



**HAL**  
open science

# Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works!

Sebastian Allmeier, Nicolas Gast

► **To cite this version:**

Sebastian Allmeier, Nicolas Gast. Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works!. Proceedings of the ACM on Measurement and Analysis of Computing Systems , 2022, 6 (1), pp.1-43. 10.1145/3508033 . hal-03600672

**HAL Id: hal-03600672**

**<https://inria.hal.science/hal-03600672>**

Submitted on 7 Mar 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Mean Field and Refined Mean Field Approximations for Heterogeneous Systems: It Works!

SEBASTIAN ALLMEIER and NICOLAS GAST, Inria, France and Univ. Grenoble Alpes, France

Mean field approximation is a powerful technique to study the performance of large stochastic systems represented as  $n$  interacting objects. Applications include load balancing models, epidemic spreading, cache replacement policies, or large-scale data centers. Mean field approximation is asymptotically exact for systems composed of  $n$  homogeneous objects under mild conditions. In this paper, we study what happens when objects are *heterogeneous*. This can represent servers with different speeds or contents with different popularities. 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  $O(1/n)$ . More importantly, we show how to adapt the refined mean field approximation, developed by the authors of [14], and show that the error of this approximation is reduced to  $O(1/n^2)$ . To illustrate the applicability of our result, we present two examples. The first addresses a list-based cache replacement model, RANDOM( $m$ ), which is an extension of the RANDOM policy. The second is a heterogeneous supermarket model. These examples show that the proposed approximations are computationally tractable and very accurate. They also show that for moderate system sizes ( $n \approx 30$ ) the refined mean field approximation tends to be more accurate than simulations for any reasonable simulation time.

CCS Concepts: • **Mathematics of computing** → *Stochastic processes; Ordinary differential equations.*

Additional Key Words and Phrases: mean field approximation; mean field models; Markov population processes; mean estimation; heterogeneity

## 1 INTRODUCTION

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 [10, 29], load balancing strategies [28, 30], caching [15] or SSDs [33]. Building on this approximation, a refined mean field approximation is introduced in [14, 16] 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 [26], 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 [18]. 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 paper, 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

---

Authors’ address: Sebastian Allmeier, sebastian.allmeier@inria.fr; Nicolas Gast, nicolas.gast@inria.fr, Inria, Grenoble, France, Univ. Grenoble Alpes, Grenoble, France.

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 [14]. These approximations are shown in [14] 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 our paper 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 [14], then

$$\begin{aligned}\mathbb{P}(S_k(t) = s) &= x_{(k,s)}(t) + O(1/n), \\ \mathbb{P}(S_k(t) = s) &= x_{(k,s)}(t) + v_{(k,s)}(t) + O(1/n^2).\end{aligned}$$

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 [13, 26, 36]. 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 [15] 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 [1], which allowed us to implement and solve the differential equations with relative ease.

*Roadmap.* The remainder of this paper is organized as follows. We describe more related work in Section 2. We introduce the heterogeneous interaction model in Section 3. We define and study the accuracy of the approximation in Section 4. This section contains the main results of the paper. We present the two numerical examples in Section 5. The proofs are given in Section 6. Finally, we conclude in Section 7. Some technical lemmas are postponed to the appendix.

## 2 RELATED WORK

*Generator and Stein's method.* Our paper builds on the recent line of work regarding the use of Stein's method [31]. 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 [4, 5]. There is still an active development in this area. For instance, this method has been used to develop higher-order diffusion approximations [3, 6]. It is used in [23] 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 [13, 25, 36, 37]. 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 [14, 16] to propose what the authors called a *refined* mean field approximation, that is similar to the system size expansion introduced in mathematical biology [19, 20, 34].

In fact, there exists a close link between the refined approximation that we propose in the paper and the approach of [14, 16]. Since the results of [14, 16] 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 [14, 16], there exists a constant  $v_{(k,s)}(t)$  such that

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

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, our paper 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 [13, 25, 35], 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 6.2). The extra difficulty in our case is to carefully analyze the remainder terms: while to analyze (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 D.

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 [30] or cache replacement policies [22, 32]. 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 [2, 27].

*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 [11] (sometimes misleadingly called the Che-approximation after it was rediscovered in [8]) or fixed-point approximation like [9]. Theoretical support exists to prove that these approximations are asymptotically correct [12, 15, 24].

The generic method that we propose in this paper 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 [15] in which a mean field approximation for the RAND( $\mathbf{m}$ ) policy is derived. This paper 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 [15] 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  $\|M\|$  is a supremum norm. The norm used in [15] is a supremum norm and we do not believe it can be derived from a scalar product. The approach that we take in this paper is radically different as we work with a comparison of generators. This allows us to correct the proof of [15] 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 C.2 that their result is a consequence of ours (and that our results give a finer bound).

### 3 THE HETEROGENEOUS POPULATION MODEL

#### 3.1 Interaction Model

We consider a population model composed of  $n$  interacting objects. Each object evolves in a finite<sup>1</sup> 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

<sup>1</sup>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.

of the system at time  $t$  is given by  $\mathbf{S}^{(n)}(t) = (S_1^{(n)}(t), \dots, S_n^{(n)}(t)) \in \mathcal{S}^n$ . We assume that the stochastic process  $\mathbf{S}^{(n)} = (\mathbf{S}^{(n)}(t))_{t \geq 0}$  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  $\mathbf{k} = (k_1, \dots, k_d)$ , that these objects jump *simultaneously*<sup>2</sup> from their states  $\mathbf{s} = (s_1, \dots, s_d)$  to  $\mathbf{s}' = (s'_1, \dots, s'_d) \neq (s_1, \dots, s_d)$  at rate  $\frac{1}{dn^{d-1}} r_{\mathbf{k}, \mathbf{s} \rightarrow \mathbf{s}'}^{(n)}$ . 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 paper, 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 paper 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  $\sigma$  of the set  $\{1 \dots d\}$ , the rate satisfy  $r_{k_1, \dots, k_d, (s_1, \dots, s_d) \rightarrow (s'_1, \dots, s'_d)}^{(n)} = r_{k_{\sigma(1)}, \dots, k_{\sigma(d)}, (s_{\sigma(1)}, \dots, s_{\sigma(d)}) \rightarrow (s'_{\sigma(1)}, \dots, s'_{\sigma(d)})}^{(n)}$ . This does not imply that objects are homogeneous but should be seen as 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.

### 3.2 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  $X_{(k,s)}^{(n)}(t)$  as

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

The state of the system is described by  $\mathbf{X}^{(n)}(t) = (X_{(k,s)}^{(n)}(t))_{k \in \{1 \dots n\}, s \in \mathcal{S}}$  with  $\mathcal{X}^{(n)} \subset \{0, 1\}^{n \times |\mathcal{S}|}$  being the set of attainable states for  $\mathbf{X}^{(n)} = (\mathbf{X}^{(n)}(t))_{t \geq 0}$ . In particular, for all  $\mathbf{x} \in \mathcal{X}^{(n)}$ , 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  $\mathbf{X}^{(n)}$  is less compact than the original representation  $\mathbf{S}^{(n)}$  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  $\mathbf{X}^{(n)}$ . Let  $\mathbf{e}_{(k,s)}^{(n)}$  denote a matrix of size  $n \times |\mathcal{S}|$  whose  $(k, s)$  component is equal to 1, all others being equal to 0. If object  $k$  transitions from state  $s$  to state  $s'$ ,  $\mathbf{X}^{(n)}$  changes into  $\mathbf{X}^{(n)} + \mathbf{e}_{(k,s')}^{(n)} - \mathbf{e}_{(k,s)}^{(n)}$ . Hence, expressed as a function of

<sup>2</sup>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.

$\mathbf{X}^{(n)}$ , the process  $\mathbf{X}^{(n)}$  jumps to (for  $k_1, k_2, \dots \in \{1 \dots n\}$  and  $s_1, s'_1, s_2, s'_2, \dots \in \mathcal{S}$ ):

$$\mathbf{X}^{(n)} + \mathbf{e}_{(k_1, s'_1)}^{(n)} - \mathbf{e}_{(k_1, s_1)}^{(n)} \quad \text{at rate } r_{k_1, (s_1) \rightarrow (s'_1)}^{(n)} X_{(k_1, s_1)}^{(n)} \quad (2a)$$

$$\mathbf{X}^{(n)} + \mathbf{e}_{(k_1, s'_1)}^{(n)} - \mathbf{e}_{(k_1, s_1)}^{(n)} + \mathbf{e}_{(k_2, s'_2)}^{(n)} - \mathbf{e}_{(k_2, s_2)}^{(n)} \quad \text{at rate } \frac{1}{2n} r_{k_1, k_2, (s_1, s_2) \rightarrow (s'_1, s'_2)}^{(n)} X_{(k_1, s_1)}^{(n)} X_{(k_2, s_2)}^{(n)} \quad (2b)$$

$$\mathbf{X}^{(n)} + \mathbf{e}_{(k_1, s'_1)}^{(n)} - \mathbf{e}_{(k_1, s_1)}^{(n)} + \dots + \mathbf{e}_{(k_d, s'_d)}^{(n)} - \mathbf{e}_{(k_d, s_d)}^{(n)} \quad \text{at rate } \frac{1}{dn^{d-1}} r_{k_1, \dots, k_d, \mathbf{s} \rightarrow \mathbf{s}'}^{(n)} X_{(k_1, s_1)}^{(n)} \dots X_{(k_d, s_d)}^{(n)} \quad (2c)$$

In the above equations (2a) corresponds to a unilateral transition of object  $k_1$  from  $s_1$  to  $s'_1$ , (2b) corresponds to transitions caused by a pairwise interaction between object  $k_1$  and  $k_2$  and (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 Main Notations

Throughout the paper, we use bold letters (like  $\mathbf{X}^{(n)}$ ,  $\mathbf{x}, \dots$ ) to denote matrices and regular letters (like  $X_{(k,s)}^{(n)}$ ,  $x_{(k,s)}$ ,  $n, \dots$ ) to denote scalars. Capital letters (like  $\mathbf{X}^{(n)}$ ,  $\mathbf{S}^{(n)}$ ) denote random variables whereas lower case letters ( $\mathbf{x}, v_{(k,s)}^{(n)}, \dots$ ) 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  $t$ ).

