsymbol{\phi}(\boldsymbol{X}(0), t) = \underbrace{\frac{1}{2} \int_0^t \mathbb{E}[\boldsymbol{g}(\boldsymbol{X}(\tau), t-\tau)] d\tau}_{\approx \boldsymbol{v}(\boldsymbol{x},t) + O(\frac{1}{N^2}) \text{ by Lemma 9 and 10.}} \tag{3.16}$$

$$+ \int_0^t \mathbb{E}[\underbrace{\sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{X}(\tau),\boldsymbol{x}'} R_2(\boldsymbol{X}(\tau), \boldsymbol{x}', t-\tau)}_{O(1/N^2) \text{ by Lemma 8.}}] d\tau. \tag{3.17}$$

By utilizing the approach we used for R_1, we prove in Lemma 8 that the last term of the above equation (that involves a sum of R_2) is of order $O(1/N^2)$. This is quite technical and done by carefully bounding the first, second, and third derivatives of $\boldsymbol{\phi}(\boldsymbol{x}, t)$ with respect to its initial condition. We are then left with the first term of Equation (3.16). By Lemma 9, the refinement term $\boldsymbol{v}$ can be expressed in integral form as $v_{(k,s)}(\boldsymbol{x}, t) = \frac{1}{2} \int_0^t g_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{x}, \tau), \tau) d\tau$. This shows that

$$\frac{1}{2} \int_0^t \mathbb{E}[g_{(k,s)}(\boldsymbol{X}(\tau), \tau)] d\tau - v_{(k,s)}(\boldsymbol{X}(0), t)$$
$$= \frac{1}{2} \int_0^t \mathbb{E}[g_{(k,s)}(\boldsymbol{X}(\tau), \tau) - g_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{X}(0), \tau), \tau)] d\tau.$$

We show in Lemma 10 that the above term is of order $O(1/N^2)$. This requires to bound up to the fourth derivative of $\boldsymbol{\phi}$ with respect to its initial condition. Plugging everything into Equation (3.16) shows that $\mathbb{E}[\boldsymbol{X}(t)] - \boldsymbol{\phi}(\boldsymbol{X}(0), t) - \boldsymbol{v}(\boldsymbol{X}(0), t) = O(1/N^2)$ and concludes the proof.

3.7 Conclusion

In this chapter, we show how to derive mean field and a refined mean field approximation for systems composed of N heterogeneous objects. Most of the results which guarantee that mean field approximation is asymptotically correct assume that the system is composed of a population of N homogeneous objects, or at least can be clustered into a finite number of classes of objects and let the number of objects in each class goes to infinity. A possible approach to derive a (refined) mean field approximation for a heterogeneous population is to consider a scaled model with C copies of each of the N objects. Classical methods show that the (refined) mean field approximations are asymptotically exact as C grows.

Our chapter is the first to show that applying this method for the original system (with $C = 1$ object of each of the N class) is indeed valid. The main results of our chapter, Namely Theorem 4 and 5, show that the accuracy of the mean field and refined mean field approximation is $O(1/N)$ and $O(1/N^2)$. We illustrate our results by considering two examples: a model of cache replacement policies, and a load balancing model. These examples show that the proposed approximations can be computed efficiently and are very accurate. They also show that taking heterogeneity into account

is important to characterize precisely the quantitative behavior of such systems. While Monte Carlo methods can be considered an alternative to the mean field approximation methods, it is not a priori clear which one is most efficient numerically. The mean field method being deterministic, its bias is due to its theoretical error (that grows in $O(1/N)$ or $O(1/N^2)$ for the refinement) plus some (generally small) rounding errors due to the use of floating point arithmetic or numerical integrations of ODEs. For Monte-Carlo's methods, their precision is proportional to the square root of the number of samples divided by the variance of the estimator considered. In our examples, the time to calculate the sample mean and a reasonably small confidence interval can exceed the computational cost of the mean field approximation, as for the cache example shown in Section 3.5.1. One drawback of the mean field methods is that it computes the probabilities $\mathbb{E}[X_{k,s}(t)]$ for each object k and state s.

When studying the performance of large computer systems, heterogeneity is often neglected since it increases the complexity of the model and because there are few tools to analyze such systems. We believe that our work has potential application in many models and will foster the development of the analysis of heterogeneous systems (such as load balancing or epidemic models).

3.8 Appendix

3.8.1 Notation List

N	system size
$r_{\boldsymbol{k},\boldsymbol{s}\to\boldsymbol{s}'}$	transition parameters; Equations (3.2a), (3.2b), (3.2c)
$\boldsymbol{S}^{(N)}(t)$	Markov chain describing the heterogeneous population model; Section 3.3.1
$\boldsymbol{X}^{(N)}(t)$	binary based representation of the model described by $\boldsymbol{S}(t)$; Section 3.3.2
$\mathcal{S}$	finite state space of objects in the population model described by $\boldsymbol{S}$ and $\boldsymbol{X}$
$s, \hat{s}, s', s_1$	notation for states
$k, \hat{k}, k', k_1$	notation for objects
$X_{(k,s)}(t)$	entry of the Markov chain $\boldsymbol{X}$ indicating if object k is in state s at time t
$\boldsymbol{f}, \boldsymbol{Q}, \boldsymbol{R}$	drift of the stochastic system and related tensors; Section 3.4.1, Appendix 3.8.2, 8
$\boldsymbol{\phi}(\boldsymbol{x}, t)$	solution to the ODE given by the drift $\boldsymbol{f}$ of the system; Section 3.4.1
$\boldsymbol{v}(\boldsymbol{x}, t)$	refinement term; Section 3.4.3, Appendix 3.8.2
$\boldsymbol{w}(\boldsymbol{x}, t)$	solution to the second set of differential equations of the refinement; Appendix 3.8.2
t, τ, ν	time and integration variables
$\mathcal{I}, \mathcal{I}_k$	sets of object-state pairs defined by $\{1, \dots, N\} \times \mathcal{S}$ and $\{k\} \times \mathcal{S}$ respectively
i, j, w, u, l	indices used for elements of the sets $\mathcal{I}$ and $\mathcal{I}_k$
$\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$	initial conditions for Markov chain $\boldsymbol{X}$ and corresponding mean field approximation
$K_{\boldsymbol{x},\boldsymbol{x}'}$	transition rate for the Markov chain $\boldsymbol{X}$ from state $\boldsymbol{x}$ to $\boldsymbol{x}'$, Lemma 8
$L_{1,2}$	bounds for first or second partial derivatives of the drift $\boldsymbol{f}$
$\boldsymbol{x} \otimes \boldsymbol{y}$	Kronecker product of $\boldsymbol{x}$ and $\boldsymbol{y}$
$\boldsymbol{x}^{\otimes 2}, \boldsymbol{x}^{\otimes 3}$	Kronecker product of $\boldsymbol{x}$ with itself ($\boldsymbol{x}^{\otimes 2} = \boldsymbol{x} \otimes \boldsymbol{x}$; $\boldsymbol{x}^{\otimes 3} = \boldsymbol{x} \otimes \boldsymbol{x} \otimes \boldsymbol{x}$); Lemma 8
$D, D^2, D_{\boldsymbol{x}}$	1st and 2Nd order derivative, derivative with respect to $\boldsymbol{x}$
$\frac{\partial}{\partial x_i}$	partial derivative with respect to x_i
$\frac{d}{dt}, \frac{\partial}{\partial t}, \dot{x}(t)$	derivative with respect to time t
$\mathbf{1}_{\{a>b\}}$	indicator function, e.g., indicating if $a > b$

3.8.2 Equation for the Mean Field and Refined Mean Field Approximations

General drift definition

For completeness, we give the general form of the drift $f^{(N)}$ for a heterogeneous interaction model having up to $d_{\max}$ interacting objects. The drift in (k, s) is derived from interactions that imply

either object k transitions into state s ($s' \to s$) or leaves state s ($s \to s'$). By considering all these interactions, the (k, s) component of the drift $f^{(N)}(\boldsymbol{x})$ is

$$\begin{aligned}
&\sum_{s' \neq s} \left(r^{(N)}_{k,(s')\to(s)} x_{(k,s')} - r^{(N)}_{k,(s)\to(s')} x_{(k,s)}\right) \\
&\quad + \frac{1}{N} \sum_{s', k_1, s_1, s_1'} \left(r^{(N)}_{k,k_1,(s',s_1')\to(s,s_1)} x_{(k,s')} x_{(k_1,s_1)} - r^{(N)}_{k,k_1,(s,s_1)\to(s',s_1')} x_{(k,s)} x_{(k_1,s_1')}\right) \\
&\qquad + \sum_{d=3,\dots,d_{\max}} \frac{1}{N^{d-1}} \sum_{\substack{k_1,\dots,k_{d-1} \\ s_1,\dots,s_{d-1} \\ s', s_1', \dots, s_{d-1}'}} r^{(N)}_{k,\dots,k_{d-1},(s',\dots,s_{d-1}')\to(s,\dots,s_{d-1})} x^{(N)}_{(k,s')} \dots x^{(N)}_{(k_{d-1},s_{d-1}')} \\
&\qquad\qquad - r^{(N)}_{k,\dots,k_{d-1},(s,\dots,s_{d-1})\to(s',\dots,s_{d-1}')} x^{(N)}_{(k,s)} \dots x^{(N)}_{(k_{d-1},s_{d-1})}
\end{aligned} \tag{3.18}$$

In the above equation, we sum over all permutations such that the first object is fixed to k. This counters the factor $1/d$ in the definition of the rates. Therefore, the sum can be written as $\frac{1}{N^{d-1}} \sum_{\substack{k_1,\dots,k_{d-1} \\ s_1,\dots,s_{d-1} \\ s', s_1, \dots, s_{d-1}'}} r^{(N)}_{k,\dots,k_{d-1},(s',\dots,s_{d-1}')\to(s,\dots,s_{d-1})} x^{(N)}_{(k,s')} \dots x^{(N)}_{(k_{d-1},s_{d-1}')}$. Without loss of generality, we fix the index order $k, k_1, \dots$ to simplify the mathematical notations. This simplification comes from the fact that we assumed for any permutation σ of the set $\{1 \dots d\}$, the rates satisfy $r^{(N)}_{k_1,\dots,k_d,(s_1,\dots,s_d)\to(s_1',\dots,s_d')} = r^{(N)}_{k_{\sigma(1)},\dots,k_{\sigma(d)},(s_{\sigma(1)},\dots,s_{\sigma(d)})\to(s'_{\sigma(1)},\dots,s'_{\sigma(d)})}$.

Definition of the Refined Mean Field Approximation

In this section, we show how the definition of the refinement term $\boldsymbol{v}$ from [54] can be adapted and how it can be computed using the rates of the model introduced in Section 3.3. In [54], the refinement term is based on a density representation of the stochastic system and therefore independent of the state of individual objects. Since our model representation takes the state of each object into account, we extend the definition of their refinement term $\boldsymbol{v}$ to object-state pairs with the following set of ODEs (for better readability we suppress the dependence on N in the definitions)

$$\begin{aligned}
\dot{v}_{(k_1,s_1)}(\boldsymbol{x}, t) &= \sum_{u \in \mathcal{I}} \frac{\partial \boldsymbol{f}_{(k_1,s_1)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x}, t)) v_u(\boldsymbol{x}, t) + \frac{1}{2} \sum_{u,l \in \mathcal{I}} \frac{\partial^2 \boldsymbol{f}_{(k_1,s_1)}}{\partial x_l \partial x_u}(\boldsymbol{\phi}(\boldsymbol{x}, t)) w_{u,l}(\boldsymbol{x}, t), \\
\dot{w}_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{x}, t) &= \sum_{u \in \mathcal{I}} w_{u,(k_2,s_2)}(\boldsymbol{x}, t) \frac{\partial \boldsymbol{f}_{(k_1,s_1)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x}, t)) \\
&\qquad + \sum_{u \in \mathcal{I}} w_{u,(k_1,s_1)}(\boldsymbol{x}, t) \frac{\partial \boldsymbol{f}_{(k_2,s_2)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x}, t)) + Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{\phi}(\boldsymbol{x}, t)),
\end{aligned}$$

with initial conditions $\boldsymbol{v}(\boldsymbol{x}, 0) = 0, \boldsymbol{w}(\boldsymbol{x}, 0) = 0$. The values of $\boldsymbol{v}$ and $\boldsymbol{w}$ should be interpreted as the leading correction terms for the first moment and covariance of $\boldsymbol{X}(t) - \boldsymbol{\phi}(\boldsymbol{x}, t)$. The value of

$\boldsymbol{Q}$ is given by the expected change of the covariance of the stochastic system which, for a given state $\boldsymbol{x}$, is

$$\boldsymbol{Q}_{(k,s),(k',s')}(\boldsymbol{x}) = \sum_{\boldsymbol{x}' \in \mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}(\boldsymbol{x}'_{(k,s)} - \boldsymbol{x}_{(k,s)})(\boldsymbol{x}'_{(k',s')} - \boldsymbol{x}_{(k',s')}).$$

To make the definitions less abstract, we give explicit formulas of $\boldsymbol{Q}$ and the derivative of the drift f when considering a heterogeneous model having at most pairwise interactions. We start by characterizing the elements of $\boldsymbol{Q}$ evaluated at $\boldsymbol{x}$ (here $s \neq s'$ and $k \neq k'$)

$$\begin{aligned}
Q_{(k,s),(k,s)}(\boldsymbol{x}) &= \sum_{s'} r_{k,(s)\to(s')} x_{(k,s)} + r_{k,(s')\to(s)} x_{(k,s')} \\
&\quad + \frac{1}{N} \sum_{k_1,s_1,s'_1,s'} r_{k,k_1,(s,s_1)\to(s',s'_1)} x_{(k,s)} x_{(k_1,s_1)} + r_{k,k_1,(s',s'_1)\to(s,s_1)} x_{(k,s')} x_{(k_1,s'_1)}, \\
Q_{(k,s),(k,s')}(\boldsymbol{x}) &= -r_{k,(s)\to(s')} x_{(k,s)} - r_{k,(s')\to(s)} x_{(k,s')} \\
&\quad + \frac{1}{N} \sum_{k_1,s_1,s'_1} -r_{k,k_1,(s,s_1)\to(s',s'_1)} x_{(k,s)} x_{(k_1,s_1)} - r_{k,k_1,(s',s_1)\to(s,s'_1)} x_{(k,s')} x_{(k_1,s_1)}, \\
Q_{(k,s),(k',s_1)}(\boldsymbol{x}) &= \frac{1}{N} \sum_{s',s'_1} r_{k,k',(s,s_1)\to(s',s'_1)} x_{(k,s)} x_{(k',s_1)} - r_{k,k',(s',s'_1)\to(s,s_1)} x_{(k,s')} x_{(k',s'_1)}.
\end{aligned}$$

The first and second partial derivatives of the drift f are given by

$$\begin{aligned}
\frac{\partial f_{(k,s)}}{\partial x_{(k,s)}}(\boldsymbol{x}) &= -\sum_{s'} r_{k,(s)\to(s')} - \frac{1}{N} \sum_{k_1 \neq k, s_1, s'_1, s' \neq s} r_{k,k_1,(s,s_1)\to(s',s'_1)} x_{(k_1,s_1)}\,, \\
\frac{\partial f_{(k,s)}}{\partial x_{(k,\tilde{s})}}(\boldsymbol{x}) &= r_{k,(\tilde{s})\to(s)} + \frac{1}{N} \sum_{k_1 \neq k, s_1, s'_1} r_{k,k_1,(\tilde{s},s_1)\to(s,s'_1)} x_{(k_1,s_1)} && \tilde{s} \neq s, \\
\frac{\partial f_{(k,s)}}{\partial x_{(\tilde{k},\tilde{s})}}(\boldsymbol{x}) &= \frac{1}{N} \sum_{s',s'_1} r_{k,\tilde{k},(s',\tilde{s})\to(s,s'_1)} x_{(k,s')} - r_{k,\tilde{k},(s,\tilde{s})\to(s',s'_1)} x_{(k,s)} && \tilde{k} \neq k,
\end{aligned}$$

$$\frac{\partial^2 f_{(k,s)}}{\partial x_{(\hat{k},\hat{s})}\partial x_{(k,s)}}(\boldsymbol{x}) = -\frac{1}{N}\sum_{s_1',s'} r_{k,\hat{k},(s,\hat{s})\to(s',s_1')} \qquad \hat{k}\neq k,$$

$$\frac{\partial^2 f_{(k,s)}}{\partial x_{(\hat{k},\hat{s})}\partial x_{(k,\tilde{s})}}(\boldsymbol{x}) = \frac{1}{N}\sum_{s_1'} r_{k,\hat{k},(\tilde{s},\hat{s})\to(s',s_1')} \qquad \hat{k}\neq k, \tilde{s}\neq s,$$

$$\frac{\partial^2 f_{(k,s)}}{\partial x_{(k,\hat{s})}\partial x_{(\tilde{k},\tilde{s})}}(\boldsymbol{x}) = \frac{1}{N}\sum_{s_1'} r_{k,\tilde{k},(\hat{s},\tilde{s})\to(s,s_1')} \qquad \tilde{k}\neq k,$$

$$\frac{\partial^2 f_{(k,s)}}{\partial x_{(k,\hat{s})}\partial x_{(\tilde{k},\tilde{s})}}(\boldsymbol{x}) = -\frac{1}{N}\sum_{s',s_1'} r_{k,\tilde{k},(s,s')\to(\tilde{s},s_1')} \qquad \tilde{k}\neq k,$$

$$\frac{\partial^2 f_{(k,s)}}{\partial x_{(\hat{k},\hat{s})}\partial x_{(\tilde{k},\tilde{s})}}(\boldsymbol{x}) = 0 \qquad \tilde{k},\hat{k}\neq k.$$

Note that if interactions of more than two objects occur, the above formulations include additional rates and higher order derivatives of the drift are non-zero.

3.8.3 Cache Replacement Policies

Computation of the exact steady-state probabilities

It is shown in [60] that the steady-state distribution of the RANDOM($\boldsymbol{m}$) cache replacement policy has a product-form, which the authors use to derive the per-object miss probability. Here, we show how to adapt the same methodology to compute the steady-state probability for an object to be in list s. Our approach is very similar to the one developed in [60] but leads to a slightly different recurrence equation.

Recall that S_k denotes the list in which object k is (where 0 means that the object is not in the cache). We say that a state $\boldsymbol{S}$ is admissible for $\boldsymbol{m}$ if the number of objects in list s is exactly m_s for all $s \in \{1,\dots,S\}$. Theorem 6 of [60] can be rephrased as follows: For any admissible state $\boldsymbol{S}$, the steady-state probability of $\boldsymbol{S}$ is equal to

$$\pi(\boldsymbol{S}) = \frac{1}{C(\boldsymbol{m},N)}\prod_{k=1}^{N}(\lambda_k)^{S_k},$$

where $C(\boldsymbol{m},N) = \sum_{\boldsymbol{S}\text{ admissible for }\boldsymbol{m}}\prod_{k=1}^{N}(\lambda_k)^{S_k}$ is a constant such that the probabilities $\pi(\boldsymbol{S})$ sum to one. Note that the constant $C(\boldsymbol{m},N)$ is not the same as the constant $E(\boldsymbol{m},N)$ defined in [60] because our configuration $\boldsymbol{S}$ does not take into account the position in a list in which an object is but only takes into account the list in which an object is: there is a $\prod_s m_s!$ factor between the two.

By decomposing the set of admissible configurations, depending on the list in which object N is (either outside the cache or in list s), we get that:

$$C(\boldsymbol{m}, N) = \sum_{\boldsymbol{S} \text{ admissible for } \boldsymbol{m}} \prod_{k=1}^{N} (\lambda_k)^{S_k} = \sum_{s=0}^{S} \sum_{\substack{\boldsymbol{S} \text{ admissible for } \boldsymbol{m} \\ \text{and object } N \text{ is in list } s}} \prod_{k=1}^{N} (\lambda_k)^{S_k}$$
$$= C(\boldsymbol{m}, N-1) + \sum_{s=1}^{S} (\lambda_N)^s C(\boldsymbol{m} - \boldsymbol{e}_s, N-1),$$

with the convention that $C(\boldsymbol{m}, N) = 0$ if $\boldsymbol{m} = \boldsymbol{0}$ or if $\sum_s m_s > N$. Indeed, there is a bijection between the admissible configurations for $\boldsymbol{m}$ with N objects in which object N is in list s and the configurations for $\boldsymbol{m} - \boldsymbol{e}_s$ with $N-1$ objects.

Similarly, the probability for object N to be in list s is the sum over all admissible configurations such that object N is in list s which corresponds to the set of admissible configurations for $\boldsymbol{m} - \boldsymbol{e}_s$ with $N-1$ objects. Hence, we have

$$\pi^{\text{exact}}_{N,s} = \frac{(\lambda_N)^s C(\boldsymbol{m} - \boldsymbol{e}_s, N-1)}{C(\boldsymbol{m}, N)}. \tag{3.19}$$

The above recurrence equations can be used to compute the exact value of $\mathbb{P}\left(S_k^{(N)} = s\right)$ for all s. By reordering the objects, it can be also used to compute the recurrence equation for all objects k. The naive complexity of such an equation grows in $O(N^2 \prod_s m_s)$ and can be lowered to $O(N \log N \prod_s m_s)$ by carefully reordering the objects. This means that for relatively small values of $\boldsymbol{m}$, it is possible to compute an exact value for $\mathbb{P}\left(S_k^{(N)} = s\right)$. Note that in practice, the complexity is quite large as soon as the list sizes grow. For instance, our implementation does not allow us to calculate the values for more than 3 lists of size 10.

Theorem 6 of [60] is a consequence of our results (and can be refined)

The cache replacement policy RAND($\boldsymbol{m}$) that we study in Section 3.5.1 is essentially the same[5] as the one studied in [60]. In [60], the authors denote by $H_s(t) = \sum_k p_k X_{k,s}(t)$ the sum of the items' popularity that are in list s at time t, and by $\rho_s(t) = \sum_k p_k x_{k,s}(t)$ its mean field approximation, where $p_k = \lambda_k / (\sum_{k'} \lambda_{k'})$ is the request probability for object k. Theorem 6 of [60] implies that for $t \leq T$

$$\mathbb{E}[\|H_s(t) - \rho_s(t)\|^2] = O(\max_k p_k + \max_s \frac{1}{m_s}),$$

[5] One difference between the two model is that we consider a continuous time model where object k is requested at rate λ_k and the authors of [60] consider a discrete-time model where object k is requested with probability $p_k = \lambda_k / \sum_\ell \lambda_\ell$. Up to re-normalizing the time by $\sum_\ell \lambda_\ell$, these two models are essentially equivalent.

which implies that if the popularities of items are such that $\lambda_k = O(1)$ and the list size are such that $m_s = O(1/N)$, then

$$\mathbb{E}[\|H_s(t) - \rho_s(t)\|^2] = O(1/N)$$

This is a $O(1/\sqrt{N})$ convergence result because it implies that

$$\mathbb{E}[\|H_s(t) - \rho_s(t)\|] = O(1/\sqrt{N}).$$

We do not think that the proof of the main result of [60] is correct. Lemma 1 implies that $M(t)$ is a Martingale such that $\mathbb{E}[\|M(t+1) - M(t)\|^2] \leq c$. Later in the proof, the authors argue that this implies that $\mathbb{E}[\|M(t)\|^2] \leq ct$. This would hold if the norm could be written as a scalar product $\|M\|_2^2 = \langle M, M \rangle$. Indeed, in such a case one would have:

$$\begin{aligned}\mathbb{E}[\|M(t+1)\|_2^2] &= \mathbb{E}[\|M(t)\|_2^2 + 2\underbrace{\langle M(t+1) - M(t), M(t)\rangle}_{=0} + \|M(t+1) - M(t)\|_2^2] \\ &\leq \mathbb{E}[\|M(t)\|_2^2] + c,\end{aligned} \tag{3.20}$$

where the second term equals 0 because $\mathbb{E}[M(t+1) - M(t) \mid M(t)] = 0$. A direct recurrence would imply that $\mathbb{E}[\|M(t+1)\|_2^2] \leq ct$.

The problem is that the norm used in [60] can be written as a supremum norm (it is a supremum norm) and we do not think that it can be written as a scalar product. This implies that one cannot use the reasoning of Equation (3.20), which means that this inequality does not hold for their case.

Yet, we claim that the result of their Theorem 6 holds, and can in fact be refined by using our approach. To see that, we rewrite the difference between H and ρ as:

$$\mathbb{E}[\|H_s(t) - \rho_s(t)\|^2] = \sum_{k_1,k_2} p_{k_1} p_{k_2} \mathbb{E}[(X_{k_1,s}(t) - x_{k_1,s}(t))(X_{k_2,s'}(t) - x_{k_2,s'}(t))] \tag{3.21}$$

We claim that the proof of Theorem 5 can be adapted to show that:

$$\mathbb{E}[(X_{k_1,s}(t) - x_{k_1,s}(t))(X_{k_2,s'}(t) - x_{k_2,s'}(t))] = \begin{cases} w_{(k_1,s),(k_2,s')} + O(1/N) & \text{if } k_1 = k_2 \\ w_{(k_1,s),(k_2,s')} + O(1/(N^2)) & \text{if } k_1 \neq k_2, \end{cases}$$

where $\boldsymbol{w}$ is defined in Appendix 3.8.2 and is such that:

$$w_{(k_1,s),(k_2,s')} = \begin{cases} O(1) & \text{if } k_1 = k_2, \\ O(1/N) & \text{if } k_1 \neq k_2. \end{cases}$$

This implies that

$$\mathbb{E}[\|H_s(t) - \rho_s(t)\|^2] = \underbrace{\sum_{k_1,k_2} p_{k_1} p_{k_2} W_{(k_1,s),(k_2,s)}(t)}_{=O(1/N)} + O(1/N^2).$$

The above equation refines Theorem 6 of [60] by not only proving that the term is of order $O(1/N)$ but also by providing the expansion term.

Cache replacement policies: time to compute the fixed point.

In Table 3.2, we show that computing the fixed point of the refined mean field approximation takes less than 50ms for a cache replacement model with two lists and $N = 50$ heterogeneous objects. To explore further how this computation time scales with N or the number of lists of the cache, we report in Table 3.3 the time to compute the fixed point of the mean field and refined mean field approximation for up to 1000 items and between 2 to 4 lists. The total number of values to be computed here is $N \times S$ where S is the number of lists. We observe that the mean field approximation is relatively fast to compute for all considered values. The refined mean field takes more time but remains reasonable when we have at most $N = 1000$ objects. For $N = 1000$ and 2 lists, the computation times is much larger (more than 10 times larger). We believe that this huge increase of computation time might be due to memory contention when `scipy` tries to solve a very big linear system (with $|N\mathcal{S}|^2 = 4$ millions of variables).

Table 3.3: Time to compute the fixed point of the mean filed and refined mean field approximation for the RAND($\boldsymbol{m}$) model for various values of N and $\boldsymbol{m}$.

N	$\mathbf{m}$	$N\lvert\mathcal{S}\rvert$	Time (mean field)	Time (Refined mean field)
30	[6, 6, 6]	90	40ms	50ms
50	[10, 10, 10]	150	50ms	72ms
100	[20, 20, 20, 20]	400	263ms	458ms
200	[40, 40, 40]	600	137ms	881ms
200	[40, 40, 40, 40]	800	370ms	2s
300	[60, 60, 60]	900	186ms	4s
500	[150, 150]	1000	121ms	6s
1000	[300, 300]	2000	222ms	71s

3.8.4 Technical lemmas

Bounds for Partial Derivatives of ϕ

In Lemma 7 we analyze the properties of the partial derivatives of $\phi_{(k,s)}(\boldsymbol{x}, t)$ with respect to the initial condition $\boldsymbol{x}$. We introduce the set $\mathcal{I} := \{1, \dots, N\} \times \mathcal{S}$ and $\mathcal{I}_k = \{k\} \times \mathcal{S}$ to simplify notations for frequently appearing sums in the lemma and proof. The set $\mathcal{I}$ encompasses all object-state tuples (k, s), the set $\mathcal{I}_k$ includes tuples (k, s) with fixed object k. We emphasize that the bounds for the partial derivatives differ substantially depending on whether $\phi_{(k,s)}(\boldsymbol{x}, t)$ is derived with respect to $(k, \hat{s}) \in \mathcal{I}_k$ or $(\tilde{k}, \tilde{s}) \in \mathcal{I} \setminus \mathcal{I}_k$. Our results show that if the sum over the states of the absolute values of the partial derivatives of $\phi_{(k,s)}(\boldsymbol{x}, t)$, $\sum_{\hat{s}\in\mathcal{S}} \left| \frac{\phi_{(k,\hat{s})}}{x_{(k,\hat{s})}}(\boldsymbol{x}, t) \right|$, is derived with respect to the same object k, i.e., in direction of a object-state pair $(k, \hat{s})$, it can be bounded independent of N. However, if the same sum is derived with respect to a pair $(\tilde{k}, \tilde{s}) \in \mathcal{I} \setminus \mathcal{I}_k$ it is of

order $O(1/N)$. Subsequently, we can show similar properties for sums of higher partial derivatives such as $\sum_{\hat{s}\in\mathcal{S}}\left|\frac{\partial\phi_{(k,\hat{s})}}{\partial x_{(\tilde{k},\tilde{s})}\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{x},t)\right|$. For the second partial derivatives we see that if at least one of the derivative direction is in $\mathcal{I}_k$ the sum is bounded by $O(1/N)$ and otherwise, if $(\tilde{k},\tilde{s}),(\hat{k},\hat{s})\in\mathcal{I}\backslash\mathcal{I}_k$, the sum is of order $O(1/N^2)$. Our analysis considers partial derivatives up to the fourth order for which we establish bounds with likewise properties. A direct consequence we frequently use is that the absolute value of the partial derivative of $\phi_{(k,s)}(\boldsymbol{x},t)$ can be bounded by the previously mentioned sums, for example $\left|\frac{\partial\phi_{(k,s)}}{\partial x_{(\tilde{k},\tilde{s})}}(\boldsymbol{x},t)\right| \leq \sum_{\hat{s}\in\mathcal{S}}\left|\frac{\phi_{(k,\hat{s})}}{\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{x},t)\right|$. Thus, the same bounds hold for $\left|\frac{\partial\phi_{(k,s)}}{\partial x_{(\tilde{k},\tilde{s})}}(\boldsymbol{x},t)\right|$ and absolute values of higher order partial derivatives.

Lemma 7. *Given the solution $\boldsymbol{\phi}$ of the ODE defined in section 3.4.1. For the partial derivatives of $\boldsymbol{\phi}_{(k,s)}$ with respect to the initial condition $\boldsymbol{x}\in\mathcal{X}$ and $(k,s)\in\mathcal{I}=\{1,\dots,N\}\times\mathcal{S}$ the following properties hold:*

(a) If i,j,w and l are in $\mathcal{I}\setminus\mathcal{I}_k=\{1,\dots,k-1,k+1,\dots,N\}\times\mathcal{S}$, i.e., none of the tuples i,j,w or l refer to object k, then

$$(a.1)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right| = O(1/N),$$

$$(a.2)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{x},t)\right| = O(1/N^2),$$

$$(a.3)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right| = O(1/N^3),$$

$$(a.4)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\right| = O(1/N^4).$$

(b) Otherwise, if any tuple i,j,w or l is in $\mathcal{I}_k=\{k\}\times\mathcal{S}$ then, for the same summations,

$$(b.1)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right| = O(1),$$

$$(b.2)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{x},t)\right| = O(1/N),$$

$$(b.3)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right| = O(1/N^2),$$

$$(b.4)\quad \sum_{s\in\mathcal{S}}\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\right| = O(1/N^3).$$

Proof. We will prove this lemma by bounding the derivative with respect to time of $\frac{\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)$ (and of the derivative of the higher order terms). The result will then follow by using Grönwall's Lemma in differential form.

First derivative – proof of (a.1) and (b.1) – Recall that $\boldsymbol{\phi}(\boldsymbol{x}, t)$ satisfies the differential equation $\frac{d}{dt}\boldsymbol{\phi}(\boldsymbol{x}, t) = f(\boldsymbol{\phi}(\boldsymbol{x}, t))$. Hence, the partial derivatives of $\frac{\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t),\ i \in \mathcal{I}$ with respect to the time t are

$$\begin{aligned}\frac{d}{dt}\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t) = \frac{\partial}{\partial x_i}\frac{d\phi_{(k,s)}}{dt}(\boldsymbol{x}, t) &= \frac{\partial (f_{(k,s)} \circ \boldsymbol{\phi})}{\partial x_i}(\boldsymbol{x}, t) \\ &= \sum_{u \in \mathcal{I}} \frac{\partial f_{(k,s)}}{\partial \phi_u}(\boldsymbol{\phi}(\boldsymbol{x}, t))\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x}, t).\end{aligned}$$

Having a closer look at the partial derivatives of f (see Appendix 3.8.2), we see that

$$\begin{aligned}\left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{x})\right| &\leq C_1 && \text{for } u \in \mathcal{I}_k, \\ \left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{x})\right| &\leq C_2/N && \text{for } u \in \mathcal{I} \setminus \mathcal{I}_k.\end{aligned}$$

Let $L_1 := \max\{C_1, C_2\}$ and define $c_i^k := \begin{cases} 1 \text{ if } i \in \mathcal{I}_k \\ 1/N \text{ otherwise} \end{cases}$. It follows that

$$\sum_{u \in \mathcal{I}} \frac{\partial f_{(k,s)}}{\partial \phi_u}(\boldsymbol{\phi}(\boldsymbol{x}, t))\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x}, t) \leq L_1 \sum_{u \in \mathcal{I}} c_u^k \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x}, t)\right|.$$

To obtain the bound on the max term, we start by bounding the change of the max with respect to time for the first partial derivatives.

$$\begin{aligned}\frac{d}{dt}\max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^{N} \left|\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t)\right| &\leq L_1 \max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^{N} \sum_{u \in \mathcal{I}} c_u^k \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x}, t)\right| \\ &\leq L_1 |\mathcal{S}| \max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^{N} \left|\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t)\right| + L_1 \frac{1}{N} |\mathcal{S}|(N-1) \max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^{N} \left|\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t)\right| \\ &\leq 2L_1 |\mathcal{S}| \max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^{N} \left|\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t)\right|\end{aligned}$$

Furthermore, for t equal to zero, $\boldsymbol{\phi}(\boldsymbol{x}, 0) = \boldsymbol{x}$ which implies that $\frac{\partial \phi_{(k,s)}(\boldsymbol{x},0)}{\partial x_i} = 1$ if $(k, s) = i$ and 0 otherwise. From this it follows directly that $\max_{i \in \mathcal{I}, s \in S} \sum_{k=1}^{N} \left|\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, 0)\right| = 1$. Applying Grönwalls Lemma to the obtained results yields $\max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^{N} \left|\frac{\partial \phi_{(k,s)}}{\partial x_i}(\boldsymbol{x}, t)\right| \leq \exp\big(2L_1|\mathcal{S}|t\big) = O(1)$.

We bound $\sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right|$ in the same manner. First, for the time derivative

$$\begin{aligned}\frac{d}{dt}\sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right| &\leq L_1|\mathcal{S}|\sum_{u\in\mathcal{I}}c_u^k\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right| \\ &\leq L_1|\mathcal{S}|\left(\sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right| + \frac{1}{N}|\mathcal{S}|^2\max_{i\in\mathcal{I},s\in\mathcal{S}}\sum_{k=1}^{N}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right|\right).\end{aligned}$$

Here we know that the second summand is $O(1)$. By definition of $\boldsymbol{\phi}$, at time zero $\sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},0)\right|$ is equal to one if i is in $\mathcal{I}_k$ and zero otherwise. Using Grönwalls Lemma, it follows

$$\sum_{s\in\mathcal{S}}\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right| = O(c_i^k) = \begin{cases}O(1) & \text{for } i\in\mathcal{I}_k,\\ O(1/N) & \text{otherwise.}\end{cases}$$

This shows (a.1) and (b.1). Note that as an important direct consequence, the same is true for $\left|\frac{\partial\phi_{(k,s)}}{\partial x_i}(\boldsymbol{x},t)\right|$.

Second derivative – proof of (a.2) and (b.2) – For the second partial derivatives we repeat the procedure by first bounding the second derivative of $\boldsymbol{\phi}$ with respect to time t. Deriving the second partial derivative of $\boldsymbol{\phi}$ with respect to time t gives

$$\begin{aligned}\frac{d}{dt}\frac{\partial^2\phi_{(k,s)}}{\partial x_j\partial x_i}(\boldsymbol{x},t) &= \frac{\partial^2 f_{(k,s)}}{\partial x_j\partial x_i}(\boldsymbol{\phi}(\boldsymbol{x},t)) = \frac{\partial}{\partial x_j}\left(\sum_{u\in\mathcal{I}}\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right) \\ &= \sum_{u,v\in\mathcal{I}}\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t) + \sum_{u\in\mathcal{I}}\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^2\phi_u}{\partial x_j\partial x_i}(\boldsymbol{x},t).\end{aligned} \tag{3.22}$$

To bound the above term, we observe that

$$\left|\frac{\partial^2 f_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{x})\right| \leq \begin{cases}C_3/N \text{ if } i \text{ or } j\in\mathcal{I}_k\\ C_4/N^2 \text{ otherwise}\end{cases}$$

for $C_3, C_4 \geq 0$. We define $L_2 = \max\{C_3, C_4\}$ with which we bound the first sum by

$$\sum_{u,v\in\mathcal{I}}\left|\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|$$
$$\leq \frac{L_2}{N}\left(\sum_{u\in\mathcal{I}_k}\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\sum_{v\in\mathcal{I}}\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|+\sum_{v\in\mathcal{I}_k}\left|\frac{\partial\phi_v}{\partial x_i}(\boldsymbol{x},t)\right|\sum_{u\in\mathcal{I}}\left|\frac{\partial\phi_u}{\partial x_j}(\boldsymbol{x},t)\right|\right)$$
$$+\frac{L_2}{N^2}\left(\sum_{u,v\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\right)$$

By previous observations $\sum_{v\in\mathcal{I}}\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right| = O(1)$ and $\sum_{u\in\mathcal{I}_k}\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right| = O(c_i^k)$ which implies that the first sum is $\frac{1}{N}L_2(O(c_i^k)+O(c_j^k))+O(\frac{1}{N^2}) = O(\frac{1}{N}(c_i^k+c_j^k))$. The second sum can be bounded, similar to the first partial derivatives, by

$$\sum_{u\in\mathcal{I}}\frac{\partial f^{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^2\phi_u}{\partial x_j\partial x_i}(\boldsymbol{x},t)\leq L_1\sum_{u\in\mathcal{I}}c_u^k\left|\frac{\partial^2\phi_u}{\partial x_j\partial x_i}(\boldsymbol{x},t)\right|.$$

Now we can derive bounds for $\max_{i,j\in\mathcal{I},s\in S}\sum_k\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{x},t)\right|$ and $\sum_s\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{x},t)\right|$. The procedure is the same as before. For the max term we get

$$\frac{d}{dt}\max_{i,j\in\mathcal{I},s\in\mathcal{S}}\sum_{k=1}^{N}\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_j\partial x_i}(\boldsymbol{x},t)\right|$$
$$\leq \max_{i,j\in\mathcal{I},s\in\mathcal{S}}\{\sum_{k=1}^{N}O(\frac{1}{N}(c_i^k+c_j^k))+L_1\sum_{k=1}^{N}\sum_{u\in\mathcal{I}}c_u^k\left|\frac{\partial^2\phi_u}{\partial x_j\partial x_i}(\boldsymbol{x},t)\right|\}$$
$$\leq O(1/N)+2L_1|\mathcal{S}|\max_{i,j\in\mathcal{I},s\in\mathcal{S}}\sum_{k=1}^{N}\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_j\partial x_i}(\boldsymbol{x},t)\right|.$$

Recall that $\boldsymbol{\phi}(\boldsymbol{x},0) = \boldsymbol{x}$ which implies that $\frac{\partial^2 \phi_{(k,s)}(\boldsymbol{x},0)}{\partial x_i \partial x_j} = 0$. Hence, $\max_{i,j\in\mathcal{I},s\in\mathcal{S}} \sum_{k=1}^{N} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},0) \right| = 0$ which allows concluding, by applying Grönwalls Lemma, that $\max_{i,j\in\mathcal{I},s\in\mathcal{S}} \sum_{k=1}^{N} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_j \partial x_i}(\boldsymbol{x},t) \right| = O(1/N)$. For $\sum_{s\in\mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right|$ we infer

$$\begin{aligned}
\frac{d}{dt} \sum_{s\in\mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right| &\leq \sum_{s\in\mathcal{S}} O(\frac{1}{N}(c_i^k + c_j^k)) + L_1 \sum_{s\in\mathcal{S}} \sum_{u\in\mathcal{I}} c_u^k \left| \frac{\partial^2 \phi_u}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right| \\
&\leq O(\frac{1}{N}(c_i^k + c_j^k)) + L_1 |\mathcal{S}| \sum_{s\in\mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right| \\
&\quad + \frac{1}{N} L_1 |\mathcal{S}|^2 \max_{i,j\in\mathcal{I},s\in\mathcal{S}} \sum_{k=1}^{N} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right| \\
&= O(\frac{1}{N}(c_i^k + c_j^k)) + L_1 |\mathcal{S}| \sum_{s\in\mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right| + O(1/N^2).
\end{aligned}$$

With $\sum_{s\in\mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},0) \right| = 0$ and Grönwall, we have $\sum_{s\in\mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{x},t) \right| = O(\frac{1}{N}(c_i^k + c_j^k))$.

Third and Fourth derivatives – For the higher order partial derivatives, the proof procedure stays the same as for the first and second partial derivatives. First, we calculate the time derivative for the partial derivatives of third and fourth order of $\boldsymbol{\phi}$. In order to obtain bounds for the max term and the sum over the states, we bound the derivatives. The time derivatives of $\frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\boldsymbol{x},t)$ and $\frac{\partial^4 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\boldsymbol{x},t)$, with $i,j,w,l \in \mathcal{I}$, are given by

$$\begin{aligned}
&\frac{d}{dt} \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\boldsymbol{x},t) \\
&= \frac{\partial}{\partial x_w} \left(\sum_{u,v\in\mathcal{I}} \frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t)) \frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t) \frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t) \right. \\
&\quad \left. + \sum_{u\in\mathcal{I}} \frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t)) \frac{\partial^2 \phi_u}{\partial x_j \partial x_i}(\boldsymbol{x},t) \right) \\
&= \sum_{u,v,o\in\mathcal{I}} \frac{\partial^3 f_{(k,s)}}{\partial x_u \partial x_v \partial x_o}(\boldsymbol{\phi}(\boldsymbol{x},t)) \frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t) \frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t) \frac{\partial \phi_o}{\partial x_w}(\boldsymbol{x},t) \\
&+ \sum_{u,v\in\mathcal{I}} \frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t)) \left(\frac{\partial^2 \phi_u}{\partial x_i \partial x_w}(\boldsymbol{x},t) \frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t) + \frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t) \frac{\partial^2 \phi_v}{\partial x_j \partial x_w}(\boldsymbol{x},t) \right) \\
&\quad + \sum_{u,v\in\mathcal{I}} \frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t)) \frac{\partial^2 \phi_u}{\partial x_i \partial x_j}(\boldsymbol{x},t) \frac{\partial \phi_v}{\partial x_w}(\boldsymbol{x},t) \\
&\quad + \sum_{u\in\mathcal{I}} \frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t)) \frac{\partial^3 \boldsymbol{\phi}_u}{\partial x_j \partial x_i \partial x_w}(\boldsymbol{x},t)
\end{aligned}$$

and

$$\begin{aligned}
&\frac{d}{dt}\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\\
&=\frac{\partial}{\partial x_l}\Big(\sum_{u,v,o\in\mathcal{I}}\frac{\partial^3 f_{(k,s)}}{\partial x_u\partial x_v\partial x_o}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\\
&+\sum_{u,v\in\mathcal{I}}\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\left(\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)+\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial^2\phi_v}{\partial x_j\partial x_w}(\boldsymbol{x},t)\right)\\
&\quad+\sum_{u,v\in\mathcal{I}}\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^2\phi_u}{\partial x_i\partial x_j}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_w}(\boldsymbol{x},t)+\sum_{u\in\mathcal{I}}\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^3\boldsymbol{\phi}_u}{\partial x_j\partial x_i\partial x_w}(\boldsymbol{x},t)\Big).
\end{aligned}$$

The above expression is equal to

$$\begin{aligned}
&\sum_{u,v,o,p\in\mathcal{I}}\frac{\partial^4 f_{(k,s)}}{\partial x_u\partial x_v\partial x_o\partial x_p}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\frac{\partial\phi_p}{\partial x_l}(\boldsymbol{x},t)\\
&\quad+\sum_{u,v,o\in\mathcal{I}}\frac{\partial^3 f_{(k,s)}}{\partial x_u\partial x_v\partial x_o}(\boldsymbol{\phi}(\boldsymbol{x},t))\left(\frac{\partial^2\phi_u}{\partial x_i\partial x_l}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right.\\
&\quad\left.+\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial^2\phi_v}{\partial x_j\partial x_l}(\boldsymbol{x},t)\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)+\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\frac{\partial^2\phi_o}{\partial x_w\partial x_l}(\boldsymbol{x},t)\right)\\
&+\sum_{u,v\in\mathcal{I}}\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\left(\frac{\partial^3\phi_u}{\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)+\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\frac{\partial^3\phi_v}{\partial x_j\partial x_l\partial x_w}(\boldsymbol{x},t)\right.\\
&\qquad\left.+\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\frac{\partial^2\phi_v}{\partial x_j\partial x_l}(\boldsymbol{x},t)+\frac{\partial^2\phi_u}{\partial x_i\partial x_l}(\boldsymbol{x},t)\frac{\partial^2\phi_v}{\partial x_j\partial x_w}(\boldsymbol{x},t)\right)\\
&\quad+\sum_{u,v\in\mathcal{I}}\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\left(\frac{\partial^2\phi_u}{\partial x_w\partial x_l}(\boldsymbol{x},t)\frac{\partial^2\phi_v}{\partial x_i\partial x_j}(\boldsymbol{x},t)+\frac{\partial\phi_v}{\partial x_w}(\boldsymbol{x},t)\frac{\partial^3\phi_u}{\partial x_i\partial x_j\partial x_l}(\boldsymbol{x},t)\right)\\
&\quad+\sum_{u\in\mathcal{I}}\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t).
\end{aligned}$$

For the third partial derivatives, we use the above equation to show first that $\max_{i,j,w\in\mathcal{I},s\in\mathcal{S}}\sum_{k=1}^{N}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right|$ is of order $O(\frac{1}{N^2})$ and that $\sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right|$ is of order $O\big(\frac{1}{N^2}(c_i^k+c_w^k+c_j^k)\big)$. To obtain a bound for the max term, we use the results obtained by the analysis of the first and second partial derivatives of $\boldsymbol{\phi}$. The overall aim is to apply Grönwalls Lemma. We bound the first three sums of the derivative, which include first and second order partial deriva-

tives of $\boldsymbol{\phi}$. We use previous analysis and bounds on the drift derivatives to obtain the following asymptotic properties. The third order partial derivatives of the drift can be bounded by

$$\left|\frac{\partial^3 f_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\boldsymbol{x})\right| \leq \begin{cases} C_5/N^2 \text{ if } i,j \text{ or } w \in \mathcal{I}_k \\ C_6/N^3 \text{ otherwise,} \end{cases}$$

with $C_5, C_6 \geq 0$ and we define $L_3 = \max\{C_5, C_6\}$. For the first sum

$$\begin{aligned}
&\sum_{u,v,o\in\mathcal{I}} \left|\frac{\partial^3 f_{(k,s)}}{\partial x_u \partial x_v \partial x_o}(\boldsymbol{\phi}(\boldsymbol{x},t))\right| \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\leq \frac{L_3}{N^3} \sum_{u,v,o\in\mathcal{I}\setminus\mathcal{I}_k} \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\quad + \frac{L_3}{N^2} \sum_{u\in\mathcal{I}_k,v,o\in\mathcal{I}} \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\quad + \frac{L_3}{N^2} \sum_{v\in\mathcal{I}_k,u,o\in\mathcal{I}} \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\quad + \frac{L_3}{N^2} \sum_{o\in\mathcal{I}_k,u,v\in\mathcal{I}} \left|\frac{\partial \phi_u}{\partial x_i}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&= O\big(\frac{1}{N^3}\big) + O\bigg(\frac{1}{N^2}(c_i^k + c_j^k + c_w^k)\bigg) = O\bigg(\frac{1}{N^2}(c_i^k + c_j^k + c_w^k)\bigg)
\end{aligned}$$

and for the second type of sums

$$\begin{aligned}
&\sum_{u,v\in\mathcal{I}} \left|\frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\right| \left|\frac{\partial^2 \phi_u}{\partial x_i \partial x_w}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \\
&\leq L_2 \frac{1}{N^2} \sum_{u,v\in\mathcal{I}\setminus\mathcal{I}_k} \left|\frac{\partial^2 \phi_u}{\partial x_i \partial x_w}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| + L_2 \frac{1}{N} \sum_{u\in\mathcal{I}_k,v\in\mathcal{I}} \left|\frac{\partial^2 \phi_u}{\partial x_i \partial x_w}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \\
&\quad + L_2 \frac{1}{N} \sum_{u\in\mathcal{I},v\in\mathcal{I}_k} \left|\frac{\partial^2 \phi_u}{\partial x_i \partial x_w}(\boldsymbol{x},t)\right| \left|\frac{\partial \phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \\
&\leq O(\frac{1}{N^3}) + L_2 \frac{1}{N} \bigg(O\big(\frac{1}{N}(c_i^k + c_w^k)\big)|S|O(1) + O(c_j^k)|S|O(\frac{1}{N}) \bigg) = O\bigg(\frac{1}{N^2}(c_i^k + c_w^k + c_j^k)\bigg).
\end{aligned}$$

The above statement also holds for any permutation of i, j and w. By summing the above terms over k we see, by definition of the c_i^k's, that

$\sum_{k=1}^{N}\sum_{u,v\in\mathcal{I}}\left|\frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right| = O(\frac{1}{N^2})$. The third sum of interest can be bounded by

$$\sum_{k=1}^{N}\sum_{u\in\mathcal{I}}\left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^3\phi_u}{\partial x_j\partial x_i\partial x_w}(\boldsymbol{x},t)\right| \leq 2|S|K_f \max_{i,j,w\in\mathcal{I},s\in\mathcal{S}}\{\sum_{k=1}^{N}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_k}(\boldsymbol{x},t)\right|\}.$$

We furthermore note that at time t=0 the third partial derivatives of $\phi_{(k,s)}$ are zero. In combination with the obtained bounds for the sums and by applying Grönwall it is shown that $\max_{i,j,w\in\mathcal{I},s\in\mathcal{S}}\sum_{k=1}^{N}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right| = O(\frac{1}{N^2})$. Next, we show that the sum over the states $\sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right|$ is bounded by $O\big(\frac{1}{N^2}(c_i^k+c_w^k+c_j^k)\big)$. First, we recall that the first two sums of the corresponding derivative are bounded by $O\big(\frac{1}{N^2}(c_i^k+c_w^k+c_j^k)\big)$. Second,

$$\begin{aligned}
&\sum_{s\in\mathcal{S}}\sum_{u\in\mathcal{I}}\left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^3\phi_u}{\partial x_j\partial x_i\partial x_w}(\boldsymbol{x},t)\right| \\
&\leq K_f\sum_{s\in\mathcal{S}}\left(\sum_{u\in\mathcal{I}_k}\left|\frac{\partial^3\phi_u}{\partial x_j\partial x_i\partial x_w}(\boldsymbol{x},t)\right| + \frac{1}{N}\sum_{u\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial^3\phi_u}{\partial x_j\partial x_i\partial x_w}(\boldsymbol{x},t)\right|\right) \\
&= K_f|S|\sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_j\partial x_i\partial x_w}(\boldsymbol{x},t)\right| + O(1/N^3).
\end{aligned}$$

Summarized, we bound $\sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right|$ by

$$\sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right| \leq O\left(\frac{1}{N^2}(c_i^k+c_w^k+c_j^k)\right) + O(1/N^3) + \sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},t)\right|.$$

Using $\sum_{s\in\mathcal{S}}\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w}(\boldsymbol{x},0)\right| = 0$ and applying Grönwalls Lemma proofs the claim. For the max term and the sum over the states of the fourth partial derivatives we repeat the same steps. First, we show that $\max_{i,j,w,l\in\mathcal{I}s\in\mathcal{S}}\sum_{k=1}^{N}\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\right|$ is bounded by $O(1/N^3)$. We bound the sums which contain first, second and third partial derivatives of $\boldsymbol{\phi}$. We use bounds on the derivatives of the drift up to the fourth order, for which

$$\left|\frac{\partial^4 f_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x})\right| \leq \begin{cases} C_7/N^3 \text{ if } i,j,w \text{ or } l\in\mathcal{I}_k \\ C_8/N^4 \text{ otherwise,}\end{cases} \quad \text{and define } L_4=\max\{C_7,C_8\}.$$

The bounds are

$$\begin{aligned}
&\sum_{u,v,o,p\in\mathcal{I}}\left|\frac{\partial^4 f_{(k,s)}}{\partial x_u\partial x_v\partial x_o\partial x_p}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_p}{\partial x_l}(\boldsymbol{x},t)\right| \\
&\leq \frac{L_4}{N^4}\sum_{u,v,o,p\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_p}{\partial x_l}(\boldsymbol{x},t)\right| \\
&\quad+\frac{L_4}{N^3}\sum_{u\in\mathcal{I}_k v,o,p\in\mathcal{I}}\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_p}{\partial x_l}(\boldsymbol{x},t)\right| \\
&\quad+\ldots+\frac{L_4}{N^3}\sum_{p\in\mathcal{I}_k u,v,o\in\mathcal{I}}\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_p}{\partial x_l}(\boldsymbol{x},t)\right| \\
&= O(\frac{1}{N^4})+O\big(\frac{1}{N^3}(c_i^k+c_j^k+c_w^k+c_l^k)\big),
\end{aligned}$$

$$\begin{aligned}
&\sum_{u,v,o\in\mathcal{I}}\left|\frac{\partial^3 f_{(k,s)}}{\partial x_u\partial x_v\partial x_o}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\leq \frac{L_3}{N^3}\sum_{u,v,o\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\quad+\frac{L_3}{N^2}\sum_{u\in\mathcal{I}_k,v,o\in\mathcal{I}}\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&\quad+\ldots+\frac{L_3}{N^2}\sum_{o\in\mathcal{I}_k,u,v\in\mathcal{I}}\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_o}{\partial x_w}(\boldsymbol{x},t)\right| \\
&= O(\frac{1}{N^4})+O\bigg(\frac{1}{N^3}(c_i^k+c_j^k+c_w^k+c_l^k)\bigg),
\end{aligned}$$

$$\begin{aligned}
&\sum_{u,v\in\mathcal{I}}\left|\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^3\phi_u}{\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \\
&\leq L_2\frac{1}{N^2}\sum_{u,v\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial^3\phi_u}{\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right| \\
&\quad L_2\frac{1}{N}\left(\sum_{u\in\mathcal{I}_k,v\in\mathcal{I}}\left|\frac{\partial^3\phi_u}{\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|+\sum_{u\in\mathcal{I},v\in\mathcal{I}_k}\left|\frac{\partial^3\phi_u}{\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\left|\frac{\partial\phi_v}{\partial x_j}(\boldsymbol{x},t)\right|\right) \\
&= O(\frac{1}{N^4})+L_2\frac{1}{N}\left(O(\frac{1}{N^2}(c_i^k+c_w^k+c_l^k))O(1)+O(\frac{1}{N^2})O(c_j^k)\right)=O(\frac{1}{N^3}(c_i^k+c_w^k+c_l^k+c_j^k))
\end{aligned}$$

and

$$\begin{aligned}
&\sum_{u,v\in\mathcal{I}}\left|\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_v}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial^2\phi_v}{\partial x_j\partial x_l}(\boldsymbol{x},t)\right| \\
&\leq L_2\frac{1}{N^2}\sum_{u,v\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial^2\phi_v}{\partial x_j\partial x_l}(\boldsymbol{x},t)\right| \\
&\quad + L_2\frac{1}{N}\left(\sum_{u\in\mathcal{I}_k,v\in\mathcal{I}}\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial^2\phi_v}{\partial x_j\partial x_l}(\boldsymbol{x},t)\right| + \sum_{u\in\mathcal{I},v\in\mathcal{I}_k}\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_w}(\boldsymbol{x},t)\right|\left|\frac{\partial^2\phi_v}{\partial x_j\partial x_l}(\boldsymbol{x},t)\right|\right) \\
&= L_2\frac{1}{N}\left(O(\frac{1}{N}(c_i^k+c_w^k))O(\frac{1}{N}) + O(\frac{1}{N}(c_j^k+c_l^k))O(\frac{1}{N})\right) \\
&= O(\frac{1}{N^3}(c_i^k+c_w^k+c_j^k+c_l^k)).
\end{aligned}$$

Note that the results hold for permutations of i,j,w,l. The remaining sum which appears in $\frac{d}{dt}\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)$ is $\sum_{u\in\mathcal{I}}\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)$. We see that by summing over k and applying the max, this term is bounded by

$$\begin{aligned}
&\max_{i,j,w,l\in\mathcal{I},s\in\mathcal{S}}\sum_{k=1}^N\sum_{u\in\mathcal{I}}\left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right| \\
&\qquad\leq 2|S|K_f\max_{i,j,w,l\in\mathcal{I},s\in\mathcal{S}}\{\sum_{k=1}^N\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\}.
\end{aligned}$$

Furthermore, $\max_{i,j,w,l\in\mathcal{I},s\in\mathcal{S}}\{\sum_{k=1}^N\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},0)\right|\}$ is zero. We see that $\sum_{k=1}^N O(\frac{1}{N^3}(c_i^k+c_w^k+c_j^k+c_l^k)) = O(\frac{1}{N^3})$ and, by applying Grönwall, it follows that $\max_{i,j,w,l\in\mathcal{I}s\in\mathcal{S}}\sum_{k=1}^N\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\right| = O(1/N^3)$. At last, we show that $\sum_{s\in\mathcal{S}}\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\right|$ is bounded by $O\big(\frac{1}{N^3}(c_i^k+c_j^k+c_w^k+c_l^k)\big)$. The proof follows the same principles as before for the third partial derivatives. The term $\frac{d}{dt}\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)$ can be separated into sums which are of order $O(\frac{1}{N^3}(c_i^k+c_w^k+c_l^k+c_j^k))$ and the additional term $\sum_{s\in\mathcal{S}}\sum_{u\in\mathcal{I}}\left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|$. The latter is bounded by

$$\begin{aligned}
&\sum_{s\in\mathcal{S}}\sum_{u\in\mathcal{I}}\left|\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\right|\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right| \leq K_f\sum_{s\in\mathcal{S}}\left(\sum_{u\in\mathcal{I}_k}\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\right. \\
&\left.+\frac{1}{N}\sum_{u\in\mathcal{I}\setminus\mathcal{I}_k}\left|\frac{\partial^4\phi_u}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right|\right) \leq |S|\sum_{s\in\mathcal{S}}\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_j\partial x_i\partial x_w\partial x_l}(\boldsymbol{x},t)\right| + \frac{1}{N}|S|^2O(\frac{1}{N^3}).
\end{aligned}$$

To conclude, we use the same steps as before and see that

$$\sum_{s\in\mathcal{S}}\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_i\partial x_j\partial x_w\partial x_l}(\boldsymbol{x},t)\right| = O\left(\frac{1}{N^3}(c_i^k+c_j^k+c_w^k+c_l^k)\right).$$

□

Bounds for Taylor Remainders

Lemma 8 gives bounds for sums of weighted remainder terms appearing in the proofs of Theorem 4 and 5. We respectively bound the weighted sums of the first and second order Taylor remainder term by two suprema which are of order $O(1/N)$ and $O(1/N^2)$.

