

Learning operators on labelled conditional distributions with applications to mean field control of non exchangeable systems

Samy MEKKAOUI*

Huyên PHAM†

Xavier WARIN ‡

March 23, 2026

Abstract

We study the approximation of operators acting on probability measures on a product space with prescribed marginal. Let I be a label space endowed with a reference measure λ , and define \mathcal{M}_λ as the set of probability measures on $I \times \mathbb{R}^d$ with first marginal λ . By disintegration, elements of \mathcal{M}_λ correspond to families of labeled conditional distributions. Operators defined on this constrained measure space arise naturally in mean-field control problems with heterogeneous, non-exchangeable agents.

Our main theoretical result establishes a universal approximation theorem for continuous operators on \mathcal{M}_λ . The proof combines cylindrical approximations of probability measures with a DeepONet-type branch-trunk neural architecture, yielding finite-dimensional representations of such operators. We further introduce a sampling strategy for generating training measures in \mathcal{M}_λ , enabling practical learning of such conditional mean-field operators.

We apply the method to the numerical resolution of mean-field control problems with heterogeneous interactions, thereby extending previous neural approaches developed for homogeneous (exchangeable) systems. Numerical experiments illustrate the accuracy and computational effectiveness of the proposed framework.

MSC Classification: 49N80, 68T07, 91B69

Key words: Non exchangeable mean field systems; mean field neural networks; learning on Wasserstein space; DeepONet

1 Introduction

Motivation of MFC with non exchangeable systems. Mean-field control provides a tractable framework for modeling and optimizing large interacting particle systems. In the classical setting, exchangeability of agents allows the population to be described by a single marginal distribution, representing the law of a representative agent, and the associated control problems depend only on this evolving law. This structure underlies standard formulations of mean-field games and mean-field control, see [3] [7], [8].

*École Polytechnique, CMAP, samy.mekkaoui at polytechnique.edu. This author is supported by the S-G Chair “Financial Risks” and the Qube-RT Chair “Deep Finance and Statistics”, samy.mekkaoui at polytechnique.edu

†École Polytechnique, CMAP, huyen.pham at polytechnique.edu. This author is supported by the BNP Paribas Chair “Futures of Quantitative Finance”, the Chair “Financial Risks”, by FiME (Laboratory of Finance and Energy Markets), and the EDF-CACIB Chair “Finance and Sustainable Development”, huyen.pham at polytechnique.edu

‡EDF R&D, and FiME, xavier.warin at edf.fr

In many applications, however, agents are heterogeneous and cannot be treated as exchangeable. Agents may differ through labels, types, spatial indices, or intrinsic parameters, and heterogeneous interactions are often modeled through graphons or related structures; see e.g. [2], [4], [11], [20]. In such non-exchangeable systems, the population is no longer characterized by a single marginal distribution. Instead, it is described by a family of probability distributions indexed by a label space. Equivalently, the population law is a probability measure on a product space with prescribed first marginal, see [23]. This reformulation leads naturally to operators acting on constrained measure spaces rather than on a single probability distribution.

The efficient approximation of such operators is the central objective of this work. More precisely, heterogeneous mean-field control problems lead to nonlinear mappings that associate to a population law its corresponding value function, decoupling field, or feedback control. These mappings act on families of conditional distributions and therefore on probability measures on a product space with prescribed marginal. We formulate this approximation problem in a functional-analytic framework and develop a neural operator methodology adapted to this structure.

Operators on labelled conditional distributions. Let I be a compact label space, say $I = [0, 1]$, endowed with a reference probability measure, e.g., the uniform distribution. We consider the constrained space

$$\mathcal{M}_\lambda := \{ \mu \in \mathcal{P}_2(I \times \mathbb{R}^d) : \text{pr}_1 \# \mu = \lambda \},$$

where $\mathcal{P}_2(I \times \mathbb{R}^d)$ is the Wasserstein space of square integrable probability measures on $I \times \mathbb{R}^d$, $I \times \mathbb{R}^d \ni (u, x) \mapsto \text{pr}_1(u, x) = u$ denotes the projection mapping on the first coordinate and $\#$ denotes the pushforward measure. By disintegration, each $\mu \in \mathcal{M}_\lambda$ can be written as

$$\mu(\text{d}u, \text{d}x) = \lambda(\text{d}u) \mu^u(\text{d}x),$$

where $(\mu^u)_{u \in I}$ is a $\lambda(\text{d}u)$ -a.e unique family of labelled conditional distributions, valued in

$$L_\lambda^2(I; \mathcal{P}_2(\mathbb{R}^d)) := \{ I \ni u \mapsto \mu^u \in \mathcal{P}_2(\mathbb{R}^d) \text{ measurable and } \int_I \mathcal{W}_2(\mu^u, \delta_0) \lambda(\text{d}u) < \infty \},$$

where \mathcal{W}_2 denotes the 2-Wasserstein distance. We study operators of the form

$$\vartheta : \mathcal{M}_\lambda \ni \mu \mapsto V(\cdot, \cdot, \mu) \in L^2(\mu), \quad (1.1)$$

for some function V on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$ with quadratic growth w.r.t. the first two arguments, and $L^2(\mu)$ is the space of square integrable functions w.r.t. $\mu \in \mathcal{M}_\lambda$. Such operators arise naturally in heterogeneous mean-field control problems, for instance as decoupling fields associated with forward–backward stochastic systems in maximum principle, see [22], [5], or feedback maps obtained from dynamic programming, see [14].

Unlike classical neural operator learning, the domain here is an infinite-dimensional space of measures subject to a marginal constraint. Preserving this structural constraint is essential both for theoretical consistency and for numerical stability.

Our main contributions. The main contributions of this work are fourfold.

1. First, we introduce a neural operator framework tailored to operators defined on the constrained Wasserstein space \mathcal{M}_λ . The construction combines cylindrical approximations of probability measures with a DeepONet-type branch–trunk architecture, yielding finite-dimensional representations that are compatible with the marginal constraint $\text{pr}_1 \# \mu = \lambda$.

2. Second, we establish a universal approximation theorem for continuous operators ϑ as in (1.1). More precisely, we prove that the proposed architecture is dense in the class of such operators with respect to the natural topology induced by the Wasserstein distance. The proof integrates approximation results for probability measures with neural operator theory and shows that the marginal constraint can be preserved at the approximation level.
3. Third, we develop a constructive sampling procedure for generating training measures in \mathcal{M}_λ . The method enforces the prescribed marginal on the label space while producing diverse families of conditional laws, thereby enabling practical training of conditional mean-field neural operators in a manner consistent with the theoretical framework.
4. Finally, we apply the proposed methodology to the numerical resolution of mean-field control problems with non-exchangeable interactions. Relying on maximum principle and dynamic programming formulations developed for heterogeneous systems, we approximate decoupling fields, value functions, and feedback controls by conditional mean-field neural operators and use them to solve the associated forward-backward stochastic differential equations and HJB equations. This extends neural mean-field control methods previously developed for homogeneous (exchangeable) systems to the heterogeneous setting.

Related work. The present work lies at the intersection of neural operator learning, approximation on probability measure spaces, and mean-field control.

Neural operator architectures aim at approximating mappings between infinite-dimensional spaces and have been successfully applied to the numerical solution of partial differential equations. DeepONet [27], [24] provide universal approximation results for nonlinear operators between Banach spaces. Related operator-learning frameworks include Fourier neural operators [25] and subsequent developments in operator approximation theory. These approaches, however, are formulated for operators acting on function spaces. In contrast, we consider operators defined on the constrained Wasserstein space \mathcal{M}_λ , whose elements are probability measures on a product domain with prescribed marginal. Extending operator-learning techniques to this setting requires combining neural operator theory with measure-theoretic representations compatible with the marginal constraint.

Learning and approximation on spaces of probability measures have been studied through permutation-invariant architectures such as DeepSets [34], see [17]. Cylindrical approximations of measures and universal approximation results on Wasserstein spaces have been investigated in [12], [18] and [29]. These approaches typically address functions defined on unconstrained spaces of measures.

Numerical methods for mean-field control and mean-field games include PDE or probabilistic schemes and neural network approaches for solving forward–backward systems and HJB equations; see, e.g., [32], [9], [30], [31]. In the non-exchangeable settings, the associated optimality systems involve operators acting on conditional distributions rather than on a single marginal law. To the best of our knowledge, neural operator approximations for such operators on constrained measure spaces arising in heterogeneous mean-field control have not been investigated.

Outline of the paper. The remainder of the paper is organized as follows. In Section 2, we introduce the functional framework for operators defined on the constrained Wasserstein space \mathcal{M}_λ , present the conditional mean-field neural operator architecture, and establish the universal approximation theorem. We also describe the sampling procedure for generating training measures in \mathcal{M}_λ and the associated training methodology. Section 3 provides numerical experiments illustrating the approximation properties of the proposed neural operators for representative conditional mean-field functionals. In Section 4, we apply the method to the numerical resolution of mean-field control

problems with non-exchangeable interactions. We describe algorithms based on maximum principle and dynamic programming formulations and approximate the associated decoupling fields, value functions, and feedback controls using conditional mean-field neural operators.

Notations.

- We denote by $\mathcal{P}_2(\mathbb{R}^d)$ the Wasserstein space of square integrable probability measures equipped with the 2-Wasserstein distance \mathcal{W}_2 . Given $\mu \in \mathcal{P}_2(\mathbb{R}^d)$, we denote by $L^2(\mu)$ the space of measurable functions on \mathbb{R}^d s.t

$$|\phi|_\mu^2 := \int_{\mathbb{R}^d} |\phi(x)|^2 \mu(dx) < +\infty.$$

Given vector-valued maps $f, g \in L^2(\mu)$, we denote $\langle f, g \rangle_{L^2(\mu)} := \int_{\mathbb{R}^d} f(x) \cdot g(x) \mu(dx)$ as their inner product where \cdot is the inner product between vectors. Given $\mu \in \mathcal{P}_2(\mathbb{R}^d)$, and $\phi \in L^2(\mu)$, we set $\mathbb{E}_{X \sim \mu}[\phi(X)] := \int_{\mathbb{R}^d} \phi(x) \mu(dx)$, where X denotes a random variable defined on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We denote by \mathbb{P}_X the law of X under \mathbb{P} . Given a measurable map $\phi : \mathbb{R}^d \rightarrow \mathbb{R}^k$ and a measure $\mu \in \mathcal{P}_2(\mathbb{R}^d)$, we denote by $\langle \phi, \mu \rangle := \int_{\mathbb{R}^d} \phi(x) \mu(dx)$.

- Given an horizon time $T > 0$ and a normed vector space $(E, \|\cdot\|_E)$, we denote by $\mathcal{C}([t, T]; E)$ the space of continuous maps from $[t, T]$ into E endowed with Borel σ -algebra and its supremum norm $\|\omega\|_{\mathcal{C}([t, T]; E)} := \sup_{t \leq s \leq T} \|\omega_s\|_E$. When $E = \mathbb{R}^d$, we will only write $\mathcal{C}_{[t, T]}^d$ and when $d = 1$, only $\mathcal{C}_{[t, T]}$. We denote by \mathbb{W}_T the Wiener measure on $\mathcal{C}_{[0, T]}^d$. We also denote by $\mathcal{C}(E)$ the space of continuous functions on E into \mathbb{R} .

2 Operator learning of conditional mean field functionals

Given a function V on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$ valued in \mathbb{R}^p , with quadratic growth condition w.r.t the first two arguments, we aim to approximate the infinite dimensional map

$$\mathcal{V} : \mu \in \mathcal{M}_\lambda \mapsto V(\cdot, \cdot, \mu) \in L^2(\mu),$$

called non exchangeable mean field function, by a map \mathcal{N} constructed by some combinations of neural networks. The mean-field neural network \mathcal{N} takes input of two parts: μ a probability measure on \mathcal{M}_λ and (u, x) in the support of μ and outputs $\mathcal{N}(\mu)(u, x)$. The quality of this approximation is measured by the error

$$L(\mathcal{N}) := \int_{\mathcal{M}_\lambda} \mathcal{E}_{\mathcal{N}}(\mu) \rho(d\mu),$$

with

$$\mathcal{E}_{\mathcal{N}}(\mu) := |\mathcal{V}(\mu) - \mathcal{N}(\mu)|_\mu^2 = \mathbb{E}_{(U, X) \sim \mu} \left[|V(U, X, \mu) - \mathcal{N}(\mu)(U, X)|^2 \right],$$

and where ρ is a probability measure over the Borel space \mathcal{M}_λ , called training measure. The learning of the mean-field functional \mathcal{N} will then be performed by minimizing over the parameters of the neural network \mathcal{N} the loss function

$$L_M(\mathcal{N}) := \frac{1}{M} \sum_{m=1}^M \mathcal{E}_{\mathcal{N}}(\mu^{(m)}),$$

where $\mu^{(m)}$, $m \in \llbracket 1, M \rrbracket$ are training samples of ρ .

2.1 Neural network approximation

We state a universal approximation theorem which will rely on the use of DeepONet architecture, the cylindrical structure and their approximation results on Wasserstein space used to learn operators on \mathcal{M}_λ . The universal approximation theorem is stated with an L^2 -distance, which is the one used in practice during the training process.

Theorem 2.1 *Let ρ be a probability measure on the Borel space \mathcal{M}_λ and V a continuous map from $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$ into \mathbb{R}^q such that $\|V\|_{L^2(\rho)}^2 := \int_{\mathcal{M}_\lambda} |V(\cdot, \cdot, \mu)|_\mu^2 \rho(d\mu) < \infty$. Then, for all $\epsilon > 0$, there exists $J, r \in \mathbb{N}^*$, maps $\varphi_1, \dots, \varphi_J \in \mathcal{C}(I \times \mathbb{R}^d)$, trunk nets $(\mathcal{T}_k)_{1 \leq k \leq r}$ from $I \times \mathbb{R}^d$ into \mathbb{R} and branch nets $(\mathcal{B}_k)_{1 \leq k \leq r}$ from $\mathbb{R}^J \rightarrow \mathbb{R}^q$ such that*

$$\int_{\mathcal{M}_\lambda} \mathbb{E}_{(U, X) \sim \mu} \left[\left| V(U, X, \mu) - \sum_{k=1}^r \mathcal{T}_k(U, X) \mathcal{B}_k(\Phi_J(\mu)) \right|^2 \right] \rho(d\mu) \leq \epsilon,$$

where $\Phi_J(\mu) := (\langle \varphi_1, \mu \rangle, \dots, \langle \varphi_J, \mu \rangle) \in \mathbb{R}^J$.

Remark 2.2 *The architecture of Theorem 2.1, i.e.*

$$I \times \mathbb{R}^d \times \mathcal{M}_\lambda \ni (u, x, \mu) \mapsto \sum_{k=1}^r \mathcal{T}_k(u, x) \mathcal{B}_k(\Phi_J(\mu)),$$

is called *DeepONetCyl*. In addition to this architecture, we could also consider the more general class of neural network architectures,

$$I \times \mathbb{R}^d \times \mathcal{M}_\lambda \ni (u, x, \mu) \mapsto \Psi_\theta(u, x, \Phi_J(\mu)),$$

where $\Psi_\theta : I \times \mathbb{R}^d \times \mathbb{R}^J \rightarrow \mathbb{R}^q$. Since this architecture produces identical results, we only present those obtained with *DeepONetCyl* in the sequel.

2.2 Proof of the universal approximation theorem

Let $\epsilon > 0$. Fix \mathcal{K} a compact subset of $I \times \mathbb{R}^d$. For simplicity, we suppose V is a \mathbb{R} -valued map but the proof can be easily extended to a vector-valued function. Therefore, let V be a continuous function over $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$ into \mathbb{R} .

Step n°1 : Separability for a dense class of functions

Since \mathcal{K} is a compact metric space, the space $\mathcal{C}(\mathcal{K})$ is separable for the uniform norm and therefore, there exists a dense family $(\varphi_n)_{n \in \mathbb{N}} \in \mathcal{C}(\mathcal{K})$ for $\|\cdot\|_{\infty; \mathcal{K}} := \sup_{x \in \mathcal{K}} |\cdot|$. Then, the family of maps $\{\mathcal{M}_\lambda(\mathcal{K}) \ni \mu \mapsto \langle \varphi_n, \mu \rangle \in \mathbb{R} : n \in \mathbb{N}\}$ where $\mathcal{M}_\lambda(\mathcal{K}) := \{\mu \in \mathcal{P}(\mathcal{K}) : \text{pr}_1 \# \mu = \lambda\}$ is such that for any $\mu \neq \nu \in \mathcal{M}_\lambda(\mathcal{K})$, there exists $j \in \mathbb{N}$ s.t $\langle \varphi_j, \mu \rangle \neq \langle \varphi_j, \nu \rangle$.

Indeed, let $\mu, \nu \in \mathcal{M}_\lambda(\mathcal{K})$ such that $\mu \neq \nu$ and let $f \in \mathcal{C}(\mathcal{K})$ such that $\langle f, \mu \rangle \neq \langle f, \nu \rangle$ (such map f exists by characterization of Borel measures on the compact metric space \mathcal{K} by bounded continuous maps) and let $\delta := \left| \int_{\mathcal{K}} f(x) d\mu(x) - \int_{\mathcal{K}} f(x) d\nu(x) \right| > 0$ and $j_0 \in \mathbb{N}$ such that $\|\varphi_{j_0} - f\|_{\infty; \mathcal{K}} < \frac{\delta}{4}$. Therefore, we have

$$\begin{aligned} \left| \int_{\mathcal{K}} \varphi_{j_0} d\mu - \int_{\mathcal{K}} \varphi_{j_0} d\nu \right| &= \left| \int_{\mathcal{K}} f d\mu - \int_{\mathcal{K}} f d\nu + \int_{\mathcal{K}} (\varphi_{j_0} - f) d\mu - \int_{\mathcal{K}} (\varphi_{j_0} - f) d\nu \right| \\ &\geq \left| \int_{\mathcal{K}} f d\mu - \int_{\mathcal{K}} f d\nu \right| - \left| \int_{\mathcal{K}} (\varphi_{j_0} - f) d\mu - \int_{\mathcal{K}} (\varphi_{j_0} - f) d\nu \right| \\ &\geq \delta - 2\|\varphi_{j_0} - f\|_{\infty; \mathcal{K}} > 0 \end{aligned}$$

For $J \in \mathbb{N}^*$, we define

$$\mathcal{M}_\lambda(\mathcal{K}) \ni \mu \mapsto \Phi_J(\mu) := (\langle \varphi_1, \mu \rangle, \dots, \langle \varphi_J, \mu \rangle) \in \mathbb{R}^J.$$

By definition of the weak convergence of probability measures and since \mathcal{K} is compact, Φ_J is clearly continuous in the topology of the Wasserstein distance.

Step n°2 : Construction of a dense sub-algebra of $\mathcal{C}(\mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K}))$

We now denote the class of maps \mathcal{A} on $\mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})$ into \mathbb{R} as

$$\mathcal{A} := \left\{ F : \mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K}) \rightarrow \mathbb{R} \mid \exists J, r \in \mathbb{N}^*, (f_k)_{1 \leq k \leq r} \subset \mathcal{C}(\mathbb{R}^J), (g_k)_{1 \leq k \leq r} \subset \mathcal{C}(\mathcal{K}) \right. \\ \left. \text{such that } F(u, x, \mu) = \sum_{k=1}^r f_k(\Phi_J(\mu)) g_k(u, x) \right\}.$$

We claim that given the supremum norm over the compact space $\mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})$, the closure $\bar{\mathcal{A}} = \mathcal{C}(\mathcal{K} \times \mathcal{P}_2^\lambda(\mathcal{K}))$.