## 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.

### 4.1 Drift and Mean Field Approximation

For a given state  $\mathbf{x} \in \mathcal{X} \subset \{0, 1\}^{n \times |\mathcal{S}|}$ , we define the drift of the system in  $\mathbf{x}$  as the expected variation of the stochastic process  $\mathbf{X}^{(n)}$  at time  $t$ :  $f^{(n)}(\mathbf{X}^{(n)}(t)) = \lim_{dt \rightarrow 0} \frac{1}{dt} \mathbb{E} [\mathbf{X}^{(n)}(t + dt) - \mathbf{X}^{(n)}(t) \mid \mathbf{X}^{(n)}(t) = \mathbf{x}]$ . Based on the transitions (2), if  $d \leq 2$  (i.e., only unilateral and pairwise interactions)<sup>3</sup>, the  $(k, s)$  component of the drift can be expressed as:

$$\begin{aligned} & \sum_{s' \neq s} (r_{k, (s') \rightarrow (s)}^{(n)} x_{(k, s')} - r_{k, (s) \rightarrow (s')}^{(n)} x_{(k, s)}) \\ & + \frac{1}{n} \sum_{s', k_1, s_1, s'_1} (r_{k, k_1, (s', s'_1) \rightarrow (s, s_1)}^{(n)} x_{(k, s')} x_{(k_1, s_1)} - r_{k, k_1, (s, s_1) \rightarrow (s', s'_1)}^{(n)} x_{(k, s)} x_{(k_1, s'_1)}) \end{aligned} \quad (3)$$

The first term corresponds to unilateral transitions while the second term corresponds to transitions caused by pairwise interactions. Note that compared to (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  $\mathbf{x} \in \mathcal{X}^{(n)} \subset \{0, 1\}^{n \times |\mathcal{S}|}$ , the above expression (3) can be extended to a function  $f^{(n)} : \text{conv}(\mathcal{X}^{(n)}) \subset [0, 1]^{n \times |\mathcal{S}|} \rightarrow \mathbb{R}^{n \times |\mathcal{S}|}$ , where  $\text{conv}(\mathcal{X}^{(n)})$  denotes the convex hull of  $\mathcal{X}$ . For a given initial condition  $\mathbf{x} \in \mathcal{X}^{(n)}$ , we define the mean field approximation of the heterogeneous population model as the solution of the ODE

<sup>3</sup>We give the general drift definition in Appendix B.1

$\frac{d}{dt}\phi^{(n)}(\mathbf{x}, t) = f^{(n)}(\phi^{(n)}(\mathbf{x}, t))$  that starts in  $\mathbf{x} = \mathbf{X}^{(n)}(0)$ , and we denote by  $\phi^{(n)}(\mathbf{x}, t)$  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  $\phi(\mathbf{x}, t)$  takes values in a bounded set. Notice, from the definition of the ODE, that  $\phi_{(k,s)}^{(n)} \in [0, 1]$ , and  $\sum_s \phi_{(k,s)}^{(n)}(\mathbf{x}, t) = 1$  for all  $t$  and  $\mathbf{x}$  in  $\text{conv}(\mathcal{X}^{(n)})$ .

## 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  $s_1, s'_1, s_2, s'_2, \dots \in \mathcal{S}$  and  $k_1, k_2, \dots \in \{1, \dots, n\}$  we have:

$$\bar{r}_{k_1, \dots, k_d, (s_1, \dots, s_d) \rightarrow (s'_1, \dots, s'_d)}^{(n)} \leq \bar{r}. \quad (4)$$

**THEOREM 4.1.** *Assume that the model, defined in Section 3.2, satisfies (4). Let  $\phi^{(n)}(\mathbf{x}, t)$  be the solution of the ODE introduced in Section 4.1 with initial condition  $\mathbf{X}^{(n)}(0) = \mathbf{x} \in \mathcal{X}^{(n)}$  and drift  $f^{(n)}$ . Then, for  $(k, s) \in \{1, \dots, n\} \times \mathcal{S}$  and  $t < \infty$ ,*

$$\mathbb{P}(S_k(t) = s) = \mathbb{E} \left[ X_{(k,s)}^{(n)}(t) \right] = \phi_{(k,s)}^{(n)}(\mathbf{x}, t) + O(1/n). \quad (5)$$

**KEY IDEAS OF THE PROOF.** A complete proof is provided in Section 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  $\phi$  w.r.t. the initial condition and time is given. In the first part of the proof, we borrow ideas from [13, 25] and use a Taylor's expansion of  $\phi^{(n)}$  to show that  $\mathbb{P}(S_k(t) = s) - \phi_{(k,s)}^{(n)}(\mathbf{x}, t)$  can be bounded by a weighted sum of  $\frac{\partial^2 \phi_{(k,s)}^{(n)}(\mathbf{x}, t)}{\partial x_{(k',s')} \partial x_{(k'',s'')}}$ . 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.  $\square$

The statement of the theorem can be interpreted as saying that the probability of an object  $k$  in the Markov chain  $\mathbf{X}$  to be in state  $s$  is approximated by  $\phi_{(k,s)}$  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  $Z_s^{(n)}(t) = \frac{1}{n} \sum_{k=1}^n X_{(k,s)}(t)$ , the stochastic process describing the fraction of objects which are in state  $s$ . It should be noted that the solution of the ODE  $\phi_{(k,s)}$ , taking values in  $[0, 1]$ , is not close to the value of  $X_{(k,s)}$ , 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 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,  $\phi_{(k,\text{in})}^{(n)}(\mathbf{x}, t)$ . We emphasize that  $X_{(k,\text{in})}(t)$  is never close to  $\phi_{(k,\text{in})}^{(n)}(\mathbf{x}, t)$ , 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.

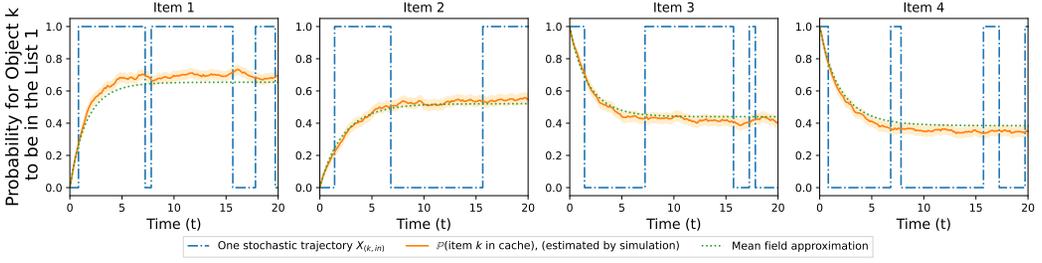


Fig. 1. Behavior of the RANDOM policy for a cache of size two and a total of four objects. For each of the four objects, we compare the stochastic system with the mean field approximations.

### 4.3 Accuracy of the Refined Mean Field Approximation

In [14, 16], 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  $\mathbf{X} = \mathbf{X}^{[1 \text{ copy}]}$ . The mean field approximation of  $\mathbf{X}^{[C \text{ copies}]}$  is also  $\phi^{(n)}$ . The idea of [14] 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

$$\mathbb{E} \left[ \mathbf{X}_{(k,s)}^{[C \text{ copies}]}(t) \right] = \phi_{(k,s)}^{(n)}(\mathbf{x}, t) + \frac{1}{C} v_{(k,s)}^{(n)}(\mathbf{x}, t) + O(1/C^2). \quad (6)$$

The values  $v_{(k,s)}^{(n)}(t)$  are shown in [14] to satisfy a system of linear ordinary differential equations whose solution can be expressed in integral form as:

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

where  $Q_{(k_1, s_1), (k_2, s_2)}(\mathbf{x})$  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  $\mathbf{X} = \mathbf{x}$ . We formally introduce and elaborate more on the refinement terms in Appendix B.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 (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 (5) and (7), what this theorem shows is that the correction  $v_{(k,s)}^{(n)}(\mathbf{x}, t)$  is of order  $O(1/n)$  and is the leading term of the  $O(1/n)$ -term of Equation (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 4.2.** *Assume that the model, defined in Section 3.2, satisfies (4). Let  $\phi^{(n)}(\mathbf{x}, t)$  be the solution of the ODE introduced in Section 4.1 with initial condition  $\mathbf{X}^{(n)}(0) = \mathbf{x} \in X^{(n)}$  and drift  $f^{(n)}$ . Let  $v^{(n)}(\mathbf{x}, t)$  be the solution of the refinement term explicitly defined in Appendix B.2. Then, for  $(k, s) \in \{1, \dots, n\} \times \mathcal{S}$  and  $t < \infty$ ,*

$$\mathbb{P}(S_k(t) = s) = \phi_{(k,s)}^{(n)}(\mathbf{x}, t) + v_{(k,s)}^{(n)}(\mathbf{x}, t) + O(1/n^2). \quad (7)$$

**KEY IDEAS OF THE PROOF.** A complete proof is provided in Section 6.4. The proof of Theorem 4.2 uses the same methodology as the proof of Theorem 4.1 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 4.2, 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  $\phi$  with respect to its initial condition are.  $\square$

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 [14] to compute a second expansion term. This expansion depends on up to the fourth derivative of  $\phi$ . 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  $\phi$ . 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 paper, 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 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 5.1.

*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 [14, 17] 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 4.1 and 4.2 can be used to obtain the following much sharper bound:

$$\mathbb{E} \left[ \mathbf{X}_{(k,s)}^{[C \text{ copies}]}(t) \right] = \phi_{(k,s)}^{(n)}(\mathbf{x}, t) + \frac{1}{C} v_{(k,s)}^{(n)}(\mathbf{x}, t) + O\left(\frac{1}{(Cn)^2}\right). \quad (8)$$

The (only) difference between this equation and (6) is that the term  $O(1/C^2)$  of (6) is replaced by the much smaller term  $O(1/(Cn)^2)$ . This implies that the accuracy of the mean field interaction

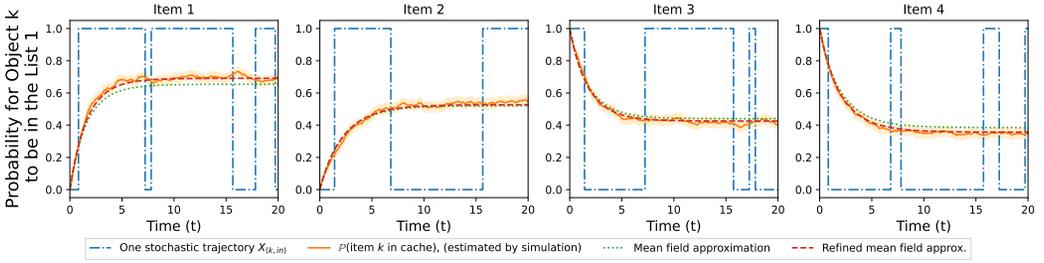


Fig. 2. Behavior of the RANDOM policy for a cache of size 2 and a total of four objects. For each of the four objects, we compare the stochastic system with the mean field and refined mean field approximations.

model does not depend on the number of homogeneous copies but only on the total number of objects.

To see why (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 4.1 and 4.2 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 (8).

#### 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  $S^{(n)}$ . Indeed, the continuous-time Markov chain  $S^{(n)}$  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 B.2, the refinement term  $v^{(n)}$  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  $\mathbf{x}$ : 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 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 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 is done by using a straightforward implementation of the ODEs given in Appendix B.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).

Table 1. Computation time of the mean field and refined mean field approximation for the RANDOM model as a function of the number of objects  $n$ .

$n$	Transient up to $T = 1000$		Steady-state (=Fixed point)	
	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

\* “—” means that the ODE solver did not finish before 30 seconds.

• (Steady-state). The last two columns correspond to the computation of the limiting value as  $t$  goes to infinity:  $x_{(k,\cdot)}^{(n)}(\infty) = \lim_{t \rightarrow \infty} x_{(k,\cdot)}^{(n)}(t)$  and  $v_{(k,\cdot)}^{(n)}(\infty) = \lim_{t \rightarrow \infty} v_{(k,\cdot)}^{(n)}(t)$ . 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 4.1, 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  $v_{(k,\cdot)}^{(n)}$  depends on the object’s identity.

## 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 4.1 and 4.2 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.

### 5.1 Application to a Cache Replacement Algorithm: the RANDOM(m) Model

As a first example we consider the list-based RANDOM(m) cache replacement policy which was introduced in [15, 21]. 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 [15] 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 [15] 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 [15] 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 [7], Theorem 4.2 provides theoretical support by showing that the error of the refined approximation is  $O(1/n^2)$ .

**5.1.1 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  $\lambda_k$ . 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( $\mathbf{m}$ ) 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  $\mathbf{X}$  are:

$$\mathbf{X} \mapsto \mathbf{X} + \mathbf{e}_{(k,s+1)} - \mathbf{e}_{(k,s)} + \mathbf{e}_{(k_1,s)} - \mathbf{e}_{(k_1,s+1)} \quad \text{at rate } \frac{\lambda_k}{m_{s+1}^{(n)}} X_{(k,s)} X_{(k_1,s+1)}. \quad (9)$$

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

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

The mean field approximation is the solution of the ODE  $\dot{\mathbf{x}} = f(\mathbf{x})$ . Note that this ODE is the same as the one given in [15].