Lemma 8. *For $\boldsymbol{x}\in\mathcal{X}$ and $\tau\in\mathbb{R}_+$, the remainder terms satisfy*

$$\mathbb{E}[\sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}R_1(\boldsymbol{x},\boldsymbol{x}',\tau)] \leq \sup_{\boldsymbol{x},\boldsymbol{y}\in Conv(\mathcal{X})}\frac{1}{2}\sum_{i,j\in\mathcal{I}}\left|\frac{\partial^2\phi^{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{y},\tau)\right||Q_{i,j}(\boldsymbol{x})| = O(1/N),$$

$$\mathbb{E}[\sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}R_2(\boldsymbol{x},\boldsymbol{x}',\tau)] \leq \sup_{\boldsymbol{x},\boldsymbol{y}\in Conv(\mathcal{X})}\frac{1}{6}\sum_{i,j,u\in\mathcal{I}}\left|\frac{\partial^3\phi^{(k,s)}}{\partial x_i\partial x_j\partial x_u}(\boldsymbol{y},\tau)\right||R_{i,j,u}(\boldsymbol{x})| = O(1/N^2).$$

Before starting the proof, recall that the first and second order remainder terms R_1 and R_2 defined in Section 3.6 are expressed as:

$$R_1(\boldsymbol{x},\boldsymbol{x}',\tau) = \int_0^1(1-\nu)\sum_{i,j\in\mathcal{I}}\frac{\partial^2\boldsymbol{\phi}}{\partial x_i\partial x_j}(\boldsymbol{x}+\nu(\boldsymbol{x}'-\boldsymbol{x}),\tau)(\boldsymbol{x}'_i-\boldsymbol{x}_i)(\boldsymbol{x}'_j-\boldsymbol{x}_j)d\nu,$$

$$R_2(\boldsymbol{x},\boldsymbol{x}',\tau) = \frac{1}{2}\int_0^1(1-\nu)^2\sum_{i,j,u\in\mathcal{I}}\frac{\partial^3\boldsymbol{\phi}}{\partial x_i\partial x_j\partial x_u}(\boldsymbol{x}+\nu(\boldsymbol{x}'-\boldsymbol{x}),\tau)(\boldsymbol{x}'_i-\boldsymbol{x}_i)(\boldsymbol{x}'_j-\boldsymbol{x}_j)(\boldsymbol{x}'_u-\boldsymbol{x}_u)d\nu,$$

and that, as defined in Appendix 3.8.2, $\boldsymbol{Q}$ and $\boldsymbol{R}$ are given by:

$$\boldsymbol{Q}(\boldsymbol{x}) = \sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}(\boldsymbol{x}'-\boldsymbol{x})^{\otimes 2} \qquad \text{and} \qquad \boldsymbol{R}(\boldsymbol{x}) = \sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{x},\boldsymbol{x}'}(\boldsymbol{x}'-\boldsymbol{x})^{\otimes 3},$$

where $(\boldsymbol{x}'-\boldsymbol{x})^{\otimes 2}$ and $(\boldsymbol{x}'-\boldsymbol{x})^{\otimes 3}$ are Kronecker products of $(\boldsymbol{x}'-\boldsymbol{x})$ with itself, i.e., $(\boldsymbol{x}'-\boldsymbol{x})^{\otimes 2}_{i,j} = (\boldsymbol{x}'_i-\boldsymbol{x}_i)(\boldsymbol{x}'_j-\boldsymbol{x}_j)$ and $(\boldsymbol{x}'-\boldsymbol{x})^{\otimes 3}_{i,j,u} = (\boldsymbol{x}'_i-\boldsymbol{x}_i)(\boldsymbol{x}'_j-\boldsymbol{x}_j)(\boldsymbol{x}'_u-\boldsymbol{x}_u)$. The two tensors $\boldsymbol{Q}$ and $\boldsymbol{R}$ can be naturally extended to $\text{Conv}(\mathcal{X})$ due to their entries being polynomials.

Proof. To prove the two statements, we first introduce some simplifying notations. We define $c^k_{k_1}$, with $k, k_1 \in \{1,\dots,N\}$, to be one if k equals k_1 and $1/N$ otherwise. By the definition of $\boldsymbol{Q}$ it follows that for $\boldsymbol{x}\in\text{Conv}(\mathcal{X})$:

$$\left|Q_{(k,s),(k_1,s_1)}(\boldsymbol{x})\right| = O(c^k_{k_1}) = \begin{cases} O(1) & \text{if } k = k_1, \\ O(1/N) & \text{otherwise.}\end{cases}$$

This can be seen by writing the elements of $\boldsymbol{Q}$ based on the two transition types (3.2a) and (3.2b), as shown in Section 3.8.2. From Lemma 7 we know $\left|\frac{\partial^2 \phi_{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)}}(\boldsymbol{y}, \tau)\right| = O(\frac{1}{N}(c_{k_1}^k + c_{k_2}^k))$ which holds for any $\boldsymbol{y} \in \mathrm{Conv}(\mathcal{X})$. Here, the big O notation hides the dependence on τ, L_1, L_2 and $|\mathcal{S}|$. By noting that $\sum_{k_1,k_2} c_{k_2}^{k_1} c_{k_1}^k = O(1)$, we conclude

$$\begin{aligned}
&\sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} \left|Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{x})\right| \left|\frac{\partial^2 \phi^{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)}}(\boldsymbol{y}, \tau)\right| \\
&= \sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} O(c_{k_2}^{k_1}) O(\frac{1}{N}(c_{k_1}^k + c_{k_2}^k)) = O(1/N),
\end{aligned}$$

where we hide the dependence on $|\mathcal{S}|$. To prove the second statement, we define

$$c_{k_1,k_2,k_3} = \begin{cases} 1 & \text{if } k_1 = k_2 = k_3 \\ \frac{1}{N} & \text{if } k_1 = k_2 \neq k_3 \text{ or } k_2 = k_3 \neq k_1 \text{ or } k_1 = k_3 \neq k_2 \\ \frac{1}{N^2} & \text{otherwise.} \end{cases}$$

By explicitly rewriting the entries of $\boldsymbol{R}$ as done in Section 3.8.2 for $\boldsymbol{Q}$, the tensor $\mathbf{R}$ is such that

$$\left|R_{(k_1,s_1),(k_2,s_2),(k_3,s_3)}(\boldsymbol{x})\right| = O(c_{k_1,k_2,k_3})$$

Lemma 7 states that the third partial derivatives of $\boldsymbol{\phi}$ are bounded by $\left|\frac{\partial^3 \phi^{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)} \partial x_{(k_3,s_3)}}(\boldsymbol{y}, \tau)\right| = O(\frac{1}{N^2}(c_{k_1}^k + c_{k_2}^k + c_{k_3}^k))$. From $\sum_{k_1,k_2,k_3} c_{k_1,k_2,k_3}(c_{k_1}^k + c_{k_2}^k + c_{k_3}^k) = O(1)$, it follows that the sum of the two terms above behaves as

$$\sum_{(k_1,s_1),(k_2,s_2),(k_3,s_3)\in\mathcal{I}} \left|R_{(k_1,s_1),(k_2,s_2),(k_3,s_3)}(\boldsymbol{x})\right| \left|\frac{\partial^3 \phi^{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)} \partial x_{(k_3,s_3)}}(\boldsymbol{y}, \tau)\right| = O(1/N^2).$$

□

Connection of Differential and Integral Form for the Refinement Term

The following Lemma 9 shows how to express the refinement term v and w in integral form. Both representations are of importance since we exploit the differential form for numerical computations whereas we use the integral form in the proofs of Theorem 5 and Lemma 10 which are related to the accuracy of the refined mean field approximation.

Lemma 9. *The solutions to the system of ODEs*

$$\frac{d}{dt}v_{(k,s)}(\boldsymbol{x},t) = \sum_{u\in\mathcal{I}} \frac{\partial \boldsymbol{f}_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))v_u(\boldsymbol{x},t) + \frac{1}{2}\sum_{u,l\in\mathcal{I}} \frac{\partial^2 \boldsymbol{f}_{(k,s)}}{\partial x_l \partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t)))w_{u,l}(\boldsymbol{x},t),$$

$$\begin{aligned}\frac{d}{dt}w_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{x},t) &= \sum_{u\in\mathcal{I}} w_{u,(k_2,s_2)}(\boldsymbol{x},t)\frac{\partial \boldsymbol{f}_{(k_1,s_1)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t)) \\ &\quad + \sum_{u\in\mathcal{I}} w_{u,(k_1,s_1)}(\boldsymbol{x},t)\frac{\partial \boldsymbol{f}_{(k_2,s_2)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t)) + Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{\phi}(\boldsymbol{x},t))\end{aligned}$$

can be expressed in integral form as

$$v_{(k,s)}(\boldsymbol{x},t) = \frac{1}{2}\int_0^t \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{\phi}(\boldsymbol{x},\tau))\frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)d\tau,$$

$$w_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{x},t) = \int_0^t \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{\phi}(\boldsymbol{x},\tau))\frac{\partial \phi_{(k_1,s_1)}}{\partial x_i}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)\frac{\partial \phi_{(k_2,s_2)}}{\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)d\tau.$$

Proof. For a sufficiently differentiable function $h : \mathbb{R}\times\mathbb{R}\mapsto\mathbb{R}$ we have

$$\frac{d}{dt}\int_0^t h(\tau,t)d\tau = h(t,t) + \int_0^t \frac{\partial h}{\partial t}(\tau,t)d\tau.$$

We define $h(\tau,t) = \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{\phi}(\boldsymbol{x},\tau))\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)$. Recall that $\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{\phi}(x,t),0) = 0$ which implies $h(t,t) = 0$. To calculate $\frac{\partial h}{\partial t}(\tau,t)$, we use the identity

$$\begin{aligned}\frac{d}{dt}\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau) &= \frac{\partial^2}{\partial x_i\partial x_j}\frac{d}{dt}\phi_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau) \\ &= \frac{\partial^2}{\partial x_i\partial x_j}f_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)) \\ &= \frac{\partial^2}{\partial x_i\partial x_j}f_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{x},t)) \\ &= \sum_{u,l\in\mathcal{I}}\frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_l}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)\frac{\partial\phi_l}{\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau) \\ &\quad + \sum_{u\in\mathcal{I}}\frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t))\frac{\partial^2\phi_u}{\partial x_i\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau),\end{aligned}$$

where the last term is the same as the one derived in (3.22). Combining these results and rearranging terms leads to

$$\begin{aligned}
&\frac{d}{dt}\frac{1}{2}\int_0^t \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{\phi}(\boldsymbol{x},\tau))\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)d\tau \\
&\quad= \sum_{u\in\mathcal{I}} \frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{x},t)) \underbrace{\frac{1}{2}\int_0^t \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{\phi}(\boldsymbol{x},\tau))\frac{\partial^2\phi_u}{\partial x_i\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)d\tau}_{v_u(\boldsymbol{x},t)} \\
&\qquad+ \sum_{u,l\in\mathcal{I}} \frac{\partial^2 f_{(k,s)}}{\partial x_u\partial x_l}(\boldsymbol{\phi}(\boldsymbol{x},t)) \\
&\qquad\quad\times \underbrace{\frac{1}{2}\int_0^t \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{\phi}(\boldsymbol{x},\tau))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)\frac{\partial\phi_l}{\partial x_j}(\boldsymbol{\phi}(\boldsymbol{x},\tau),t-\tau)d\tau}_{w_{u,l}},
\end{aligned}$$

which is the ODE describing $v_{(k,s)}(\boldsymbol{x},t)$. We obtain the integral form for $w_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{x},t)$ by application of the same steps.

□

Comparison of the Refinement Term $\boldsymbol{v}$ and the Quadratic Taylor Term

In Lemma 10 below, we bound the difference of the refinement term $\boldsymbol{v}$ and the quadratic term of the second order Taylor expansion appearing in the proof of Theorem 5. By defining $g_{(k,s)}(\boldsymbol{y},\tau) = \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{y})\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{y},t-\tau)$ we see that the entries of the refinement term $\boldsymbol{v}$ in integral form can be expressed as $v_{(k,s)}(\boldsymbol{x},t) = \frac{1}{2}\int_0^t g_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{x},\tau),\tau)d\tau$. Similarly, the time integral over the expectation of the quadratic term of the Taylor expansion is given by $\frac{1}{2}\int_0^t \mathbb{E}[g_{(k,s)}(\boldsymbol{X}(\tau),\tau)]d\tau$. The latter arises due to the comparison of generator approach used in the proof of Theorem 5 and the subsequent Taylor expansion of order two. The lemma shows that the difference of the two terms decreases quadratically with the system size N and allows, in combination with Lemma 8, to obtain the accuracy bounds for the refined mean field approximation.

Lemma 10. *Define $g_{(k,s)}(\boldsymbol{y},\tau) = \sum_{i,j\in\mathcal{I}} Q_{i,j}(\boldsymbol{y})\frac{\partial^2\phi_{(k,s)}}{\partial x_i\partial x_j}(\boldsymbol{y},t-\tau)$ with $\boldsymbol{\phi}$ being the solution to the ODE defined in Section 3.4.1 and $\boldsymbol{Q}$ as defined in Appendix 3.8.2. Then*

$$\frac{1}{2}\int_0^t \mathbb{E}[g_{(k,s)}(\boldsymbol{X}(\tau),\tau) - g_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{x},\tau),\tau)]d\tau = O(1/N^2).$$

Proof. We follow a similar proof concept as in Theorem 4. First, we define $h_\tau(\boldsymbol{y}) = g_{(k,s)}(\boldsymbol{y},\tau)$ and rewrite

$$\frac{1}{2}\int_0^t \mathbb{E}[g_{(k,s)}(\boldsymbol{X}(\tau),\tau) - g_{(k,s)}(\boldsymbol{\phi}(\boldsymbol{x},\tau),\tau)]d\tau = \frac{1}{2}\int_0^t \mathbb{E}[h_\tau(\boldsymbol{X}(\tau)) - h_\tau(\boldsymbol{\phi}(\boldsymbol{x},\tau))]d\tau. \tag{3.23}$$

By definition of $g_{(k,s)}$, h_τ is twice continuously differentiable. Second, using Lemma 6, we see that $\mathbb{E}[h_\tau(\boldsymbol{X}(\tau)) - h_\tau(\boldsymbol{\phi}(\boldsymbol{x},\tau))]$ is equal to

$$\int_0^\tau \mathbb{E}[\sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{X}(\nu),\boldsymbol{x}'}\big(h_\tau(\boldsymbol{\phi}(\boldsymbol{x}',\tau-\nu)) - h_\tau(\boldsymbol{\phi}(\boldsymbol{X}(\nu),\tau-\nu))\big) \tag{3.24}$$

$$-D_x(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{X}(\nu),\tau-\nu)f(\boldsymbol{X}(\nu))]d\nu. \tag{3.25}$$

We use a second order Taylor expansion to express $h_\tau(\boldsymbol{\phi}(\boldsymbol{x}',\tau-\nu)) = (h_\tau\circ\boldsymbol{\phi})(\boldsymbol{x}',\tau-\nu)$ around $\boldsymbol{X}(\nu)$. The constant and linear term of the expansion are $h_\tau(\boldsymbol{\phi}(\boldsymbol{X}(\nu),\tau-\nu))$ and $D_x(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{X}(\nu),\tau-\nu)\Delta X(\nu)$ respectively. By realizing that the sum $\sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{X}(\nu),\boldsymbol{x}'}D_x(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{X}(\nu),\tau-\nu)\Delta X(\nu)$ is equal to $D_x(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{X}(\nu),\tau-\nu)f(\boldsymbol{X}(\nu))$, it follows that equation (3.25) is equal to the remainder of the Taylor expansion

$$\int_0^\tau \mathbb{E}[\sum_{i,j\in\mathcal{I}}\sum_{\boldsymbol{x}'\in\mathcal{X}} K_{\boldsymbol{X}(\nu),\boldsymbol{x}'}\Delta X_i(\nu)\Delta X_j(\nu)$$
$$\times\int_0^1(1-\omega)\frac{\partial^2}{\partial x_i\partial x_j}(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{X}(\nu)+\omega\Delta\boldsymbol{X}(\nu),\tau-\nu)d\omega]d\nu.$$

Taking the supremum over all possible values of $\boldsymbol{X}(\tau)$ as well as $\boldsymbol{x}$ and using the definition of $\boldsymbol{Q}$, the above term is bounded by

$$\frac{1}{2}\int_0^\tau \sup_{\boldsymbol{y},\boldsymbol{z}\in\mathrm{Conv}(\mathcal{X})}\sum_{i,j\in\mathcal{I}}|Q_{i,j}(\boldsymbol{y})|\left|\frac{\partial^2}{\partial x_i\partial x_j}(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{z},\tau-\nu)\right|d\nu. \tag{3.26}$$

The rest of the proof is then essentially a careful analysis of the above sum. For that, we use again Lemma 7 but also need bounds on up to the second derivative of $\boldsymbol{Q}$ (This is needed because in the above expression the function h_τ is defined as a function of $\boldsymbol{Q}$). The latter makes the rest of the proof long and technical but the main ideas are essentially similar to the ones used in Lemma 7. The second derivative of $h\circ\boldsymbol{\phi}$ satisfies (for $i,j\in\mathcal{I}$):

$$\frac{\partial^2}{\partial x_i\partial x_j}(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{z},\tau-\nu) = \frac{\partial}{\partial x_j}\big(\sum_{u\in\mathcal{I}}\frac{\partial h_\tau}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{z},\tau-\nu)\big)$$
$$=\sum_{u\in\mathcal{I}}\frac{\partial h_\tau}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\frac{\partial^2\phi_u}{\partial x_i\partial x_j}(\boldsymbol{z},\tau-\nu) \tag{3.27}$$
$$+\sum_{u,r\in\mathcal{I}}\frac{\partial^2 h_\tau}{\partial x_u\partial x_r}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{z},\tau-\nu)\frac{\partial\phi_r}{\partial x_j}(\boldsymbol{z},\tau-\nu). \tag{3.28}$$

To bound the above term, we need to study $\frac{\partial h_\tau}{\partial x_u}(\boldsymbol{z}) = \frac{\partial g_{(k,s)}}{\partial x_u}(\boldsymbol{z},\tau)$ and $\frac{\partial^2 h_\tau}{\partial x_u \partial x_r}(\boldsymbol{z}) = \frac{\partial^2 g_{(k,s)}}{\partial x_u \partial x_r}(\boldsymbol{z},\tau)$. Applying the chain rule to the definition of h_τ shows that the first partial derivative of h_τ is

$$\begin{aligned}\frac{\partial h_\tau}{\partial x_u}(\boldsymbol{z}) &= \frac{\partial}{\partial x_u}\left(\sum_{q,l} Q_{q,l}(\boldsymbol{z})\frac{\partial \phi_{(k,s)}}{\partial x_q \partial x_l}(\boldsymbol{z}, t-\tau)\right)\\ &= \sum_{q,l\in\mathcal{I}} \frac{\partial Q_{q,l}}{\partial x_u}(\boldsymbol{z})\frac{\partial^2 \phi_{(k,s)}}{\partial x_q \partial x_l}(\boldsymbol{z}, t-\nu) + \sum_{q,l\in\mathcal{I}} Q_{q,l}(\boldsymbol{z})\frac{\partial^3 \phi_{(k,s)}}{\partial x_q \partial x_l \partial x_u}(\boldsymbol{z}, t-\nu).\end{aligned}$$

Similarly, the second partial derivative is

$$\begin{aligned}\frac{\partial^2 h_\tau}{\partial x_r \partial x_u}(\boldsymbol{z}) &= \frac{\partial}{\partial x_r}\left(\sum_{q,l\in\mathcal{I}} \frac{\partial Q_{q,l}}{\partial x_u}(\boldsymbol{z})\frac{\partial^2 \phi_{(k,s)}}{\partial x_q \partial x_l}(\boldsymbol{z}, t-\nu) + \sum_{q,l\in\mathcal{I}} Q_{q,l}(\boldsymbol{z})\frac{\partial^3 \phi_{(k,s)}}{\partial x_q \partial x_l \partial x_u}(\boldsymbol{z}, t-\nu)\right)\\ &= \sum_{q,l\in\mathcal{I}} \frac{\partial^2 Q_{q,l}}{\partial x_u \partial x_r}(\boldsymbol{z})\frac{\partial^2 \phi_{(k,s)}}{\partial x_q \partial x_l}(\boldsymbol{z}, t-\nu) + \frac{\partial Q_{q,l}}{\partial x_u}(\boldsymbol{z})\frac{\partial^3 \phi_{(k,s)}}{\partial x_q \partial x_l \partial x_r}(\boldsymbol{z}, t-\nu)\\ &\quad + \frac{\partial Q_{q,l}}{\partial x_r}(\boldsymbol{z})\frac{\partial^3 \phi_{(k,s)}}{\partial x_q \partial x_l \partial x_u}(\boldsymbol{z}, t-\nu) + Q_{q,l}(\boldsymbol{z})\frac{\partial^4 \phi_{(k,s)}}{\partial x_q \partial x_l \partial x_u \partial x_r}(\boldsymbol{z}, t-\nu).\end{aligned}$$

What remains is to bound the sums appearing in the above derivatives. We use the notations $c_{k_1}^k$ and c_{k_1,k_2,k_3}, as in the proof of Lemma 8. From the representation of $\boldsymbol{Q}$ given in Appendix 3.8.2, it can be seen that $Q_{(k_1,s_1),(k_2,s_2)}(z) = O(c_{k_2}^{k_1})$, $\frac{\partial Q_{(k_1,s_1),(k_2,s_2)}}{\partial x_{(k_3,s_3)}} = O(c_{k_1,k_2,k_3})$ and

$$\frac{\partial^2 Q_{(k_1,s_1),(k_2,s_2)}}{\partial x_{(k_3,s_3)} \partial x_{(k_4,s_4)}} = \begin{cases} O(1/N) & \text{if } (k_1,k_2) = (k_3,k_4) \text{ or } (k_4,k_3),\\ O(1/N^2) & \text{if } k_1 = k_3, k_4 \text{ or } k_2 = k_3, k_4,\\ O(1/N^3) & \text{otherwise.}\end{cases}$$

Lemma 7 gives bounds for the partial derivatives of $\boldsymbol{\phi}$. This enables us to develop an upper bound for $|\frac{\partial h_\tau}{\partial x_u}(z)|$,

$$\begin{aligned}\left|\frac{\partial h_\tau}{\partial x_{(k',s')}}(z)\right| &\le \sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} \left|\frac{\partial Q_{(k_1,s_1),(k_2,s_2)}}{\partial x_{(k',s')}}(z)\right| \left|\frac{\partial^2 \phi_{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)}}(z, t-\nu)\right|\\ &\quad + \sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} \left|Q_{(k_1,s_1),(k_2,s_2)}(z)\right| \left|\frac{\partial^3 \phi_{(k,s)}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)} \partial x_{(k',s')}}(z, t-\nu)\right|\\ &= \sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} O\big(c_{k_1,k_2,k'}\big) O\big(\frac{1}{N}(c_{k_1}^k + c_{k_2}^k)\big)\\ &\quad + \sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} O\big(c_{k_2}^{k_1}\big) O\big(\frac{1}{N^2}(c_{k_1}^k + c_{k_2}^k + c_{k_3}^k)\big) = O(\frac{1}{N} c_{k'}^k).\end{aligned}$$

With the above observations we bound the first sum of (3.28) by

$$\begin{aligned}
&\sum_{u\in\mathcal{I}}\left|\frac{\partial h_\tau}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\right|\left|\frac{\partial^2\phi_u}{\partial x_i\partial x_j}(\boldsymbol{z},\tau-\nu)\right| \\
&= \sum_{(k',s')\in\mathcal{I}}\left|\frac{\partial g_{(k,s)}}{\partial x_{(k',s')}}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu),\tau)\right|\left|\frac{\partial^2\phi_{(k',s')}}{\partial x_{(k_1,s_1)}\partial x_{(k_2,s_2)}}(\boldsymbol{z},\tau-\nu)\right| \\
&= \sum_{(k',s')\in\mathcal{I}} O(\frac{1}{N}c^k_{k'})O(\frac{1}{N}(c^{k'}_{k_1}+c^{k'}_{k_2})) = O(\frac{1}{N^2}(c^k_{k_1}+c^k_{k_2})).
\end{aligned}$$

For the second partial derivatives of $h_\tau(\boldsymbol{z}) = g_{(k,s)}(\boldsymbol{z}, t-\tau)$, we note that all sums which appear in the explicit form of the partial derivative are bounded by $O(\frac{1}{N^2}(c^k_{k'}+c^k_{\hat{k}}))$. Using the bounds for $\boldsymbol{Q}$ and $\boldsymbol{\phi}$ and their respective partial derivatives we see that

$$\begin{aligned}
&\sum_{(k_1,s_1),(k_2,s_2)}\left|Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{z})\right|\left|\frac{\partial^4\phi_{(k,s)}}{\partial x_{(k_1,s_1)}\partial x_{(k_2,s_2)}\partial x_{(k',s')}\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{z},t-\nu)\right| \\
&= \sum_{(k_1,s_1),(k_2,s_2)} O(c^{k_1}_{k_2})O(\frac{1}{N^3}(c^k_{k_1}+c^k_{k_2}+c^k_{k'}+c^k_{\hat{k}})) = O(\frac{1}{N^2}(c^k_{k'}+c^k_{\hat{k}})),
\end{aligned}$$

and that

$$\begin{aligned}
&\sum_{(k_1,s_1),(k_2,s_2)}\left|\frac{\partial Q_{(k_1,s_1),(k_2,s_2)}}{\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{z})\right|\left|\frac{\partial^3\phi_{(k,s)}}{\partial x_{(k_1,s_1)}\partial x_{(k_2,s_2)}\partial x_{(k',s')}}(\boldsymbol{z},t-\nu)\right| \\
&\qquad = \sum_{(k_1,s_1),(k_2,s_2)} O(c_{k_1,k_2,\hat{k}})O(\frac{1}{N^2}(c^k_{k_1}+c^k_{k_2}+c^k_{k'})) = O(\frac{1}{N^2}(c^k_{k'}+c^k_{\hat{k}})),
\end{aligned}$$

as well as

$$\sum_{(k_1,s_1),(k_2,s_2)}\left|\frac{\partial^2 Q_{(k_1,s_1),(k_2,s_2)}}{\partial x_{(\hat{k},\hat{s})}\partial x_{(k',s')}}(z)\right|\left|\frac{\partial^2\phi_{(k,s)}}{\partial x_{(k_1,s_1)}\partial x_{(k_2,s_2)}}(\boldsymbol{z},t-\nu)\right| = O(\frac{1}{N^2}(c^k_{k'}+c^k_{\hat{k}})).$$

The last bound follows with careful case-by-case analysis for the second derivative of $\frac{\partial^2 Q_{(k_1,s_1),(k_2,s_2)}}{\partial x_{(\hat{k},\hat{s})}\partial x_{(k',s')}}$. As a direct consequence $\left|\frac{\partial^2 h_\tau}{\partial x_{(k',s')}\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\right| = O(\frac{1}{N^2}(c^k_{k'}+c^k_{\hat{k}}))$. This enables us to establish a bound for the second sum of (3.28),

$$\begin{aligned}
&\sum_{u,r\in\mathcal{I}}\left|\frac{\partial^2 h_\tau}{\partial x_u\partial x_r}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\right|\left|\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{z},\tau-\nu)\right|\left|\frac{\partial\phi_r}{\partial x_j}(\boldsymbol{z},\tau-\nu)\right| \\
&= \sum_{(k',s'),(\hat{k},\hat{s})\in\mathcal{I}}\left|\frac{\partial^2 g_{(k,s)}}{\partial x_{(k',s')}\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu),\tau)\right|\left|\frac{\partial\phi_{(k',s')}}{\partial x_{(k_1,s_1)}}(\boldsymbol{z},\tau-\nu)\right|\left|\frac{\partial\phi_{(\hat{k},\hat{s})}}{\partial x_{(k_2,s_2)}}(\boldsymbol{z},\tau-\nu)\right| \\
&= \sum_{(k',s'),(\hat{k},\hat{s})\in\mathcal{I}} O(\frac{1}{N^2}(c^k_{k'}+c^k_{\hat{k}}))O(c^{k'}_{k_1})O(c^{\hat{k}}_{k_2}) = O(\frac{1}{N^2}(c^k_{k_1}+c^k_{k_2})).
\end{aligned}$$

For the last part of the proof, we use the obtained results to bound $\sum_{i,j\in\mathcal{I}}|Q_{i,j}(\boldsymbol{y})|\left|\frac{\partial^2}{\partial x_i\partial x_j}(h_\tau\circ\boldsymbol{\phi})(\boldsymbol{z},\tau-\nu)\right|$. By equation (3.28) we see that (3.26) is equal to

$$\begin{aligned}
\frac{1}{2}\int_0^\tau \sup_{\boldsymbol{y},\boldsymbol{z}\in\mathrm{Conv}(\mathcal{X})}\sum_{i,j\in\mathcal{I}}|Q_{i,j}(\boldsymbol{y})|\Bigg|&\sum_{u\in\mathcal{I}}\frac{\partial h_\tau}{\partial x_u}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\frac{\partial^2\phi_u}{\partial x_i\partial x_j}(\boldsymbol{z},\tau-\nu) \\
&+\sum_{u,r\in\mathcal{I}}\frac{\partial^2 h_\tau}{\partial x_u\partial x_r}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu))\frac{\partial\phi_u}{\partial x_i}(\boldsymbol{z},\tau-\nu)\frac{\partial\phi_r}{\partial x_j}(\boldsymbol{z},\tau-\nu)\Bigg|d\nu.
\end{aligned}$$

Indeed, we bound the supremum by the two following terms

$$\sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}}\left|Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{y})\right| \tag{3.29}$$

$$\times\sum_{(k',s')\in\mathcal{I}}\left|\frac{\partial g_{(k,s)}}{\partial x_{(k',s')}}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu),\tau)\right|\left|\frac{\partial^2\phi_{(k',s')}}{\partial x_{(k_1,s_1)}\partial x_{(k_2,s_2)}}(\boldsymbol{z},\tau-\nu)\right|$$

$$= \sum_{(k_1,s_1),(k_2,s_2)} O(c^{k_1}_{k_2})O(\frac{1}{N^2}(c^k_{k_1}+c^k_{k_2})) = O(1/N^2) \tag{3.30}$$

and

$$\sum_{(k_1,s_1),(k_2,s_2)\in\mathcal{I}} \left|Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{y})\right| \sum_{(k',s'),(\hat{k},\hat{s})\in\mathcal{I}} \left|\frac{\partial^2 g_{(k,s)}}{\partial x_{(k',s')}\partial x_{(\hat{k},\hat{s})}}(\boldsymbol{\phi}(\boldsymbol{z},\tau-\nu),\tau)\right| \times \left|\frac{\partial \phi_{(k',s')}}{\partial x_{(k_1,s_1)}}(\boldsymbol{z},\tau-\nu)\right| \left|\frac{\partial \phi_{(\hat{k},\hat{s})}}{\partial x_{(k_2,s_2)}}(\boldsymbol{z},\tau-\nu)\right|$$

$$= \sum_{(k_1,s_1),(k_2,s_2)} O(c_{k_2}^{k_1}) O\left(\frac{1}{N^2}(c_{k_1}^{k} + c_{k_2}^{k})\right) = O(1/N^2). \tag{3.31}$$

The bounds (3.30) and (3.31) show that (3.26) is of order $O(1/N^2)$, where the hidden constant depends on $\tau, \bar{r}, |\mathcal{S}|$, from which the claim of the Lemma follows. □

4 RMF Tool - A Numerical Tool Box

This chapter presents the RMF Tool, a numerical toolbox that takes the description of a stochastic population model and numerically computes its mean field approximations and refinement.

This chapter is based on our publication
S. Allmeier and N. Gast. "Rmf Tool - A Library to Compute (Refined) Mean Field Approximation(s)". *ACM SIGMETRICS Performance Evaluation Review* 4, 2, 2022, pp. 35–40. ISSN: 0163-5999. DOI: 10.1145/3543146.3543156.

Contents

4.1 Introduction

Mean field approximation is widely applied to analyze the behavior of large stochastic systems. It applies to systems composed of N interacting objects. The idea of the approximation is to consider that objects within the system evolve independently. This transforms the study of a multi-dimensional stochastic process into much smaller stochastic processes that are weakly coupled. Under mild conditions, the mean field approximation is described by a finite set of deterministic ordinary differential

equations (ODEs). As such, it can be simulated at low computational cost. Mean field approximation finds widespread use in fields such as epidemic spreading [43, 90], load balancing strategies [87, 91], the study of cache replacement strategies [60] or SSDs [108].

Classical models to which mean field approximation applies are the class of density dependent population processes (DDPPs, [77]), whose definition is recalled in Section 4.2 – epidemic spreading or load balancing models are typical examples of DDPPs. If X is a density dependent population process in d dimensions, its mean field approximation is the solution of a system of non-linear ODEs $\dot{x} = f(x)$ where $f : \mathbb{R}^d \to \mathbb{R}^d$ is called the drift of the system. Computing the mean field approximation can be easily automated, as the drift f can be expressed easily from the model's definition. Our tool incorporates this but, more importantly, allows going further.

Building on mean field approximation, the authors of [54, 56] introduce the notion of *refined* mean field approximation. This approximation consists in adding an expansion term to the original approximation. Denoting by x the value of the mean field approximation, it is shown in [55] that there exists a deterministic quantity $v(t)$ such that:

$$\mathbb{E}[X(t)] = \underbrace{x(t) + \frac{1}{N}v(t)}_{\text{refined m.f. approx.}} + O(\frac{1}{N^2}).$$

The quantity $v(t)$ is the solution of a time-inhomogeneous linear ODE. As shown in the aforementioned papers, the construction of this set of ODEs is direct from the model description but involves computing the derivatives of the drift, which can be cumbersome.

The purpose of *rmf_tool* – the refined mean field tool – is to make mean field and refined mean field approximation easily computable. Our tool is composed of a Python library. The tool takes as input a description of the system, which can be either a density dependent population process or a heterogeneous population model, and can be used to compute the mean field and refined mean field approximations numerically. The tool relies on standard libraries (like `numpy` and `scipy`) to construct and solve the corresponding ODEs. The tool is provided with a series of examples to demonstrate its expressiveness and the accuracy of the various approximations.

Related tools There exist a large number of tools that provide methods to construct and simulate stochastic population models. Yet, to the best of our knowledge, the only tool that provides a way to analyze size expansion methods (which are essentially equivalent to our refined mean field approximation) is the iNA software of [104]. The iNA is a complete simulation toolbox (that includes its own graphical interface). Compared to this software, we use a more lightweight approach by providing a pure python library that can be easily integrated.

Roadmap The chapter is centered around the tool. We first describe the set of models to which the tool applies in Section 4.2, along with examples on how they can be defined within the tool. We then describe what are the mean field and refined mean field approximation, and how one can use the tool to compute them in Section 4.3. We detail some technical challenges in Section 4.4 and conclude in Section 4.5.

Reproducibility Our tool is provided as an open-source software at https://github.com/Ngast/rmf_tool. The code to reproduce this chapater along with all figures is available at https://gitlab.inria.fr/gast/toolpaper_rmf.

4.2 Models

The tool that we develop accepts three kinds of models: homogeneous population processes (HomPP), density dependent population processes (DDPPs) and heterogeneous population models (HetPP). First, we describe the notion of the HomPP of which DDPP and HetPP are generalizations.

4.2.1 Homogeneous population process

Population Processes are widely used to describe the evolution of a number of interacting objects (or individuals). A homogeneous population model consists of N interacting objects that each evolves in a finite state space $\{1 \dots d\}$. All objects have similar transition rates which are a combination of unilateral and pairwise interactions, i.e. objects can change their state with or without interacting with one other member of the population. Let $X_s(t)$ be the fraction of objects that are in state s at time t. We assume that $\boldsymbol{X} = (X_1 \dots X_d)$ is a continuous time Markov chain whose transitions are such that for all state s, s', $\tilde{s}$, $\tilde{s'}$:

(Uni.) An object in state s moves to state s' at rate $a_{s,s'}$.

(Pair.) A pair of objects in state $(s, \tilde{s})$ moves to state s', $\tilde{s}'$ at rate $b_{s,\tilde{s},s',\tilde{s}'}/N$.

Note that for pairwise interactions, the rate is scaled by $1/N$ as the number of pairs of objects is N times larger than the number of objects.

Example: the SIS model One of the simplest examples of population process is the epidemic model called the SIS model. In an SIS model, each object can be in one of the two states S (susceptible) or I (infected). Susceptible objects can become infected from an external source (unilateral transition) or when meeting an infected individual (pairwise transition). An infected individual can recover and become susceptible again (unilateral transition). We assume that an individual becomes infected at rate α by an external source, and recovers at rate β. Moreover, assume that the rate at which two individuals meet is γ/N and that when a susceptible meets an infected individual, the susceptible gets infected.

With our tool, we define a class called HomPP for which we specify the transition rates and an initial state. For the SIS model above, with $\alpha, \beta, \gamma = 1$, we write:

```
import rmf_tool as rmf

model = rmf.HomPP()

d, S, I = 2, 0, 1

A, B = np.zeros((d, d)), Np.zeros((d, d, d, d))
A[S, I] = 1                 # \alpha
A[I, S] = 1                 # \beta
```

```
B[S, I, I, I] = 1          # \gamma
model.add_rate_tensors(A, B)
```

The specified model can be used to simulate stochastic trajectories of the underlying process for various population sizes. It can also be used to compute the mean field approximation and the refinements (see Section 4.3). For instance, if one wants to simulate a trajectory for a population of size $N = 1000$, where all individuals are susceptible in the initial state, one would write:

```
model.set_initial_state([1,0])
t, X = model.simulate(N=1000, time=2)
```

STATE REPRESENTATION Recall that $X_s(t)$ is the fraction of objects in state s at time t. The transitions of such a model can be expressed[1] as:

(Uni.) When an object moves from s to s', this changes $\boldsymbol{X}$ into $\boldsymbol{X} + \frac{1}{N}(\boldsymbol{e}_{s'} - \boldsymbol{e}_s)$. As $nX_s(t)$ is the number of objects in state s, this transition occurs at rate $na_{s,s'}X_s(t)$.

(Pair.) When a pair moves from $(s, \tilde{s})$ to $(s', \tilde{s}')$, this changes $\boldsymbol{X}$ into $\boldsymbol{X} + \frac{1}{N}(\boldsymbol{e}_{s'} + \boldsymbol{e}_{\tilde{s}'} - \boldsymbol{e}_{\tilde{s}} - \boldsymbol{e}_s)$. This transition occurs at rate $nb_{s,\tilde{s},s',\tilde{s}'}X_s(t)X_{\tilde{s}(t)}$.

Written in a compact way, those transitions are:

$$\boldsymbol{x} \to \boldsymbol{x} + \frac{1}{N}(\boldsymbol{e}_{s'} - \boldsymbol{e}_s) \qquad \text{at rate} \qquad na_{s,s'}x_s \tag{4.1}$$

$$\boldsymbol{x} \to \boldsymbol{x} + \frac{1}{N}(\boldsymbol{e}_{s'} + \boldsymbol{e}_{\tilde{s}'} - \boldsymbol{e}_{\tilde{s}} - \boldsymbol{e}_s) \qquad \text{at rate} \qquad nb_{s,\tilde{s},s',\tilde{s}'}x_s x_{\tilde{s}}. \tag{4.2}$$

4.2.2 DENSITY DEPENDENT POPULATIONS PROCESS

The class of homogeneous population model that we define is a subclass of density dependent population processes (DDPPs) that are introduced by Kurtz in the 70s [77]. For a given N, a DDPP defines a stochastic process $\boldsymbol{X} \in \mathbb{R}^d$. The transitions of the process are specified by a finite set of vectors $\mathcal{L} \subset \mathbb{R}^d$, and a set of corresponding rate functions $\beta_\ell : \mathbb{R}^d \to \mathbb{R}^+$ for all $\ell \in \mathcal{L}$. The process $\boldsymbol{X}$ jumps from $\boldsymbol{x}$ to $\boldsymbol{x} + \ell/N$ at rate $N\beta_\ell(\boldsymbol{x})$.

It should be clear from Equation (4.1)-(4.2) that HomPP is a special case of DDPP. DDPPs generalize HomPP since they allow to choose arbitrary transition rates as opposed to combinations of unilateral and pairwise transition. In the case where $nX_s(t)$ denotes the number of individuals in state s at time t, the vector $\ell \in \mathcal{L}$ indicates how many individuals are created or destroyed by a transition. For instance, if $d = 3$, $\ell = (1, -1, 0)$ corresponds to having one additional individual in state 1 and one less in state 2 (this occurs typically when one individual moves from state 2 to state 1), $\ell = (0, 0, 2)$ corresponds to the creation of two additional individuals in state 3.

[1] The notation $\boldsymbol{e}_s \in \{0, 1\}^d$ correspond to a vector of size d whose s entry is equal to 1, all others being 0.

THE SIS MODEL AS A DDPP To illustrate the relation between DDPPs and HomPP, consider the SIS model defined in the previous section and recall that $(X_S(t), X_I(t))$ is the fraction of susceptible and of infected individuals. The transitions $\ell \in \mathcal{L}$ and their corresponding rates β_ℓ are:

Event	Transition ℓ	Rate $\beta_\ell(\boldsymbol{x})$
infection from ext. source	(-1,1)	$\alpha \boldsymbol{x}_S$
recovery	(1,-1)	$\beta \boldsymbol{x}_I$
infection from a meeting	(-1,1)	$\gamma \boldsymbol{x}_S \boldsymbol{x}_I$

Within our tool, we define a class called `DDPP` that can be used to define DDPPs directly from their mathematical definition. For the above SIS example, we would write:

```
import rmf_tool as rmf
model = rmf.DDPP()
alpha, beta, gamma = 1,1,1
model.add_transition([-1,1], lambda x: alpha*x[0])
model.add_transition([1,-1], lambda x: beta*x[1])
model.add_transition([-1,1], lambda x: gamma*x[0]*x[1])
```

As for the HomPP, the model can then be used to simulate the stochastic process, to compute the mean field approximation and the refinements. The syntax is identical. If one wants to run a simulation with a population of $N = 1000$ where at the beginning all individuals are in the first state (susceptible), one would write:

```
model.set_initial_state([1,0])
t, X = model.simulate(N=1000, time=2)
```

4.2.3 HETEROGENEOUS POPULATION PROCESS

In [5], the authors extend the notion of the HomPPs to deal with populations of heterogeneous objects. As before, the heterogeneous population model consists of N interacting objects which each evolve in a finite state space $\{1 \dots d\}$. Each object has a specific transition rate which is a combination of unilateral or pairwise interactions. In contrast to the HomPP, transition rates are object dependent:

- The object k moves from state s to state s' at rate $a_{k,s,s'}$.
- The pair (k, k') moves from states $(s, \tilde{s})$ to states $(s', \tilde{s}')$ at rate $b_{k,k',s,\tilde{s},s',\tilde{s}'}/N$.

Note that the difference between a homogeneous population process and a heterogeneous population process is that the rate tensors a and b depend on the object id k. As a result, the process $\boldsymbol{X} = (X_1 \dots X_d)$ where $X_s(t)$ is the fraction of objects in state s is not a Markov process. Let the stochastic process $\boldsymbol{Z} \in \{0, 1\}^{N \times d}$ describe the evolution of the population where $Z_{k,s} = 1$ indi-

cates that object k is in state s and $Z_{k,s} = 0$ if it is not. The process $\boldsymbol{Z}$ is a Markov process whose transitions are:

$$\boldsymbol{z} \mapsto \boldsymbol{z} - \boldsymbol{e}_{k,s} + \boldsymbol{e}_{k,s'} \qquad \text{at rate} \qquad a_{k,s,s'} z_{k,s},$$

$$\boldsymbol{z} \mapsto \boldsymbol{z} - \boldsymbol{e}_{k,s} + \boldsymbol{e}_{k,s'} - \boldsymbol{e}_{\tilde{k},\tilde{s}} + \boldsymbol{e}_{\tilde{k},\tilde{s}'} \qquad \text{at rate} \qquad \frac{1}{N} b_{k,\tilde{k},s,\tilde{s},s',\tilde{s}'} z_{k,s} z_{\tilde{k},\tilde{s}}.$$

These transitions generalize (4.1)-(4.2).

Example: Heterogeneous SIS model To set up a heterogeneous version of the previous SIS model we use the HetPP class of the toolbox. In contrast to the HomPP and DDPP class, the model can not be defined independent of the system size, *i.e.*, N and d have to be defined to initialize the model. For instance, to set up a SIS model where objects are almost identical but some recover slower than others, we can use the code:

```
import rmf_tool.src.heterogeneous_rmf_tool as hrmf
model = hrmf.HetPP()
N, d = 20, 2
S, I = 0, 1
A, B = np.zeros((N, d, d)), Np.zeros((N, N, d, d, d, d))
A[:, S, I] = np.ones((N))
A[:, I, S] = np.random.rand(N)   # Hetero. recovery rates
B[:, :, S, I, I, I] = (1/N) * np.ones((N, N))
model.add_rate_tensors(A, B)
```

Here, the tensor A and B specify the transition rates where A[k,s,s'] $= a_{k,s,s'}$ and B[k,$\tilde{k}$,s,$\tilde{s}$,s',$\tilde{s}$'] $= \frac{1}{N} b_{k,\tilde{k},s,\tilde{s},s',\tilde{s}'}$. The corresponding transition vectors of the model are derived from the non zero rates of the tensors. The methods of the HetPP class are coherent to the HomPP and DDPP class.

4.3 Mean field approximations and refinements

4.3.1 Mean field approximation (homogeneous)

For a given DDPP, and a given state $\boldsymbol{x} \in \mathbb{R}^d$, we define the drift in state $\boldsymbol{x}$ as

$$f(\boldsymbol{x}) = \sum_{\ell \in \mathcal{L}} \ell \beta_\ell(\boldsymbol{x}).$$

The drift corresponds to the average variation of the model, as it is the sum of state changes ℓ weighted by the rate at which these changes occur.

For a given initial condition $\boldsymbol{x}(0)$, the mean field approximation of a DDPP is the solution of the ODE:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}).$$

The same holds true for any HomPP since the class of DDPP is essentially a scaled generalization of the former. Thus, all methods which are available for DDPPs are available for HomPP as well. Within our tool, the mean field approximation can be easily computed with:

```
t, X = model.ode(time=2)
```

It is known from [77] that under very general conditions (essentially that f is Lipschitz-continuous), the stochastic trajectories of $\boldsymbol{X}$ converge to the mean field approximation $\boldsymbol{x}$ as the scaling parameter N goes to infinity. We illustrate the accuracy of the mean field approximation in Figure 4.1, where we compare two stochastic trajectories of the system for populations of $N = 100$ and $N = 1000$ individuals, with the mean field approximation.

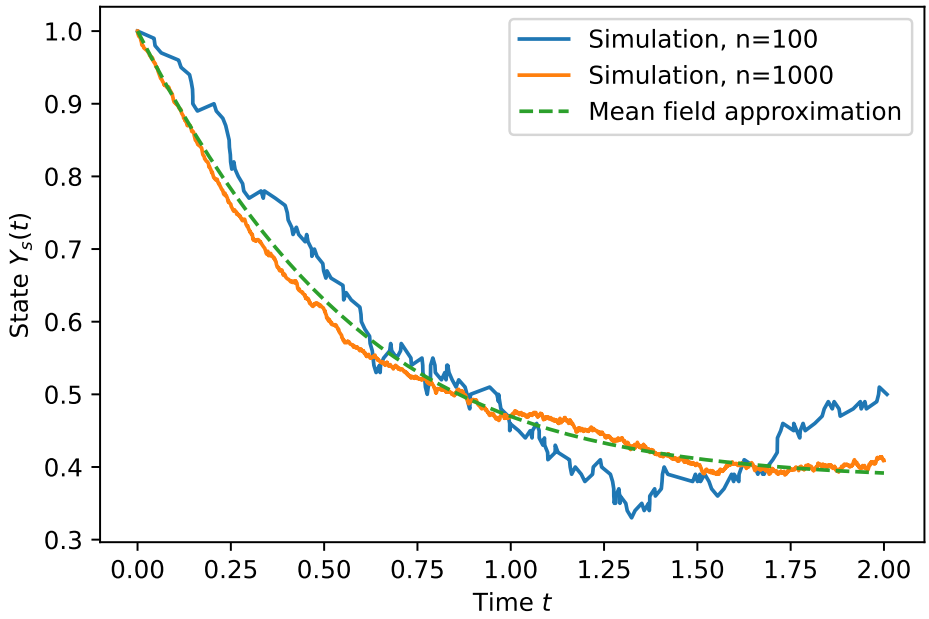


Figure 4.1: Example: Simulation of the SIS (DDPP model)

4.3.2 The refined mean field approximation

It is shown in [54, 56] that when the drift of the DDPP is twice differentiable, there exists a time varying vector $\boldsymbol{v}$ and a time varying matrix $\boldsymbol{w}$ such that:

$$\mathbb{E}[\boldsymbol{X}(t)] = \boldsymbol{x}(t) + \frac{1}{N}\boldsymbol{v}(t) + O(\frac{1}{N^2});$$
$$\mathrm{Var}[\boldsymbol{X}(t)] = \frac{1}{N}\boldsymbol{w}(t) + O(\frac{1}{N^2}),$$

where $\mathrm{Var}[\boldsymbol{X}(t)]$ is the covariance matrix of the stochastic process $\boldsymbol{X}$.

The above equation holds for any finite time. It is shown in [54] that, for the transient regime, $\boldsymbol{v}$ and $\boldsymbol{w}$ satisfy a time-inhomogeneous linear ODEs. If there exists a point $\boldsymbol{x}(\infty)$ such that for all

initial condition $\boldsymbol{x}(0) \in \mathbb{R}^d$, the solution of the ODE converges to $\boldsymbol{x}(\infty)$ exponentially fast, then this equation holds uniformly in time and in particular is also true for the steady-state $t = +\infty$. In the latter case, the following linear equation (that is called a Lyapunov equation) is satisfied:

$$\boldsymbol{w}J + J^T\boldsymbol{w} + \boldsymbol{q} = 0, \tag{4.3}$$

where J is the Jacobian of the drift f evaluated at $\boldsymbol{x}(\infty)$ and $\boldsymbol{q} = \sum_\ell \ell \otimes \ell\beta_\ell(\boldsymbol{x}(\infty))$. The vector $\boldsymbol{v}(\infty) = J^{-1}(D \cdot \boldsymbol{w})$, where D is the second derivative of f evaluated at $\boldsymbol{x}(\infty)$ and $\cdot$ denotes a tensor product: $(D \cdot \boldsymbol{w})_i = \sum_{jk} D_{i,jk} w_{jk}$ and $D_{i,jk} = (\partial^2 f / \partial x_j \partial x_k)$ evaluated in $x(\infty)$.

This means that they can be easily solved numerically. The tool provides methods to automatically compute these constants for the transient or the steady-state regime. These functions rely on `scipy`'s functions: for the transient regime it uses the `solve_ivp` from `scipy` and for the steady-state the function `solve_continuous_lyapunov`. An example of the tool is:

```
t, x, v, _ =\
       model.meanFieldExpansionTransient(order=1,time=2)
x_inf, v_inf,_ =\
       model.meanFieldExpansioNSteadyState(order=1)
x_simu, _ = ddpp.steady_state_simulation(N=n, time=20000)
```

where the last line estimates $\mathbb{E}[\boldsymbol{X}(\infty)]$ by simulating a trajectory of 20000 events and computes the average over the end of the trajectory.

This result is illustrated in Table 4.1, where we compare the mean field approximation, the refined mean field approximation and an estimation of the steady-state probability $\mathbb{E}[X_s(\infty)]$ computed by simulation. We observe that if the mean field approximation is already very accurate, its refined version is close to being exact.