Following Stone-Weierstrass, it is sufficient to show that \mathcal{A} is a sub-algebra of $\mathcal{C}(\mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K}))$ which contains constant maps and where for any $(k_1, \mu_1) \neq (k_2, \mu_2)$ where $k_1, k_2 \in \mathcal{K}$ and $\mu_1, \mu_2 \in \mathcal{M}_\lambda(\mathcal{K})$, there exists $F \in \mathcal{A}$ such that $F(k_1, \mu_1) \neq F(k_2, \mu_2)$.

- The stability by addition and multiplication are clear.
- It is clear that \mathcal{A} contains constant maps by taking $J, r = 1, f_1 = 1 \in \mathcal{C}(\mathbb{R})$ and $g_1 = 1 \in \mathcal{C}(\mathcal{K})$.
- Let $k_1 \neq k_2$. In this case, take $J = 1, r = 1, f_1 = 1$ and $g_1(k) = d(k, k_1)$ for which it is clear that $F(k_1, \mu_1) = g_1(k_1) = 0 \neq g_1(k_2) = F(k_2, \mu_2)$ where $d((u_1, x_1), (u_2, x_2)) = |u_1 - u_2| + |x_1 - x_2|$ is the product distance. Suppose now $k_1 = k_2$ and $\mu_1 \neq \mu_2$. By the previous result, there exists $j_0 \in \mathbb{N}^*$ such that $\langle \varphi_{j_0}, \mu_1 \rangle \neq \langle \varphi_{j_0}, \mu_2 \rangle$. Then, we take $J = j_0, r = 1, g = 1$ and $f_1 : (x_1, \dots, x_{j_0}) \mapsto x_{j_0} \in \mathcal{C}(\mathbb{R}^{j_0})$. In this case, we have $F(k_1, \mu_1) = \langle \varphi_{j_0}, \mu_1 \rangle \neq \langle \varphi_{j_0}, \mu_2 \rangle = F(k_2, \mu_2)$.

Therefore by Stone-Weierstrass, \mathcal{A} is a dense sub-algebra of $\mathcal{C}(\mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K}))$. Therefore, for any $\epsilon > 0$, there exists $J, r \in \mathbb{N}^*, (f_k)_{1 \leq k \leq r} \in \mathcal{C}(\mathbb{R}^J)$ and $(g_k)_{1 \leq k \leq r} \in \mathcal{C}(\mathcal{K})$ such that

$$\sup_{(u, x, \mu) \in \mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})} \left| V(u, x, \mu) - \sum_{k=1}^r f_k(\Phi_J(\mu)) g_k(u, x) \right| \leq \epsilon.$$

Now, noticing that $\tilde{\mathcal{K}} = \Phi_J(\mathcal{M}_\lambda(\mathcal{K}))$ is compact as the image of a compact by a continuous map, and by the classical universal approximation theorem for finite dimensional functions, for any $k \in \llbracket 1, r \rrbracket$, there exists a feedforward neural network $\mathcal{B}_k : \mathbb{R}^J \rightarrow \mathbb{R}$ and $\mathcal{T}_k : I \times \mathbb{R}^d \rightarrow \mathbb{R}$ such that for given $\delta > 0$

$$\begin{cases} \max_{1 \leq k \leq r} \sup_{z \in \tilde{\mathcal{K}}} |f_k(z) - \mathcal{B}_k(z)| \leq \delta, \\ \max_{1 \leq k \leq r} \sup_{z \in \mathcal{K}} |g_k(z) - \mathcal{T}_k(z)| \leq \delta, \end{cases}$$

Denoting now the cylindrical DeepONetCyl map on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda(\mathcal{K})$ as $\text{DeepONetCyl}(\mu)(u, x) := \sum_{k=1}^r \mathcal{B}_k(\Phi_J(\mu)) \mathcal{T}_k(u, x)$. We now show that

$$\sup_{(u, x, \mu) \in \mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})} |V(u, x, \mu) - \text{DeepONetCyl}(u, x, \mu)| \leq \epsilon.$$

Denote the positive and finite constants M_f and M_g as

$$\begin{cases} M_f := \max_{1 \leq k \leq r} \sup_{z \in \tilde{\mathcal{K}}} |f_k(z)|, \\ M_g := \max_{1 \leq k \leq r} \sup_{z \in \mathcal{K}} |g_k(z)|, \end{cases}$$

Then $\sup_{z \in \tilde{\mathcal{K}}} |\mathcal{B}_k(z)| \leq M_f + \delta$ and $\sup_{z \in \mathcal{K}} |\mathcal{T}_k(z)| \leq M_g + \delta$.

Let $(k, \mu) \in \mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})$. Then, we have

$$\begin{aligned} \|V - \text{DeepONet}\|_{\infty; \mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})} &= \left\| \sum_{k=1}^r f_k(\Phi_J(\cdot)) g_k - \sum_{k=1}^r \mathcal{B}_k(\Phi_J(\cdot)) \mathcal{T}_k \right\|_{\infty; \mathcal{K} \times \mathcal{M}_\lambda(\mathcal{K})} \\ &\leq \sum_{k=1}^r \left(\|f_k\|_{\infty; \tilde{\mathcal{K}}} \|g_k - \mathcal{T}_k\|_{\infty; \mathcal{K}} + \|\mathcal{T}_k\|_{\infty; \mathcal{K}} \|f_k - \mathcal{B}_k\|_{\infty; \tilde{\mathcal{K}}} \right) \\ &\leq r(M_f \delta + (M_g + \delta)\delta) \end{aligned}$$

Choosing δ small enough is enough to get the required result.

Step n°3 : Approximation theorem on $L^2(\rho)$

Let $\epsilon > 0$ and ν a probability measure on \mathcal{M}_λ . Given $M \geq 0$ we truncate the map V by defining V_M on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$ as

$$V_M(u, x, \mu) := \begin{cases} V(u, x, \mu) & \text{if } |V(u, x, \mu)| \leq M, \\ M \frac{V(u, x, \mu)}{|V(u, x, \mu)|} & \text{if } |V(u, x, \mu)| > M, \end{cases}$$

so that $|V_M(u, x, \mu)| \leq M$ for every $(u, x, \mu) \in I \times \mathbb{R}^d \times \mathcal{M}_\lambda$. By definition of V_M , we have the point-wise convergence: $V_M \xrightarrow{M \rightarrow \infty} V$. Therefore, for a fixed $\mu \in \mathcal{M}_\lambda$ by looking over the space $L^2(\mu)$ and noticing that $|V_M(\cdot, \cdot, \mu) - V(\cdot, \cdot, \mu)| \leq |V(\cdot, \cdot, \mu)| \in L^2(\mu)$. Therefore, it implies by the dominated convergence theorem that the map $\mu \mapsto |V_M(\cdot, \cdot, \mu) - V(\cdot, \cdot, \mu)|_\mu$ converges point-wisely to 0 and since $|V_M(\cdot, \cdot, \mu) - V(\cdot, \cdot, \mu)|_\mu \leq |V(\cdot, \cdot, \mu)|_\mu \in L^2(\rho)$ by assumption on V , we conclude by the convergence dominated theorem that $\|V - V_M\|_{L^2(\rho)}^2 \rightarrow 0$. We can therefore choose $M > \frac{\sqrt{\epsilon}}{4}$ such that

$$\|V - V_M\|_{L^2(\rho)}^2 \leq \frac{\epsilon}{8}. \quad (2.1)$$

Now, we consider some compact set $\mathcal{K} \subset I \times \mathbb{R}^d$ such that $\rho(\mathcal{M}_\lambda \setminus \mathcal{M}_\lambda(\mathcal{K})) \leq \frac{\epsilon}{80M^2}$ following Lusin's theorem on complete and separable metric spaces (see Theorem C.1 in [24]) and we note that V_M is continuous on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$. Applying the universal approximation theorem from **Step n°2**, we get $r, J \in \mathbb{N}$, $(\varphi_k)_{1 \leq k \leq r} \in \mathcal{C}(\mathbb{R}^J)$ and $(\psi_k)_{1 \leq k \leq r} \in \mathcal{C}(I \times \mathbb{R}^d)$ such that

$$\sup_{(u, x) \in \mathcal{K}, \mu \in \mathcal{M}_\lambda(\mathcal{K})} |V_M(u, x, \mu) - \sum_{k=1}^r \varphi_k(\Phi_J(\mu)) \psi_k(u, x)| \leq \frac{\sqrt{\epsilon}}{4}.$$

Therefore, we have

$$\begin{aligned} \left| \sum_{k=1}^r \varphi_k(\Phi_J(\mu)) \psi_k(u, x) \right| &\leq |V_M(u, x, \mu)| + \left| V_M(u, x, \mu) - \sum_{k=1}^r \varphi_k(\Phi_J(\mu)) \psi_k(u, x) \right|, \\ &\leq M + \frac{\sqrt{\epsilon}}{4} \leq 2M, \quad (u, x) \in \mathcal{K}, \mu \in \mathcal{M}_\lambda(\mathcal{K}). \end{aligned} \quad (2.2)$$

We note that after suitably modifying the linear output layers of the branch and trunk nets neural nets of the previous DeepONetCyl (see Theorem 3.1 in [24]), we can rewrite

$$\sum_{k=1}^r \varphi_k(\Phi_J(\mu)) \psi_k(u, x) = \sum_{k=1}^l \mathcal{B}_k(\Phi_J(\mu)) \mathcal{T}_k(u, x),$$

where the trunk neural nets $(\mathcal{T}_k)_{1 \leq k \leq l}$ are orthonormal in $L^2(\mu)$ for some $l \leq r$. In particular, we have

$$\left| \sum_{k=1}^l \mathcal{B}_k(\Phi_J(\mu)) \mathcal{T}_k(\cdot, \cdot) \right|_{\mu}^2 = |\mathcal{B}(\Phi_J(\mu))|^2, \quad \mu \in \mathcal{M}_{\lambda}, \quad (2.3)$$

and

$$|\mathcal{B}(\Phi_J(\mu))| \leq 2M, \quad \forall \mu \in \mathcal{M}_{\lambda},$$

where we denoted $\mathcal{B}(\mu) := (\mathcal{B}_1(\mu), \dots, \mathcal{B}_l(\mu))$. Now, by the clipping lemma (see Lemma C.2 in [24]), there exists a neural network $\gamma : \mathbb{R}^l \rightarrow \mathbb{R}^l$, satisfying

$$\begin{cases} |\gamma(y) - y| \leq \frac{\sqrt{\epsilon}}{4}, & \text{if } |y| \leq M + \frac{\sqrt{\epsilon}}{4}, \\ |\gamma(y)| \leq 2M, & \forall y \in \mathbb{R}^l. \end{cases} \quad (2.4)$$

We now define the following DeepONetCyl over $I \times \mathbb{R}^d \times \mathcal{M}_{\lambda}$ as

$$\text{DeepONetCyl}(\mu)(u, x) := \sum_{k=1}^l \gamma(\mathcal{B}_k(\Phi_J(\mu))) \mathcal{T}_k(u, x),$$

and we have

$$|\text{DeepONetCyl}(\mu)(\cdot, \cdot)|_{\mu} = |\gamma(\mathcal{B}(\Phi_J(\mu)))| \leq 2M,$$

by (2.4) and since $(\mathcal{T}_k)_{1 \leq k \leq l}$ are orthonormal in $L^2(\mu)$. Moreover, we have

$$\begin{aligned} |V_M(\cdot, \cdot, \mu) - \text{DeepONetCyl}(\mu)(\cdot, \cdot)|_{\mu} &\leq |V_M(\cdot, \cdot, \mu) - \sum_{k=1}^l \mathcal{B}_k(\Phi_J(\mu)) \mathcal{T}_k(\cdot, \cdot)|_{\mu} \\ &\quad + \left| \sum_{k=1}^l \mathcal{B}_k(\Phi_J(\mu)) \mathcal{T}_k(\cdot, \cdot) - \text{DeepONetCyl}(\mu)(\cdot, \cdot) \right|_{\mu} \\ &\leq \frac{\sqrt{\epsilon}}{4} + \frac{\sqrt{\epsilon}}{4} = \frac{\sqrt{\epsilon}}{2}. \end{aligned}$$

from (2.2), (2.3) and (2.4). We therefore have

$$\begin{aligned} C &:= \int_{\mathcal{M}_{\lambda}} |V_M(\cdot, \cdot, \mu) - \text{DeepONetCyl}(\mu)(\cdot, \cdot)|_{\mu}^2 \rho(d\mu) \\ &\leq \int_{\mathcal{M}_{\lambda}(\mathcal{K})} |V_M(\cdot, \cdot, \mu) - \text{DeepONetCyl}(\mu)(\cdot, \cdot)|_{\mu}^2 \rho(d\mu) \\ &\quad + 2 \int_{\mathcal{M}_{\lambda} \setminus \mathcal{M}_{\lambda}(\mathcal{K})} (|V_M(\cdot, \cdot, \mu)|_{\mu}^2 + |\text{DeepONetCyl}(\mu)(\cdot, \cdot)|_{\mu}^2) \rho(d\mu) \\ &\leq \frac{\epsilon}{4} + 2(M^2 + 4M^2) \frac{\epsilon}{80M^2} = \frac{3\epsilon}{8}, \end{aligned}$$

where we used $|V_M| \leq M$ and $|\text{DeepOnetCyl}(\mu)(\cdot, \cdot)|_\mu \leq 2M$. Now, recalling (2.1), we have

$$\begin{aligned} D &:= \int_{\mathcal{M}_\lambda} |V(\cdot, \cdot, \mu) - \text{DeepOnetCyl}(\cdot, \cdot, \mu)|^2 \rho(d\mu) \\ &\leq 2 \int_{\mathcal{M}_\lambda} |V(\cdot, \cdot, \mu) - V_M(\cdot, \cdot, \mu)|_\mu^2 \rho(d\mu) \\ &\quad + 2 \int_{\mathcal{M}_\lambda} |V_M(\cdot, \cdot, \mu) - \text{DeepOnetCyl}(\cdot, \cdot, \mu)|_\mu^2 \rho(d\mu) \leq \frac{2\epsilon}{8} + 2\frac{3\epsilon}{8} = \epsilon. \end{aligned}$$

The proof is therefore completed.

2.3 Data generation

The training of neural networks for approximating mean field functions relies on sampling $\mu \in \mathcal{M}_\lambda$ and a random variable (U, X) whose law is μ . We first give a Lemma which provides a simple way to sample on \mathcal{M}_λ .

Lemma 2.3 (Sampling in \mathcal{M}_λ).

Let ν be a non-atomic reference probability in $\mathcal{P}_2(\mathbb{R}^d)$ and $(\mu^u)_u \in L^2(I; \mathcal{P}_2(\mathbb{R}^d))$. Then, there exists a measurable map $T \in L^2(\lambda \otimes \nu; \mathbb{R}^d)$ such that $T(u, \cdot) \# \nu = \mu^u \lambda(du)$ -a.e.

Proof. The proof is a simple application of the randomization lemma (see [21]). Indeed, since the map $I \ni u \mapsto \mu^u \in \mathcal{P}_2(\mathbb{R}^d)$ is measurable, there exists a measurable map $F : I \times [0, 1] \rightarrow \mathbb{R}^d$ such that $(U, F(U, R)) \sim \mu(du, dx) = \mu^u(dx)\lambda(du)$ where $(U, R) \sim \mathcal{U}([0, 1]) \otimes \mathcal{U}([0, 1])$ where R is used for randomization and U for the labeling of the agents. Disintegrating μ over I , we get that $F(u, \cdot) \# \mathcal{U}([0, 1]) = \mu^u \lambda(du)$ -a.e.

Let $\Phi : \mathbb{R}^d \rightarrow [0, 1]$ be a measurable isomorphism satisfying $\Phi \# \nu = \mathcal{U}([0, 1])$ (which exists since ν is non-atomic on a Borel space). Defining the measurable map T on $I \times \mathbb{R}^d$ into \mathbb{R}^d as $T(u, x) := F(u, \Phi(x))$, we end up with the result. We now verify that $T \in L^2(\lambda \otimes \nu; \mathbb{R}^d)$, i.e.

$$\int_{I \times \mathbb{R}^d} |T(u, x)|^2 \nu(dx) \lambda(du) < +\infty. \quad (2.5)$$

Now, (2.5) follows noticing that for $\lambda(du)$ -a.e., $\int_{\mathbb{R}^d} |T(u, x)|^2 \nu(dx) = \int_{\mathbb{R}^d} |y|^2 \mu^u(dy)$ since $T(u, \cdot) \# \nu = \mu^u$ and because $(\mu^u)_u \in L^2(I; \mathcal{P}_2(\mathbb{R}^d))$. \square

From Lemma 2.3, we design two ways to sample on \mathcal{M}_λ .

S1: Given a non-atomic measure $\nu \in \mathcal{P}_2(\mathbb{R}^d)$, we sample a dense class of maps $(T_\theta)_{\theta \in \Theta} \in L^2(\lambda \otimes \nu; \mathbb{R}^d)$ (for instance $(T_\theta)_{\theta \in \Theta}$ can be sampled as a parametrized class of neural networks for a set $\Theta \subset \mathbb{R}^m$. In this case, $\mathbb{P}_{(U, T_\theta(U, Y))} \in \mathcal{M}_\lambda$, where U and Y are random variables on $(\Omega, \mathcal{F}, \mathbb{P})$ s.t $(U, Y) \sim \lambda \otimes \nu$.

S2: Alternatively, we can sample a class of non-atomic measures $\nu \in \mathcal{S} \subset \mathcal{P}_2(\mathbb{R}^d)$, given a fixed map $T \in L^2(\mu \otimes \nu)$. In this case, we start by sampling $(U, Y) \sim \lambda \otimes \nu$ where $\nu \in \mathcal{S}$ and we set $(U, T(U, Y))$ whose law belongs to \mathcal{M}_λ by construction. However, this method only helps us to sample in the set $\{\lambda(du)T(u, \cdot) \# \nu(dy) : \nu \in \mathcal{S}\}$.

Remark 2.4 (Special case of $d = 1$).

In the special case of $d = 1$, the map T can be explicitly constructed. Indeed, we take $\nu = \mathcal{U}([0, 1])$. Then, given $(\mu^u)_u \in \mathcal{P}_2(\mathbb{R})$ we denote its cumulative distribution function as $F_{\mu^u}(x) := \mu^u((-\infty; x])$ and we define its quantile function Q_{μ^u} as

$$Q_{\mu^u}(t) := \inf \{x \in \mathbb{R} : F_{\mu^u}(x) \geq t\}, \quad t \in (0, 1).$$

Then, defining $T(u, x) := Q_{\mu^u}(x)$ gives the required result.

2.4 Training the mean-field operator

The algorithm consists in the following parts.