The transitions of this model are pairwise interactions with  $b_{k,k_1,s,s+1,s+1,s}^{(n)} = \frac{\lambda_k}{m_{s+1}^{(n)}/n}$  as defined in (2b). There is no unilateral transition. To apply Theorems 4.1 and 4.2, 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  $\lambda_k$  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.

**5.1.2 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 [1]. We implement the Markov chain, the drift  $f$ , the drift derivatives and  $\mathbf{Q}$  for the approximations based on the transitions (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 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  $s = 0, 1, \dots, 3$ , against the "true" value  $\sum_{k=1}^n \lambda_k \mathbb{E}[X_{(k,s)}]$ ,  $s = 0, 1, \dots, 3$  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  $m$  is small enough.

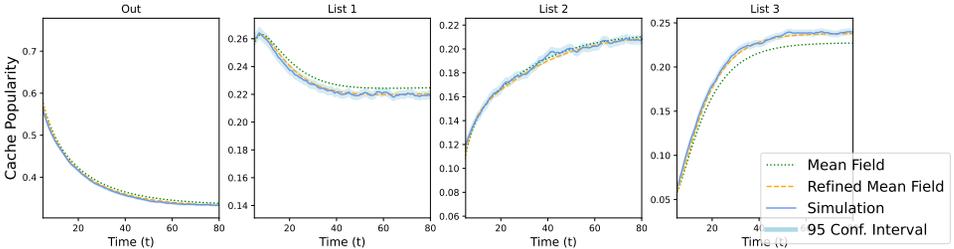


Fig. 3. Transient state comparison of cache popularities.

**5.1.3 Steady-state Analysis.** The previous results show that the mean field and refined mean field can accurately approximate the transient behavior of the  $\text{RANDOM}(m)$  policy. In Figure 4, we compare the steady-state values 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 4.1 and 4.2 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.

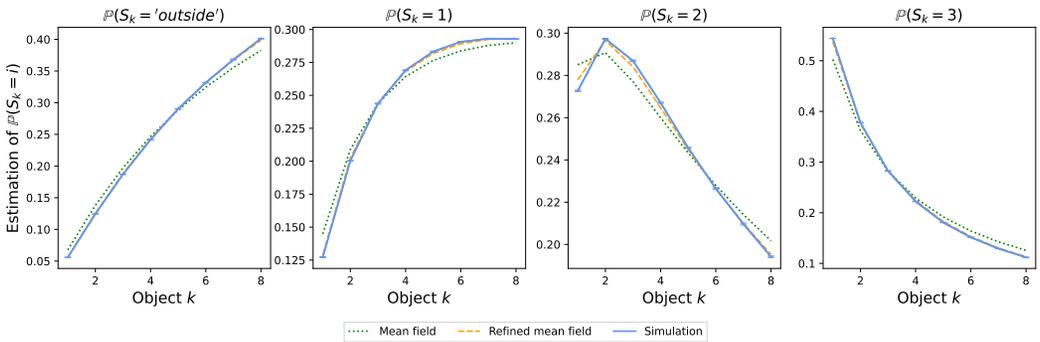


Fig. 4. Steady-state probabilities estimated by simulation, mean field and refined mean field approximation.

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

Table 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.

$n$	Mean field			Refined mean field			Simulation	
	Error	$n \times \text{Error}$	(time)	Error	$n^2 \times \text{Error}$	(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)

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.

We show in Appendix C.1 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  $\mathbf{m}$ . We call this value  $\pi_{(k,s)}^{\text{exact}}$ . We also compute an estimation  $\pi_{(k,s)}^{\text{method}}$  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

$$\text{Error}(\text{method}) = \frac{1}{n} \sum_{k,s} \left| \pi_{(k,s)}^{\text{method}} - \pi_{(k,s)}^{\text{exact}} \right|. \quad (10)$$

We report in Table 2 the error of the three estimation methods (mean field, refined mean field and simulation). By Theorem 4.1 and 4.2, 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 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 2 for  $n = 50$  (i.e., extremely small).

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

**5.2.1 Model and Approximations.** In our second example, we consider a variation of the well studied two-choice model [28]. 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 [30] 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  $\lambda n$ , we call  $\lambda$  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  $\mu_i$ . 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  $\mathbf{X} = (X_{(k,s)})_{(k,s)}$  is a Markov chain whose transitions are (for all  $k, k_1 \in \{1 \dots n\}$ ):

$$\mathbf{X} \mapsto \mathbf{X} - e_{(k,s)} + e_{(k,s-1)} \quad \text{at rate } \mu_k X_{(k,s)}, \quad (11a)$$

$$\mathbf{X} \mapsto \mathbf{X} + e_{(k,s+1)} - e_{(k,s)} \quad \text{at rate } (2\lambda n \mathbf{1}_{\{s_1 \geq s+1\}} + \lambda n \mathbf{1}_{\{s_1 = s\}}) \frac{X_{(k,s)} X_{(k_1,s_1)}}{n}. \quad (11b)$$

In the above equation, the first type of transition (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 (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  $r_{k,(s) \rightarrow (s-1)}^{(n)} = \mu_k$  and pairwise interactions with  $r_{k,k_1,(s,s_1) \rightarrow (s+1,s_1)}^{(n)} = (2\lambda \mathbf{1}_{\{s_1 \geq s+1\}} + \lambda \mathbf{1}_{\{s_1 = s\}})/n$ . The bound (4), required to apply Theorems 4.1 and 4.2, is verified when the values of  $\lambda$  and  $\mu_k$  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 (11b) can then be rewritten as

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

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

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

**5.2.2 Numerical Comparison.** As for the caching example, we adapt methods of the toolbox [1] 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 (11a) and (11b) we implement the Markov chain and define the drift  $f$ , the derivatives of  $f$  and the tensor  $\mathbf{Q}$ . 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  $19 \times 10^6$  events of the Markov chain after a warp-up of  $5 \times 10^5$  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 7 that the queue length distribution vanishes very quickly for high queue sizes. We collect all simulation results in Figures 5, 6 and 7.

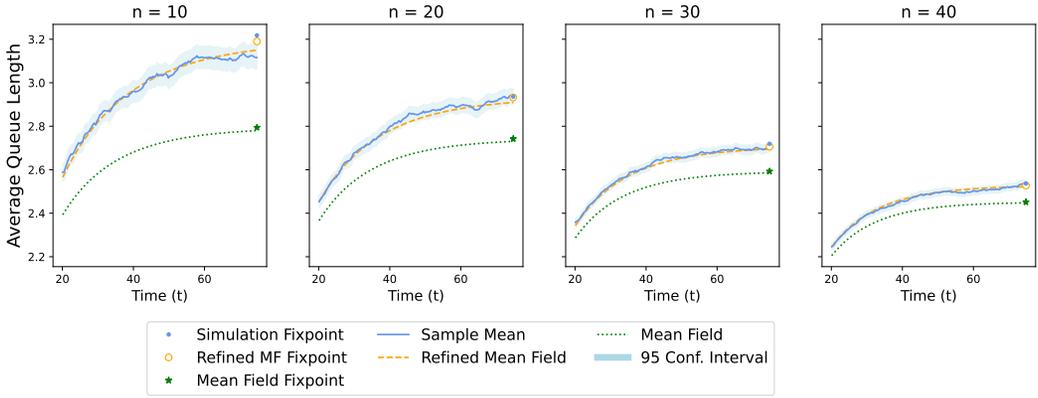
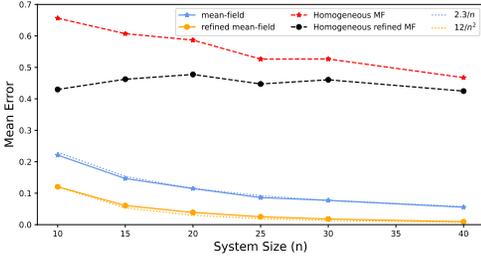


Fig. 5. Average Queue Size of simulation mean vs. mean field vs. refined mean field approximation.

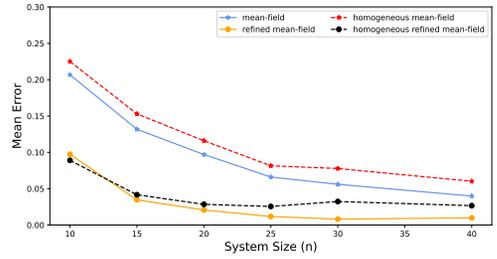
Figure 5 shows the average queue size of the system. We plot the sample mean of the average queue size with a 95-percent confidence interval against the average queue size calculated from the mean field and refined mean field approximation. We observe that as  $n$  grows, both the mean field and the 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 “homogeneous” 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  $\text{Error}(\text{method}) = \frac{1}{n} \sum_{(k,s)} \left| \mathbb{E} [X_{(k,s)}(\infty)] - \pi_{(k,s)}^{\text{method}} \right|$  the mean error, where  $\mathbb{E} [X_{(k,s)}(\infty)]$  is the steady-state of the stochastic system, approximated by simulation, and  $\pi_{(k,s)}^{\text{method}}$  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 6a and 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.

Fig. 6. Steady-state mean error comparison of heterogeneous and homogeneous models.

Last, in Figure 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 [16] for the refined mean field for homogeneous systems.

## 6 PROOFS

This section contains the proofs of the main theorems. After recalling some notations in Section 6.1, we start with a first technical lemma in Section 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 4.1 in Section 6.3 and Theorem 4.2 in Section 6.4. To ease the reading, some technical lemmas – whose proof are not complicated but long and technical – are postponed to the appendix.

### 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  $\mathbf{X}, f, \dots$  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  $h : \mathcal{X} \times \mathbb{R}^+ \rightarrow \mathbb{R}$ , we denote by  $D_{\mathbf{x}}h$  the derivative of  $h$  with respect to the first coordinate  $\mathbf{x}$  and by  $D_t h$  the derivative with respect to the second coordinate. This means that for a given pair  $(\mathbf{y}, s) \in \mathcal{X} \times \mathbb{R}^+$ , the quantity  $D_{\mathbf{x}}h(\mathbf{y}, s)$  and  $D_t h(\mathbf{y}, s)$  are the derivatives of  $h$  with respect to  $\mathbf{x}$  and  $t$  evaluated at the point  $(\mathbf{y}, s)$ .

For convenience, we will denote by  $K_{\mathbf{x}, \mathbf{x}'}^{(n)}$  the rate at which the Markov chain  $\mathbf{X}$  jumps from  $\mathbf{x}$  to  $\mathbf{x}'$  for  $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ . With this notation, for intuition, the transitions (2a) and (2b) correspond to (for

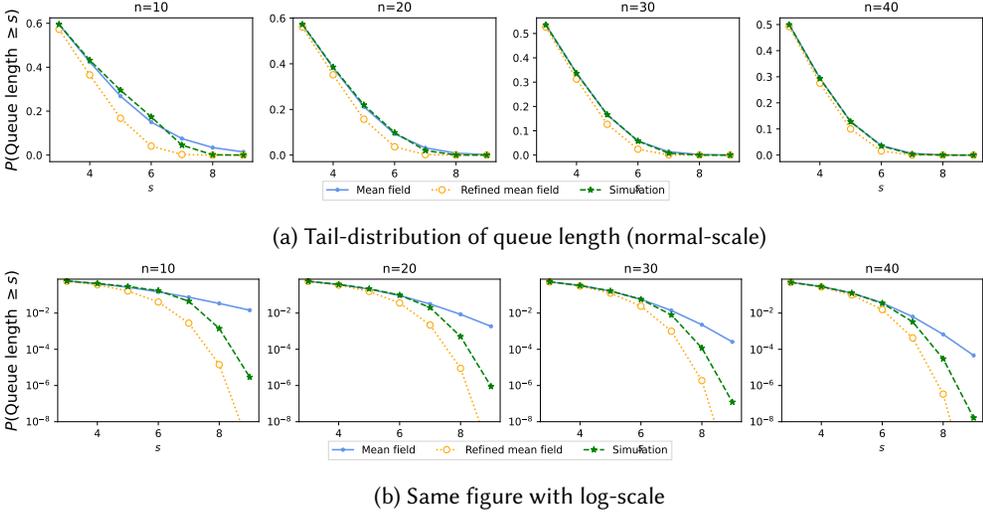


Fig. 7. Queue length distribution of simulated expectation, mean field approximation and refined mean field approximation in steady-state for system sizes  $n = 10, 20, 30, 40$ .