N	M-f $\boldsymbol{x}(\infty)$	Refined $\boldsymbol{x}(\infty) + \frac{1}{N}\boldsymbol{x}(\infty)$	Simulation
10	0.382	0.394	0.394 ± 0.004
20	0.382	0.388	0.389 ± 0.003
30	0.382	0.386	0.386 ± 0.002

Table 4.1: SIS model: Illustration of the accuracy of the mean field and refined mean field approximations for steady-state.

Note that the tool also allows to compute the second order refinement term as defined in [54]. This can be done by changing the `order=1` into `order=2` in the code. The time to compute this approximation is much larger than the time to compute the refined mean field approximation (that corresponds to a first order expansion).

4.3.3 Heterogeneous mean field approximation and refinements

The heterogeneous mean field approximation and its refinement differs from the homogeneous case in the sense that transitions are dependent on the state of single objects. For the stochastic process this is taking into account by considering an object dependent representation. The intuition of the mean field approximation is as before, for the drift we consider the sum over all transitions weighted

by their transition rate. Let the drift in state z be denoted by $f^{het}(z)$, then, the mean field approximation is again the solution to the ode having f^{het} as drift with initial condition $z(0)$. If both, $a_{k,s,s'}$ and $b_{k,\tilde{k},s,\tilde{s},s',\tilde{s}'}$ are uniformly bounded, it holds, as shown in [5] that the adapted mean field and refined mean field approximation capture the probability of the objects to be in a state with an accuracy of $O(1/N)$ and $O(1/N^2)$, i.e.

$$\begin{aligned}\mathbb{E}[Z_{k,s}(t)] &= \mathbb{P}(Z_{k,s}(t)=1) = z_{k,s}(t) + O(1/N), \\ \mathbb{E}[Z_{k,s}(t)] &= \mathbb{P}(Z_{k,s}(t)=1) = z_{k,s}(t) + v_{k,s}(t) + O(1/N^2).\end{aligned} \tag{4.4}$$

The term $v_{k,s}(t)$ refers to the adapted refinement term whose precise definition can be found in [5, Appendix B].

Simulations of stochastic trajectories, mean field and the refinement methods can be calculated by calling the same functions as for the homogeneous case. Note that second order refinement methods are note available for the current version since they are computationally too expensive.

Due to the setup of the heterogeneous population process, single simulation trajectories are not close to the mean field approximation but close to the sample mean of the stochastic system, that is, (4.4) holds but $Z_{k,s}(t)$ does not converge to $z_{k,s}(t)$ as N goes to infinity (contrary to what appends to the DDPP case for which one can show [77] that $X_s(t)$ converges in probability to its mean field approximation $x_s(t)$, which is what is observed in Figure 4.1). Hence, to study the accuracy of the mean field and refined mean field approximation in the heterogeneous context, we provide the additional methods `sampleMean`, `sampleMeanVariance` with which the sample mean and sample variance can be calculated. To calculate an approximated mean with 100 samples, we set the initial state to have only susceptible objects and write:

```
model.set_initial_state(np.ones((N,d))*np.array([1,0]))
t_mean, mean, var =\
      model.sampleMeanVariance(time=2, samples=100)
```

In order to compare the results to a one of the homogeneous models one should consider the sum $Y_s(t) = \frac{1}{N}\sum_{k=1}^{N} Z_{(k,s)}(t)$, which is a density representation of the heterogeneous population process. It can be shown that $Y_s(t)$ converges in probability to its mean field approximation $y_s(t)$, as the number of objects grows.

4.4 Implementation challenges

Most of the toolbox functionality is a direct implementation of the equations defined in [54, 59], with the use of functions from `numpy` or `scipy` to integrate differential equations or solve linear equations. Yet, there are some implementation challenges among which we list two here: how to automatically compute the drift's derivatives (Section 4.4.1), and how to deal with model that do not satisfy the exact assumptions of [59] needed for the steady-state (Section 4.4.2).

4.4.1 Automatic differentiation

To compute the refined mean field approximation, one needs to compute the first and second derivatives of the drift function f. We implement three different methods. The first is to use a finite dif-

ference method. $\partial f_i(\boldsymbol{x})/\partial x_j \approx (f_i(\boldsymbol{x} + \varepsilon \boldsymbol{e}_j) - f_i(\boldsymbol{x}))/\varepsilon$. This is the most robust method but is relatively slow and has a limited precision due to the choice of ε. The second method that we implement is to use the package `sympy` that allows for symbolic computation and can be used to compute derivatives. The third method is a method based on `autograd` from `jax` that uses automatic differentiation. These two methods are both faster and more accurate than finite difference methods. Yet, they cannot differentiate all functions. For instance, if the drift involves a sinus function and if the DDPP model is defined using the `numpy.sin` function, then the `sympy` will not be able to differentiate this function as it does not understand the `numpy` function. Here, `autograd` will work.

4.4.2 Dimension reduction

In order for the equation (4.3) to have a unique solution, the assumption used in [59] is that all solutions of the mean field ODE $\dot{\boldsymbol{x}} = f(\boldsymbol{x})$ converge to the same fixed point $\boldsymbol{x}(\infty)$, regardless of the initial condition $\boldsymbol{x}(0) \in \mathbb{R}^d$. Yet, in practice, many models are naturally described as d-dimensional DDPP but evolve in a smaller dimensional space $\mathcal{X} \subset \mathbb{R}^d$. This is for instance the case for the SIS model of Section 4.2 that evolves in a space of dimension 1 because $x_S + x_I = 1$. It further implies that the Lyapunov equation (4.3) does not have a unique solution. As such, one cannot apply directly the theorem of [59] to this SIS model.

A mathematical solution to this is to redefine our SIS model to obtain a model in dimension 1. By replacing the occurrences of x_I by $1 - x_S$ in all equations. Yet, if this is easily done for reducing a 2D model to a 1D model, it can be cumbersome when going from a 20D to a 15D model. Our tool allows doing this automatically. This is how we can obtain Table 4.1 while using the DDPP defined in Section 4.2.

Our approach to this problem is to compute the rank of the set of transitions $\mathcal{L}$. If this rank is $d' < d$, this means that the model evolves in a d'-dimensional state space. In particular, the jacobian A used in Equation (4.3) has dimension at most d'. Our code uses the SVD decomposition of A to transform the d-dimensional Lyapunov equation (4.3) into d'-dimensional equation. This is particularly useful for heterogeneous models composed of N objects that each evolve in a S dimensional state: a natural description of the model is to construct a NS-dimensional DDPP, but that evolves in a subset of dimension $N(S-1)$ or even smaller. For instance, the cache replacement policy studied in [5, 36, 60] with N objects and S lists is naturally described as a $N(S+1)$ process but evolves in fact in a $NS - S$ state space. Using the automated dimension reduction greatly simplifies the definition of the model in the tool.

4.5 Conclusion and Discussion

In this chapter, we present a tool, called *rmf_tool*, that can be used to define and study mean field interaction models. The tool is build-in with a stochastic simulator, and methods to compute the mean field approximation and refined approximation of a given model. The tool consists on a Python library and models can be directly be defined as python objects. In the present chapter, we illustrate how the tool can be used by using a simple SIS model. Below, we discuss in more detail the applicability of the tool by giving a few examples of application, and by analyzing the computation time.

4.5.1 To which model does this apply?

The tool is provided with a number of examples that demonstrates the use of the tool and the accuracy of the approximation. These examples include:

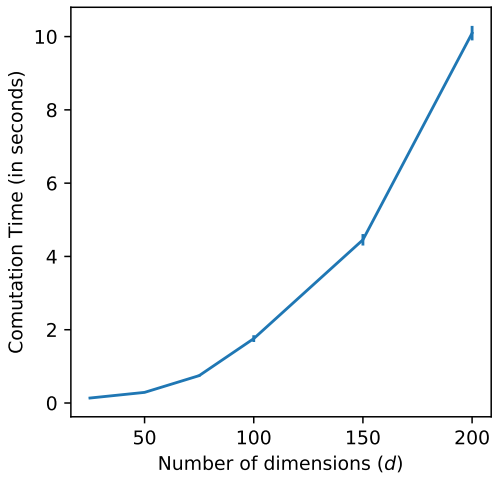
- The power of two choice model of [89]. This example models a simple, yet powerful, load balancing strategy in a system composed of N servers.
- The bike-sharing model of [50]. It models a city where CN bikes moves in a city composed of N stations.
- A epidemic model called the SIR model (that is a generalization of the SIS model presented in the chapter).

Although they are not directly provided as examples in the repository, the tool is also used in [5, 36] to analyze the performance of cache replacement policies. These cache replacement policies are examples of non-homogeneous population models.

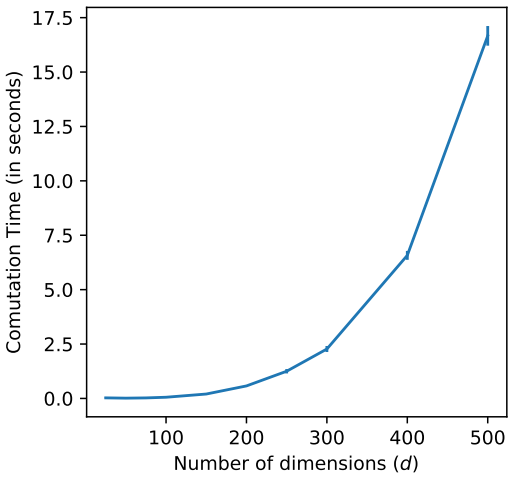
4.5.2 Analysis of the computation time

To give an idea of the time needed to compute the refined approximation, we report in Figure 4.2 the time taken by the tool to compute the refined mean field approximation as a function of the system size. The first line corresponds to a homogeneous model of dimension d: in this model, we consider the power of two choice model of [89] where we bound the queue length by d. We report the numbers of [54]. We observe that for this model, we can solve the problem for a few hundreds of dimension in less than a few tens of seconds. Note that for this example, the jacobian and the second derivative can be computed in close form. Hence, the reported time does not include the time that would be taken if one were to use symbolic differentiation.

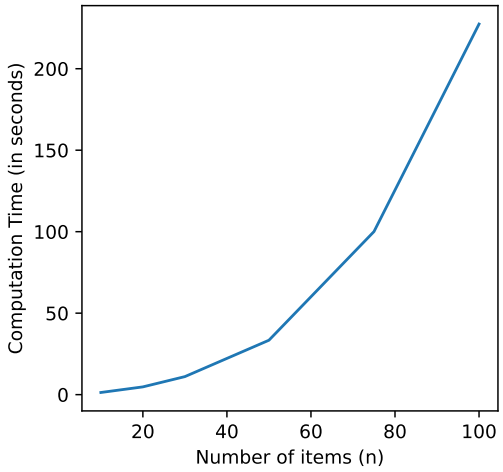
In the second line of Figure 4.2, we report the time taken to solve the heterogeneous SIS model defined in Section 4.2.3 with N different objects. For this example, we use the `HetPP` class. Note that this class does not use symbolic differentiation since the derivative can be directly computed by using the A and B tensors. The model here is a $2N$ dimensional model. We observe that the time taken here for a model with $2N$ dimension is larger than the time for a homogeneous model of dimension $2N$. We believe that the run time could be improved by using sparse tensor multiplications and will consider this question for future work.



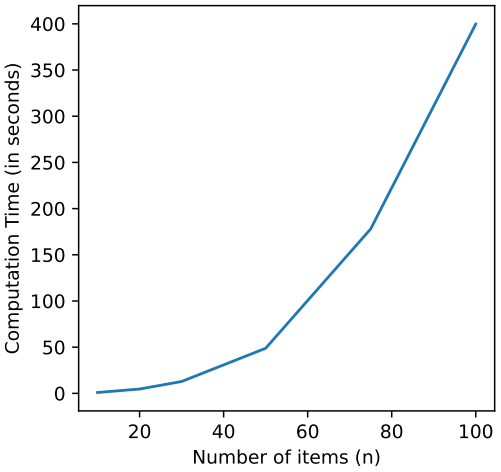
(a) Homogeneous transient



(b) Homogeneous steady-state



(c) Heterogeneous transient



(d) Heterogeneous steady-state

Figure 4.2: Analysis of the computation time: for the transient regime, we compute $\boldsymbol{v}(t)$ for $t \in [0, 10]$. For the steady-state, we compute $\boldsymbol{v}(\infty)$.

5 Accuracy of Graphon Mean Field Models

In this chaper, we present accuracy results of the graphon mean field approximation for interacting particle systems with dense, graph based connections. The chaper presents currently unpublished results.

Contents

5.1 Introduction

Mean field approximations are ubiquitously used in the study of large scale stochastic systems. Its origins date back to the '60s and '70s with the foundational papers of Kurtz [75, 76], McKean [83], Norman [93] and others. While originating from statistical physics, the mean field methodology has found application in many areas, such as communication networks [81], load balancing [87], and the study of epidemics [99]. The fundamental idea of the mean field method is to represent the particle process by a Markovian state descriptor, which is based on averaged quantities of the system. A quantity commonly used is state occupancy processes, e.g., the averaged load of stations for a bike sharing system [50] or the fraction of servers having at least a certain queue length in load balancing systems [87]. For the classical mean field method to be applied, it is crucial that the particles of the system are homogeneous and therefore exchangeable, which ensures the Markov property for the mentioned aggregate quantities.

Yet, many systems of interest are fundamentally based on heterogeneous behavior and weaker connectedness of particles, which can lead to greatly altered dynamics and inaccuracy of the classical mean field approximation. One of the main implications of such systems is that the quintessential exchangeability assumption for the classical mean field method breaks down. This further implies the necessity to keep track of the evolution of the whole range of particles in the system, making the analysis often prohibitive.

Contributions We provide bias bounds for the graphon mean field approximation for finite-sized systems consisting of $N \in \mathbb{N}$ interacting particles for which the graph G^N models the connection of the population. Our results show that it is possible to derive bias bounds which largely depend on the convergence properties of the graph sequence G^N and of its limiting graphon G. To be more precise, we start from a stochastic interacting particle model of finite size N, where each individual k is characterized by a time-varying state $S_k(t)$. The connection of the particle to the population is given by the edges (k, l) of a graph G^N. Based on this description, we construct a (deterministic) integro-differential equation based on the graphon G, and show that it has a unique solution $\boldsymbol{x}^G(t)$ that we call the graphon mean field approximation. This differential equation is constructed such that for fixed N, $x^G_{k/N,s}(t)$ approximates the probability of particle $k \in [N]$ to be in state s at time t. Denoting $\mathbb{P}(S_k^{G^N}(t) = s)$ this probability, our main result shows that

$$\mathbb{P}(S_k^{G^N}(t) = s) = x^G_{k/N,s}(t) + \mathcal{O}(\frac{1}{N} + \left|\left|\left|G^N - G\right|\right|\right|_{L_2}),$$

where $\left|\left|\left|G^N - G\right|\right|\right|_{L_2}$ is the L_2 distance of the step graphon representation of G^N and the graphon G, which is equivalent to the distance implied by graphon typical cut norm, see Theorem 15 for a precise description.

To show the extent of the result, we consider two specific graph sampling strategies for G^N. In the first case (which we denote *deterministic sampling*), G^N is the discretization of the limiting graphon G, with $G^N_{kl} := G_{k/N,l/N} \in (0,1]$ being the strength of interaction between two particles. We consider this case as it illustrates how our result can be used to model In the second case (that we call *stochastic sampling*), G^N is a random graph generated from G, where an edge is present between two nodes k and ℓ with probability $G(k/N, l/N)$. The second case corresponds to a generalization of Erdős–Rényi graph and stochastic block models to possibly non-uniform probabilities.

Imposing some mild assumptions such as a piecewise Lipschitz condition on the graphon G or symmetry of G, in the case of stochastic sampling, we can bound the distance between the graph G^N and the graphon G by:

$$|||G^N - G|||_{L_2} = \begin{cases} \mathcal{O}(N^{-1}). & \text{(Deterministic Sampling)} \\ \mathcal{O}(\sqrt{\frac{\log(N)}{N}}) \text{ w.h.p.}, & \text{(Stochastic Sampling)} \end{cases}$$

Here, with high probability (w.h.p) means that the right-hand side in case of the stochastic sampling holds with probability at least $1 - 2/N$. The precise assumptions and results are given in the Corollaries 17 and 16. We point out that the main result applies to the considered cases but is not limited to those. It is possible to consider other graph sampling strategies that satisfy our assumptions and for which the L_2 distance between the step graphon representation of G^N and the graphon G can be bounded. To illustrate our results and emphasize their applicability, we provide two examples, one each for the stochastic and deterministic sampling method. Our first example considers a load-balancing system with stochastically drawn connections between servers. Jobs in the load balancing system arrive at a server site, where each server similarly acts as a dispatcher and keeps or forwards the job according to the JSQ(2) policy based on its connected neighbors. The second example illustrates the application of deterministic sampling for a bike sharing system, with particles being stations and the graphon determining the popularity of the stations. In both cases, a simple discretized version of the integro-differential equation already yields precise estimations of the system dynamics while the numerical complexity of the approximation only slightly increases compared to the homogeneous case.

Organization The chapter is organized as follows. In Section 5.2 we introduce the heterogeneous particle model. Section 5.3 defines graphon, related sampling methods and related preliminary results. Our the definition of the approximation, the main results and proofs are displayed in Section 5.4. Last, in Section 5.5 we present the two numerical examples.

5.1.1 Related Work

Dynamical Systems on Random Graphs In recent years there has been growing interest in the behavior of interacting particles that are interconnected by an underlying graph topology, for example, [1, 14, 19, 20] and previously mentioned references. For a general introduction to the topic of random graph networks and limiting graphon functions for dense graphs, we refer to the works [82, 106]. The majority of papers focus on dense graphs, having edges of order N^2 such as Erdős–Rényi type graphs or graphs generated by stochastic block models. Additional related work

can be found in the game theoretic setting for graphon mean field games, see for example [7, 35]. In the 'not-so-dense' setting available results are more limited, with some newer references being [13, 44]. In the case of sparse graphs, such as d-regular graphs or random geometric graphs, the typical mean field methods break down as they fail to capture the importance of the spatial graph structure and its implications on the local dynamics of particles. Some recent works in this setting include [52, 53, 96].

Load Balancing on Graphs Our load balancing example is inspired by the recent works of [34, 100, 116, 117]. Here, the authors study a variety of load balancing systems with dynamics based on compatibility or locality constraints, which give rise to intricate connectivity between dispatchers and servers. While not directly transferable into the framework of this paper, the authors similarly deal with graph-based systems and limit approximations that are strongly related to the ones our framework suggests. Note that the techniques developed in these papers are very model-specific and allow for transitions to depend on the states of multiple queues, whereas our approach aims at deriving results for a more general framework for transition rates with the restriction to the case of pair interactions.

Generator and Stein's Method For our proofs, we adapt techniques used in [5, 55, 59], which in turn rely on the use of Stein's method [102]. The method is used to estimate and bound the distance between two random variables through their respective generators. Since the works [31, 33] Stein's method has seen an increase in the stochastic network community and is an actively evolving area.

5.2 Heterogeneous Network Particle System

5.2.1 The interaction model

We consider particle systems with $N \in \mathbb{N}$ interacting particles. Connections between particles are characterized by a (possibly weighted) adjacency matrix $G^N \in [0,1]^{N\times N}$ of size $N \times N$, with G^N_{kl} indicating the connection strength between particle k and l. Each of the particles has a finite state space $\mathcal{S}$ where the state of the k-th object at time $t \geq 0$ is denoted by $S_k^{G^N}(t)$. As we see later, G^N can correspond to a random graph for which $G^N_{kl} \in \{0,1\}$ indicates the presence or absence of an edge between the particles k and l or can be an arbitrary weighted matrix, see Section 5.3.

The state of the whole system is denoted by $\boldsymbol{S}^{G^N}(t) := \left(S_1^{G^N}, \dots, S_N^{G^N}\right)(t) \in \mathcal{S}^N$. We assume that the process $\boldsymbol{S}^{G^N} := (\boldsymbol{S}^{G^N}(t))_{t\geq 0}$ is a continuous time Markov chain (CTMC) with the dynamics of the system described as in the following.
Each particle $k \in [N]$ changes its state from s_k to $s'_k \in \mathcal{S}$ in one of the two ways:

(Unilateral) The particle independently changes its state at rate $r^{N,\text{uni}}_{k,s_k\to s'_k}$.

(Pairwise) The local change of states is triggered by another particle $l \in [N]$ that is in state $s_l \in \mathcal{S}$. This occurs at rate $r^{N,\text{pair}}_{k,s_k\to s'_k,s_l} G^N_{kl}/N$.

Note that the rate functions are assumed to be heterogeneous, i.e., they can depend on the items k and l. The rates have a $1/N$ factor, as each particle can potentially interact with all $N-1$ remaining particles. Hence, our condition implies that the total rate of transitions of the particle system is of order $O(N)$ and that the transition rates for all particles are of the same order. As we will further discuss in Example 5.5.1, our results can also be used if the scaling factor depends not on the system size N but on the node degrees. We further want to point out that we restrict our framework to the interaction of two particles, which can be utilized to model many relevant interacting particle systems on graphs such as *e.g.*, epidemic spreading, power-of-two-choices load-balancing, or bike sharing systems. It is nonetheless possible to use the same underlying approach to extend the framework and results to interactions of higher order. This, however, comes at the cost of increasingly cumbersome notations and limited added theoretical insight.

5.2.2 The binary state representation

In order to ease computations and definitions, we will use a binary representation of the state based on indicator functions. We denote the new representation by $\boldsymbol{X}^{(N)} = (X^{(N)}_{k,s}(t))_{\substack{k\in[N]\ s\in\mathcal{S}\\ t\geq 0}}$, where

$$X^{(N)}_{k,s}(t) := \mathbb{1}_{\{S^{G^N}_k(t)=s\}} := \begin{cases} 1 & \text{if object } k \text{ is in state } s \text{ at time } t,\\ 0 & \text{otherwise.}\end{cases}$$

The space of attainable states is denoted by $\mathcal{X}^N \subset \{0,1\}^{N\times\mathcal{S}}$.

While this representation is less compact than the original, it allows for an easier definition of transition rates as well as the definition of the mean field approximation. Denote by $e^N_{k,s}$ a matrix of size $N \times |\mathcal{S}|$ whose (k,s) component is equal to 1, all other entries being zero. For each $k \in [N]$, and $s_k, s'_k \in \mathcal{S}$, $X^{(N)}$ jumps to $X^{(N)} + e^N_{k,s_k} - e^N_{k,s'_k}$ at rate

$$X_{k,s_k} r^{N,\text{uni}}_{k,s_k\to s'_k} + X_{k,s_k} \sum_{l\in[N]} \sum_{s_l\in\mathcal{S}} r^{N,\text{pair}}_{k,s_k\to s'_k,s_l} \frac{G^N_{kl}}{N} X_{l,s_l}. \tag{5.1}$$

In the above equation, the first term of the rate corresponds to the unilateral transition of the particle $k \in [N]$ changing its state from s_k to s'_k, as this transition occurs at intensity $r_{k,s_k\to s'_k}$ if particle k is in state s_k (*i.e.*, $X_{k,s_k} = 1$). The second term describes the pairwise transitions leading to the state change of particle k from s_k to s'_k. Similar to the unilateral one, the transition can only happen if particle k is in state s_k, represented in the rate by the prefactor X_{k,s_k}. The remaining factor corresponds to the interaction with other particles, expressed by the weighted sum over all other particles and their states. The intensity of the transition is scaled by $r^{N,\text{pair}}_{k,s_k\to s'_k,s_l}$ and the connectivity of the particles is given by the connectivity matrix $G^N \in [0,1]^{N\times N}$.

5.2.3 Drift of the system of size N

By using the state representation $\boldsymbol{X}^{(N)}$, we define what we call the *drift* of the system of N particles as the expected change for $\boldsymbol{X}^{(N)}$ in state $\boldsymbol{X} \in \mathcal{X}^{(N)}$. It is equal to the sum of all possible transitions

of the changes induced by this transition times the rate at which the transition occurs. We denote this quantity as $\boldsymbol{F}^{G^N}(\boldsymbol{X})$. By using Equation (5.1), it is equal to:

$$\boldsymbol{F}^{G^N}(\boldsymbol{X}) = \sum_{k\in[N], s_k, s'_k \in \mathcal{S}} (\boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}) \left[X_{k,s_k} r^{N,\text{uni}}_{k,s_k \to s'_k} + X_{k,s_k} \sum_{l\in[N]} \sum_{s_l \in \mathcal{S}} r^{N,\text{pair}}_{k,l,s_k \to s'_k, s_l} \frac{G^N_{kl}}{N} X_{l,s_l} \right]$$

The quantity $\boldsymbol{F}^{G^N}(\boldsymbol{X})$ is a vector-valued function of $\boldsymbol{X}$. By reorganizing the above sum, $F^{G^N}_{k,s}(\boldsymbol{X})$ –its (k,s) component– is equal to

$$F^{G^N}_{k,s_k}(\boldsymbol{X}) = \sum_{s'\in\mathcal{S}} \left[X_{k,s'} r^{N,\text{uni}}_{k,s'_k \to s_k} - X_{k,s} r^{N,\text{uni}}_{k,s_k \to s'_k} + \sum_{l\in[N]} \sum_{s_l \in \mathcal{S}} (X_{k,s'_k} r^{N,\text{pair}}_{k,l,s'_k \to s_k, s_l} - X_{k,s} r^{N,\text{pair}}_{k,l,s\to s'_k, s_l}) \frac{G^N_{kl}}{N} X_{l,s_l} \right].$$

In the following, it will be convenient to replace the above sum by matrix multiplications. To do so, let us denote by X_k the vector $(X_{k,s})_{s\in\mathcal{S}}$. The above equation can be written as:

$$F^{G^N}_{k,s}(\boldsymbol{X}) = \boldsymbol{R}^{N,\text{uni}}_{k,s} \boldsymbol{X}_k + \boldsymbol{X}^T_k \sum_{l\in[N]} \boldsymbol{R}^{N,\text{pair}}_{k,l,s} \boldsymbol{X}_l \frac{G^N_{kl}}{N}, \tag{5.2}$$

where $\boldsymbol{R}^{N,\text{uni}}_{k,s}$ is a row vector whose s' component is $r^{N,\text{uni}}_{k,s'\to s}$ if $s' \neq s$ and $-\sum_{\tilde{s}} r^{N,\text{uni}}_{k,s\to\tilde{s}}$ for $s = s'$; and $\boldsymbol{R}^{N,\text{pair}}_{k,l,s}$ is a matrix whose (s', s_l) component is $r^{N,\text{pair}}_{k,l,s'\to s,s_l}$ if $s' \neq s$ and $-\sum_{\tilde{s}} r^{N,\text{pair}}_{k,l,s\to\tilde{s},s_l}$ if $s = s'$.

5.2.4 Representation of $\boldsymbol{X}^{(N)}$, G^N and F^{G^N} as function from $[0,1]$

To study the limit as N goes to infinity, it will be convenient to view the functions $\boldsymbol{X}^{(N)}$ not as a vector with N components $(X^{(N)}_k)_{k\in[N]}$ but as a function $(X^N_u)_{u\in[0,1]}$, where for any state $s \in \mathcal{S}$, we set

$$X^{G^N}_{u,s} := X^{G^N}_{k,s} \in \{0,1\} \text{ for } u \in ((k-1)/N, k/N].$$

By abuse of notation, we do not introduce separate notations for the discrete and continuous variables but make the distinction by reserving the subscript letters $k, l \in [N]$ for the discrete case and $u, v \in (0,1]$ for the continuous variable.

Similarly, we also write $G^N_{uv} = G^N_{kl}$ for $u \in ((k-1)/N, k/N]$ and $v \in ((l-1)/N, l/N]$, $F^{G^N}_{u,s} = F^{G^N}_{k,s}$, and $\boldsymbol{R}^{N,\text{uni}}_{k,s} = \boldsymbol{R}^{N,\text{uni}}_{u,s}$. By using this notation, the sum of $k \in [N]$ –for instance in (5.2)– can be replaced by an integral, i.e., $F^{G^N}_{k,s}(\boldsymbol{X}) = F^{G^N}_{u,s}(\boldsymbol{X}) = \boldsymbol{R}^{N,\text{uni}}_{u,s} \boldsymbol{X}_u + \boldsymbol{X}^T_u \int_0^1 \boldsymbol{R}^{N,\text{pair}}_{u,v,s} \boldsymbol{X}_v G^N_{uv} dv$ for $u \in ((k-1)/N, k/N]$.

5.2.5 Notations

Throughout the paper, matrices, and vectors are written in bold letters, i.e. $\boldsymbol{X}, \boldsymbol{x}, \ldots$, and regular letters are used to denote scalars like $X_{u,s}, r^{\text{uni}}_{u,s\to s'}$. The indices $s, s_k, s'_k, s_l, \ldots$ are reserved for the

states, $k, l, \ldots$ are reserved for particles, and u, v are reserved for values in the unit interval. When we write that a quantity h is of order $\mathcal{O}(1/N)$ or $h = \mathcal{O}(1/N)$ equivalently, this means that there exists a constant C such that $h \leq \frac{C}{N}$. Most of our results will be expressed in terms of L_2 norm (see the definition in Section 5.3.3). The space $L_2(0,1]$ refers to the quotient space of the square Lebesgue integrable functions. Throughout we will deal with vectors of $L_2(0,1]$ functions, i.e., $f, g \in L_2(0,1]^{\mathcal{S}}$. We for the vectors g, f we denote the scalar product by $\langle f, g \rangle_{L_2(0,1]} = \sum_{s\in\mathcal{S}} \int_0^1 f_{u,s} g_{u,s}\, du$ for and induced norm $\|f\|_{L_2} = \sqrt{\sum_{s\in\mathcal{S}} \int_0^1 f_u^2 du}$. For a function $G : (0,1]^2 \to \mathbb{R}$ denote by $|||G|||_{L_2} = \sup_{\{f\in L_2(0,1], \|f\|_{L_2(0,1]}\leq 1\}} \sqrt{\int_0^1 (\int_0^1 G_{uv} f_v\, dv)^2 du}$ the L_2 operator norm.

5.3 Limiting Graph and Graphon

In this section, we specify the properties that the interaction graph G^N needs to satisfy as N goes to infinity. To do so, we introduce the notion of graphon and define the associated cut-norm. We also introduce two sampling methods that can be used to generate a graph with N nodes from a graphon. This section only reviews the material that is necessary for our results, and we refer to the famous book [82] for a detailed introduction to graphons.

5.3.1 Graphons

In this chapter, what we call a graphon is a measurable function[1] $G : (0,1]^2 \to (0,1]$. The notion of graphon can be viewed as a generalization of the notion of graph. Indeed, for any N, a weighted graph G^N can be viewed as a piecewise constant function defined on $(0,1]^2$, where the value of this function at a point $(u,v) \in (0,1]^2$ is equal to G^N_{kl} whenever $u \in ((k-1)/N, k/N]$ and $v \in ((l-1)/N, l/N]$. Hence, a finite graph is a graphon that has a special structure. The notion of graphon generalizes the notion of finite graph by allowing G to be any measurable function. We provide an illustration of a graphon and of a finite-graph viewed as a graphon on Figure 5.1.

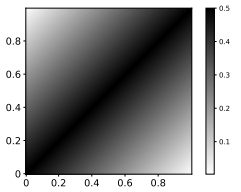
Throughout the paper, we consider piecewise Lipschitz continuous graphons, which are defined as follows.

Definition 11 (Piecewise Lipschitz Graphon)**.** *A graphon G is called piecewise Lipschitz if there exists a constant L_G and a finite partition of $(0,1]$ of non-overlapping intervals $\mathcal{A}_k = (a_{k-1}, a_k]$ with $0 = a_0 < a_1 < \ldots < a_{K_G} = 0$ for a finite $K_G \in \mathbb{N}$, such that for any $k_1, k_2 \in [K+1]$ and pairs $(u,v), (u',v') \in \mathcal{A}_{k_1} \times \mathcal{A}_{k_2}$*

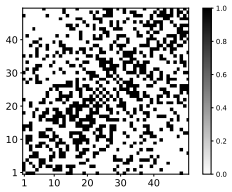
$$|G_{uv} - G_{u'v'}| \leq L_G(|u-u'| + |v-v'|).$$

A particular case of a piecewise Lipschitz continuous graphon is the case of a step graphon, that is such that the Lipschitz constant $L_P = 0$. This implies that the function G is constant on all intervals $\mathcal{A}_i \times \mathcal{A}_j$. For instance, all finite graphs can be seen as step graphons by using the partition such that $\mathcal{A}_k = (k - 1/N, k/N]$ with N being the number of nodes in the graph.

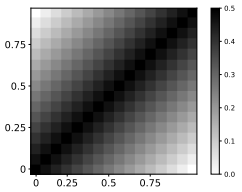
[1] In the literature, graphons are often restricted to symmetric functions, which correspond to undirected graphs. This restriction is not needed for our case.



(a) Graphon corresponding to $G_{uv} = \frac{1}{2}(1 - |u - v|)$



(b) Step graphon associated to a sampled graph with $N = 50$ nodes.



(c) Step graphon obtained from graphon discretization for $N = 16$.

Figure 5.1: Exemplary connectivity functions sampled after the methods of Section 5.3.2 and their corresponding graphon.

5.3.2 Generation of a Finite Graph G^N from a Graphon G

Based on a given graphon G, we consider two distinct sampling methods to generate a finite graph G^N from G:

- In the first case, termed *deterministic sampling*, G^N is a discretization of G on N^2 uniformly sampled points, i.e., $G^N_{kl} = G(k/N, l/N), k, l \in [N]$.
- For the second case, termed *stochastic sampling*, and under the preliminary assumption that the graphon is symmetric, the values G^N are drawn according to independent Bernoulli random variables, i.e., $G^N_{kl} = \text{Bernoulli}\big(G(k/N, l/N)\big)$, and $G^N_{lk} = G^N_{kl}$ for all $k < l$.

In the case that the graphon is constant for all $u, v \in (0, 1]$, the stochastic sampling method is equivalent to sampling an Erdős–Rényi graph. If the graphon is block-wise constant, the sampling is similar to the stochastic block model; see [106].

5.3.3 Graphon Distances and Convergence

To measure the distance between two graphs (and in particular to quantify how fast does a finite graph G^N converge to G), we will use the L_2 operator norm. More precisely, for a measurable function $f : [0, 1] \to \mathbb{R}$, we denote by $\|f\|_{L_2(0,1]} = \sqrt{\int_0^1 f(u)^2 du}$ the L_2 norm of f. For a graphon G, we denote by $|||G|||$ the operator norm of G in L_2, that is:

$$|||G||| = \sup_{\{f \in L_2(0,1] \text{ such that } \|f\|_{L_2(0,1]} \leq 1\}} \|Gf\|_{L_2(0,1]}.$$

The distance between two graphons (or finite graphs) G and G' is measured as $|||G - G'|||$.

To bound the L_2 distance between a piecewise Lipschitz graphon and the function associated to a randomly sampled graph, we utilize the results displayed in [8]. The authors show that for a large enough (see Definition 13) number N of graph nodes, in our case particles, the distance between the graphon and the graph function can be upper bounded as follows:

Lemma 12 (Theorem 1 from [8])**.** *Let G be a symmetric piecewise Lipschitz graphon, and let G^N be a stochastic sampling of it as defined in 5.3.2. Then, for large enough N (as in Definition 13) with*

probability as least $1-\delta$ *the distance in the* L_2 *operator norm between the graphon* G *and the sampled graph* G^N *is bounded by*

$$\left|\left|\left|G - G^N\right|\right|\right|_{L_2(0,1]} \leq \sqrt{\frac{4\log\left(2N/\delta\right)}{N}} + 2\sqrt{\frac{(L_G^2 - K_G^2)}{N^2} + \frac{K_G}{N}} =: \psi_{\delta,G}(N). \tag{5.3}$$

Definition 13 (Large Enough N [8])**.** *Given a piecewise Lipschitz graphon* G *with partition of* $(0,1]$ *of non-overlapping intervals* $\mathcal{A}_k = (a_{k-1}, a_k]$ *as in Definition 11 and* $\delta < e^{-1}$*. The quantity* N *is called 'large enough', if*

$$\frac{2}{N} < \min_{k\in\{0..K_G\}}(a_k - a_{k-1}), \tag{5.4a}$$

$$\frac{1}{N}\log\left(\frac{2N}{\delta}\right) + \frac{2K_G + 3L_G}{N} < \sup_{u\in(0,1]}\int_0^1 G_{u,v}\,dv \quad \textit{and} \tag{5.4b}$$

$$Ne^{-N/5} < \delta. \tag{5.4c}$$

Implied by Theorem 12, by setting $\delta_N = \frac{2}{N}$, we see that the distance between a piecewise Lipschitz graphon and a sampled graph is with high probability of order $\left|\left|\left|G - G^N\right|\right|\right|_{L_2(0,1]} = \mathcal{O}\left(\sqrt{\frac{\log(N)}{N}}\right)$. The meaning of 'with high probability' in this context is that the right-hand side holds with probability at least $1 - 2/N$.

5.4 Main Results

5.4.1 Assumptions

In Section 5.2, we constructed a model that depends on a scaling parameter N. We state here the necessary assumptions that we will use in order to state the accuracy of the graphon mean field approximation.

- (A_1) The state space $\mathcal{S}$ is finite.
- (A_2) The graphon G is piecewise Lipschitz continuous.
- (A_3) There exists bounded and piecewise Lipschitz-continuous rate functions for $r^{\text{uni}}_{u,s_u\to s'_u}$ and $r^{\text{pair}}_{u,v,s_u\to s'_u,s_v}$ for $u, v \in (0,1]$ and $s_u, s'_u, s_v \in \mathcal{S}$ such that the rates function of the original N particle systems have the relation:

$$r^{\text{uni}}_{k/N,s_k\to s'_k} = r^{N,\text{uni}}_{k,s_k\to s'_k} \quad \text{and} \quad r^{\text{pair}}_{k/N,l/N,s_k\to s'_k,s_l} = r^{N,\text{pair}}_{k,l,s_k\to s'_k,s_l} \quad \text{for } k,l \in [N]. \tag{5.5}$$

The first assumption is technical and simplifies the definition of the L_2 space. The second assumption is very classical regularity assumption when studying sequences of graphs that converge to graphons [82]. In practice, our bounds will depend on the distance between the original graph

G^N and the graphon G. The last assumption ensures that the particle transition rates for the finite and the graphon system are equal. We point out that assumption (A_3) can be generalized by replacing the equality in (5.5) with bounds for the distance of $r^{N,\cdot}$ and $r^{\cdot}$ therefore modifying the bounds of the theorem to include terms of the form $\left\| r^{N,\cdot} - r^{\cdot} \right\|$.

5.4.2 The Graphon Mean Field Approximation

The graphon mean field approximation aims at approximating the dynamics of the original system $\boldsymbol{X}$. We define the graphon related drift similarly to the drift of the particle system in Equation (5.2). For $u, v \in (0, 1]$ the graphon based drift is defined by

$$F^G_{u,s}(\boldsymbol{x}) = \boldsymbol{R}^{\text{uni}}_{u,s}\boldsymbol{x}_u + \boldsymbol{x}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v} \boldsymbol{x}_v dv. \tag{5.6}$$

This equation is identical to the original drift equation (5.2) with two modifications: First, the rates do not depend on N. Second, the discrete variables $k, l \in [N]$ are replaced by continuous variables $u, v \in (0, 1]$, which notably replaces the sum over l by an integral over v.

Based on the drift function $\boldsymbol{F}^G$ and the initial condition x_0, we call the graphon mean field approximation the solution of the following differential equation:

$$\boldsymbol{x}^G(t, \boldsymbol{x}_0) = \boldsymbol{x}_0 + \int_0^t \boldsymbol{F}^G(\boldsymbol{x}^G(\tau, \boldsymbol{x}_0))d\tau. \tag{5.7}$$

Lemma 14. *Let $\boldsymbol{F}^G$ denote the deterministic drift defined in Equation (5.6) and $\boldsymbol{x}^G(t, \boldsymbol{x}_0)$ the solution of the graphon mean field approximation at time t with initial condition $\boldsymbol{x}_0$ as in Equation (5.7). It holds that $\boldsymbol{x}^G$ is well-defined, has a unique solution and is differentiable with respect to its initial condition. Furthermore, this derivative is Lipschitz continuous.*

The proof is postponed to Section 5.6.1

5.4.3 Accuracy of the approximation

The following result provides a bound on the distance between the stochastic system and the graphon mean field approximation. This bound is stated as the L_2 distance between $\mathbb{E}[\boldsymbol{X}^{(N)}]$ and $\boldsymbol{x}^G$. Recall that by definition, $\mathbb{E}[X^{(N)}_{k,s}(t)] = \mathbb{P}(S_k(t) = s)$ is the probability for an item k to be in state s at time t. Hence, our theorem shows that the graphon mean field model is an accurate approximation of the state distribution of the particles. The statement of the theorem should be interpreted as saying that the distribution of the particles over the states $\mathcal{S}$ for any time $t \geq 0$ is approximated by the graphon mean field approximation $\boldsymbol{x}^G$ with accuracy depending on the system size N as well as the distance between G^N and graphon G. In particular, if G^N converges to G (for the graphon norm), then the graphon mean field approximation is asymptotically exact.

Theorem 15 (L_2 convergence)**.** *Let $\boldsymbol{X}^{(N)}(t) = (X^{(N)}_{k,s})(t)_{k\in[N],s\in\mathcal{S}}$ be the stochastic particle system of size N related to a graph instance G^N. Let $\boldsymbol{x}^G(t) = (x^G(t, \boldsymbol{x}_0))_{u\in(0,1],s\in\mathcal{S}}$ be the mean field*

approximation of the particle system as defined in (5.7) *for an initial condition* $\boldsymbol{x}_0 = \boldsymbol{X}^{(N)}(0)$[2]. *Assume additionally that conditions* (A_1) - (A_3) *are fulfilled and let* $t > 0$ *be arbitrary but fixed. Then, there exist constants* $C_A, C_B \geq 0$ *such that*

$$\left\| \mathbb{E}[X^{(N)}(t) \mid \boldsymbol{X}^{(N)}(0), G^N] - x^G(t, \boldsymbol{X}^{(N)}(0)) \right\|_{L_2} = \frac{C_A}{N} + C_B \left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|. \tag{5.8}$$

The proof of Theorem 15 is postponed to Section 5.6.2. The constants C_A, C_B of Equation (5.8) depend on the uniform bound of the rates, the time t and the Lipschitz constants of the graphon. In the subsequent Corollaries 16 and 17 we will see, that if the G^N is generated by one of the methods illustrated in Section 5.3.2, precise bounds on the distance $\left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|$ can be obtained. To illustrate our results and underline that the constants are small in practice, we provide examples in Section 5.5. We point out that Theorem 15 is applicable for a wide range of construction methods for G^N. The subsequent corollaries illustrate cases of stochastic and deterministic sampling methods. We emphasize, however, that any method that allows the construction of densely connected graphs, for which bounds of $\left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|$ are attainable, is viable. At last, we want to point out that by using the same framework, it seems feasible to extend our results to interactions of triplets or higher order interactions seems feasible. Yet, due to our generic choice of transition rates, this would be linked to increasingly heavy notations for the dynamics while only giving little more insight into the accuracy of the graphon mean field method.

5.4.4 Case Specific Bounds for $\left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|_{L_2}$

The subsequent corollaries, give specific bounds for the case that G^N was generated as described in Section 5.3.2. In the first case, if G^N is obtained though discretization of G, Corollary 16 shows that the accuracy is of order $\mathcal{O}(1/N)$. Our second Corollary 17 specifies big-$\mathcal{O}$ convergence rates if the graph G^N is sampled stochastically from the graphon. In this case, the accuracy is with high probability of order $\mathcal{O}(\sqrt{\frac{\log N}{N}})$.

Case 1: Graphon Discretization

The corollary give accuracy bounds for the approximation in the case that the graph G^N is obtained as a discretized version of G. The result gives bounds on the difference between the distributions of the particles in the stochastic system and the approximate values obtained through the graphon mean field approximation.

Corollary 16 (Dense Heterogeneous System)**.** *Assume* (A_1) - (A_3) *and let* $\boldsymbol{x}^G$ *and* $\boldsymbol{X}^{(N)}$ *be defined as in Theorem 15. Let* $k \in [N], s \in \mathcal{S}$ *and* $t \geq 0$ *be arbitrary but fixed. If* G^N *is generated by the deterministic sampling method of 5.3.2, i.e., a discretization of the graphon* G*, it holds that*

$$\left\| \mathbb{E}[\boldsymbol{X}^{(N)}(t) \mid \boldsymbol{X}^{(N)}(0), G^N] - \boldsymbol{x}^G(t, \boldsymbol{X}^{(N)}(0)) \right\|_{L_2} \leq \frac{C_A + C_B C_{G^N}}{N} \tag{5.9}$$

[2]Here, $\boldsymbol{X}^{(N)}(0)$ is interpreted as a vector in $L_2(0,1]^{|\mathcal{S}|}$.

The proof is postponed to Section 5.6.3. The constants C_A, C_B are as in Theorem 15. The additional constant C_{G^N} relates to the discretization error of the deterministic sampling method. For this case, it is noteworthy that the accuracy of the approximation aligns with the results one obtains for the homogeneous setting as described in [55, 59] while allowing for heterogeneous connectivity and rates among the population.

Case 2: Random Graph

Our second corollary provides accuracy bounds for interacting particle systems on dense random graphs. By the definition of the graph sampling methods, see Section 5.3.2, with high probability, the number of connected neighbors for each particle is of order N. This ensures that for large enough systems that the neighborhood of each node serves as a local representation of the overall system state. This leads to the following result:

Corollary 17 (Graphon System Approximation)**.** *Assume the conditions (A_1) - (A_3) as in Theorem 15 for a symmetric graphon G. Let $\boldsymbol{X}^{(N)}(t) = (X^{(N)}_{k,s})(t)_{k\in[N],s\in\mathcal{S}}$ be a stochastic particle system of size N with G^N obtained through the stochastic sampling method as defined in 5.3.2. Let $t \geq 0$ and $k \in [N], s \in \mathcal{S}$ be arbitrary but fixed. Then, with probability at least $1-2/N$ and for 'large enough' N, as defined in Definition 13, the graph G^N is sampled such that*

$$\left\| \mathbb{E}[\boldsymbol{X}^{(N)}(t) \mid \boldsymbol{X}^{(N)}(0), G^N] - \boldsymbol{x}^G(t, \boldsymbol{X}^{(N)}(0)) \right\|_{L_2} \leq \frac{C_A}{N} + C_B \psi_G(N) \tag{5.10}$$

where $\psi_G(N) := \sqrt{\frac{8\log(N)}{N}} + 2\sqrt{\frac{(L_G^2-K_G^2)}{N^2} + \frac{K_G}{N}}$ with L_G being the Lipschitz constant of the graphon G and K_G the size of the partition as defined in 11.

The proof of the corollary is postponed to Section 5.6.3.

5.5 Numerical Experiments

In this section, we present two examples which support the statements of our theoretical results and illustrate the applicability of the framework. For the first example, we look at a load balancing model with communication restrictions of the servers imposed by a graph. In this example, the focus is on the stochastically sampled graph, which imposes heterogeneous rates due to connectedness of the servers. For the dynamics of the system, we see each node as a dispatcher / server pair applying the JSQ(2) policy with respect to itself and connected servers whenever a job arrives. For the second example, we consider a heterogeneous bike-sharing system. Here, the focus lies on the heterogeneity of the popularity of the stations, which affects the flow of bikes through the system.

5.5.1 Load Balancing Example

Model

The considered load balancing model is a jump process in the Markovian setting. We consider a system where the servers-dispatcher pairs are connected through a graph structure where each server-dispatcher is represented by a node. All servers have a finite maximal queue length $K_L \in \mathbb{N}$. The

connections between the server-dispatcher pairs are denoted by G^N_{kl} and are sampled according to Section 5.3.2 using the stochastic sampling method. The graphon G used to sample the edges is the same as Figure 5.1a. Based on the described connectivity structure, jobs arrive to a server-dispatch pair following a Poisson arrival process with rate $\lambda_L > 0$. Arriving jobs are distributed according to the JSQ(2) policy idea, i.e., the dispatcher considers its own server state as well as another randomly sampled but connected server and forwards the job to the server with the lesser load. In the case that both servers have the same queue length, the job is assigned randomly among the two. In case both servers have a full queue, the job is discarded. The service time of a job is exponentially distributed with mean μ_L and jobs are handled in first come, first served order. For a system of size N with graph instance G^N and state $\boldsymbol{X}^{(N)} = (X^{(N)}_{k,s})_{k\in[N],s=0..K_L}$ the transitions of the system are

$$\boldsymbol{X}^{(N)} \to \boldsymbol{X}^{(N)} + \boldsymbol{e}^N_{k,s+1} - \boldsymbol{e}^N_{k,s} \tag{5.11}$$

$$\text{at rate } \lambda_L X^{(N)}_{k,s} \sum_{l\in[N]} \left(\frac{G^N_{kl}}{d^{(N)}(k)} + \frac{G^N_{kl}}{d^{(N)}(l)}\right)\left(\frac{1}{2}X_{l,s} + \sum_{s_l\geq s+1} X_{l,s_l}\right)\mathbb{1}_{s<K_L} \tag{5.12}$$

$$\boldsymbol{X}^{(N)} \to \boldsymbol{X}^{(N)} + \boldsymbol{e}^N_{k,s-1} - \boldsymbol{e}^N_{k,s} \quad \text{at rate } X^{(N)}_{k,s}\mu_L\mathbb{1}_{s>0} \tag{5.13}$$

where $\boldsymbol{e}_{k,s}$ is a $N \times |\mathcal{S}|$ matrix having a one at the (k, s) entry and zero values everywhere else and $d^{(N)}(k)$ is the degree of node k. Equation (5.13) corresponds to the completion of a job by server k when the queue is of size $1 \leq s \leq K_L$. The second type of transitions, equation (5.12), corresponds to adding a job to k having $0 \leq s \leq K_L - 1$ jobs in the buffer. In this case, the queue size is increased by one from s to $s + 1$. To explain the transition rate we see that the servers k can be selected in two ways. By selecting k or another connected server l first and the other second. In the case that both queues have equal length, the chance that the job to arrives at server k is $1/2$. If both buffers are full, the job is discarded. In the case that a node associated to server l is isolated, i.e., has no edges, we define $\frac{G^N_{kl}}{d^{(N)}(l)}$ to be zero.

Drift and Graphon Mean Field Approximation

For the deterministic drif, we replace the values of $\frac{G^N}{d^{(N)}(k)}$ by the ones of the graphon $\frac{G}{d(v)}$ with $d(v) := \int_0^1 G_{v,\nu}d\nu$ and the sums over particles by an integral over $(0, 1]$. For a given state $\boldsymbol{x} = (x_s)_{s=0,\dots,K_L}$ with $x_s \in L_2(0, 1]$, the drift evaluated at $(u, s) \in (0, 1] \times \{0, \dots, K_L\}$ is defined by

$$F^G_{u,s}(\boldsymbol{x}) = x_{u,s}\mathbb{1}_{s<K_L}\int_0^1 \lambda_L\left(\frac{G_{u,v}}{d(u)} + \frac{G_{u,v}}{d(v)}\right)\left(\frac{1}{2}x_{v,s} + \sum_{s_v\geq s+1} x_{v,s_v}\right)dv \; - x_{u,s}\mu_L\mathbb{1}_{s>0}.$$

The graphon mean field approximation is then defined as in Equation (5.7).

Derivation of Accuracy Bounds

While edges between the server-dispatcher pair are sampled according to the stochastic sampling method of Section 5.3.2, the dependence on of the rates on the node degree, prevents a direct application of the results of Corollary 17. In particular, to apply Corrollary 17 it is assumed that the graph edges are scaled by a factor of $1/N$ instead of the node degree. To resolve this issue and

obtain accuracy bounds similar to the one given by the Corollary, we start by defining $\tilde{G}^N_{kl} := (\frac{G^N_{kl}}{d^{(N)}(k)} + \frac{G^N_{kl}}{d^{(N)}(l)})\frac{N}{16}$, $r^{N,\text{pair}}_{k,l,s\to s+1,s_l} = \lambda_L(\frac{1}{2}\mathbb{1}_{s_l=s} + \mathbb{1}_{s_l>s})\mathbb{1}_{s<K_L}$, and $r^{N,\text{uni}}_{k,s\to s-1} = \mu_L\mathbb{1}_{s>0}$ with $r^{N,\text{pair}}$ being the pairwise transitions and $r^{N,\text{uni}}$ the uniateral transitions for $s, s', s_l \in \{0, ..., K_L\}$. This allows to obtain the drift of the stochastic system similar to the one introduced in Equation (5.2) namely

$$F^{G^N}_{k,s}(\boldsymbol{X}) = X_{k,s}\sum_{s_l\geq s}\sum_{l\in[N]} 16\frac{\tilde{G}^N_{kl}}{N} r^{N,\text{pair}}_{k,l,s\to s+1,s_l} X_{l,s'} - X_{k,s} r^{N,\text{uni}}_{k,s\to s-1}.$$

We proceed similarly for the drift of the graphon mean field approximation by defining $\tilde{G}_{uv} := (\frac{G_{u,v}}{d(u)} + \frac{G_{u,v}}{d(v)})\frac{1}{16}$, $r^{\text{pair}}_{k,l,s\to s+1,s_l} := \lambda_L(\frac{1}{2}\mathbb{1}_{s_l=s} + \mathbb{1}_{s_l>s})\mathbb{1}_{s<K_L}$ and $r^{N,\text{uni}}_{k,s\to s-1} := \mu_L\mathbb{1}_{s>0}$ to rewrite

$$F^{G}_{u,s}(\boldsymbol{x}) = x_{u,s}\int_0^1 \sum_{s_v\geq s} 16\tilde{G}_{uv} r^{\text{pair}}_{k,l,s\to s+1,s_v} x_{v,s_v} dv - x_{u,s} r^{N,\text{uni}}_{k,s\to s-1}.$$

Note that for the above reformulations, the result of our main Theorem 15 is still applicable and yields constants $C_A, C_B \geq 0$ such that $\left\|\mathbb{E}[\boldsymbol{X}^{(N)}(t) \mid G^N] - \boldsymbol{x}^G(t)\right\|_{L_2} \leq \frac{C_A}{N} + C_B\left|\left|\left|\tilde{G}^N - \tilde{G}\right|\right|\right|_{L_2}$. Using the definition of the $\tilde{G}^N, \tilde{G}$, we obtain $\left|\left|\left|\tilde{G}^N - \tilde{G}\right|\right|\right|_{L_2} \leq \frac{1}{16}\Big(\left\|G^N - G\right\|_{L_2}\frac{N}{\min_k d^N(k)} + \sup_{u\in(0,1]}\left|\sum_{k=1}^N \frac{N}{d^N(k)}\mathbf{1}_{u\in(k-1/N,k/N]} - \frac{1}{d(u)}\right|\Big)$. For $\left|\left|\left|G^N - G\right|\right|\right|_{L_2}$, we can use the bound as in Corollary 17 which is of order $\mathcal{O}(\sqrt{\frac{\log N}{N}})$ with probability at least $1 - \frac{2}{N}$ and large enough N (as in Definition 13). To obtain similar bounds for the node degree, the multiplicative Chernoff bound can be used, i.e., $\mathbb{P}\Big(\frac{N}{d^N(k)} \geq \frac{N}{(1-\gamma)\mathbb{E}[d^N(k)]}\Big) = \mathbb{P}\Big(d^N(k) \leq (1-\gamma)\mathbb{E}[d^N(k)]\Big) \leq \exp(-\frac{\mathbb{E}[d^n(k)]\gamma^2}{3})$. Taking $\gamma = \sqrt{\frac{3\log N}{\mathbb{E}[d^N(k)]}}$ yields $\mathbb{P}\Big(\frac{N}{d^N(k)} \geq \frac{N}{(1-\gamma)\mathbb{E}[d^N(k)]}\Big) \leq \frac{2}{N}$. By the triangle inequality we have $\left|\sum_{k=1}^N \frac{N}{d^N(k)}\mathbf{1}_{u\in(k-1/N,k/N]} - \frac{1}{d(u)}\right| \leq \left|\sum_{k=1}^N (\frac{N}{d^N(k)} - \mathbb{E}[\frac{N}{d^N(k)}])\mathbf{1}_{u\in(k-1/N,k/N]}\right| + \left|\sum_{k=1}^N \mathbb{E}[\frac{N}{d^N(k)}]\mathbf{1}_{u\in(k-1/N,k/$ We apply Chernoff bound the first summand to show that the difference concentrates around zero, i.e., $\mathbb{P}\Big(\frac{N}{d^N(k)} \leq \frac{N}{(1-\gamma)\mathbb{E}[d^N(k)]}\Big) \geq 1 - 2/N$ and therefore $\frac{N}{d^N(k)} - \frac{N}{\mathbb{E}[d^N(k)]} \leq N(\frac{1}{(1-\gamma)\mathbb{E}[d^N(k)]} - \frac{1}{\mathbb{E}[d^N(k)]}) = \mathcal{O}(\sqrt{\frac{\log N}{N}})$ with probability at least $1 - 2/N$. The case $\frac{N}{d^N(k)} \leq \frac{N}{(1-\gamma)\mathbb{E}[d^N(k)]}$ follows using the same arguments. For the second summand we use the statement of Lemma 23 to obtain an error bound of order $\mathcal{O}(1/N)$. This shows that for large enough N, $\left|\left|\left|\tilde{G}^N - \tilde{G}\right|\right|\right|_{L_2}$ is of order $\mathcal{O}(\sqrt{\frac{\log N}{N}})$.