1. In order to use the DeepONetCyl structure and the universal approximation of Theorem 2.1, we need to consider a finite but dense functions in order to compute the quantities $\Phi_J(\mu)$. In practice, we samples polynomial maps and we choose r and J as hyperparameters. In fact, for $1 \leq i \leq J$, we choose the moment maps $\varphi_i(u, x) := |x|^i + u^i$ and we notice that $\langle \varphi_i, \mu \rangle = \mathbb{E}_{(U, X) \sim \mu} [|X|^i]$ up to a constant for $\mu \in \mathcal{M}_\lambda$.
2. We draw samples $(\mu^{(m)})_{1 \leq m \leq M}$ of probability measures on \mathcal{M}_λ (recalling the previous section). We introduce two neural networks \mathcal{T}^{θ_1} parametrized with θ_1 and \mathcal{B}^{θ_2} parametrized by θ_2 . We then minimize over the parameters $\theta := (\theta_1, \theta_2)$ the following loss function

$$L(\theta) := \frac{1}{M} \sum_{m=1}^M \mathbb{E}_{(U, X) \sim \mu^{(m)}} \left[\left| V(U, X, \mu^{(m)}) - \sum_{k=1}^r \mathcal{T}_k^{\theta_1}(U, X) \mathcal{B}_k^{\theta_2}(\Phi_J(\mu^{(m)})) \right|^2 \right].$$

It can be summarized in the following algorithm :

Algorithm 1: Non exchangeable mean field network approximation of a map

Input: Number of distributions M , batch size N , number of epoch e , learning rate ρ , $\theta = (\theta_1, \theta_2)$ initial parameters of neural networks $\mathcal{T}^{\theta_1} = (\mathcal{T}_1^{\theta_1}, \dots, \mathcal{T}_r^{\theta_1})$ and $\mathcal{B}^{\theta_2} = (\mathcal{B}_1^{\theta_2}, \dots, \mathcal{B}_r^{\theta_2})$, J number of moments, r number of sensors.

for each epoch e **do**

Sample $(U_{m,n}, X_{m,n}) \sim \mu^{(m)}$ for $n \in \llbracket 1, N \rrbracket$ where $\mu^{(m)} \in \mathcal{M}_\lambda$ for $m \in \llbracket 1, M \rrbracket$

Calculate $\Phi_m = (\frac{1}{N} \sum_{n=1}^N |X_{m,n}|^j)_{j \in \llbracket 1, J \rrbracket}$ for $m \in \llbracket 1, M \rrbracket$

Calculate target function

$$J(\theta) = \frac{1}{MN} \sum_{n=1}^M \sum_{j=1}^N \left(V(U_{m,n}, X_{m,n}, \mu^{(m)}) - \sum_{k=1}^r \mathcal{T}_k^{\theta_1}(U_{m,n}, X_{m,n}) \mathcal{B}_k^{\theta_2}(\Phi_m) \right)^2$$

$\theta = \theta - \rho \nabla J(\theta)$

end

Return: θ

3 Numerical experiments

We test our algorithms by computing the mean-squared error (MSE) for different cases of non exchangeable mean-field functions V on $I \times \mathbb{R} \times \mathcal{M}_\lambda$. As in the literature, we use a graphon map G (i.e. a measurable map on $I \times I$ into \mathbb{R}_+) to represent the interaction between two agents. We propose the two following maps to illustrate our Algorithm 1.

V1: A first order non exchangeable mean-field interaction

$$V(u, x, \mu) := x - \mathbb{E}_{(U, X) \sim \mu} [G(u, U)X].$$

V2: A second order non exchangeable mean-field interaction

$$V(u, x, \mu) := \mathbb{E}_{(U, X) \sim \mu} [(x - G(u, U)X)^2].$$

We first generate samples in \mathcal{M}_λ by using **S2** in Section 2.3, i.e., we fix a transport map T and we sample a class of non-atomic measures. We first propose two simple cases for choice of map T given the unknown link between X and $U \sim \mathcal{U}([0, 1])$.

T1: $T(u, y) := uy$,

T2: $T(u, y) := uy + (uy)^2$.

We then draw $X = T(U, Y)$, with Y sampled according to a distribution picked randomly as a mixture of five gaussian laws. In fact, for each m , we sample for $k \in \llbracket 1, 5 \rrbracket$, $W_{m,k} \sim \mathcal{U}([0, 1])$, $(\mu_{m,k}, \sigma_{m,k}) \sim \mathcal{U}([0, 1])^2$, and then we compute for any $n \in \llbracket 1, N \rrbracket$

$$Y_{m,n} = \frac{1}{\sum_{k=1}^5 W_{m,k}} \sum_{k=1}^5 W_{m,k} Z_{m,k,n}, \quad \text{with } Z_{m,k,n} \sim \mathcal{N}(\mu_{m,k}, \sigma_{m,k}^2).$$

We consider two cases of interacting functions:

1. A smooth graphon

$$G_1(u, v) := e^{-uv}.$$

2. A block-wise graphon

$$G_2(u, v) := L \sum_{i=1}^L \frac{1}{1 + e^{((i-1)L(u-v))}} 1_{(\frac{i-1}{L}, \frac{i}{L}]}(u) 1_{(\frac{i-1}{L}, \frac{i}{L}]}(v), \quad (3.1)$$

as in [1], modeling $L = 5$ teams, each team with a given interaction between agents (in the first team agents have an homogeneous interaction) and the different teams have no interaction. The function is represented on Figure 1.

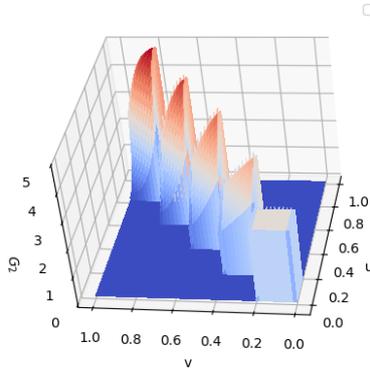


Figure 1: G_2 graphon.

All the tests are achieved using the ADAM optimization method with a learning rate of 0.001. The default value for r is 10 and we consider two architectures for the networks \mathcal{T}^{θ_1} and \mathcal{B}^{θ_2} :

- The first one is the classical feedforward network using a tanh activation function using 3 hidden layers of 10 neurons.
- The second one is a Kolmogorov Arnold Network [26] using two hidden layers of 10 neurons and a grid size equal to 5. In this version, one dimensional functions are approximated using splines (Spline KAN). Notice that no universal approximation theorem is available for this network.

- The third one is a Kolmogorov Arnold Network using P1 type finite element functions with adapting support to approximate one dimensional functions (P1KAN) [33]. This network is specially effective to approximate irregular functions. Its convergence is supported by a universal approximation theorem. With this network, we use two hidden layers of 10 neurons and a grid size equal to 10.

In the algorithm we use $N = 50000$ samples to approximate the distribution and take one distribution at each iteration of the gradient algorithm.

We now illustrate the convergence of the non exchangeable network using a GPU Nvidia H100 94Go HBM2. All convergence plots are given with 100000 iterations calculating the accuracy every hundred of iterations and smoothing the result obtained with a rolling window of 10 values.

On Figures 2, 3 we give the convergence depending on the number of moments. Not surprisingly with mean-field function in $\mathbf{V1}$, the convergence rate is independent of the number of moment, while with mean-field function in $\mathbf{V2}$ at least two moments are necessary.

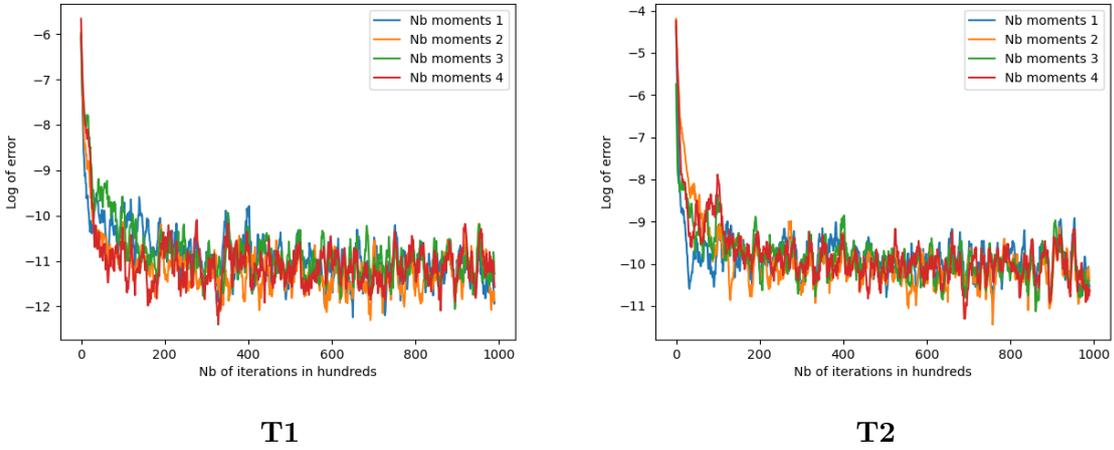


Figure 2: Mean-field function in $\mathbf{V1}$, graphon G_1 using the feedforward network : log of the error obtained during training depending on J the number of moments.

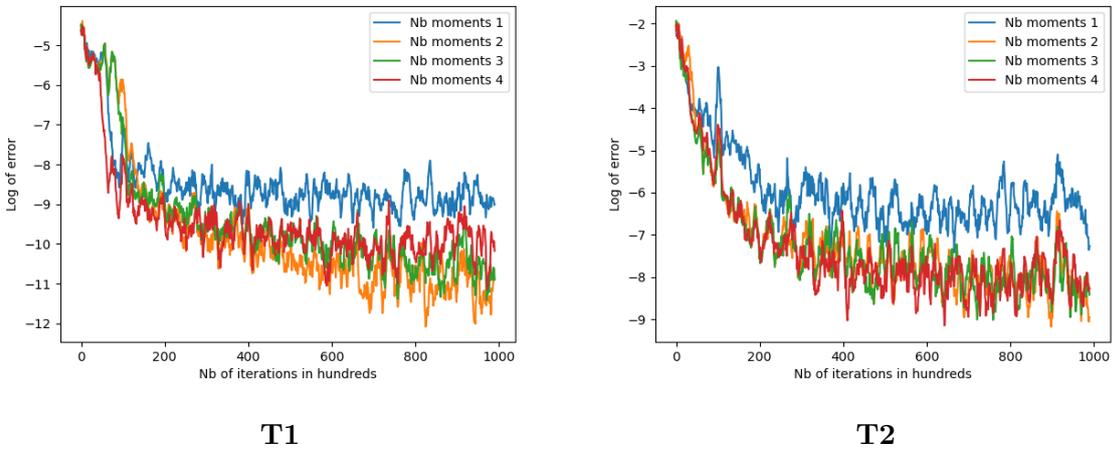


Figure 3: Mean-field function in $\mathbf{V2}$, graphon G_1 using the feedforward network : log of the error obtained during training on J the number of moments.

Figure 4 illustrates the fact that choice of the network architecture is crucial to get a very good

convergence. For these very regular cases, Spline KAN outperforms the two other networks that behave similarly. KAN networks are known to be more expensive to use than feedforwards but here computing times are very similar. For mean-field function in **V2**, with transport map from **T2**, with one moment, 100 iterations takes 3.56 seconds with a feedforward while it takes 3.74 seconds with the Spline KAN. Using 4 moments, the feedforward network takes 3.61 seconds while the Spline KAN takes 3.75 seconds. Results are similar with the P1KAN network. The crucial point here is to effectively calculate the interaction matrix.

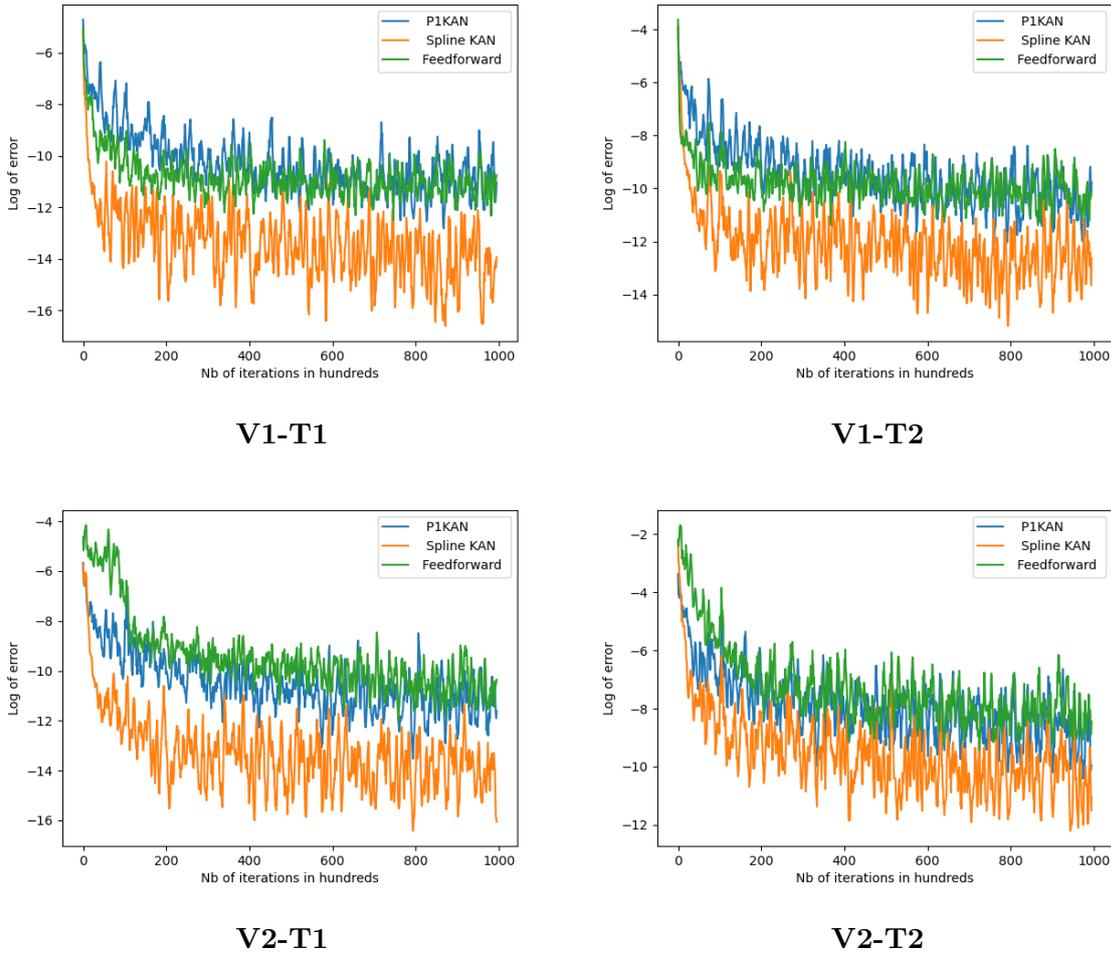
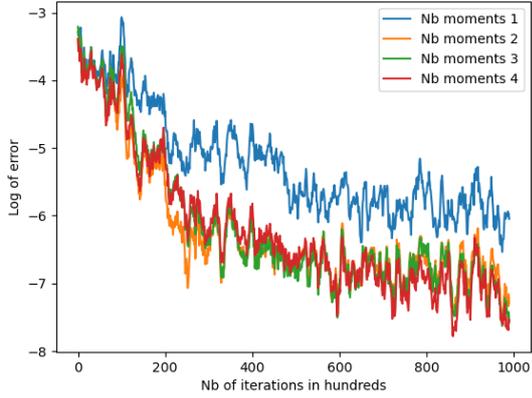
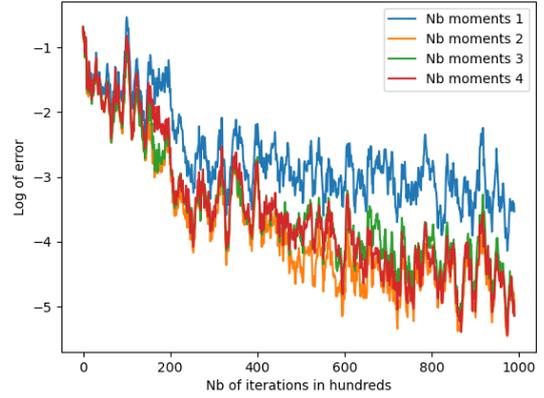


Figure 4: Comparison of feedforward and KAN with G_1 interaction function using 3 moments

On Figure 5, we plot the convergence using the G_2 function which is far more irregular for the two cases of **T2**. The convergence is much more difficult to achieve and much more erratic.



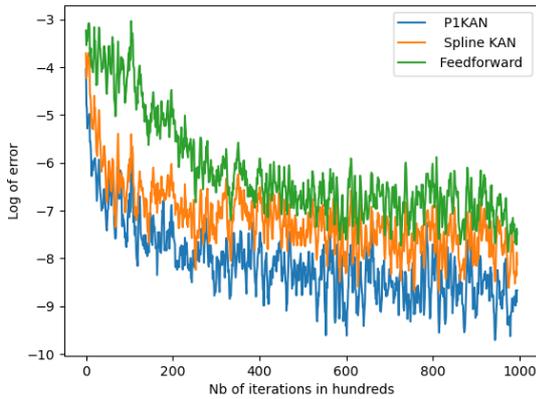
T1



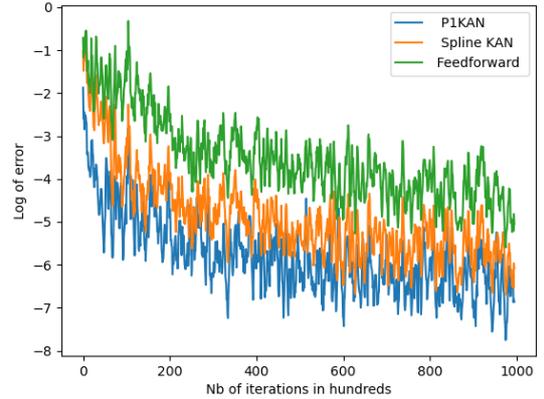
T2

Figure 5: Mean-field function from $\mathbf{V2}$, graphon G_2 interaction using the feedforward network : log of the error obtained during training on J the number of moments

On Figure 6, we show that the KAN networks converge better and faster than the feedforward. As the functions to approximate are rather irregular, the P1KAN network outperforms the two other ones.



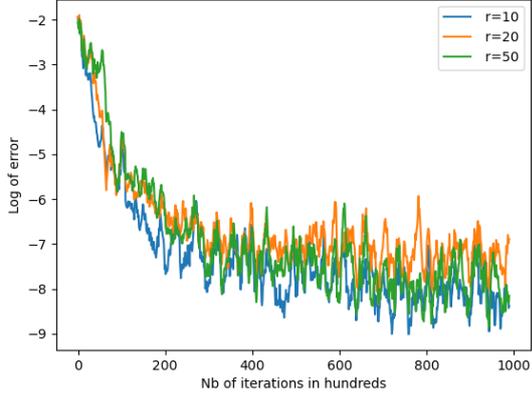
T1



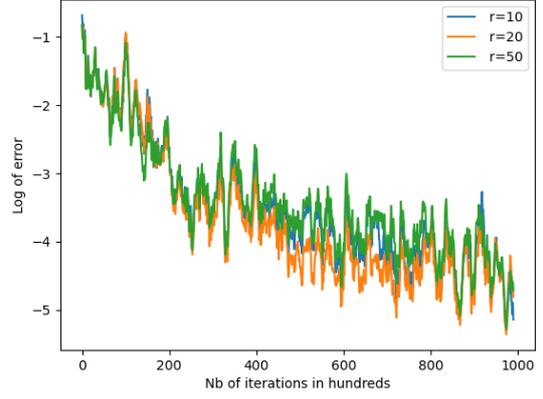
T2

Figure 6: Comparison of feedforward and KAN with G_2 interaction function using 3 moments for mean-field function $\mathbf{V2}$.