$\mathbf{x} \in \mathcal{X}$  and  $k, k_1 \in \{1, \dots, n\}$ ,  $s, s', s_1, s'_1 \in \mathcal{S}$  with  $(s, s_1) \neq (s', s'_1)$

$$K_{\mathbf{x}, \mathbf{x} + \mathbf{e}_{(k, s')} - \mathbf{e}_{(k, s)}}^{(n)} = r_{k, (s) \rightarrow (s')} X_{(k, s)}^{(n)},$$

$$K_{\mathbf{x}, \mathbf{x} + \mathbf{e}_{(k, s')} - \mathbf{e}_{(k, s)} + \mathbf{e}_{(k_1, s'_1)} - \mathbf{e}_{(k_1, s_1)}}^{(n)} = \frac{1}{n} r_{k, k_1, (s, s_1) \rightarrow (s', s'_1)} X_{(k, s)} X_{(k_1, s_1)}.$$

## 6.2 Comparison of the Generators

LEMMA 6.1. Let  $\mathbf{X}(t)$  be the continuous time Markov chain defined in Section 3.2. Let  $\boldsymbol{\phi}(\mathbf{x}, t)$  be the value at time  $t$  of the solution of the ODE  $\frac{d}{dt} \boldsymbol{\phi}(\mathbf{x}, t) = f(\boldsymbol{\phi}(\mathbf{x}, t))$  with initial condition  $\mathbf{x} \in \mathcal{X}$ . We have

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

PROOF. The following calculations are based on the ideas used in the proofs of [25, Theorem 1] and [13, Theorem 3.1]. By defining  $\boldsymbol{\psi}(\tau) = \mathbb{E}[\boldsymbol{\phi}(\mathbf{X}(\tau), t - \tau)]$  we can rewrite  $\mathbb{E}[\mathbf{X}(t) - \boldsymbol{\phi}(\mathbf{X}(0), t)] = \boldsymbol{\psi}(t) - \boldsymbol{\psi}(0)$ . At first, we derive the time derivative of  $\boldsymbol{\psi}$ . We start by looking at the expected change at a given time  $\tau$ ,  $\frac{d}{ds} \mathbb{E}[\boldsymbol{\phi}(\mathbf{X}(\tau + s), t - (\tau + s)) \mid \mathbf{X}(\tau)]|_{s=0}$ , which can be written as

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

In the limit, the first difference corresponds to the generator of  $\mathbf{X}$  at  $\tau$  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

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

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

### 6.3 Proof of Theorem 4.1 (Mean Field Approximation)

By Lemma 6.1, we have

$$\mathbb{E}[\mathbf{X}(t) - \phi(\mathbf{X}(0), t)] = \int_0^t \mathbb{E} \left[ \sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{X}(\tau), \mathbf{x}'} (\phi(\mathbf{x}', t - \tau) - \phi(\mathbf{X}(\tau), t - \tau)) - D_x \phi(\mathbf{X}(\tau), t - \tau) f(\mathbf{X}(\tau)) \right] d\tau. \quad (12)$$

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

$$\phi(\mathbf{x}', \tau) - \phi(\mathbf{x}, \tau) = D_x \phi(\mathbf{x}, \tau) (\mathbf{x}' - \mathbf{x}) + R_1(\mathbf{x}, \mathbf{x}', \tau), \quad (13)$$

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

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

Moreover, by definition of the drift, one has  $\sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{x}, \mathbf{x}'} (\mathbf{x}' - \mathbf{x}) = f(\mathbf{x})$ . Combining this with (13) and plugging this into equation (12) shows that

$$\mathbb{E}[\mathbf{X}(t) - \phi(\mathbf{X}(0), t)] = \int_0^t \mathbb{E} \left[ \sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{X}(\tau), \mathbf{x}'} R_1(\mathbf{X}(\tau), \mathbf{x}', t - \tau) \right] d\tau. \quad (14)$$

To conclude the proof, we show in Lemma D.2 that  $\sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{x}, \mathbf{x}'} R_1(\mathbf{x}, \mathbf{x}', \tau)$  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  $\phi$  as a function of the initial condition. This is where we use the assumptions on the rates  $r$ .

### 6.4 Proof of Theorem 4.2 (Refined mean field approximation)

The proof of Theorem 4.2 uses the same methodology as the proof of Theorem 4.1 with two additional ideas: The first is to use a second-order Taylor expansion instead of the first order expansion used in (13). The second is to express the refinement term  $\mathbf{v}$  as an integral of quantities that depend on the second derivative of  $\phi$ .

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

$$\phi(\mathbf{x}', \tau) - \phi(\mathbf{x}, \tau) = D_x \phi(\mathbf{x}, \tau) (\mathbf{x}' - \mathbf{x}) + \sum_{i,j \in \mathcal{I}} Q_{i,j}(\mathbf{x}) \frac{\partial^2 \phi}{\partial x_i \partial x_j} (\mathbf{x}, \tau) + R_2(\mathbf{x}, \mathbf{x}', \tau), \quad (15)$$

where the remainder term  $R_2$  is equal to

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

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

and  $Q_{i,j}(\mathbf{x})$  is given by  $\sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{x},\mathbf{x}'}(x'_i - x_i)(x'_j - x_j)$  which we formally introduce in Appendix B.2.

To simplify notations, let  $\mathbf{g}(\mathbf{x}, \tau) := \sum_{i,j \in \mathcal{I}} Q_{i,j}(\mathbf{x}) \frac{\partial^2 \phi}{\partial x_i \partial x_j}(\mathbf{x}, \tau)$ . Similarly to (14), we have

$$\mathbb{E}[\mathbf{X}(t)] - \phi(\mathbf{X}(0), t) = \underbrace{\frac{1}{2} \int_0^t \mathbb{E}[\mathbf{g}(\mathbf{X}(\tau), t-\tau)] d\tau}_{\approx \mathbf{v}(\mathbf{x}, t) + O(\frac{1}{n^2}) \text{ by D.3 and D.4.}} + \underbrace{\int_0^t \mathbb{E}[\sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{X}(\tau), \mathbf{x}'} R_2(\mathbf{X}(\tau), \mathbf{x}', t-\tau)] d\tau}_{O(1/n^2) \text{ by Lemma D.2.}}. \quad (16)$$

By using the approach we used for  $R_1$ , we prove in Lemma D.2 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  $\phi(\mathbf{x}, t)$  with respect to its initial condition. We are then left with the first term of Equation (16). By Lemma D.3, the refinement term  $\mathbf{v}$  can be expressed in integral form as  $v_{(k,s)}(\mathbf{x}, t) = \frac{1}{2} \int_0^t g_{(k,s)}(\phi(\mathbf{x}, \tau), \tau) d\tau$ . This shows that

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

We show in Lemma D.4 that the above term is of order  $O(1/n^2)$ . This requires to bound up to the fourth derivative of  $\phi$  with respect to its initial condition. Plugging everything into Equation (16) shows that  $\mathbb{E}[\mathbf{X}(t)] - \phi(\mathbf{X}(0), t) - \mathbf{v}(\mathbf{X}(0), t) = O(1/n^2)$  and concludes the proof.

## 7 CONCLUSION

In this paper, 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 paper 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 paper, namely Theorem 4.1 and 4.2, 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 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).

## ACKNOWLEDGMENTS

This work is supported by the French National Research Agency (ANR) through REFINO Project under Grant ANR-19-CE23-0015.

## REFERENCES

- [1] Sebastian Allmeier and Nicolas Gast. 2021. rmftool-A library to Compute (Refined) Mean Field Approximation (s). In *TOSME 2021*.
- [2] François Baccelli and Thibaud Tallefumier. 2019. Replica-mean-field limits for intensity-based neural networks. *SIAM Journal on Applied Dynamical Systems* 18, 4 (2019), 1756–1797.
- [3] Anton Braverman. 2017. *Stein’s method for steady-state diffusion approximations*. Ph.D. Dissertation. Cornell University.
- [4] Anton Braverman and Jim Dai. 2017. Stein’s method for steady-state diffusion approximations of  $M/Ph/n + M$  systems. *The Annals of Applied Probability* 27, 1 (2017), 550–581.
- [5] Anton Braverman, JG Dai, and Jiekun Feng. 2017. Stein’s method for steady-state diffusion approximations: an introduction through the Erlang-A and Erlang-C models. *Stochastic Systems* 6, 2 (2017), 301–366.
- [6] Anton Braverman, J. G. Dai, and Xiao Fang. 2020. High Order Steady-State Diffusion Approximations. *arXiv:2012.02824* (Dec. 2020). [arXiv:2012.02824](https://arxiv.org/abs/2012.02824)
- [7] Giuliano Casale and Nicolas Gast. 2020. Performance analysis methods for list-based caches with non-uniform access. *IEEE/ACM Transactions on Networking* (2020).
- [8] Hao Che, Zhijun Wang, and Ye Tung. 2001. Analysis and design of hierarchical web caching systems. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 3. IEEE, 1416–1424.
- [9] Asit Dan and Don Towsley. 1990. 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*. 143–152.
- [10] Guilherme Ferraz de Arruda, Francisco A. Rodrigues, and Yamir Moreno. 2018. Fundamentals of Spreading Processes in Single and Multilayer Complex Networks. *Physics Reports* 756 (Oct. 2018), 1–59. <https://doi.org/10.1016/j.physrep.2018.06.007> [arXiv:1804.08777](https://arxiv.org/abs/1804.08777)
- [11] Ronald Fagin. 1977. Asymptotic miss ratios over independent references. *J. Comput. System Sci.* 14, 2 (1977), 222–250.
- [12] Christine Fricker, Philippe Robert, and James Roberts. 2012. A versatile and accurate approximation for LRU cache performance. In *2012 24th international teletraffic congress (ITC 24)*. IEEE, 1–8.
- [13] Nicolas Gast. 2017. Expected values estimated via mean-field approximation are  $1/N$ -accurate. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1, 1 (2017), 17.
- [14] Nicolas Gast, Luca Bortolussi, and Mirco Tribastone. 2019. Size Expansions of Mean Field Approximation: Transient and Steady-State Analysis. *Performance Evaluation* 129 (Feb. 2019), 60–80. <https://doi.org/10.1016/j.peva.2018.09.005>
- [15] Nicolas Gast and Benny Van Houdt. 2015. 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*. ACM Press, Portland, Oregon, USA, 123–136. <https://doi.org/10.1145/2745844.2745850>
- [16] Nicolas Gast and Benny Van Houdt. 2017. A Refined Mean Field Approximation. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1, 2 (Dec. 2017), 33:1–33:28. <https://doi.org/10.1145/3154491>
- [17] Nicolas Gast and Benny Van Houdt. 2017. A refined mean field approximation. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1, 2 (2017), 1–28.
- [18] M. Gabriela M. Gomes, Ricardo Aguas, Rodrigo M. Corder, Jessica G. King, Kate E. Langwig, Caetano Souto-Maior, Jorge Carneiro, Marcelo U. Ferreira, and Carlos Penha-Gonçalves. 2020. *Individual Variation in Susceptibility or Exposure to SARS-CoV-2 Lowers the Herd Immunity Threshold*. Preprint. *Epidemiology*. <https://doi.org/10.1101/2020.04.27.20081893>
- [19] Ramon Grima. 2010. 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.
- [20] Ramon Grima, Philipp Thomas, and Arthur V. Straube. 2011. How accurate are the nonlinear chemical Fokker-Planck and chemical Langevin equations? *The Journal of Chemical Physics* 135, 8 (2011).
- [21] M Hazewinkel, F Calogero, Yu. I Manin, A. H. G Rinnooy Kan, and G.-C Rota. 1987. *Stochastic Analysis of Computer Storage*. Springer Netherlands, Dordrecht.
- [22] R. Hirade and T. Osogami. 2010. Analysis of Page Replacement Policies in the Fluid Limit. *Oper. Res.* 58 (July 2010), 971–984. <https://doi.org/10.1287/opre.1090.0761>

