University of California
Santa Barbara

Reinforcement Learning for Mean Field Games and Mean Field Control problems

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Statistics and Applied Probability

by

Andrea Angiuli

Committee in charge:

Professor Jean-Pierre Fouque, Chair
Professor Michael Ludkovski
Professor Alex Shkolnik

September 2021

The Dissertation of Andrea Angiuli is approved.

Professor Michael Ludkovski

Professor Alex Shkolnik

Professor Jean-Pierre Fouque, Committee Chair

September 2021

Reinforcement Learning for Mean Field Games and Mean Field Control problems

To Her

Acknowledgements

The PhD program in the Statistic and Applied Probability Department at UCSB has been a truly fulfilling experience for which I will always be in debt with my Supervisor, Prof. Jean-Pierre Fouque, and all the exceptional Faculty members of this excellent research institution.

Prof. Jean-Pierre Fouque has been an extraordinary role model in this challenging and beautiful journey. His influence on my studies started during a seminar in Mean Field Games when I was a student at the University of Milan. The passion and pride shown for his results and collaborators were tremendously inspiring. Having the opportunity to be in his classes and enjoy his unique ability of picturing in the students' minds mathematical abstract ideas has been an extremely valuable gift for which I am deeply grateful. His supervision has been a continuous opportunity to grow on the technical and personal levels. His borderless knowledge as a scientist, his fairness and empathy as a mentor supported me to reach results I would have never imagined.

I would like to thank Prof. Mathieu Laurière for being always available for endless meetings crucial for the success of our projects, but most importantly for being a friendly support during this long journey.

I would like to thank Prof. Enrico Janelli from the University of Bari (Italy) for teaching me the endless beauty of Real and Complex Analyses. He was an enlightening and unforgettable mind. His love for the Mathematics was extremely contagious and I hope to be able to bring it in my own path.

I would like to thank Prof. Raya Feldman and Prof. Sreenivasa Rao Jammalamadaka for their excellent classes on my first year of program, for being extremely welcoming in this new reality and allowing me to build strong bases to proceed in my research.

I would like to thank Prof. Mike Ludkovski for being a point of reference of this

Department, for being a great instructor and for the discussions and challenges he offered me during these years. I would like to express my gratitude to him and Prof. Shkolnik for being part of my committee.

I would like to thank all the amazing friends I found in this Department: Prof. Nils Detering, Prof. Moritz Voss, Prof. Roberto Molinari, Aditya Maheshwari and Brennan Hall. I am grateful for sharing their knowledge together with joyful memories.

I would like to thank Pierre-O for being the perfect friend in a new adventure, for making the Department a cheerful place and being the best partner in crime for so many nights.

Thanks Ségolène for being an amazing friends, for keeping my French up and being always ready to create beautiful memories. Thanks for being such a strong and independent woman.

I would like to thank Maddalena, the person who trusted in me since my application process. I am thankful to her for showing me her academic rigor and ability to overcome the challenges that research brings. She has always been my source of peace and love in the light and dark moments. Thanks for being always ready to share joy and smiles.

I would like to thank Tina for showing me a new and beautiful version of Santa Barbara, welcoming me in her community, and reminding me that you can find people thousands of miles from home with whom the language or the origins are not barriers to feel existentially bounded. You made this place a new home.

I would like to thank my family for the education and support they provided me. I am lifelong grateful to have received the opportunities and tools to independently find my own path feeling free to express myself through my passions, my errors and my achievements.

Curriculum Vitæ
Andrea Angiuli

Education

2021	Ph.D. in Statistics and Applied Probability (Expected), University of California, Santa Barbara.
2019	M.A. in Statistics, University of California, Santa Barbara.
2015	M.Sc. in Applied Mathematics, University of Milan, Italy.
2011	B.Sc. in Pure Mathematics, University of Bari, Italy.