On Figure 7, we show that the number of sensors r used is not critical.



G_1 interaction function.



G_2 interaction function.

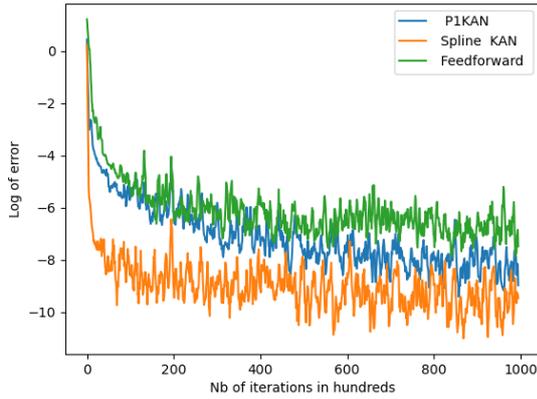
Figure 7: Impact of the number of sensors r on the convergence for mean-field function $\mathbf{V2}$ and transport map $\mathbf{T2}$ with 3 moments.

Finally, we adopt the sampling method in $\mathbf{S1}$. We suppose now that $Y \sim \mathcal{N}(0, 1)$, and we consider the family of random transport maps $\omega \mapsto T(\omega, \cdot, \cdot) \in L^2(\lambda \otimes \mathcal{N}(0, 1))$:

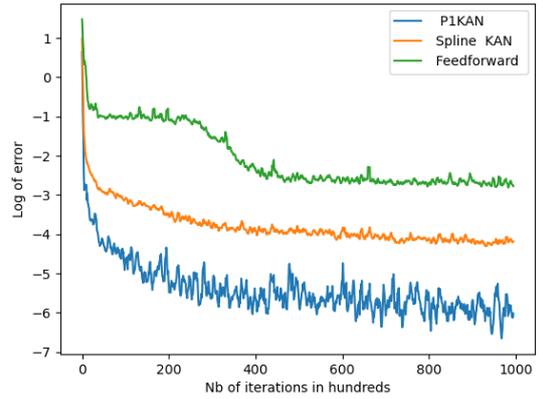
$$T(\omega, u, y) := A(\omega)uy + B(\omega)(uy + u^2y^2),$$

where $(A, B) \sim \mathcal{U}([0, 1])^{\otimes 2}$ and then we sample $X = T(U, Y)$ with (U, Y) independent of (A, B) .

On Figure 8, we show with $J = 3$ moments that the convergence is achieved especially when a KAN network is used. We observe that the convergence curve is smoother than in the sampling method $\mathbf{S2}$. Again with the G_2 interaction function, the P1KAN network outperform the two other networks.



Graphon G_1



Graphon G_2

Figure 8: Comparison of feedforward and KAN using 3 moments for mean-field function $\mathbf{V2}$ with the sampling method $\mathbf{S1}$.

4 Application to optimal control of non exchangeable mean field systems

In this section, we propose an application of the theory developed above to solve optimal control problem for non exchangeable mean field systems. Concisely, it can be formulated as the extension of the usual standard McKean-Vlasov control problem without assuming homogeneity between the agents. This naturally leads to an infinite dimensional control problem since the agents are no longer homogeneous. We present below some standard methods to solve optimal controls involving this class of controlled systems.

4.1 Background on controlled non exchangeable mean field systems

We first introduce the framework of optimal control on non exchangeable mean field systems in a label-state formulation. On a complete filtered probability space $(\Omega, \mathcal{F}, \mathbb{P})$ satisfying the usual hypothesis, we are given the following random variables

- A uniform random variable U over $[0, 1]$ used for encoding the heterogeneity.
- A \mathbb{R}^n -valued Brownian motion $W := (W_t)_{0 \leq t \leq T}$ independent of U .
- A \mathbb{R}^d -valued initial condition ξ with law m independent of W .
- We denote by $\mathbb{F}^W = (\mathcal{F}_t^W)_{0 \leq t \leq T}$ the natural filtration generated by W and by $\mathbb{F} = (\mathcal{F}_t)_{0 \leq t \leq T}$ the filtration given by $\mathcal{F}_t := \mathcal{F}_t^W \vee \sigma(U)$ where σ denotes the σ -algebra generated by U , augmented with the \mathbb{P} -null sets.

We fix drift and diffusion functions $b, \sigma : I \times \mathbb{R}^d \times \mathcal{M}_\lambda \times A \rightarrow \mathbb{R}^d, \mathbb{R}^{d \times n}$ on which we make the following standard assumptions

Assumption 4.1 *The functions $b, \sigma : I \times \mathbb{R}^d \times \mathcal{M}_\lambda \times A \rightarrow \mathbb{R}^d, \mathbb{R}^{d \times n}$ are Borel measurable. Moreover, there exists positive constants $L \geq 0, M \geq 0$ such that*

$$\begin{cases} |b(u, x, \mu, a) - b(u, x', \mu', a)| & \leq L (|x - x'| + \mathcal{W}_2(\mu, \mu')) \\ |\sigma(u, x, \mu, a) - \sigma(u, x', \mu', a)| & \leq L (|x - x'| + \mathcal{W}_2(\mu, \mu')) \end{cases}$$

and

$$|b(u, 0, \lambda \otimes \delta_0, a)| + |\sigma(u, 0, \lambda \otimes \delta_0, a)| \leq M(1 + |a|),$$

for every $u \in I, x, x' \in \mathbb{R}^d, \mu, \mu' \in \mathcal{M}_\lambda$ and $a \in A$.

We denote by \mathcal{A} the set of admissible controls that are \mathbb{F} -progressively measurable process valued in a convex measurable space (A, \mathcal{A}) such that $\mathbb{E} \left[\int_0^T |\alpha_t|^2 dt \right] < \infty$, and by \mathcal{I}_t the set of admissible initial conditions defined as $\mathcal{I}_t = \{ \xi : \xi \text{ is } \mathcal{F}_t\text{-measurable and } \mathbb{E}[|\xi|^2] < \infty \}$.

Given $\alpha \in \mathcal{A}$ and $\xi \in \mathcal{I}_t$, we consider the following controlled state process $X = (X_s^{\xi, \alpha})_{t \leq s \leq T}$ satisfying the following SDE

$$\begin{cases} dX_s & = b(U, X_s, \mathbb{P}_{(U, X_s)}, \alpha_s) ds + \sigma(U, X_s, \mathbb{P}_{(U, X_s)}, \alpha_s) dW_s, \\ X_t & = \xi, \end{cases} \quad (4.1)$$

Theorem 4.2 *Given $\xi \in \mathcal{I}_t$ and $\alpha \in \mathcal{A}$ and under Assumption 4.1, there exists a unique strong solution $X = (X_s)_{t \leq s \leq T}$ to Equation (4.1).*

Proof. The proof is postponed to Appendix B.1. □

From standard estimations, there exists a positive constant C such that

$$\mathbb{E} \left[\sup_{t \leq s \leq T} |X_s|^2 \right] \leq C \left(1 + \mathbb{E} [|\xi|^2] + \mathbb{E} \left[\int_t^T |\alpha_s|^2 ds \right] \right).$$

We next introduce the two reward functions $f : I \times \mathbb{R}^d \times \mathcal{M}_\lambda \times A \rightarrow \mathbb{R}$ and $g : I \times \mathbb{R}^d \times \mathcal{M}_\lambda \rightarrow \mathbb{R}$ on which we make the following assumptions.

Assumption 4.3 *The functions f and g are Borel measurable and there exists a constant $M \geq 0$ such that*

$$\begin{cases} -M(1 + |x|^2 + \mathcal{W}_2(\mu, \lambda \otimes \delta_0)^2) \leq f(u, x, \mu, a) \leq M(1 + |x|^2 + \mathcal{W}_2(\mu, \lambda \otimes \delta_0)^2 + |a|^2) \\ |g(u, x, \mu)| \leq M(1 + |x|^2 + \mathcal{W}_2(\mu, \lambda \otimes \delta_0)^2), \end{cases}$$

for every $u \in I$, $x \in \mathbb{R}^d$, $\mu \in \mathcal{M}_\lambda$ and $a \in A$.

We define the cost functional as the \mathbb{R} -valued map J as follows

$$J(t, \xi, \alpha) := \mathbb{E} \left[\int_t^T f(U, X_s, \mathbb{P}_{(U, X_s)}, \alpha_s) dt + g(U, X_T, \mathbb{P}_{(U, X_T)}) \right], \quad (4.2)$$

where we stressed the dependence of J in the initial condition ξ . Under Assumptions 4.1, 4.3 and from Theorem 4.2, we get that $J(\alpha)$ is well defined for any $\alpha \in \mathcal{A}$. The goal is now to study the optimal control problem consisting in minimizing the function J over \mathcal{A} , that is, computing

$$V_0 := \inf_{\alpha \in \mathcal{A}} J(0, \xi, \alpha), \quad (4.3)$$

and to find an optimal control $\alpha^* \in \mathcal{A}$, i.e. s.t $V_0 = J(\alpha^*)$. More generally, at any time $t \in [0, T]$, we can define the cost functional to be minimized as

$$V(t, \xi) = \inf_{\alpha \in \mathcal{A}} J(t, \xi, \alpha), \quad t \in [0, T]. \quad (4.4)$$

In the sequel, we will analyze two known methods namely, the Pontryagin's maximum principle and the Bellman equation to solve (4.3). The Pontryagin's maximum principle will lead to the study of a fully coupled forward backward stochastic differential equation (FBSDE) from which we will be able to characterize the optimality of a control α^* whereas the dynamic programming equation will help us to characterize the Bellman function V through a partial differential equation (PDE) and from a backward recursion which enables the possibility to design several efficient algorithms (see [30]).

Remark 4.4 The current formulation (i.e. (4.1)-(4.2)) can be viewed as a label-state formulation of the non exchangeable mean field system studied in [22, 5] for the stochastic maximum principle formulation and in [14] for the derivation of the dynamic programming equation and the Bellman equation. The main issue in these formulations is the necessity to deal with an uncountable continuum of controlled state processes $(X^u)_{u \in I}$ for which joint measurability over space $I \times \Omega$ is not guaranteed due to a collection of i.i.d. Brownian motions $\{W^u : u \in I\}$. Moreover, this formulation is not well suited for numerical experiments as one would need to discretize the number of processes $(X^{i,N})_{1 \leq i \leq N}$ to consider. Since we are working at the level of the marginal laws of the processes, we work under the label-state formulation which lacks of a pathwise interpretation but for which the equality of the laws is preserved (see [28]). In the sequel, we will define the necessary quantities to characterize an optimal control either in the stochastic maximum principle or in the Bellman equation but since the proofs are really similar to the ones presented in [22, 14], we will just give the main ideas of the proof and refer to the proofs therein for further details.

4.1.1 Stochastic maximum principle and FBSDE equation

For simplicity, we will look at $t = 0$ but the analysis could be extended to any time t . For the stochastic maximum principle, we define the real-valued Hamiltonian map H as

$$H(u, x, \mu, a, y, z) = b(u, x, \mu, a) \cdot y + \sigma(u, x, \mu, a) : z + f(u, x, \mu, a), \quad (4.5)$$

where \cdot refers to the inner product between two matrices, for any $(u, x, \mu, a, y, z) \in I \times \mathbb{R}^d \times \mathcal{M}_\lambda \times A \times \mathbb{R}^d \times \mathbb{R}^{d \times n}$. This map will be used to derive necessary and sufficient conditions for the optimality of an admissible control. It will rely on the notions of derivative and convexity over the space \mathcal{M}_λ introduced in Appendix A. In fact, under some regularity assumptions that will be precised on the map H and defining the process $\hat{\alpha} = (\hat{\alpha}_t)_{0 \leq t \leq T}$ as

$$\hat{\alpha}_t = \hat{\alpha}(U, X_t, \mathbb{P}_{(U, X_t)}, Y_t, Z_t) = \inf_{a \in A} H(U, X_t, \mathbb{P}_{(U, X_t)}, Y_t, Z_t, a),$$

where $(X_t, Y_t, Z_t)_{0 \leq t \leq T}$ is the solution to the following system of FBSDE

$$\begin{cases} dX_t &= b(U, X_t, \mathbb{P}_{(U, X_t)}, \hat{\alpha}_t)dt + \sigma(U, X_t, \mathbb{P}_{(U, X_t)}, \hat{\alpha}_t)dW_t, \\ dY_t &= -\partial_x H(U, X_t, \mathbb{P}_{(U, X_t)}, P_t, Z_t, \hat{\alpha}_t)dt - \tilde{\mathbb{E}} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} H(\tilde{U}, \tilde{X}_t, \mathbb{P}_{(U, X_t)}, \tilde{Y}_t, \tilde{Z}_t, \tilde{\alpha}_t)(U, X_t) \right] dt \\ &+ Z_t dW_t \\ Y_T &= \partial_x g(U, X_T, \mathbb{P}_{(U, X_T)}) + \tilde{\mathbb{E}} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} g(\tilde{U}, \tilde{X}_T, \mathbb{P}_{(U, X_T)})(U, X_T) \right], \end{cases} \quad (4.6)$$

where $(\tilde{U}, \tilde{X}_t, \tilde{Y}_t, \tilde{Z}_t, \tilde{\alpha}_t)$ is an independant copy $(U, X_t, Y_t, Z_t, \alpha_t)$ defined on another probability space $(\tilde{\Omega}, \tilde{\mathcal{F}}, \tilde{\mathbb{P}})$, we will show that $\hat{\alpha}$ yields an optimal control, i.e. a solution to (4.3).

We now make the necessary assumptions on the regularity on the maps involved in the control problem to derive the stochastic maximum principle.

Assumption 4.5

- (1) The maps (b, σ) are differentiable with respect to (x, a) . Moreover, the maps $\partial_x(b, \sigma)$ are assumed to be uniformly bounded. Finally, the maps $(x, \mu, a) \mapsto \partial_x(b, \sigma, f)(u, x, \mu, a)$ and $(x, \mu, a) \mapsto \partial_a(b, \sigma, f)$ are continuous for λ -a.e $u \in I$.
- (2) The maps (b, σ) are assumed to have Fréchet differentiable linear functional derivatives $\partial_{\tilde{x}} \frac{\delta}{\delta m} b$ and $\partial_{\tilde{x}} \frac{\delta}{\delta m} \sigma$ satisfying the following properties

$$\begin{cases} \left| \partial_{\tilde{x}} \frac{\delta}{\delta m} b(u, x, \mu, a)(\tilde{u}, \tilde{x}) - \partial_{\tilde{x}} \frac{\delta}{\delta m} b(u, x', \mu', a')(\tilde{u}', \tilde{x}') \right| \leq L(|x - x'| + |\tilde{x} - \tilde{x}'| + \mathcal{W}_2(\mu, \mu')), \\ \left| \partial_{\tilde{x}} \frac{\delta}{\delta m} \sigma(u, x, \mu, a)(\tilde{u}, \tilde{x}) - \partial_{\tilde{x}} \frac{\delta}{\delta m} \sigma(u, x', \mu', a')(\tilde{u}', \tilde{x}') \right| \leq L(|x - x'| + |\tilde{x} - \tilde{x}'| + \mathcal{W}_2(\mu, \mu')), \end{cases}$$

for every $u, \tilde{u} \in I$, $x, x', \tilde{x}, \tilde{x}' \in \mathbb{R}^d$ and $\mu, \mu' \in \mathcal{M}_\lambda$ and

$$\left| \partial_{\tilde{x}} \frac{\delta}{\delta m} b(u, 0, \lambda \otimes \delta_0, a) \right| + \left| \partial_{\tilde{x}} \frac{\delta}{\delta m} \sigma(u, 0, \lambda \otimes \delta_0, a) \right| \leq M(1 + |a|),$$

for every $u, \tilde{u} \in I$ and $a \in A$.

- (3) The maps f and g are differentiable with respect to (x, a) . Moreover, $\partial_x(f, g)$ and $\partial_a f$ are assumed to be uniformly bounded $\lambda(d\mu)$ -a.e. Finally, the maps $(x, \mu, a) \mapsto \partial_x f(u, x, \mu, a)$, $\partial_a f(u, x, \mu, a)$ and $(x, \mu) \mapsto \partial_x g(u, x, \mu)$ are continuous for λ -a.e $u \in I$.

(4) The functions f and g admit Fréchet differentiable linear functional derivatives. Moreover, for any progressively measurable process X such that $\mathbb{E}\left[\sup_{0 \leq t \leq T} |X_t|^2\right] < \infty$, the following quantities are uniformly bounded $\lambda(\mathrm{d}u) - \text{a.e}$

$$\tilde{\mathbb{E}}\left[|\partial_{\tilde{x}} \frac{\delta}{\delta m} f(u, x, \mu, a)(\tilde{U}, \tilde{X})|^2\right], \text{ and } \tilde{\mathbb{E}}\left[|\partial_{\tilde{x}} \frac{\delta}{\delta m} g(u, x, \mu)(\tilde{U}, \tilde{X})|^2\right].$$

Proposition 4.6 *Let Assumptions 4.1, 4.3, 4.5 hold and assume furthermore that the Hamiltonian map H defined in (4.5) is a convex function in its last variable, i.e. the map*

$$A \ni a \mapsto H(u, x, \mu, y, z, a),$$

is convex for any $(u, x, \mu, y, z) \in I \times \mathbb{R}^d \times \mathcal{M}_\lambda \times \mathbb{R}^d \times \mathbb{R}^{d \times n}$. Let $(\alpha_t)_{0 \leq t \leq T}$ be an optimal control and $(X_t, Y_t, Z_t)_{0 \leq t \leq T}$ the respectively associated controlled state processes and adjoint processes defined in (4.6). Then the optimal control problem is a point-wise minimizer of H , i.e. for any $a \in A$.

$$H(U, X_t, \mathbb{P}_{(U, X_t)}, Y_t, Z_t, \alpha_t) \leq H(U, X_t, \mathbb{P}_{(U, X_t)}, Y_t, Z_t, a) \quad \mathrm{d}t \otimes \mathrm{d}\mathbb{P} - \text{a.e.} \quad (4.7)$$

Proof. As the proof is fairly similar to the one presented in [22, 5], we present only the key steps in Appendix B.2. \square

Remark 4.7 Under the following stronger convexity assumptions, we can turn the necessary condition into a sufficient condition, namely we need to assume that

- (1) The map $\mathbb{R}^d \times \mathcal{M}_\lambda \ni (x, \mu) \mapsto g(U, x, \mu)$ is convex $\mathbb{P} - \text{a.s}$
- (2) The map $\mathbb{R}^d \times \mathcal{M}_\lambda \times A \ni (x, \mu, a) \mapsto H(U, x, \mu, Y_t, Z_t, a)$ is convex $\mathrm{d}t \otimes \mathrm{d}\mathbb{P} - \text{a.e.}$

With this additional requirements and if α^* satisfies (4.7), then α^* yields an optimal control.

At this point, we are able to identify a potential optimal control α in the form of a progressively measurable map of (U, X, Y, Z) as a minimizer of the Hamiltonian map. However, plugging the potential optimal control obtained from (4.7) in the forward and adjoint equations leads to the study of a fully coupled FBSDE, namely (4.6), for which we need to prove existence and unicity. Motivated by the numerical applications, we will concentrate ourselves to the subclass of linear dynamics and quadratic cost functional for which we will prove existence and uniqueness in the following section. In the existing literature, existence and uniqueness of the resulting FBSDE system can be proven under linear dynamics but for a slightly broader class of cost functional. We refer to [22, 6] for full details.