- [23] Liam Hodgkinson, Ross McVinish, and Philip K Pollett. 2020. Normal approximations for discrete-time occupancy processes. *Stochastic Processes and their Applications* 130, 10 (2020), 6414–6444.
- [24] Bo Jiang, Philippe Nain, and Don Towsley. 2018. 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), 1–31.
- [25] Vassili N. Kolokoltsov, Jiajie Li, and Wei Yang. 2012. Mean Field Games and Nonlinear Markov Processes. *arXiv:1112.3744* (April 2012). [arXiv:1112.3744](https://arxiv.org/abs/1112.3744)
- [26] Thomas G. Kurtz. 1978. Strong Approximation Theorems for Density Dependent Markov Chains. *Stochastic Processes and their Applications* 6, 3 (Feb. 1978), 223–240. [https://doi.org/10.1016/0304-4149\(78\)90020-0](https://doi.org/10.1016/0304-4149(78)90020-0)
- [27] Marc Mézard, Giorgio Parisi, and Miguel Angel Virasoro. 1987. Spin glass theory and beyond: An Introduction to the Replica Method and Its Applications. *World Scientific Publishing Company* (1987), 232–237.
- [28] M. Mitzenmacher. Oct./2001. The Power of Two Choices in Randomized Load Balancing. *IEEE Transactions on Parallel and Distributed Systems* 12, 10 (Oct./2001), 1094–1104. <https://doi.org/10.1109/71.963420>
- [29] Antonio Montalbán, Rodrigo M. Corder, and M. Gabriela M. Gomes. 2020. Herd Immunity under Individual Variation and Reinfection. *arXiv:2008.00098* (Nov. 2020). [arXiv:2008.00098](https://arxiv.org/abs/2008.00098)
- [30] Arpan Mukhopadhyay and Ravi R. Mazumdar. 2015. Analysis of Load Balancing in Large Heterogeneous Processor Sharing Systems. *arXiv:1311.5806* (Feb. 2015). [arXiv:1311.5806](https://arxiv.org/abs/1311.5806)
- [31] Charles Stein. 1986. Approximate computation of expectations. *Lecture Notes-Monograph Series* 7 (1986), i–164.
- [32] N. Tsukada, R. Hirade, and N. Miyoshi. 2012. Fluid Limit Analysis of FIFO and RR Caching for Independent Reference Models. *Perform. Eval.* 69, 9 (Sept. 2012), 403–412. <https://doi.org/10.1016/j.peva.2012.05.008>
- [33] Benny Van Houdt. 2013. A Mean Field Model for a Class of Garbage Collection Algorithms in Flash-Based Solid State Drives. *ACM SIGMETRICS Performance Evaluation Review* 41, 1 (June 2013), 191–202. <https://doi.org/10.1145/2494232.2465543>
- [34] N. G. van Kampen. 2007. *Stochastic processes in physics and chemistry*. Elsevier, Amsterdam; Boston; London.
- [35] Lei Ying. 2015. On the Rate of Convergence of Mean-Field Models: Stein’s Method Meets the Perturbation Theory. *arXiv preprint arXiv:1510.00761* (2015).
- [36] Lei Ying. 2016. On the Approximation Error of Mean-Field Models. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*. ACM, 285–297.
- [37] Lei Ying. 2017. 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 (2017), 12.

## A NOTATION LIST

$n$	system size
$a_{\{\dots\}}, b_{\{\dots\}}$	unilateral and pairwise transition parameters; Equations (2a), (2b)
$S(t)$	Markov chain describing the heterogeneous population model; Section 3.1
$X(t)$	binary based representation of the model described by $S(t)$ ; Section 3.2
$\mathcal{S}$	finite state space of objects in the population model described by $S$ and $X$
$s, \hat{s}, s', s_1$	notation for states
$k, \hat{k}, k', k_1$	notation for objects
$X_{(k,s)}(t)$	entry of the Markov chain $X$ indicating if object $k$ is in state $s$ at time $t$
$f, \mathbf{Q}, \mathbf{R}$	drift of the stochastic system and related tensors; Section 4.1, Appendix B.2, Lemma D.2
$\phi(\mathbf{x}, t)$	solution to the ODE given by the drift $f$ of the system; Section 4.1
$\mathbf{v}(\mathbf{x}, t)$	refinement term; Section 4.3, Appendix B.2
$\mathbf{w}(\mathbf{x}, t)$	solution to the second set of differential equations of the refinement; Appendix B.2
$t, \tau, \nu$	time and integration variables
$\mathcal{I}, \bar{\mathcal{I}}_k$	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 $\bar{\mathcal{I}}_k$
$\mathbf{x}, \mathbf{y}, \mathbf{z}$	initial conditions for Markov chain $X$ and corresponding mean field approximation
$K_{\mathbf{x}, \mathbf{x}'}$	transition rate for Markov chain $X$ from state $\mathbf{x}$ to $\mathbf{x}'$ , Lemma D.2
$L_{1,2}$	bounds for first or second partial derivatives of the drift $f$
$\mathbf{x} \otimes \mathbf{y}$	Kronecker product of $\mathbf{x}$ and $\mathbf{y}$
$\mathbf{x}^{\otimes 2}, \mathbf{x}^{\otimes 3}$	Kronecker product of $\mathbf{x}$ with itself ( $\mathbf{x}^{\otimes 2} = \mathbf{x} \otimes \mathbf{x}$ ; $\mathbf{x}^{\otimes 3} = \mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x}$ ); Lemma D.2
$D, D^2, D_{\mathbf{x}}$	1st and 2nd order derivative, derivative with respect to $\mathbf{x}$
$\frac{\partial}{\partial x_i}$	partial derivative with respect to $x_i$
$\frac{d}{dt}, \frac{\partial}{\partial t}, \dot{\mathbf{x}}(t)$	derivative with respect to time $t$
$\mathbf{1}_{\{a>b\}}$	indicator function, i.e., indicating if $a > b$

## B EQUATION FOR THE MEAN FIELD AND REFINED MEAN FIELD APPROXIMATIONS

### B.1 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  $s$  ( $s' \rightarrow s$ ) or leaves state  $s$  ( $s \rightarrow s'$ ). By considering all these interactions, the  $(k, s)$  component of the drift  $f^{(n)}(\mathbf{x})$  is

$$\begin{aligned}
& \sum_{s' \neq s} (r_{k,(s') \rightarrow (s)}^{(n)} \mathbf{x}(k, s') - r_{k,(s) \rightarrow (s')}^{(n)} \mathbf{x}(k, s)) \\
& + \frac{1}{n} \sum_{s', k_1, s_1, s'_1} (r_{k, k_1, (s', s'_1) \rightarrow (s, s_1)}^{(n)} \mathbf{x}(k, s') \mathbf{x}(k_1, s_1) - r_{k, k_1, (s, s_1) \rightarrow (s', s'_1)}^{(n)} \mathbf{x}(k, s) \mathbf{x}(k_1, s'_1)) \\
& + \sum_{d=3, \dots, d_{\max}} \frac{1}{n^{d-1}} \sum_{\substack{k_1, \dots, k_{d-1} \\ s_1, \dots, s_{d-1} \\ s', s'_1, \dots, s'_{d-1}}} r_{k, \dots, k_{d-1}, (s', \dots, s'_{d-1}) \rightarrow (s, \dots, s_{d-1})}^{(n)} \mathbf{x}(k, s') \cdots \mathbf{x}(k_{d-1}, s'_{d-1}) \\
& - r_{k, \dots, k_{d-1}, (s, \dots, s_{d-1}) \rightarrow (s', \dots, s'_{d-1})}^{(n)} \mathbf{x}(k, s) \cdots \mathbf{x}(k_{d-1}, s_{d-1})
\end{aligned} \tag{17}$$

In the above equation, we sum over all permutations such that the first objects is fixed to  $k$ . This counters the factor  $1/d$  of the definition of the rates. Therefore, the sum can be written as  $\frac{1}{n^{d-1}} \sum_{\substack{k_1, \dots, k_{d-1} \\ s_1, \dots, s_{d-1}}} r_{k, \dots, k_{d-1}, (s', \dots, s'_{d-1}) \rightarrow (s, \dots, s_{d-1})}^{(n)} \mathcal{X}^{(n)}(k, s') \cdots \mathcal{X}^{(n)}(k_{d-1}, s'_{d-1})$ . 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  $\sigma$  of the set  $\{1 \dots d\}$ , the rates satisfy  $r_{k_1, \dots, k_d, (s_1, \dots, s_d) \rightarrow (s'_1, \dots, s'_d)}^{(n)} = r_{k_{\sigma(1)}, \dots, k_{\sigma(d)}, (s_{\sigma(1)}, \dots, s_{\sigma(d)}) \rightarrow (s'_{\sigma(1)}, \dots, s'_{\sigma(d)})}^{(n)}$ .

## B.2 Definition of the Refined Mean Field Approximation

In this section, we show how the definition of the refinement term  $\mathbf{v}$  from [14] can be adapted and how it can be computed using the rates of the model introduced in section 3. In [14], 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  $\mathbf{v}$  to object-state pairs with the following set of ODEs (for better readability we suppress the dependence on  $n$  in the definitions)

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

with initial conditions  $\mathbf{v}(\mathbf{x}, 0) = 0$ ,  $\mathbf{w}(\mathbf{x}, 0) = 0$ . The values of  $\mathbf{v}$  and  $\mathbf{w}$  should be interpreted as the leading correction terms for the first moment and covariance of  $\mathbf{X}(t) - \boldsymbol{\phi}(\mathbf{x}, t)$ . The value of  $\mathbf{Q}$  is given by the expected change of the covariance of the stochastic system which, for a given state  $\mathbf{x}$ , is

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

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

$$\begin{aligned} Q_{(k, s), (k, s)}(\mathbf{x}) &= \sum_{s'} r_{k, (s) \rightarrow (s')} \mathcal{X}(k, s) + r_{k, (s') \rightarrow (s)} \mathcal{X}(k, s') \\ &\quad + \frac{1}{n} \sum_{k_1, s_1, s'_1, s'} r_{k, k_1, (s, s_1) \rightarrow (s', s'_1)} \mathcal{X}(k, s) \mathcal{X}(k_1, s_1) + r_{k, k_1, (s', s'_1) \rightarrow (s, s_1)} \mathcal{X}(k, s') \mathcal{X}(k_1, s'_1), \\ Q_{(k, s), (k, s')}(\mathbf{x}) &= -r_{k, (s) \rightarrow (s')} \mathcal{X}(k, s) - r_{k, (s') \rightarrow (s)} \mathcal{X}(k, s') \\ &\quad + \frac{1}{n} \sum_{k_1, s_1, s'_1} -r_{k, k_1, (s, s_1) \rightarrow (s', s'_1)} \mathcal{X}(k, s) \mathcal{X}(k_1, s_1) - r_{k, k_1, (s', s'_1) \rightarrow (s, s_1)} \mathcal{X}(k, s') \mathcal{X}(k_1, s_1), \\ Q_{(k, s), (k', s_1)}(\mathbf{x}) &= \frac{1}{n} \sum_{s', s'_1} r_{k, k', (s, s_1) \rightarrow (s', s'_1)} \mathcal{X}(k, s) \mathcal{X}(k', s_1) - r_{k, k', (s', s'_1) \rightarrow (s, s_1)} \mathcal{X}(k, s') \mathcal{X}(k', s'_1), \quad k \neq k'. \end{aligned}$$