Professional Experiences

- Quant Research Intern, Bloomberg L.P., New York City, Summer 2021
  Calibration of path dependent volatility models to SPX and VIX smiles
- Research scientist Intern, Pime Machine Learning Team, Amazon, Seattle, Summer 2020
  Development of unsupervised selection techniques and variational auto-encoders for market segmentation

Accademic Experiences

- Graduate Research Fellow, Institute of Pure and Applied Mathematics - University of California, Los Angeles, Spring 2020
  Workshops on theory and numerical methods for High Dimensional Hamilton-Jacobi Partial Differential Equations
- Team Leader, Financial Mathematics Team Challenge, Rio de Janeiro, Brazil (FMTC-BR), Summers 2019, 2018
  2019 Project on order book modeling for the Brazilian equity market,
  2018 Project on developing innovative calibration techniques for interest rate models.
- Graduate Researcher, Centre International de Rencontres Mathématiques, Marseille, France, Summer 2017
  Development of numerical methods to solve mean field games.
- Research Intern, CMAP, École Polytechnique, Paris, France, Spring-Summer 2014
  Project on stochastic gradient descent methods to solve backward stochastic differential equations for pricing derivatives.

Working paper

- A. Angiuli, R. Hu, *Deep Reinforcement Learning for Mean Field Games and Mean Field Control Problems in Continuous Spaces*, 2021

Publications

- A. Angiuli, J.P. Fouque, M. Laurière, *Reinforcement Learning for Mean Field Games, with Applications to Economics*, to appear in Machine Learning in Financial Markets: A Guide to Contemporary Practice edited by Agostino Capponi, Charles-Albert Lehalle; Cambridge University Press , 2021, https://arxiv.org/pdf/2106.13755.pdf
- A. Angiuli, J.P. Fouque, M. Laurière, *Unified reinforcement Q-learning for mean field game and control problems*, to appear in Mathematics of Control, Signals, and Systems (MCSS), 2021, https://arxiv.org/pdf/2006.13912.pdf
- A. Angiuli, C. V. Graves, H. Li, J.-F. Chassagneux, F. Delarue, R. Carmona, *Numerical Probabilistic Approach to MFG, ESAIM: Proceedings and Surveys*, 2017, https://www.esaim-proc.org/articles/proc/pdf/2019/01/proc196505.pdf

Scientific Notes

- A. Angiuli, C. Antunes, A. De Genaro, M. R. Moresco, C. Paolucci, Modeling order book dynamics, FMTC-BR, 2019
- A. Angiuli, C. Antunes, T. A. McWalter, C. Paolucci, A. Sombra, *A calibration of the Lognormal Forward LIBOR Model*, FMTC-BR event webpage, 2018, https://emap.fgv.br/sites/emap.fgv.br/files/u77/report_2018.pdf

Abstract

Reinforcement Learning for Mean Field Games and Mean Field Control problems

by

Andrea Angiuli

In this manuscript, we develop reinforcement learning theory and algorithms for differential games with large number of homogenous players, focusing on applications in finance/economics.

Stochastic differential games are notorious for their tractability barrier in computing Nash equilibria (social optima) in the competitive (resp. cooperative) framework. Our work aims to overcome this limitation by merging mean field theory, reinforcement learning and multi-scale stochastic approximation.

In recent years, the question of learning in MFG and MFC has garnered interest, both as a way to compute solutions and as a way to model how large populations of learners converge to an equilibrium. Of particular interest is the setting where the agents do not know the model, which leads to the development of reinforcement learning (RL) methods.

After reviewing the literature on this topic, we introduce a new definition of asymptotic mean field games and mean field control problems which naturally connects with the RL framework. We unify these problems through a two-timescale approach and develop a Q-learning based solving scheme in the case of finite spaces. Our first proposed algorithm learns either the MFG or the MFC solution depending on the choice of parameters. To illustrate this method, we apply it to an infinite horizon linear quadratic example. We discuss convergence results based on stochastic approximation theory.

This approach is extended to the case of the interaction through the distribution of the controls of the population and finite horizon. The second algorithm is tested on two examples from economic/finance: a mean field problem of accumulated consumption with HARA utility function, and a trader's optimal liquidation problem. The heterogeneity of the chosen examples shows the flexibility of our approach.

We conclude by presenting our on-going work on solving problems in continuous spaces. We present our Unified 3-scale Actor Critic algorithm based on three learning rules. The first two refer to the optimal strategy (the actor) and the value function (the critic). An additional learning rule is adopted to target the distribution of the population at equilibrium. This method is tested on two examples of the infinite horizon case.