4.1.2 Dynamic programming and HJB equation.

Solution to (4.4) can also be characterized by the dynamic programming method. We will restrict ourselves to give the main results that will be used to design algorithms. As we show for the case of the stochastic maximum principle, the proof relies on similar arguments and we refer to [14] for a rigorous analysis of the HJB equation (note that their HJB is stated on the space $L^2(I; \mathcal{P}_2(\mathbb{R}^d))$) but the analysis can be easily extended to our setting to the space \mathcal{M}_λ . This approach is based on a recursive argument after defining the decoupled value function $V : I \times [0, T] \times \mathbb{R}^d \times \mathcal{M}_\lambda$ of problem which satisfies

$$V(t, U, X_t, \mathbb{P}_{(U, X_t)}) = \inf_{\alpha \in \mathcal{A}} \mathbb{E} \left[\int_t^{t+h} f(U, X_s, \mathbb{P}_{(U, X_s)}, \alpha_s) \mathrm{d}s + V(t+h, X_{t+h}, \mathbb{P}_{(U, X_{t+h})}) | \mathcal{F}_t \right], (4.8)$$

for any $t \in [0, T)$ and $h \in (0, T - t]$. and starting from the terminal condition $V(T, u, x, \mu) = g(u, x, \mu)$ for $(u, x, \mu) \in I \times \mathbb{R}^d \times \mathcal{M}_\lambda$. Following the result in [14] which can be adapted to our current setting (in view of the Pontryagin's Maximum principle) and assuming that for any $(u, t, x, \mu, p, M) \in I \times [0, T] \times \mathbb{R}^d \times \mathcal{M}_\lambda \times \mathbb{R}^d \times \mathbb{R}^{d \times d}$, there exists a minimizer

$$\hat{a}(u, x, \mu, p, M) \in \arg \min_{a \in A} H(u, x, \mu, p, M, a) \quad (4.9)$$

where the Hamiltonian map H is defined as

$$H(u, x, \mu, p, M, a) := b(u, x, \mu, a) \cdot p + \frac{1}{2} \sigma \sigma^\top(u, x, \mu, a) : M + f(u, x, \mu, a),$$

one can show by sending formally $h \rightarrow 0$ in (4.8) that

$$\begin{cases} \partial_t V(t, u, x, \mu) + b(u, x, \mu, \hat{a}(u, x, \mathcal{U}(u, t, x, \mu), \partial_x \mathcal{U}(u, t, x, \mu))) \cdot \partial_x V(u, t, x, \mu) \\ + \frac{1}{2} \sigma \sigma^\top(u, x, \mu, \mathcal{U}(u, t, x, \mu), \partial_x \mathcal{U}(t, u, x, \mu)) : \partial_x^2 V(t, u, x, \mu) \\ + \mathbb{E}_{(U, \xi) \sim \mu} \left[b(U, \xi, \mu, \hat{a}(U, \xi, \mathcal{U}(t, U, \xi, \mu), \partial_x \mathcal{U}(t, U, \xi, \mu))) \cdot \partial_{\tilde{x}} \frac{\delta}{\delta m} V(t, u, x, \mu)(U, \xi) \right. \\ \left. + \frac{1}{2} \sigma \sigma^\top(t, U, \xi, \mu, \hat{a}(U, \xi, \mathcal{U}(t, U, \xi, \mu), \partial_x \mathcal{U}(t, U, \xi, \mu))) : \partial_{\tilde{x}}^2 \frac{\delta}{\delta m} V(t, u, x, \mu)(U, \xi) \right] \\ + f(u, x, \mu, \hat{a}(u, t, x, \mathcal{U}(t, u, x, \mu), \partial_x \mathcal{U}(t, u, x, \mu))) + \frac{1}{2} \mathcal{U}^2(t, u, x, \mu) = 0 \\ V(T, u, x, \mu) = g(u, x, \mu), \end{cases}$$

where the master field $\mathcal{U} : I \times [0, T] \times \mathbb{R}^d \times \mathcal{M}_\lambda$ is defined as

$$\begin{aligned} \mathcal{U}(t, u, x, \mu) &:= \partial_x V(t, u, x, \mu) + \mathbb{E}_{(U, \xi) \sim \mu} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} V(t, U, \xi, \mu)(u, x) \right] \\ &= \partial_{\tilde{x}} \frac{\delta}{\delta m} v(t, \mu)(u, x), \quad \text{with } v(t, \mu) := \mathbb{E}_{(U, \xi) \sim \mu} [V(t, U, \xi, \mu)]. \end{aligned}$$

If the optimal feedback control \hat{a} obtained in (4.9) is Lipschitz in all its variables, then we get from (4.1) an optimal feedback control given by

$$\mathbf{a}^*(t, u, x, \mu) = \hat{a}(u, x, \mu, \mathcal{U}(t, u, x, \mu), \partial_x \mathcal{U}(t, u, x, \mu)), \quad (t, u, x, \mu) \in [0, T] \times I \times \mathbb{R}^d \times \mathcal{M}_\lambda.$$

4.2 The class of linear quadratic control problem

For sake of simplicity, we will present the computations in the case of a constant volatility term σ but the whole analysis could be performed by relaxing this hypothesis.

Given $\alpha \in \mathcal{A}$, and ξ an admissible initial condition and the set of controls $A \subset \mathbb{R}^m$ for $m \in \mathbb{N}^*$, we consider the controlled state process satisfying the SDE

$$\begin{cases} dX_t &= \left[A(U) + B(U)X_t + \mathbb{E}_{(\tilde{U}, \tilde{X}_t) \sim \mathbb{P}_{(U, X_t)}} [G_B(U, \tilde{U})\tilde{X}_t] + C(U)\alpha_t \right] dt + \sigma(U) dW_t, \\ X_0 &= \xi, \end{cases} \quad (4.10)$$

where $A \in L^2(I; \mathbb{R}^d)$, $B \in L^\infty(I; \mathbb{R}^{d \times d})$, $C \in L^\infty(I; \mathbb{R}^{d \times m})$, $G_B \in L^2_{\text{sym}}(I \times I; \mathbb{R}^{d \times d})$ and $\sigma \in L^2(I; \mathbb{R}^{d \times n})$ where the spaces L^2 , L^∞ and L^2_{sym} have been introduced in B.3.

The cost functional J is given by

$$\begin{aligned} J(\alpha) &= \mathbb{E} \left[\int_0^T \left(Q(U)(X_t - \tilde{\mathbb{E}}[\tilde{G}_Q(U, \tilde{U})\tilde{X}_t]) \cdot (X_t - \tilde{\mathbb{E}}[\tilde{G}_Q(U, \tilde{U})\tilde{X}_t]) + \alpha_t^\top N(U)\alpha_t \right) dt \right. \\ &\quad \left. + H(U)(X_T - \tilde{\mathbb{E}}[\tilde{G}_H(U, \tilde{U})\tilde{X}_T]) \cdot (X_T - \tilde{\mathbb{E}}[\tilde{G}_H(U, \tilde{U})\tilde{X}_T]) \right], \end{aligned} \quad (4.11)$$

where $Q, H \in L^\infty(I; \mathbb{S}_+^d)$ where \mathbb{S}_+^d denotes the set of positive symmetric matrices over $\mathbb{R}^{d \times d}$, $\tilde{G}_Q, \tilde{G}_H \in L_{\text{sym}}^2(I \times I; \mathbb{R}^{d \times d})$ and $N \in L^\infty(I; \mathbb{S}_{>+}^m)$ where $\mathbb{S}_{>+}^m$ denotes the set of positive definite symmetric matrices over $\mathbb{R}^{m \times m}$. Moreover, we suppose that there exists $c > 0$ such that

$$\langle N(U)y, y \rangle \geq c|y|^2, \quad \mathbb{P} - \text{a.s.}, \quad \forall y \in \mathbb{R}^m.$$

We also suppose that the operators $T_{\tilde{G}_Q}$ and $T_{\tilde{G}_H}$ are positive symmetric operators (where the operator notations have been introduced in B.3).

Remark 4.8 Under the assumptions on the model coefficients, we notice that for any admissible initial condition ξ and any $\alpha \in \mathcal{A}$

$$J(t, \xi, \alpha) \geq 0, \quad \forall t \in [0, T]. \quad (4.12)$$

This assumption is required as it will help us to derive an a-priori estimate on the triangular Riccati system which will arise from this control problem. However, in the numerical examples, we may add cross-product terms between the state and the control namely terms in the form $2\alpha_t^\top I(U)X_t$ and $\tilde{\mathbb{E}}[\alpha_t^\top G_I(U, \tilde{U})\tilde{X}_t]$ where $I \in L^\infty(I; \mathbb{R}^{m \times d})$ and $G_I \in L^2(I \times I; \mathbb{R}^{m \times d})$ even if we are not able to prove the existence and uniqueness of the associated Riccati equations in this setting.

Recalling the FBSDE system in (4.6), and following Proposition 4.6, an optimal control α^* should satisfy

$$\alpha_t^* = -\frac{1}{2}N(U)^{-1}C(U)^\top Y_t,$$

and following the notion of derivative introduced in (A.1), we end up with

$$\begin{cases} dX_t &= (A(U) + B(U)X_t + \tilde{\mathbb{E}}[G_B(U, \tilde{U})\tilde{X}_t] - \frac{1}{2}C(U)N(U)^{-1}C(U)^\top Y_t)dt \\ &\quad + \sigma(U)dW_t \\ dY_t &= (C_Y(U)Y_t + 2C_X(U)X_t + 2\tilde{\mathbb{E}}[\Psi_X(U, \tilde{U})\tilde{X}_t] + \tilde{\mathbb{E}}[\Psi_Y(U, \tilde{U})\tilde{Y}_t])dt + Z_t dW_t, \\ X_0 &= \xi, \\ Y_T &= 2(H(U)X_T + \tilde{\mathbb{E}}[G_H(U, \tilde{U})\tilde{X}_T]), \end{cases} \quad (4.13)$$

$$\begin{cases} C_Y(u) &= B(u)^\top, \\ C_t^X(u) &= Q(u), \\ \Psi_t^X(u, \tilde{u}) &= G_Q(u, \tilde{u}), \\ \Psi_t^Y(u, \tilde{u}) &= G_B^*(u, \tilde{u}) - G_I^*(u, \tilde{u})N_t(\tilde{u})^{-1}\Gamma_t(\tilde{u}), \\ G_Q(u, \tilde{u}) &= (\tilde{G}_Q \circ Q \circ \tilde{G}_Q)(u, \tilde{u}) - (Q(u) + Q(\tilde{u}))\tilde{G}_Q(u, \tilde{u}), \\ G_H(u, \tilde{u}) &= (\tilde{G}_H \circ H \circ \tilde{G}_H)(u, \tilde{u}) - (H(u) + H(\tilde{u}))\tilde{G}_H(u, \tilde{u}) \end{cases}$$

where we introduced the notations (see Appendix B.3 for the \circ and $*$ notations).

Theorem 4.9 *Under the assumptions on the model coefficients, the FBSDE (4.13) is uniquely solvable and the optimal control $\hat{\alpha}$ is then given as*

$$\hat{\alpha}_t = -\frac{1}{2}N(U)^{-1}C(U)^\top Y_t, \quad \forall t \in [0, T]. \quad (4.14)$$

Moreover, we have the following form for the adjoint process $Y = (Y_t)_{0 \leq t \leq T}$

$$Y_t = K_t(U)X_t + \tilde{\mathbb{E}}_{(\tilde{U}, \tilde{X}_t) \sim \mathbb{P}_{(U, X_t)}}[\tilde{K}_t(U, \tilde{U})\tilde{X}_t] + \Lambda_t(U), \quad (4.15)$$

where $K \in \mathcal{C}_1([0, T]; L^\infty([0, T]; \mathbb{S}_+^d))$ is the unique solution of the infinite dimensional Riccati equations

$$\begin{cases} \dot{K}_t(u) + \Phi(u, K_t(u)) - U(u, K_t(u))N(u)^{-1}U(u, K_t(u)) = 0 \\ K_T(u) = H(u) \end{cases} \quad (4.16)$$

for $\lambda(du)$ - a.e and where we introduced the measurable maps $\Phi : I \times \mathbb{S}_+^d \rightarrow \mathbb{R}^{d \times d}$ and $U : I \times \mathbb{S}_+^d \rightarrow \mathbb{R}^{d \times d}$ as

$$\begin{cases} \Phi(u, \kappa) & := B(u)^\top \kappa + \kappa^\top B(u), \\ U(u, \kappa) & := C(u)^\top \kappa. \end{cases}$$

Moreover, $\bar{K} \in \mathcal{C}_1([0, T], L_{sym}^2(I \times I, \mathbb{R}^{d \times d}))$ is the unique solution to the abstract Riccati equation on the Hilbert space $L^2(I \times I; \mathbb{R}^{d \times d})$

$$\begin{cases} \dot{\bar{K}}_t + \Psi(t, K_t, \bar{K}_t) - L(t, K_t, \bar{K}_t) = 0, \\ \bar{K}_T = G_H, \end{cases} \quad (4.17)$$

where we introduced the measurable map $\Psi, L : [0, T] \times L^\infty(I; \mathbb{S}_+^d) \times L^2(I \times I; \mathbb{R}^{d \times d}) \rightarrow L^2(I \times I; \mathbb{R}^{d \times d})$ and $V : L^2(I \times I; \mathbb{R}^{d \times d}) \rightarrow L^2(I \times I; \mathbb{R}^{d \times d})$ defined as

$$\begin{cases} \Psi(t, K_t, \bar{k})(u, v) & := K_t(u)G_B(u, v) + G_B^*(u, v)K_t(v) + B(u)^\top \bar{k}(u, v) + \bar{k}^*(u, v)B(v) \\ & \quad + (\bar{k}^* \circ G_A)(u, v) + (G_A^* \circ \bar{k})(u, v) + G_Q(u, v), \\ L(t, K_t, \bar{k})(u, v) & := U(u, K_t(u))N(u)^{-1}V(u, v)(\bar{k}) - V(u, v)^*(\bar{k})N(v)^{-1}U(v, K_t(v)) \\ & \quad - (V^* \circ N \circ V)(u, v), \\ V(u, v)(\bar{k}) & := C(u)^\top \bar{k}(u, v), \end{cases}$$

and where $\Lambda \in \mathcal{C}_1([0, T]; L^2(I; \mathbb{R}^d))$ is the unique solution to the linear equation on the Hilbert space $L^2(I; \mathbb{R}^d)$

$$\begin{cases} \dot{\Lambda}_t + F(t, \Lambda_t) - M(t, \Lambda_t) = 0, \\ \Lambda_T = 0. \end{cases} \quad (4.18)$$

where we introduced the measurable maps $F, M : [0, T] \times L^2(I; \mathbb{R}^d) \rightarrow L^2(I; \mathbb{R}^d)$ defined as

$$\begin{cases} F(t, \lambda)(u) & := B(u)^\top \lambda(u) + T_{G_A^*}(\lambda)(u) + K_t(u)A(u) + T_{\bar{K}_t}(A)(u), \\ M(t, \lambda)(u) & := U(u, K_t(u))^\top N(u)^{-1}C(u)^\top \lambda(u) + T_{V^*}(NC^\top \lambda)(u). \end{cases}$$

Proof. The proof of this result is discussed in Appendix B.4. \square

Remark 4.10 In fact, the Riccati for K (4.16) in Theorem 4.9 is standard as it can be solved u by u $\lambda(du)$ -a.e. Moreover, a known formula is known for scalar Riccati equations (see Equation (2.50) in [7]) and we will rely on this for the resolution of \bar{K} in (4.17).

4.3 Algorithms

We now illustrate the methodology introduced in the previous sections to solve this class of non exchangeable mean field control problems. We will, in the spirit of the methods developed in [30], develop two methods to solve the associated control problem. The first one will rely on a global learning method as in [19] which we will refer as Deep Graphon while the other one will take advantage of the FBSDE reformulation of the control problem which we will refer as Deep Graphon BSDE in line with [16]. Notice that even if we apply only two algorithms for the resolution, all kinds of algorithms developed in [30] can be used. Moreover, we give a Riccati solver for the abstract Riccati term \bar{K} over the Hilbert space $L^2(I \times I; \mathbb{R}^{d \times d})$.

4.3.1 Riccati solver

The Riccati equations depend on the values taken by U and involve integration with respect to a random variable \tilde{U} independent with respect to U and with the same law. To make the system solvable, we must assume that the interaction function and the coefficients Q and H are such that $G_Q(u, \tilde{u})$ and $G_H(u, \tilde{u})$ can be computed analytically. This restricts the class of admissible interaction functions and, for example, prevents us from using the function G_2 in (3.1).

We are then left to numerically integrate (4.16), (4.17), and (4.18) for $(U_i)_{i \in \llbracket 1, N \rrbracket}$, a given set of samples drawn from $\mathcal{U}([0, 1])$. Algorithm 1 provides the procedure used to solve the Riccati equations.

Note that, to obtain a highly accurate reference solution, we use a very small time step, significantly smaller than the one for in the neural-network-based algorithm.

Algorithm 1 Riccati solver for the non exchangeable mean field terms

Input: $(u_i)_{i \in \llbracket 1, N \rrbracket}$ samples from $\mathcal{U}([0, 1])$, L number of time steps, Δt the time step

Initialize : for $(i, j) \in \llbracket 1, N \rrbracket^2$

$$\bar{K}_L(u_i, u_j) = G_H(u_i, u_j), \quad K_L(u_i) = H(u_i), \quad \Lambda_L(u_i) = 0.$$

for $l = L - 1, \dots, 0$, **do**

 Calculate K term: for $i \in \llbracket 1, N \rrbracket$

$$\begin{aligned} \Phi(u_i) &= B(u_i)^\top K_{l+1}(u_i) + K_{l+1}(u_i)^\top B(u_i), \\ U(u_i) &= C(u_i)^\top K_{l+1}(u_i), \\ K_l(u_i) &= K_{l+1}(u_i) + \Delta t (\Phi(u_i) - U(u_i)N(u_i)^{-1}U(u_i)). \end{aligned}$$

 Calculate Λ term: for $i \in \llbracket 1, N \rrbracket$

$$F(u_i) = B(u_i)^\top \Lambda_{l+1}(u_i) + \frac{1}{M} \sum_{m=1}^N G_A(u_m, u_i) \Lambda_{l+1}(u_m) + K_{l+1}(u_i) A(u_i) + \frac{1}{M} \sum_{m=1}^N \bar{K}_{l+1}(u_i, u_m) A(u_m),$$

$$M(u_i) = K_{l+1}^\top(u_i) C(u_i) N(u_i)^{-1} C(u_i)^\top \Lambda_{l+1}(u_i) + \frac{1}{M} \sum_{m=1}^N \bar{K}_{l+1}(u_m, u_i) C(u_i) N(u_m) C^\top(u_m) \Lambda_{l+1}(u_m),$$

$$\Lambda_l(u_i) = \Lambda_{l+1}(u_i) + \Delta t (F(u_i) - M(u_i)).$$

 Calculate \bar{K} term: for $(i, j) \in \llbracket 1, N \rrbracket^2$

$$\begin{aligned} \Psi(u_i, u_j) &= K_t(u_i) G_B(u_i, u_j) + G_B(u_j, u_i) K_t(u_j) + B(u_i)^\top \bar{K}_{l+1}(u_i, u_j) + \bar{K}_{l+1}(u_j, u_i) B(u_j) + \\ &\quad \frac{1}{M} \sum_{m=1}^N \bar{K}_{l+1}(u_m, u_i) G_A(u_m, u_j) + \frac{1}{M} \sum_{m=1}^N G_A(u_m, u_i) \bar{K}_{l+1}(u_m, u_j) + G_Q(u_i, u_j), \end{aligned}$$

$$V(u_i, u_j) = C(u_i)^\top \bar{K}_{l+1}(u_i, u_j),$$

$$L(u_i, u_j) = U(u_i, K_t(u_i)) N(u_i)^{-1} V(u_i, u_j) - V(u_j, u_i) N^{-1}(u_j) U(u_j, K_t(u_j)) -$$

$$\frac{1}{M} \sum_{m=1}^N V(u_m, u_i) N(u_m) V(u_m, u_j),$$

$$\bar{K}_l(u_i, u_j) = \bar{K}_{l+1}(u_i, u_j) + \Delta t (\Psi(u_i, u_j) - L(u_i, u_j)).$$

Remark 4.11 In the linear quadratic case treated below, the values for K and Λ can be known analytically in some special cases, therefore we are left to solve only (4.17).