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

$$\begin{aligned}
\frac{\partial f_{(k,s)}}{\partial x_{(k,s)}}(\mathbf{x}) &= -\sum_{s'} r_{k,(s)\rightarrow(s')} - \frac{1}{n} \sum_{k_1 \neq k, s_1, s'_1, s' \neq s} r_{k,k_1,(s,s_1)\rightarrow(s',s'_1)} x_{(k_1,s_1)} , \\
\frac{\partial f_{(k,s)}}{\partial x_{(k,\tilde{s})}}(\mathbf{x}) &= r_{k,(\tilde{s})\rightarrow(s)} + \frac{1}{n} \sum_{k_1 \neq k, s_1, s'_1} r_{k,k_1,(\tilde{s},s_1)\rightarrow(s,s'_1)} x_{(k_1,s_1)} && \tilde{s} \neq s, \\
\frac{\partial f_{(k,s)}}{\partial x_{(\tilde{k},\tilde{s})}}(\mathbf{x}) &= \frac{1}{n} \sum_{s',s'_1} r_{k,\tilde{k},(s',\tilde{s})\rightarrow(s,s'_1)} x_{(k,s')} - r_{k,\tilde{k},(s,\tilde{s})\rightarrow(s',s'_1)} x_{(k,s)} && \tilde{k} \neq k, \\
\frac{\partial^2 f_{(k,s)}}{\partial x_{(\hat{k},\tilde{s})} \partial x_{(k,s)}}(\mathbf{x}) &= -\frac{1}{n} \sum_{s'_1, s'} r_{k,\hat{k},(s,\tilde{s})\rightarrow(s',s'_1)} && \hat{k} \neq k, \\
\frac{\partial^2 f_{(k,s)}}{\partial x_{(\hat{k},\tilde{s})} \partial x_{(k,\tilde{s})}}(\mathbf{x}) &= \frac{1}{n} \sum_{s'_1} r_{k,\hat{k},(\tilde{s},\tilde{s})\rightarrow(s',s'_1)} && \hat{k} \neq k, \tilde{s} \neq s, \\
\frac{\partial^2 f_{(k,s)}}{\partial x_{(k,\tilde{s})} \partial x_{(\tilde{k},\tilde{s})}}(\mathbf{x}) &= \frac{1}{n} \sum_{s'_1} r_{k,\tilde{k},(\tilde{s},\tilde{s})\rightarrow(s,s'_1)} && \tilde{k} \neq k, \\
\frac{\partial^2 f_{(k,s)}}{\partial x_{(k,\tilde{s})} \partial x_{(\tilde{k},\tilde{s})}}(\mathbf{x}) &= -\frac{1}{n} \sum_{s',s'_1} r_{k,\tilde{k},(s,s')\rightarrow(\tilde{s},s'_1)} && \tilde{k} \neq k, \\
\frac{\partial^2 f_{(k,s)}}{\partial x_{(\hat{k},\tilde{s})} \partial x_{(\tilde{k},\tilde{s})}}(\mathbf{x}) &= 0 && \tilde{k}, \hat{k} \neq k.
\end{aligned}$$

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.

## C CACHE REPLACEMENT POLICIES

### C.1 Computation of the exact steady-state probabilities

It is shown in [15] that the steady-state distribution of the RANDOM( $\mathbf{m}$ ) 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 [15] 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  $\mathbf{S}$  is admissible for  $\mathbf{m}$  if the number of objects in list  $s$  is exactly  $m_s$  for all  $s \in \{1, \dots, S\}$ . Theorem 6 of [15] can be rephrased as follows: For any admissible state  $\mathbf{S}$ , the steady-state probability of  $\mathbf{S}$  is equal to

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

where  $C(\mathbf{m}, n) = \sum_{\mathbf{S} \text{ admissible for } \mathbf{m}} \prod_{k=1}^n (\lambda_k)^{S_k}$  is a constant such that the probabilities  $\pi(\mathbf{S})$  sum to one. Note that the constant  $C(\mathbf{m}, n)$  is not the same as the constant  $E(\mathbf{m}, n)$  defined in [15] because our configuration  $\mathbf{S}$  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:

$$\begin{aligned} C(\mathbf{m}, n) &= \sum_{S \text{ admissible for } \mathbf{m}} \prod_{k=1}^n (\lambda_k)^{S_k} = \sum_{s=0}^S \sum_{S \text{ admissible for } \mathbf{m} \text{ and object } n \text{ is in list } s} \prod_{k=1}^n (\lambda_k)^{S_k} \\ &= C(\mathbf{m}, n-1) + \sum_{s=1}^S (\lambda_n)^s C(\mathbf{m} - \mathbf{e}_s, n-1), \end{aligned}$$

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

$$\pi_{n,s}^{\text{exact}} = \frac{(\lambda_n)^s C(\mathbf{m} - \mathbf{e}_s, n-1)}{C(\mathbf{m}, n)}. \quad (18)$$

The above recurrence equations can be used to compute the exact value of  $\mathbb{P}(S_k^{(n)} = s)$  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  $\mathbf{m}$ , it is possible to compute an exact value for  $\mathbb{P}(S_k^{(n)} = s)$ . 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.

## C.2 Theorem 6 of [15] is a consequence of our results (and can be refined)

The cache replacement policy  $\text{RAND}(\mathbf{m})$  that we study in Section 5.1 is essentially the same<sup>5</sup> as the one studied in [15]. In [15], 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 [15] implies that for  $t \leq T$

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

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 [15] is correct. Lemma 1 implies is 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

<sup>5</sup>One difference between the two model is that we consider a continuous time model where object  $k$  is requested at rate  $\lambda_k$  and the authors of [15] 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.

implies that  $\mathbb{E} [\|M(t)\|_2^2] \leq ct$ . 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:

$$\begin{aligned} \mathbb{E} [\|M(t+1)\|_2^2] &= \mathbb{E} \left[ \|M(t)\|_2^2 + 2 \underbrace{\langle M(t+1) - M(t), M(t) \rangle}_{=0} + \|M(t+1) - M(t)\|_2^2 \right] \\ &\leq \mathbb{E} [\|M(t)\|_2^2] + c, \end{aligned} \quad (19)$$

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 [15] can be written as a supremum norm (it is a supremum norm) and we not think that it can be written as a scalar product. This implies that one cannot use the reasoning of Equation (19), 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  $\rho$  as:

$$\mathbb{E} [\|H_s(t) - \rho_s(t)\|^2] = \sum_{k_1, k_2} p_{k_1} p_{k_2} \mathbb{E} [(X_{k_1, s}(t) - x_{k_1, s}(t))(X_{k_2, s'}(t) - x_{k_2, s'}(t))] \quad (20)$$

We claim that the proof of Theorem 4.2 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  $w$  is defined in Appendix B.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 refined Theorem 6 of [15] by not only proving that the term is of order  $O(1/n)$  but also by providing the expansion term.

### C.3 Cache replacement policies: time to compute the fixed point.

In Table 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 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  $|nS|^2 = 4$  millions of variables).

Table 3. Time to compute the fixed point of the mean field and refined mean field approximation for the RAND( $\mathbf{m}$ ) model for various values of  $n$  and  $\mathbf{m}$ .

$n$	$\mathbf{m}$	$n \mathcal{S} $	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

## D TECHNICAL LEMMAS

### D.1 Bounds for Partial Derivatives of $\phi$

In Lemma D.1 we analyze the properties of the partial derivatives of  $\phi_{(k,s)}(\mathbf{x}, t)$  with respect to the initial condition  $\mathbf{x}$ . 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  $\phi_{(k,s)}(\mathbf{x}, t)$  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  $\phi_{(k,s)}(\mathbf{x}, t)$ ,  $\sum_{\hat{s} \in \mathcal{S}} \left| \frac{\partial \phi_{(k,\hat{s})}}{\partial \mathbf{x}_{(k,\hat{s})}}(\mathbf{x}, t) \right|$ , 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  $\sum_{\hat{s} \in \mathcal{S}} \left| \frac{\partial^2 \phi_{(k,\hat{s})}}{\partial \mathbf{x}_{(\tilde{k},\tilde{s})} \partial \mathbf{x}_{(\tilde{k},\tilde{s})}}(\mathbf{x}, t) \right|$ . 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  $(\tilde{k}, \tilde{s}), (\hat{k}, \hat{s}) \in \mathcal{I} \setminus \mathcal{I}_k$ , 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  $\phi_{(k,s)}(\mathbf{x}, t)$  can be bounded by the previously mentioned sums, for example  $\left| \frac{\partial \phi_{(k,s)}}{\partial \mathbf{x}_{(\tilde{k},\tilde{s})}}(\mathbf{x}, t) \right| \leq \sum_{\hat{s} \in \mathcal{S}} \left| \frac{\partial \phi_{(k,\hat{s})}}{\partial \mathbf{x}_{(\tilde{k},\tilde{s})}}(\mathbf{x}, t) \right|$ . Thus, the same bounds hold for  $\left| \frac{\partial \phi_{(k,s)}}{\partial \mathbf{x}_{(\tilde{k},\tilde{s})}}(\mathbf{x}, t) \right|$  and absolute values of higher order partial derivatives.

LEMMA D.1. *Given the solution  $\phi$  of the ODE defined in section 4.1. For the partial derivatives of  $\phi_{(k,s)}$  with respect to the initial condition  $\mathbf{x} \in \mathcal{X}$  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*

$$\begin{aligned}
 (a.1) \quad \sum_{s \in \mathcal{S}} \left| \frac{\partial \phi_{(k,s)}}{\partial x_i}(\mathbf{x}, t) \right| &= O(1/n), & (a.2) \quad \sum_{s \in \mathcal{S}} \left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\mathbf{x}, t) \right| &= O(1/n^2), \\
 (a.3) \quad \sum_{s \in \mathcal{S}} \left| \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, t) \right| &= O(1/n^3), & (a.4) \quad \sum_{s \in \mathcal{S}} \left| \frac{\partial^4 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\mathbf{x}, t) \right| &= O(1/n^4).
 \end{aligned}$$

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

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

$$(b.3) \quad \sum_{s \in \mathcal{S}} \left| \frac{\partial^3 \phi_{(k,s)}(\mathbf{x}, t)}{\partial x_i \partial x_j \partial x_w} \right| = O(1/n^2), \quad (b.4) \quad \sum_{s \in \mathcal{S}} \left| \frac{\partial^4 \phi_{(k,s)}(\mathbf{x}, t)}{\partial x_i \partial x_j \partial x_w \partial x_l} \right| = O(1/n^3).$$

PROOF. We will prove this lemma by bounding the derivative with respect to time of  $\frac{\phi_{(k,s)}}{\partial x_i}(\mathbf{x}, t)$  (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  $\phi(\mathbf{x}, t)$  satisfies the differential equation  $\frac{d}{dt} \phi(\mathbf{x}, t) = f(\phi(\mathbf{x}, t))$ . Hence, the partial derivatives of  $\frac{\phi_{(k,s)}}{\partial x_i}(\mathbf{x}, t)$ ,  $i \in \mathcal{I}$  with respect to the time  $t$  are

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

Having a closer look at the partial derivatives of  $f$  (see Appendix B.2), we see that

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

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

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

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.