Contents

Chapter 1

Introduction

Dynamic games with many players are pervasive in today's highly connected world. In many models the agents are indistinguishable since they have the same dynamics and cost functions. Moreover, the interactions are often anonymous since each player is influenced only by the empirical distribution of all the agents. However, such games become intractable when the number of agents becomes very large. Mean field games have been introduced independently by Lasry and Lions [55], and Huang, Malhamé and Caines [52] to tackle such situations by passing to the limit and considering games with an infinite number of players interacting through the population distribution. The solution of the limiting problem represents an approximation of the $N-$ player game and their connection is formalized by the " master equation", a partial differential equation stated on the space of probability measures (see Cardaliaguet et al. [19] for further details). Although the standard formulation of MFG focuses on finding Nash equilibria, social optima arising in a cooperative setting have also been studied under the term of mean field control by Bensoussan et al. [11] or control of McKean-Vlasov dynamics by Lasry and Lions [55]. Equilibria or social optima in such games can be characterized in a tractable way through forward-backward systems of partial differential equations (PDE) or stochastic differential equations (SDE) [24, 55].

In this work, we propose algorithms to solve mean field problems using ideas from Reinforcement Learning (RL). Reinforcement learning (RL) is a branch of machine learning (ML) which studies the interactions of an agent within an environment in order to maximize a reward signal. Applications of RL in economics and finance have recently attracted a lot of interest, see *e.g.* [32]. However, since our problems involve mean-field interactions, the population distribution requires a special treatment. In our setup, the agent is feeding an action to the environment which produces the next state and a reward (or cost). The environment also updates in an automatic way (without decision) the distributions of states and controls (see the diagram in Figure 4.1). The environment can be viewed as a "black box" or as a "simulator" depending on the problem, but, in any case, it generates the new state if the dynamics is unknown and the reward if not computable by the agent. It is also interesting to note that even in cases where the dynamics and the reward structure are known but complicated, then our algorithms can be viewed as a numerical method for computing the optimal strategy for the corresponding MFG or MFC problems.

Since the introduction of MFG theory, several numerical methods have been proposed, see *e.g.* [2, 56] and the references therein. In our paper [8], we detail two methods based on the probabilistic approach and apply them to five benchmark problems. Both are based on a Picard scheme; importantly, we combine each of them with a generic continuation method that permits to extend the time horizon (or equivalently the coupling strength between the two equations) for which the Picard iteration converges.

Recently, several methods to solve MFGs based on machine learning tools have been proposed relying either on the probabilistic approach [36, 27, 38, 62] or the analytical approach [3, 28, 69, 18, 59, 56]. They combine neural network approximations and stochastic optimization techniques to solve McKean-Vlasov control problems, mean field

FBSDE or mean field PDE systems; see Carmona and Laurière [22] for a recent survey and applications to finance. These methods are based on the knowledge of the model, but the question of learning solutions to MFG and MFC without full knowledge of the model have also attracted a surge of interest.

As far as learning methods for mean field problems are concerned, most works focus either on MFG or on MFC. Yang et al. [77] use a mean field approximation in the context of multi-agent reinforcement learning (MARL) to reduce the computational cost. Yang et al. [76] use inverse reinforcement learning to learn the dynamics of a mean field game on a graph. To approximate stationary MFG solutions, Guo et al. [47] use fixed point iterations on the distribution combined with Q-learning to learn the best response at each iteration. Anahtarci et al. [5] combine this kind of learning scheme together with an entropic regularization. Convergence of an actor-critic method for linear-quadratic MFG has been studied in [37]. Model-free learning for finite horizon MFG has been studied by Mishra et al. in [63] using a backward scheme. Fictitious play without or with reinforcement learning has been studied respectively in [20, 49] and [34, 67, 75], or online mirror descent [48, 65]. These iterative methods have been proved to converge under a monotonicity condition which is weaker than the strict contraction property used to ensure convergence of fixed point iterations. They can be extended to continuous space problems using deep reinforcement learning as *e.g.* in [66]. A two timescale approach to solve MFG with finite state and action spaces has been proposed in [61, 70].

To learn MFC optima, Mahajan and Subramanian [70] design a gradient based algorithm. Model-free policy gradient method has been proved to converge for linear-quadratic problems in [29, 72], whereas Q-learning for a "lifted" Markov decision process on the space of distributions has been studied in [30, 44, 46]. Optimality conditions and propagation of chaos type result for mean field Markov decision processes are studied by Motte and Pham in [64].