Implementation

For the simulation, we set the parameters of the load balancing system as follows: The arrival rate is set to $\lambda_L = 1$, the server rate to $\mu_L = 1.1$ for all servers, and the maximal server capacity is $K_L = 10$. For each system size N, we sample a graph that remains fixed for all simulations and obtain the sample mean by averaging over 2000 sampled trajectories for all system sizes. To compute

the solution of the graphon mean field approximation, we discretize the drift by a simple step discretization. To be more precise, let $\gamma \in \mathbb{N}$ be the discretization parameter and $U^\gamma = \{U_i^\gamma\}_{i=1..\gamma}$ be the even partition of the interval $(0, 1]$ with $U_i^\gamma := (i - 1/\gamma, i/\gamma]$. We define by $F_B^\gamma(\boldsymbol{x})_{u,s} := \sum_{i=1}^{\gamma} \mathbb{1}_{U_i^\gamma}(u) F_B(\boldsymbol{x})_{i/\gamma,s}$ the values of the discretized version of F_B. Throughout the numerical experiments, we set $\gamma = 100$. We choose this simple discretization method as it provides good approximation results and low computation times, i.e., for a rudimentary implementation using a NumPy ode solver, we are able to solve the discretization in a few seconds.[3] The Figures 5.2 and 5.3 show the results of our numerical experiments. The first Figure 5.2 shows the approximation accuracy for a single server and different system sizes. We see that already for $N = 40$ the sample mean is very close to the approximation value, which supports the statement of Corollary 17. In the second Figure 5.3, we compare the values for the fraction of servers with at least s jobs for $s = 1, ..., 4$ as described in the figure caption. Remarkably here, even for small system sizes, the values obtained by the sample mean are close to the ones obtained by the approximation. As we observe the by Corollary 17 suggested behavior for single particles inb Figure 5.2, the increased accuracy for $N = 20, 30$ is likely caused by an averaging effect and a well-connected graph instance G^N.

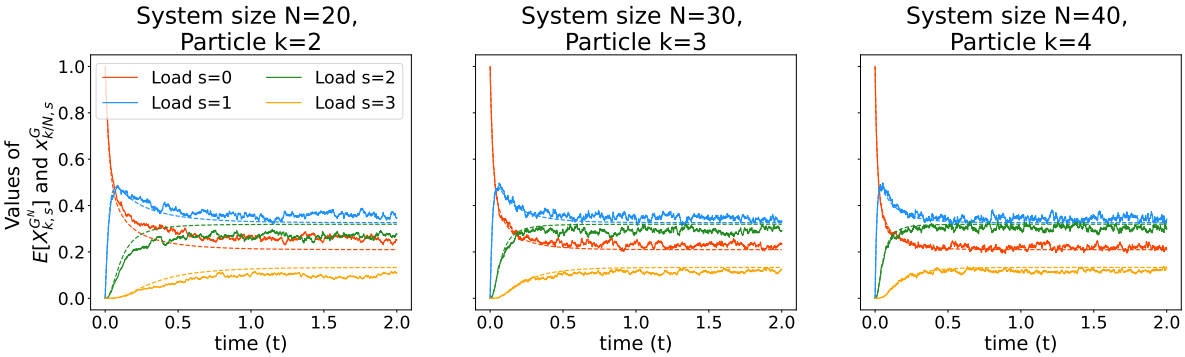


Figure 5.2: The figure shows the evolution of $\mathbb{P}(X_{k,s}^{(N)}(t) = 1) = \mathbb{E}[X_{k,s}^{(N)}(t)]$, the probability for a server-dispatcher pair to have $s = 0, ..., 3$ jobs for $t \in [0, 2]$. In each plot the results for one system size $N = 20, 30, 40$ are displayed. Throughout, the sample mean values are plotted against the values of the graphon mean field approximation.

5.5.2 Heterogeneous Bike-Sharing System

Model

We consider a bike-sharing model following the setup used in [50, 51]. The model consists of $N \in \mathbb{N}$ bike stations, representing the particles in the system. Each station has finite capacity $K_B \in \mathbb{N}$. The system has a fleet of bikes of size $M := \lfloor \alpha N \rfloor$ which, at time $t = 0$, is evenly distributed amongst the stations, i.e., α bikes per station. The evolution of the system depends on the movement of bikes. We differ between the two cases, bikes in the system can either be stationary or in transit between stations. To align with the notations used in the theory part of the paper, we denote

[3] For intricate intensity and graphon functions it might be necessary to refine and tune the discretization approach.

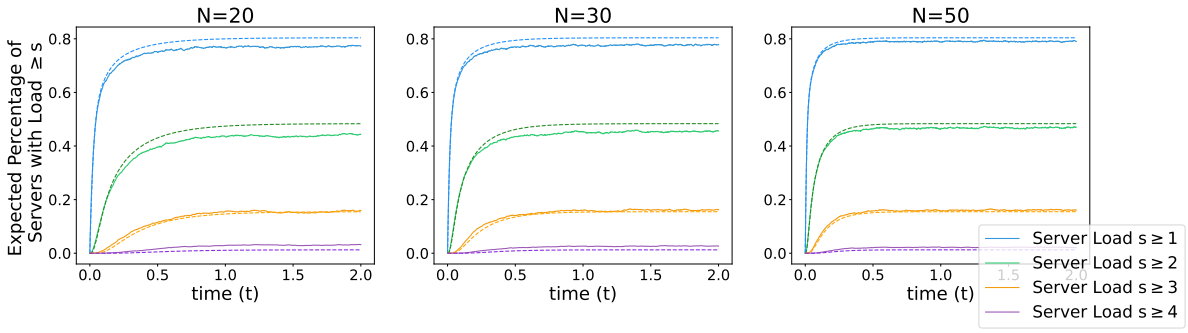


Figure 5.3: The figure compares the values for the fraction of servers with at least s jobs obtained by the sample mean and the approximation. The quantities are calculated as $\mathbb{E}[Q_s(t)] = \mathbb{E}[\sum_{k\in[N]} \frac{1}{N} \sum_{s'\geq s} X^{(N)}_{k,s'}]$ for the sample mean and $q_s(t) = \int_0^1 \sum_{s'\geq s} x^G_{u,s'} du$ for the approximation.

by $S^N_k(t), i \in [N]$ the number of bikes at station i for time $t \geq 0$. For the system, heterogeneity comes from the varying popularity of stations. Hence, we denote by $p_B : (0,1] \mapsto \mathbb{R}_{\geq 0}$ the popularity function. Based on the popularity function p_B, we define the graphon of the bike sharing system by $G_B(u,v) := p_B(v)/\int_0^1 p_B(\nu)d\nu$. For the system of size N, connectivity for the stations is obtained by deterministic sampling as described in Section 5.3.2, i.e., we discretize the graphon based on the system size N. For two stations $k, l \in [N]$ connectivity is therefore defined by $G^{N,B}_{kl} := G_B(k/N, l/N) = p_B(l/N)/\int_0^1 p_B(\nu)d\nu$. Based on the connectivity between stations, it remains to define the dynamics of the system. For a given state $(S_1, ..., S_N) \in [K_B]^N$ the bikes move between stations in the following way:

- Customers arrive at a stations $k \in [N]$ with rate λ_B leading to the transitions

$$S \to S - \boldsymbol{e}^N_k \qquad \text{at rate} \qquad \lambda_B \mathbb{1}_{S_k>0}.$$

- The travel time of bikes between stations is exponentially distributed with rate $\mu_B > 0$. Hence, the for station $k \in [N]$, the arrival rate of bikes is the product of the scaled popularity of station times the traveling bikes $(M - \sum_k S^N_k)$ weighted by the travel time, i.e., $\frac{p_B(k/N)}{\int_0^1 p_B(\nu)d\nu}\big(M - \sum_k S^N_k\big)\mu_B$. This implies the transition

$$S \to S + \boldsymbol{e}^N_k \qquad \text{at rate} \qquad \frac{p_B(k/N)}{\int_0^1 p_B(\nu)d\nu}(M - \sum_k S^N_k)\mu_B \mathbb{1}_{S_k<K_B}.$$

For the above, the notation $\boldsymbol{e}^N_k$ refers to the unit vector of size N having a one at entry k and zeros everywhere else.

Drift & Limiting System

To define the limiting system, we derive the drift implied by the above transitions and the defined graphon G_B. We start by considering the indicator state representation as outlined in Section 5.2, i.e., the drift is a vector of size $K+1$ of $L_2(0,1]$ functions. In contrast to the stochastic rates, for the drift, the sum over particles is replaced by an integral over and the values of G^N replaced by the graphon values. For a state $\boldsymbol{x} = (x_s)_{s=0,\dots,K}$ with $x_s \in L_2(0,1]$, the drift of the system at $(u,s) \in (0,1] \times \mathcal{S} = \{0,\dots,K_B\}$ is defined by

$$F^G_{u,s}(\boldsymbol{x}) := x_{u,s}\left(\frac{p_B(u)}{\int_0^1 p_B(\nu)d\nu}\big(M - \int_0^1 \sum_{s=0}^{K_B} x_{v,s}dv\big)\mu_B \mathbb{1}_{s<K_B} - \lambda_B \mathbb{1}_{s>0}\right).$$

By definition of the system and particularly the graphon, it is immediate that the assumptions of Theorem 15 and Corollary 16 hold, therefore guaranteeing the accuracy of the approximation.

Implementation & Results

For our simulations, we set the popularity function of the bike sharing system to $p_B(v) := 1-0.5v$. The travel rate and customer arrival rate are $\mu_B = 1$ and $\lambda_B = 1$ respectively. For the plots of Figure 5.4 and Figure 5.5 we calculate the sample mean by averaging over 7500 simulations for each system size. For each plot in the first Figure 5.4, the mean field trajectory for single item and state is plotted against the sample mean. Here the horizontal axis represents time t and the vertical axis the probability for the item to be in the state. In the second Figure 5.5, a comparison of the sample mean against the values of the approximation for fixed time and state is shown. As used for the theoretical results, the state of the stochastic system is represented as a step function with constant value $\mathbb{E}[X^{(N)}_{k,s}]$ on the intervals $(k-1/N, k/N]$ for $k = 1..N$. In accordance with our theoretical results, the plot shows that with increasing system size the graphon mean field approximation becomes more accurate and captures the state distribution of the particles well.

5.6 Proofs

This section provides the proof of the main statements of this paper.

5.6.1 Proof of Lemma 14

Proof. Proof. We prove the uniqueness and continuous differentiability of the differential equation (5.7) as well as the Lipschitz properties of the drift. The proof is the consequence of two lemmas:

- We show in Lemma 18 below that the drift is locally Lipschitz continuous in L_2. By [46], this implies the existence of a local solution and the uniqueness of this solution.
- In Lemma 19 we show that the directional derivatives of F^G are well-defined and Lipschitz continuous. This property ensures that the $\boldsymbol{x}^G$ is also continuously differentiable (for a proof of this property in general Banach spaces see for example [46]).

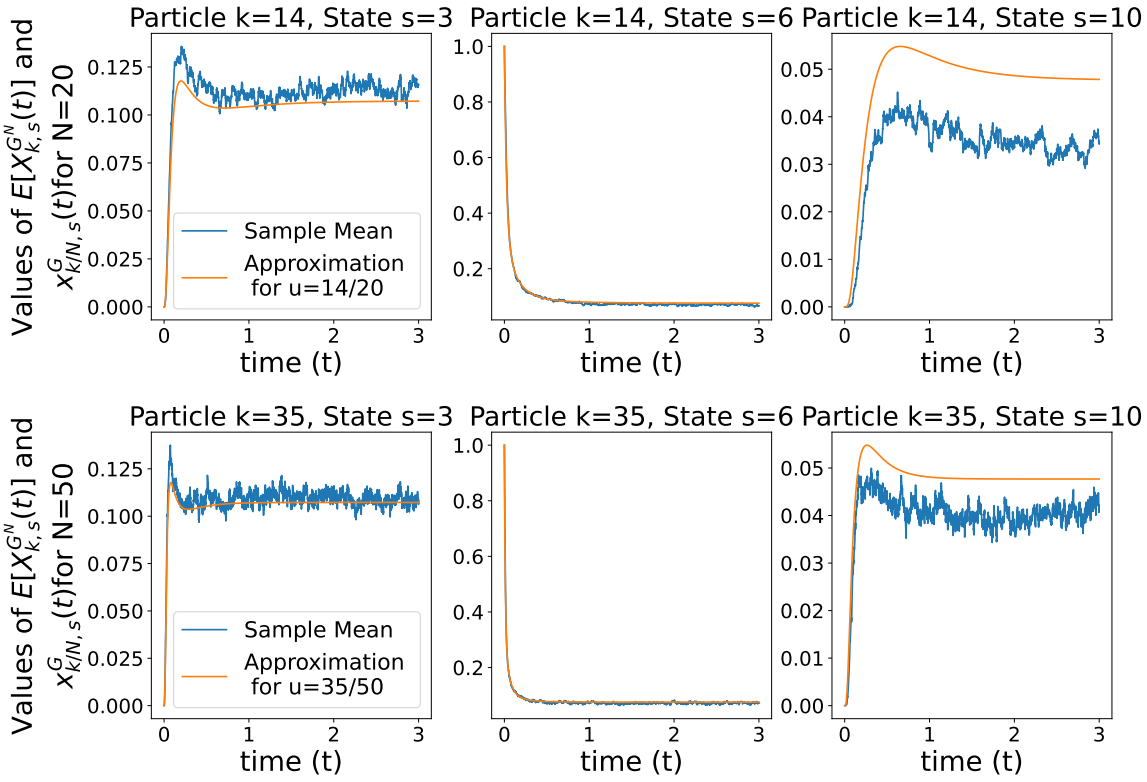


Figure 5.4: The plot shows the sample mean of $\mathbb{E}[X^{(N)}_{k,s}(t)]$ for $N = 20, 50$ and the value of the graphon mean field approximation $\boldsymbol{x}^G_{k/N,s}(t)$ for $t \in [0, 3]$. The plots are generated for the states $s = 3, 6, 10$ and $k = 14, 35$.

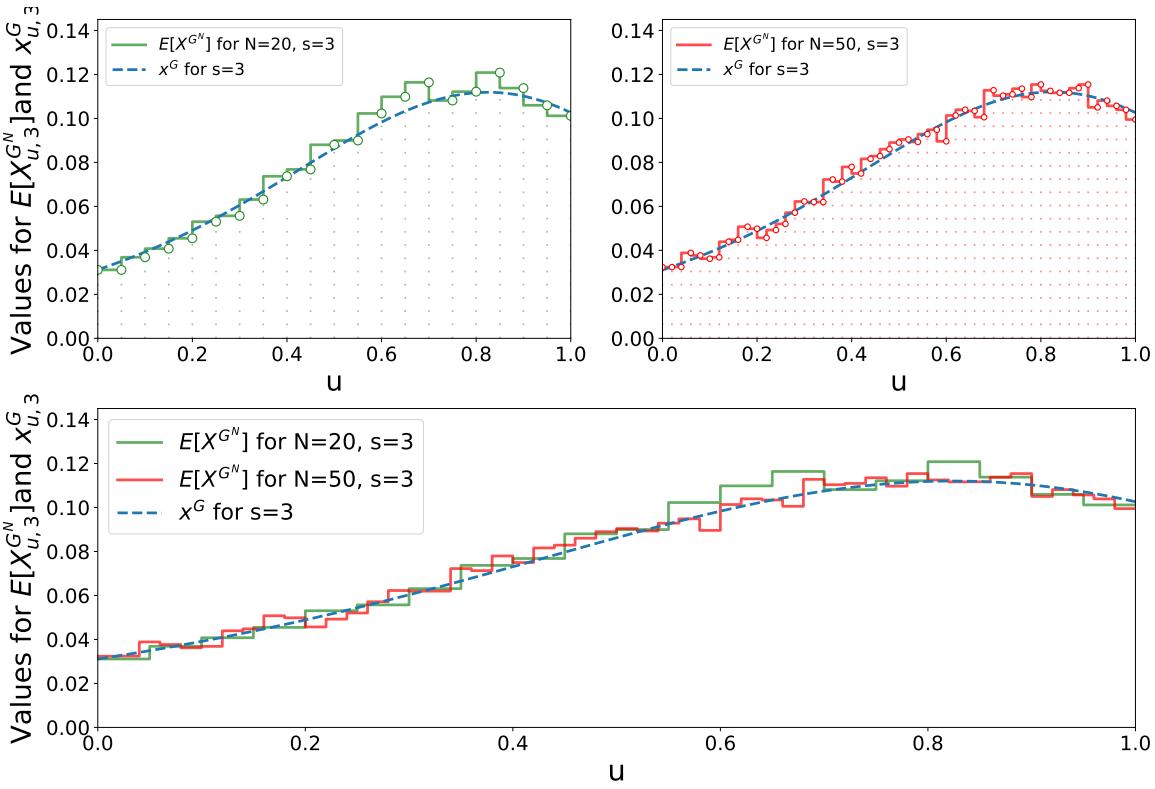


Figure 5.5: For a fixed $t = 2.5$ and state $s = 3$, the plots compare the graphon mean field approximation $\boldsymbol{x}^G$ against sample means for systems of size $N = 20, 50$. As for the main theorem of our results, we represent the values of the stochastic system as step function with constant values on the intervals $(k - 1/N, k/N], k \in [N]$. The figure shows that for increasing system size, the values of the stochastic system indeed approach the trajectory of the deterministic system. In the upper plots, the approximation is plotted against values of the sample mean for one system size $N = 20, 50$ each. In the lower plot, both functions are overlain for better comparison.

□

Lemma 18 (Local Lipschitz Continuity of the Drift)**.** *Let $L_{\boldsymbol{F}^G} = 2(C_R|\mathcal{S}| + 2C_R|\mathcal{S}|^2|||G|||)$ where C_R is the bound on the rate functions, $|||G|||$ the operator norm of the graphon G and $|\mathcal{S}|$ the size of the finite state space. Then for all $\boldsymbol{x}_\cdot = (x_{\cdot,s})_{s\in\mathcal{S}}, \boldsymbol{y}_\cdot = (y_{\cdot,s})_{s\in\mathcal{S}}$ with $x_{\cdot,s}, y_{\cdot,s} \in L_2(0,1]$ and $\|x\|, \|y\| \leq 1$ with $|x_s(u)|, |y_{u,s}| \leq 1$ almost everywhere[4], we have:*

$$\left\|\boldsymbol{F}^G(\boldsymbol{x}) - \boldsymbol{F}^G(\boldsymbol{y})\right\|_{L_2} \leq L_{\boldsymbol{F}^G}\|\boldsymbol{x} - \boldsymbol{y}\|_{L^2}.$$

Proof. Proof. Define $\hat{\boldsymbol{F}}^G_{u,s}(\boldsymbol{x},\boldsymbol{y}) := \boldsymbol{R}^{\text{uni}}_{u,s}\boldsymbol{x}_u + \boldsymbol{x}_u^T\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s}G_{u,v}\boldsymbol{y}_v\,dv$, i.e., we replace $\boldsymbol{x}_v$ by $\boldsymbol{y}_v$ under the integral. Applying the triangle inequality to the L_2 norm gives

$$\begin{aligned}\left\|\boldsymbol{F}^G(\boldsymbol{x}) - \boldsymbol{F}^G(\boldsymbol{y})\right\|_{L_2} &= \left\|\boldsymbol{F}^G(\boldsymbol{x}) - \hat{\boldsymbol{F}}^G(\boldsymbol{x},\boldsymbol{y}) + \hat{\boldsymbol{F}}^G(\boldsymbol{x},\boldsymbol{y}) - F(\boldsymbol{y})\right\|_{L_2}\\ &\leq \left\|\boldsymbol{F}^G(\boldsymbol{x}) - \hat{\boldsymbol{F}}^G(\boldsymbol{x},\boldsymbol{y})\right\|_{L_2} + \left\|\hat{\boldsymbol{F}}^G(\boldsymbol{x},\boldsymbol{y}) - \boldsymbol{F}^G(\boldsymbol{y})\right\|_{L_2}.\end{aligned}$$

Writing out the definitions and using the bounds of the rate vectors / matrices and the bound of the graphon one immediately obtains

$$\begin{aligned}\left\|\boldsymbol{F}^G(\boldsymbol{x}) - \hat{\boldsymbol{F}}^G(\boldsymbol{x},\boldsymbol{y})\right\|_{L_2} &= \sqrt{\sum_{s\in\mathcal{S}}\int_0^1\left(\boldsymbol{R}^{\text{uni}}_{u,s}\boldsymbol{x}_u + \boldsymbol{x}_u^T\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s}G_{u,v}\boldsymbol{x}_v dv - \boldsymbol{R}^{\text{uni}}_{u,s}\boldsymbol{x}_u - \boldsymbol{x}_u^T\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s}G_{u,v}\boldsymbol{y}_v\right.}\\ &= \sqrt{\sum_{s\in\mathcal{S}}\int_0^1\left(\boldsymbol{x}_u^T\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s}G_{u,v}(\boldsymbol{x}_v - \boldsymbol{y}_v)dv\right)^2 du} \leq 2|\mathcal{S}|^3C_R|||G|||\|\boldsymbol{x} - \boldsymbol{y}\|_{L_2}\end{aligned}$$

and

$$\begin{aligned}\left\|\hat{\boldsymbol{F}}^G(\boldsymbol{x},\boldsymbol{y}) - \boldsymbol{F}^G(\boldsymbol{y})\right\|_{L_2} &= \sqrt{\sum_{s\in\mathcal{S}}\int_0^1\left(\boldsymbol{R}^{\text{uni}}_{u,s}(\boldsymbol{x}_u - \boldsymbol{y}_u) + (\boldsymbol{x}_u - \boldsymbol{y}_u)^T\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s}G_{u,v}\boldsymbol{y}_v\,dv\right)^2 du}\\ &\leq \left(2C_R|\mathcal{S}|^2 + 2|\mathcal{S}|^3C_R|||G|||\right)\|\boldsymbol{x} - \boldsymbol{y}\|_{L_2}.\end{aligned}$$

Last, we conclude that $\left\|F^G(\boldsymbol{x}) - F^G(\boldsymbol{y})\right\|_{L_2} \leq (2(C_R|\mathcal{S}|^2 + 2C_R|\mathcal{S}|^3|||G|||))\|\boldsymbol{x} - \boldsymbol{y}\|_{L_2}$, where the $|\mathcal{S}|^2$ and $|\mathcal{S}|^3$ terms come from the vector and matrix notation used for the unilateral and pairwise rates as well as the sum over $\mathcal{S}$.

□

Lemma 19 (Lipschitz Continuous Directional Derivative of $\boldsymbol{F}^G$)**.** *The directional derivative of the drift F^G for $\boldsymbol{x} = (x_s)_{s\in\mathcal{S}}$ with $x_s \in L_2(0,1]$ is given by*

$$D_xF^G_{u,s}(\boldsymbol{x})(\boldsymbol{h}) := \boldsymbol{R}^{uni}_{u,s'}h_u + h_u^T\int_0^1 \boldsymbol{R}^{pair}_{u,v,s}G_{u,v}\boldsymbol{x}_v dv + \boldsymbol{x}_u^T\int_0^1 \boldsymbol{R}^{pair}_{u,v,s}G_{u,v}\boldsymbol{h}_v dv \tag{5.14}$$

[4] Here, *almost everywhere* means that the property holds except on a subset of $(0,1]$ with measure zero.

where $\boldsymbol{R}^{uni}_{u,s}$, $\boldsymbol{R}^{pair}_{u,v,s}$ *and* G *are Lebesgue integrable functions. Furthermore* $D_x F^G_{u,s}(\boldsymbol{x})(h)$ *is Lipschitz-continuous in* x.

Proof. Proof. Recall the definition of $\boldsymbol{F}^G(\boldsymbol{x}) = \boldsymbol{R}^{\text{uni}}_{u,s'}\boldsymbol{x}_u + \boldsymbol{x}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s'} G_{u,v}\boldsymbol{x}_v \, dv$. We define $D_x F^G_{u,s}(\boldsymbol{x})(\boldsymbol{h})$ for $(u,s) \in (0,1]\times\mathcal{S}$ as in Equation (5.14). To see that $D_x\boldsymbol{F}^G(x)(h)$ is a directional derivative of $\boldsymbol{F}^G$ in $\boldsymbol{x} \in L_2(0,1]^{\mathcal{S}}$ in direction $h \in L_2$, we show that it fulfills

$$\lim_{\epsilon \downarrow 0} \frac{1}{\epsilon} \left\| \boldsymbol{F}^G(\boldsymbol{x}+\epsilon\boldsymbol{h}) - \boldsymbol{F}^G(\boldsymbol{x}) - D_x\boldsymbol{F}^G(\boldsymbol{x})(\epsilon\boldsymbol{h}) \right\|_{L_2} = 0.$$

By definition

$$\begin{aligned}
&\left\| \boldsymbol{F}^G(\boldsymbol{x}+\epsilon\boldsymbol{h}) - \boldsymbol{F}^G(x) - D_x\boldsymbol{F}^G(\boldsymbol{x})(\boldsymbol{h}) \right\|^2_{L_2} \\
&= \int_0^1 \sum_s \Bigg(\boldsymbol{R}^{\text{uni}}_{u,s}(\boldsymbol{x}_u + \epsilon\boldsymbol{h}_u) + (\boldsymbol{x}_u + \epsilon\boldsymbol{h}_u)^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}(\boldsymbol{x}_v + \epsilon\boldsymbol{h}_v)\, dv \\
&\quad - \boldsymbol{R}^{\text{uni}}_{u,s'}\boldsymbol{x}_u - \boldsymbol{x}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s'} G_{u,v}\boldsymbol{x}_v \, dv \\
&\quad - \epsilon\boldsymbol{R}^{\text{uni}}\boldsymbol{h}_u - \epsilon\boldsymbol{h}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{x}_v dv - \epsilon\boldsymbol{x}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{h}_v dv \Bigg)^2 du \\
&= \int_0^1 \sum_s \Bigg(\epsilon\boldsymbol{R}^{\text{uni}}_{u,s}\boldsymbol{h}_u + \epsilon\boldsymbol{x}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{h}_v dv + \epsilon\boldsymbol{h}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{x}_v \, dv \\
&\quad + \epsilon^2 \boldsymbol{h}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{h}_v \, dv \\
&\quad - \epsilon(\boldsymbol{R}^{\text{uni}}\boldsymbol{h}_u - \boldsymbol{h}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{x}_v dv - \boldsymbol{x}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{h}_v dv) \Bigg)^2 du \\
&= \int_0^1 \sum_s \left(\epsilon^2 \boldsymbol{h}_u^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s} G_{u,v}\boldsymbol{h}_v dv \right)^2 du.
\end{aligned}$$

Taking dividing by ϵ and taking $\epsilon \to 0$ shows that $D_x\boldsymbol{F}^G(\boldsymbol{x})(\boldsymbol{h})$ is indeed a directional derivative of $\boldsymbol{F}^G$ in $\boldsymbol{x}$. By definition, $D_x\boldsymbol{F}^G_{u,s}(\boldsymbol{x})(\boldsymbol{h})$ is linear in $\boldsymbol{x}$ and as $\boldsymbol{R}^{\text{uni}}_{u,s}$, $\boldsymbol{R}^{\text{pair}}_{u,v,s}$ and $G_{u,v}$ are bounded functions, the Lipschitz continuity in L_2 follows.

□

5.6.2 Proof of Theorem 15

In the following, we present the proof of Theorem 15.

Proof. Proof. Let $\boldsymbol{X}^{(N)}_t$ be the $L_2(0,1]^{|\mathcal{S}|}$ representation of the stochastic system as defined in 5.2.4. For a fixed $t \geq 0$, define

$$\boldsymbol{\nu}^N(\tau) = \mathbb{E}[\boldsymbol{x}^G\big(t-\tau, \boldsymbol{X}^{(N)}(\tau)\big)] \tag{5.15}$$

for which we suppress the dependence of the expectation on the graph G^N and initial state $\boldsymbol{X}^{(N)}(0)$ from here on. Using the definition of $\boldsymbol{\nu}^N$ we rewrite

$$\left\|\mathbb{E}[\boldsymbol{X}^{(N)}(t)] - x^G(t, \boldsymbol{X}^{(N)}(0))\right\|_{L_2} = \sqrt{\sum_{s\in\mathcal{S}} \int_0^1 \left(\mathbb{E}[X^{(N)}_{u,s}(t)] - x^G_{u,s}(t, \boldsymbol{X}^{(N)}(0))\right)^2 du}$$
$$= \sqrt{\sum_{s\in\mathcal{S}} \int_0^1 \left(\nu^N_{u,s}(t) - \nu^N_{u,s}(0)\right)^2 du}. \tag{5.16}$$

To compare the drift of the ODE against the drift of the stochastic process and ultimately bound them, we first want to rewrite $\nu^N_{u,s}(t) - \nu^N_{u,s}(0) = \int_0^t \frac{d}{d\tau}\nu^N_{u,s}(\tau)$, with $\frac{d}{d\tau}\nu^N(\tau)$ being the quantity that fulfills $\boldsymbol{\nu}^N(b) - \boldsymbol{\nu}^N(a) = \int_a^b \frac{d}{d\tau}\boldsymbol{\nu}^N(\theta)d\theta$ for arbitrary $a, b \in [0, t]$ with $a < b$. In Lemma 20 we show that $\frac{d}{d\tau}\boldsymbol{\nu}(\theta)$ exists almost everywhere for $\theta \in (0, t)$ and is almost everywhere equal to

$$\mathbb{E}\left[\sum_{s'} \int_0^1 \left[D_x \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau))\left(\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau)) - \boldsymbol{F}^{G^N}(\boldsymbol{X}^{(N)}(\tau))\right)\right]_{v,s'} dv\right] \tag{5.17}$$
$$+ \mathbb{E}\left[\tilde{R}_1(\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau)))\right] d\tau. \tag{5.18}$$

In the above sum, $D_x \boldsymbol{x}^G(\tau, \boldsymbol{X}^{(N)}(\tau))\left(\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau)) - \boldsymbol{F}^{G^N}(\boldsymbol{X}^{(N)}(\tau))\right)$ is the directional derivative of $\boldsymbol{x}^G$ in its initial condition in direction $\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau)) - \boldsymbol{F}^{G^N}(\boldsymbol{X}^{(N)}(\tau))$. We aim to bound the sum by using the properties of the derivative of $\boldsymbol{x}^G$ and the differences between the drift of the deterministic and stochastic system with respect to the L_2 norm. The technical details to bound the remainder term and the difference between the drifts are moved to the Lemmas 21 and 22. From the application of the latter lemmas, one obtains the bounds

$$C_{D_{x(0)}}\left(\frac{2L_{\boldsymbol{R}^{\text{pair}}} + 16C^2_{\boldsymbol{R}^{\text{pair}}}K_{\boldsymbol{R}^{\text{pair}}}}{N} - \frac{16C^2_{\boldsymbol{R}^{\text{pair}}}K^2_{\boldsymbol{R}^{\text{pair}}}}{N^2} + C^2_{\boldsymbol{R}^{\text{pair}}}|\mathcal{S}|^2 |||G - G^N|||\right)$$

for Equation (5.17) and $C_{\tilde{R}}/N$ for Equation (5.18). It follows by applying the bounds and rearranging terms, that

$$\left\|\mathbb{E}[\boldsymbol{X}^{(N)}(t)] - x^G(t, \boldsymbol{X}^{(N)}(0))\right\|_{L_2} \leq \frac{C_A}{N} + C_B |||G^N - G|||$$

for some finite constants $C_A, C_B > 0$ additionally to the previous bound also depend on t. This concludes the proof.

□

5.6.3 Proof of Corollary 17 and Corollary 16

In this section, we prove the corollaries of Theorem 15. In each case, we use an additional lemma to obtain a bound on $|||G^N - G|||$.

Proof. Proof.[Proof of Corollary 17] By the application of Theorem 15 we have

$$\left\| \mathbb{E}[\boldsymbol{X}^{(N)}(t)] - x^G(t, \boldsymbol{X}^{(N)}(0)) \right\|_{L_2} = \frac{C_A}{N} + C_B \left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|.$$

It therefore remains to bound the L_2 distance $\left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|$ between the graphon G and graph G^N. By application of Lemma 12 with probability at least $1 - \delta$ and for 'large enough' N

$$\left|\!\left|\!\left| G^N - G \right|\!\right|\!\right| \leq \sqrt{\frac{4 \log(2N/\delta)}{N}} + 2\sqrt{\frac{(L_G^2 - K_G^2)}{N^2} + \frac{K_G}{N}} =: \psi_{\delta,G}(N).$$

Defining and substitution $\delta = 2/N$ into the above equation concludes the proof.

□

Proof. Proof.[Proof of Corollary 16] The corollary is a direct implication of Theorem 17 and Lemma 23: due to G^N representing a discretized version of G it directly follows that $\left|\!\left|\!\left| G - G^N \right|\!\right|\!\right|$ is of order $\mathcal{O}(1/N)$, which concludes the proof.

□

5.7 Conclusion

In this paper, we study an approximation for a system of particles interacting on a graph. We show that when the interacting graph converges to a graphon, the underlying behavior of the stochastic system converges to a deterministic limit, which we call the graphon mean field approximation. While this result is similar to other results in the literature – that show that graphon mean field approximations are asymptotically exact in some settings –, our main contribution is to provide precise bounds on the accuracy of this approximation. We showed indeed that the distance between the original finite-N system and the graphon mean field approximation can be bounded by a term $\mathcal{O}(1/N)$ that depends on the number of particles of the system plus a term $\mathcal{O}(\left|\!\left|\!\left| G^N - G \right|\!\right|\!\right|)$ that depends on the distance between the original graph G^N and the graphon G, when measured as a L_2 operator.

This paper aims to be methodological. It shows that it is possible to obtain bounds for a system with a graph structure, and not mere asymptotic convergence results. To keep the presentation reasonable, we intentionally considered a relatively simple model, for instance by restricting our attention to pairwise interactions between nodes or unilateral jumps, and by considering that the rate functions r^N are discretized versions of the limiting rates r. We believe that by doing so, the proof is easier to follow and could then be adapted to more general cases.

The focus of this paper is to study the case of dense graphs that converge to graphons. This implies that the total number of edges per node is of order $\mathcal{O}(N)$. We believe that our methodology could be applicable in the not-so-dense case when the number of edges per node goes to infinity at a sublinear rate. This would probably give bounds that converge more slowly to zero. Complementary to this paper, another interesting question is the case of sparse graphs, where the number of neighbors per node remains bounded when the number of nodes N goes to infinity. Yet, this would require

a fundamentally different approach: studying such a problem is out of the scope of our tools since the graphon mean field approximation is not asymptotically exact for sparse graphs.

5.8 Lemmas

This section contains the most technical lemmas of the paper.

5.8.1 Derivative of ν (Lemma 20)

The following Lemma derives a representation of the derivative of ν, as defined in the proof of Theorem 15, which depends on the derivative of the graphon mean field approximation with respect to the initial condition, the transitions of the stochastic system and the deterministic drift.

Lemma 20. *For $\nu^N(\tau)$ and $\tau \in (0,t)$ we define*

$$\frac{d}{d\tau}\nu^N(\tau) := \lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}[\boldsymbol{x}^G(t-(\theta+r),\boldsymbol{X}^{(N)}(\theta+r))]-\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta))]\Big)d\theta.$$

for which, by construction and continuity properties of $\boldsymbol{x}^G$ and $\boldsymbol{X}^{(N)}$, it holds that for arbitrary $a,b\in[0,T], a<b$, $\int_a^b \frac{d}{d\tau}\nu^N(\tau)=\nu^N(b)-\nu^N(a)$. Furthermore, the right-hand side is almost everywhere equal to

$$\begin{aligned}\mathbb{E}[\sum_{s'}\int_0^1\Big[D_x\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\Big(\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau))-\boldsymbol{F}^{G^N}(\boldsymbol{X}^{(N)}(\tau))\Big)\Big]_{v,s}du]\\ +\mathbb{E}[\tilde{R}_1(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau)))].\end{aligned}$$

which is well defined and bounded due to the properties of the drifts $\boldsymbol{F}^{G^N}$, $\boldsymbol{F}^G$ and the approximation $\boldsymbol{x}^G$. In this context, $\tilde{R}_1$ refers to the summed Taylor remainder term defined by

$$\begin{aligned}\tilde{R}_1\big(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\big) := \sum_{k,s_k,s'_k} R_1(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau)),\boldsymbol{e}^N_{k,s'_k}-\boldsymbol{e}^N_{k,s_k})\\ \times\left[X^{(N)}_{k,s_k}r^{N,uni}_{k,s_k\to s'_k}+X_{k,s_k}\sum_{l\in[N]}\sum_{s_l}r^{N,pair}_{k,l,s_k\to s'_k,s_l}\frac{G^N_{kl}}{N}X_{l,s_l}\right].\end{aligned}$$

Proof. Proof. Recall, that $\boldsymbol{X}^{(N)}$ is in the following handled as a L_2 function as described in Section 5.2.4. We start the proof by rewriting

$$\begin{aligned}
\frac{d}{d\tau}\nu(\tau) &:= \lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}[\boldsymbol{x}^G(t-(\theta+r),\boldsymbol{X}^{(N)}(\theta+r))]-\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta))]\Big)d\theta\\
&= \lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}[\boldsymbol{x}^G(t-(\theta+r),\boldsymbol{X}^{(N)}(\theta+r))]-\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))]\Big)d\theta\\
&\quad+\lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))]-\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta))]\Big)d\theta\\
&= \lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}\Big[\boldsymbol{x}^G(t-\theta,\boldsymbol{x}^G(-r,\boldsymbol{X}^{(N)}(\theta+r)))-\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))\Big]\Big)d\theta\\
&\quad+\lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}\Big[\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))-\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta))\mid\boldsymbol{X}^{(N)}(\tau)]\Big]\Big)d\theta\\
&= \lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\mathbb{E}\Big[\lim_{r\downarrow 0}\frac{1}{r}\Big(\boldsymbol{x}^G(t-\theta,\boldsymbol{x}^G(-r,\boldsymbol{X}^{(N)}(\theta+r)))-\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))\Big)\Big]d\theta\\
&\quad+\lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\mathbb{E}\Big[\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))-\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta))\mid\boldsymbol{X}^{(N)}(\tau)]\Big)\Big]d\theta
\end{aligned}$$

For the first equality, an artificial zero is added by adding and subtracting the term $\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))$ and using dominated convergence in combination with the boundedness of the solution of the differential equation and the stochastic system almost everywhere. For the second equality, we use the tower property to write the difference in the conditional expectation with respect to $\boldsymbol{X}^{(N)}(\tau)$. The last equality, follows by dominated convergence which allows to take the limit inside the outer expectation and by definition of $\boldsymbol{x}^G$ as well as the drift $\boldsymbol{F}^G$ which allows to rewrite $\boldsymbol{x}^G(t-(\theta+r),\boldsymbol{X}^{(N)}(\theta+r)) = \boldsymbol{x}^G(t-\theta,\boldsymbol{x}^G(-r,\boldsymbol{X}^{(N)}(\theta+r)))$. By definition, $\boldsymbol{x}^G$ is continuously differentiable in L_2 with respect to its initial condition. The property follows by a general version of Grönwalls inequality and the Lipschitz properties of the drift, also see [46] for a proof of this property. Using the differentiability in combination with the Lebesgue differentiation theorem for L_p spaces and application of the chain rule we can rewrite

$$\begin{aligned}
&\lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\mathbb{E}\Big[\lim_{r\downarrow 0}\frac{1}{r}\Big(\boldsymbol{x}^G(t-\theta,\boldsymbol{x}^G(-r,\boldsymbol{X}^{(N)}(\theta+r)))-\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta+r))\Big)\Big]d\theta\\
&=\mathbb{E}\Big[D_x\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau))\Big]
\end{aligned}\tag{5.19}$$

where equality holds almost everywhere and $D_x\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau))$ is the directional derivative of $\boldsymbol{x}^G$ in its initial condition with direction $\boldsymbol{F}^G(\boldsymbol{X}^{(N)}(\tau))$. For the second integral, we apply Lebesgue differentiation theorem in combination with the definition of the transitions for the stochastic system. In particular, we use that the probability for a small $d\tau > 0$ the

state of the stochastic system changes to $\boldsymbol{X}^{(N)}(\tau + d\tau) = \boldsymbol{X}^{(N)}(\tau) + \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}$ is given by $d\tau X^{(N)}_{k,s_k}(\tau) r^{N,\text{uni}}_{k,s_k \to s'_k} + d\tau X^{(N)}_{k,s_k}(\tau) \sum_{l\in[N]} \sum_{s_l} r^{N,\text{pair}}_{k,l,s_k \to s'_k, s_l} \frac{G^N_{kl}}{N} X^{(N)}_{l,s_l}(\tau) + d\tau)$. This shows

$$
\begin{aligned}
&\lim_{h\downarrow 0} \int_\tau^{\tau+h} \mathbb{E}\left[\lim_{r\downarrow 0} \frac{1}{r}\Big(\boldsymbol{x}^G(t-\theta, \boldsymbol{X}^{(N)}(\theta + r)) - \boldsymbol{x}^G(t-\theta, \boldsymbol{X}^{(N)}(\theta))\Big)\right] d\theta \\
&= \mathbb{E}\Bigg[\lim_{d\tau\downarrow 0} \frac{1}{d\tau} \sum_{k,s_k,s'_k} \Big(\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau) + \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}) - \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau))\Big) \\
&\qquad \times \left(d\tau X^{(N)}_{k,s_k}(\tau) r^{N,\text{uni}}_{k,s_k\to s'_k} + d\tau X^{(N)}_{k,s_k}(\tau) \sum_{l\in[N]} \sum_{s_l} r^{N,\text{pair}}_{k,l,s_k\to s'_k,s_l} \frac{G^N_{kl}}{N} X^{(N)}_{l,s_l}(\tau) + o(d\tau)\right)\Bigg]
\end{aligned}
$$

where equality holds almost everywhere. Lastly, by continuous differentiability of $\boldsymbol{x}^G$ with respect to its initial condition, we apply the L_2 version of the Taylor expansion, see Lemma 24, for $\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau) + \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k})$ around $\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau))$. Note again, that the equality for the expansion hold almost everywhere. This allows to rewrite

$$
\begin{aligned}
\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau) + \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}) =& \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau)) + \int_0^1 \sum_{s'} \Big[D_x \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau))\Big(\boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}\Big)\Big] \\
&+ R_1(\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau)), \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}).
\end{aligned}
$$

Here, $D_x \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau))\Big(\boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}\Big)$ is the directional derivative of $\boldsymbol{x}^G$ in direction $\boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}$ which should be interpreted as L_2 similar as for the L_2 representation of $\boldsymbol{X}^{(N)}$. By $R_1(\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau)), \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k})$ we denote the Taylor remainder term of first order defined as

$$
\begin{aligned}
&R_1(\boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau)), \boldsymbol{e}^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k}) \\
&= \int_0^1 (1-\delta) \int_0^1 \sum_{s'} \Big[\Big(D_x \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau) + \delta(e^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k})) - D_x \boldsymbol{x}^G(t-\tau, \boldsymbol{X}^{(N)}(\tau))\Big) \\
&\qquad \times (e^N_{k,s'_k} - \boldsymbol{e}^N_{k,s_k})\Big]_{v,s'} \, dv \, d\delta.
\end{aligned}
$$

Adding this into the previous right-hand side and rearranging the terms yields by using the linearity of the directional derivative

$$\mathbb{E}\Bigg[\sum_{k,s_k,s'_k}\Bigg(\int_0^1\sum_{s'}\Big[D_x\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))(\boldsymbol{e}^N_{k,s'_k}-\boldsymbol{e}^N_{k,s_k})\Big]_{v,s'}dv + R_1(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau)),\boldsymbol{e}^N_{k,s'_k}-\boldsymbol{e}^N_{k,s_k})\Bigg)$$
$$\times\left(X^{(N)}_{k,s_k}(\tau)r^{N,\text{uni}}_{k,s_k\to s'_k}+X^{(N)}_{k,s_k}(\tau)\sum_{l\in[N]}\sum_{s_l}r^{N,\text{pair}}_{k,l,s_k\to s'_k,s_l}\frac{G^N_{kl}}{N}X^{(N)}_{l,s_l}(\tau)\right)\Bigg]$$
$$=\mathbb{E}\left[\int_0^1\sum_{s'}\Big[D_x\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\boldsymbol{F}^{G^N}(\boldsymbol{X}^{(N)}(\tau))\Big]_{v,s'}dv\right]+\mathbb{E}\Big[\tilde{R}_1\big(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\big)\Big].$$

with

$$\tilde{R}_1\big(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\big):=\sum_{k,s_k,s'_k}R_1(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau)),\boldsymbol{e}^N_{k,s'_k}-\boldsymbol{e}^N_{k,s_k}) \tag{5.20}$$
$$\times\left(X^{(N)}_{k,s_k}r^{N,\text{uni}}_{k,s_k\to s'_k}+X^{(N)}_{k,s_k}\sum_{l\in[N]}\sum_{s_l}r^{N,\text{pair}}_{k,l,s_k\to s'_k,s_l}\frac{G^N_{kl}}{N}X^{(N)}_{l,s_l}\right).$$

Ultimately, we rewrite the initial integral by using the formulations of Equations (5.19) and (5.20) to

$$\lim_{h\downarrow 0}\frac{1}{h}\int_\tau^{\tau+h}\lim_{r\downarrow 0}\frac{1}{r}\Big(\mathbb{E}[\boldsymbol{x}^G(t-(\theta+r),\boldsymbol{X}^{(N)}(\theta+r))]-\mathbb{E}[\boldsymbol{x}^G(t-\theta,\boldsymbol{X}^{(N)}(\theta))]\Big)d\theta$$
$$=\mathbb{E}[\sum_{s'}\int_0^1\Big[D_x\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau))\Big(F^G(\boldsymbol{X}^{(N)}(\tau))-\boldsymbol{F}^{G^N}(\boldsymbol{X}^{(N)}(\tau))\Big)\Big]_{v,s'}dv\mid G^N,\boldsymbol{X}^{(N)}(0)]$$
$$+\mathbb{E}[\tilde{R}_1(\boldsymbol{x}^G(t-\tau,\boldsymbol{X}^{(N)}(\tau)))\mid G^N,\boldsymbol{X}^{(N)}(0)]$$

where equality holds almost everywhere. This concludes the proof.

□

5.8.2 Bound on the Taylor Remainder Term (Lemma 21)

The subsequent Lemma states an upper bound for the remainder term arising in Lemma 20 which ultimately is used to obtain the bound for Theorem 15.

Lemma 21. *For any attainable state $\boldsymbol{X}\in\mathcal{X}^N$ of the stochastic process and $\tau\geq 0$, the expectation of the Taylor remainder term $\tilde{R}_1\big(x^G_{u,s}(t,\boldsymbol{X})\big)$[5] as defined in Equation (5.20) is of order $\mathcal{O}(1/N)$.*

[5] In order for $\boldsymbol{x}^G_t$ to be properly defined, $\boldsymbol{X}$ should be understood in this context as a $L_2(0,1]^{|\mathcal{S}|}$ function.

Proof. Proof. For a state $\boldsymbol{X} \in \mathcal{X}^N$, recall the definition of Equation (5.20)

$$\tilde{R}_1\big(x^G_{u,s}(\tau, \boldsymbol{X})\big) = \sum_{k\in[N], s_k, s'_k \in \mathcal{S}} R_1(x^G_{u,s}(\tau, .), \boldsymbol{X}, e^{(k,s'_k)} - e^{(k,s_k)}) \\ \times \big(X^{(N)}_{k/N,s_k} r^{N,\text{uni}}_{k,s_k\to s'_k} + X^{(N)}_{k/N,s_k} \sum_{s_l\in\mathcal{S}} \int_0^1 r^{\text{pair}}_{k/N,v,s_k\to s'_k,s_l} G^N_{k/N,v} X^{(N)}_{v,s_l} dv\big)$$

where we use that $r^{N,\text{pair}}_{k,l,s_k\to s'_k,s_l} = r^{\text{pair}}_{k/N,v,s_k\to s'_k,s_l}$ for $v \in (l - 1/N, l]$ and R_1 given by

$$R_1(x^G_{u,s}(\tau, .), \boldsymbol{X}, \boldsymbol{e}^N_{(k,s_k)} - \boldsymbol{e}^N_{(k,s_k)}) = \\ \int_0^1 (1-\delta) \sum_{s'} \int_0^1 \Big[\big(D_x \boldsymbol{x}^G\big(\tau - \delta, \boldsymbol{X} + \delta(\boldsymbol{e}^N_{(k,s_k)} - \boldsymbol{e}^N_{(k,s_k)})\big) - D_x \boldsymbol{x}^G\big(\tau, \boldsymbol{X}\big)\big)(\boldsymbol{e}^N_{(k,s'_k)} - \boldsymbol{e}^N_{(k,s_k)})\Big]_{v,s'} dv\, d\delta.$$

To be clear, as in the proof of Lemma 20, $\big[D_x \boldsymbol{x}^G(t, \boldsymbol{x})(\boldsymbol{y})\big]_{v,s'}$ refers to the directional derivative of $\boldsymbol{x}^G$ in its initial condition in direction $\boldsymbol{y}$ evaluated at (v, s'). By the properties of $\boldsymbol{x}^G$, as stated in Lemma 14, $D_x x^G$ is Lipschitz continuous in with respect to the initial condition. Thus, we can bound

$$\sum_{s'} ||\big(D_x x^G\big(\tau, \boldsymbol{X} + \delta(\boldsymbol{e}^N_{(k,s_k)} - \boldsymbol{e}^N_{(k,s_k)})\big) - D_x x^G_{u,s}\big(\tau, \boldsymbol{X}\big)\big)(\boldsymbol{e}^N_{(k,s'_k)} - \boldsymbol{e}^N_{(k,s_k)})||_{L_2} \\ \leq \sum_{s'} L_{D_{x(0)}x^G} \big\|(\boldsymbol{e}^N_{(k,s_k)} - \boldsymbol{e}^N_{(k,s_k)})\big\|_{L_2}$$

Application of Cauchy-Schwarz then yields

$$R_1(x^G_{u,s}(t, .), \boldsymbol{X}, \boldsymbol{e}^N_{(k,s'_k)} - \boldsymbol{e}^N_{(k,s_k)}) \\ \leq \sum_{k,s_k,s'_k} L_{D_x x^G} \big\|(\boldsymbol{e}^N_{(k,s_k)} - \boldsymbol{e}^N_{(k,s_k)})\big\|^2_{L_2} \Big| \big(X^{(N)}_{k/N,s_k} r^{N,\text{uni}}_{k,s_k\to s'_k} + X^{(N)}_{k/N,s_k} \sum_{s_l\in\mathcal{S}} \int_0^1 r^{\text{pair}}_{k/N,v,s_k\to s'_k,s_l} G^N_{k/N,v} X^{(N)}_{v,s_l} dv\big)\Big|.$$

By the boundedness of the rates and the finite state space, it is immediate to see that terms of the form

$$X^{(N)}_{k/N,s_k} r^{\text{uni}}_{k,s_k\to s'_k} + X^{(N)}_{k/N,s_k} \sum_{s_l\in\mathcal{S}} \int_0^1 r^{\text{pair}}_{k/N,v,s_k\to s'_k,s_l} G^N_{k/N,v} X^{(N)}_{v,s_l} dv$$

are bounded. Additionally, by definition of $e^{(k,s_k)}$ which are non-zero only on a sub-interval of $(0, 1]$ of size $\frac{1}{N}$, $\left\|e^N_{(k,s_k)} - \boldsymbol{e}^N_{(k,s_k)}\right\|^2_{L_2} \leq 2/N$. Lastly, application of the mentioned properties and bounds show that there exists a constant $C_{\tilde{R}} > 0$ such that $\tilde{R}_1\big(x^G_{u,s}(t, \boldsymbol{X})\big) = C_{\tilde{R}}/N = \mathcal{O}(1/N)$.

□

5.8.3 Bound on the Difference of Stochastic and Deterministic Drift (Lemma 22)

Lemma 22 is concerned with bounding the difference between the drift of the stochastic and deterministic system. As stated in the Lemma, the bound depends mainly on the L_2 difference between rates and between the graphon and graph of the stochastic system.

Lemma 22. *Let $D_x x^G(t, \boldsymbol{X})(\boldsymbol{y})$ be the directional derivative of $\boldsymbol{x}^G$ with respect to the initial condition in direction $\boldsymbol{y}$ in L_2. Then,*

$$\int_0^1 \sum_{s'} \Big[D_x x^G(t, \boldsymbol{X}) \Big(\boldsymbol{F}^G\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) - \boldsymbol{F}^{G^N}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) \Big) \Big]_{v,s'} dv$$

is bounded by $6C_{D_{x(0)}} \Big(\frac{2L_{\boldsymbol{R}^{pair}} + 16C^2_{\boldsymbol{R}^{pair}} K_{\boldsymbol{R}^{pair}}}{N} - \frac{16C^2_{\boldsymbol{R}^{pair}} K^2_{\boldsymbol{R}^{pair}}}{N^2} + C^2_{\boldsymbol{R}^{pair}} |\mathcal{S}|^2 |||G - G^N||| \Big)$.