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

Furthermore, for  $t$  equal to zero,  $\phi(\mathbf{x}, 0) = \mathbf{x}$  which implies that  $\frac{\partial \phi_{(k,s)}(\mathbf{x}, 0)}{\partial x_i} = 1$  if  $(k, s) = i$  and 0 otherwise. From this it follows directly that  $\max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^n \left| \frac{\partial \phi_{(k,s)}}{\partial x_i}(\mathbf{x}, 0) \right| = 1$ . Applying Grönwall's Lemma to the obtained results yields  $\max_{i \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^n \left| \frac{\partial \phi_{(k,s)}}{\partial x_i}(\mathbf{x}, t) \right| \leq \exp(2L_1 |\mathcal{S}| t) = O(1)$ .

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

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

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

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

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

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

$$\begin{aligned} \frac{d}{dt} \frac{\partial^2 \phi_{(k,s)}}{\partial x_j \partial x_i}(\mathbf{x}, t) &= \frac{\partial^2 f_{(k,s)}}{\partial x_j \partial x_i}(\phi(\mathbf{x}, t)) = \frac{\partial}{\partial x_j} \left( \sum_{u \in \mathcal{I}} \frac{\partial f_{(k,s)}}{\partial x_u}(\phi(\mathbf{x}, t)) \frac{\partial \phi_u}{\partial x_i}(\mathbf{x}, t) \right) \\ &= \sum_{u,v \in \mathcal{I}} \frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\phi(\mathbf{x}, t)) \frac{\partial \phi_u}{\partial x_i}(\mathbf{x}, t) \frac{\partial \phi_v}{\partial x_j}(\mathbf{x}, t) + \sum_{u \in \mathcal{I}} \frac{\partial f_{(k,s)}}{\partial x_u}(\phi(\mathbf{x}, t)) \frac{\partial^2 \phi_u}{\partial x_j \partial x_i}(\mathbf{x}, t). \end{aligned} \quad (21)$$

To bound the above term, we observe that

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

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

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

By previous observations  $\sum_{v \in \mathcal{I}} \left| \frac{\partial \phi_v}{\partial x_j}(\mathbf{x}, t) \right| = O(1)$  and  $\sum_{u \in \mathcal{I}_k} \left| \frac{\partial \phi_u}{\partial x_i}(\mathbf{x}, t) \right| = O(c_i^k)$  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

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

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

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

Recall that  $\boldsymbol{\phi}(\mathbf{x}, 0) = \mathbf{x}$  which implies that  $\frac{\partial^2 \phi^{(k,s)}(\mathbf{x}, 0)}{\partial x_i \partial x_j} = 0$ . Hence,  $\max_{i,j \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^n \left| \frac{\partial^2 \phi^{(k,s)}(\mathbf{x}, 0) \right| = 0$  which allows concluding, by applying Grönwall's Lemma, that  $\max_{i,j \in \mathcal{I}, s \in \mathcal{S}} \sum_{k=1}^n \left| \frac{\partial^2 \phi^{(k,s)}(\mathbf{x}, t) \right| = O(1/n)$ . For  $\sum_{s \in \mathcal{S}} \left| \frac{\partial^2 \phi^{(k,s)}(\mathbf{x}, t) \right|$  we infer

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

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

**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  $\frac{\partial^3 \phi^{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, t)$  and  $\frac{\partial^4 \phi^{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\mathbf{x}, t)$ , with  $i, j, w, l \in \mathcal{I}$ , are given by

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

and

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

The above quantity is equal to

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

For the third partial derivatives, we use the above equation to show first that  $\max_{i,j,w \in I, s \in S} \sum_{k=1}^n \left| \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, t) \right|$  is of order  $O(\frac{1}{n^2})$  and that  $\sum_{s \in S} \left| \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, t) \right|$  is of order  $O\left(\frac{1}{n^2}(c_i^k + c_w^k + c_j^k)\right)$ . To obtain a bound for the max term, we use the results obtained by the analysis of the first and second partial derivatives of  $\phi$ . The overall aim is to apply Grönwall's Lemma. We bound the first three sums of the derivative, which include first and second order partial derivatives of  $\phi$ . 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

$$\frac{\partial^3 f_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}) \leq \begin{cases} C_5/n^2 & \text{if } i, j \text{ or } w \in I_k \\ C_6/n^3 & \text{otherwise,} \end{cases}$$

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

For the first sum

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

and for the second type of sums

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

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  $\sum_{k=1}^n \sum_{u,v \in I} \left| \frac{\partial^2 f_{(k,s)}}{\partial x_u \partial x_v}(\phi(\mathbf{x}, t)) \right| \left| \frac{\partial^2 \phi_u}{\partial x_i \partial x_w}(\mathbf{x}, t) \right| \left| \frac{\partial \phi_v}{\partial x_j}(\mathbf{x}, t) \right| = O\left(\frac{1}{n^2}\right)$ .

The third sum of interest can be bounded by

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

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

$$\max_{i,j,w \in I, s \in S} \sum_{k=1}^n \left| \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, t) \right| = O\left(\frac{1}{n^2}\right).$$

Next, we show that the sum over the states  $\sum_{s \in \mathcal{S}} \left| \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, t) \right|$  is bounded by  $O\left(\frac{1}{n^2}(c_i^k + c_w^k + c_j^k)\right)$ . First, we recall that the first two sums of the corresponding derivative are bounded by  $O\left(\frac{1}{n^2}(c_i^k + c_w^k + c_j^k)\right)$ . Second,

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

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

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

Using  $\sum_{s \in \mathcal{S}} \left| \frac{\partial^3 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w}(\mathbf{x}, 0) \right| = 0$  and applying Grönwall's 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  $\max_{i,j,w,l \in \mathcal{I}} \sum_{k=1}^n \left| \frac{\partial^4 \phi_{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\mathbf{x}, t) \right|$  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

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

The bounds are

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

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

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

and

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

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

applying the max, this term is bounded by

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

Furthermore,  $\max_{i,j,w,l \in I, s \in S} \left\{ \sum_{k=1}^n \left| \frac{\partial^4 \phi_u}{\partial x_j \partial x_i \partial x_w \partial x_l}(\mathbf{x}, 0) \right| \right\}$  is zero. We see that  $\sum_{k=1}^n O\left(\frac{1}{n^3}(c_i^k + c_w^k + c_j^k + c_l^k)\right) = O\left(\frac{1}{n^3}\right)$  and, by applying Grönwall, it follows that  $\max_{i,j,w,l \in I, s \in S} \sum_{k=1}^n \left| \frac{\partial^4 \phi^{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\mathbf{x}, t) \right| = O(1/n^3)$ . At last, we show that  $\sum_{s \in S} \left| \frac{\partial^4 \phi^{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\mathbf{x}, t) \right|$  is bounded by  $O\left(\frac{1}{n^3}(c_i^k + c_w^k + c_j^k + c_l^k)\right)$ .

The proof follows the same principles as before for the third partial derivatives. The term  $\frac{d}{dt} \frac{\partial^3 \phi^{(k,s)}}{\partial x_i \partial x_j \partial x_w \partial x_l}(\mathbf{x}, t)$  can be separated into sums which are of order  $O\left(\frac{1}{n^3}(c_i^k + c_w^k + c_j^k + c_l^k)\right)$  and the additional term  $\sum_{s \in S} \sum_{u \in I} \left| \frac{\partial f^{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\mathbf{x}, t)) \right| \left| \frac{\partial^4 \phi_u}{\partial x_j \partial x_i \partial x_w \partial x_l}(\mathbf{x}, t) \right|$ . The latter is bounded by

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

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

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

□

## D.2 Bounds for Taylor Remainders

Lemma D.2 gives bounds for sums of weighted remainder terms appearing in the proofs of Theorem 4.1 and 4.2. 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 D.2. For  $\mathbf{x} \in \mathcal{X}$  and  $\tau \in \mathbb{R}_+$ , the remainder terms satisfy

$$\begin{aligned} \mathbb{E} \left[ \sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{x}, \mathbf{x}'} R_1(\mathbf{x}, \mathbf{x}', \tau) \right] & \leq \sup_{\mathbf{x}, \mathbf{y} \in \text{conv}(\mathcal{X})} \frac{1}{2} \sum_{i,j \in I} \left| \frac{\partial^2 \phi^{(k,s)}}{\partial x_i \partial x_j}(\mathbf{y}, \tau) \right| |Q_{i,j}(\mathbf{x})| = O(1/n), \\ \mathbb{E} \left[ \sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{x}, \mathbf{x}'} R_2(\mathbf{x}, \mathbf{x}', \tau) \right] & \leq \sup_{\mathbf{x}, \mathbf{y} \in \text{conv}(\mathcal{X})} \frac{1}{6} \sum_{i,j,u \in I} \left| \frac{\partial^3 \phi^{(k,s)}}{\partial x_i \partial x_j \partial x_u}(\mathbf{y}, \tau) \right| |R_{i,j,u}(\mathbf{x})| = O(1/n^2). \end{aligned}$$

Before starting the proof, recall that the first and second order remainder terms  $R_1$  and  $R_2$  defined in Section 6 are expressed as:

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

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

and that, as defined in Appendix B.2,  $\mathbf{Q}$  and  $\mathbf{R}$  are given by:

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

where  $(\mathbf{x}' - \mathbf{x})^{\otimes 2}$  and  $(\mathbf{x}' - \mathbf{x})^{\otimes 3}$  are Kronecker products of  $(\mathbf{x}' - \mathbf{x})$  with itself, i.e.,  $(\mathbf{x}' - \mathbf{x})_{i,j}^{\otimes 2} = (\mathbf{x}'_i - \mathbf{x}_i)(\mathbf{x}'_j - \mathbf{x}_j)$  and  $(\mathbf{x}' - \mathbf{x})_{i,j,u}^{\otimes 3} = (\mathbf{x}'_i - \mathbf{x}_i)(\mathbf{x}'_j - \mathbf{x}_j)(\mathbf{x}'_u - \mathbf{x}_u)$ . The two tensors  $\mathbf{Q}$  and  $\mathbf{R}$  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  $c_{k_1}^k$ , with  $k, k_1 \in \{1, \dots, n\}$ , to be one if  $k$  equals  $k_1$  and  $1/n$  otherwise. By the definition of  $\mathbf{Q}$  it follows that for  $\mathbf{x} \in \text{conv}(\mathcal{X})$ :

$$|Q_{(k,s), (k_1, s_1)}(\mathbf{x})| = O(c_{k_1}^k) = \begin{cases} O(1) & \text{if } k = k_1, \\ O(1/n) & \text{otherwise.} \end{cases}$$

This can be seen by writing the elements of  $\mathbf{Q}$  based on the two transition types (2a) and (2b), as shown in Section B.2.

From Lemma D.1 we know  $\left| \frac{\partial^2 \phi_{(k,s)}}{\partial x_{(k_1, s_1)} \partial x_{(k_2, s_2)}}(\mathbf{y}, \tau) \right| = O(\frac{1}{n}(c_{k_1}^k + c_{k_2}^k))$  which holds for any  $\mathbf{y} \in \text{conv}(\mathcal{X})$ . Here, the big  $O$  notation hides the dependence on  $\tau, L_1, L_2$  and  $|\mathcal{S}|$ . By noting that  $\sum_{k_1, k_2} c_{k_2}^{k_1} c_{k_1}^k = O(1)$ , we conclude

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

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  $\mathbf{R}$  as done in Section B.2 for  $\mathbf{Q}$ , the tensor  $\mathbf{R}$  is such that

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

Lemma D.1 states that the third partial derivatives of  $\phi$  are bounded by  $\left| \frac{\partial^3 \phi^{(k,s)}}{\partial x_{(k_1, s_1)} \partial x_{(k_2, s_2)} \partial x_{(k_3, s_3)}}(\mathbf{y}, \tau) \right| = O(\frac{1}{n^2}(c_{k_1}^k + c_{k_2}^k + c_{k_3}^k))$ . 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

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

□

### D.3 Connection of Differential and Integral Form for the Refinement Term

The following Lemma D.3 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 4.2 and Lemma D.4 which are related to the accuracy of the refined mean field approximation.