In [6], we proposed a unified two timescale Q-learning algorithm to solve both MFG and MFC problems in an infinite horizon stationary regime. The key idea is to iteratively update estimates of the distribution and the Q-function with different learning rates. Suitably choosing these learning rates enables the algorithm to learn the solution of the MFG or the one of the MFC. A slow updating of the distribution of the state leads to the Nash equilibrium of the competitive MFG and the algorithm learns the corresponding optimal strategy. A rapid updating of the distribution leads to learning of the optimal control of the corresponding cooperative MFC. Moreover, in contrast with other approaches, our algorithm does not require the environment to output the population distribution which means that a single agent can learn the solution of mean field problems.

In [7], we extended this algorithm in two directions: finite horizon setting, and "extended" mean field problems which involve the distribution of controls as well. That demonstrates the flexibility of our two timescale algorithm and broadens the range of applications.

In the on-going work [9], we merge our solving paradigm with deep reinforcement learning to solve mean field problems in continuous state and action spaces. The proposed Unified 3-scale Mean Field Actor Critic algorithm (U3-MF-AC) inherits two learning rules from the classical Actor Critic approach studied in [17]. They refer respectively to the optimal strategy (the actor) and the value function (the critic). Additionally, the distribution of the population at equilibrium is learned at a different schedule. The choice of the three scales is crucial in defining the convergence of this method to the solution of a MFG or MFC problem.

The rest of the dissertation is organized as follows. In Chapter 2, we introduce the framework of classical Reinforcement Learning. The definition of a Markov Decision Process (MDP) is recalled together with the $Q-$learning algorithm, one of the most popular model free approach to solve it. In Chapter 3, we propose a new definition of infinite horizon mean field problems in discrete time and space: the Asymptotic MFG and MFC problems. Comparison with classical (non-asymptotic) and stationary formulations are also made. In Chapters 4 and 5, we discuss our results from [6] and [7]. We present how we connected reinforcement learning and mean field problems through a multi-scale stochastic viewpoint. We illustrate our algorithms on a general linear quadratic model and two examples from economics: a mean field accumulation problem in Section 5.4 and an optimal execution problem for a mean field of traders in Section 5.5. In Chapter 6, we show our current direction of research through some preliminary results on deep reinforcement learning for mean field games. We then conclude in Chapter 7.

Chapter 2

Reinforcement Learning

A Markov Decision Process (MDP) formalizes the sequential making decision problem of an agent interacting with an environment. At each discrete time n, given a state space $\mathcal{X}$ and an action space $\mathcal{A}$, the agent observes the state of the environment $X_n \in \mathcal{X}$ and chooses an action $A_n \in \mathcal{A}$. Due to the agent's action, the environment evolves to a state $X_{n+1} \in \mathcal{X}$ and assigns a reward $r_{n+1} = r(X_n, A_n)$. The goal of the agent is to find the optimal strategy π which assigns to each state of the environment the optimal action in order to maximize the aggregate discounted rewards.

RL aims to solve MDPs without assuming any (or partial) knowledge of the dynamics of the environment and the reward structure. In orther to do that, RL algorithms are based on trials and errors. A complete overview on the evolution of this field is given in [71].

2.1 Learning the optimal action value function

The Q-learning method was introduced in [73] to solve a discrete time MDP with finite state and action spaces. It is based on the evaluation of the optimal action-value table, $Q^*(x,a)$, which represents the maximum expected aggregate discounted rewards when starting in state x and choosing the first action a, i.e.

$$\begin{aligned} Q^*(x,a) &= \max_{\pi} Q^{\pi}(x,a) \\ &= \max_{\pi} \mathbb{E}\left[r(X_0,A_0) + \sum_{n=1}^{\infty} \gamma^n r(X_n,\pi(X_n)) \,\middle|\, X_0 = x, A_0 = a \right], \end{aligned} \tag{2.1}$$

where $\gamma \in (0,1)$ is a discounting factor, and $X_{n+1} = b(X_n, \pi(X_n))$. The maximum is taken over strategies (or policies) π, which are functions of the state taking values in the action space $\mathcal{A}$. Intuitively, the Q^* function quantifies the optimal reward-to-go of an agent starting at x, using action a for the first step and then acting optimally afterwards. In other words, the value of $Q^*(x,a)$ is the reward of using a when in state x, plus the maximal reward possible after that, i.e. the reward induced by using the optimal control. Since the state's dynamics b (and sometimes the reward function r) are unknown to the agent, the algorithm is characterized by the trade-off between exploration of the environment and exploitation of the current available information. This is typically accomplished by the implementation of an ϵ-greedy policy. The greedy action which maximizes the immediate reward is chosen with probability $1-\epsilon$ and a random action otherwise, i.e.

$$\pi^{\epsilon}(x) = \begin{cases} a \in Unif(\mathcal{A}), & \text{with probability} \quad \epsilon, \\ a^* = \arg\max_{a \in A} Q^*(x,a), & \text{with probability} \quad 1-\epsilon. \end{cases} \tag{2.2}$$

Note that this is the randomized policy which will be used in the algorithm presented in Chapter 4, but as the optimal strategies will turn out to be deterministic (as ϵ goes to zero over learning episodes), in the following, we present the problems and the Q-learning approach only using deterministic policies called controls and denoted by α instead of π (see [64] for additional details on randomized policies).