Proof. Proof. Implied by the finite values of $\boldsymbol{X} \in \mathcal{X}^N$

$$\begin{aligned}
&\int_0^1 \sum_{s'} \Big[D_x x^G(t, \boldsymbol{X}) \Big(\boldsymbol{F}^G\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) - \boldsymbol{F}^{G^N}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) \Big) \Big]_{v,s'} dv \\
&\leq C_{N, D_x x^G} \int_0^1 \sum_{s'} \Big(\boldsymbol{F}^G_{v,s'}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) - \boldsymbol{F}^{G^N}_{v,s'}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) \Big)^2 dv.
\end{aligned}$$

where $C_{N, D_x x^G}$ is essentially obtained by taking the maximum over all $\boldsymbol{X} \in \mathcal{X}^N$. To conclude it remains to be shown that $\int_0^1 \Big(F^G_{v,s'}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) - F^{G^N}_{v,s'}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) \Big)^2 dv$ is small. By definition we are looking at

$$\begin{aligned}
&\int_0^1 \sum_{s'} \Big(F^G_{u,s'}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) - F^{G^N}_{u,s'}\big(\boldsymbol{x}^G(t, \boldsymbol{X})\big) \Big)^2 du \\
&= \int_0^1 \sum_{s'} \Big(\boldsymbol{R}^{\text{uni}}_{u,s'} \boldsymbol{x}^G_u(t, \boldsymbol{X}) + \boldsymbol{x}^G_u(t, \boldsymbol{X})^T \int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s'} G_{u,v} \boldsymbol{x}^G_v(t, \boldsymbol{X})\, dv \\
&\quad - \boldsymbol{R}^{N,\text{uni}}_{u,s'} \boldsymbol{x}^G_u(t, \boldsymbol{X}) - \boldsymbol{x}^G_u(t, \boldsymbol{X})^T \int_0^1 \boldsymbol{R}^{N,\text{pair}}_{u,v,s'} G^N_{u,v} \boldsymbol{x}^G_v(t, \boldsymbol{X}) dv \Big)^2 du \\
&\leq 2 \int_0^1 \sum_{s'} \Big((\boldsymbol{R}^{\text{uni}}_{u,s'} - \boldsymbol{R}^{N,\text{uni}}_{u,s'}) \boldsymbol{x}^G_u(t, \boldsymbol{X}) \Big)^2 du \qquad (5.21) \\
&\quad + 2 \int_0^1 \sum_{s'} \Big(\boldsymbol{x}^G_u(t, \boldsymbol{X})^T \big(\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s'} G_{u,v} \boldsymbol{x}^G_v(t, \boldsymbol{X}) dv - \int_0^1 \boldsymbol{R}^{N,\text{pair}}_{u,v,s'} G^N_{u,v} \boldsymbol{x}^G_v(t, \boldsymbol{X}) dv \big) \Big)^2 du \qquad (5.22)
\end{aligned}$$

where the inequality follows as for $f, g \in L_2(0,1]$ it holds that $\|f+g\|^2 \leq 2(\|f\|^2 + \|g\|^2)$. We separately bound the two summands. The application of Lemma 23 directly gives a bound for the first

$$2\int_0^1 \sum_{s'} \Big((\boldsymbol{R}^{\text{uni}}_{u,s'} - \boldsymbol{R}^{N,\text{uni}}_{u,s'}) \boldsymbol{x}^G_u(t,\boldsymbol{X}) \Big)^2 du \leq 2\left(\frac{2L_{\boldsymbol{R}^{\text{uni}}} + 16C^2_{\boldsymbol{R}^{\text{uni}}} K_{\boldsymbol{R}^{\text{uni}}}}{N} - \frac{16C^2_{\boldsymbol{R}^{\text{uni}}} K^2_{\boldsymbol{R}^{\text{uni}}}}{N^2}\right).$$

For the second term, first, we add and subtract the term $\boldsymbol{x}^G_u(t,\boldsymbol{X})^T \int_0^1 \boldsymbol{R}^{N,\text{pair}}_{u,v,s'} G_{u,v} \boldsymbol{x}^G_v(t,\boldsymbol{X}) dv$ and use again the previously applied inequality to get the two terms

$$4\int_0^1 \sum_{s'} \left(\boldsymbol{x}^G_u(t,\boldsymbol{X})^T \Big(\int_0^1 \boldsymbol{R}^{\text{pair}}_{u,v,s'} - \boldsymbol{R}^{N,\text{pair}}_{u,v,s'} \Big) G_{u,v} \boldsymbol{x}^G_v(t,\boldsymbol{X}) dv \right)^2 \text{ and} \tag{i}$$

$$4\int_0^1 \sum_{s'} \left(\boldsymbol{x}^G_u(t,\boldsymbol{X})^T \int_0^1 \boldsymbol{R}^{N,\text{pair}}_{u,v,s'} \big(G_{u,v} - G^N_{u,v} \big) \boldsymbol{x}^G_v(t,\boldsymbol{X}) \big) dv \right)^2. \tag{ii}$$

At its core, it remains to bound the difference of $\boldsymbol{R}^{\text{pair}} - \boldsymbol{R}^{N,\text{pair}}$ rate matrices and the difference $G - G^N$ between the graphon and the graph. Similar to the bound for the unilateral rates, we use Lemma 23 to bound Equation (i), i.e.,

$$\text{(i)} \leq 4\left(\frac{2L_{\boldsymbol{R}^{\text{pair}}} + 16C^2_{\boldsymbol{R}^{\text{pair}}} K_{\boldsymbol{R}^{\text{pair}}}}{N} - \frac{16C^2_{\boldsymbol{R}^{\text{pair}}} K^2_{\boldsymbol{R}^{\text{pair}}}}{N^2}\right).$$

The second Equation (ii) can be bounded by $4C^2_{\boldsymbol{R}^{\text{pair}}} |\mathcal{S}|^2 \big|\!\big| G - G^N \big|\!\big|$. To obtain this bound, we use the bounds $C_{\boldsymbol{R}^{\text{pair}}}$ for $\boldsymbol{R}^{N,\text{pair}}$, $\boldsymbol{R}^{\text{pair}}$, the fact that $\boldsymbol{R}^{\text{pair}}$ is a $\mathcal{S}\times\mathcal{S}$ matrix, and lastly that $\big\|x^G\big\|_{L_2} = 1$. Application of the bounds yields

$$\begin{aligned}
&\sum_{s'} \langle D_{x(0)} x^G_{u,s}(t,\boldsymbol{X})_{s'}, F^G_{s'}\big(\boldsymbol{x}^G(t,\boldsymbol{X})\big) - F^{G^N}_{s'}(\boldsymbol{x}^G(t,\boldsymbol{X}))\rangle_{L_2} \\
&\leq 6C_{D_{x(0)}} \left(\frac{2L_{\boldsymbol{R}^{\text{pair}}} + 16C^2_{\boldsymbol{R}^{\text{pair}}} K_{\boldsymbol{R}^{\text{pair}}}}{N} - \frac{16C^2_{\boldsymbol{R}^{\text{pair}}} K^2_{\boldsymbol{R}^{\text{pair}}}}{N^2} + C^2_{\boldsymbol{R}^{\text{pair}}} |\mathcal{S}|^2 \big|\!\big| G - G^N \big|\!\big| \right)
\end{aligned}$$

which concludes the proof.

□

5.8.4 Bound on the Approximation of a Piecewise Lipschitz Function (Lemma 23)

In several part of the paper, the difference between a piecewise Lipschitz function and a discretized version needs to be quantified. The next lemma states an upper bound depending on the coarseness of the discretization as well as properties induces by the definition of piecewise Lipschitz continuity.

Lemma 23. *Let B be a bounded piecewise Lipschitz function with $K \in \mathbb{N}$ blocks, Lipschitz constant $L_B < \infty$ as defined in 11 and bound $C_B \geq 0$ such that $|B(u,v)| \leq C_B$. Furthermore, let B^N be its*

discretized version defined by $B^N(u,v) = \sum_{i,j=1}^N B(i/N, j/N)\mathbb{1}_{I_i^N}(u)\mathbb{1}_{I_j^N}(v)$. *Assuming that* N *is large enough as in Definition 13, it holds for* $f \in L_2(0,1]$ *with* $\|f\|_{L_2} \leq 1$ *that*

$$\int_0^1 \left| \int_0^1 \big(B(u,v) - B^N(u,v)\big) f(v) dv \right| du \leq \frac{2L_B + 16C_B^2 K}{N} - \frac{16C_B^2 K^2}{N^2}.$$

Note, in the case of B *being Lipschitz continuous, i.e.,* $K = 0$*, the right-hand side reduces to the anticipated bound of* $2L_B/N$.

Proof. Proof. By Cauchy-Schwarz inequality on $L_2(0,1]$ it follows that

$$\begin{aligned}\int_0^1 \big(B(u,v) - B^N(u,v)\big) f(v) dv &\leq \int_0^1 \big(B(u,v) - B^N(u,v)\big)^2 dv \|f\|_{L_2}^2 \\ &\leq \int_0^1 \big(B(u,v) - B^N(u,v)\big)^2 dv.\end{aligned}$$

For the rest of the proof we refer to [8] as the reasoning is identical as for their proof of Theorem 1, staring at Equation (31). The sole adaptation made is to include the case that B can take values outside of $(0,1]$, leading to $2C_B$ as the bound of the difference $\big|B(u,v) - B^N(u,v)\big|$.

□

5.8.5 Taylor Expansion with Remainder for Banach Spaces (Lemma 24)

For completeness, we reformulate the result from [22] pp. 524-525.

Lemma 24. *Suppose* E, F *are real Banach spaces,* $U \subset E$ *an open and nonempty subset and* $f \in \mathcal{C}^n(U,F)$ *(*n*-times continuously differentiable). Given* $x_0 \in U$ *choose* $r > 0$ *such that* $x_0 + B_r \subset U$*, where* B_r *is the open ball in* E *with center* 0 *and radius* r*. Then for all* $\ell \in B_r$ *we have, using the abbreviation* $(\ell)^k = (\ell, ..., \ell)$*,*

$$f(x_0 + \ell) = \sum_{k=0}^n \frac{1}{k!} f^{(k)}(x_0)(\ell)^k + R_n(f, x_0, \ell), \tag{5.23}$$

where the remainder R_n *has the form*

$$R_n(f, x_0, \ell) = \frac{1}{(n-1)!} \int_0^1 (1-\theta)^{n-1} \big[f^{(n)}(x_0 + \theta\ell) - f^{(n)}(x_0)\big](\ell)^n d\theta. \tag{5.24}$$

Part III

Systems with Rapidly Changing Environments

6 Bias and Refinement of Multiscale Mean Field Models

This chapter analyzes the approximation error of the 'average' mean field approximation for a two-timescale model (X, Y), where the slow component X describes a population of interacting particles which is fully coupled with a rapidly changing environment Y. We further derive a bias correction term for the steady-state from which we define a new approximation called the refined 'average' mean field approximation with increase accuracy.

In this chapter, we present the results of our paper

S. Allmeier and N. Gast. "Bias and Refinement of Multiscale Mean Field Models". *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 7:1, 2023, 23:1–23:29. DOI: 10.1145/3579336.

Contents

6.1 Introduction

The mean field approximation finds widespread application when interested in analyzing the macroscopic behavior of large-scale stochastic systems composed of interacting particles. Its assets lie in a reduction of the model complexity, simplified analysis of the system due to absence of stochastic components, and reduction of computation time compared to a stochastic simulation. The mean field approximation can even yield closed form solutions for the steady-state, e.g., for the well known JSQ(d) model [87]. The mean field approximation is generally given by a set of ordinary differential equations which arise from the assumption that, for large system sizes, the evolution of the particles are stochastically independent of another. This idea works well if the number of particles is large and if the particles can be clustered into a few groups with statistically identical behavior [17, 58, 76, 77, 81]. The framework established by Kurtz [76] to derive (weak) convergence results for the stochastic system justifying the use of the mean field approximations finds sustained attention in the literature.

More recently, the authors of [55, 113] showed that for finite system sizes the bias of the mean field approximation is of order $1/N$ when compared to the mean behavior of the system. Here, N is the scaling parameter, which usually refers to the number of homogeneous particles in the system. Additional works such as [54, 59] introduced corrections, called refinement terms, which effectively increase the rate of accuracy of the approximation and therefore the rate of convergence. The most notable term is the first-order bias refinement, since it offers a convincing trade-off between a significant accuracy gain and additional computation cost.

While these classical mean field results hold for a broad class of models, most of the results can not be transferred to systems with more intricate dynamics such as the two-timescale case, studied for instance in [17, 25]. A two-timescale process consists of two coupled components, one evolving slowly compared to the other. The slowly evolving component is often represented by a system of interacting particles, where the state of each particle evolves as a function of the empirical distribution of all particles but also as a function of the state of the fast component, e.g., a fast changing environment. These types of processes and their 'averaged' mean field adaptation have been of interest since the 1960s and became increasingly relevant in the study of modern and complex systems. We refer to [92] for an extensive literature discussion. An important area of application comes from

the field of computer networks. Examples include loss networks [67], large-scale random access networks with interference graphs [37, 39] or storage networks [49]. Another recent field of literature from which we draw inspiration are chemical reaction networks. The works of Kang et al. [72, 73] and Ball et al. [11] establish central limit theorems for large multiscale models motivated by biological and chemical processes. Another application in the field of biology is given by [98] who use the mean field idea to study neural plasticity models.

Contributions The aforementioned papers underline that the mean field idea can be adapted to two-timescale models and prove that the 'average' mean field approximation is asymptotically exact as N goes to infinity. They do not, however, provide theoretical results which guarantee the accuracy or performance bounds of the approximation for finite systems sizes. This chapter aims at filling this gap. We derive accuracy bounds for the 'average' mean field approximation in the transient regime and steady-state which show that its bias is of order $O(1/N)$, N being the scaling parameter of the stochastic system. We further derive bias correction terms for the steady-state, from which we define a new approximation called the refined 'average' mean field approximation, whose bias is of order $O(1/N^2)$. To prove these accuracy bounds, we develop a framework for two-timescale stochastic models whose slow component is comparable to the concept of density dependent population processes as introduced by Kurtz [77]. Based on this representation, we utilize a combination of generator comparison techniques, Poisson equations as well as derivative bounds on the solution of the Poisson equation. This allows us to bound the bias with respect to the scaling parameter N. To take it a step further, we then prove the existence of correction terms which approximate the bias of the 'average' mean field and allow defining the refined 'average' mean field. To support the practical application of the refinements, we provide an algorithmic way to compute the correction terms. This includes methods to numerically solve the Poisson equation and obtain its derivatives. We illustrate the computation of the refinement terms and confirm the accuracy of the obtained bounds by considering a random access CSMA model. Using the example we show that even for relatively small $N \approx 10$ the refined 'average' mean field approximation almost exactly indicates the steady-state of the model.

Methodological Advances & Technical Challenges To obtain our results, we build on the recent line of work on Stein's method [102]. This method allows calculating the distance between two random variables by looking at the distance between the generators of two related systems. Recently, the method reemerged in publications within the stochastic network community, in particular the works of Braverman et al. [30, 31, 33]. In this chapter, we use the Poisson equation idea in two ways. First and foremost, we use a Poisson equation, called the 'fast' Poisson equation, to bound the distance between a function – in this case the drift – and its average version given by the steady-state distribution of the fast process as introduced in Section 6.4.2. This step is integral to our chapter and constitutes the building block to obtain the error bounds and closed form expressions as it allows to analyze the distance between the coupled stochastic process and its decoupled counterpart. The analysis however bears many technical intricacies one needs to overcome. This includes solving the Poisson equation, stating its derivative bounds and deducing computable expressions used to calculate the bias term. Second, for our steady-state results, we make use of another Poisson equation to compare the stochastic system to the equilibrium point of the approximation. The

latter is related to the methods used in [59, 113] but significantly extends the ideas as the derivation of the bias exhibits novel refinement terms which originate from the coupling and correct the error of the 'averaging' method used for the mean field. The closely linked technical challenges include obtaining numerically feasible formulas for the 'new' refinement terms. Here, we utilize the derived solution of the 'fast' Poisson equation to further specify the fluctuations of the stochastic system around the equilibrium point. Carefully analyzing the combination of the two Poisson equations enables us to obtain the new refinement terms. The closed-form bias terms that we obtained from the steady-state analysis are significantly more complicated than the ones of [59] as they further correct the error made by the averaging method.

Applicability & Numerical Difficulties In this chaper, we make use of a CSMA model to demonstrate the applicability of our results. There are several other examples captured by our framework, such as the Michaelis-Menten enzyme model of [73] or the storage network investigated in [49]. For the latter, the authors observe that when looking at the right timescale the loss of the network can be characterized by a local equilibrium which is obtained using the averaging method. The biochemical Michaelis-Menten enzyme model describes the dynamics between three time-varying species, the enzyme, a substrate and a product. Following the description of the model as in [73] and by using the right scaling arguments, the model exhibits two timescales, the fast reacting and state changing enzymes and the slower changing concentrations of the substrate and product. Our framework can be used on both examples to guarantee accuracy results and can be used to compute refined approximations.

To compute the 'average' mean field and refinement terms, one has to overcome numerical difficulties which arise from the averaging method. First, in order to compute the 'average' drift, one needs to compute the steady-state probabilities of the fast system, which might not be available in the closed form as for the CSMA model. For this case we provide computational notes in the appendix which aim to facilitate the computation. Another problem for the refinement term is the need for the first and second derivatives of the 'average' drift. For the CSMA model we used symbolic representation of the transition rates from which we define the drift and its average version. This method allows to numerically compute the derivatives using 'sympy', a Python library for symbolic computation. This method is relatively easy to implement but has a very large computation time. This computation time could be reduced by implementing a faster computation of the derivatives. In fact, to compute the derivative of the matrix K^+ which is closely linked to the solution of the 'fast' Poisson, we provide supplementary computational notes which describe how to obtain an efficient implementation.

Roadmap The chapter is organized as follows. In Section 6.2, we formally introduce the two-timescale model, its corresponding 'average' mean field approximation and make regularity assumptions on the system. In Section 6.3 we state the main results of this chapter. Section 6.3.1 states the results for the transient regime, Section 6.3.2 for the steady-state and Section 6.3.3 justifies the existence of bias correction terms. Section 6.4 holds the proofs of the aforementioned results. In Section 6.5 we apply our theoretical results to the unsaturated random-access network model of [38, 39]. Some technical lemmas and definitions are postponed to the appendix.

Reproducibility The code to reproduce the chapter along with all figures and the implementation of the unsaturated random-access network model is available at https://gitlab.inria.fr/sallmeie/bias-and-refinement-of-multiscale-mean-field-models.

6.2 Stochastic System and Mean Field Approximation

We consider a two-timescale, coupled, continuous time Markov chain $(\boldsymbol{X}_s^{(N)}, \boldsymbol{Y}_s^{(N)})_{s\geq 0}$ parametrized by a scaling factor N, for which we study the behavior as N tends to infinity. As we will see in the examples, N typically represents the number of objects that interact together. This section introduces the precise model and fixes notations.

6.2.1 Model

For a fixed scaling parameter N, the stochastic process $(\boldsymbol{X}_s^{(N)}, \boldsymbol{Y}_s^{(N)})_{s\geq 0}$ is a continuous time Markov chain that evolves in a state-space $\mathcal{X}^{(N)} \times \mathcal{Y}$. The set $\mathcal{Y}$ is finite and does not depend on N. We further require that for all N, the sets $\mathcal{X}^{(N)}$ are subsets of a convex and compact set $\mathcal{X} \subset \mathbb{R}^{d_x}$. In what follows, unless it is ambiguous in the context, we drop the dependence on N to lighten the notations.

This model has two-timescales in the sense that the size order of jumps of $\boldsymbol{Y}_s$ is N-times larger than the ones of $\boldsymbol{X}_s$. More precisely, we assume that there exists a finite number of transitions $(\ell, \boldsymbol{y}') \in \mathcal{T}$ with their corresponding transition rate functions $\alpha_{\ell,\boldsymbol{y}'} \geq 0$, both being independent of N, such that for all possible states $(\boldsymbol{X}_s, \boldsymbol{Y}_s) \in \mathcal{X} \times \mathcal{Y}$:

$$(\boldsymbol{X}_s, \boldsymbol{Y}_s) \text{ jumps to } (\boldsymbol{X}_s + \ell/N, \boldsymbol{y}') \qquad \text{at rate } N\alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_s, \boldsymbol{Y}_s). \tag{6.1}$$

The above defines continuous time Markov chains with discrete state-space whose realizations are Càdlàg, i.e., right continuous with a left limit for every time t. Note that in Equation (6.1), we assume that the transition rates α are defined for all $x \in \mathcal{X}$ (and not just for $x \in \mathcal{X}^{(N)}$).

The notion of slow-fast system comes from the fact that the jumps of the slow component $\mathcal{X}$ are $O(N)$ times smaller than the jumps of the fast component $\mathcal{Y}$ while transition rates are of the same scale. As we will see later, the different timescales imply that for large N, the slow component $\boldsymbol{X}_s$ will 'see' the fast component $\boldsymbol{Y}_s$ as if it is stationary with distribution $(\pi_{\boldsymbol{y}}(\boldsymbol{X}_s))_{\boldsymbol{y}\in\mathcal{Y}}$ which we formally define in Section 6.2.2.

6.2.2 Drift, Average Drift and Mean Field Approximation

The jumps of the stochastic system (6.1) can affect the fast and/or the slow component. In what follows, we construct an approximation that consists (i) in considering that the slow component is not stochastic but evolves deterministically according to its drift (which is its average change), and (ii) in using a time-averaging method that shows the stochastic process $\boldsymbol{Y}$ as being in some stationary state given $\boldsymbol{x}$. This leads us to two definitions:

(i) We call the drift of the slow system (or more concisely the drift) the average change of $\boldsymbol{X}_s$. It is the sum over all possible transitions of the rate of transition multiplied by the changes that

such transitions induce on $\boldsymbol{X}_t$. By the form of the transitions in (6.1), if the process starts in $(\boldsymbol{X}_t, \boldsymbol{Y}_t) = (\boldsymbol{x}, \boldsymbol{y})$, the drift is given by:

$$F(\boldsymbol{x}, \boldsymbol{y}) := \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) \ell \in \mathbb{R}^{d_x}. \tag{6.2}$$

This drift function depends on the state of the fast system $\boldsymbol{y}$.

(ii) For the fast component, we define a transition kernel $K_{\boldsymbol{y},\boldsymbol{y}'}(\boldsymbol{x})$ that is the rate at which the process $\boldsymbol{Y}$ jumps from $\boldsymbol{y}$ to $\boldsymbol{y}' \neq \boldsymbol{y}$ (divided by N), with the usual convention that $K_{\boldsymbol{y},\boldsymbol{y}}(\boldsymbol{x}) = -\sum_{\boldsymbol{y}' \neq \boldsymbol{y}} K_{\boldsymbol{y},\boldsymbol{y}'}(\boldsymbol{x})$:

$$K_{\boldsymbol{y},\boldsymbol{y}'}(\boldsymbol{x}) = \sum_{\ell} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}). \tag{6.3}$$

As $\mathcal{Y}$ is finite, for a fixed $\boldsymbol{x}$, $K(\boldsymbol{x})$ is a matrix that corresponds to the kernel of a continuous time Markov chain. Our assumptions will imply that for all $\boldsymbol{x}$, the process associated with $K(\boldsymbol{x})$ has a unique stationary distribution, which we denote by the vector $\pi(\boldsymbol{x}) = (\pi_{\boldsymbol{y}}(\boldsymbol{x}))_{\boldsymbol{y} \in \mathcal{Y}}$.

Based on the drift (6.2) of the stochastic system, we define its 'average' version $\bar{F}$ by averaging over the stationary distribution of the fast component. That is:

$$\bar{F}(\boldsymbol{x}) := \sum_{y} \pi_y(\boldsymbol{x}) F(\boldsymbol{x}, \boldsymbol{y}).$$

For an initial state $\boldsymbol{x}$ and $t \geq 0$, we call the mean field approximation the solution $\phi_t(\boldsymbol{x})$ of the initial value problem

$$\frac{d}{dt}\phi_t(\boldsymbol{x}) = \bar{F}(\phi_t(\boldsymbol{x})), \qquad \phi_0(\boldsymbol{x}) = \boldsymbol{x}. \tag{6.4}$$

Such an approximation is also called a fluid approximation.

6.2.3 Main assumptions

As we show later, under mild regularity conditions on the transition rate functions α, the mean field approximation captures the dynamics of $\boldsymbol{X}_t$ well and with a decreasing bias of order $1/N$:

$$\mathbb{E}[h(\boldsymbol{X}_t) \mid \boldsymbol{X}_0, \boldsymbol{Y}_0 = \boldsymbol{x}, \boldsymbol{y}] - h(\phi_t(\boldsymbol{x})) \leq \frac{C_h(t)}{N} + o(\frac{1}{N}),$$

for a sufficiently regular h and equality for $t = +\infty$. This holds for any finite t under assumption (A_1)-(A_2) below. It also holds for the steady-state regime $t = +\infty$ under the additional assumption (A_3). For the steady-state we also show that C_h can be computed numerically and use it to propose a refined approximation. To obtain these results for finite time, we will assume that:

(A_1) The set of transitions $\mathcal{T}$ is finite and for all $\ell, \boldsymbol{y}' \in \mathcal{T}, \alpha_{\ell,\boldsymbol{y}'} \in \mathcal{D}^2(\mathcal{X}\times\mathcal{Y})$, where $\mathcal{D}^k(\mathcal{X}\times\mathcal{Y})$ the space of functions from $\mathcal{X}\times\mathcal{Y}$ to $\mathbb{R}$ for which the Hölder norm $\|h\|_{k,1}$ is finite[1].

(A_2) For all $\boldsymbol{x} \in \mathcal{X}$, the matrix $K(\boldsymbol{x})$ defined in Equation (6.3) has a unique irreducible class.

(A_3) The ODE (6.4) has a unique, exponentially stable equilibrium, that we denote by ϕ_∞, i.e., there exist $a, b > 0$ s.t. $\|\phi_t(\boldsymbol{x}) - \phi_\infty\| \leq a\exp(-bt)$ for all $\boldsymbol{x} \in \mathcal{X}$.

Assumptions (A_1) and (A_3) are classical to ensure mean field convergence results. As stated in [59, 113], requiring that the transition rates are twice differentiable which is necessary to guarantee the existence of derivatives for the drift and the differential equation needed for the proofs of the theorems. Assumption (A_3) ensures the existence of a unique equilibrium point to which all trajectories of the differential equation converge. This classical assumption is essentially needed to show that the stationary distribution of the stochastic process converges to a deterministic limit, see [17]. It guarantees the stability of both the ODE and the 'slow' Poisson equation used in the proof.

By Assumption (A_2), we mean that for all $\boldsymbol{x}$ the Markov chain should have a unique subset of states that is irreducible (there can be additional states, but they should all be transient). This assumption is equivalent to assuming the uniqueness of the stationary distribution of the Markov chain induced by the generator matrix $K(\boldsymbol{x})$ which is essential to define the 'averaged' drift and mean field approximation. For a given $\boldsymbol{x}$, the stationary distribution $\pi(\boldsymbol{x})$ will be non-zero for all states that are in the irreducible component ($\pi_{\boldsymbol{y}}(\boldsymbol{x}) > 0$ for such $\boldsymbol{y}$'s) and will be zero for the others ($\pi_{\boldsymbol{y}}(\boldsymbol{x}) = 0$ for all states that are transient for $K(\boldsymbol{x})$). This assumption is slightly more general than assuming that $K(\boldsymbol{x})$ is irreducible because it allows for transient states.

We will show later that the assumptions (A_1) and (A_2) imply that $\bar{F}$ is Lipschitz continuous because they imply that $\pi(\boldsymbol{x})$ is Lipschitz-continuous (see Lemma 28). This implies that the mean field approximation (6.4) is well defined.

6.3 Main Results

This section includes our main results which are threefold. In 6.3.1 we obtain accuracy results for the mean field approximation in the transient regime. In 6.3.2 we obtain comparable results when the stochastic system is in its steady-state. Lastly, in 6.3.3 we introduce a correction term for the steady-state, give accuracy bounds and display closed form expressions of the corrections.

6.3.1 Transient Regime

Our model is a two timescale model, which makes it amenable to be analyzed by time-averaging methods such as the one used in [17, 37, 39, 98]. Such methods guarantee that the stochastic process $\boldsymbol{X}_t$ converges to the solution of the ODE given by (6.4). Yet, most of the papers that use averaging methods do not quantify the rate at which this convergence occurs. Our first result, Theorem 25, states that the difference between the 'average' mean field approximation ϕ derived from the average drift $\bar{F}$ approximates the average behavior of the slow component of the two-timescale system with a

[1] i.e., functions that are k-times differentiable with Lipschitz continuous derivatives (see a more precise definition of this norm in Definition 33 of Appendix 6.7.1)

bias asymptotically bounded by $\bar{C}_h/N$. This result is similar to the one obtained in [54] for classical density dependent process.

Theorem 25. *Consider the two-timescale stochastic system $(\boldsymbol{X}^{(N)}, \boldsymbol{Y}^{(N)})_{t\geq 0}$ as introduced in Section 2 starting at $\boldsymbol{X}_0^{(N)}, \boldsymbol{Y}_0^{(N)} = \boldsymbol{x}, \boldsymbol{y}$. Recall that $\phi(\boldsymbol{x})$ is the solution of the ODE (4) with initial condition $\boldsymbol{x}$. Further, assume (A_1) and (A_2). For any $h \in \mathcal{D}^2(\mathcal{X})$ and $t > 0$, there exists a constant $\bar{C}_h(t)$ such that for all $N \in \mathbb{N}$:*

$$N\Big(\mathbb{E}[h(\boldsymbol{X}_t^{(N)}) \mid \boldsymbol{X}_0^{(N)}, \boldsymbol{Y}_0^{(N)} = \boldsymbol{x}, \boldsymbol{y}] - h\big(\phi_t(\boldsymbol{x})\big)\Big) \leq \bar{C}_h(t). \tag{6.5}$$

Main element of the proofs. The full proof of this theorem is given in Section 6.4.3. It is decomposed into two parts that correspond to the two approximations (scaling and averaging):

- The first is to approximate the infinitesimal generator $Lh(\boldsymbol{X}_t, \boldsymbol{Y}_t)$ (defined in Section 6.4.1) of the stochastic system by a process whose drift is $D_x h(\boldsymbol{X}_t) \cdot F(\boldsymbol{X}_t, \boldsymbol{Y}_t)$. This part uses that the rate functions α and h are differentiable with respect to the slow variable $\boldsymbol{x}$.
- The second is that the actual drift of the mean field approximation is $\bar{F}(\boldsymbol{x}) = \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{x}) f(\boldsymbol{x}, \boldsymbol{y})$ and not the $F(\boldsymbol{x}, \boldsymbol{y})$ obtained from the stochastic system. This leads us to bound a term of the form $\bar{F}(\boldsymbol{x}) - F(\boldsymbol{x}, \boldsymbol{y})$.

The first error term is bounded by using generator techniques similar to the ones used for classical mean field models as in [55]. It is treated in Section 6.4.3. To bound the second term, we use averaging ideas similar to the ones of [17] that are related to how fast the fast timescale converges to its stationary distribution. This is dealt with in Section 6.4.3. □

6.3.2 Steady-State Results

Theorem 25 guarantees that the mean field is a good approximation for any finite time interval. In order to obtain a similar result for the stationary case, an almost necessary condition is that the ODE (6.4) has a unique fixed point to which all trajectories converge [17]. To obtain the equivalent of Theorem 25, we assume that this unique fixed point is exponentially stable, as is classically done to obtain steady-state guarantees [59, 113]. This assumption is summarized in (A_3) and leads to the following result.

Theorem 26. *Assume (A_1) - (A_3), and assume that for all N, the stochastic system has a stationary distribution. Denote by $(\boldsymbol{X}_\infty^{(N)}, \boldsymbol{Y}_\infty^{(N)})$ a pair of variables having this stationary distribution. Then, for $h \in \mathcal{D}^2(\mathcal{X} \times \mathcal{Y})$ there exists a constant C_h such that:*

$$\lim_{N\to\infty} N\left(\mathbb{E}[h(\boldsymbol{X}_\infty^{(N)}, \boldsymbol{Y}_\infty^{(N)})] - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\phi_\infty) h(\phi_\infty, \boldsymbol{y})\right) = C_h.$$

Main elements of the proof. A detailed proof is provided in Section 6.4.4. To prove the result, the first essential step is to generate two Poisson equations:

1. The first is used to study $\mathbb{E}[h(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{X}_\infty) h(\boldsymbol{X}_\infty, \boldsymbol{y})]$, which is the difference between the true expectation and a hypothetical case where $\boldsymbol{Y}_\infty$ would be independent from $\boldsymbol{X}_\infty$ and distributed according to the stationary distribution $\pi(\boldsymbol{X}_\infty)$.

2. The second is used to study $\mathbb{E}[\sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{X}_\infty) h(\boldsymbol{X}_\infty, \boldsymbol{y}) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\phi_\infty) h(\phi_\infty, \boldsymbol{y})]$, which is the error of the 'slow' process compared to the 'average' mean field.

The rest of the proof treats these Poisson equations to establish the constant C_h. This is done by generator comparison techniques and application of derivative bounds for the solutions of the Poisson equation. □

We note that contrary to Theorem 25, Theorem 26 allows for functions h that can depend on both the slow and the fast components: $\boldsymbol{x}$ and $\boldsymbol{y}$. In fact, Theorem 25 would not be true if we allow functions h to depend on $\boldsymbol{Y}_t$ as in Theorem 26, because of the fast transitions of $\boldsymbol{Y}_t$. For the steady-state, we use that $(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)$ starts in steady-state.

In Theorem 27 below, we will show that, for the steady-state and for functions that only depend on $\boldsymbol{x}$, we can go further and propose an almost closed-form expression for the term C_h. This new bias term provides a refined accuracy of order $O(1/N^2)$. To prove this result, we will use that Theorem 26 allows functions h which depend on $\boldsymbol{X}$ and $\boldsymbol{Y}$ to show that this new approximation is $O(1/N^2)$-accurate. We show below that in fact the constant C_h can be computed by a numerical algorithm, and can therefore be used to define a *refined approximation*, similarly to what is done for classical mean field model in [59].

6.3.3 Steady-State Refinement

To obtain a refined approximation, we utilize ideas introduced in [59] and propose an almost-closed form expression for the term C_h of Theorem 26. As we will see later in the proofs, the bias correction term is composed of two distinct components:

1. The first one (terms V and W) is the analogue of the V and W terms of [59] and corresponds to the difference between the stochastic jumps of the slow system versus having a ODE corresponding to the (non-averaged) drift $f(\boldsymbol{x}, \boldsymbol{y})$.

2. The second component (terms T, S and U) corresponds to approximating $\boldsymbol{Y}$ by its stationary distribution $\pi_{\boldsymbol{y}}(\boldsymbol{x})$, and its consequence on the behavior of the slow system $\boldsymbol{X}$.

The proofs of the expression for V and W are essentially those derived in [59]. The second correction component (terms T, S and U) is related to the difference between drift and its average version. It involves studying the intricate and coupled dynamics of $\boldsymbol{X}$ and $\boldsymbol{Y}$ which, to the best of our knowledge, has not been studied and yields novel results.

Theorem 27. *Assume (A_1) - (A_3). Then, there exist vectors V, S, T and matrices W and U that are solutions of linear systems of equations such that for $h \in \mathcal{D}^2(\mathcal{X})$, we define C_h of Theorem 26 as:*

$$C_h = \sum_i \frac{\partial h}{\partial x_i}(\phi_\infty)(V_i + T_i + S_i) + \frac{1}{2} \sum_{i,j} \frac{\partial^2 h}{\partial x_i x_j}(\phi_\infty)\big(W_{i,j} + U_{i,j}\big).$$

Proof. This theorem is a consequence of Proposition 29 that shows the existence of the above constants, combined with Proposition 30 which provides the linear equations satisfied by V and W, and Proposition 31 which provides the equations satisfied by T, U and V. □

We would like to emphasize that the result of Theorem 27 is only valid for functions h that do *not* depend on $\boldsymbol{y}$. This shows that there exists a computable constant C_h such that, in steady-state:

$$\mathbb{E}[h(\boldsymbol{X}_\infty)] = \phi_\infty + \frac{C_h}{N} + o(1/N).$$

Similar to [59], we call $\phi_\infty + \frac{C_h}{N}$ the *refined approximation* of $\mathbb{E}[h(\boldsymbol{X}_\infty)]$. As we see in our numerical experiment in Section 6.5, this constant is computable and can provide a more accurate approximation than the classical mean field.

In fact, combined with Theorem 26, we can show that the $o(1/N)$ term is a $O(1/N^2)$ term. This shows that the refined approximation is $O(1/N^2)$-accurate. Note, for functions h that do depend on $\boldsymbol{y}$, the existence of such a function is guaranteed by Theorem 26 but a closed-form expression is currently out of our reach.

6.4 Proofs

6.4.1 Stochastic Semi-Groups and Generators

Given the stochastic process of Section 6.2.1, we define the stochastic semi-group operator which maps a pair of initial values $(\boldsymbol{x}, \boldsymbol{y})$ and a function h to the expected value of the system at time t. This semi-group ψ_t associates to a function $h : \mathcal{X}^{(N)} \times \mathcal{Y} \to \mathbb{R}$ the function $\psi_t h : \mathcal{X}^{(N)} \times \mathcal{Y} \to \mathbb{R}$ defined as:

$$\psi_t h(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{E}[h(\boldsymbol{X}_t, \boldsymbol{Y}_t) \mid X_0, Y_0 = \boldsymbol{x}, y]. \tag{6.6}$$

Using the right continuity of the slow-fast system, it is easy to verify that ψ_t is indeed a C_0-semi-group (see a more precise definition in Definition 32 in Appendix 6.7.1).

The infinitesimal generator of the stochastic process is the operator L that maps a function $h \in \mathcal{D}^k(\mathcal{X} \times \mathcal{Y})$ to $Lh : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ defined by:

$$Lh(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\ell, \boldsymbol{y}' \in \mathcal{T}} N\alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y})(h(x + \ell/N, \boldsymbol{y}') - h(\boldsymbol{x}, \boldsymbol{y})). \tag{6.7}$$

Note that Lh is obtained by considering average infinitesimal change of the stochastic system starting in $(\boldsymbol{x}, \boldsymbol{y})$, i.e.,

$$Lh(\boldsymbol{x}, \boldsymbol{y}) = \lim_{t \downarrow 0} \big(\psi_t h(\boldsymbol{x}, \boldsymbol{y}) - \psi_0 h(\boldsymbol{x}, \boldsymbol{y})\big)/t = \lim_{t \downarrow 0} \big(\mathbb{E}[h(\boldsymbol{X}_t, \boldsymbol{Y}_t) \mid X_0, Y_0 = \boldsymbol{x}, \boldsymbol{y}] - h(\boldsymbol{x}, \boldsymbol{y})\big)/t.$$

Similarly to the notations of the semi-group and generator of the stochastic process, for a given function $h \in \mathcal{D}^k(\mathcal{X})$, we denote by $\Phi_t h(\boldsymbol{x}) := h(\phi_t(\boldsymbol{x}))$ the C_0-semi-group corresponding to ODE (6.4). For differentiable h, the infinitesimal generator is given by

$$\Lambda h(\boldsymbol{x}) := D_x h(\boldsymbol{x}) \bar{F}(\boldsymbol{x}). \tag{6.8}$$

To strengthen intuition, the time derivative of $h(\phi_t(\boldsymbol{x}))$ can be expressed as

$$\frac{d}{dt} h(\phi_t(\boldsymbol{x})) = D_x h(\phi_t(\boldsymbol{x})) \bar{F}(\phi_t(\boldsymbol{x})) = \Phi_t \Lambda h(\boldsymbol{x}). \tag{6.9}$$

By commuting Λ and Φ_t we have $\Lambda \Phi_t h(\boldsymbol{x}) = D_x (h \circ \phi_t)(\boldsymbol{x}) \bar{F}(\boldsymbol{x})$ which is equal to (6.9) by the results of Appendix 6.7.1. We will use this property to prove the theorems of the following section.

Abuse of Notations and Dependence on the Fast Component The semi-group and generator of the stochastic system are generally defined for functions h in $\mathcal{D}^k(\mathcal{X} \times \mathcal{Y})$, i.e., functions which depend on the slow and fast component whereas the semi-group and generator of the ODE are defined for h in $\mathcal{D}^k(\mathcal{X})$ that do *not* depend on the fast component. What we refer to as abuse of notation is the notation we use for the mapping ψ_t of a function $h \in \mathcal{D}^k(\mathcal{X})$. $\psi_t h$ still depends on the fast component even if h does not (since the evolution $\boldsymbol{X}_t$ depends on the state of $\boldsymbol{Y}_t$ and the initial values $(\boldsymbol{x}, \boldsymbol{y})$):

$$\psi_t h(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{E}[h(\boldsymbol{X}_t) \mid \boldsymbol{X}_0, \boldsymbol{Y}_0 = \boldsymbol{x}, \boldsymbol{y}], \tag{6.10}$$

and

$$Lh(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\ell, \boldsymbol{y}'} N \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) \big(h(\boldsymbol{x} + \ell/N) - h(\boldsymbol{x})\big).$$

In essence, this allows to use the notion of the semi-group and generator to functions of the slow process. For consistency, we do the same for the ODE: for an arbitrary $\boldsymbol{y} \in \mathcal{Y}$, $\phi_t(\boldsymbol{x}, \boldsymbol{y}) := \phi_t(\boldsymbol{x})$ and therefore $\Phi_t h(\boldsymbol{x}, \boldsymbol{y}) = h(\phi_t(\boldsymbol{x}, \boldsymbol{y})) = h(\phi_t(\boldsymbol{x}))$ which is motivated by the abuse of notation in (6.10). This will merely be used in the proofs and allows focusing on integral proof ideas instead of complex notations.

6.4.2 Generator of the Fast Process and Regularity of the Poisson Equation

The generator L describes the changes induced by the transitions of the fast and slow process. In our analysis, it will be useful to analyze the changes due to the jumps in Y only. We denote by L_{fast} the generator of the Markov chain induced by $K(\boldsymbol{x})$ that takes as input a function $h \in \mathcal{D}^k(\mathcal{X} \times \mathcal{Y})$ and associates another function $L_{\text{fast}} h$ defined by:

$$L_{\text{fast}} h(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\boldsymbol{y}'} K_{\boldsymbol{y}, \boldsymbol{y}'}(\boldsymbol{x}) (h(\boldsymbol{x}, \boldsymbol{y}') - h(\boldsymbol{x}, \boldsymbol{y})).$$

Compared to (6.1), L_{fast} is independent of N because the transition rates are rescaled by $1/N$.

Under Assumption (A_2), for all $\boldsymbol{x}$, the matrix $K(\boldsymbol{x})$ characterizes a Markov chain on the state space $\mathcal{Y}$ that has a unique stationary distribution denoted by $\pi(\boldsymbol{x})$. $\pi_y(\boldsymbol{x})$ is the stationary probability of $y \in \mathcal{Y}$ of the Markov chain induced by the kernel $K(\boldsymbol{x})$. Subsequently, we will study the distance between the 'true' stochastic process $\boldsymbol{Y}_t$ and an averaged system where the distribution of $\boldsymbol{Y}_t$ is replaced by the stationary distribution $\pi(\boldsymbol{x})$. To quantify the error made when replacing $\boldsymbol{Y}_t$ with the stationary distribution, we consider the following Poisson equation:

$$h(\boldsymbol{x}, \boldsymbol{y}) - \sum_{y \in \mathcal{Y}} h(\boldsymbol{x}, \boldsymbol{y}) \pi_y(\boldsymbol{x}) = L_{\text{fast}} G_h^{\text{fast}}(\boldsymbol{x}, \boldsymbol{y}). \tag{6.11}$$

For a given function $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$ (n is arbitrary but finite), a function $G_h^{\text{fast}} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$ that satisfies the above equation is called a solution to this Poisson equation. We will have particular interest in $h(\boldsymbol{x}, \boldsymbol{y}) = F(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{d_x}$ namely when the drift $F(\boldsymbol{x}, \boldsymbol{y})$ is compared to its 'average' version $\bar{F}(\boldsymbol{x}) = \sum_y F(\boldsymbol{x}, \boldsymbol{y}) \pi(\boldsymbol{y})$.

The existence of a regular solution to this Poisson equation is guaranteed by the following Lemma 28. Note that the solution of the above Poisson equation is not unique: If G_h^{fast} is a solution, then for any constant $c \in \mathbb{R}^n$, a function $G_h^{\text{fast}} + c$ is also a solution. Later in the proofs of the theorems, when we talk about 'a solution of the Poisson equation', we refer to the solution given in Lemma 28.

Lemma 28. *Assume (A_2). Then for all $\boldsymbol{x} \in \mathcal{X}$:*

1. *The Markov chain corresponding to $K(\boldsymbol{x})$ has a unique stationary distribution that we denote by $\pi(\boldsymbol{x})$. We denote by $\Pi(\boldsymbol{x}) = \mathbb{1}\, \pi^T(\boldsymbol{x})$ the matrix where each line is equal to $\pi(\boldsymbol{x})$.*

2. *The matrix $(K(\boldsymbol{x}) + \Pi(\boldsymbol{x}))$ is invertible and its inverse is a generalized inverse of $K(\boldsymbol{x})$.*

3. *Define $K^+(\boldsymbol{x}) = (K(\boldsymbol{x}) + \Pi(\boldsymbol{x}))^{-1}(I - \Pi(\boldsymbol{x}))$, then, for all functions $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$, $G_h(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\boldsymbol{y}'} K^+_{\boldsymbol{y},\boldsymbol{y}'}(\boldsymbol{x}) h(\boldsymbol{x}, \boldsymbol{y}')$ is a solution of the Poisson Equation (6.11).*

If, in addition, Assumption (A_1) holds, then $K^+(\boldsymbol{x})$ is twice differentiable in $\boldsymbol{x}$.

The proof is provided in Appendix 6.8.1. Note in particular, this result implies that if h is (twice) differentiable in $\boldsymbol{x}$ then the same holds true for G_h^{fast}.

6.4.3 Proof of Theorem 25 - Transient State Proof

The proof of Theorem 25 can be decomposed in two main parts. We first use a generator transformation to show that the slow system is well approximated by a system whose drift is $D_x h(\boldsymbol{X}_t)\, F(\boldsymbol{X}_t, \boldsymbol{Y}_t)$ (6.4.3). This leads us to treat terms of the form $\bar{F}(\boldsymbol{X}_t) - F(\boldsymbol{X}_t, \boldsymbol{Y}_t)$. To study them, we use the solution of the Poisson equation for the fast system (6.11). The second part is the more technical and novel. It is detailed in 6.4.3. Some technical lemmas are postponed to Appendix 6.8.2.

Error due to replacing the stochastic jumps of $\boldsymbol{X}_t$ by the drift

For $h \in \mathcal{D}^2(\mathcal{X})$ recall that the C_0-semi-groups of the stochastic system and of the ODE are defined as

$$\psi_s h(\boldsymbol{x}_0, \boldsymbol{y}_0) = \mathbb{E}[h(\boldsymbol{X}_s) \mid \boldsymbol{X}_0, \boldsymbol{Y}_0 = \boldsymbol{x}_0, \boldsymbol{y}_0] \quad \text{and} \quad \Phi_t h(\boldsymbol{x}_0, \boldsymbol{y}_0) = h(\phi_t(\boldsymbol{x}_0)).$$

We define $\nu_s h(\boldsymbol{x}, \boldsymbol{y}) := \psi_s \Phi_{t-s} h(\boldsymbol{x}, \boldsymbol{y})$ and rewrite

$$\mathbb{E}[h(\boldsymbol{X}_t) - h(\phi_t(\boldsymbol{x})) \mid \boldsymbol{X}_0, \boldsymbol{Y}_0 = \boldsymbol{x}_0, \boldsymbol{y}_0] = \nu_t h(\boldsymbol{x}_0, \boldsymbol{y}_0) - \nu_0 h(\boldsymbol{x}_0, \boldsymbol{y}_0) = \int_0^t \frac{d}{ds} \nu_s h(\boldsymbol{x}_0, \boldsymbol{y}_0) ds. \tag{6.12}$$

To show that the last equation indeed holds true, observe that

$$\begin{aligned} \frac{d}{ds} \nu_s h(\boldsymbol{x}_0, \boldsymbol{y}_0) &= \frac{d}{ds} \psi_s \Phi_{t-s} h(\boldsymbol{x}_0, \boldsymbol{y}_0) \\ &= L \psi_s \Phi_{t-s} h(\boldsymbol{x}_0, \boldsymbol{y}_0) - \psi_s \Lambda \Phi_{t-s} h(\boldsymbol{x}_0, \boldsymbol{y}_0). \end{aligned} \tag{6.13}$$

By regularity assumptions on the transition rates and bounded state space the above equation is finite for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X} \times \mathcal{Y}$ and time $s \geq 0$. The dominated convergence theorem thus justifies the interchange of derivative and integral and validates the last equality. As pointed out in Appendix 6.7.1, ψ_s and L, the stochastic semi-group and its infinitesimal generator, commute, i.e., $L\psi_s = \psi_s L$. Hence,

$$\begin{aligned} (6.13) &= \psi_s (L - \Lambda) \Phi_{t-s} h(\boldsymbol{x}_0, \boldsymbol{y}_0) \\ &= \mathbb{E}[(L - \Lambda) \Phi_{t-s} h(\boldsymbol{X}_s, \boldsymbol{Y}_s) \mid \boldsymbol{X}_0, \boldsymbol{Y}_0 = \boldsymbol{x}_0, \boldsymbol{y}_0], \end{aligned} \tag{6.14}$$

where the last line follows by definition of ψ_s. Let $g(\boldsymbol{x}) = h(\phi_{t-s}(\boldsymbol{x}))$. g is twice differentiable with respect to the initial condition $\boldsymbol{x}$ as it lies in $\mathcal{D}^2(\mathbb{R}^d, \mathbb{R})$ by Lemma 34. By the use of Taylor expansion rewrite Lg as

$$\begin{aligned} Lg(\boldsymbol{x}, \boldsymbol{y}) &= \sum_{\ell, \boldsymbol{y}' \in \mathcal{T}} N \alpha_{\ell, y'}(\boldsymbol{x}, \boldsymbol{y}) (g(\boldsymbol{x} + \frac{\ell}{N}) - g(\boldsymbol{x})) \\ &= \sum_{\ell, \boldsymbol{y}' \in \mathcal{T}} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) D_x g(\boldsymbol{x}) \ell + \frac{1}{N} \sum_{\ell, \boldsymbol{y}' \in \mathcal{T}} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) D_x^2 g(\boldsymbol{x}) \cdot (\ell, \ell) + o(\frac{\|g\|_{2,1} \sup_\ell |\ell|^2}{N}) \\ &= Dg(\boldsymbol{x}) F(\boldsymbol{x}, \boldsymbol{y}) + \frac{1}{N} \sum_{\ell, \boldsymbol{y}' \in \mathcal{T}} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) D_x^2 g(\boldsymbol{x}) \cdot (\ell, \ell) + o(\frac{C_R \|g\|_{2,1} \sup_\ell |\ell|^2}{N}), \end{aligned}$$

with $D^2 g(\boldsymbol{x}) \cdot (\ell, \ell) = \sum_{m,n} \frac{\partial^2 g}{\partial x_m \partial x_n}(\boldsymbol{x}) \ell_m \ell_n$. The convergence depends further on the Hölder norm of $h(\phi(x))$, bounds on the transition rates C_R and jump sizes $\sup|\ell|$. The generator of the

ODE related semi-group, Λ, introduced in Equation (6.8), maps g to $\Lambda g(\boldsymbol{x},\boldsymbol{y}) = D_x g(\boldsymbol{x})\bar{F}(\boldsymbol{x})$. Hence, we have

$$(L-\Lambda)g(\boldsymbol{x},\boldsymbol{y}) = D_x g(\boldsymbol{x})(F(\boldsymbol{x},\boldsymbol{y})-\bar{F}(\boldsymbol{x})) + \frac{1}{N}\sum_{\ell,y'\in\mathcal{T}} \alpha_{\ell,y'}(\boldsymbol{x},\boldsymbol{y}) D_x^2 g(\boldsymbol{x})\cdot(\ell,\ell) + o(1/N).$$

Error due to replacing the drift by the average drift

Next, we have a closer look at the first summand of the right hand side in the above equation. Denote by G_F^{fast} a solution of the Poisson equation (6.11) where the function "h' of (6.11) is set to the drift F. Since $D_x g(\boldsymbol{x})$ does not depend on $\boldsymbol{y}$ we have by definition of the Poisson equation and L_{fast}:

$$D_x g(\boldsymbol{x})(F(\boldsymbol{x},\boldsymbol{y})-\bar{F}(\boldsymbol{x})) = D_x g(\boldsymbol{x}) L_{\text{fast}} G_F^{\text{fast}}(\boldsymbol{x},\boldsymbol{y}) = L_{\text{fast}} D_x g(\boldsymbol{x}) G_F^{\text{fast}}(\boldsymbol{x},\boldsymbol{y}). \tag{6.15}$$

Combining the above with equation (6.14) and plugging everything into the integral of equation (6.12), we get:

$$\begin{aligned}\mathbb{E}[h(\boldsymbol{X}_t) - h(\phi_t(\boldsymbol{x}_0))] = \int_0^t \mathbb{E}[L_{\text{fast}} D_x(h\circ\phi_{t-s})(\boldsymbol{X}_s)\cdot G_F^{\text{fast}}(\boldsymbol{X}_s,\boldsymbol{Y}_s)] \\ + \frac{1}{N}\mathbb{E}[\sum_{\ell,\boldsymbol{y}'\in\mathcal{T}} \alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_s,\boldsymbol{Y}_s) D_x^2(h\circ\phi_{t-s})(\boldsymbol{X}_s)\cdot(\ell,\ell)]ds + o(1/N),\end{aligned} \tag{6.16}$$

where we suppress the conditioning on the initial values $\boldsymbol{X}_0,\boldsymbol{Y}_0 = \boldsymbol{x}_0,\boldsymbol{y}_0$. Let $H_s(\boldsymbol{x},\boldsymbol{y}) := \mathbb{E}[D_x(h\circ\phi_{t-s})(X_s)G_F^{\text{fast}}(X_s,Y_s) \mid \boldsymbol{X}_0,\boldsymbol{Y}_0 = \boldsymbol{x},\boldsymbol{y}]$. Applying Lemma 35 with the function $g_s(\boldsymbol{X}_s,\boldsymbol{Y}_s) = D_x(h\circ\phi_{t-s})(X_s)G_F^{\text{fast}}(X_s,Y_s)$ implies that

$$\begin{aligned}0 = \frac{1}{N}H_t(\boldsymbol{x},\boldsymbol{y}) - \frac{1}{N}H_0(\boldsymbol{x},\boldsymbol{y}) - \int_0^t \mathbb{E}[\frac{1}{N}D_x\big(D_x h(\phi_{t-s}(\boldsymbol{X}_s))\cdot\bar{F}(\phi_{t-s}(\boldsymbol{X}_s))\big)\cdot G_F^{\text{fast}}(\boldsymbol{X}_s,\boldsymbol{Y}_s)]ds \\ + \mathbb{E}[\frac{1}{N}L D_x(h\circ\phi_{t-s})(X_s)G_F^{\text{fast}}(X_s,Y_s)]ds.\end{aligned} \tag{6.17}$$

As h and ϕ_{t-s} are twice continuously differentiable, by compactness of $\mathcal{X}$, and as G_F^{fast} is finite, $D_x\big(D_x h(\phi_{t-s}(\boldsymbol{X}_s))\bar{F}(\phi_{t-s}(\boldsymbol{X}_s))\big)G_F^{\text{fast}}(\boldsymbol{X}_s,\boldsymbol{Y}_s)$ is bounded. Moreover, by Lemma 36 we have

$$\begin{aligned}&(L_{\text{fast}} - \frac{1}{N}L)D_x(h\circ\phi_{t-s})(X_s)G_F^{\text{fast}}(X_s,Y_s) \\ &= \frac{1}{N}\sum_{\ell,\boldsymbol{y}'} \alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_s,\boldsymbol{Y}_s)\ell\, D_x(D_x(h\circ\phi_{t-s})(\boldsymbol{X}_s)G_F^{\text{fast}}(\boldsymbol{X}_s,\boldsymbol{y}')) + o(1/N).\end{aligned}$$

Define

$$C_h(t) := \int_0^t \mathbb{E}[\sum_{\ell,\boldsymbol{y}'\in\mathcal{T}} \alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_s,\boldsymbol{Y}_s) D_x^2 h(\phi_{t-s}(\boldsymbol{X}_s))\cdot(\ell,\ell)]ds$$
$$+ \int_0^t \mathbb{E}[D_x\big(D_x h(\phi_{t-s}(\boldsymbol{X}_s))\cdot \bar{F}(\phi_{t-s}(\boldsymbol{X}_s))\big)\cdot G_F^{\text{fast}}(\boldsymbol{X}_s,\boldsymbol{Y}_s)]ds$$
$$+ \int_0^t \mathbb{E}[\sum_{\ell,\boldsymbol{y}'} \alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_s,\boldsymbol{Y}_s)\ell\, D_x(D_x(h\circ\phi_{t-s})(\boldsymbol{X}_s) G_F^{\text{fast}}(\boldsymbol{X}_s,\boldsymbol{y}'))]ds + \mathbb{E}[H_t(\boldsymbol{X}_t,\boldsymbol{Y}_t)] - H_0(\boldsymbol{x},\boldsymbol{y}),$$

which lets us rewrite

$$N(\mathbb{E}[h(\boldsymbol{X}_t)] - h(\phi_t(\boldsymbol{x}))) = C_h(t) + o(1).$$

By assumption, all the terms of $C_h(t)$ are bounded. Therefore, the right-hand side of the equation is bounded by a quantity we call $\bar{C}_h(t)$.