LEMMA D.3. *The solutions to the system of ODEs*

$$\begin{aligned} \frac{d}{dt} v_{(k,s)}(\mathbf{x}, t) &= \sum_{u \in I} \frac{\partial f_{(k,s)}}{\partial x_u}(\boldsymbol{\phi}(\mathbf{x}, t)) v_u(\mathbf{x}, t) + \frac{1}{2} \sum_{u,l \in I} \frac{\partial^2 f_{(k,s)}}{\partial x_l \partial x_u}(\boldsymbol{\phi}(\mathbf{x}, t)) w_{u,l}(\mathbf{x}, t), \\ \frac{d}{dt} w_{(k_1,s_1),(k_2,s_2)}(\mathbf{x}, t) &= \sum_{u \in I} w_{u,(k_2,s_2)}(\mathbf{x}, t) \frac{\partial f_{(k_1,s_1)}}{\partial x_u}(\boldsymbol{\phi}(\mathbf{x}, t)) + \sum_{u \in I} w_{u,(k_1,s_1)}(\mathbf{x}, t) \frac{\partial f_{(k_2,s_2)}}{\partial x_u}(\boldsymbol{\phi}(\mathbf{x}, t)) \\ &\quad + Q_{(k_1,s_1),(k_2,s_2)}(\boldsymbol{\phi}(\mathbf{x}, t)) \end{aligned}$$

can be expressed in integral form as

$$\begin{aligned} v_{(k,s)}(\mathbf{x}, t) &= \frac{1}{2} \int_0^t \sum_{i,j \in I} Q_{i,j}(\boldsymbol{\phi}(\mathbf{x}, \tau)) \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\boldsymbol{\phi}(\mathbf{x}, \tau), t - \tau) d\tau, \\ w_{(k_1,s_1),(k_2,s_2)}(\mathbf{x}, t) &= \int_0^t \sum_{i,j \in I} Q_{i,j}(\boldsymbol{\phi}(\mathbf{x}, \tau)) \frac{\partial \phi_{(k_1,s_1)}}{\partial x_i}(\boldsymbol{\phi}(\mathbf{x}, \tau), t - \tau) \frac{\partial \phi_{(k_2,s_2)}}{\partial x_j}(\boldsymbol{\phi}(\mathbf{x}, \tau), t - \tau) d\tau. \end{aligned}$$

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

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

where the last term is the same as the one derived in (21).

Combining these results and rearranging terms leads to

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

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

#### D.4 Comparison of the Refinement Term $\mathbf{v}$ and the Quadratic Taylor Term

In Lemma D.4 below, we bound the difference of the refinement term  $\mathbf{v}$  and the quadratic term of the second order Taylor expansion appearing in the proof of Theorem 4.2. By defining  $g_{(k,s)}(\mathbf{y}, \tau) = \sum_{i,j \in \mathcal{I}} Q_{i,j}(\mathbf{y}) \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\mathbf{y}, t - \tau)$  we see that the entries of the refinement term  $\mathbf{v}$  in integral form can be expressed as  $v_{(k,s)}(\mathbf{x}, t) = \frac{1}{2} \int_0^t g_{(k,s)}(\boldsymbol{\phi}(\mathbf{x}, \tau), \tau) d\tau$ . Similarly, the time integral over the expectation of the quadratic term of the Taylor expansion is given by  $\frac{1}{2} \int_0^t \mathbb{E}[g_{(k,s)}(\mathbf{X}(\tau), \tau)] d\tau$ . The latter arises due to the comparison of generator approach used in the proof of Theorem 4.2 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 D.2, to obtain the accuracy bounds for the refined mean field approximation.

LEMMA D.4. Define  $g_{(k,s)}(\mathbf{y}, \tau) = \sum_{i,j \in \mathcal{I}} Q_{i,j}(\mathbf{y}) \frac{\partial^2 \phi_{(k,s)}}{\partial x_i \partial x_j}(\mathbf{y}, t - \tau)$  with  $\boldsymbol{\phi}$  being the solution to the ODE defined in Section 4.1 and  $\mathbf{Q}$  as defined in Appendix B.2. Then

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

PROOF. We follow a similar proof concept as in Theorem 4.1. First, we define  $h_\tau(\mathbf{y}) = g_{(k,s)}(\mathbf{y}, \tau)$  and rewrite

$$\frac{1}{2} \int_0^t \mathbb{E}[g_{(k,s)}(\mathbf{X}(\tau), \tau) - g_{(k,s)}(\boldsymbol{\phi}(\mathbf{x}, \tau), \tau)] d\tau = \frac{1}{2} \int_0^t \mathbb{E}[h_\tau(\mathbf{X}(\tau)) - h_\tau(\boldsymbol{\phi}(\mathbf{x}, \tau))] d\tau. \quad (22)$$

By definition of  $g_{(k,s)}$ ,  $h_\tau$  is twice continuously differentiable. Second, using Lemma 6.1, we see that  $\mathbb{E}[h_\tau(\mathbf{X}(\tau)) - h_\tau(\boldsymbol{\phi}(\mathbf{x}, \tau))]$  is equal to

$$\int_0^\tau \mathbb{E} \left[ \sum_{\mathbf{x}' \in \mathcal{X}} K_{\mathbf{X}(v), \mathbf{x}'} (h_\tau(\boldsymbol{\phi}(\mathbf{x}', \tau - v)) - h_\tau(\boldsymbol{\phi}(\mathbf{X}(v), \tau - v))) - D_x(h_\tau \circ \boldsymbol{\phi})(\mathbf{X}(v), \tau - v) f(\mathbf{X}(v)) \right] dv. \quad (23)$$

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

$\nu)\Delta X(\nu)$  is equal to  $D_x(h_\tau \circ \phi)(\mathbf{X}(\nu), \tau - \nu)f(\mathbf{X}(\nu))$ , it follows that equation (23) is equal to the remainder of the Taylor expansion

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

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

$$\frac{1}{2} \int_0^\tau \sup_{\mathbf{y}, \mathbf{z} \in \text{conv}(X)} \sum_{i,j \in I} |Q_{i,j}(\mathbf{y})| \left| \frac{\partial^2}{\partial x_i \partial x_j} (h_\tau \circ \phi)(\mathbf{z}, \tau - \nu) \right| d\nu. \quad (24)$$

The rest of the proof is then essentially a careful analysis of the above sum. For that, we use again Lemma D.1 but also need bounds on up to the second derivative of  $\mathbf{Q}$  (This is needed because in the above expression the function  $h_\tau$  is defined as a function of  $\mathbf{Q}$ ). The latter makes the rest of the proof long and technical but the main ideas are essentially similar to the ones used in Lemma D.1.

The second derivative of  $h \circ \phi$  satisfies (for  $i, j \in I$ ):

$$\begin{aligned} \frac{\partial^2}{\partial x_i \partial x_j} (h_\tau \circ \phi)(\mathbf{z}, \tau - \nu) &= \frac{\partial}{\partial x_j} \left( \sum_{u \in I} \frac{\partial h_\tau}{\partial x_u} (\phi(\mathbf{z}, \tau - \nu)) \frac{\partial \phi_u}{\partial x_i} (\mathbf{z}, \tau - \nu) \right) \\ &= \sum_{u \in I} \frac{\partial h_\tau}{\partial x_u} (\phi(\mathbf{z}, \tau - \nu)) \frac{\partial^2 \phi_u}{\partial x_i \partial x_j} (\mathbf{z}, \tau - \nu) + \sum_{u,r \in I} \frac{\partial^2 h_\tau}{\partial x_u \partial x_r} (\phi(\mathbf{z}, \tau - \nu)) \frac{\partial \phi_u}{\partial x_i} (\mathbf{z}, \tau - \nu) \frac{\partial \phi_r}{\partial x_j} (\mathbf{z}, \tau - \nu). \end{aligned} \quad (25)$$

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

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

Similarly, the second partial derivative is

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

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 D.2. From the representation of  $\mathbf{Q}$  given in Appendix B.2, it can be seen that  $Q_{(k_1, s_1), (k_2, s_2)}(\mathbf{z}) = O(c_{k_2}^{k_1})$ ,  $\frac{\partial Q_{(k_1, s_1), (k_2, s_2)}}{\partial x^{(k_3, s_3)}} = O(c_{k_1, k_2, k_3})$  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 D.1 gives bounds for the partial derivatives of  $\phi$ . This enables us to develop an upper bound for  $|\frac{\partial h_\tau}{\partial x_u}(z)|$ ,

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

With the above observations we bound the first sum of (25) by

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

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

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

and that

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

and

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

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  $\left| \frac{\partial^2 h_\tau}{\partial x_{(k', s')} \partial x_{(\hat{k}, \hat{s})}}(\phi(\mathbf{z}, \tau - \nu)) \right| = O\left(\frac{1}{n^2}(c_{k'}^k + c_{\hat{k}}^k)\right)$ . This enables us to establish a

bound for the second sum of (25),

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

For the last part of the proof, we use the obtained results to bound  $\sum_{i,j \in \mathcal{I}} |Q_{i,j}(\mathbf{y})| \left| \frac{\partial^2}{\partial x_i \partial x_j} (h_\tau \circ \boldsymbol{\phi})(\mathbf{z}, \tau - \nu) \right|$ . By equation (25) we see that (24) is equal to

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

Indeed, we bound the supremum by the two following terms

$$\begin{aligned}
& \sum_{(k_1,s_1),(k_2,s_2) \in \mathcal{I}} |Q_{(k_1,s_1),(k_2,s_2)}(\mathbf{y})| \sum_{(k',s') \in \mathcal{I}} \left| \frac{\partial g_{(k,s)}}{\partial x_{(k',s')}}(\boldsymbol{\phi}(\mathbf{z}, \tau - \nu), \tau) \right| \left| \frac{\partial^2 \phi_{(k',s')}}{\partial x_{(k_1,s_1)} \partial x_{(k_2,s_2)}}(\mathbf{z}, \tau - \nu) \right| \\
&= \sum_{(k_1,s_1),(k_2,s_2)} O(c_{k_2}^{k_1}) O\left(\frac{1}{n^2}(c_{k_1}^k + c_{k_2}^k)\right) = O(1/n^2) \tag{26}
\end{aligned}$$

and

$$\begin{aligned}
& \sum_{(k_1,s_1),(k_2,s_2) \in \mathcal{I}} |Q_{(k_1,s_1),(k_2,s_2)}(\mathbf{y})| \sum_{(k',s'),(\hat{k},\hat{s}) \in \mathcal{I}} \left| \frac{\partial^2 g_{(k,s)}}{\partial x_{(k',s')} \partial x_{(\hat{k},\hat{s})}}(\boldsymbol{\phi}(\mathbf{z}, \tau - \nu), \tau) \right| \\
& \quad \times \left| \frac{\partial \phi_{(k',s')}}{\partial x_{(k_1,s_1)}}(\mathbf{z}, \tau - \nu) \right| \left| \frac{\partial \phi_{(\hat{k},\hat{s})}}{\partial x_{(k_2,s_2)}}(\mathbf{z}, \tau - \nu) \right| \\
&= \sum_{(k_1,s_1),(k_2,s_2)} O(c_{k_2}^{k_1}) O\left(\frac{1}{n^2}(c_{k_1}^k + c_{k_2}^k)\right) = O(1/n^2). \tag{27}
\end{aligned}$$

The bounds (26) and (27) show that (24) 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.  $\square$

Received October 2021; revised December 2021; accepted January 2022