The state value function with respect to a deterministic control function α is given by

$$V^\alpha(x) = \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n r(X_n, \alpha(X_n)) \,\Big|\, X_0 = x\right].$$

One of the main advantages of computing the optimal action-value function instead of the optimal state value function is that from the former, one can directly recover the optimal control, given by $\arg\max_a Q^*(x,a)$. This is particularly important in order to design model-free method.

Q-learning [73] is one of the most popular model-free methods in RL for discrete time, discrete and finite state/action spaces. In order to introduce the algorithm, we review some of the classical results relative to the value function V^α and the corresponding action value function Q^α.

Lemma 1 *Let $\alpha : \mathcal{X} \to \mathcal{A}$ be a deterministic policy. The state value function $V^\alpha : \mathcal{X} \to \mathbb{R}$ can be derived from the action state value function $Q^\alpha : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ as*

$$V^\alpha(x) = Q^\alpha(x, \alpha(x)). \tag{2.3}$$

Proof:

$$\begin{aligned} V^\alpha(x) &= \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n r(X_n, \alpha(X_n)) \Big| X_0 = x\right] \\ &= \mathbb{E}\left[r(X_0, A_0) + \sum_{n=1}^{\infty} \gamma^n r(X_n, \alpha(X_n)) \Big| X_0, A_0 = \alpha(X_0)\right] = Q(x, \alpha(x)) \end{aligned}$$

■

Lemma 2 *(Bellman equation Q^α) The action state value function $Q^\alpha : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ satisfies the Bellman equation given by*

$$Q^\alpha(x,a) = r(x,a) + \gamma \mathbb{E}\left[Q^\alpha(X_1, \alpha(X_1)) \big| X_0 = x, A_0 = a\right] \tag{2.4}$$

Proof:

$$\begin{aligned}
Q^{\alpha}(x,a) &= \mathbb{E}\left[r(X_0,A_0)+\sum_{n=1}^{\infty}\gamma^n r(X_n,\alpha(X_n))\Big|X_0=x,A_0=a\right]\\
&\overset{\text{(TP)}}{=} r(x,a)+\gamma\mathbb{E}\left[\mathbb{E}\left[\sum_{n=1}^{\infty}\gamma^{n-1}r(X_n,\alpha(X_n))\Big|X_0=x,A_0=a,X_1\right]\Big|X_0=x,A_0=a\right]\\
&\overset{\text{(MP)}}{=} r(x,a)+\gamma\mathbb{E}\left[\mathbb{E}\left[\sum_{n=1}^{\infty}\gamma^{n-1}r(X_n,\alpha(X_n))\Big|X_1\right]\Big|X_0=x,A_0=a\right]\\
&= r(x,a)+\gamma\mathbb{E}\left[V^{\alpha}(X_1)\Big|X_0=x,A_0=a\right]\\
&\overset{(2.3)}{=} r(x,a)+\gamma\mathbb{E}\left[Q^{\alpha}(X_1,\alpha(X_1))\big|X_0=x,A_0=a\right],
\end{aligned}$$

where TP and MP stands for tower and Markov property respectively. ■

Proposition 1 *(Policy improvement) Let $\alpha_1 : \mathcal{X} \to \mathcal{A}$ and $\alpha_2 : \mathcal{X} \to \mathcal{A}$ be two deterministic policies such that*

$$V^{\alpha_1}(x) \leqslant Q^{\alpha_1}(x,\alpha_2(x)), \quad \forall x \in \mathcal{X}. \tag{2.5}$$