Once $\Lambda_l(u_i)$, $K_l(u_i)$, $\bar{K}_l(u_i, u_j)$, $(i, j) \in \llbracket 1, N \rrbracket^2$ are computed for $l \in \llbracket 0, L \rrbracket$, we can recover an estimation of $X_{l\Delta t}(u_i)$ for $i \in \llbracket 1, N \rrbracket$ solving the Euler scheme and we estimate $Y_{l\Delta t}$:

$$Y_{l\Delta}(u_i) \simeq K_{l\Delta t}(u_i)X_{l\Delta t}(u_i) + \frac{1}{M} \sum_{m=1}^M \bar{K}_{l\Delta t}(u_i, u_m)X_{l\Delta t}(u_m) + \Lambda_{l\Delta t}(U_i), \quad (4.19)$$

and then recover the optimal control using (4.14).

4.3.2 Deep Graphon

The Deep Graphon algorithm allows us to compute the optimal control α associated with the optimization problem (4.10)-(4.11) for all X_0 sampled from a probability space \mathbb{M} . It uses two networks: \mathcal{T}^{θ_1} , parametrized by θ_1 , and \mathcal{B}^{θ_2} , parametrized by θ_2 , both taking values in $\mathbb{R}^{r \times m}$, where m denotes the dimension of the control. The first network \mathcal{T}^{θ_1} inputs $(t, u, x) \in [0, T] \times [0, 1] \times \mathbb{R}^d$ and the second ones inputs $(t, y) \in [0, T] \times \mathbb{R}^J$ where y represents the moment vector of the distribution. We note $\theta := (\theta_1, \theta_2)$.

To simplify the notation, we assume in Algorithm 2 that at each iteration a single distribution is sampled to initialize X_0 . The batched version of the algorithm is straightforward.

Algorithm 2: Deep Graphon algorithm for non exchangeable mean-field algorithm

Input: N batch size, L number of time steps, θ parameters of the networks, ρ learning rate

for each epoch e do

Sample $(U_n, X_n) \sim \mathcal{U}([0, 1]) \otimes \mu$, $n \in \llbracket 1, N \rrbracket$ where μ is sampled in \mathbb{M} .

Initialize cost: $\hat{C}_n = 0$ for $n \in \llbracket 1, N \rrbracket$, $\Delta t = \frac{T}{L}$.

for $l = 1 \dots, L$ do

Calculate $\tilde{W} = (\frac{1}{N} \sum_{n=1}^N |X_n|^j)_{j \in \llbracket 1, J \rrbracket}$,

Calculate $\hat{W} = \mathcal{B}^{\theta_2}(l\Delta t, \tilde{W})$,

Calculate $W_n = \mathcal{T}^{\theta_1}(l\Delta t, U_n, X_n)$ for $n \in \llbracket 1, N \rrbracket$,

Estimate the control $\alpha_n = \sum_{k=1}^r W_{n,r} \hat{W}_r$ for $n \in \llbracket 1, N \rrbracket$.

Update cost : for $n \in \llbracket 1, N \rrbracket$,

$$d_n = Q(U_n)(X_n - \frac{1}{N} \sum_{m=1}^N \tilde{G}_Q(U_n, U_m)X_m),$$

$$e_n = (X_n - \frac{1}{N} \sum_{m=1}^N \tilde{G}_Q(U_n, U_m)X_m),$$

$$\hat{C}_n = \hat{C}_n + \Delta t [d_n \cdot e_n] + \alpha_n^\top N(U_n)\alpha_n.$$

Sample $g_n \sim \mathcal{N}(0, I_d)$ for $n \in \llbracket 1, N \rrbracket$.

Estimate X drift : for $n \in \llbracket 1, N \rrbracket$,

$$T_n^X = A(U_n) + B(U_n)X_n + \frac{1}{N} \sum_{m=1}^N G_B(U_n, U_m)X_m + C(U_n)\alpha_n.$$

Update the state : for $n \in \llbracket 1, N \rrbracket$, $X_n = X_n + \Delta t T_n^X + \sigma(U_n)g_n\sqrt{\Delta t}$.

end

Add terminal cost : for $n \in \llbracket 1, N \rrbracket$,

$$\hat{C}_n = \hat{C}_n + H(U_n)(X_n - \frac{1}{N} \sum_{m=1}^N \tilde{G}_H(U_n, U_m)X_m) \cdot (X_n - \frac{1}{N} \sum_{m=1}^N \tilde{G}_H(U_n, U_m)X_m).$$

Cost function to minimize: $J(\theta) = \frac{1}{N} \sum_{n=1}^N \hat{C}_n$.

$\theta = \theta - \rho \nabla J(\theta)$

end

4.3.3 Deep Graphon BSDE

The Deep Graphon BSDE Algorithm 2 allows us to calculate the optimal $Z = (Z_t)_{0 \leq t \leq T}$ and the cost function Y_0 for all X_0 sampled in a probability space \mathbb{M} by solving the FBSDE (4.13).

It uses four different networks:

- Two networks to approximate Y_0 : ϕ^{ζ_1} with parameter ζ_1 with input in $I \times \mathbb{R}^d$ and output in $\mathbb{R}^{r \times d}$, ψ^{ζ_2} with parameter ζ_2 with input in \mathbb{R}^J and output in $\mathbb{R}^{r \times d}$. We note $\zeta = (\zeta_1, \zeta_2)$.
- Two others neural network, \mathcal{T}^{θ_1} parametrized with θ_1 and \mathcal{B}^{θ_2} parametrized by θ_2 , both with output in $\mathbb{R}^{r \times d}$, are used to learn the Z term in the BSDE. The first \mathcal{T}^{θ_1} inputs $(t, u, x) \in [0, T] \times [0, 1] \times \mathbb{R}^d$ and the second ones inputs $(t, y) \in [0, T] \times \mathbb{R}^J$. We still note $\theta = (\theta_1, \theta_2)$.

We present Algorithm 3.

Algorithm 3: Deep Graphon BSDE algorithm for non exchangeable mean-field algorithm

Input: N the batch size, L number of time steps, ζ, θ, ρ learning rate

for each epoch e **do**

Sample $(U_n, X_n) \sim \mathcal{U}([0, 1]) \otimes \mu$, $n \in \llbracket 1, N \rrbracket$ where μ sampled in \bar{M} ,

Calculate $\tilde{W} = (\frac{1}{N} \sum_{n=1}^N |X_n|^j)_{j \in \llbracket 1, J \rrbracket}$.

Calculate $\hat{W} = \psi^{\zeta_2}(\tilde{W})$.

Calculate $W_n = \phi^{\zeta_1}(U_n, X_n)$ for $n \in \llbracket 1, N \rrbracket$.

Initialize $Y_n = \sum_{k=1}^r W_{n,r} \hat{W}_r$ for $n \in \llbracket 1, N \rrbracket$, $\Delta t = \frac{T}{L}$.

for $l = 1 \dots, L$ **do**

Calculate $\tilde{W} = (\frac{1}{N} \sum_{n=1}^N |X_n|^j)_{j \in \llbracket 1, J \rrbracket}$.

Calculate $\hat{W} = \mathcal{B}^{\theta_2}(l\Delta t, \tilde{W})$.

Calculate $W_n = \mathcal{T}^{\theta_1}(l\Delta t, U_n, X_n)$ for $n \in \llbracket 1, N \rrbracket$.

Estimate the control $Z_n = \sum_{k=1}^r W_{n,r} \hat{W}_r$ for $n \in \llbracket 1, N \rrbracket$.

Estimate Y trend: for $n \in \llbracket 1, N \rrbracket$,

$$T_n^Y = C_Y(U_n)Y_n + 2C_X(U_n)X_n + \frac{2}{N} \sum_{m=1}^N \Psi_X(U_n, U_m)X_m + \frac{1}{N} \sum_{m=1}^N \Psi_Y(U_n, U_m)Y_m.$$

Sample $g_n \sim \mathcal{N}(0, I_d)$ for $n \in \llbracket 1, N \rrbracket$.

Update the value function : for $n \in \llbracket 1, N \rrbracket$, $\hat{Y}_n = Y_n + T_n^Y \Delta t + Z_n g_n \sqrt{\Delta t}$.

Estimate X drift : for $n \in \llbracket 1, N \rrbracket$,

$$T_n^X = A(U_n) + B(U_n)X_n + \frac{1}{N} \sum_{m=1}^N G_B(U_n, U_m)X_m - \frac{1}{2}C(U_n)N(U_n)^{-1}C(U_n)^\top Y_n.$$

Update the state : for $n \in \llbracket 1, N \rrbracket$, $X_n = X_n + \Delta t T_n^X + \sigma(U_n)g_n \sqrt{\Delta t}$.

Store the value function $Y_n = \hat{Y}_n$ for $n \in \llbracket 1, N \rrbracket$.

end

Calculate target: for $n \in \llbracket 1, N \rrbracket$, $\hat{T}_n = 2\left(H(U_n)X_n + \frac{1}{N} \sum_{m=1}^N G_H(U_n, U_m)X_m\right)$.

Calculate the error $J(\zeta, \theta) = \frac{1}{N} \sum_{j=1}^N \left(\hat{T}_n - Y_n\right)^2$.

$(\zeta, \theta) = (\zeta, \theta) - \rho \nabla J(\zeta, \theta)$.

end

4.4 Numerical experiments

We present several numerical experiments illustrating the optimal control of non-exchangeable mean field systems. We first consider a linear–quadratic control problem arising in finance. We then turn to a more complex example, illustrating our approach in a non-toy, nonlinear and non-quadratic setting. In the sequel we assume that X_0 is sampled from the space of distributions \mathbb{M} , generated randomly as a mixture of three Gaussian laws. At each iteration of the gradient descent algorithm,

we sample for $k \in \llbracket 1, 3 \rrbracket$, $W_k \sim \mathcal{U}([0, 1])$ and $(\mu_k, \sigma_k) \sim \mathcal{U}([0, 1])^2$, and then for any $n \in \llbracket 1, N \rrbracket$ we compute

$$X_{0,n} = \frac{1}{\sum_{k=1}^3 W_k} \sum_{k=1}^3 W_k Z_{k,n}, \quad \text{with } Z_{k,n} \sim \mathcal{N}(\mu_k, \sigma_k^2).$$

All results are obtained using a classical feedforward neural network with 3 hidden layers of 10 neurons or the spline KAN with 2 hidden layers of 10 neurons with 5 grid meshes. We take the G_1 interaction function. In all the tests, we take $r = 10$ for the DeepONet network. Since we must discretize both in time and with a high number of trajectories to accurately represent the distributions, GPU memory becomes the limiting factor.

For all experiments we use the ADAM gradient descent algorithm with a learning rate of 0.001, running for 80,000 iterations.

The networks are trained using N trajectories to sample distributions with Euler schemes discretized using L time steps. Then, for a given distribution $\mu \in \mathbb{M}$, we estimate the cost function $C^{\text{Alg}}(\mu)$ (depending on the resolution method) associated with the optimal control obtained (for example using the time-discretized version of (4.11) in the linear–quadratic case). This cost is estimated using a distribution sampled with the N trajectories generated from the samples of X_0 , U , and the Brownian motions discretized with L time steps. In the two examples tested below, we either have:

- an estimate of the optimal control and cost function $C^{\text{anal}}(\mu)$ from the Riccati equation using the same samples of X_0 , U , and the Brownian motion,
- or an analytical control that allows us to estimate the cost function $C^{\text{anal}}(\mu)$ with the same samples.

Sampling 1000 distributions $(\mu_m)_{1 \leq m \leq 1000}$, we obtain $(C^{\text{Alg}}(\mu_m))_{m \in \llbracket 1, 1000 \rrbracket}$ and $(C^{\text{anal}}(\mu_m))_{m \in \llbracket 1, 1000 \rrbracket}$, and we can report different error measures whenever Alg = *DG* (Deep Graphon) and Alg = *BSDE* (Deep Graphon BSDE)

$$\begin{cases} E_{\text{Alg}}^{\text{abs}} & := \frac{1}{1000} \sum_{m=1}^{1000} |C^{\text{Alg}}(\mu_m) - C^{\text{anal}}(\mu_m)|, \\ E_{\text{Alg}}^{L_2} & := \frac{1}{1000} \sum_{m=1}^{1000} |C^{\text{Alg}}(\mu_m) - C^{\text{anal}}(\mu_m)|^2, \\ E_{\text{Alg}}^{\text{Sup}} & := \max_{m \in \llbracket 1, 1000 \rrbracket} |C^{\text{Alg}}(\mu_m) - C^{\text{anal}}(\mu_m)|. \end{cases}$$

4.4.1 A systemic risk model

The following example is an extension of the mean-field systemic risk model introduced first in [10] to the case of heterogeneous banks. The following model representing the log monetary reserve of each bank $u \in I$ as the process $(X_t|U = u)_{0 \leq t \leq T}$

$$\begin{cases} dX_t & = \left(\kappa(U)(\tilde{\mathbb{E}}_{(\tilde{U}, \tilde{X}_t)}[\tilde{G}_\kappa(U, \tilde{U})\tilde{X}_t] - X_t) + \alpha_t \right) dt + \sigma(U)dW_t, \\ X_0 & = \xi, \end{cases}$$

where $\kappa \in L^\infty(I; \mathbb{R}_-)$ acts like a mean reversion term, $\sigma \in L^2(I; \mathbb{R})$ measures the volatility of the bank reserve and $\tilde{G}_\kappa \in L^2(I \times I; \mathbb{R})$ is a graphon, i.e a measurable, bounded and symmetric map from $I \times I$ to \mathbb{R}_+ measuring the rate of borrowing between banks. Moreover, $\alpha = (\alpha_t)_{0 \leq t \leq T}$ is the control rate of borrowing/ lending to a central bank that aims to minimize the functional cost

$$J(\alpha) = \mathbb{E} \left[\int_0^T f(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t) dt + g(U, X_T, \mathbb{P}_{(U, X_T)}) \right],$$

where the running and terminal cost functions are given by

$$\begin{cases} f(u, x, \mu, a) & := \eta \left(x - \int_{I \times \mathbb{R}^d} (\tilde{G}_\eta(u, v) \tilde{x}) \mu(dv, d\tilde{x}) \right)^2 + a^2 + qa \left(x - \int_I \tilde{G}_q(u, v) \tilde{x} \mu(dv, d\tilde{x}) \right), \\ g(u, x, \mu) & := r \left(x - \int_{I \times \mathbb{R}^d} (\tilde{G}_r(u, v) \tilde{x}) \mu(dv, d\tilde{x}) \right)^2, \end{cases}$$

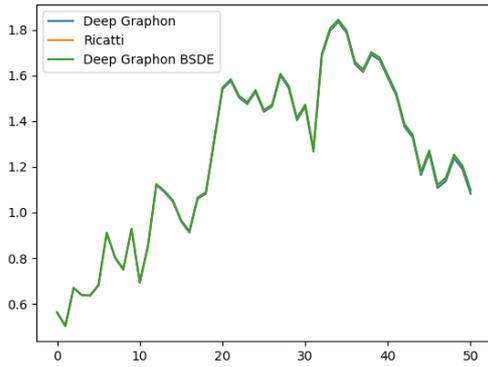
for some constants $\eta \geq 0, r \geq 0$ and q .

This model falls in the setting developed in Section 4.2 and applying Theorem 4.9 (to its extended version, see Remark 4.8), the optimal control $\hat{\alpha}$ is given by

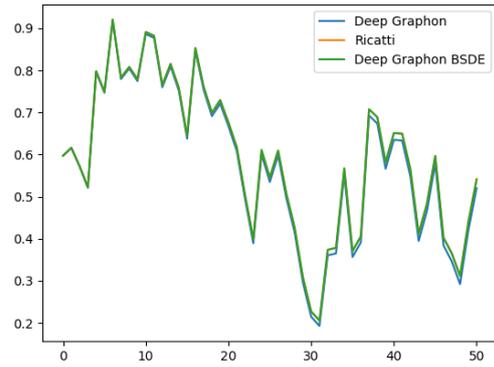
$$\alpha_t^* = \frac{1}{2} \left(q(X_t - \tilde{\mathbb{E}}[\tilde{G}_q(U, \tilde{U}) \tilde{X}_t]) - Y_t \right), \quad \forall t \in [0, T].$$

where Y is given by (4.19) and where K, \bar{K}, Λ are the solution to the associated Riccati equations.

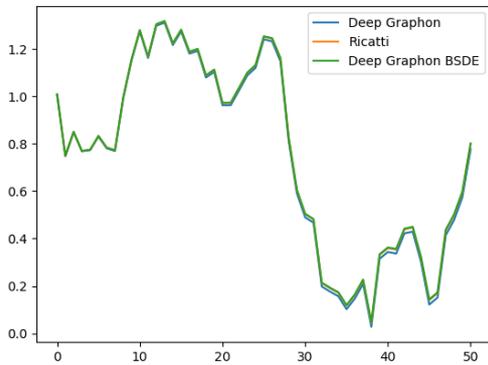
We now present some numerical examples which illustrate the algorithms developed and their accuracy in learning the optimal trajectory X^* associated to the optimal control α^* using $N = 10000$ and $L = 50$. We plot it by approximating the optimal trajectory with one and four moments and show the results. The results are very stable with the number of moments taken but taking four moments for this simple interaction case slightly degrades the results.