6.4.4 Proof of Theorem 26 - Steady-State Proof

Adding and subtracting $\mathbb{E}[\sum_{\boldsymbol{y}} \pi_y(\boldsymbol{X}_\infty)h(\boldsymbol{X}_\infty,\boldsymbol{y})]$ to $\mathbb{E}[h(\boldsymbol{X}_\infty,\boldsymbol{Y}_\infty) - \sum_{\boldsymbol{y}} \pi_y(\phi_\infty)h(\phi_\infty,\boldsymbol{y})]$ yields:

$$\mathbb{E}[h(\boldsymbol{X}_\infty,\boldsymbol{Y}_\infty) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\phi_\infty)h(\phi_\infty,\boldsymbol{y})] = \mathbb{E}[h(\boldsymbol{X}_\infty,\boldsymbol{Y}_\infty) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{X}_\infty)h(\boldsymbol{X}_\infty,\boldsymbol{y})] \quad \text{(A)}$$
$$+ \mathbb{E}[\sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{X}_\infty)h(\boldsymbol{X}_\infty,\boldsymbol{y}) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\phi_\infty)h(\phi_\infty,\boldsymbol{y})]. \quad \text{(B)}$$

Treating the two terms (A) and (B) separately, we define G_h^{fast} and G_h^{slow} as the solutions to the Poisson equations

$$L_{\text{fast}} G_h^{\text{fast}}(\boldsymbol{x},\boldsymbol{y}) = h(\boldsymbol{x},\boldsymbol{y}) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{x})h(\boldsymbol{x},\boldsymbol{y}), \qquad \text{('fast' Poisson Equation)}$$
$$\Lambda G_h^{\text{slow}}(\boldsymbol{x}) = \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\boldsymbol{x})h(\boldsymbol{x},\boldsymbol{y}) - \sum_{\boldsymbol{y}} \pi_{\boldsymbol{y}}(\phi_\infty)h(\phi_\infty,\boldsymbol{y}). \qquad \text{('slow' Poisson Equation)}$$

Recall that the existence and regularity properties of the function G_h^{fast} are investigated in Lemma 28. For G_h^{slow}, it is known (e.g., [55]) that the exponential stability of the unique fixed point (A_3) along with the smoothness of F guarantees that $G_h^{\text{slow}} \in D^2(\mathcal{X})$.

Next we use that taking the expectation of the generator L applied to an arbitrary function g over the stationary distribution of $(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)$ is zero, i.e., $\mathbb{E}[Lg(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] = 0$. By using Lemma 36, it follows

$$\begin{aligned}(\text{A}) &= \mathbb{E}[L_{\text{fast}} G_h^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) - \frac{1}{N} L G_h^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] \\ &= \mathbb{E}[\frac{1}{N} \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) \ell \, D_x G_h^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{y}') + o(1/N)].\end{aligned}$$

For the second term, we apply the 'slow' Poisson Equation equation and subtract the term $\mathbb{E}[L G_h^{\text{slow}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] = 0$ which yields

$$(\text{B}) = \mathbb{E}[\Lambda G_h^{\text{slow}}(\boldsymbol{X}_\infty) - L G_h^{\text{slow}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)]. \tag{6.18}$$

Note that even if G_h^{slow} only depends on $\boldsymbol{x}$, the function $L G_h^{\text{slow}}$ does depend on $\boldsymbol{x}$ and $\boldsymbol{y}$ because the rate functions that appear in the generator L do depend on $\boldsymbol{x}$ and $\boldsymbol{y}$. By using the Taylor expansion, $L G_h^{\text{slow}}$ equals

$$\begin{aligned}L G_h^{\text{slow}}(\boldsymbol{x}, \boldsymbol{y}) &= \sum_{\ell, \boldsymbol{y}'} N \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) \big(G_h^{\text{slow}}(x + \frac{\ell}{N}) - G_h^{\text{slow}}(\boldsymbol{x}) \big) \\ &= \sum_{\ell, \boldsymbol{y}'} N \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) D_x G_h^{\text{slow}}(\boldsymbol{x}) \frac{\ell}{N} \\ &\quad + \frac{1}{2} \sum_{\ell, \boldsymbol{y}'} N \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) D_x^2 G_h^{\text{slow}}(\boldsymbol{x}) \frac{(\ell, \ell)}{N^2} + o(\frac{\|\ell\|^2}{N}).\end{aligned}$$

By application of the definition of $F(\boldsymbol{x}, \boldsymbol{y})$, the first term of the right hand side of this equation is equal to $F(\boldsymbol{x}, \boldsymbol{y}) D_x G_h^{\text{slow}}(\boldsymbol{x})$. Moreover, by definition, $\Lambda G_h^{\text{slow}}(\boldsymbol{X}_\infty) = D_x G_h^{\text{slow}}(\boldsymbol{x}) \bar{F}(\boldsymbol{x})$. This shows that

$$\begin{aligned}&\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(\bar{F}(\boldsymbol{X}_\infty) - F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)) \\ &\quad + \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) \big(\frac{1}{2} D_x^2 G_h^{\text{slow}}(\boldsymbol{X}_\infty) \cdot \frac{(\ell, \ell)}{N} \big) + o(\frac{\|\ell\|}{N^2})].\end{aligned}$$

As G_h^{slow} has a bounded second derivative, the second term is of order $O(1/N)$. However, to bound the first summand we need to investigate $\bar{F}(\boldsymbol{x}) - F(\boldsymbol{x}, \boldsymbol{y}) = \sum_y \pi_{\boldsymbol{y}'}(\boldsymbol{x}) F(\boldsymbol{x}, \boldsymbol{y}') - F(\boldsymbol{x}, \boldsymbol{y})$. Falling back on ('fast' Poisson Equation) allows two write

$$\begin{aligned}&D_x G_h^{\text{slow}}(\boldsymbol{x})(\bar{F}(\boldsymbol{x}) - F(\boldsymbol{x}, \boldsymbol{y})) \\ &\quad = -D_x G_h^{\text{slow}}(\boldsymbol{x}) L_{\text{fast}} G_F^{\text{fast}}(\boldsymbol{x}, \boldsymbol{y}) = -L_{\text{fast}} D_x G_h^{\text{slow}}(\boldsymbol{x}) G_F^{\text{fast}}(\boldsymbol{x}, \boldsymbol{y}).\end{aligned}$$

The last equality holds due to the definition of L_{fast}, and $D_x G_h^{\text{slow}}(\boldsymbol{x})$ which depends only on $\boldsymbol{x}$. By adding the zero term $\mathbb{E}[\frac{1}{N} L D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty) G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)]$ and using Lemma 36. We see that

$$\begin{aligned}
&\mathbb{E}[-L_{\text{fast}} D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty) G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) + \frac{1}{N} L D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty) G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] \\
&= \mathbb{E}[\frac{1}{N} \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, y'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) \ell \, D_x (D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty) G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{y}')) + o(1/N)].
\end{aligned}$$

Put together, the error term of (B) is

$$\begin{aligned}
\mathbb{E}\Big[\frac{1}{N} \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) \ell \, D_x (D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty) G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{y}')) \\
+ \frac{1}{2N} \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) D_x^2 G_h^{\text{slow}}(\boldsymbol{X}_\infty)(\ell, \ell)\Big].
\end{aligned} \tag{6.19}$$

By defining C_h as the sum of the residual term $\mathbb{E}[\sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) \ell \, D_x G_h^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{y}')]$ for (A) and (6.19) the by N scaled, the proof concludes.

6.4.5 Proof of Theorem 27 (Refinement Theorem, and Closed form Expressions)

In this section we first decompose the constant C_h in two terms in Proposition 29. We then study these two terms in Proposition 30 and 31 where we obtain the closed form expressions for the correction terms which allow to numerically obtain the corrections.

We define two functions $Q(x, y)$ and $J_h(x, y)$ (the latter is defined for a function $h \in \mathcal{D}^2(\mathcal{X})$):

$$\begin{aligned}
Q(\boldsymbol{x}, \boldsymbol{y}) &= \sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) \ell \ell^T, \\
J_h(\boldsymbol{x}, \boldsymbol{y}) &= -\sum_{\ell, \boldsymbol{y}'} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y}) D_x \big(D_x G_h^{\text{slow}}(\boldsymbol{x}) G_F^{\text{fast}}(\boldsymbol{x}, \boldsymbol{y}')\big) \ell,
\end{aligned}$$

and we denote by $\bar{J}_h(x) = \sum_y \pi_y(x) J(x, y)$ and $\bar{Q}(x) = \sum_y \pi_y(x) Q(x, y)$ their "average" version. The following proposition holds.

Proposition 29. *Assume (A1)–(A3). Then, for any $h \in \mathcal{D}^2(\mathcal{X})$, we have:*

$$N\big(\mathbb{E}[h(\boldsymbol{X}_\infty)] - h(\phi_\infty)\big) = \frac{1}{2} D_x^2 G_h^{slow}(\phi_\infty) \bar{Q}(\phi_\infty) + \bar{J}_h(\phi_\infty) + o(1). \tag{6.20}$$

Proof. Let $h \in \mathcal{D}^2(\mathcal{X})$ and let us denote G_h^{slow} the solution of ('slow' Poisson Equation). As h does not depend on y, this function is such that for any x: $h(x) - h(\phi_\infty) = \Lambda G_h^{\text{slow}}(x) =$

$D_x G_h^{\text{slow}}(x)\bar{F}(x)$. The following steps are similar to the ones for (B) in the proof of Theorem 26. Hence, applying this for $\boldsymbol{X}_\infty$ and taking the expectation, we get:

$$N\mathbb{E}[h(\boldsymbol{X}_\infty) - h(\phi_\infty)] = N\mathbb{E}[\Lambda G_h^{\text{slow}}(\boldsymbol{X}_\infty)] = N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)\bar{F}(\boldsymbol{X}_\infty)].$$

Recall that for any bounded function $g \in \mathcal{D}(\mathcal{X}\times\mathcal{Y})$, $\mathbb{E}[Lg(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] = 0$. Hence,

$$N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)\bar{F}(\boldsymbol{X}_\infty) - LG_h^{\text{slow}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)].$$

Similarly to what we do to prove Theorem 26, we plug the definition of L in the above equation and use a Taylor expansion to show that this equals

$$\begin{aligned}
&N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)\bar{F}(\boldsymbol{X}_\infty) - \sum_{\ell,\boldsymbol{y}'} N\alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)\big(G_h^{\text{slow}}(\boldsymbol{X}_\infty + \frac{\ell}{N}) - G_h^{\text{slow}}(\boldsymbol{X}_\infty)\big)] \\
&= N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)\bar{F}(\boldsymbol{X}_\infty) - \sum_{\ell,\boldsymbol{y}'} N\alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)\Big(D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)\frac{\ell}{N} \\
&\qquad\qquad + \frac{1}{2}D_x^2 G_h^{\text{slow}}(\boldsymbol{X}_\infty)\frac{(\ell,\ell)}{N^2} + o(\frac{\|\ell\|}{N^2})\Big)] \\
&= N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(\bar{F}(\boldsymbol{X}_\infty) - F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)) - \frac{1}{2N}D_x^2 G_h^{\text{slow}}(\boldsymbol{X}_\infty)Q(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] + o(1).
\end{aligned} \tag{6.21}$$

The rest of the proof follows using

- Lemma 37 to show $N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(\bar{F}(\boldsymbol{X}_\infty) - F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty))] = \bar{J}(\phi_\infty) + o(1)$;
- Theorem 26 which states $\mathbb{E}[\frac{1}{2}D_x^2 G_h^{\text{slow}}(\boldsymbol{X}_\infty)Q(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] = \frac{1}{2}D_x^2 G_h^{\text{slow}}(\phi_\infty)\bar{Q}(\phi_\infty) + o(1)$.

Applying the above equations to (6.21) implies the statement of Proposition 29. □

In the rest of the section, we show that the quantity $\frac{1}{2}D_x^2 G_h^{\text{slow}}(\phi_\infty)\bar{Q}(\phi_\infty) + \bar{J}(\phi_\infty)$ can be easily computed numerically by solving linear systems of equations. As shown in Proposition 30 and 31, we obtain five different correction terms:

- The first two $V \in \mathbb{R}^{d_x}, W \in \mathbb{R}^{d_x\times d_x}$ are closely related to the ones obtained in [59][Theorem 3.1] and derived from the term $\frac{1}{2}D_x^2 G_h^{\text{slow}}(\phi_\infty)\bar{Q}(\phi_\infty)$. This term is essentially identical to the refinement term of [59].
- In Proposition 31 we derive three other refinement terms $S, T \in \mathbb{R}^{d_x}$ and $U \in \mathbb{R}^{d_x\times d_x}$ which give closed form descriptions of $\bar{J}(\phi_\infty)$. These terms are novel and take into account the difference of the average drift $\bar{F}$ and the actual drift F obtained from the stochastic system. The proof is based on exact expressions for the Poisson equations $G^{\text{fast}}, G^{\text{slow}}$ and relies on Lemma 28.

Correction terms V and W

To obtain the first result, let A and B be the Jacobian and Hessian matrix of the 'average' drift, i.e.,

$$A_{i,j} = \frac{\partial \bar{F}_i}{\partial x_j}(\phi_\infty) \quad \text{and} \quad B_{i,k_1,k_2} = \frac{\partial^2 \bar{F}_i}{\partial x_{k_1} x_{k_2}}(\phi_\infty). \tag{6.22}$$

By the exponential stability of the fixed point (Assumption (A_3)), the matrix A is invertible and the Lyapunov equation $AW + WA^T + \bar{Q} = 0$ has a unique solution. We denote by W its solution and define the vector V as

$$V_i = -\frac{1}{2} \sum_j (A^{-1})_{i,j} \sum_{k_1,k_2} B_{j,k_1,k_2} W_{k_1,k_2}.$$

Proposition 30. *Assume (A_1) - (A_3) and $h \in \mathcal{D}^2(\mathcal{X})$. Then:*

$$\frac{1}{2} D_x^2 G_h^{slow}(\phi_\infty) \bar{Q}(\phi_\infty) = \sum_i \frac{\partial h}{\partial x_i} V_i + \frac{1}{2} \sum_{i,j} \frac{\partial^2 h}{\partial x_i x_j} W_{i,j}\,.$$

Proof. The proposition is a direct consequence of the results of [59][Theorem 3.1] which we apply to the function $\bar{F}(x) = \sum_y \pi_y(x) F(x,y)$. □

Correction terms T, S, U

Proposition 31. *Assume (A_1) - (A_3) and $h \in \mathcal{D}^2(\mathcal{X})$. The closed form solution of $\bar{J}$ at the equilibrium point ϕ_∞ is given by*

$$\bar{J}_h(\phi_\infty) = \sum_i \frac{\partial h}{\partial x_i}(\phi_\infty)(T_i + S_i) + \sum_{i,j} \frac{\partial^2 h}{\partial x_i \partial x_j}(\phi_\infty) U_{i,j}.$$

U is the unique solution to the Sylvester equation

$$AX + XA^T = -O, \qquad \textit{with}$$
$$O = \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) K^+_{y',:}(\phi_\infty) F(\phi_\infty, :) \ell^T,$$

where A, B are the Jacobian, Hessian of the average drift as defined in (6.22) and K^+ as given in Lemma 28. T, S are defined by

$$T_i := \sum_j A^{-1}_{i,j} \sum_{k_1,k_2} B_{j,k_1,k_2} U_{k_1,k_2},$$
$$S_i := \sum_k A^{-1}_{i,k} \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y)$$
$$\times \left(\sum_{y'} F_k(\phi_\infty, y') \nabla_x^T K^+_{y',y'}(\phi_\infty) \ell + K^+_{y',y'}(\phi_\infty) \nabla_x^T F_k(\phi_\infty, y') \ell \right).$$

The proof of this proposition is given in Appendix 6.8.5.

6.5 Example: CSMA Model

To illustrate our results, we consider the unsaturated CSMA random-access networks studied in [39]. In this paper, the authors use a two-scale model to study the performance of a CSMA algorithm with many nodes. The slow process corresponds to the arrival of jobs and the fast process corresponds to the activation and deactivation of nodes. The authors of this paper derive a mean field approximation and show that it is asymptotically exact. With our methods, we go two steps further:

- Theorem 25 and 26 show that not only the mean field approximation is asymptotically exact but also that the error is only of order $O(1/N)$.
- By using Theorem 27, we can compute a refinement term. Our numerical example shows that, similarly to what happens for classical one-scale mean field models [59], this refinement term is extremely accurate. It is much more accurate than the classical mean field approximation when the studied system is not too large.

(a) The linear 3 Node Graph.

(b) The 5 Node Graph.

Figure 6.1: Two examples of interference graphs.

6.5.1 Model Description

We consider a model with C server types, with N statistically identical servers for each class. All servers communicate through a wireless medium using a random-access protocol and have a finite

buffer size B. The C classes form an interference graph $G = (\mathcal{C}, \mathcal{E})$ with $\mathcal{C} = \{1, \dots, C\}$ the classes and $\mathcal{E}$ the network specific edges. (see for instance Figure 6.1 for examples of graphs with three or five classes). This interference graph indicates that two servers cannot transmit simultaneously if they either are of the same class or belong to an adjacent class. For each class $c \in \mathcal{C}$, we will denote by $y_c \in \{0, 1\}$ a variable that equals 1 if a node of class c is transmitting and 0 otherwise. We denote by $\mathcal{Y}$ the set of possible activation vectors $y \in \{0, 1\}^C$. It is equal to the set of the independent sets of the graph, i.e., all activity vectors for which an active node has only inactive neighbors. For instance, for the graph with three classes which are linearly connected as shown in Figure 6.1a, the set of feasible states is given by

$$\mathcal{Y} = \{(0, 0, 0), (1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 0, 1)\}.$$

Any node can turn active if there are no neighboring active nodes. Once its transmission is finished, it changes back to an idle state. For a given class c, we define $\mathcal{Y}_c^+ := \{y \in \mathcal{Y} : y_c = 1\}$ as the subset of states for which node c is active, and by $\mathcal{Y}_c^- := \{y \in \mathcal{Y} : y_c = y_d = 0 \, \forall d \text{ s.t. } (c, d) \in \mathcal{E}\}$ the subset of states from which node c can turn active, i.e., all neighboring nodes are inactive. For instance, for the linear 3 node graph of Figure 6.1a and the class $c = 1$ this yields the following sets

$$\mathcal{Y}_1^+ = \{(1, 0, 0), (1, 0, 1)\}, \qquad \mathcal{Y}_1^- = \{(0, 0, 0), (0, 0, 1)\}.$$

As in [39], in this non-saturated model, we consider that if a node is in class c, new packets arrive to this node at rate $\lambda_c > 0$. If a node has a packet to transmit and no neighboring node is transmitting, then this node becomes active at rate $\nu_c > 0$. We assume that a transmission from a node of class c takes an exponential time of duration $1/(N\mu_c) > 0$, after which the packet leaves the system.

To illustrate the model dynamics, we provide short videos for of the linear 3 node graph:

- Video Illustration of the Dynamics for $N = 10$
- Video Illustration of the Dynamics for $N = 1000$

6.5.2 Two-scale model representation

The model as described above fits in our two-timescale representation. To see why, for each class $c \in \mathcal{C}$ and buffer size $b \in \{0 \dots B\}$, we define $X_{t,(c,b)}^{(N)}$ as the fraction of servers of class c that have at least b jobs in their queue at time t. We denote by $\boldsymbol{X}_t$ the vector of all possible $X_{t,(c,b)}$ for all $c \in \mathcal{C}$ and $b \in \{0 \dots B\}$. The fast component $\boldsymbol{Y}_t$ is the activation at time t: $Y_{t,(c)} = 1$ if a node of class c is transmitting at time t and 0 otherwise.

Using this representation, we characterize the possible transitions. Given a state pair $(\boldsymbol{x}, \boldsymbol{y})$, the transitions are represented as a transition vector of the form $(\ell/N, \boldsymbol{y}')$ and a corresponding transition rate $N\alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y})$ such that the state $(\boldsymbol{x}, \boldsymbol{y})$ jumps to $(\boldsymbol{x} + \ell/N, \boldsymbol{y}')$ at rate $N\alpha_{\ell,\boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y})$. The transitions can be distinguished into three types:

- Arrival of a packet to a server of class $c \in \mathcal{C}$:

$$\begin{array}{llll} (\frac{e_{c,i}}{N}, \boldsymbol{y}) & \text{at rate} & N\lambda_c(1 - x_{c,i}) & \text{for } i = 1, \\ (\frac{e_{c,i}}{N}, \boldsymbol{y}) & \text{at rate} & N\lambda_c(x_{c,i-1} - x_{c,i}) & \text{for } 2 \le i \le B. \end{array}$$

- Back-off of a server of class $c \in \mathcal{C}$ with at least one packet if the class activity vector allows the back-off, i.e., $\boldsymbol{y} \in \mathcal{Y}_c^-$:

$$\begin{array}{llll} (-\frac{e_{c,i}}{N}, \boldsymbol{y} + e_c) & \text{at rate} & N\nu_c(x_{c,i} - x_{c,i+1}) & \text{for } 1 \leq i < B, \\ (-\frac{e_{c,i}}{N}, \boldsymbol{y} + e_c) & \text{at rate} & N\nu_c x_{c,i} & \text{for } i = B. \end{array}$$

- Transmission completion of an active node of class $c \in \mathcal{C}$, i.e., $\boldsymbol{y} \in \mathcal{Y}_c^+$:

$$(0, \boldsymbol{y} - e_c) \quad \text{at rate} \quad N\mu_c.$$

We denote the set of all possible transition vectors $(\ell/N, \boldsymbol{y}')$ from a state pair $(\boldsymbol{x}, \boldsymbol{y})$ by $\mathcal{T}(\boldsymbol{x}, \boldsymbol{y})$.

6.5.3 Steady-State distribution $\pi_y(x)$ and average drift

By using the above transition definitions, the matrix K is given by:

- If $\boldsymbol{y} \in \mathcal{Y}_c^-$, then $K(\boldsymbol{y}, \boldsymbol{y} + e_c) = N\nu_c x_{c,1}$,
- If $\boldsymbol{y} \in \mathcal{Y}_c^+$, then $K(\boldsymbol{y}, \boldsymbol{y} - e_c) = N\mu_c$,

all other entries of the matrix being 0.

This representation is used by the authors of [39] to derive the product-form stationary distribution for a fixed server state $\boldsymbol{x}$. This product form is closely related to the product-form stationary distribution of saturated networks as found in [24, 107, 111]: The quantity $\pi_{\boldsymbol{y}}(x)$ is calculated as follows:

$$\pi_{\boldsymbol{y}}(x) := \frac{Z(\boldsymbol{x}, \boldsymbol{y})}{Z(\boldsymbol{x})}, \qquad \text{with} \qquad Z(\boldsymbol{x}, \boldsymbol{y}) = \prod_{c \in \mathcal{C}} \Big(\frac{\nu_c}{\mu_c} x_{c,1}\Big)^{y_c}, \qquad Z(\boldsymbol{x}) = \sum_{\boldsymbol{y} \in \mathcal{Y}} Z(\boldsymbol{x}, \boldsymbol{y}).$$

Following our definition of Section 6.2.2 the drift and its average version are generically defined by:

$$F(\boldsymbol{x}, \boldsymbol{y}) = \sum_{(\ell, \boldsymbol{y}') \in \mathcal{T}(\boldsymbol{x}, \boldsymbol{y})} \alpha_{\ell, \boldsymbol{y}'}(\boldsymbol{x}, \boldsymbol{y})\ell, \qquad \text{and} \qquad \bar{F}(\boldsymbol{x}) = \sum_{w \in \mathcal{Y}} \pi_{\boldsymbol{y}}(\boldsymbol{x}) F(\boldsymbol{x}, \boldsymbol{y}).$$

For the drift $F(\boldsymbol{x}, \boldsymbol{y})$ of the random access model this leads to the closed form expression

$$\begin{aligned} F(\boldsymbol{x}, \boldsymbol{y}) = \sum_{c \in \mathcal{C}} \Big(& \sum_{1 < i \leq B} e_{c,i}\, \lambda_c(x_{c,i-1} - x_{c,i}) + e_{c,1}\, \lambda_c(1 - x_{c,1}) \\ & - \mathbf{1}_{\{y \in \mathcal{Y}_c^-\}} \Big(\sum_{1 \leq i < B} e_{c,i}\, \nu_c(x_{c,i} - x_{c,i+1}) + e_{c,B}\, \nu_c x_{c,B} \Big) \Big). \end{aligned}$$

It should be clear that assumption (A_1) holds in our case because the rates given in Section 6.5.2 are all continuous in $\boldsymbol{x}$ (in fact they are all linear). Moreover, the model also satisfies Assumption (A_2): For a given $\boldsymbol{x}$, the set of irreducible states for $K(\boldsymbol{x})$ contains all the feasible activation vectors y such that $y_c = 0$ if $x_{(c,1)} = 0$. The condition $x_{c,1} = 0$ implies that the nodes of class c do

not have any packets to transmit. The situation of Assumption (A_3) is more complicated. To the best of our knowledge, there has not been a complete stability characterization for the unsaturated random access CSMA model. Cecchi et al. show in [39] that in the case of a complete interference graph stability conditions can be derived which assure global exponential stability. They further conjectured that similar results hold for general interference graphs. In our analysis, we assume that (A_3) holds.

6.5.4 Numerical results

To study the accuracy of the mean field approximation and the refined term proposed in Theorem 26, we implemented a Python library[2] to simulate the system and compute the mean field approximation and the refinement term. This library is generic and can take as an input any instance of the model which we defined above. For instance, in the Code Cell 6.1, we illustrate how to use this library to construct a model where the interference graph is as in Figure 6.1b, the rates are $\lambda = [.5, .7, .7, .6, .4], \nu = [4, 3, 3, 3, 3], \mu = [3, 3, 2, 4, 2]$, and the buffer size is equal to 10. This cell shows how to initialize the 5 node model and obtain the approximation and refinement from our implementation. We also perform the same experiments with the linear 3 node model, for which we provide the results in Appendix 6.10. Note that the results are qualitatively very similar.

Listing 6.1: Initialization and Computation of Mean Field and Refinements.

```
# Graph structure (this is the five node example)
G = np.array([[0,1,1,1,0],
              [1,0,0,0,0],
              [1,0,0,0,1],
              [1,0,0,0,1],
              [0,0,1,1,0]])
# rates and buffer size
_lambda = np.array([.5,.7,.7,.6,.4])
nu = np.array([4,3,3,3,3])
mu = np.array([3,3,2,4,2])
buffer_size = 10

# We define the model, compute a trajectory and the refinement term.
csma = symbolic_CSMA(nu, mu, _lambda, G, buffer_size)
T, X = csma.ode(time=200)                          # mean field ODE
v, s, t, w, u = csma.compute_refinements(X[-1]) # steady-state refinement
```

In order to compute the refinement terms, the library needs to compute various derivatives (of the drift or of the matrix $K^+(\boldsymbol{x})$). To implement this, we rely on symbolic differentiation provided by the `sympy` library [85]. As we see later in Table 6.1, the use of the symbolic differentiation is the performance bottleneck of our implementation. In Appendix 6.9 we furthermore show how to obtain the stationary distribution and the derivative of $K^+(x)$ numerically.

[2] https://gitlab.inria.fr/sallmeie/bias-and-refinement-of-multiscale-mean-field-models

Transient regime and illustration of Theorem 25 To illustrate the accuracy of the mean field, we use the 5 node model described in the Code Cell 6.1. We first simulate the CSMA model and compare it with the mean field ODE. The results are reported in Figure 6.2 where we plot the mean field approximation against a sample mean $\mathbb{E}[\boldsymbol{X}_t]$ derived from 1000 simulations. Initially, all servers are idle. The plot shows the share of servers of class $c = 3$ that have at least one job, that is $\mathbb{E}[X_{t,(3,1)}]$. We compare the results for a model with $N = 10$ servers per class (top right), $N = 20$ (bottom right), or $N = 50$ (left). We observe that in all cases, the evolution of the stochastic system is very well predicted by the mean field approximation. To quantify this more precisely, each plot contains a zoom on the trajectory between the time $t = 8$ to $t = 13$. These zooms show that for $N = 50$, the quantity $\mathbb{E}[X^{(N)}_{t,(3,1)}]$ is almost indistinguishable from the mean field approximation. For $N = 10$ or $N = 20$, the estimated average is slightly above the mean field curve, but the confidence intervals remain almost equal to the error.

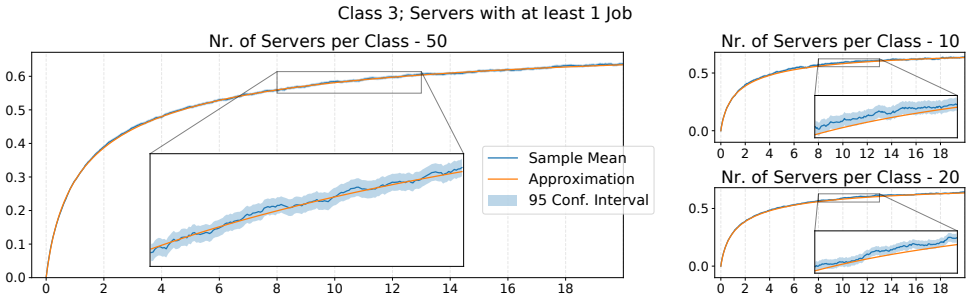


Figure 6.2: Illustration of the transient behavior of the CSMA model. We compare the 'average' mean field and stochastic simulations for three different scaling parameters: $N = 10$, $N = 20$ and $N = 50$.

Steady-state and refined accuracy While Theorems 25 and 26 provide a guarantee on the accuracy of the mean field approximation, Theorem 27 shows that it is possible to compute an approximation that is more accurate than the original mean field approximation. We illustrate this in Figure 6.3 where we show the steady-state average queue lengths for the same 5 node graph. The sample mean and confidence interval are computed from 40 steady-state samples which again are obtained from independent time-averages of 7.5×10^6 events of the Markov chain after a warm-up of 2.5×10^6 events. For a class c and a buffer size b, the quantity $\sum_{b=1}^{B} \mathbb{E}[X_{(c,b)}]$ is equal to the steady-state average queue length of each server of class c. In Figure 6.3, we consider different values of N, and calculate the average queue length by the following three methods:

- By using a stochastic simulator of the original CSMA model.
- By using the fixed point of the mean field approximation: $\sum_{b=1}^{B} (\phi_\infty)_{(c,b)}$.
- By computing the refinement term, C, of Theorem 27 and using $\sum_{b=1}^{B} (\phi_\infty)_{(c,b)} + (C)_{(c,b)}/N$.

When looking at the scale of the y-axis, we see that in all cases, the accuracy of the mean field approximation is already quite good. More importantly, we also observe that in all cases, the refined

approximation seems almost exact: For all considered cases, the refined approximation lies within the 95 percent confidence interval of the simulations and seems to work well even for a small number of servers, $N \approx 10, 20$. This result is similar to the one observed for one-timescale mean field models in [59].

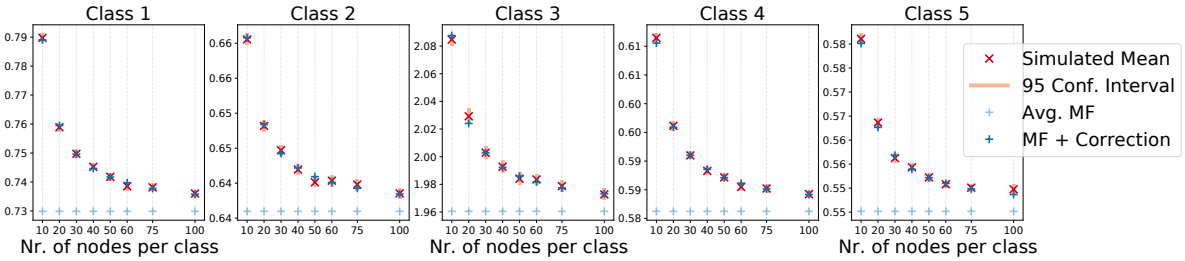


Figure 6.3: Queue Length Distribution for the 5 Node Graph of Figure (6.1b).

Computation Time While the previous figure shows that the refined approximation provides an increase in accuracy for small values of N, it comes at the cost of an increase in computation time because one needs to compute the various derivatives of the rate functions and to solve a new linear systems of equations. In order to quantify the additional computation time, we measure the time taken by our implementation to compute the refinement terms which are reported in Table 6.1. We compare the 5 node model studied before and a 3-node model whose interference graph is as in Figure 6.1a. We observe that the time taken to compute the refinement term is significant, in particular for the 5-node model. Yet, when looking more carefully at what takes time, we realize that most of the computation time is taken by the symbolic differentiation. Indeed, to simplify our implementation, we used the automated differentiation method of `sympy`. While this yields simplifications for the implementation, we encountered that it massively slows down the refinement computation times. Through code profiling it showed that around 95 percent of the computing time is taken by `sympy` methods such as differentiation and evaluation of symbolic expressions. For smaller interference graphs, e.g., linear 2 / 3 node graphs, this effect is not limiting. For larger graphs, the differentiation turns out to be the restricting factor. In Table 6.1 we state the computation times for a linear 3 node model and for the setup described before.

We would like to emphasize that the goal of our implementation is to illustrate the theoretical statements, and thus we did not focus on efficiency. The table shows that if one wants to adapt our implementation to work with larger graphs, it would be sufficient to implement a more efficient differentiation method. For instance, this could be done by using closed form expression of the derivatives, or by using automatic differentiation methods, or by using finite difference methods. We believe that such methods would probably be much faster.

6.6 Conclusion

In this paper we investigate the accuracy of the classical *averaging method* that is used to study two timescale models. We study a generic two timescale model and show that under mild regularity conditions, the bias of this 'average' mean field approximation is of order $O(1/N)$. This result holds

	Jakobian (A)	Hessian (B)	v + w	s + t + u	Total	Sympy
3 Node	6.08 (6.08)	30.71 (30.71)	0.19 (0)	5.64 (3.81)	42.63 (40.6)	95.22%
5 Node	97.09 (97.09)	897.15 (897.15)	1. (0)	122.36 (86.09)	1117.6 (1080.33)	96.66%

Table 6.1: Refinement computation times for Random Access CSMA model for a 3 node linear graph and the 5 node graph as in Figures 6.1a, 6.1b. Times are given in seconds. In parentheses, we indicate the time taken by differentiation and subsequent sympy methods. These numbers show that the Sympy code takes more than 95% of the computation time.

for any finite time-horizon, and extends to the steady-state regime under the classical assumption that the system has a unique and stable fixed point. Our results show the existence of a bound $\bar{C}_h$ for the bias term for any regular function h:

$$\mathbb{E}[h(\boldsymbol{X}_t)] = \underbrace{\underbrace{h(\Phi_t(\boldsymbol{x}))}_{\text{classical 'average' mean field}} + \underbrace{\frac{1}{N}C_h(t)}_{O(1/N)\text{ expansion of the bias}}}_{\text{refined 'average' mean field}} + \; o(1/N).$$

For the steady-state regime $t = +\infty$, we propose an algorithmic method to calculate this term C_h. This correction term can be computed by solving linear systems and is therefore easily numerically computable. We show on an example that, similarly to what was done for classical one timescale models [59], the bias term leads to an approximation that is almost exact for small values of N like $N = 10, 20$.

An interesting open question would be to obtain a characterization of $C_h(t)$ for the transient regime. Yet, it is not clear to us if those expressions would be usable as their size grows quickly with the system size. From an application point of view, our examples show that the new approximation leads to very accurate estimates for CSMA models. We believe that the same should hold for other multiscale models.

6.7 Definitions

In this section we revise some essential definitions and properties used in the paper. As these definitions are well established, we only briefly recall them to provide a self-contained paper.

6.7.1 C_0-Semi-Group, Hölder Norm, ODE differentiability

Definition 32 (C_0 Semi-Group [94] Definition 2.1)**.** *T_s is called strongly continuous semi-group (or C_0-semi-group) if*

$$T_0 = Id, \qquad T_{s+t} = T_s T_t \text{ for all } s, t \geq 0, \qquad \lim_{t \downarrow 0} T_t z = z \text{ for all } z. \tag{6.23}$$

Since we will only work with C_0-semi-groups we will simply refer to them as semi-groups.

Semi-group & Generator commutation The generator of a C_0-semi-group is defined by

$$Az = \lim_{t \to 0} \frac{1}{t}(T_t z - T_0 z).$$

A direct consequence of the definition of the generator and a standard property is that it commutes with its defining C_0-semi-group, i.e.,

$$AT_s h(z) = T_s A h(z) = \frac{d}{ds} T_s h(z).$$

Note, this therefore holds true for the semi-groups given by the stochastic system $\Psi_s h(x, y) = \mathbb{E}[h(X_s, Y_s) \mid X_0, Y_0 = x, y]$ with generator $Lh(x, y)$ as in Equation (6.7) and the semi-group of the ODE $\Phi_s h(x) = h(\phi_s(x))$ with generator $\Lambda h(x) = D_x h(x) \bar{F}(x)$.

Definition 33 (Hölder Norm and Space)**.** *For $U \subset \mathbb{R}^n$ and $u \in C^k(U)$*

$$\|u\|_{k,\gamma} := \sum_k \sup_{x \in U} \left\| D^k u(x) \right\| + \sum_k \sup \left\{ \frac{\left\| D^k u(x) - D^k u(y) \right\|}{\|x - y\|^\gamma} \mid x, y \in U, x \neq y \right\}$$

is called Hölder norm. The space of functions for which the norm is finite is called Hölder space and denoted by $\mathcal{D}^k_\gamma(U)$. For the case $\gamma = 1$ the Hölder space encloses all functions who are k-times continuously differentiable with bounded derivatives and who's k-th derivatives are Lipschitz continuous. The latter we simply denote by $\mathcal{D}^k(U)$.

An important implication is that all Hölder continuous functions are uniformly continuous.

Lemma 34 (Drift induced differentiability [95] Theorem 1 p.80)**.** *Let E be an open subset of $\mathbb{R}^n$ containing x_0 and assume that $f \in C^k(E)$. Then there exists an $a > 0$ and $\delta > 0$ such that for all $y \in N_\delta(x_0)$[3] the initial value problem*

$$\dot{x} = F(x), \qquad x(0) = y,$$

has a unique solution u which is k-times continuously differentiable with respect to the initial condition for $t \in [-a, a]$.

Proof. Theorem 1 p.80-83 and Remark 1 p.83 of [95]. □

6.8 Technical Lemmas and Proofs

6.8.1 Proof of Lemma 28

By assumption (A_2), the transition matrix $K(x)$ has a unique irreducible class. As pointed out in Section 6.2.3, the corresponding Markov chain has a unique stationary distribution that we denote

[3] $N_\delta(x_0) := \{x \in \mathbb{R}^n : \|x - x_0\| \leq \delta\}$

by $\pi(x)$. Let $\Pi(x) := \mathbb{1}\pi^T(x)$ be the matrix whose lines are all equal to $\pi(x)$. By [68](Theorem 3.5, p.17), $\big(K(x)+\Pi(x)\big)$ is non-singular and its inverse $(K(x)+\Pi(x))^{-1}$ is a generalized inverse to $K(x)$, which means that it satisfies $K(x)(K(x)+\Pi(x))^{-1}K(x) = K(x)$.

To obtain the solution to the Poisson equation (6.11) we only consider the case where h takes values in $\mathbb{R}$. The extension to a function h that takes values in $\mathbb{R}^n$ is straightforward as it corresponds to n independent Poisson equations.

Let us suppress the dependence on x for clarity. Recall that $K^+ = (K+\Pi)^{-1}(I-\Pi)$ and let us study the product KK^+:

$$\begin{aligned}
KK^+ &= K(K+\Pi)^{-1}(I-\Pi) && \textit{(by definition)}\\
&= K(K+\Pi)^{-1}(I+K-K-\Pi)\\
&= K(K+\Pi)^{-1} + K(K+\Pi)^{-1}K\\
&\quad - K(K+\Pi)^{-1}(K+\Pi) && \textit{(expanding the product)}\\
&= K(K+\Pi)^{-1} && \textit{(the last two terms equal } \pm K\textit{)}\\
&= (K+\Pi)(K+\Pi)^{-1} - \Pi(K+\Pi)^{-1} && \textit{(Adding and subtracting } \Pi(K+\Pi)^{-1}\textit{)}\\
&= I-\Pi,
\end{aligned}$$

where the last equality holds because $\pi^T\mathbb{1} = I$ and therefore $\Pi\,\Pi = \mathbb{1}\pi^T\mathbb{1}\pi^T = \mathbb{1}\pi^T = \Pi$. Combined with $\Pi\,K = 0$, this shows that $\Pi = \Pi(K+\Pi)$ and therefore $\Pi(K+\Pi)^{-1} = \Pi$.

The above computations show that if $G_h^{\text{fast}}(x,y) = \sum_{y'} K^+_{y,y'}(x)h(x,y')$, then:

$$K(x)G_h^{\text{fast}}(x,y) = h(x,y) - \sum_{y'}\pi_{y'}(x)h(x,y'),$$

which shows that G_h^{fast} is the solution of the Poisson equation.

The differentiability of G_h^{fast} follows from the differentiability of h and K^+: Under Assumption (A_1), $K(x)$ is continuously differentiable. By Assumption (A_2), $K(x)$ has a unique irreducible class, this implies $\Pi(x)$ is continuously differentiable, for which we refer to [70], and therefore further implies that $(I+\Pi(x))^{-1}$ and $K^+(x)$ are continuously differentiable. This proves that G_h^{fast} is continuously differentiable in x.

6.8.2 Technical Lemmas used to prove Theorem 25 (Transient Regime)

Lemma 35. *For arbitrary but fixed $t > 0$, let $g : (s,x,y) \in [0,t]\times\mathcal{X}\times\mathcal{Y} \mapsto g_s(x,y) \in \mathbb{R}$ be a continuous function that is continuously differentiable in s and let $H_s(x,y) := \mathbb{E}[g_s(X_s,Y_s) \mid X_0,Y_0 = x,y]$. Then:*

$$H_t(x,y) - H_0(x,y) = \int_0^t \mathbb{E}[\frac{d}{d\tau}g_{s+\tau}(X_s,Y_s)\mid_{\tau=0}]ds + \int_0^t \mathbb{E}[Lg_s(X_s,Y_s)]ds.$$

Proof. By definition of the generator L, the quantity $H_t(x,y) = \mathbb{E}[g_t(X_t,Y_t) \mid X_0,Y_0 = x,y]$ is right sided differentiable, i.e., $\frac{d^+}{ds}f(s) = \lim_{ds\to 0^+}\frac{1}{ds}\big(f(s+ds)-f(s)\big)$, with respect to time. Using semi-group properties and bounds, its derivative is

$$\frac{d^+}{ds}H_s(x,y) = \mathbb{E}[\frac{d}{d\tau}g_{s+\tau}(X_s,Y_s)\mid_{\tau=0}] + \mathbb{E}[Lg_s(X_s,Y_s)].$$

The first term corresponds to the derivation of g_s with respect to time and the second term to the changes of the stochastic system in (X_s,Y_s). The lemma therefore follows by using that $H_t(x,y) - H_0(x,y) = \int_0^t \frac{d^+}{ds}H_s(X_s,Y_s)$. □

Lemma 36 (Bound for $L_{\text{fast}}g(x,y)$)**.** *Take the two-timescale stochastic system as introduced in Section 6.2.1 with generator L and assume (A_1) and (A_2). Let L_{fast} be as defined in Section 6.4.2. Then, for $g \in \mathcal{D}^1(\mathcal{X}\times\mathcal{Y})$ and $(x,y)\in\mathcal{X}\times\mathcal{Y}$*

$$L_{fast}g(x,y) - \frac{1}{N}Lg(x,y) = \frac{1}{N}\sum_{\ell,y'}\alpha_{\ell,y'}(x,y)\ell\, D_x g(x,y') + o(1/N).$$

Proof. By definition of L_{fast}, for a continuous function g the values of $L_{\text{fast}}g(x,y)$ and $\frac{1}{N}Lg(x,y)$ coincide in the limit. For finite N, we first look at $\frac{1}{N}Lg(x,y)$ which is given by

$$\frac{1}{N}Lg(x,y) = \frac{1}{N}\sum_{\ell,y'}N\alpha_{\ell,y'}(x,y)(g(x+\frac{\ell}{N},y') - g(x,y)).$$

Using the continuity of g in x, and definition of $L_{\text{fast}}g(x,y)$ we have

$$\begin{aligned}L_{\text{fast}}g(x,y) = \lim_{N\to\infty}\frac{1}{N}Lg(x,y) &= \sum_{\ell,y'}\alpha_{\ell,y'}(x,y)(\lim_{N\to\infty}g(x+\frac{\ell}{N},y') - g(x,y))\\ &= \sum_{\ell,y'}\alpha_{\ell,y'}(x,y)(g(x,y') - g(x,y)).\end{aligned}$$

Using Taylor's theorem

$$\begin{aligned}Lg(x,y) &= N\sum_{\ell,y'}\alpha_{\ell,y'}(x,y)(g(x+\frac{\ell}{N},y') - g(x,y))\\ &= N\sum_{\ell,y'}\alpha_{\ell,y'}(x,y)(g(x,y') - g(x,y) + \ell\, D_x g(x,y') + o(\|\ell\|))\\ &= NL_{\text{fast}}g(x,y) + \sum_{\ell,y'}\alpha_{\ell,y'}(x,y)\ell\, D_x g(x,y') + o(1).\end{aligned}$$

□

6.8.3 Technical Lemma used to prove Theorem 27 (Steady-State Refinement)

The next Lemma justifies that the first term of (6.21) is approximated in the limit by the 'average' version of J_h as defined in Section 6.4.5. Second, Lemma 38 gives stability conditions for the solution of the 'slow' Poisson Equation.

Lemma 37. *Assume (A_1)-(A_3) in particular assume $h \in \mathcal{D}^1(\mathcal{X})$. Further assume that G_F^{fast}, solution to the 'fast' Poisson Equation, is cont. differentiable in x and G_h^{slow}, solution to the 'slow' Poisson Equation, is twice cont. differentiable in x. Let $\bar{J}(\phi_\infty) = \sum_y \pi(\phi_\infty) J(\phi_\infty, y)$ be the 'average' version of $J_h(x,y) = \sum_{\ell,y'} \alpha_{\ell,y'}(x,y)\, D_x\big(D_x G_h^{slow}(x) G_F^{fast}(x,y')\big)\ell$ in ϕ_∞, then*

$$N\mathbb{E}[D_x G_h^{slow}(\boldsymbol{X}_\infty)\big(\bar{F}(\boldsymbol{X}_\infty) - F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)\big)] = \bar{J}(\phi_\infty) + o(1).$$

Proof. To prove the lemma, let G_F^{fast} be the solution to the Poisson equation as given in ('fast' Poisson Equation). We use G_F^{fast} to rewrite

$$\begin{aligned}
&N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(\bar{F}(\boldsymbol{X}_\infty) - F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty))] \\
&\quad = -N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty) - \bar{F}(\boldsymbol{X}_\infty))] \\
&\quad = -N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(L_{\text{fast}} G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty))] \\
&\quad = -N\mathbb{E}[L_{\text{fast}} D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty))].
\end{aligned}$$

Adding $\mathbb{E}[L D_x G^{\text{slow}}(\boldsymbol{X}_\infty)(G_h^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty))] = 0$ and applying the steps as in the proof of Lemma 36 we see that

$$\begin{aligned}
&- N\mathbb{E}[L_{\text{fast}} D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)) - L D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)(G_F^{\text{fast}}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty))] \\
&= -\mathbb{E}[\sum_{\ell,y'} \alpha_{\ell,y'}(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)\big(D_x(D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty) G_F^{\text{fast}}(\boldsymbol{X}_\infty, y'))\big)\ell\,] + o(1). \qquad (6.24)
\end{aligned}$$

The last equality follows directly from the definition of L_{fast} since G_h only depends on x. Using Theorem 26 we see that $\mathbb{E}[J_h(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)]$ with $J_h(x,y) := -\sum_{\ell,y'} \alpha_{\ell,y'}(x,y)\ell\ D_x\big(D_x G_h^{\text{slow}}(x) G^{\text{fast}}(x,y')\big)$ of equation (6.24) is approximated by $\bar{J}(\phi_\infty) = \sum_y \pi(\phi_\infty) J(\phi_\infty, y)$. This concludes the proof as it implies

$N\mathbb{E}[D_x G_h^{\text{slow}}(\boldsymbol{X}_\infty)\big(\bar{F}(\boldsymbol{X}_\infty) - F(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)\big)] = \mathbb{E}[J_h(\boldsymbol{X}_\infty, \boldsymbol{Y}_\infty)] + o(1) = \bar{J}_h(\phi_\infty) + o(1)$. □

6.8.4 Stability of G_h^{slow}

Lemma 38 (Stability)**.** *Assume that $\bar{F}$ and ϕ are k-times differentiable with uniformly continuous derivatives and that ϕ has a unique exponentially stable attractor ϕ_∞. Then the k-th derivative of $G_h^{slow} : x \mapsto \int_0^\infty h(\phi_s x) - h(\phi_\infty) ds$ is bounded and equal to $\int_0^\infty D_x^k (h \circ \phi_s)(x) ds$.*

Proof. This is a consequence of [59][Lemma 3.5]. □

6.8.5 Proof of Proposition 31

Proof. In the first part of the proof we find computable expressions of $D_x G_h^{\text{slow}}$, $D_x^2 G_h^{\text{slow}}$ and $D_x G_F^{\text{fast}}$. These expressions allow us to rewrite J_h and construct the closed form representation of J_h as well as $\bar{J}_h$ in the following steps. By definition

$$\begin{aligned} J_h(x,y) &= -\sum_{\ell,y'} \alpha_{\ell,y'}(x,y)\, D_x\big(D_x G_h^{\text{slow}}(x) G_F^{\text{fast}}(x,y')\big)\ell \\ &= -\sum_{\ell,y'} \alpha_{\ell,y'}(x,y)\, \big(G_F^{\text{fast}}(x,y')^T D_x^2 G_h^{\text{slow}}(x) + D_x G_h^{\text{slow}}(x) D_x G_F^{\text{fast}}(x,y')\big)\ell. \end{aligned} \tag{6.25}$$

By Lemma 28 there exists a matrix $K^+(x)$ such that $G_F^{\text{fast}}(x,y')$ has the form

$$G_F^{\text{fast}}(x,y') = \sum_{y'\in\mathcal{Y}} K^+_{y',y''}(x) F(x,y'').$$

Assumption (A_1) which assures differentiability of the transition rates α with respect to x, also implies differentiability for G_F^{fast}. Therefore,

$$D_x G_F^{\text{fast}}(x,y') = \sum_{y''\in\mathcal{Y}} F(x,y'')\nabla_x^T K^+_{y',y''}(x) + K^+_{y',y''}(x) D_x F(x,y''),$$

with $\nabla_x^T = \left[\frac{\partial}{\partial x_1},\dots,\frac{\partial}{\partial x_n}\right]$, $D_x f = (\frac{\partial}{\partial x_j} f_i)_{i,j=1\dots n}$.

Using the results of [59][Lemma 3.6] with A, B the first and second derivative of $\bar{F}$ as defined in (6.22), it holds that for the equilibrium point ϕ_∞

$$D_x G_h^{\text{slow}}(\phi_\infty)_i = \sum_j \frac{\partial h}{\partial x_j}(\phi_\infty) \int_0^\infty \big(D_x\phi_s(\phi_\infty)\big)_{j,i} ds = \sum_j \frac{\partial h}{\partial x_j}(\phi_\infty)(-A)^{-1}_{j,i}$$

as well as

$$\begin{aligned} D_x^2 G_h^{\text{slow}}(\phi_\infty)_{n,m} &= \sum_{i,j} \frac{\partial^2 h}{\partial x_i \partial x_j}(\phi_\infty) \int_0^\infty \big(D_x\phi_s(\phi_\infty)\big)_{j,n} \big(D_x\phi_s(\phi_\infty)\big)_{i,m} ds \\ &\quad + \sum_i \frac{\partial h}{\partial x_i}(\phi_\infty) \int_0^\infty \big(D_x^2\phi_s(\phi_\infty)\big)_{j,m,n} ds \\ &= \sum_{i,j} \frac{\partial^2 h}{\partial x_i \partial x_j}(\phi_\infty) \int_0^\infty \big(e^{As}\big)_{j,n} \big(e^{As}\big)_{i,m} ds \\ &\quad + \sum_i \frac{\partial h}{\partial x_i}(\phi_\infty) \Big(\sum_j (-A)^{-1}_{i,j} \sum_{k_1,k_2} B_{j,k_1,k_2} \int_0^\infty \big(e^{As}\big)_{k_1,n} \big(e^{As}\big)_{k_2,m} ds\Big). \end{aligned} \tag{6.26}$$

Next, the above equations are used to rewrite (6.25). To obtain a closed form expression for the left summand, we start by looking at the sum

$$\sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) \sum_{y'} K^+_{y',y'} F_m(\phi_\infty, y') \ell_n \int_0^\infty \left(e^{As}\right)_{k_1,n} \left(e^{As}\right)_{k_2,m}. \tag{6.27}$$

A solution to the above equation is given by the following Lyapunov equation. To ease notations, define

$$O_{m,n} := \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) K^+_{y',:}(\phi_\infty) F_m(\phi_\infty, :) \ell_n,$$

or, equivalently in matrix notation, $O = \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) K^+_{y',:}(\phi_\infty) F(\phi_\infty, :) \ell^T$.

If a matrix U solves[4] the Sylvester equation (for X)

$$AX + XA^T = -O, \tag{6.28}$$

it is equal to (6.27). Applying this identity and (6.26) to the first summand of $\bar{J}(\phi_\infty)$ which is the 'average' version of (6.25), lets us rewrite

$$\begin{aligned}
&\sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y)\, G_F^{\text{fast}}(\phi_\infty, y')^T D_x^2 G_h^{\text{slow}}(\phi_\infty) \ell \\
&\quad = \sum_{i,j} \frac{\partial^2 h}{\partial x_i \partial x_j}(\phi_\infty) U_{i,j} + \sum_i \frac{\partial h}{\partial x_i}(\phi_\infty) \sum_j (-A)^{-1}_{i,j} \sum_{k_1,k_2} B_{j,k_1,k_2} U_{k_1,k_2}.
\end{aligned}$$

For the second major summand appearing in the definition of $\bar{J}(\phi_\infty)$, writing out the solutions to the Poisson equations and their derivatives yields

$$\begin{aligned}
&\sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y)\, D_x G_h^{\text{slow}}(\phi_\infty) D_x G_F^{\text{fast}}(\phi_\infty, y') \ell \\
&= \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) \times \\
&\quad \sum_{j,k} \left(\sum_i \frac{\partial h}{\partial x_i}(\phi_\infty) (-A)^{-1}_{i,k} \right) \\
&\qquad \times \left(\sum_{y'} \frac{\partial}{\partial x_j} K^+_{y',y'}(\phi_\infty) F_k(\phi_\infty, y') + K^+_{y',y'}(\phi_\infty) \frac{\partial}{\partial x_j} F_k(\phi_\infty, y') \right) \ell_j.
\end{aligned}$$

[4] As A is non-singular, A and $-A^T$ don't share any eigenvalues and therefore equation (6.28) has a unique solution.