Then, the policy α_2 must be as good or better than the policy α_1, that is

$$V^{\alpha_1}(x) \leqslant V^{\alpha_2}(x), \quad \forall x \in \mathcal{X}.$$

Proof:

$$\begin{aligned}
V^{\alpha_1}(x) \leqslant & Q^{\alpha_1}(x, \alpha_2(x)) \\
& \overset{(2.4)}{=} r(x, \alpha_2(x)) + \gamma \mathbb{E}\left[Q^{\alpha_1}(X_1, \alpha(X_1))\Big| X_0 = x, A_0 = \alpha_2(x)\right] \\
& \overset{(2.3)}{=} r(x, \alpha_2(x)) + \gamma \mathbb{E}\left[V^{\alpha_1}(X_1)\big| X_0 = x, A_0 = \alpha_2(x)\right] \\
& \overset{(2.5)}{\leqslant} r(x, \alpha_2(x)) + \gamma \mathbb{E}\left[Q^{\alpha_1}(X_1, \alpha_2(X_1)\big| X_0 = x, A_0 = \alpha_2(x)\right] \\
& \overset{(2.4)}{=} r(x, \alpha_2(x)) + \gamma \mathbb{E}\left[r(X_1, \alpha_2(X_1)) + \gamma Q^{\alpha_1}(X_2, \alpha_2(X_2)\big| X_0 = x, A_0 = \alpha_2(x)\right] \\
& \vdots \\
& \leqslant \mathbb{E}\left[\sum_{n=0}^{k} \gamma^n r(X_n, \alpha_2(X_n)) + \gamma^{k+1} V^{\alpha_1}(X_{k+1})\big| X_0 = x, A_0 = \alpha_2(x)\right]
\end{aligned}$$

By taking the limit $k \to \infty$ follows

$$V^{\alpha_1}(x) \leqslant \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n r(X_n, \alpha_2(X_n))\Big| X_0 = x\right] = V^{\alpha_2}(x)$$

■

Corollary 1 *The optimal value function $V^* : \mathcal{X} \to \mathbb{R}$ can be derived by $Q^* : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ as*

$$V^*(x) = \max_a Q^*(x, a), \quad x \in \mathcal{X}. \tag{2.6}$$

Proof: Let α be a deterministic policy and Q^α its corresponding action value function. The policy α_1 derived by Q^α as

$$\alpha_1(x) = \arg\max_a Q^\alpha(x, a), \quad \forall x \in \mathcal{X}$$

satisfies

$$Q^\alpha(x, \alpha_1(x)) = \max_a Q^\alpha(x, a) \geqslant V^\alpha(x), \quad \forall x \in \mathcal{X}.$$

Then, it follows by the the policy improvement theorem that

$$V^{\alpha_1}(x) \geqslant V^\alpha(x), \quad \forall x \in \mathcal{X}.$$

The improvement step can be repeated only a finite number of times given that $\mathcal{X}$ and $\mathcal{A}$ are finite, discrete spaces, i.e. $\exists N > 0$ such that $\alpha^*(x) = \alpha_N(x) = \arg\max_a Q^{\alpha_N}(x,a)$ and

$$\max_a Q^*(x) = Q^*(x, \alpha^*(x)) = V^*(x), \quad \forall x \in \mathcal{X},$$

where Q^*, V^* corresponds to the value functions with respect to α^*. Might α^* be still sub-optimal? No, according to the Optimality Theorem from Bellman and Dreyfus, [10].

■

Theorem 1 *(Bellman equation Q^*) The optimal action value function $Q^* : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ satisfies the Bellman equation given by*

$$Q^*(x,a) = r(x,a) + \gamma \mathbb{E}\left[\max_{a'} Q^*(X_1, a') \middle| X_0 = x, A_0 = a\right] \tag{2.7}$$

Proof:

$$\begin{aligned} Q^*(x,a) = &\overset{(2.4)}{=} r(x,a) + \gamma \mathbb{E}\left[Q^*(X_1, \alpha^*(X_1)) \middle| X_0 = x, A_0 = a\right] \\ &\overset{(2.3)}{=} r(x,a) + \gamma \mathbb{E}\left[V^*(X_1) \middle| X_0 = x, A_0 = a\right] \\ &\overset{(2.6)}{=} r(x,a) + \gamma \mathbb{E}\left[\max_{a'} Q^*(X_1, a') \middle| X_0 = x, A_0 = a\right] \end{aligned}$$

■

The function Q^* is the unique solution of equation (2.7) which is of the type $Q^* = G(Q^*)$ for a function $G : \mathbb{R}^{|\mathcal{X}| \times |\mathcal{A}|} \to \mathbb{R}$ satisfying

$$\|G(Q) - G(Q')\|_\infty \leqslant \gamma \|Q - Q'\|_\infty.$$

In particular, the sequence $Q^{n+1} = G(Q^n)$ for $n \geqslant 0$ converges to Q^* at exponential rate (see [15], Chapter 10.3 for more details).