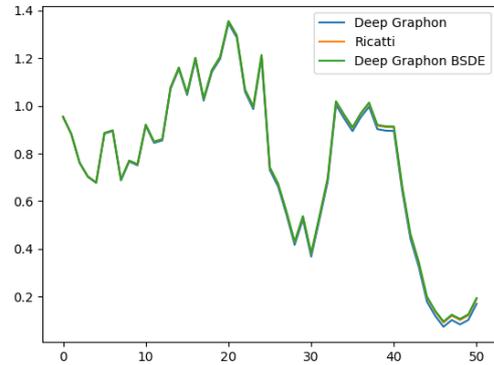
Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$



Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$

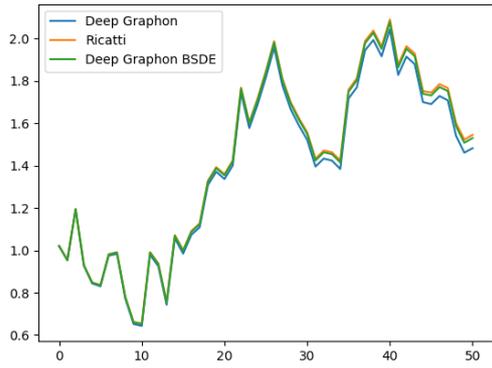


Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$

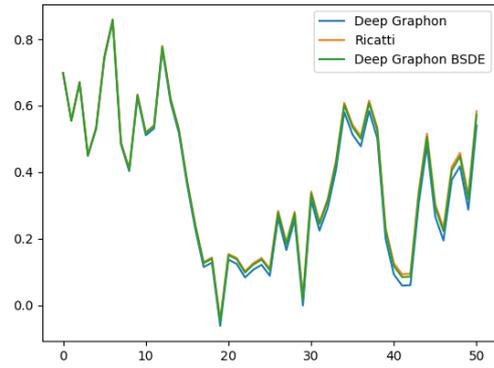


Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$

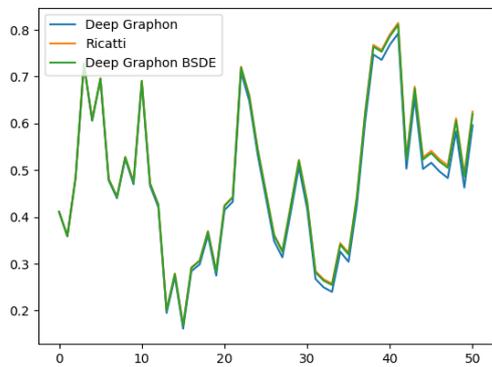
Figure 9: One moment approximation of the optimal feedback map a^* and associated optimal state trajectory X^* (with the feedforward network):



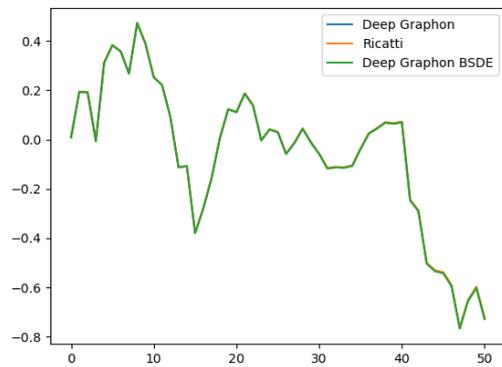
Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$



Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$



Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$



Optimal trajectory of $(X_t^*)_{0 \leq t \leq T}$

Figure 10: Four moments approximation of the optimal feedback map a^* (with the feedforward network)

We also give below two tables summarizing the results obtained sampling 1000 distributions.

Criteria	$E_{\text{Alg}}^{\text{abs}}$	$E_{\text{Alg}}^{L_2}$	$E_{\text{Alg}}^{\text{sup}}$	time
<i>DG</i>	7.5e-4	1.8e-6	7.9e-3	14610
<i>BSDE</i>	3e-4	9.7e-7	1.5e-2	10660

Table 1: One moment

Criteria	$E_{\text{Alg}}^{\text{abs}}$	$E_{\text{Alg}}^{L_2}$	$E_{\text{Alg}}^{\text{Sup}}$	time
<i>DG</i>	1e-3	3.4e-6	1e-2	14650
<i>BSDE</i>	2.7e-4	5e-7	6e-3	10770

Table 2: two moments

Criteria	E_X^{abs}	$E_X^{L_2}$	E_X^{Sup}	time
<i>DG</i>	3e-4	3.5e-7	4.7e-3	14770
<i>BSDE</i>	2.5e-4	4.4e-7	9.7e-3	10870

Table 3: three moments

Criteria	E_X^{abs}	$E_X^{L_2}$	E_X^{Sup}	time
<i>DG</i>	9e-4	3.5e-6	1.3e-2	14900
<i>BSDE</i>	3e-4	2e-6	2.9e-2	11110

Table 4: four moments

Table 5: Error with Riccati as reference depending on the number of moments J used as inputs of neural networks. Feedforward with tangent 3 layers of 10 neurons

Criteria	E_X^{abs}	$E_X^{L_2}$	E_X^{Sup}	time
<i>DG</i>	1e-4	3.8e-8	8e-4	17630
<i>BSDE</i>	4.5e-5	2.8e-8	2.7e-3	13180

Table 6: one moment

Criteria	E_X^{abs}	$E_X^{L_2}$	E_X^{Sup}	time
<i>DG</i>	2.8e-5	5.2e-9	8e-4	17690
<i>BSDE</i>	5.4e-5	2.2e-8	1.6e-3	12850

Table 7: two moments

Criteria	E_X^{abs}	$E_X^{L_2}$	E_X^{Sup}	time
<i>DG</i>	8.5e-5	2.1e-8	7.e-4	17400
<i>BSDE</i>	1.e-4	2e-7	6.6e-3	13060

Table 8: three moments

Criteria	E_X^{abs}	$E_X^{L_2}$	E_X^{Sup}	time
<i>DG</i>	2.7e-4	2.1e-5	1.4e-1	17740
<i>BSDE</i>	1.2e-4	6.9e-7	1.7e-2	13190

Table 9: four moments

Table 10: Error with Riccati as reference depending on the number of moments J used as inputs of neural networks. Spline KAN with 2 layers of 10 neurons, 5 grid meshes.

Results in Tables 5 , 10 show the accuracy of the method. The KAN network allow us to get more accurate results. The use of a high number of moments degrades the results.

4.4.2 A non linear quadratic example

As an illustration of the algorithm above, we consider a one-dimensional model with

$$\begin{cases} b(t, u, x, \mu, a) &= \beta(t, u, x, \mu) + a, \\ f(t, u, x, \mu, a) &= F(t, u, x, \mu) + \frac{a^2}{2}, \\ g(u, x, \mu) &= \mathbb{E}_{(U, \xi) \sim \mu} [w(x - G(u, U)\xi)], \end{cases}$$

where w is a map on $\mathcal{C}_2(\mathbb{R})$, F is a map on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$ to be chosen later and G is a symmetric map over $I \times I$. In our setting, we are looking for a value function to the Bellman equation in the form $V(u, t, x, \mu) = e^{\eta(T-t)} \mathbb{E}_{(U, \xi) \sim \mu} [w(x - G(u, U)\xi)]$ for a positive constant $\eta \geq 0$ and by simple calculations, the master field map is given by

$$\mathcal{U}(t, u, x, \mu) = e^{\eta(T-t)} \left(\mathbb{E}_{(U, \xi) \sim \mu} [w'(x - G(u, U)\xi)] - \mathbb{E}_{(U, \xi) \sim \mu} [G(U, u)w'(\xi - G(U, u)x)] \right).$$

Taking $w = \cos$ and defining to alleviate notations the following quantities

$$\left\{ \begin{array}{l} A(u, \mu) \\ B(u, \mu) \\ k(u, v) \\ C(u, x, \mu) \\ D(u, x, \mu) \\ K_{\cos}(u, \mu) \\ K_{\sin}(u, \mu) \\ B_{\cos}(t, u, x, \mu) \\ B_{\sin}(t, u, x, \mu) \\ U_{\cos}(t, u, x, \mu) \\ U_{\sin}(t, u, x, \mu) \end{array} \right. := \begin{array}{l} \mathbb{E}_{(U, \xi) \sim \mu} [\cos(G(u, U)\xi)], \\ \mathbb{E}_{(U, \xi) \sim \mu} [\sin(G(u, U)\xi)], \\ \eta + \frac{\sigma^2}{2} (1 + G^2(u, v)) \\ \mathbb{E}_{(U, \xi) \sim \mu} [G(u, U)\sin(\xi)\cos(G(u, U)x)] \\ \mathbb{E}_{(U, \xi) \sim \mu} [G(u, U)\cos(\xi)\sin(G(u, U)x)] \\ \mathbb{E}_{(U, \xi) \sim \mu} [k(u, U)\cos(G(u, U)\xi)], \\ \mathbb{E}_{(U, \xi) \sim \mu} [k(u, U)\sin(G(u, U)\xi)], \\ \mathbb{E}_{(U, \xi) \sim \mu} [(G(u, U)\beta(t, U, \xi, \mu) - \beta(t, u, x, \mu))\cos(G(u, U)\xi)], \\ \mathbb{E}_{(U, \xi) \sim \mu} [(G(u, U)\beta(t, U, \xi, \mu) - \beta(t, u, x, \mu))\sin(G(u, U)\xi)], \\ \mathbb{E}_{(U, \xi) \sim \mu} [(\mathcal{U}(t, u, x, \mu) - G(u, U)\mathcal{U}(t, U, \xi, \mu))\cos(G(u, U)\xi)], \\ \mathbb{E}_{(U, \xi) \sim \mu} [(\mathcal{U}(t, u, x, \mu) - G(u, U)\mathcal{U}(t, U, \xi, \mu))\sin(G(u, U)\xi)], \end{array}$$

choosing the map F to be equal to

$$F(t, u, x, \mu) = e^{\eta(T-t)} \left(\cos(x)\alpha(t, u, x, \mu) + \sin(x)\gamma(t, u, x, \mu) \right) - \frac{1}{2} e^{2\eta(T-t)} \Lambda(u, x, \mu)^2,$$

where we set

$$\left\{ \begin{array}{l} \alpha(t, u, x, \mu) \\ \gamma(t, u, x, \mu) \\ \Lambda(u, x, \mu) \end{array} \right. := \begin{array}{l} K_{\cos}(u, \mu) + B_{\sin}(t, u, x, \mu) + U_{\sin}(t, u, x, \mu), \\ K_{\sin}(u, \mu) - B_{\cos}(t, u, x, \mu) - U_{\cos}(t, u, x, \mu), \\ -\sin(x)A(u, \mu) + \cos(x)B(u, \mu) + C(u, x, \mu) - D(u, x, \mu), \end{array}$$

the function V defined above satisfies the Bellman equation. Moreover, the optimal control a^* is defined as

$$a^*(t, u, x, \mu) = -\partial_{\bar{x}} \frac{\delta}{\delta m} v(t, \mu)(u, x), \quad \text{with } v(t, \mu) = \mathbb{E}_{(U, \xi) \sim \mu} [V(t, U, \xi, \mu)].$$

We give below the results of the learning of the value function V using $N = 6000$ and $L = 50$. We observe that the error is much important than in the LQ case. This error is independent of the number J of moment taken. Some numerical tries seem to indicate that the error is mainly due to the Euler scheme resolution and the use of KAN does not improve the results.

NbMom	$E_{\text{Alg}}^{\text{abs}}$	$E_{\text{Alg}}^{L_2}$	$E_{\text{Alg}}^{\text{Sup}}$	time
1	8.9e-3	1.2e-4	4.4e-2	42600
2	8.6e-3	1.1e-4	3.2e-2	42900
3	8.7e-3	1.1e-4	3.4e-2	42800
4	8.4e-3	1.2e-4	3.4e-2	42000

Table 11: Result on non LQ problem with interacting network with feed forward : 3 hidden layers with 10 neurons

A Analysis tools over the space \mathcal{M}_λ

In this Appendix, we present the main analysis tools used to handle maps defined over the space \mathcal{M}_λ . It essentially relies on the notion of linear functional derivative and/or equivalently on the

Lion's derivative for which we recall a nice introduction in [7] (Chapter 5). The notion of derivative and convexity we are going to introduce is fairly inspired from the works in [22, 14] where the maps were instead defined over the space $L^2(I; \mathcal{P}_2(\mathbb{R}^d))$. In fact, both notions of derivatives are fairly similar but for sake of completeness, we reintroduce it in full generality.

A.1 A notion of derivative

Definition A.1 (Linear functional derivative on \mathcal{M}_λ).

1. Given a function $v : \mathcal{M}_\lambda \rightarrow \mathbb{R}$, we say that a measurable function

$$\frac{\delta}{\delta m} v : \mathcal{M}_\lambda \times I \times \mathbb{R}^d \ni (\mu, u, x) \mapsto \frac{\delta}{\delta m} v(\mu)(u, x) \in \mathbb{R},$$

is the linear functional derivative (or flat derivative) of v

- (1) For every compact $K \in \mathcal{M}_\lambda$, there exists a compact $C_K > 0$ such that

$$\left| \frac{\delta}{\delta m} v(\mu)(u, x) \right| \leq C_K(1 + |x|^2),$$

for every $u \in I$, $x \in \mathbb{R}^d$ and $\mu \in K$.

- (2) For every $\mu, \nu \in \mathcal{M}_\lambda$, we have

$$\begin{aligned} v(\nu) - v(\mu) &= \int_0^1 \int_{I \times \mathbb{R}^d} \frac{\delta}{\delta m} v((1-\theta)\mu + \theta\nu)(u, x) d(\nu - \mu)(u, x) d\theta, \\ &= \int_0^1 \int_I \int_{\mathbb{R}^d} \frac{\delta}{\delta m} v((1-\theta)\mu + \theta\nu)(u, x) d(\nu^u - \mu^u)(x) \lambda(du) d\theta \end{aligned}$$

2. We say that the function v admits a continuously differentiable flat derivative if

- (1) v admits a flat derivative $\frac{\delta}{\delta m} v$ satisfying $x \mapsto \frac{\delta}{\delta m} v(\mu)(u, x)$ is Fréchet differentiable with Fréchet derivative denoted by $x \mapsto \partial_{\bar{x}} \frac{\delta}{\delta m} v(\mu)(u, x)$ for all $(\mu, u) \in \mathcal{M}_\lambda \times I$
- (2) The map $(\mu, x) \mapsto \partial_{\bar{x}} \frac{\delta}{\delta m} v(\mu)(u, x)$ is continuous from $\mathcal{M}_\lambda \times I$ into \mathbb{R}^d for λ -a.e $u \in I$.
- (3) For every compact set $K \subset \mathcal{M}_\lambda$, there exists a constant $C_K \geq 0$ such that

$$\left| \partial_{\bar{x}} \frac{\delta}{\delta m} v(\mu)(u, x) \right| \leq C_K(1 + |x|),$$

for every $u \in I$, $x \in \mathbb{R}^d$ and $\mu \in K$.

3. We say that $v : [0, T] \times \mathcal{M}_\lambda \rightarrow \mathbb{R}$ is of class $\tilde{\mathcal{C}}^{1,2}([0, T] \times \mathcal{M}_\lambda)$ if

- (1) For every $\mu \in \mathcal{M}_\lambda$, the map $t \mapsto v(t, \mu)$ is continuously differentiable on $[0, T]$ and we denote by $(t, \mu) \mapsto \partial_t v(t, \mu)$ its time derivative.
- (2) For every $t \in [0, T]$, the derivative $\frac{\delta}{\delta m} v(t, \mu)(u, x)$ exists and its measurable in all its arguments.
- (3) $\frac{\delta}{\delta m} v(t, \mu)(u, x)$ is twice continuously differentiable on \mathbb{R}^d , i.e for any $(t, \mu, u) \in [0, T] \times \mathcal{M}_\lambda \times I$, the map $x \mapsto \frac{\delta}{\delta m} v(t, \mu)(u, x)$ and the gradient and the Hessian matrix

$$\partial_{\bar{x}} \frac{\delta}{\delta m} v : [0, T] \times \mathcal{M}_\lambda \times I \times \mathbb{R}^d \rightarrow \mathbb{R}^d, \quad \partial_{\bar{x}}^2 \frac{\delta}{\delta m} v : [0, T] \times \mathcal{M}_\lambda \times I \times \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$$

satisfy the following growth conditions: There exists a positive constant C s.t

$$\begin{cases} \left| \partial_{\bar{x}} \frac{\delta}{\delta m} v(t, \mu)(u, x) \right| \leq C(1 + |x| + \mathcal{W}(\mu, m \otimes \delta_0)), \\ \left| \partial_{\bar{x}}^2 \frac{\delta}{\delta m} v(t, \mu)(u, x) \right| \leq C, \end{cases}$$

for every $t \in [0, T]$, $u \in I$, $x \in \mathbb{R}^d$ and $\mu \in \mathcal{M}_\lambda$.

(4) The map $[0, T] \times \mathcal{M}_\lambda \ni (t, \mu) \mapsto \partial_t v(t, \mu)$ is continuous.

(5) For every $u \in I$ and every compact set $H \subset \mathbb{R}^d$, the functions $\partial_{\tilde{x}} \frac{\delta}{\delta m} v(t, \mu)(u, x)$ and $\partial_{\tilde{x}}^2 \frac{\delta}{\delta m} v(t, \mu)(u, x)$ are continuous functions of $(t, \mu) \in [0, T] \times \mathcal{M}_\lambda$ uniformly in $x \in H$.

Remark A.2

- In the core of the paper, the map v will be mainly defined over the space $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$. In this setting, the flat derivative of v is defined as a measurable map

$$\frac{\delta}{\delta m} v : I \times \mathbb{R}^d \times \mathcal{M}_\lambda \times I \times \mathbb{R}^d \ni (u, x, \mu, \tilde{u}, \tilde{x}) \mapsto \frac{\delta}{\delta m} v(u, x, \mu)(\tilde{u}, \tilde{x}) \in \mathbb{R},$$

and you can extend the previous points from Definition (A.1).

- Some example of standard type of functions for which we can compute the linear functional derivatives are discussed in [14, 22].

A.2 A notion of convexity

Definition A.3 (Convexity on \mathcal{M}_λ).

Given a function $v : \mathcal{M}_\lambda \rightarrow \mathbb{R}$ supposed to admit a continuously differentiable flat derivative in the sense of Definition A.1, we say that v is convex if for every $\mu, \mu' \in \mathcal{M}_\lambda$, we have

$$v(\mu') - v(\mu) \geq \mathbb{E} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} v(\mu)(U, X) \cdot (X' - X) \right],$$

where $(U, X) \sim \mu$ and $(U, X') \sim \mu'$.

More generally, if v is now defined on $I \times \mathbb{R}^d \times \mathcal{M}_\lambda$, we say that v is said to be convex if for every $(x, \mu) \in \mathbb{R}^d \times \mathcal{M}_\lambda$ and $(x', \mu') \in \mathbb{R}^d \times \mathcal{M}_\lambda$ and for $\lambda(du) - a.e$, we have

$$v(u, x', \mu') - v(u, x, \mu) \geq \partial_x v(u, x, \mu) \cdot (x' - x) + \mathbb{E} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} v(u, x, \mu)(U, X) \cdot (X' - X) \right],$$

where $(U, X) \sim \mu$ and $(U, X') \sim \mu'$.