By using vector notation and rearranging the sums this is equal to

$$\sum_i \frac{\partial h}{\partial x_i}(\phi_\infty) \sum_k (-A)^{-1}_{i,k} \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) \times \left(\sum_{y'} F_k(\phi_\infty, y') \nabla_x^T K^+_{y',y'}(\phi_\infty) \ell + K^+_{y',y'}(\phi_\infty) \nabla_x^T F_k(\phi_\infty, y') \ell \right). \tag{6.29}$$

Lastly, define $T_i := \sum_j A^{-1}_{i,j} \sum_{k_1,k_2} B_{j,k_1,k_2} U_{k_1,k_2}$ and

$$S_i := \sum_k A^{-1}_{i,k} \sum_y \pi_y(\phi_\infty) \sum_{\ell,y'} \alpha_{\ell,y'}(\phi_\infty, y) \times \left(\sum_{y'} F_k(\phi_\infty, y') \nabla_x^T K^+_{y',y'}(\phi_\infty) \ell + K^+_{y',y'}(\phi_\infty) \nabla_x^T F_k(\phi_\infty, y') \ell \right).$$

This concludes the proof as by definition of T,S, and U we have:

$$\bar{J}(\phi_\infty) = \sum_i \frac{\partial h}{\partial x_i}(\phi_\infty)(T_i + S_i) + \sum_{i,j} \frac{\partial^2 h}{\partial x_i \partial x_j}(\phi_\infty) U_{i,j}.$$

□

6.9 Computational Notes

The first note describes how to calculate the steady-state probabilities $\pi(x)$ associated to the transition matrix $K(x)$.

Note 39 (Note on the computation for the stationary probabilities.)**.** *As the finite state continuous time Markov chain with generator $K(x)$ has unique irreducible class, there exists a non-trivial unique stationary distribution $\pi(x)$[5]. Denote by $\{1, \dots, m\}$ the states of the Markov chain. $\pi(x)$ is obtained by solving the linear system*

$$v\, K(x) = \mathbf{0}, \quad \sum_{i=1}^m v_i = 1,$$

with $v \geq 0$ component wise. By definition, $K(x)$ is of rank $m-1$. Using its structure, i.e., $\sum_{y'} K(x)_{y,y'} = 0\ \forall y$, we can rewrite the above over-determined linear system by replacing the last column of the generator with $\mathbb{1} = [1, \dots, 1]^T$ yielding

$$v\, [K(x)_{:,1}, \dots, K(x)_{:,m-1}, \mathbb{1}] = [\underbrace{0, \dots, 0}_{m-1 \text{ times}}, 1],$$

[5] As the Markov chain is allowed to have transient states the stationary distribution can take zero values.

for which the solution is the stationary distribution $\pi(x)$. By $K(x)_{:,y},\ y = 1,\dots,m-1$ we denote the y-th column of $K(x)$.

As shows in Lemma 28 the solution to the 'fast' Poisson Equation 6.11 has the form $G_h^{\text{fast}}(x,y) = \sum_{y'} K^+_{y,y'}(x)h(x,y')$. To compute the bias correction terms, it is necessary to calculate the first derivative of G_h^{fast} with respect to x. Since the computation of the derivative of $(K(x)+\Pi(x))^{-1}$ can be non-trivial and time consuming, the note below elaborates how the derivative can be efficiently obtained.

Note 40 (Computation of $D_x K^+(x)$)**.** *By definition $K^+(x) = (K(x)+\Pi(x))^{-1}(I-\Pi(x))$. Using basic matrix derivation rules one has*

$$\begin{aligned}\frac{\partial}{\partial x_i}K^+(x) &= \frac{\partial}{\partial x_i}\big((K(x)+\Pi(x))^{-1}(I-\Pi(x))\big)\\ &= \frac{\partial}{\partial x_i}\big(K(x)+\Pi(x)\big)^{-1}\big(I-\Pi(x)\big) - \big(K(x)+\Pi(x)\big)^{-1}\frac{\partial}{\partial x_i}\Pi(x). \qquad (6.30)\end{aligned}$$

Since the numerical difficulty lies only the computation of $D_x\big(K(x)+\Pi(x)\big)^{-1}$ we focus solely on it. Define $E(x) = (K(x)+\Pi(x))$ and let I be the identity matrix. As pointed out in Lemma 28 $(K(x)+\Pi(x))$ is indeed invertible and thus for the partials derivative $\frac{\partial}{\partial x_i}, i = 0\dots d_x$ the following holds:

$$\begin{aligned}&\frac{\partial}{\partial x_i}I = \frac{\partial}{\partial x_i}(E(x)E^{-1}(x))\\ \Leftrightarrow\quad &0 = (\frac{\partial}{\partial x_i}E(x))E^{-1}(x) + E(x)\frac{\partial}{\partial x_i}E^{-1}(x)\\ \Leftrightarrow\quad &\frac{\partial}{\partial x_i}E^{-1}(x) = -E^{-1}(x)(\frac{\partial}{\partial x_i}E(x))E^{-1}(x)\\ \Leftrightarrow\quad &\frac{\partial}{\partial x_i}(K(x)+\Pi(x))^{-1} = -(K(x)+\Pi(x))^{-1}\big(\frac{\partial}{\partial x_i}(K(x)+\Pi(x))\big)(K(x)+\Pi(x))^{-1}.\end{aligned}$$

The above can now be used to compute Equation (6.30).

6.10 Numerical Results for the 3 Node Model

For completeness, we give the numerical results of the linear 3 node model of Graph 6.1a. To obtain the results we modify the parameters of the code of Code Cell 6.1 to match with the 3 node setup. The new model is defined as in Code Cell 6.2.

Listing 6.2: Initialization of Mean Field for the 3 Node Model.

```
# Graph structure (this is the 3 node example)
G = np.array([[0,1,0],
              [1,0,1],
              [0,1,0]])
# rates & buffer size
```

```
_lambda = np.array([0.4,0.2,0.5])
nu = np.array([1.2,2.,1.5])
mu = np.array([1.4,1.3,1.7])
buffer_size = 10
```

As seen in Figure 6.4, we obtain similar results as for the 5 node model when considering steady-state average queue length values. The sample mean and confidence interval are computed from 40 steady-state samples in the same manner as for the 5 node example. The refined approximation almost exactly indicates the stationary value of the stochastic process even for small N while the mean field approximation gets more accurate as N grows.

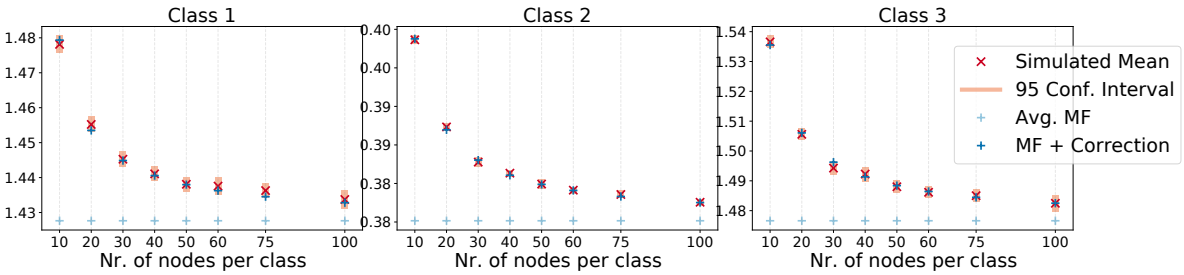


Figure 6.4: Stationary Queue Length Distribution for the 3 Node Graph of Figure (6.1a).

7 Application to Stochastic Approximation

In this chapter, we adapt our results of [4] to the stochastic approximation setting. We show how the methodology developed in our paper can be used to derive accuracy results and a bias extension for the stochastic approximation setting with constant stepsize and state-dependent Markovian noise data.

Contents

7.1 Introduction

The study of stochastic approximation dates back to the seminal paper of Robbins and Monroe published in the 1950s [97]. The basic paradigm is a stochastic difference equation

$$X_{n+1} = X_n + \alpha_n Z_n,$$

where X_n takes values in some Euclidean space, Z_n is a random variable, and $\alpha_n > 0$ the stepsize which is small and possibly tends to zero as $n \to \infty$. In this simple form, X is a parameter of a system and the random variable Z_n is a function of noisy data observed for the state X_n. Classical works on stochastic approximation focus on settings with diminishing stepsize which satisfies $\sum_{n=1}^{\infty} \alpha_n = \infty$ and $\sum_{n=1}^{\infty} \alpha_n^2 < \infty$. Under suitable conditions, it has been shown that a system of this form

converges in the L^2 sense [97] as well as almost sure [23]. It is well known that $(X_n)_{n\in\mathbb{N}}$ can be seen as a noisy approximation of an ordinary differential equation (ODE)

$$\dot{x} = \overline{f}(x) \qquad \text{where} \qquad \overline{f}(x) := \mathbb{E}[Z_n \mid X_n = x].$$

Based on this relation, it is a classical technique to study the convergence of the stochastic iteration using the asymptotic properties of ODE, see [26]. Over the last half-century, the theory of stochastic approximation with vanishing stepsize has been well developed and is covered in multiple books [18, 27, 78].

In this chapter, we shift our attention to the setting of constant stepsize approximation which has more recently seen a spark of interest in the scientific community due to its simplicity, performance, and fast convergence. A growing line of work has been dedicated to this setting. For a few pointers see [21, 41, 79, 101]. To set the general framework for these notes, we are interested in the case when the dynamics of the stochastic process $X_n^{(\alpha)}$ follow the evolution equation

$$X_{n+1}^{(\alpha)} := X_n^{(\alpha)} + \alpha f(X_n^{(\alpha)}, Y_n^{(\alpha)}) \in \mathbb{R}^d \tag{7.1}$$

for $n \in \mathbb{N}$ and fixed α. Here, $(X_n^{(\alpha)})_{n\in\mathbb{N}}$ takes values in a compact subset $\mathcal{X} \subset \mathbb{R}^D$ and $(Y_n^{(\alpha)})_{n\in\mathbb{N}}$ being the respective noise sequence, taking values in a finite state space $\mathcal{Y}$ independent of α, and $f : \mathcal{X} \times \mathcal{Y}$ is assumed to be a deterministic function. For comparison, in the classical setting of Robbins and Monro [97], Y_n takes i.i.d. values such that $\mathbb{E}[f(x, Y_n)] = 0, x \in \mathcal{X}$.

7.2 Setting

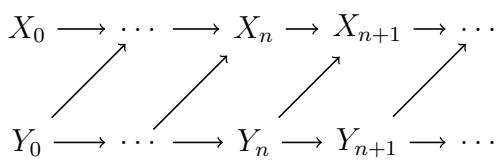
As our results are related to the work [69], in which the authors study the bias of linear stochastic approximation with constant stepsize and Markovian noise data, we will point out the differences in the setup and further generalizations. Before stating the main results, we introduce the underlying framework and dynamics of the stochastic and deterministic systems. We consider the case where the transitions of the noise $Y_n^{(\alpha)}$ at time $k \in \mathbb{N}$ can depend on the state of $X_n^{(\alpha)}$ and have the Markov property, i.e.,

$$\mathbb{P}(Y_{n+1}^{(\alpha)} = y' \mid Y_n^{(\alpha)} = y, X_n^{(\alpha)} = x). \tag{7.2}$$

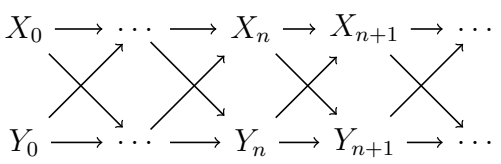
The considered type of noise dynamics is one of the striking differences between the setup in this chapter and the works of [69]. To illustrate this, we compare the two dependency graphs in Table 7.1. To study the asymptotic properties of X_n it is useful to introduce the 'fixed-x' process. This processes is given by the Markov chain $(Y_n(x))_{k\in\mathbb{N}}$ which has the transitions

$$\mathbb{P}(Y_{n+1}^{(\alpha)} = y' \mid Y_n^{(\alpha)} = y, X_n^{(\alpha)} = x) =: P_{y,y'}(x), \tag{7.3}$$

i.e., the Markov chain arising if the value of $X_n^{(\alpha)}$ is held constant at x. We will throughout assume that $P(x)$ has a unique irreducible class with stationary distribution $\pi(x) = (\pi_y(x))_{y\in\mathcal{Y}}$ for all $x \in \mathcal{X}$. This compares to the continuous-time Markov chain induced by the transition kernel given



Markovian Noise, as considered in [69]



state-dependent Markovian Noise, as considered in this chapter.

Table 7.1: Dependency Graphs

in Equation (6.3) of our paper [4]. The induced Markov chain similarly depends on a fixed state of the 'slow' process. The transition matrix and its corresponding steady-state distribution will become important to obtain computable expressions for the bias later on in the proof. It is a well known fact, e.g. see [78], that the ODE characterizing the behavior of X_n for small α is given by

$$\frac{d}{dt}\phi_t(x) = \overline{f}(\phi_t(x)) \quad \text{with} \quad \phi_0(x) = x. \tag{7.4}$$

We call $\overline{f}(x)$ the average drift in x which is defined by

$$\overline{f}(x) := \sum_{y \in \mathcal{Y}} \pi_y(x) f(x, y). \tag{7.5}$$

To ensure that the ODE is well defined, we impose that the average drift $\overline{f}$ is Lipschitz continuous. We further assume that ϕ_t is globally, asymptotically stable, and around its equilibrium point exponentially stable. For sufficiently large n, X_n will be approximated by $\phi_{n\alpha}$. As in our case, the noise admits Markovian and state-dependent behavior ϕ_t admits a non-zero asymptotic bias.

The detailed analysis of the bias is the key goal of this chapter. We study $\mathbb{E}[X_n] - \phi_{n\alpha}$ for finite n in Theorem 41 and in the second Theorem 42 the bias of ϕ_∞ given $\frac{1}{N}\sum_{n=1}^{N} \mathbb{E}[h(X_n^{(\alpha)})]$ as N tends to infinity. Our first results show that the bias can be bounded by a constant αC which holds for any $n \in \mathbb{N}$ if α is small enough and if both systems start from the same initial condition. The second theorem, shows that it is possible to obtain a computable bias representation V_h which depends on $P, \phi_\infty, \pi(\phi_\infty)$ and f but is independent of α.

At last, we point out some differences between our results and the ones obtained in [69]. First, we allow f to be a generic function, whereas the authors of [69] consider the case of linear stochastic approximation. Second, we consider *state-dependent* Markovian noise, as illustrated in Table 7.1 and allow periodicity. Finally, we consider the difference between $h(X_n)$ and $h(\phi_\infty)$ for any thrice differentiable function h. For this setting, we obtain accuracy results for the bias and a computable bias first-order expression. These types of results are comparable to the ones we have obtained in [4] and expand what has been shown in [69] to more intricate noise settings.

7.3 Assumptions

We summarize the necessary assumptions:

- $\mathbf{A}_1$ Let $h : \mathcal{X} \to \mathbb{R}$ be thrice continuously differentiable, with bounded and Lipschitz continuous derivatives.
- $\mathbf{A}_2$ Let $(X_n^{(\alpha)})_{n\in\mathbb{N}}$ follow the dynamics as in Equation (7.1) with deterministic initial value $x_0 \in \mathcal{X}$ and let it take values in $\mathcal{X}$, a compact subset of $\mathbb{R}^d$.
- $\mathbf{A}_3$ Let $(Y_n(x))_{n\in\mathbb{N}}$ be the 'fixed-x' discrete time Markov chain with transition matrix $(P_{y,y'}(x))_{y,y'\in\mathcal{Y}}$ as defined in Equation (7.3) and unique steady-state distribution $(\pi_y(x))_{y\in\mathcal{Y}}$. Furthermore, let $P(x)$ be twice continuously differentiable in x.
- $\mathbf{A}_4$ Let $(\phi_t(x))_{t\in\mathbb{N}}$ with initial condition x be globally asymptotically stable and exponentially stable in a neighborhood $\mathcal{N}$ around its equilibrium point ϕ_∞. Furthermore, let $\overline{f}$, as defined in Equation (7.5), be thrice continuously differentiable with bounded derivatives.

7.4 Main Results

Theorem 41. *Assume* $\mathbf{A}_1$ - $\mathbf{A}_4$ *and let* $0 < \alpha \ll 1$ *be small enough and* $n \in \mathbb{N}$, $x_0 \in \mathcal{X}$. *Then there exists a constant* $\tilde{C}' > 0$ *such that*

$$\left|\mathbb{E}[h(X_n^{(\alpha)})] - h(\phi_{\alpha n}(x_0))\right| \leq \alpha\tilde{C}'$$

with $X_0^{(\alpha)} = x_0$.

Proof. The theorem is a direct consequence of Lemma 44. □

The theorem shows that the bias of the expected state of the stochastic system is of order α with respect to ϕ_n.

Theorem 42. *Assume that* $\mathbf{A}_1$ - $\mathbf{A}_4$ *hold. Then there exists a computable constant* V_h *independent of* α *and* n, *and a constants* $C' > 0$ *such that for small enough* $0 < \alpha \ll 1$ *and* $x_0 \in \mathcal{X}$:

$$\limsup_{N\to\infty}\left|\frac{1}{N}\sum_{n=1}^{N}\mathbb{E}[h(X_n^{(\alpha)})] - h(\phi_\infty) - \alpha V_h\right| \leq \alpha^2 C'. \tag{7.6}$$

Proof. The proof is postponed to Section 7.5. □

Our second theorem states that for the averaged iterates of the expectation of $X_n^{(\alpha)}$, i.e., $\frac{1}{N}\sum_{n=1}^{N}\mathbb{E}[X_n^{(\alpha)}]$, the bias of $h(\phi_\infty)$ has computable first order extension which, aside from the prefactor α, is independent of the stepsize. We consider the averaged iterates as it eliminates eventual oscillations caused by periodic noise sequences, we illustrate this in Figure 7.1. In Lemma 46 we state how the bias expression can be computed. To illustrate the results of Theorem 43, we consider

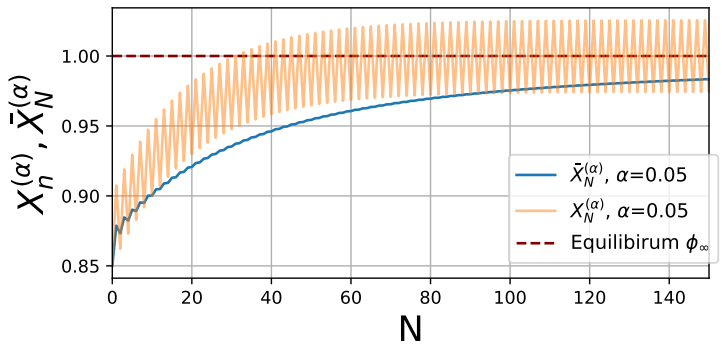


Figure 7.1: A sample trajectory of $X_n^{(\alpha)}$ with periodic $\boldsymbol{Y}_n^{(\alpha)}$ illustrating the necessity of averaging the iterates. For comparison the values of $\bar{X}_N^{(\alpha)} := \frac{1}{N}\sum_{n=1}^{N} X_n^{(\alpha)}$ are plotted.

a short example with drift and noise transition matrix defined as

$$f(x,y) := -x + 2y \qquad \text{and} \qquad P(x) = \begin{pmatrix} \sin(x)^2 & \cos(x)^2 \\ \cos(x)^2 & \sin(x)^2 \end{pmatrix}.$$

$X_n^{(\alpha)}$ is as in Equation 7.1 and the transition of $Y_n^{(\alpha)} \in \{0,1\}$ are given by $P(X_{n-1}^{(\alpha)})$. In Figure 7.2 we plot the values of $\frac{1}{N}\sum_{n=1}^{N} X_n^{(\alpha)}$ for increasing N for the stochastic system with initial state $X_0^{(\alpha)} = 0, Y_0^{(\alpha)} = 0$. As we expect by the statement of the theorems, for diminishing α and increasing T, the bias of $\frac{1}{N}\sum_{n=1}^{N} X_n^{(\alpha)}$ approaches the value of the equilibrium ϕ_∞.

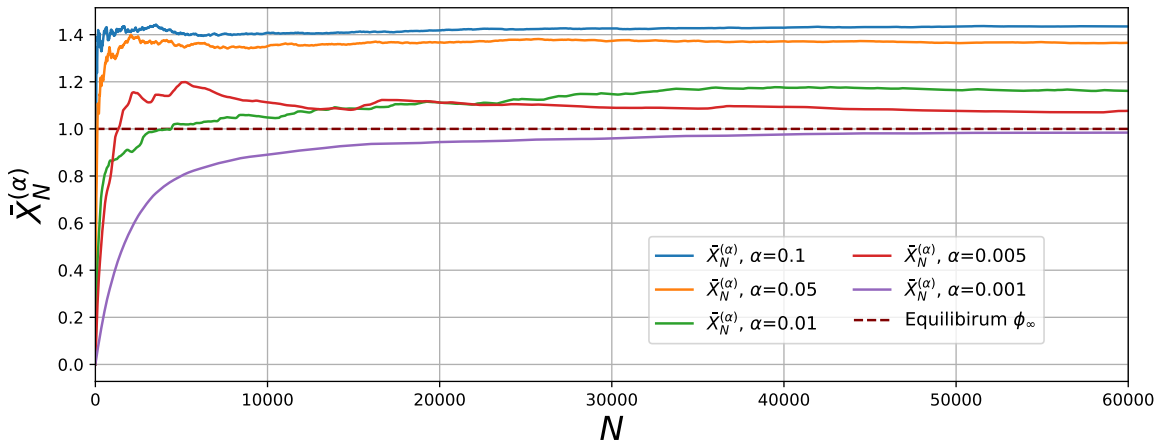


Figure 7.2: Illustration of $\bar{X}_N^{(\alpha)} := \frac{1}{N}\sum_{n=1}^{N} X_n^{(\alpha)}$ for different values of $\alpha = 0.1, \ldots, 0.001$. The figure gives an impression of the relation between the stepsize α and the size of the bias of ϕ_∞.

7.5 Proof of Theorem 42

For the proofs, we introduce the Euler discretization of the ODE with stepsize α

$$\varphi_{n+1}^{(\alpha)}(x) = \varphi_n^{(\alpha)}(x) + \alpha \overline{f}(\varphi_n^{(\alpha)}(x)). \tag{7.7}$$

To distinguish the discretization from the ODE, we will use k, n exclusively to denote the iterations for discrete systems and mark the dependence on the stepsize α in the superscript.

The proof of Theorem 42 is based on the statements of Lemma 43 which we present next. Subsequently, we show how the theorem is obtained from the lemma.

Lemma 43. *Assume that* $\mathbf{A}_1$ *-* $\mathbf{A}_4$ *hold. Then there exists a computable constant* V_h *independent of* α *and* n*, and a constants* $C', C'' > 0$ *such that for small enough* $0 < \alpha \ll 1$*:*

$$\limsup_{N\to\infty} \left| \frac{1}{T} \sum_{n=N+1}^{N+T} \mathbb{E}[h(X_n^{(\alpha)})] - h(\phi_\infty) - \alpha V_h \right| \leq \alpha^2 C' + \frac{1}{T} C''. \tag{7.8}$$

Proof. The statement of the theorem is a consequence of Theorem 44, Lemma 47 and Lemma 45. □

Proof of Theorem 42. To proof the statement of the theorem, define

$$u_n = |E[h(X_n)] - h(\phi_\infty) - \alpha V_h| - \alpha^2 C'$$

with h, X_n, ϕ_∞, V_h as in Lemma 43. By the same lemma we have that for all $T \in \mathbb{N}$

$$\lim_{N\to\infty} \frac{1}{T} \sum_{k=N+1}^{N+T} u_k \leq \frac{C''}{T}. \tag{7.9}$$

Define $s_k = \frac{1}{K} \sum_{m=1}^{K} u_{m+(k-1)K}$ for $k \geq 1$ which is average sum over K consecutive terms of u starting in $(k-1)K$. Now let $\varepsilon > 0$ and set $K = \lfloor C/\varepsilon \rfloor$. By Equation (7.9), we have that $\lim_{k\to\infty} v_k \leq \varepsilon$, which further implies $\lim_{T\to\infty} (1/T) \sum_{k=1}^{T} v_k \leq \varepsilon$ as a consequence of the Cesaro Lemma. To finish the proof, separate the sum

$$\frac{1}{T} \sum_{k=1}^{T} u_k = \frac{K}{T} \sum_{k=1}^{\lfloor T/K \rfloor} v_k + \frac{1}{T} \sum_{k=\lfloor T/K \rfloor + 1}^{T} u_k. \tag{7.10}$$

The second term of the right-hand-side of (7.10) is bounded by K/T. The second quantity is smaller than ε as T tends to infinity. As by assumption this is true for all $\varepsilon > 0$, it holds that

$$\lim_{T\to\infty} \frac{1}{T} \sum_{k=1}^{T} u_k \leq 0 \tag{7.11}$$

which proves the statement of the Theorem. □

7.6 Lemmas for Bias Analysis

The lemma shows that the difference between the stochastic system and deterministic can be separated into a bias term which is bounded and of order α and a residual term of order α^2. The theorem furthermore gives a characterization of the bias term. In the main results, we will see that, when considering an averaged bias expression, it is possible to obtain a computable expression that approximates the bias.

Lemma 44. *Assume* $\mathbf{A}_1$ *-* $\mathbf{A}_4$ *and let* $0 < \alpha \ll 1$ *be small enough and* $n \in \mathbb{N}$*. Then*

$$\left|\mathbb{E}[h(X_n^{(\alpha)})] - h(\phi_{\alpha n}(x)) - \alpha V_{h,n}^{\alpha}\right| \leq \alpha^2 \tilde{C}$$

with $X_0^{(\alpha)} = x_0$ *and*

$$\begin{aligned} V_{h,n}^{(\alpha)} := \sum_{k=0}^{n} \mathbb{E}[D(h \circ \varphi_k)\big(f(X_{n-k-1}, Y_{n-k-1}) - \overline{f}(X_{n-k-1}) \\ - \alpha \sum_{k=0}^{n} D^2(h \circ \varphi_k)(X_{n-k-1}) \cdot Q(X_{n-k-1}, Y_{n-k-1})]. \end{aligned}$$

and $Q(x,y) := \big(f(x,y) - \overline{f}(x)\big)^{\otimes 2}$*. Here,* φ_k *is as defined in Equation (7.7). Under the given assumptions,* $V_{h,n}^{\alpha}$ *is bounded independent of* α *and* n*.*

Proof of Lemma 44. In the proof we omit the superscript $X_k^{(\alpha)}, Y_k^{(\alpha)}$ when the dependence on α is clear from context. Next, define $\nu_{n,k} = \mathbb{E}[h(\varphi_k(X_{n-k}))]$ with $\varphi_k(X_{n-k})$ denoting the value of the k-th iteration of φ with initial state X_{n-k}. This is similar to the method used in Chapter 2 with ν being comparable to the quantity defined in Equation (2.14). Using $\nu_{n,k}$ we rewrite

$$\mathbb{E}[h(X_n)] - h(\varphi_n) = \nu_{n,0} - \nu_{n,n} = \sum_{k=0}^{n-1} \nu_{n,k} - \nu_{n,k+1}. \tag{7.12}$$

with the difference terms being

$$\begin{aligned} \nu_{n,k} - \nu_{n,k+1} &= h(\varphi_k(X_{n-k})) - h(\varphi_{k+1}(X_{n-(k+1)})) && (7.13) \\ &= h\Big(\varphi_k\big(X_{n-(k+1)} + \alpha f(X_{n-(k+1)}, Y_{n-(k+1)})\big)\Big) && (7.14) \\ &\quad - h\Big(\varphi_k\big(X_{n-(k+1)} + \alpha \overline{f}(X_{n-(k+1)})\big)\Big). && (7.15) \end{aligned}$$

For the summed difference $\sum_{k=0}^{n-1} \nu_{n,k} - \nu_{n,k+1}$, we look at the Taylor expansion of the difference $\nu_{n,k} - \nu_{n,k+1}$. For the second order Taylor expansion define

$$\alpha\, \mathbb{E}[D(h \circ \varphi_k)(X_{n-k-1})(f(X_{n-k-1}, Y_{n-k-1}) - \overline{f}(X_{n-k-1}))] =: \alpha\, a_{n,k}^{(\alpha)} \tag{7.16}$$

from the linear part of the expansion and

$$\begin{aligned}
&\frac{\alpha^2}{2}\mathbb{E}[\sum_{i,j} D_x^2(h \circ \varphi_k)_{ij}(X_{n-k-1})(f_i(X_{n-k-1}, Y_{n-k-1}) - \overline{f}_i(X_{n-k-1})) \\
&\qquad\qquad \times (f_j(X_{n-k-1}, Y_{n-k-1}) - \overline{f}_j(X_{n-k-1}))] \\
&= \frac{\alpha^2}{2}\mathbb{E}[D_x^2(h \circ \varphi_k)(X_{n-k-1}) \cdot (f(X_{n-k-1}, Y_{n-k-1}) - \overline{f}(X_{n-k-1}))^{\otimes 2}] =: \frac{\alpha^2}{2} b_{n,k}^{(\alpha)}
\end{aligned} \tag{7.17}$$

from the quadratic part. Here, $\otimes$ refers to the Kronecker product and we use $\cdot$ to denote summed multiplication of the components for the matrices. Using the regularity assumptions posed on φ_k and its derivatives, the remainder terms have the form

$$\begin{aligned}
&\frac{\alpha^3}{2}\int_0^1 (1-\nu)^2 \sum_{i,j,k} \frac{\partial^3 (h \circ \varphi_k)}{\partial x_i \partial x_j \partial x_k}(X_{n-k-1} + \nu\alpha f(X_{n-k-1}, Y_{n-k-1}))f(X_{n-k-1}, Y_{n-k-1})_i \\
&\qquad \times f(X_{n-k-1}, Y_{n-k-1})_j f(X_{n-k-1}, Y_{n-k-1})_k d\nu \leq \frac{\alpha^3}{2} C_{D_x^3(h\circ\varphi_k)} 3 C_f C_R
\end{aligned}$$

for Equation (7.14) and similarly for the remainder of Equation (7.15) where f is replaced by $\overline{f}$. The constants refer to the bounds the derivative of $h \circ \varphi_k$, f and the finite residual term arising through the remainder. We recall that by Lemma 49 there exists a $\hat{C} > 0$ independent of α and n such that $\alpha \sum_{k=0}^{n-1} C_{D_x^3(h\circ\varphi_k)} \leq C$. Finally, this shows that the there exists some $C_0 > 0$, such that sum of the remainder terms obtained from the Taylor expansion of the Equations (7.14) and (7.15) is bounded by $\alpha^2 C_0$. Now, define $V_{h,n}^{(\alpha)} = \sum_{k=0}^{n-1}(a_{n,k}^{(\alpha)} + \alpha b_{n,k}^{(\alpha)})$, then

$$\left| h(X_n) - h(\varphi_n) - \alpha V_{h,n}^{(\alpha)} \right| \leq \alpha^2 C_0$$

which holds for all n.
We will now show that the terms $\sum_{k=0}^{n-1} a_{n,k}^{(\alpha)}$ and $\sum_{k=0}^{n-1} \alpha b_{n,k}^{(\alpha)}$ are bounded and of order $O(\alpha)$. We start with the bound of $\sum_{k=0}^{n-1} \alpha b_{n,k}^{(\alpha)}$. By Lemma 49,

$$\begin{aligned}
&\left| \sum_{k=0}^{n-1} \mathbb{E}[D_x^2(h \circ \varphi_k)(X_{n-k-1}) Q(X_{n-k-1}, Y_{n-k-1})] \right| \\
&\qquad \leq \alpha C_Q \underbrace{\sum_{k=0}^{\infty} C_{D_x^2(h\circ\varphi_k)}}_{O(1/(\alpha - \frac{\alpha^2}{2}))} =: C_2.
\end{aligned}$$

By application of the first statement of Lemma 50 we get $\sum_{k=0}^{n-1} a_{n,k}^{(\alpha)} \leq C_1$ with $C_1 > 0$ as in the Lemma. This directly implies for $\tilde{C}' := C_1 + C_2$ and by definition of $V_{h,n}^{(\alpha)}$

$$|\mathbb{E}[h(X_n)] - h(\varphi_n)| \leq \alpha \tilde{C}' + \alpha^2 C_0.$$

□

Lemma 45. *Define*

$$\begin{aligned} V_h^\alpha := \alpha \sum_{k=0}^{\infty} D_x(h \circ \varphi_k)(\phi_\infty) \sum_{y,y'} D_x G_f(\phi_\infty, y') f(\phi_\infty, y)^T (I + K(\phi_\infty))_{y,y'} \pi_y(\phi_\infty) \\ + \alpha \sum_{k=0}^{\infty} D_x^2(h \circ \varphi_k)(\phi_\infty) \cdot \Big(\sum_{y,y'} G_f(\phi_\infty, y') f(\phi_\infty, y)^T (I + K(\phi_\infty))_{y,y'} \pi_y(\phi_\infty) \\ + \frac{1}{2} \bar{Q}(\phi_\infty) \Big). \end{aligned} \tag{7.18}$$

Here, φ is as defined in Equation (7.7) and depends on α. It follows from Lemma 47 that there exist constants $C', C'' > 0$ such that

$$\left| \alpha \frac{1}{T} \sum_{n=N+1}^{N+T} V_{h,n}^{(\alpha)} - \alpha V_h^{(\alpha)} \right| \leq \frac{1}{T} C' + \alpha^2 C''. \tag{7.19}$$

Proof. To see that Lemma 47 can be applied to obtain the result, we see that G_{g_k} of Equation (7.26) can be rewritten as

$$\begin{aligned} D_x G_{g_k}(\phi_\infty, y) f(\phi_\infty, y') &= G_f(\phi_\infty, y)^T D_x^2(h \circ \varphi_k)(\phi_\infty) f(\phi_\infty, y') \\ &\quad + D_x(h \circ \varphi_k)(\phi_\infty) D_x G_f(\phi_\infty, y) f(\phi_\infty, y') \\ &= D_x^2(h \circ \varphi_k)(\phi_\infty) \circ (G_f(\phi_\infty, y) f(\phi_\infty, y')^T \\ &\quad + D_x(h \circ \varphi_k)(\phi_\infty) D_x G_f(\phi_\infty, y) f(\phi_\infty, y'). \end{aligned}$$

This follows by definition of g_k and G_{g_k}. To conclude, we just apply Lemma 47 with the prefactor α. □

7.6.1 Accuracy and Computation of the Asymptotic Bias V_h

Lemma 46. *Under the assumptions* $\mathbf{A}_1$ *-* $\mathbf{A}_4$*, the constant* $V_h^{(\alpha)}$ *as defined in Equation* (7.18) *of Lemma* 45 *is approximated by a computable constant* V_h*. It holds that there exists a constant* $C' > 0$ *such that*

$$\left| V_h - V_h^{(\alpha)} \right| \leq \alpha C'.$$

V_h is given as follows:
Define $A_{i,j} := \frac{\partial \overline{f}}{\partial x_i}(\phi_\infty)$*, the Jakobian matrix of* $\overline{f}$ *in* ϕ_∞*, and* $B_{i,jk} = \frac{\partial \overline{f}_i}{\partial x_i \partial x_j}(\phi_\infty)$ *the second derivative. The* V_h *is defined by:*

$$V_h = D_x h(\phi_\infty) A^{-1}(S + B \cdot W) + D_x^2 h(\phi_\infty) \cdot W \tag{7.20}$$

with $B \cdot W := (\sum_{j,k} B_{i,jk} W_{j,k})_i$*. Here,* W *is the unique solution to the Sylvester equation* $AW + WA^T + O = 0$*.* O *and* S *are defined by*

$$O := \sum_{y,y'} \pi_y(\phi_\infty) G_f(\phi_\infty, y') f(\phi_\infty, y)^T (K(\phi_\infty) + I)_{y,y'} + \frac{1}{2} \bar{Q}(\phi_\infty),$$
$$S := \sum_{y,y'} \pi_y(\phi_\infty) f(\phi_\infty, y) (K(\phi_\infty) + I)_{y,y'} DG_f(\phi_\infty, y')^T$$

with $\bar{Q}(\phi_\infty) := \sum_y \pi_y(\phi_\infty)(f(x,y) - \overline{f}(x))(f(x,y) - \overline{f}(x))^T$ *and* $K(x) := P(x) - I$*.*

Proof. To start, use the definitions of O and S from the Theorem to rewrite Equation (7.18). As we are only interested in the computation of $V_h^{(\alpha)}$ for $t = \infty$, we ignore the prefactor α and thus have the equations

$$V_h^{(\alpha)} = \Big(\sum_{k=0}^{\infty} D_x(h \circ \varphi_k)(\phi_\infty) S + \sum_{k=0}^{\infty} D_x^2(h \circ \varphi_k)(\phi_\infty) \cdot O \Big). \tag{7.21}$$

with φ as defined in Equation (7.7). To ease notation, we turn away from writing the dependence on ϕ_∞ in the next steps. Using that $\varphi_k(\phi_\infty) = \phi_\infty$, the right hand side of (7.21) without the prefactor α is equal to

$$D_x(h \circ \varphi_k) S + D_x^2(h \circ \varphi_k) \cdot O \tag{7.22}$$
$$= D_x h D_x \varphi_k S + D_x^2 h \cdot D_x \varphi_k O D_x \varphi_k^T + D_x h D_x^2 \varphi_k \cdot O \tag{7.23}$$

with $D_x h D_x^2 \varphi = (\sum_i \frac{\partial h}{\partial x_i} \frac{\partial \varphi_i}{\partial x_m \partial x_n})_{m,n}$ and $\cdot$ denoting the sum over the element wise product between the matrices. The terms S and O are computable due to Lemma 28 which gives a computable expression for G_f and Note 40 of [4] showing how to efficiently obtain the derivative values for

G_f.[1] Both references point to our two timescale mean field paper for continuous time [4] but can be adapted to the discrete time case. Therefore, in what follows, we are concerned about obtaining computable expressions for the remaining infinite sums

$$\sum_{k=0}^{\infty} D_x\varphi_k; \qquad \sum_{k=0}^{\infty} D_x\varphi_k O D_x\varphi_k^T; \qquad \sum_{k=0}^{\infty} D_x^2\varphi_k \cdot O.$$

Starting with the first sum, we recall that by definition $D_x\varphi_1 = I + \alpha A$. Using the chain rule, we have $D_x\varphi_k = D_x(\varphi_1 \circ \varphi_{k-1}) = (I + \alpha A)D_x\varphi_{k-1} = (I + \alpha A)^k$. As A is Hurwitz and for small enough α, $\sum_{k=0}^{\infty}(I+\alpha A)^k$ is finite and we obtain $\sum_{k=0}^{\infty}(I+\alpha A)^k = (\alpha A)^{-1}$ as a standard result for geometric series. For the second sum we apply the identity for $D_x\varphi_k$ which yields

$$\sum_{k=0}^{\infty} D_x\varphi_k O D_x\varphi_k^T = \sum_{k=0}^{\infty}(I + \alpha A)^k O D_x(I + \alpha A^T)^k =: W^{(\alpha)}.$$

As before, due to the Hurwitz property of A and small enough α, $\|I + \alpha A\| < 1$ and which implies well definedness of the above. As we aim to obtain a computable expression independent of α, we look at the continuous-time Sylvester equation having the form $AX - XA^T + O = 0$. By the definition of A and O the equation has a unique solution which we denote W. As φ_k is a discretized version of the ODE (7.4), it is a standard property that the solution to the discrete time Sylvester $W^{(\alpha)}$ can be written as $W^{(\alpha)} = W + \alpha\tilde{C}_L$, where $\tilde{C}_L$ is obtained by bounding $\left\|AOA^T\right\|$. For the last sum, we see that by using the chain rule as before

$$\begin{aligned} D_x^2\varphi_k &= \left(\frac{\partial\varphi_k}{\partial x_m \partial x_n}\right)_{i,mn} \\ &= \left(\sum_{j=1}^{k}\sum_{a}\left((I + \alpha A)^{k-j}\right)_{i,a}\sum_{b,c}\alpha B_{a,bc}\left((I + \alpha A)^j\right)_{b,m}\left((I + \alpha A)^j\right)_{c,n}\right)_{i,mn}. \end{aligned}$$

Therefore,

$$D^2\varphi_k \cdot O = \sum_{j=1}^{k}(I + \alpha A)^{k-j}\alpha B \cdot (I + \alpha A)^j O(I + \alpha A^T)^j.$$

[1] To see that the statement of Lemma 28 remains true for discrete time, one can simply adapt the proof by replacing the statement of Theorem 3.5 of [68] by Theorem 3.3 with $K(x)$ as defined before in this section. The rest of the proof is then completed as in the continuous time case.

where we abuse notation and denote by $\cdot$ the sum over the element-wise multiplication of the partial derivatives for a fixed index of B, i.e., $\sum_{m,n} B_{i,mn}\big((I+\alpha A)^j O(I+\alpha A^T)^j\big)_{m,n}$. Inserting this into the sum from $k=0$ to infinity yields

$$\begin{aligned}
&\sum_{k=0}^{\infty}\sum_{j=1}^{k}(I+\alpha A)^{k-j}\alpha B\cdot(I+\alpha A)^j O(I+\alpha A^T)^j \\
&\quad=\sum_{j=1}^{\infty}\sum_{k=j}^{\infty}(I+\alpha A)^{k-j}\alpha B\cdot(I+\alpha A)^j O(I+\alpha A^T)^j \\
&\quad=\sum_{j=1}^{\infty}\sum_{k=0}^{\infty}(I+\alpha A)^{k}\alpha B\cdot(I+\alpha A)^j O(I+\alpha A^T)^j \\
&\quad=\sum_{j=1}^{\infty}(\alpha A)^{-1}\alpha B\cdot(I+\alpha A)^j O(I+\alpha A^T)^j \\
&\quad=(\alpha A)^{-1}\alpha B\cdot(I+\alpha A)\Big(\sum_{j=0}^{\infty}(I+\alpha A)^j O(I+\alpha A^T)^j\Big)(I+\alpha A^T) \\
&\quad=(\alpha A)^{-1}\alpha B\cdot(I+\alpha A)W^{(\alpha)}(I+\alpha A^T) \\
&\quad=A^{-1}B\cdot W^{(\alpha)}+\alpha C'.
\end{aligned}$$

With C' being a bound on the terms arising from the above formula witch have a prefactor α. Using the connection between the discrete and continuous version of the Sylvester Equation, we can write

$$\left\|A^{-1}B\cdot(W^{(\alpha)}-W)\right\|\leq\alpha C_L.$$

To finalize the proof, recall the right-hand side of Equation (7.23). As we have obtained approximate and computable expressions for $D_x\varphi_k$, $D_x\varphi_k O D_x\varphi_k$ and $D_x^2 x\varphi_k$ we use those to define

$$V_h=D_x h(\phi_\infty)A^{-1}(S+B\cdot W)+D_x^2 h(\phi_\infty)\cdot W. \tag{7.24}$$

By the previously discussed accuracy of the solution to the Sylvester equation W, we know that $\left|V_h^{(\alpha)}-V_h\right|\leq\alpha(C_L+C')=:\alpha C'$. □

7.6.2 Expansion Term Approximation Bound

Lemma 47 (Accuracy Bound for the Deterministic Bias Approximation)**.** *Assume* $\mathbf{A}_1$ - $\mathbf{A}_4$, $0 < \alpha \ll 1$ *and* $T \in \mathbb{N}$. *Define* $Q(x,y) := (f(x,y) - \overline{f}(x))^{\otimes 2}$ *and* $\bar{Q}(x) := \sum_y \pi_y(x) Q(x,y)$, *then there exist constants* $C_1', C_1'', C_1''' > 0$ *such that*

$$\begin{aligned}\limsup_{N\to\infty} \Bigg| \alpha \frac{1}{T} \sum_{n=N+1}^{N+T} \frac{1}{2} \sum_{k=0}^{n-1} \mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n+\tau-k}) \cdot Q(X_{n+\tau-k}, Y_{n+\tau-k})] \\ - \alpha \frac{1}{2} \sum_{k=0}^{\infty} D_x^2(h\circ\varphi_k)(\phi_\infty) \cdot \bar{Q}(\phi_\infty) \Bigg| \\ \leq \alpha^2 C_1' + \frac{1}{T} C_1'' + \alpha C_1'''. \end{aligned} \tag{7.25}$$

Second, there exist constants $C_2', C_2'', C_2''' > 0$ *such that*

$$\begin{aligned}\limsup_{N\to\infty} \Bigg| \alpha \frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} \mathbb{E}[D_x(h\circ\varphi_k)(X_{n-k-1})(f(X_{n-k-1}, Y_{n-k-1}) - \overline{f}(X_{n-k-1}))] \\ - \alpha \sum_{k=0}^{\infty} \alpha \sum_{y,y'} D_x G_{g_k}(\phi_\infty, y') f(\phi_\infty, y)(I + K(\phi_\infty))_{y,y'} \pi_y(\phi_\infty) \Bigg| \\ \leq \frac{1}{T} C_2 + \alpha^2 C_2'' \end{aligned} \tag{7.26}$$

with $g_k(x,y) := D_x(h\circ\varphi_k)(x)(f(x,y) - \overline{f}(x))$ *and* φ *as defined in Equation* (7.7).

Proof of Lemma 47. We begin with the proof of Equation (7.25). In the following, the arguments are made for finite n. We will then show that by letting n go to infinity, the bound remains finite. We start with Equation (7.25). The argument to obtain the bound is split into two parts: First we show

$$\begin{aligned}\Bigg| \frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=1}^{n} \mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n-k-1}) \cdot \Big(Q(X_{n-k-1}, Y_{n-k-1}) - \bar{Q}(X_{n-k-1}) \Big)] \Bigg| \\ \leq \alpha C_1' + \frac{1}{T(\alpha - \frac{\alpha^2}{2})} C_1''. \end{aligned} \tag{7.27}$$

Second we show

$$\begin{aligned}\Bigg| \frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} \mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n-k-1}) \cdot \bar{Q}(X_{n-k-1}) \\ - \frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} D_x^2(h\circ\varphi_k)(\varphi_{n-k-1}) \cdot \bar{Q}(\varphi_{n-k-1})] \Bigg| \leq C_1'''. \end{aligned} \tag{7.28}$$

To show Equation 7.27, we argue similar as in Lemma 50 and extend the difference by

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n-k-1})\Big(Q(X_{n-k-1},Y_{n-k-1})-\bar{Q}(X_{n-k-1}) \\ +G_Q(X_{n-k-1},Y_{n-k-1})-G_Q(X_{n-k-1},Y_{n-k}) \\ +G_Q(X_{n-k-1},Y_{n-k})-G_Q(X_{n-k},Y_{n-k}) \\ +G_Q(X_{n-k},Y_{n-k})-G_Q(X_{n-k-1},Y_{n-k-1})\Big)].$$

As discussed in the proof of the lemma,

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n-k-1})\Big(Q(X_{n-k-1},Y_{n-k-1})-\bar{Q}(X_{n-k-1}) \\ +G_Q(X_{n-k-1},Y_{n-k-1})-G_Q(X_{n-k-1},Y_{n-k})\Big)]=0.$$

Again by the same arguments as in Lemma 50, there exists a constant $C'>0$ such that

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n-k-1})\Big(G_Q(X_{n-k-1},Y_{n-k})-G_Q(X_{n-k},Y_{n-k})\Big)\leq\alpha C'$$

Lastly,

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\mathbb{E}[D_x^2(h\circ\varphi_k)(X_{n-k-1})\Big(G_Q(X_{n-k},Y_{n-k})-G_Q(X_{n-k-1},Y_{n-k-1})\Big)] \\ \leq\frac{1}{T(\alpha-\frac{\alpha^2}{2})}C''+\alpha C'''$$

due to Lemma 52 with $C'',C'''>0$. To obtain the bound on Equation (7.28) we use utilize the results of Theorem 44 applied to $h_k(x):=D_x^2(h\circ\varphi_k)(x)\cdot\bar{Q}(x)$ which shows

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}|\mathbb{E}[h_k(X_{n-k-1})-h_k(\varphi_{n-k-1})]|\leq\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\alpha C_{D_x^2\varphi_k}\tilde{C}\leq C''''.$$

In the above, $\tilde{C}$ is the constant arising from Theorem 44 and we use that $\alpha \sum_{k=0}^{n-1} C_{D_x^2 \varphi_k} = O(1)$. This implies the existence of a constant $C'''' < \infty$ independent of α and n serving as upper bound. To finish our line of arguments in order to obtain the bound in Equation (7.27), we lastly show that

$$\begin{aligned}&\sum_{k=0}^{\infty} h_k(\phi_\infty) - \sum_{k=0}^{N} h_k(\varphi_{n-k}) \\ &= \underbrace{\sum_{k=0}^{N} h_k(\phi_\infty) - h_k(\varphi_{N-k})}_{=O(\exp(-\alpha N))} + \underbrace{\sum_{k=N}^{\infty} h_k(\phi_\infty)}_{=O(\exp(-\alpha N))}.\end{aligned}$$

The first property is due to the exponentially fast convergence of φ_n to ϕ_∞ for sufficiently large n, the second one a consequence of Lemma 49. By using the above results, and defining $C_1' = C' + C'''$, $C_1'' = C''$ and $C_1''' = C''''$ finally gives

$$\begin{aligned}\limsup_{N\to\infty} \Bigg| \alpha \frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} \mathbb{E}[D_x^2(h \circ \varphi_k)(X_{n+\tau-k}) \cdot Q(X_{n+\tau-k}, Y_{n+\tau-k})] \\ - \sum_{k=0}^{\infty} D_x^2(h \circ \varphi_k)(\phi_\infty) \cdot \bar{Q}(\phi_\infty) \Bigg| \leq \alpha^2 C_1' + \frac{1}{T} C_1'' + \alpha C_1'''.\end{aligned}$$

This concludes the first part of the proof.
For the second part, we start by defining

$$g_k(x, y) = D_x(h \circ \varphi_k)(x)(f(x, y) - \overline{f}(x)).$$

By definition $\bar{g}_k(x) := \sum_y \pi_y(x) g_k(x, y) = 0$ for all $x \in \mathcal{X}$. We can furthermore rewrite

$$\begin{aligned}&\frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} \mathbb{E}[D_x(h \circ \varphi_k)(X_{n-k-1}) \Big(f(X_{n-k-1}, Y_{n-k-1}) - \overline{f}(X_{n-k-1}) \Big) \\ &= \frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} \mathbb{E}[g_k(X_{n-k-1}, Y_{n-k-1}) - \bar{g}_k(X_{n-k-1})].\end{aligned}$$

Similar to what we have done for the first statement of the lemma, we look at the extended difference of $g_k(X_{n-k-1}, Y_{n-k-1}) - \bar{g}_k(X_{n-k-1})$ in form of

$$\begin{aligned}g_k(X_{n-k-1}, Y_{n-k-1}) - \bar{g}_k(X_{n-k-1}) &+ G_{g_k}(X_{n-k-1}, Y_{n-k-1}) - G_{g_k}(X_{n-k-1}, Y_{n-k}) \\ &+ G_{g_k}(X_{n-k-1}, Y_{n-k}) - G_{g_k}(X_{n-k}, Y_{n-k}) \\ &+ G_{g_k}(X_{n-k}, Y_{n-k}) - G_{g_k}(X_{n-k-1}, Y_{n-k-1}).\end{aligned}$$

As before we argue that the first line is zero. Therefore, we restrict our attention to the remaining terms. For the last line, we use that

$$G_{g_k}(x,y) = D(h\circ\varphi_k)(x)G_f(x,y).$$

Then by summing over $\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}$ and using Lemma 52 we obtain

$$\left|\frac{1}{T}\sum_{N+1}^{N+T}\sum_{k=0}^{n-1} G_{g_k}(X_{n-k},Y_{n-k}) - G_{g_k}(X_{n-k-1},Y_{n-k-1}\right| \leq \frac{1}{T(\alpha-\frac{\alpha^2}{2})}\tilde{C}' + \alpha\tilde{C}''.$$

For the remaining term

$$G_{g_k}(X_{n-k-1},Y_{n-k}) - G_{g_k}(X_{n-k},Y_{n-k})$$

we have a look at the Taylor expansion of $G_{g_k}(X_{n-k},Y_{n-k})$ around X_{n-k-1}. This gives us the identity

$$\begin{aligned}&G_{g_k}(X_{n-k-1},Y_{n-k}) - G_{g_k}(X_{n-k},Y_{n-k})]\\ &= \alpha D_x G_{g_k}(X_{n-k-1},Y_{n-k})f(X_{n-k-1},Y_{n-k-1}) + \alpha^2 R_{1,G_{g_k}}.\end{aligned}$$

with R_1 denoting the remainder excluding the alpha term. We have now split the equation into two parts

$$\frac{1}{N}\sum_{n=T_1}^{T+N}\sum_{k=0}^{n-1}\alpha\mathbb{E}[D_xG_{g_k}(X_{n-k-1},Y_{n-k})f(X_{n-k-1},Y_{n-k-1})] + \underbrace{\sum_{k=0}^{n-1}\alpha^2 C_{D_x(h\circ\varphi_k)}R_{1,G_f}}_{\leq\alpha\tilde{C}'''}.$$

Here, $\tilde{C}''' < 0$ is independent of n or α. The remaining parts of the proof concern bounding the difference

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\mathbb{E}[\alpha D_xG_{g_k}(X_{n-k-1},Y_{n-k})f(X_{n-k-1},Y_{n-k-1})] \tag{7.29}$$

$$-\sum_{k=0}^{\infty}\alpha\sum_{y,y'}D_xG_{g_k}(\phi_\infty,y')f(\phi_\infty,y)(I+K(\phi_\infty))_{y,y'}\pi_y(\phi_\infty). \tag{7.30}$$

We see that for the conditional expectation

$$\begin{aligned}&\alpha\mathbb{E}[D_xG_{g_k}(X_{n-k-1},Y_{n-k})f(X_{n-k-1},Y_{n-k-1}) \mid X_{n-k-1},Y_{n-k-1}]\\ &= \alpha\sum_{y'}D_xG_{g_k}(X_{n-k-1},y')f(X_{n-k-1},Y_{n-k-1})(I+K(X_{n-k-1}))_{Y_{n-k-1},y'}.\end{aligned}$$

To proceed, it is necessary to look closer at $D_x G_{g_k}(x,y)$ which we can rewrite by definition of g_k as

$$D_x G_{g_k}(x,y) = D_x\big(D_x(h\circ\varphi_k)(x)G_f(x,y)\big) = D_x^2(h\circ\varphi_k)(x)G_f(x,y) + D_x(h\circ\varphi_k)(x)G_f(x,y).$$

To simplify notation we introduce $m_k(x,y) := \sum_{y'} D_x G_{g_k}(x,y')f(x,y)(I+K(x))_{y,y'}$ and $\bar{m_k}(x) = \sum_y \pi_y(x)m(x,y)$. Considering the identity of the conditional expectation, the identity of G_{g_k} which lets us apply Lemma 50 and using $\bar{m}_k$ we have

$$\begin{aligned}|\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\alpha\mathbb{E}[D_x G_{g_k}(X_{n-k-1},Y_{n-1})f(X_{n-k-1},Y_{n-k-1})]\\ -\alpha\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\bar{m}_k(X_{n-k-1})\Big]| \le \alpha\tilde{C}''''.\end{aligned}$$

We continue to look at the difference, and by the same argument as for the first statement of this lemma

$$\left|\alpha\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\bar{m}_k(X_{n-k-1}) - \alpha\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\bar{m}_k(\varphi_{n-k-1})\right| \le \alpha\tilde{C}'''''$$

. Lastly and again similar to the final remarks of the first part of this proof

$$\left|\alpha\sum_{k=0}^{N}\bar{m}_k(\varphi_{n-k-1}) - \alpha\sum_{k=0}^{\infty}\bar{m}_k(\phi_\infty)\right| = O(\exp(-\alpha N)).$$

It remains to resolve and gather the obtained bounds to conclude the proof. Define $C_2' := \tilde{C}', C_2'' := \tilde{C}'' + \tilde{C}'''' + \tilde{C}''''' + \tilde{C}'''''$. As all the bound we obtained are either independent of n or tend to zero exponentially fast, we obtain the asymptotic bound

$$\begin{aligned}\limsup_{n\to\infty}\Big|\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\mathbb{E}[D_x(h\circ\varphi_k)(X_{n-k-1})(f(X_{n-k-1},Y_{n-k-1}) - \overline{f}(X_{n-k-1}))]\\ -\alpha\sum_{k=0}^{\infty}\sum_{y,y'}D_x G_{g_k}(\phi_\infty,y')f(\phi_\infty,y)(I+K(\phi_\infty))_{y,y'}\pi_y(\phi_\infty)\Big|\\ \le \frac{1}{T(\alpha-\frac{\alpha^2}{2})}C_2' + \alpha C_2''.\end{aligned}$$

Multiplication with an additional α gives the desired result as in the statement of the Lemma. □

7.7 Exponentially Decaying Bounds for $D_x^i \varphi_k(x)$

The statement of the next lemma is foundational to ensure sufficiently fast convergence of the derivative terms $D_x \varphi_k(x)$ to zero. This critical property is continuously referenced in the proof to guarantee the finiteness of the resulting series.