In the stochastic approximation version of equation (2.7), the conditional expectation is replaced by the evaluation at the random variable $X_{\cdot}$ and it is weighted with a small learning rate β_n, i.e.

$$Q_{n+1}(x,a) = Q_n(x,a) + \beta_n \mathbb{1}_{X_n,A_n}(x,a)\left[r(x,a) + \gamma \max_{a'} Q(X_{n+1},a') - Q_n(x,a)\right] \quad (2.8)$$

Equation (2.8) represents the update rule of the Q-learning algorithm. Its convergence can be analyzed by studying the limiting O.D.E.

$$\dot{q}(t) = \Lambda(t)\left(G(q(t)) - q(t)\right) \quad (2.9)$$

where $\Lambda(t)$ is a diagonal matrix with a probability vector along its diagonal. Under suitable assumptions, the O.D.E. (2.9) and the corresponding sequence $(Q_n)_{n\geqslant 0}$ converge to Q^*. In particular, a requirement necessary for convergence is that each pair (x,a) has a positive probability to be visited, e.g. applying an $\epsilon-$greedy policy. Algorithm 1 describes the practical implementation of Q-learning.

Algorithm 1 Q-learning

```
Require: $\mathcal{X} = \{x_0, \dots, x_{|\mathcal{X}|-1}\}$ : finite state space,
    $\mathcal{A} = \{a_0, \dots, a_{|\mathcal{A}|-1}\}$ : finite action space,
    $\epsilon$ : parameter related to the $\epsilon-$greedy policy,
    $(\beta_n)_{n \geqslant 0}$: learning rates schedule,
    $tol_Q$ : break rule tolerance.
Initialization: $Q^0(x,a) = 0$ for all $(x,a) \in \mathcal{X} \times \mathcal{A}$
for step $n = 1, 2, \dots$ do
    Observe state $X_n$ provided by the environment
    Choose action $A_n$ using the $\epsilon$-greedy policy derived from $Q_n(X_n, \cdot)$
    Observe reward $r_{n+1} = r(X_n, A_n)$ and next state $X_{n+1}$ provided by the environment
    Update $Q$:
    $Q_{n+1}(x,a) = Q_n(x,a) + \beta_n \mathbb{1}_{x,a}(X_n, A_n)[r_{n+1} + \gamma \max_{a' \in \mathcal{A}} Q_n(X_{n+1}, a') - Q_n(x,a)]$
end for
if $\|Q_n - Q_{n-1}\|_\infty < tol_Q$ then
    break
end if
return $\tilde{Q}^*$
```

In the next chapters, we will discuss the results presented in our works [6] and [7]. In particular, we will show how the Q-learning algorithm 1 can be redesigned as a two-timescale stochastic approximation scheme able to solve mean field problems.

Chapter 3

Mean field Games and Mean Field Control problems

Mean field games are the result of the application of mean field techniques from physics into game theory. The mean field interaction is introduced to describe the behavior of a large number N of indistinguishable players with symmetric interactions. The complexity of the system would be intractable if we were to describe all the pairwise interactions. A solution to this problem is given by describing the interactions of each player i with the empirical distribution of the other players. As the number of players increases, the impact of each of them on the empirical distribution decreases. By the principle of propagation of chaos (law of large numbers) each player becomes asymptotically independent from the others and its interaction is with its own distribution making the statistical structure of the system simpler. Two types of mean field problems can be distinguished between a mean field game and a mean field control depending on the goal the agents try to achieve. The aim of a mean field game is to find an equivalent of a Nash equilibrium in a non-cooperative N-player game when the number of players becomes large. On the other hand, a mean field control problem analyzes the social optimum in a cooperative game within a large population. In this case, all the agents follow the same strategy provided by a central planner. Since the seminal works [55], and [52, 51], the research in mean field game theory attracted a huge interest. We refer to the extensive works [24], and [12] for further details.