B Optimal control related results

B.1 Proof of Theorem 4.2

The proof follows from a standard fixed point argument on the complete metric space $(\mathcal{M}_\lambda(I \times \mathcal{C}_{[t, T]}^d); \mathcal{W}_2)$ and is a straightforward adaptation of the proof of Theorem 2.6 in [14]. Indeed, define the map Φ as

$$\mathcal{M}_\lambda(I \times \mathcal{C}_{[t, T]}^d) \ni \nu \mapsto \Phi(\nu) = \mathbb{P}_{(U, X^\nu)},$$

where X^ν denotes the solution to the standard SDE with Lipschitz coefficients

$$\begin{cases} dX_s^\nu &= b(U, X_s^\nu, \nu_s, \alpha_s) ds + \sigma(U, X_s^\nu, \nu_s, \alpha_s) dW_s, \\ X_t^\nu &= \xi, \end{cases} \tag{B.1}$$

and where $\nu_s = x_s \# \nu$ with $x_s : I \times \mathcal{C}_{[t, T]}^d \ni (u, \omega) \mapsto x_s(u, \omega) = (u, \omega_s) \in I \times \mathbb{R}^d$ denotes the projection map. Under Assumption 4.1, the SDE (B.1) admits a continuous \mathbb{F} -adapted process and therefore $\mathbb{P}_{(U, X^\nu)}$ can be viewed as a probability measure on $\mathcal{P}^\lambda(I \times \mathcal{C}_{[t, T]}^d)$. Moreover, under standard estimates it is easy to verify that $\mathbb{E} \left[\sup_{0 \leq t \leq T} |X_t^\nu|^2 \right] < \infty$ such that $\mathbb{P}_{(U, X^\nu)} \in \mathcal{M}_\lambda(I \times \mathcal{C}_{[t, T]}^d)$

and therefore $\Phi : \mathcal{M}_\lambda(I \times \mathcal{C}_{[t,T]}^d) \rightarrow \mathcal{M}_\lambda(I \times \mathcal{C}_{[t,T]}^d)$ is well defined. Now, from standard estimates, we prove that Φ has a unique fixed point $\bar{\nu}$ in $\mathcal{M}_\lambda(I \times \mathcal{C}_{[t,T]}^d)$. Indeed, we have (assuming for simplicity $b = 0$)

$$\begin{aligned} \mathbb{E} \left[\sup_{t \leq s \leq r} |X_s^\nu - X_s^\mu|^2 \right] &\leq C \mathbb{E} \left[\int_t^r |\sigma(U, X_s^\nu, \nu_s, \alpha_s) - \sigma(U, X_s^\mu, \mu_s, \alpha_s)|^2 ds \right] \\ &\leq C \mathbb{E} \left[\int_t^r (\mathbb{E}[|X_s^\nu - X_s^\mu|^2] + \mathcal{W}_2(\nu_s, \mu_s)^2) ds \right] \\ &\leq C \mathbb{E} \left[\int_t^r (\mathbb{E}[\sup_{t \leq q \leq s} |X_q^\nu - X_q^\mu|^2] + \mathcal{W}_2(\nu_s, \mu_s)^2) ds, \right] \end{aligned}$$

where C is a positive constant which can change from line to line. From Grönwall's lemma, we end up with

$$\mathcal{W}_2(\Phi(\nu), \Phi(\mu))^2 \leq \mathbb{E} \left[\sup_{t \leq s \leq T} |X_s^\nu - X_s^\mu|^2 \right] \leq C(T-t)\mathcal{W}_2(\nu, \mu)^2,$$

and where we used $\mathcal{W}_2(\nu_s, \mu_s) \leq \mathcal{W}_2(\nu, \mu)$ for any $t \leq s \leq T$. Now, from standard arguments we conclude that the sequence $(\nu^{(k)})_{k \in \mathbb{N}}$ defined as

$$\begin{cases} \nu^{(k+1)} = \Psi(\nu^{(k)}), \\ \nu^{(0)} \text{ arbitrary point in } \mathcal{M}_\lambda(I \times \mathcal{C}_{[t,T]}^d). \end{cases}$$

is a Cauchy sequence for \mathcal{W}_2 and converges on $\mathcal{M}_\lambda(I \times \mathcal{C}_{[t,T]}^d)$ towards a unique fixed point $\bar{\nu}$. Considering the associated process $X^{\bar{\nu}}$ yields the unique solution to Theorem 4.2.

B.2 Proof of Proposition 4.6

The proof is essentially an adaption of the proof in [22] and therefore we just give the main ideas.

Step n°1 : Definition of the variation process.

Given an admissible control $\beta = (\beta_t)_{0 \leq t \leq T} \in \mathcal{A}$, we define the control $\delta := \beta - \alpha$. We notice that $\delta \in \mathcal{A}$ since \mathcal{A} is a convex set. We now define the \mathbb{R}^d -valued variation process $V = (V_t)_{0 \leq t \leq T}$ associated to the process X defined in (4.1) as the solution to the following SDE

$$\begin{cases} dV_t &= \left[\gamma_t \cdot V_t + \tilde{\mathbb{E}} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} b(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t)(\tilde{U}, \tilde{X}_t) \tilde{V}_t \right] + \eta_t \cdot \delta_t \right] dt \\ &+ \left[\hat{\gamma}_t \cdot V_t + \tilde{\mathbb{E}} \left[\partial_{\tilde{x}} \frac{\delta}{\delta m} b(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t)(\tilde{U}, \tilde{X}_t) \tilde{V}_t \right] + \hat{\eta}_t \cdot \delta_t \right] dW_t, \\ V_0 &= 0, \end{cases}$$

and where we denoted for every $t \in [0, T]$

$$\begin{cases} \gamma_t &= \partial_x b(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t), & \eta_t &= \partial_a b(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t), \\ \hat{\gamma}_t &= \partial_x \sigma(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t), & \hat{\eta}_t &= \partial_a \sigma(U, X_t, \mathbb{P}_{(U, X_t)}, \alpha_t). \end{cases}$$

We note that under the Assumptions 4.1-4.5 that (X, V) is an SDE satisfying the Assumptions of Theorem 4.2 and hence, (X, V) and hence V is uniquely defined.

We now denote the family of admissible controls $\alpha^\epsilon = \alpha + \epsilon \delta$ for $\epsilon \in [0, 1]$ and by $X^\epsilon := X^{\alpha^\epsilon}$ the associated controlled state process. Then, one can check under Assumptions 4.5.

$$\mathbb{E} \left[\sup_{0 \leq t \leq T} \left| \frac{X_t^\epsilon - X_t}{\epsilon} - V_t \right|^2 \right] \xrightarrow{\epsilon \rightarrow 0} 0. \quad (\text{B.2})$$

Step n°2 : Gâteaux derivative of J .

Now, relying in (B.2), on Assumptions B.2 and on Definition of the adjoint process Y in (4.6), one can check after some straightforward computations that

$$\lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \left(J(\alpha + \epsilon(\beta - \alpha)) - J(\alpha) \right) = \mathbb{E} \left[\int_0^T \partial_a H((U, X_t, \mathbb{P}_{(U, X_t)}, Y_t, Z_t) \cdot (\beta_t - \alpha_t) dt \right]. \quad (\text{B.3})$$

Now relying on the convexity assumption of the map $a \mapsto H(u, x, \mu, y, z, a)$ for any $(u, x, \mu, y, z) \in I \times \mathbb{R}^d \times \mathcal{P}_2^\lambda(I \times \mathbb{R}^d) \times \mathbb{R}^d \times \mathbb{R}^{d \times n}$, one can derive the statement of Proposition 4.6 from the Gateaux derivatives of J in (B.3)

B.3 Additional notations for the linear quadratic case

We work under the complete Hilbert space $L^2(I \times I; \mathbb{R}^{d \times d})$ and $L^2(I; \mathbb{R}^d)$ and we introduce the following notations

1. We say that a kernel $K \in L^2(I \times I; \mathbb{R}^{d \times d})$ is symmetric if the following holds

$$K(v, u)^\top = K(u, v), \quad du \otimes dv \text{ a.e.} \quad (\text{B.4})$$

We denote by $L_{\text{sym}}^2(I \times I; \mathbb{R}^{d \times d})$ the space of kernels $K \in L^2(I \times I; \mathbb{R}^{d \times d})$ satisfying (B.4).

2. Given a kernel $K \in L^2(I \times I; \mathbb{R}^{d \times d})$, its associated linear integral Hilbert-Schmidt operator T_K is defined as

$$L^2(I; \mathbb{R}^d) \ni f \mapsto T_K(f)(\cdot) = \int_I K(\cdot, v) f(v) dv \in L^2(I; \mathbb{R}^d).$$

We say that T_K is a symmetric non-negative operator on $L^2(I; \mathbb{R}^d)$ if its associated kernel satisfies (B.4) and $\langle f, T_K f \rangle_{L^2(I; \mathbb{R}^d)} \geq 0$ for every $f \in L^2(I; \mathbb{R}^d)$.

3. Given a kernel $K \in L^2(I \times I; \mathbb{R}^{d \times d})$, we define $K^* \in L^2(I \times I; \mathbb{R}^{d \times d})$ as

$$K^*(u, v) := K(v, u)^\top.$$

When $K \in L_{\text{sym}}^2(I \times I; \mathbb{R}^{d \times d})$, we have $K^* = K$. We also notice that $T_{K^*} = (T_K)^*$ where $(T_K)^*$ denotes the adjoint operator of T_K .

4. Given two kernels $K, W \in L^2(I \times I; \mathbb{R}^{d \times d})$, it is easy to check that the operator $T_K \circ T_W$ is associated with the kernel $(K \circ W)$ defined by

$$(K \circ W)(u, v) := \int_I K(u, w) W(w, v) dw,$$

5. Given $L \in L^\infty(I; \mathbb{R}^{d \times d})$, we define the multiplicative operator associated to L as the linear operator on $L^2(I; \mathbb{R}^d)$ defined by

$$L^2(I; \mathbb{R}^d) \ni f \mapsto \left(I \ni w \mapsto M_L(f)(w) = L^w f(w) \in \mathbb{R}^d \right) \in L^2(I; \mathbb{R}^d).$$

Given $K, W \in L^2(I \times I; \mathbb{R}^{d \times d})$ and $L \in L^\infty(I; \mathbb{R}^{d \times d})$, the operator $T_K \circ M_L \circ T_W$ is associated with the kernel $(K \circ L \circ W)$ defined by

$$(K \circ L \circ W)(u, v) = \int_I K(u, w) L(w) W(w, v) dw.$$

Given (E, d) a Polish metric space (e.g. $\mathbb{R}^d, \mathcal{P}_2(\mathbb{R}^d)$) endowed with its natural Borel σ -algebra., we also introduce the following spaces.

(1)

$$\begin{cases} L^2(I; E) := \{\varphi : I \ni u \mapsto \varphi(u) \text{ measurable and } \int_I d(\varphi(u), e)^2 du < +\infty\}, \\ L^2(I \times I; E) := \{\varphi : I \times I \ni (u, v) \mapsto \varphi(u, v) \text{ measurable and } \int_{I \times I} d(\varphi(u, v), e)^2 du < +\infty\}. \end{cases}$$

Moreover, if E is a normed metric space with norm $(\|\cdot\|)$ (e.g \mathbb{R}^d), we introduce

(2)

$$\begin{cases} L^\infty(I; E) := \{\varphi : I \ni u \mapsto \varphi(u) \text{ measurable and } \operatorname{ess\,sup}_{u \in I} \|\varphi(u)\|_E < +\infty\}, \\ L^\infty(I \times I; E) := \{\varphi : I \times I \ni (u, v) \mapsto \varphi(u, v) \text{ measurable and } \operatorname{ess\,sup}_{u, v \in I} \|\varphi(u, v)\|_E < +\infty\}. \end{cases}$$

B.4 Proof of Theorem 4.9

The proof is essentially a combination of the proofs in [22, 13] and therefore we just give the main ideas of the proofs in the current setting.

Step n°1 : Ansatz form for Y .

Motivated by the standard empirical link between the value function and the adjoint process Y in the stochastic maximum principle and the linear-quadratic parametrization, we are looking for a solution to the FBSDE (4.13) by guessing

$$Y_t = K_t(U)X_t + \tilde{\mathbb{E}}[\bar{K}_t(U, \tilde{U})\tilde{X}_t] + \Lambda_t, \quad 0 \leq t \leq T, \quad (\text{B.5})$$

where $K \in \mathcal{C}^1([0, T]; L^\infty(I; \mathbb{S}_+^d))$, $\bar{K} \in \mathcal{C}^1([0, T]; L_{\text{sym}}^2(I \times I; \mathbb{R}^d \times \mathbb{R}^d))$ and $\Lambda \in \mathcal{C}^1([0, T]; L^2(I; \mathbb{R}^d))$ are to be determined through Riccati equations. Plugging the ansatz (B.5) into the dynamics of X and after some tedious but straightforward computations, we obtain the Riccati equations stated in Theorem 4.9.

Step n°2 : Solvability of the Riccati equations

The solvability of (K, \bar{K}, Λ) , i.e., the proof of their existence and unicity over respectively the spaces $\mathcal{C}^1([0, T]; L^\infty(I; \mathbb{S}_+^d))$, $\mathcal{C}^1([0, T]; L_{\text{sym}}^2(I \times I; \mathbb{R}^{d \times d}))$ and $\mathcal{C}^1([0, T]; L^2(I; \mathbb{R}^d))$ is done in [13] (see [15] for an extension to the common noise setting). Essentially, the Riccati on K is standard and can be solved by standard theory. However, the Riccati equation for \bar{K} is of a new type due to the heterogenous structure of interactions and its solvability requires explicitly the assumptions on the model coefficients resulting in the positivity of J (see (4.12)). Indeed, we get a fundamental relation (see Proposition 3.1 in [13]) and we are able to bound the operator norm of $T_{\bar{K}_t}$ uniformly in time which gives us an a-priori estimate on the solution and helps us to go backward in time and get global existence and uniqueness over $[0, T]$. Finally, Λ is a linear ODE on the Hilbert space $L^2(I; \mathbb{R}^d)$ and can be solved from standard theory on infinite dimensional spaces.

Therefore, we identified a solution (X, Y, Z) to the FBSDE. This solution is in fact unique and the unicity is proved in [22] so we refer to the references therein for a full proof of this result.

References

- [1] G.A. Alvarez, E. Bayraktar, and I. Ekren. Contracting a crowd of heterogeneous agents. *arXiv preprint arXiv:2507.09415*, 2025.
- [2] E. Bayraktar, S. Chakraborty, and R. Wu. Graphon mean field systems. *The Annals of Applied Probability*, 33(5):3587–3619, 2023.
- [3] A. Bensoussan, J. Frehse, and S.P. Yam. *Mean field games and mean field type control theory*. Briefs in Mathematics. Springer, 2013.
- [4] P. E. Caines and M. Huang. Graphon mean field games and their equations. *SIAM Journal on Control and Optimization*, 59(6):4373–4399, 2021.
- [5] Z. Cao and M. Laurière. Probabilistic analysis of graphon mean field control. *arXiv preprint arXiv:2505.19664*, 2025.
- [6] R. Carmona and F. Delarue. Forward–backward stochastic differential equations and controlled mckean–vlasov dynamics. *The Annals of Probability*, 43(5):2647–2700, 2015.
- [7] R. Carmona and F. Delarue. *Probabilistic Theory of Mean Field Games with Applications I: Mean Field FBSDEs, Control, and Games*, volume 83 of *Probability Theory and Stochastic Modelling*. Springer, Cham, 2018.
- [8] R. Carmona and F. Delarue. *Probabilistic Theory of Mean Field Games with Applications II: Mean Field Games with Common Noise and Master Equations*, volume 84 of *Probability Theory and Stochastic Modelling*. Springer, Cham, 2018.
- [9] R. Carmona and M. Laurière. Convergence analysis of machine learning algorithms for the numerical solution of mean-field control and games II-the finite horizon case. *Annals of Applied Probability*, 32(6), 2022.
- [10] R. A. Carmona, J-P Fouque, and L-H Sun. Mean field games and systemic risk. *Communications in Mathematical Sciences*, 13(4):911–933, 2015.
- [11] F. Coppini, A. De Crescenzo, and H. Pham. Nonlinear graphon mean-field systems. *Stochastic Processes and their Applications*, 190, 2025.
- [12] C. Cuchiero, M. Larsson, and S. Svaluto-Ferro. Probability measure-valued polynomials. *Electronic Journal of Probability*, 24:1–32, 2019.
- [13] A. De Crescenzo, F. de Feo, and H. Pham. Linear-quadratic optimal control for non-exchangeable mean-field sdes and applications to systemic risk. *ESAIM COCV*, 2026.
- [14] A. De Crescenzo, M. Fuhrman, I. Kharroubi, and H. Pham. Mean-field control of non exchangeable systems. *ESAIM COCV*, 32(3), 2025.
- [15] F. de Feo and S. Mekkaoui. Optimal control of heterogeneous mean-field stochastic differential equations with common noise and applications to financial models. *arXiv preprint arXiv:2511.18636*, 2025.
- [16] W. E, J. Han, and A. Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Communications in Mathematics and Statistics*, 5(4):349–380, 2017.
- [17] M. Germain, M. Laurière, H. Pham, and X. Warin. DeepSets and their derivative networks for solving symmetric pdes. *Journal of Scientific Computing*, 91, 2022.
- [18] X. Guo, H. Pham, and X. Wei. Itô formula for flows of measures on semimartingales. *Stochastic Processes and their Applications*, 159:350–390, 2023.
- [19] J. Han and W. E. Deep learning approximation for stochastic control problems. *NIPS*, 2016.
- [20] P.-E. Jabin, D. Poyato, and J. Soler. Mean-field limit of non-exchangeable systems. *Communications on Pure and Applied Mathematics*, 78(4):651–741, 2025.

- [21] O. Kallenberg. *Foundations of Modern Probability*. Springer, 2 edition, 2002.
- [22] I. Kharroubi, S. Mekkaoui, and H. Pham. Stochastic maximum principle for optimal control problem of non exchangeable mean field systems, 2025.
- [23] D. Lacker and A. Soret. A label-state formulation of stochastic graphon games and approximate equilibria on large networks. *Mathematics of Operations Research*, 48(4):1987–2018, 2023.
- [24] S. Lanthaler, S. Mishra, and G. Karniadakis. Error estimates for deepnets: A deep learning framework in infinite dimensions. *Transactions of Mathematics and its Applications*, 6(1):tnac001, 2022.
- [25] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Sturat, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv:2003.03485*, 2020.
- [26] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. KAN: Kolmogorov-Arnold networks. *arXiv preprint arXiv:2404.19756*, 2024.
- [27] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis. Learning nonlinear operators via deepnet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3):218–229, 2021.
- [28] S. Mekkaoui and H. Pham. Analysis of non-exchangeable mean field markov decision processes with common noise: From Bellman equation to quantitative propagation of chaos. *Hal-0550160*, 2026.
- [29] H. Pham and X. Warin. Mean-field neural networks: learning mappings on Wasserstein space. *Neural Networks*, 168:380–393, 2023.
- [30] H. Pham and X. Warin. Mean-field neural networks-based algorithms for McKean-Vlasov control problems. *Journal of Machine Learning*, 3:176–214, 2024.
- [31] C. Reisinger, W. Stockinger, and Y. Zhang. A fast iterative PDE-based algorithm for feedback controls of nonsmooth mean-field control problems. *SIAM Journal of Scientific Computing*, 46(4), 2024.
- [32] L. Ruthotto, S. Osher, L. Nurbekyan, and S. Fung. A machine learning framework for solving high-dimensional mean field game and mean field control problems. *Proceedings of the National Academy of Sciences*, pages 9183–9193, 2020.
- [33] Xavier Warin. P1-Kan: an effective Kolmogorov-Arnold network with application to hydraulic valley optimization. *arXiv preprint arXiv:2410.03801*, 2024.
- [34] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. Smola. FDeepSets. *Advances in Neural Information Processing*, 2017.