Lemma 48 (Exponentially Small Bounds for Derivatives)**.** *Let $\varphi_k^{(\alpha)}$ be globally asymptotically stable and locally exponentially stable with Lipschitz continuous and continuously differentiable drift f and corresponding Lipschitz constant L_f and α small enough. Then there exists a neighborhood $\mathcal{N}'$ of the equilibrium point ϕ_∞ such that $\varphi_k(x)^{(\alpha)}$s in $\mathcal{N}'$ is Lipschitz continuous with exponentially decaying constant L_k. Furthermore, the derivatives $D_x^i \varphi_k(x)^{(\alpha)}, i = 1, 2, 3$ are bounded with exponentially decaying bounds, i.e., there exist constants $C_{1,i}, C_{2,i} > 0$ such that*

$$\left\| D_x^i \varphi_k^{(\alpha)} \right\| \leq C_{1,i} \exp(-k\alpha C_{2,i}) =: C_{D_x^i \varphi_k}. \tag{7.31}$$

Proof. We extend the proof of Lemma 6.3 of [55] to the discrete time setting. Let $\phi_\infty = 0$ and $\varphi_{k+1}^{(\alpha)} = \varphi_k^{(\alpha)} + \alpha \overline{f}(\varphi_k^{(\alpha)})$. As $\overline{f}$ is continuously differentiable it admits a linear expansion in a neighborhood $\mathcal{N}$ of the equilibrium, i.e., for $x \in \mathcal{N}$

$$\overline{f}(x) = Ax + g(x) \quad \text{with} \quad A = D\overline{f}(\phi_\infty) = D\overline{f}(0) \quad \text{and} \quad \|g(x)\|/\|x\| \xrightarrow{x \to 0} 0.$$

By the stability assumption $\|I + \alpha A\| =: \lambda^{(\alpha)} < 1$. We choose a second bounded neighborhood $\mathcal{N}' \subset \mathcal{N}$ around ϕ_∞ such that $g(x) \leq \delta \|x\|$ and $\lambda^{(\alpha)} \exp(\delta) < 1$. We now show that φ is Lipschitz continuous around the equilibrium point. Assume that $x \in \mathcal{N}'$ then, based on the linear expansion of the drift $f(\varphi_k(x)^{(\alpha)}) = A\varphi_k^{(\alpha)}(x) + g(\varphi_k^{(\alpha)}(x))$, the solution to the ODE near its equilibrium can be written as

$$\varphi_k^{(\alpha)}(x) = (I + \alpha A)^{k-1} x + \sum_{j=0}^{k-1} (I + \alpha A)^{k-1-j} g(\varphi_j^{(\alpha)}(x)).$$

For $z_k := \left\| \varphi_k^{(\alpha)}(x + y) - \varphi_k^{(\alpha)}(x) \right\|$ with $x, x + y \in \mathcal{N}'$ we have by the linear expansion

$$\begin{aligned} z_k &= \left\| (I + \alpha A)^{k-1} y - \sum_{j=0}^{k-1} (I + \alpha A)^{k-1-j} \left(g(\varphi_j^{(\alpha)}(x + y)) - g(\varphi_j(x)) \right) \right\| \\ &\leq (\lambda^{(\alpha)})^{k-1} \|y\| + \sum_{j=0}^{k-1} (\lambda^{(\alpha)})^{k-1-j} \|g(\varphi_j(x + y)) - g(\varphi_j(x))\| \\ &\leq (\lambda^{(\alpha)})^{k-1} \|y\| + \sum_{j=0}^{k-1} (\lambda^{(\alpha)})^{k-1-j} \delta z_j. \end{aligned}$$

Therefore, with the use of Grönwalls inequality we obtain $z_k \leq (\lambda^{(\alpha)})^k \|y\| \exp(\delta \sum_{j=0}^{k-1} (\lambda^{(\alpha)})^{k-1-j})$ which shows the Lipschitz continuity of $\varphi_k(x)$ in $\mathcal{N}'$ near the equilibrium point with Lipschitz constant $L_k := (\lambda^{(\alpha)})^k \|y\| \exp(\delta \sum_{j=0}^{k-1} (\lambda^{(\alpha)})^{k-1-j})$. As $\lambda < 1$ and by selection of δ, the Lipschitz constant decays exponentially with increasing k. The previous reasoning can be extended to the derivatives of $\varphi_k^{(\alpha)}$ in order to obtain exponentially decaying bounds for $D_x^i \varphi_k, i = 1, 2, 3$ as in Equation (7.31). □

Lemma 49. *The exponentially decaying bound on the derivatives*

$$\left\| D_x^i \varphi_k^\alpha \right\| \leq C_{1,i} \exp(-k\alpha C_{2,i}) =: C_{D_x^i \varphi_k} \qquad \textit{for} \qquad i = 1, 2, 3$$

with $C_{2,i} > 1$ implies that for the infinite sums

$$\begin{aligned} \sum_{k=k_0}^{\infty} C_{D_x^i \varphi_k} &= \sum_{k=k_0}^{\infty} C_{1,i} \exp(-\alpha k C_{2,i}) \\ &= \exp(-\alpha k_0 C_{2,i}) \sum_{k=0}^{\infty} C_{1,i} \exp(-\alpha k C_{2,i}) \\ &= \frac{\exp(-\alpha k_0 C_{2,i})}{1 - \exp(-\alpha C_{2,i})} \leq C' \frac{1}{\alpha - \frac{\alpha^2}{2}} \end{aligned}$$

for some $0 < C' < \infty$. Throughout the proofs, we will in addition use that

$$\alpha \frac{1}{\alpha - \frac{\alpha^2}{2}} \leq 2 \qquad \textit{for} \qquad 0 < \alpha < 1.$$

7.8 Technical Lemmas

In this subsection, we state the more technical lemma needed to obtain the bound on the bias and accuracy bound for the bias approximation. Lemma 50 gives bounds on the summed differences between a noise perturbed function and its averaged counterpart. Lemma 51 shows that sums of the type $\sum_{k=0}^{n-1} D_x(h \circ \varphi_k)(X_{n-k-1}) \Big(F_1(X_{n-k}, Y_{n-k}) - F_1(X_{n-k-1}, Y_{n-k-1}) \Big)$ are bounded. Last, Lemma 52 shows that when averaging, i.e., looking at sums of the type $\frac{1}{T} \sum_{n=N+1}^{N+T} \sum_{k=0}^{n-1} D_x \varphi_k(X_{n-k-1}) \big(F(X_{n-k}, Y_{n-k}) - F(X_{n-k-1}, Y_{n-k-1}) \big)$, we can obtain more precise bounds depending on T and α.

Lemma 50. *Under assumptions* $\mathbf{A}_1$ *-* $\mathbf{A}_4$*, there exists a constants* $C_1, C_2, C_3 > 0$ *such that for continuously differentiable* $F_1 : \mathcal{X} \times \mathcal{Y} \to R^d$*,* $F_2 : \mathcal{X} \times \mathcal{Y} \to R^{d \times d}$

$$\sum_{k=0}^{n-1} D_x(h \circ \varphi_k)(X_k)\Big(F_1(X_{n-k-1}, Y_{n-k-1}) - \bar{F}_1(X_{n-k-1})\Big) \leq \tilde{C}_1, \tag{7.32}$$

$$\sum_{k=0}^{n-1} D_x^2(h \circ \varphi_k)(X_k) \cdot \Big(F_2(X_{n-k-1}, Y_{n-k-1}) - \bar{F}_2(X_{n-k-1})\Big) \leq \tilde{C}_2. \tag{7.33}$$

Proof. We show how to obtain the bound for the first equation. The second bound is then obtained by following the same steps. For Equation (7.32), by adding and subtracting terms into the difference to get

$$\begin{aligned}\sum_{k=0}^{n-1} D_x(h \circ \varphi_k)(X_k)\Big(&F_1(X_{n-k-1}, Y_{n-k-1}) - \bar{F}_1(X_{n-k-1}) && (7.34)\\ &+ G_{F_1}(X_{n-k-1}, Y_{n-k-1}) - G_{F_1}(X_{n-k-1}, Y_{n-k}) && (7.35)\\ &+ G_{F_1}(X_{n-k-1}, Y_{n-k}) - G_{F_1}(X_{n-k}, Y_{n-k}) && (7.36)\\ &+ G_{F_1}(X_{n-k}, Y_{n-k}) - G_{F_1}(X_{n-k-1}, Y_{n-k-1})\Big). && (7.37)\end{aligned}$$

By definition of $G_h(x, y)$,

$$\begin{aligned}\mathbb{E}[G_{F_1}(X_k, Y_k) - G_{F_1}(X_k, Y_{k+1}) \mid X_k, Y_k] &= -\sum_{y'}(I + K(X_k))_{Y_k, y'} G_{F_1}(X_k, y') \\ &= -(F_1(X_k, Y_k) - \bar{F}_1(X_k))\end{aligned}$$

from which follows that (7.35) is equal to zero. This is similar to the property given in Section 6.4.2 of our paper [4]. To bound the difference terms (7.37) we use Lemma 51. We get

$$\left|\sum_{k=0}^{n-1} D_x(h \circ \varphi_k)(X_k)(G_{F_1}(X_{n-k}, Y_{n-k}) - G_{F_1}(X_{n-k-1}, Y_{n-k-1}))\right| \leq C_1',$$

To bound the last sum, note that f is differentiable and bounded. By definition $|X_{k+1} - X_k| = \alpha|f(X_k, Y_k)| \leq \alpha C_f$, which by application of Taylor gives

$$\|G_{F_1}(X_k, Y_{k+1}) - G_{F_1}(X_{k+1}, Y_{k+1})\| \leq \alpha C_{D_x G_{F_1}} C_f.$$

For Equation (7.36), this yields

$$\begin{aligned}&\left\|\sum_{k=0}^{n-1} D_x(h\circ\varphi_k)(X_k)\cdot(G_{F_1}(X_k,Y_{k+1})-G_{F_1}(X_{k+1},Y_{k+1}))\right\|\\&\quad\leq \alpha\underbrace{\sum_{k=0}^{\infty} C_{D_x(h\circ\varphi_k)}}_{O(1/(\alpha-\frac{\alpha^2}{2}))} C_{D_xG_{F_1}} C_f =: C_1''.\end{aligned}$$

By merging the above observations we obtain

$$\left|\sum_{k=0}^{n-1} D_x(h\circ\varphi_k)(X_k)\Big(F_1(X_{n-k-1},Y_{n-k-1})-\bar{F}_1(X_{n-k-1})\Big)\right|\leq C_1'+C_1''=:\tilde{C}_1 \tag{7.38}$$

which proves the statement of Equation (7.32). As mentioned in the beginning, the application of the same steps to Equation (7.33) gives the second bound. □

Lemma 51. *Under assumptions* $\mathbf{A}_1$ *-* $\mathbf{A}_4$*, there exist constants* $C_1, C_2, C_3 > 0$ *such that for continuously differentiable* $F_1 : \mathcal{X}\times\mathcal{Y}\to R^d$*,* $F_2 : \mathcal{X}\times\mathcal{Y}\to R^{d\times d}$*,* $F_3 : \mathcal{X}\times\mathcal{Y}\to R^{d\times d\times d}$

$$||\sum_{k=0}^{n-1} D_x(h\circ\varphi_k)(X_{n-k-1})\Big(F_1(X_{n-k},Y_{n-k})-F_1(X_{n-k-1},Y_{n-k-1})\Big)||\leq C_1, \tag{7.39}$$

$$||\sum_{k=0}^{n-1} D_x^2(h\circ\varphi_k)(X_{n-k-1})\cdot\Big(F_2(X_{n-k},Y_{n-k})-F_2(X_{n-k-1},Y_{n-k-1})\Big)||\leq C_2, \tag{7.40}$$

$$\begin{aligned}||\sum_{k=0}^{n-1}\sum_{i,j,k}\frac{\partial^3}{\partial_i\partial_j\partial_k}(h\circ\varphi_k)(X_{n-k-1})\cdot\Big((F_3)_{i,j,k}(X_{n-k},Y_{n-k})\\-(F_3)_{i,j,k}(X_{n-k-1},Y_{n-k-1})\Big)||\leq C_3.\end{aligned} \tag{7.41}$$

Proof. We show how to obtain the constant C_1. By shifting the indices of the second sum of Equation (7.39), we rewrite

$$\begin{aligned}
&\sum_{k=0}^{n-1} D_x\varphi_k(X_{n-k-1})F_1(X_{n-k}, Y_{n-k}) \\
&\quad - \sum_{k=0}^{n-1} D_x\varphi_k(X_{n-k-1})F_1(X_{n-k-1}, Y_{n-k-1}) \\
&= D_x\varphi_0(X_{n-1})F(X_n, Y_n) - D_x\varphi_{n-1}(X_0)F_1(X_0, Y_0) \qquad (7.42) \\
&\quad + \sum_{k=1}^{n-1}\Big(D_x\varphi_k(X_{n-k-1}) - D_x\varphi_{k-1}(X_{n-k})\Big)F_1(X_{n-k}, Y_{n-k}). \qquad (7.43)
\end{aligned}$$

By assumption, the terms of line (7.42) are bounded independent of α. To obtain the bound for the sum in Equation (7.43) we add an artificial zero and define

$$\begin{aligned}
&D_x\varphi_k(X_{n-k-1}) - D_x\varphi_{k-1}(X_{n-k}) \\
&= \underbrace{\Big(D_x\varphi_k(X_{n-k-1}) - D_x\varphi_{k-1}(X_{n-k-1})}_{E_1} \\
&\quad + \underbrace{D_x\varphi_{k-1}(X_{n-k-1}) - D_x\varphi_{k-1}(X_{n-k})}_{E_2}.
\end{aligned}$$

Recall $X_{n-k} = X_{n-k-1} + \alpha f(X_{n-k-1}, Y_{n-k-1})$ with f bounded. By expansion around X_{n-k-1} and application of the bounds for the derivative terms and drift of the stochastic system f as well as denoting by C_R the residual terms which arise from the remainder term of the expansion,

$$\|E_2\| \leq \alpha C_{D_x^2\varphi_{k-1}} C_{R,1} C_f.$$

Using that $D_x(\varphi_k)(x) = D_x(\varphi_{k-1} \circ \varphi_1)(x)$, for E_1 we obtain

$$\begin{aligned}
E_1 &= D_x(\varphi_{k-1} \circ \varphi_1)(X_{n-k-1}) - D_x\varphi_{k-1}(X_{n-k-1}) \\
&= \alpha D_x\overline{f}(X_{n-k-1}) \underbrace{D_x\varphi_{k-1}(X_{n-k-1} + \alpha\overline{f}(X_{n-k-1}))}_{E_{1,1}} - D_x\varphi_{k-1}(X_{n-k-1}).
\end{aligned}$$

Considering the expansion of $E_{1,1}$ around X_{n-k-1} and denoting by C_R the residual terms arising from the expansion which are independent of φ_k and α. We bound

$$\begin{aligned}
|E_1| &\leq \big\|(\alpha D_x\overline{f}(X_{n-k-1}) - I)D_x\varphi_{k-1}(X_{n-k-1})\big\| + \alpha^2 C_f C_{D_x f} C_{D_x^2\varphi_{k-1}} C_R \\
&\leq \alpha(C_{D_x f} + 1)C_{D_x^2\varphi_{k-1}} + \alpha^2 C_f C_{D_x f} C_{D_x^2\varphi_{k-1}} C_R.
\end{aligned}$$

Under consideration of Lemma 49 and by the above observations, the lines (7.42) and (7.43) are bounded by

$$(C_{D_x\varphi_0} + C_{D_x\varphi_{n-1}})C_F + \sum_{k=1}^{n-1} \alpha C_{D_x^2\varphi_{k-1}} C_R C_f C_F$$
$$+ \alpha(C_{D_x f} + 1) \sum_{k=1}^{n-1} C_{D_x^2\varphi_{k-1}} + \alpha^2 C_f C_{D_x f} \sum_{k=1}^{n-1} C_{D_x^2\varphi_k} C_R := C_1 < \infty.$$

Finally, to obtain similar constants for the higher derivatives of φ_k, we follow the same reasoning. □

Lemma 52. *Given the same assumptions as in Lemma 51, the following holds:*

$$||\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1} D_x\varphi_k(X_{n-k-1})\big(F(X_{n-k},Y_{n-k}) - F(X_{n-k-1},Y_{n-k-1})\big)|| \tag{7.44}$$
$$\leq \frac{1}{T(\alpha - \frac{\alpha^2}{2})}\tilde{C}_{1,1} + \alpha\tilde{C}_{1,2} \tag{7.45}$$
$$||\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1} D_x^2(h\circ\varphi_k)(X_{n-k-1})\cdot\Big(F_2(X_{n-k},Y_{n-k}) - F_2(X_{n-k-1},Y_{n-k-1})\Big)|| \tag{7.46}$$
$$\leq \frac{1}{T(\alpha - \frac{\alpha^2}{2})}\tilde{C}_{2,1} + \alpha\tilde{C}_{2,2}, \tag{7.47}$$
$$||\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1}\sum_{i,j,k}\frac{\partial^3}{\partial_i\partial_j\partial_k}(h\circ\varphi_k)(X_{n-k-1})\cdot\Big((F_3)_{i,j,k}(X_{n-k},Y_{n-k}) - (F_3)_{i,j,k}(X_{n-k-1},Y_{n-k-1})\Big)|| \tag{7.48}$$
$$\leq \frac{1}{T(\alpha - \frac{\alpha^2}{2})}\tilde{C}_{3,1} + \alpha\tilde{C}_{3,2}. \tag{7.49}$$

Proof.

$$\frac{1}{T}\sum_{n=N+1}^{N+T}\sum_{k=0}^{n-1} D_x\varphi_k(X_{n-k-1})\big(F(X_{n-k},Y_{n-k}) - F(X_{n-k-1},Y_{n-k-1})\big)$$
$$= \sum_{k=0}^{N-1}\frac{1}{T}\sum_{\tau=1}^{T} D_x\varphi_{k+\tau-1}(X_{N+T-(k+\tau)-1})\big(F(X_{N+T-(k+\tau)},Y_{N+T-(k+\tau)})$$
$$- F(X_{N+T-(k+\tau)-1},Y_{N+T-(k+\tau)-1})\big) \tag{7.50}$$

We will now look at the inner sum for fixed k. To reduce notation, define

$$x_\tau^{(k)} := D_x\varphi_{k+\tau-1}(X_{N+T-(k+\tau)-1}),$$
$$y_\tau^{(k)} := F(X_{N+T-k-\tau}, Y_{N+T-k-\tau}).$$

Using the simplified notation and by application of the summation by parts identity

$$\begin{aligned}(7.50) &= -\sum_{k=0}^{N-1}\frac{1}{T}\sum_{\tau=1}^{T} x_\tau^{(k)}(y_{\tau+1}^{(k)} - y_\tau^{(k)}) \\ &= -\sum_{k=0}^{N-1}\frac{1}{T}\left(x_{T+1}^{(k)}y_{T+1}^{(k)} - y_1^{(k)}y_1^{(k)} - \sum_{\tau=1}^{T} y_{\tau+1}^{(k)}(x_{\tau+1}^{(k)} - x_\tau^{(k)})\right).\end{aligned}$$

Bounding $\sum_{k=0}^{N-1}\frac{1}{T}x_{T+1}^{(k)}y_{T+1}^{(k)}$, we obtain

$$\frac{1}{T}\sum_{k=0}^{N-1} x_{T+1}^{(k)}y_{T+1}^{(k)} - y_1^{(k)}y_1^{(k)} \le \frac{1}{T}2C_F\sum_{k=0}^{N-1} C_{D\varphi_k}.$$

By convergence properties of the derivatives of φ_k, there exists a constant $\tilde{C}_1$ such that $\frac{1}{T}2C_F\sum_{k=0}^{N-1} C_{D\varphi_k} \le \frac{1}{T(\alpha-\frac{\alpha^2}{2})}\tilde{C}_1$. Second, we argue similarly to the proof of Lemma 51, that there exists a constant $\tilde{C}_2$ such that

$$\sum_{k=0}^{N-1}\frac{1}{T}\sum_{\tau=1}^{T}(x_{\tau+1}^{(k)} - x_\tau^{(k)})y_{\tau+1}^{(k)} \le \alpha\tilde{C}_2.$$

With this we obtain the desired result. By the same line or argument, we obtain similar bound for the time averages of the higher order derivatives. □

8 Conclusion

In this thesis, we presented our contributions towards the grand question of the applicability and accuracy of the mean field approximation to systems with non-homogeneous properties. Starting with an overview of the general methodology in the second chapter, we show how generator comparison techniques can be used to bound the distance between a stochastic population system and its mean field approximation. We further elaborate on how to obtain the refinement terms in the case of density dependent population systems. The methodology serves as a basis for the following chapters in which we repeatedly fall back on these steps and adapt them to more intricate settings.

In the second part, we focus on the study of heterogeneous systems. We show that in the case of individual-level heterogeneity, we can construct a deterministic system approximating the probability of the individuals to be in their states. We show that this mean field approximation has an accuracy of order $O(1/N)$. We then continue by constructing a refinement term for the individuals which further reduces the error of the approximation to an order of $O(1/N^2)$. In the subsequent chapter, we introduce the RMF Tool, a numerical toolbox which facilitates the implementation of population models and allow the automated construction of the (refined) mean field approximation. The tool incorporates the aforementioned heterogeneous setting. In Chapter 5, we shift to population models with graph based connections. We show that the accuracy results of the 'graphon' mean field approximation strongly depend on the cut norm distance between the graph and the graphon. For graphs obtained through deterministic and stochastic sampling, we further state precise accuracy bounds for finite system sizes.

The last part of the thesis is dedicated to coupled systems. In Chapter 6, we look at the two-timescale systems consisting of slowly changing populations of interacting particles and a coupled rapidly changing environment. Looking at the behavior of finite-sized systems, we show that the 'average' mean field approximation has a bias of order $O(1/N)$. Moreover, we show that by carefully studying the bias the refinement ideas can be adapted for coupled systems yielding new refinement terms which account for the use of the averaging principle. We show how a numerically feasible extension term can be derived for the steady-state and that its utilization reduces the bias of the approximation to be of order $O(1/N^2)$. In the last chapter, we apply the mentioned results and refinement ideas to the stochastic approximation setting with constant stepsize α and state-dependent Markovian noise data. By adapting the proof procedure, we obtain comparable results as in the population setting can be obtained. We show that for a fixed α the bias between the stochastic approximation algorithm and the corresponding ODE is of order $O(\alpha)$ and can be further refined by an extension term.

Summarizing, the presented work shows that the idea behind the mean field approximation is widely applicable and that the methodology provided in Chapter 2 can be universally used to provide accuracy results. Nonetheless, the results obtained and assumptions made raise further questions as

to what extent different and more general models can be considered. In the next section, we will sum up some open questions and extensions for other population models.

8.1 Open Questions and Future Work

Mean Field Approximation and Refinements for Non-Differentiable Drifts To derive the refinement terms in this thesis as well as the ones of [54, 56], it is necessary to impose relatively strict continuity assumptions on the drift of the system, e.g., twice continuously differentiable for the first order refinement. While we think these assumptions are reasonable and fulfilled in many cases, it is natural to ask to what extent such assumptions on the differentiability can be loosened and replaced by other methods. Approaching the question from the stochastic approximation point of view, it has been shown in [15, 16, 78] that the asymptotic behavior can be described using differential inclusions. Specific applications for the mean field approximation however remain sparse, with [58] being one of the more recent advances in the field. To the best of our knowledge, in this setting a detailed bias analysis has not been conducted and remains an open question.

Non Markovian Dynamics In many real-world applications, state changes of a system are not exponentially distributed. In the case of load balancing, statistical analysis suggests that service times are not solely exponentially distributed but potentially follow Log-Normal, Gamma, or Phase-type distributions. In [2, 3, 28] the authors propose PDE methods which generalize the mean field approach in the load balancing context. The adaptation of the refinement idea into these settings is an open research direction.

Extension for Not-So-Dense / Sparse Graph Based Systems In Chapter 5, we carried out some analysis on the accuracy of 'graphon' mean field models with respect to population systems on densely connected graph structures, dense meaning that the number of neighbors of individuals is of order N. As motivated in [96], the extension towards graphs slower increasing or even bounded edge degrees for the individuals is non-trivial. Local interactions might have substantial influence on the global behavior of the system due to non vanishing correlation of the individuals. Some recent publications developing new approaches include [13, 53, 100].

Expansion of the RMF Tool In Chapter 4, we have introduced the RMF Tool which supports the implementation of the mean field and, most notably, of the refined mean field approximation. Its current functionality allows the application to homogeneous, density dependent and heterogeneous population processes, with the latter referring to the individual level heterogeneity as discussed in Chapter 3. Extending the toolbox's functionality to include our results on the 'average' mean field and 'graphon' mean field approximations would represents a natural next step to extend the scope of the toolbox.

Bibliography

1. E. Abbe. "Community Detection and Stochastic Block Models". *Foundations and Trends® in Communications and Information Theory* 14:1-2, 2018, pp. 1–162. ISSN: 1567-2190, 1567-2328. DOI: 10.1561/0100000067.
2. R. Aghajani, X. Li, and K. Ramanan. *Mean-Field Dynamics of Load-Balancing Networks with General Service Distributions*. 22, 2015. arXiv: 1512.05056 [math]. URL: http://arxiv.org/abs/1512.05056. preprint.
3. R. Aghajani and K. Ramanan. "The Hydrodynamic Limit of a Randomized Load Balancing Network". *The Annals of Applied Probability* 29:4, 2019, pp. 2114–2174. ISSN: 1050-5164, 2168-8737. DOI: 10.1214/18-AAP1444.
4. S. Allmeier and N. Gast. "Bias and Refinement of Multiscale Mean Field Models". *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 7:1, 2023, 23:1–23:29. DOI: 10.1145/3579336.
5. S. Allmeier and N. Gast. "Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works!" *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 6:1, 25, 2022, 13:1–13:43. DOI: 10.1145/3508033.
6. S. Allmeier and N. Gast. "Rmf Tool - A Library to Compute (Refined) Mean Field Approximation(s)". *ACM SIGMETRICS Performance Evaluation Review* 4, 2, 2022, pp. 35–40. ISSN: 0163-5999. DOI: 10.1145/3543146.3543156.
7. A. Aurell, R. Carmona, G. Dayanıklı, and M. Laurière. "Finite State Graphon Games with Applications to Epidemics". *Dynamic Games and Applications* 12:1, 1, 2022, pp. 49–81. ISSN: 2153-0793. DOI: 10.1007/s13235-021-00410-2.
8. M. Avella-Medina, F. Parise, M. T. Schaub, and S. Segarra. *Centrality Measures for Graphons: Accounting for Uncertainty in Networks*. 28, 2018. DOI: 10.1109/TNSE.2018.2884235. arXiv: 1707.09350 [physics, stat]. preprint.
9. F. Baccelli, M. Davydov, and T. Taillefumier. "Replica-Mean-Field Limits of Fragmentation-Interaction-Aggregation Processes". *Journal of Applied Probability* 59:1, 2022, pp. 38–59. ISSN: 0021-9002, 1475-6072. DOI: 10.1017/jpr.2021.31.
10. F. Baccelli and T. Taillefumier. "Replica-Mean-Field Limits for Intensity-Based Neural Networks". *SIAM Journal on Applied Dynamical Systems* 18:4, 2019, pp. 1756–1797.
11. K. Ball, T. G. Kurtz, L. Popovic, and G. Rempala. "Asymptotic Analysis of Multiscale Approximations to Reaction Networks". *The Annals of Applied Probability* 16:4, 1, 2006. ISSN: 1050-5164. DOI: 10.1214/105051606000000420.

12. A. D. Barbour. "Stein's Method and Poisson Process Convergence". *Journal of Applied Probability* 25:A, 1988, pp. 175–184. ISSN: 0021-9002, 1475-6072. DOI: 10.2307/3214155.

13. E. Bayraktar, S. Chakraborty, and R. Wu. *Graphon Mean Field Systems*. 5, 2022. arXiv: 2003.13180 [math]. URL: http://arxiv.org/abs/2003.13180. preprint.

14. E. Bayraktar and R. Wu. "Mean Field Interaction on Random Graphs with Dynamically Changing Multi-Color Edges". *Stochastic Processes and their Applications* 141, 2021, pp. 197–244. ISSN: 0304-4149. DOI: 10.1016/j.spa.2021.07.005.

15. M. Benaim, J. Hofbauer, and S. Sorin. "Stochastic Approximations and Differential Inclusions". *SIAM Journal on Control and Optimization* 44:1, 2005, pp. 328–348. ISSN: 0363-0129, 1095-7138. DOI: 10.1137/S0363012904439301.

16. M. Benaim, J. Hofbauer, and S. Sorin. "Stochastic Approximations and Differential Inclusions, Part II: Applications". *Mathematics of Operations Research* 31:4, 2006, pp. 673–695. ISSN: 0364-765X, 1526-5471. DOI: 10.1287/moor.1060.0213.

17. M. Benaim and J.-Y. Le Boudec. "A Class of Mean Field Interaction Models for Computer and Communication Systems". *Performance Evaluation* 65:11, 2008, pp. 823–838.

18. A. Benveniste, M. Metivier, and P. Priouret. *Adaptive Algorithms and Stochastic Approximations*. Springer Science & Business Media, 6, 2012. 373 pp. ISBN: 978-3-642-75894-2. Google Books: KF7sCAAAQBAJ.

19. G. Bet, F. Coppini, and F. R. Nardi. *Weakly Interacting Oscillators on Dense Random Graphs*. 24, 2022. arXiv: 2006.07670 [math]. URL: http://arxiv.org/abs/2006.07670. preprint.

20. S. Bhamidi, A. Budhiraja, and R. Wu. "Weakly Interacting Particle Systems on Inhomogeneous Random Graphs". *Stochastic Processes and their Applications* 129:6, 2019, pp. 2174–2206. ISSN: 0304-4149. DOI: 10.1016/j.spa.2018.06.014.

21. J. Bhandari, D. Russo, and R. Singal. "A Finite Time Analysis of Temporal Difference Learning With Linear Function Approximation". In: *Proceedings of the 31st Conference On Learning Theory*. Conference On Learning Theory. PMLR, 3, 2018, pp. 1691–1692. URL: https://proceedings.mlr.press/v75/bhandari18a.html.

22. P. Blanchard and E. Brüning. *Mathematical Methods in Physics: Distributions, Hilbert Space Operators, Variational Methods, and Applications in Quantum Physics*. Vol. 69. Progress in Mathematical Physics. Springer International Publishing, Cham, 2015. ISBN: 978-3-319-14044-5. DOI: 10.1007/978-3-319-14045-2.

23. J. R. Blum. "Approximation Methods Which Converge with Probability One". *The Annals of Mathematical Statistics* 25:2, 1954, pp. 382–386. ISSN: 0003-4851. JSTOR: 2236739. URL: https://www.jstor.org/stable/2236739.

24. R. Boorstyn, A. Kershenbaum, B. Maglaris, and V. Sahin. "Throughput Analysis in Multihop CSMA Packet Radio Networks". *IEEE Transactions on Communications* 35:3, 1987, pp. 267–274. ISSN: 0090-6778. DOI: 10.1109/TCOM.1987.1096769.

25. C. Bordenave, D. McDonald, and A. Proutiere. "A Particle System in Interaction with a Rapidly Varying Environment: Mean Field Limits and Applications". 16, 2009. arXiv: math/0701363. URL: http://arxiv.org/abs/math/0701363.

26. V. S. Borkar and S. P. Meyn. "The O.D.E. Method for Convergence of Stochastic Approximation and Reinforcement Learning". *SIAM Journal on Control and Optimization* 38:2, 2000, pp. 447–469. ISSN: 0363-0129. DOI: 10.1137/S0363012997331639.

27. V. S. Borkar. *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge University Press ; Hindustan Book Agency, Cambridge, UK : New York : New Delhi, 2008. 164 pp. ISBN: 978-0-521-51592-4.

28. M. Bramson, Y. Lu, and B. Prabhakar. "Randomized Load Balancing with General Service Time Distributions". *ACM SIGMETRICS Performance Evaluation Review* 38:1, 14, 2010, pp. 275–286. ISSN: 0163-5999. DOI: 10.1145/1811099.1811071.

29. A. Braverman. "Stein's Method for Steady-State Diffusion Approximations". 26, 2017. arXiv: 1704.08398 [math]. URL: http://arxiv.org/abs/1704.08398.

30. A. Braverman. "The Prelimit Generator Comparison Approach of Stein's Method". 7, 2021. arXiv: 2102.12027 [math]. URL: http://arxiv.org/abs/2102.12027.

31. A. Braverman and J. G. Dai. "Stein's Method for Steady-State Diffusion Approximations of M/Ph/n+M Systems". *The Annals of Applied Probability* 27:1, 1, 2017. ISSN: 1050-5164. DOI: 10.1214/16-AAP1211.

32. A. Braverman, J. G. Dai, and X. Fang. "High Order Steady-State Diffusion Approximations". 7, 2020. arXiv: 2012.02824 [math]. URL: http://arxiv.org/abs/2012.02824.

33. A. Braverman, J. G. Dai, and J. Feng. "Stein's Method for Steady-State Diffusion Approximations: An Introduction through the Erlang-A and Erlang-C Models". *Stochastic Systems* 6:2, 2017, pp. 301–366.

34. A. Budhiraja, D. Mukherjee, and R. Wu. "Supermarket Model on Graphs". *The Annals of Applied Probability* 29:3, 1, 2019. ISSN: 1050-5164. DOI: 10.1214/18-AAP1437.

35. P. E. Caines and M. Huang. "Graphon Mean Field Games and Their Equations". *SIAM Journal on Control and Optimization* 59:6, 2021, pp. 4373–4399. ISSN: 0363-0129, 1095-7138. DOI: 10.1137/20M136373X.

36. G. Casale and N. Gast. "Performance Analysis Methods for List-Based Caches With Non-Uniform Access". *IEEE/ACM Transactions on Networking* 29:2, 2021, pp. 651–664. ISSN: 1558-2566. DOI: 10.1109/TNET.2020.3042869.

37. E. Castiel, S. Borst, L. Miclo, F. Simatos, and P. Whiting. "Induced Idleness Leads to Deterministic Heavy Traffic Limits for Queue-Based Random-Access Algorithms". *The Annals of Applied Probability* 31:2, 1, 2021. ISSN: 1050-5164. DOI: 10.1214/20-AAP1609. arXiv: 1904.03980 [math].

38. F. Cecchi. *Mean-Field Limits for Ultra-Dense Random-Access Networks*. Technische Universiteit Eindhoven, 1, 2018. ISBN: 978-90-386-4415-8.

39. F. Cecchi, S. C. Borst, J. S. H. van Leeuwaarden, and P. A. Whiting. "Mean-Field Limits for Large-Scale Random-Access Networks". 24, 2019. arXiv: 1611.09723 [math]. URL: http://arxiv.org/abs/1611.09723.

40. H. Che, Z. Wang, and Y. Tung. "Analysis and Design of Hierarchical Web Caching Systems". In: IEEE INFOCOM. Vol. 3. IEEE, Anchorage, AK, USA, 2001, pp. 1416–1424. ISBN: 978-0-7803-7016-6. DOI: 10.1109/INFCOM.2001.916637.

41. Z. Chen, S. T. Maguluri, S. Shakkottai, and K. Shanmugam. *A Lyapunov Theory for Finite-Sample Guarantees of Asynchronous Q-Learning and TD-Learning Variants*. 4, 2023. DOI: 10.48550/arXiv.2102.01567. arXiv: 2102.01567 [cs, math, stat]. preprint.

42. A. Dan and D. Towsley. "An Approximate Analysis of the LRU and FIFO Buffer Replacement Schemes". In: *Proceedings of the 1990 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*. SIGMETRICS '90. Association for Computing Machinery, New York, NY, USA, 1, 1990, pp. 143–152. ISBN: 978-0-89791-359-1. DOI: 10.1145/98457.98525.

43. G. F. de Arruda, F. A. Rodrigues, and Y. Moreno. "Fundamentals of Spreading Processes in Single and Multilayer Complex Networks". *Physics Reports* 756, 2018, pp. 1–59. ISSN: 03701573. DOI: 10.1016/j.physrep.2018.06.007. arXiv: 1804.08777.

44. J.-F. Delmas, P. Frasca, F. Garin, V. C. Tran, A. Velleret, and P.-A. Zitt. *Individual Based SIS Models on (Not so) Dense Large Random Networks*. 2023. arXiv: 2302.13385 [math, q-bio].

45. M. V. Der Boor, S. C. Borst, J. S. H. Van Leeuwaarden, and D. Mukherjee. "Scalable Load Balancing in Networked Systems: A Survey of Recent Advances". *SIAM Review* 64:3, 2022, pp. 554–622. ISSN: 0036-1445, 1095-7200. DOI: 10.1137/20M1323746.

46. B. K. Driver. "Analysis Tools with Applications". *Lecture notes*, 2003.

47. K. R. Duffy. "Mean Field Markov Models of Wireless Local Area Networks". *Markov Processes and Related Fields* 16:2, 2010, pp. 295–328. ISSN: 1024-2953. URL: https://eprints.maynoothuniversity.ie/6221/.

48. R. Fagin. "Asymptotic Miss Ratios over Independent References". *Journal of Computer and System Sciences* 14:2, 1977, pp. 222–250.

49. M. Feuillet and P. Robert. "A Scaling Analysis of a Transient Stochastic Network". *Advances in Applied Probability* 46:2, 2014, pp. 516–535. ISSN: 0001-8678, 1475-6064. DOI: 10.1239/aap/1401369705.

50. C. Fricker and N. Gast. "Incentives and Redistribution in Homogeneous Bike-Sharing Systems with Stations of Finite Capacity". *EURO Journal on Transportation and Logistics* 5:3, 2016, pp. 261–291. ISSN: 2192-4384. DOI: 10.1007/s13676-014-0053-5.

51. C. Fricker, N. Gast, and H. Mohamed. "Mean Field Analysis for Inhomogeneous Bike Sharing Systems". In: *AofA*. Vol. DMTCS Proceedings vol. AQ, 23rd Intern. Meeting on Probabilistic, Combinatorial, and Asymptotic Methods for the Analysis of Algorithms (AofA'12). DMTCS, Montreal, Canada, 2012. DOI: 10.46298/dmtcs.3006.

52. A. Ganguly. "Non-Markovian Interacting Particle Systems on Large Sparse Graphs: Hydrodynamic Limits and Marginal Characterizations". Brown Univerity, 2022. 208 pp. URL: https://repository.library.brown.edu/studio/item/bdr:pfzykyap/.

53. A. Ganguly and K. Ramanan. *Hydrodynamic Limits of Non-Markovian Interacting Particle Systems on Sparse Graphs*. 5, 2022. arXiv: 2205.01587 [math]. URL: http://arxiv.org/abs/2205.01587. preprint.

54. N. Gast, L. Bortolussi, and M. Tribastone. "Size Expansions of Mean Field Approximation: Transient and Steady-State Analysis". *Performance Evaluation* 129, 2019. DOI: 10.1016/j.peva.2018.09.005.

55. N. Gast. "Expected Values Estimated via Mean-Field Approximation Are 1/N-Accurate". *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1:1, 13, 2017, pp. 1–26. ISSN: 24761249. DOI: 10.1145/3084454.

56. N. Gast. "Refined Mean Field Tool", 2018. URL: https://github.com/ngast/rmf_tool.

57. N. Gast. "Refinements of Mean Field Approximation". 30, 2020. URL: http://polaris.imag.fr/nicolas.gast/pdfs/hdr_gast.pdf.

58. N. Gast and B. Gaujal. "Markov Chains with Discontinuous Drifts Have Differential Inclusion Limits". *Performance Evaluation}* 69:12, 2012, pp. 623–642. ISSN: 0759-1063, 2070-2779. DOI: 10.1177/075910639203700105.

59. N. Gast and B. Van Houdt. "A Refined Mean Field Approximation". *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1:2, 2017, 33:1–33:28. DOI: 10.1145/3154491.

60. N. Gast and B. Van Houdt. "Transient and Steady-state Regime of a Family of List-based Cache Replacement Algorithms". In: *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems - SIGMETRICS '15*. The 2015 ACM SIGMETRICS International Conference. ACM Press, Portland, Oregon, USA, 2015, pp. 123–136. ISBN: 978-1-4503-3486-0. DOI: 10.1145/2745844.2745850.

61. M. G. M. Gomes, R. Aguas, R. M. Corder, J. G. King, K. E. Langwig, C. Souto-Maior, J. Carneiro, M. U. Ferreira, and C. Penha-Gonçalves. *Individual Variation in Susceptibility or Exposure to SARS-CoV-2 Lowers the Herd Immunity Threshold*. preprint. Epidemiology, 2, 2020. DOI: 10.1101/2020.04.27.20081893.

62. R. Grima. "An Effective Rate Equation Approach to Reaction Kinetics in Small Volumes: Theory and Application to Biochemical Reactions in Nonequilibrium Steady-State Conditions". *The Journal of chemical physics* 133:3, 2010, 07B604.

63. R. Grima, P. Thomas, and A. V. Straube. "How Accurate Are the Nonlinear Chemical Fokker-Planck and Chemical Langevin Equations?" *The Journal of Chemical Physics* 135:8, 2011.

64. M. Hazewinkel, F. Calogero, Y. I. Manin, A. H. G. Rinnooy Kan, and G.-C. Rota. *Stochastic Analysis of Computer Storage*. Springer Netherlands, Dordrecht, 1987. ISBN: 978-90-277-2515-8. URL: https://doi.org/10.1007/978-0-585-27373-0.

65. R. Hirade and T. Osogami. "Analysis of Page Replacement Policies in the Fluid Limit". *Operations Research* 58, 2010, pp. 971–984. ISSN: 0030-364X. DOI: 10.1287/opre.1090.0761.

66. L. Hodgkinson, R. McVinish, and P. K. Pollett. "Normal Approximations for Discrete-Time Occupancy Processes". 10, 2018. arXiv: 1801.00542 [math]. URL: http://arxiv.org/abs/1801.00542.

67. P. Hunt and T. Kurtz. "Large Loss Networks". *Stochastic Processes and their Applications* 53:2, 1994, pp. 363–378. ISSN: 03044149. DOI: 10.1016/0304-4149(94)90071-X.

68. J. J. Hunter. "Generalized Inverses and Their Application to Applied Probability Problems". *Linear Algebra and its Applications* 45, 1982, pp. 157–198. ISSN: 00243795. DOI: 10.1016/0024-3795(82)90218-X.

69. D. Huo, Y. Chen, and Q. Xie. *Bias and Extrapolation in Markovian Linear Stochastic Approximation with Constant Stepsizes*. 19, 2022. arXiv: 2210.00953 [cs, math, stat]. URL: http://arxiv.org/abs/2210.00953. preprint.

70. I. C. F. Ipsen and C. D. Meyer. "Uniform Stability of Markov Chains". *SIAM Journal on Matrix Analysis and Applications* 15:4, 1994, pp. 1061–1074. ISSN: 0895-4798, 1095-7162. DOI: 10.1137/S0895479892237562.

71. B. Jiang, P. Nain, and D. Towsley. "On the Convergence of the TTL Approximation for an LRU Cache under Independent Stationary Request Processes". *ACM Transactions on Modeling and Performance Evaluation of Computing Systems (TOMPECS)* 3:4, 2018, pp. 1–31.

72. H.-W. Kang and T. G. Kurtz. "Separation of Time-Scales and Model Reduction for Stochastic Reaction Networks". *The Annals of Applied Probability* 23:2, 1, 2013. ISSN: 1050-5164. DOI: 10.1214/12-AAP841.

73. H.-W. Kang, T. G. Kurtz, and L. Popovic. "Central Limit Theorems and Diffusion Approximations for Multiscale Markov Chain Models". *The Annals of Applied Probability* 24:2, 1, 2014. ISSN: 1050-5164. DOI: 10.1214/13-AAP934.

74. V. N. Kolokoltsov, J. Li, and W. Yang. "Mean Field Games and Nonlinear Markov Processes". 6, 2012. arXiv: 1112.3744 [math]. URL: http://arxiv.org/abs/1112.3744.

75. T. G. Kurtz. "Limit Theorems for Sequences of Jump Markov Processes Approximating Ordinary Differential Processes". *Journal of Applied Probability* 8:2, 1971, pp. 344–356. ISSN: 0021-9002, 1475-6072. DOI: 10.2307/3211904.

76. T. G. Kurtz. "Solutions of Ordinary Differential Equations as Limits of Pure Jump Markov Processes". *Journal of Applied Probability* 7:1, 1970, pp. 49–58. ISSN: 0021-9002, 1475-6072. DOI: 10.2307/3212147.

77. T. G. Kurtz. "Strong Approximation Theorems for Density Dependent Markov Chains". *Stochastic Processes and their Applications* 6:3, 1978, pp. 223–240. ISSN: 03044149. DOI: 10.1016/0304-4149(78)90020-0.

78. H. J. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Vol. 35. Stochastic Modelling and Applied Probability. Springer-Verlag, New York, 2003. ISBN: 978-0-387-00894-3. DOI: 10.1007/b97441.

79. C. Lakshminarayanan and C. Szepesvari. "Linear Stochastic Approximation: How Far Does Constant Step-Size and Iterate Averaging Go?" In: *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*. International Conference on Artificial Intelligence and Statistics. PMLR, 31, 2018, pp. 1347–1355. URL: https://proceedings.mlr.press/v84/lakshminarayanan18a.html.

80. J.-M. Lasry and P.-L. Lions. "Mean Field Games". *Japanese Journal of Mathematics* 2:1, 2007, pp. 229–260. ISSN: 0289-2316, 1861-3624. DOI: 10.1007/s11537-007-0657-8.

81. J.-Y. Le Boudec, D. McDonald, and J. Mundinger. "A Generic Mean Field Convergence Result for Systems of Interacting Objects". In: *Fourth International Conference on the Quantitative Evaluation of Systems (QEST 2007)*. Fourth International Conference on the Quantitative Evaluation of Systems (QEST 2007). IEEE, Edinburgh, Scotland, UK, 2007, pp. 3–18. ISBN: 978-0-7695-2883-0. DOI: 10.1109/QEST.2007.8.

82. L. Lovász. *Large Networks and Graph Limits*. Vol. 60. Colloquium Publications. American Mathematical Society, Providence, Rhode Island, 12, 2012. ISBN: 978-0-8218-9085-1. DOI: 10.1090/coll/060.

83. H. P. McKean. "Propagation of Chaos for a Class of Non-Linear Parabolic Equations". *Stochastic Differential Equations (Lecture Series in Differential Equations, Session 7, Catholic Univ., 1967)*, 1967, pp. 41–57.

84. S. Mei, A. Montanari, and P.-M. Nguyen. "A Mean Field View of the Landscape of Two-Layers Neural Networks". 28, 2018. arXiv: 1804.06561 [cond-mat, stat]. URL: http://arxiv.org/abs/1804.06561.

85. A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, Am. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, Š. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz. "SymPy: Symbolic Computing in Python". *PeerJ Computer Science* 3, 2, 2017, e103. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.103.

86. M. Mézard, G. Parisi, and M. A. Virasoro. "Spin Glass Theory and beyond: An Introduction to the Replica Method and Its Applications". *World Scientific Publishing Company*, 1987, pp. 232–237.

87. M. Mitzenmacher. "The Power of Two Choices in Randomized Load Balancing". *IEEE Transactions on Parallel and Distributed Systems* 12:10, 2001, pp. 1094–1104. ISSN: 10459219. DOI: 10.1109/71.963420.

88. M. Mitzenmacher. "Analyzing Distributed Join-Idle-Queue: A Fluid Limit Approach". In: *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference On*. IEEE, 2016, pp. 312–318.

89. M. D. Mitzenmacher. "The Power of Two Random Choices in Randomized Load Balancing". PhD thesis. PhD thesis, Graduate Division of the University of California at Berkley, 1996.

90. A. Montalbán, R. M. Corder, and M. G. M. Gomes. "Herd Immunity under Individual Variation and Reinfection". 2020. arXiv: 2008.00098 [physics, q-bio].

91. A. Mukhopadhyay and R. R. Mazumdar. "Analysis of Load Balancing in Large Heterogeneous Processor Sharing Systems". 2015. arXiv: 1311.5806 [cs, math].

92. *Multiscale Methods*. Vol. 53. Texts Applied in Mathematics. Springer New York, New York, NY, 2008. ISBN: 978-0-387-73828-4. DOI: 10.1007/978-0-387-73829-1.

93. M. F. Norman. *Markov Processes and Learning Models*. Vol. 84. Academic Press New York, 1972. ISBN: 0-12-521450-2.

94. A. Pazy. *Semigroups of Linear Operators and Applications to Partial Differential Equations*. Red. by J. E. Marsden, L. Sirovich, and F. John. Vol. 44. Applied Mathematical Sciences. Springer New York, New York, NY, 1983. ISBN: 978-1-4612-5563-5. DOI: 10.1007/978-1-4612-5561-1.

95. L. Perko. *Differential Equations and Dynamical Systems*. 3rd ed. Texts in Applied Mathematics 7. Springer, New York, 2001. 553 pp. ISBN: 978-0-387-95116-4.

96. K. Ramanan. "Beyond Mean-Field Limits for the Analysis of Large-Scale Networks". *Queueing Systems* 100:3-4, 2022, pp. 345–347. ISSN: 0257-0130, 1572-9443. DOI: 10.1007/s11134-022-09845-9.

97. H. Robbins and S. Monro. "A Stochastic Approximation Method". *The Annals of Mathematical Statistics* 22:3, 1951, pp. 400–407. ISSN: 0003-4851. JSTOR: 2236626. URL: https://www.jstor.org/stable/2236626.

98. P. Robert and G. Vignoud. "Stochastic Models of Neural Synaptic Plasticity". 9, 2021. arXiv: 2010.08195 [math, q-bio]. URL: http://arxiv.org/abs/2010.08195.

99. A. Roy, C. Singh, and Y. Narahari. "Recent Advances in Modeling and Control of Epidemics Using a Mean Field Approach". *Sādhanā* 48:4, 27, 2023, p. 207. ISSN: 0973-7677. DOI: 10.1007/s12046-023-02268-z.

100. D. Rutten and D. Mukherjee. "Mean-field Analysis for Load Balancing on Spatial Graphs". In: *Abstract Proceedings of the 2023 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*. SIGMETRICS '23. Association for Computing Machinery, New York, NY, USA, 19, 2023, pp. 27–28. DOI: 10.1145/3578338.3593552. URL: https://doi.org/10.1145/3578338.3593552 (visited on 03/22/2024).

101. R. Srikant and L. Ying. "Finite-Time Error Bounds For Linear Stochastic Approximation and TD Learning".

102. C. Stein. "Approximate Computation of Expectations". *Lecture Notes-Monograph Series* 7, 1986, pp. i–164.

103. W. J. Stewart. *Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling*. Princeton University Press, Princeton, N.J, 2009. 758 pp. ISBN: 978-0-691-14062-9.

104. P. Thomas, H. Matuschek, and R. Grima. "Intrinsic Noise Analyzer: A Software Package for the Exploration of Stochastic Biochemical Kinetics Using the System Size Expansion". *PLoS ONE* 7:6, 12, 2012. Ed. by J. Peccoud, e38518. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0038518.

105. N. Tsukada, R. Hirade, and N. Miyoshi. "Fluid Limit Analysis of FIFO and RR Caching for Independent Reference Models". *Performance Evaluation* 69:9, 2012, pp. 403–412. ISSN: 01665316. DOI: 10.1016/j.peva.2012.05.008.

106. R. Van Der Hofstad. *Random Graphs and Complex Networks*. Cambridge University Press, Cambridge, 2017. ISBN: 978-1-316-77942-2. DOI: 10.1017/9781316779422.

107. P. M. van de Ven, S. C. Borst, J. S. H. van Leeuwaarden, and A. Proutière. "Insensitivity and Stability of Random-Access Networks". *Performance Evaluation*. Performance 2010 67:11, 1, 2010, pp. 1230–1242. ISSN: 0166-5316. DOI: 10.1016/j.peva.2010.08.011.

108. B. Van Houdt. "A Mean Field Model for a Class of Garbage Collection Algorithms in Flash-Based Solid State Drives". *ACM SIGMETRICS Performance Evaluation Review* 41:1, 14, 2013, pp. 191–202. ISSN: 0163-5999. DOI: 10.1145/2494232.2465543.

109. N. G. van Kampen. *Stochastic Processes in Physics and Chemistry*. Elsevier, Amsterdam; Boston; London, 2007. ISBN: 978-0-444-52965-7.

110. N. Vvedenskaya, R. Dobrushin, and F. Karpelevich. "A Queueing System with a Choice of the Shorter of Two Queues - an Asymptotic Approach". *Problemy Peredachi Informatsii*, 32(1):20–34 1996.

111. X. Wang and K. Kar. "Throughput Modelling and Fairness Issues in CSMA/CA Based Ad-Hoc Networks". In: *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.* Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Vol. 1. 2005, 23–34 vol. 1. DOI: 10.1109/INFCOM.2005.1497875.

112. Y. Yang, R. Luo, M. Li, M. Zhou, W. Zhang, and J. Wang. "Mean Field Multi-Agent Reinforcement Learning". In: *Proceedings of the 35th International Conference on Machine Learning*. International Conference on Machine Learning. PMLR, 3, 2018, pp. 5571–5580. URL: https://proceedings.mlr.press/v80/yang18d.html.

113. L. Ying. "On the Approximation Error of Mean-Field Models". In: *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science - SIGMETRICS '16*. The 2016 ACM SIGMETRICS International Conference. ACM Press, Antibes Juan-les-Pins, France, 2016, pp. 285–297. ISBN: 978-1-4503-4266-7. DOI: 10.1145/2896377.2901463.

114. L. Ying. "Stein's Method for Mean Field Approximations in Light and Heavy Traffic Regimes". *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1:1, 13, 2017, pp. 1–27. ISSN: 2476-1249, 2476-1249. DOI: 10.1145/3084449.

115. K. Zhang, Z. Yang, and T. Başar. "Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms". In: *Handbook of Reinforcement Learning and Control*. Ed. by K. G. Vamvoudakis, Y. Wan, F. L. Lewis, and D. Cansever. Studies in Systems, Decision and Control. Springer International Publishing, Cham, 2021, pp. 321–384. ISBN: 978-3-030-60990-0. DOI: 10.1007/978-3-030-60990-0_12.

116. Z. Zhao and D. Mukherjee. "Optimal Rate-Matrix Pruning For Heterogeneous Systems". *ACM SIGMETRICS Performance Evaluation Review* 51:4, 2024, pp. 26–27. ISSN: 0163-5999. DOI: 10.1145/3649477.3649492.

117. Z. Zhao, D. Mukherjee, and R. Wu. "Exploiting Data Locality to Improve Performance of Heterogeneous Server Clusters". *Stochastic Systems*, 2024. ISSN: 1946-5238. DOI: 10.1287/stsy.2022.0040.