We start by presenting three formulations of MFG and MFC problems: non-asymptotic, asymptotic, and stationary. All these problems are on an infinite horizon and for the sake of consistency with the RL literature, we present them in a discrete time and space framework. We will however resort to continuous time and space models in Chapter 4 in order to obtain simple benchmarks. Note that, as customary in the MFG literature, without loss of generality, we minimize a cost instead of maximizing a reward.

Let $\mathcal{X}$ and $\mathcal{A}$ be finite sets corresponding to states and actions. We denote by $\Delta^{|\mathcal{X}|}$ the simplex in dimension $|\mathcal{X}|$, which we identify with the space of probability measures on $\mathcal{X}$. Let $p : \mathcal{X} \times \mathcal{A} \times \Delta^{|\mathcal{X}|} \to \Delta^{|\mathcal{X}|}$ be a transition kernel. We will sometimes view it as a function:

$$p : \mathcal{X} \times \mathcal{X} \times \mathcal{A} \times \Delta^{|\mathcal{X}|} \to [0, 1], \qquad (x, x', a, \mu) \mapsto p(x'|x, a, \mu),$$

which will be interpreted as the probability, at any given time step, to jump to state x' when starting from state x and using action a and when the population distribution is μ.

Let $f : \mathcal{X} \times \mathcal{A} \times \Delta^{|\mathcal{X}|} \to \mathbb{R}$ be a running cost function. We interpret $f(x, a, \mu)$ as the one-step cost, at any given time step, incurred to a representative agent who is at state x and uses action a while the population distribution is μ. For a random variable X, we denote its law by $\mathcal{L}(X)$. We will focus on feedback controls, i.e., functions of the state of the agent and possibly of time.

3.1 Non-asymptotic formulations

In the usual formulation for time-dependent MFG and MFC, the interactions between the players are through the distribution of states at the current time. More precisely, in a MFG, one typically looks for $(\hat{\alpha}, \hat{\boldsymbol{\mu}})$ where $\hat{\alpha} : \mathbb{N} \times \mathcal{X} \to \mathcal{A}$ and $\hat{\boldsymbol{\mu}} = (\hat{\mu}_n)_{n \geqslant 0} \in (\Delta^{|\mathcal{X}|})^{\mathbb{N}}$ is a flow of probability distributions on $\mathcal{X}$, such that the following two conditions hold:

1. Optimality of the best response map: $\hat{\alpha}$ is the minimizer of

$$\alpha \mapsto J^{MFG}(\alpha; \hat{\boldsymbol{\mu}}) = \mathbb{E}\left[\sum_{n=0}^{+\infty} \gamma^n f(X_n^{\alpha,\hat{\boldsymbol{\mu}}}, \alpha_n(X_n^{\alpha,\hat{\boldsymbol{\mu}}}), \hat{\mu}_n)\right],$$

where $\alpha_n(\cdot) := \alpha(n, \cdot)$ and the process $X^{\alpha,\hat{\boldsymbol{\mu}}}$ follows the dynamics given by:

$$X_{n+1}^{\alpha,\hat{\boldsymbol{\mu}}} \sim p\left(\cdot | X_n^{\alpha,\hat{\boldsymbol{\mu}}}, \alpha_n(X_n^{\alpha,\hat{\boldsymbol{\mu}}}), \hat{\mu}_n\right)$$

with initial distribution $X_0^{\alpha,\hat{\boldsymbol{\mu}}} \sim \mu_0$;

2. Fixed point condition: $\hat{\mu}_n = \mathcal{L}(X_n^{\hat{\alpha},\hat{\boldsymbol{\mu}}})$ for every $n \geqslant 0$.

In a MFC problem, the goal is to find α^* such that the following condition holds: α^* is the minimizer of

$$\alpha \mapsto J^{MFC}(\alpha) = \mathbb{E}\left[\sum_{n=0}^{+\infty} \gamma^n f(X_n^{\alpha}, \alpha_n(X_n^{\alpha}), \mathcal{L}(X_n^{\alpha}))\right],$$

where the process X^{α} follows the dynamics:

$$X_{n+1}^{\alpha} \sim p\left(\cdot | X_n^{\alpha}, \alpha_n(X_n^{\alpha}), \mathcal{L}(X_n^{\alpha})\right)$$

with initial distribution $X_0^{\alpha} \sim \mu_0$. Note that p is the same transition probability function as for the MFG above but we plug the law $\mathcal{L}(X_n^{\alpha})$ of X_n^{α} instead of a given distribution $\hat{\mu}_n$. In other words, the MFC problem is of McKean-Vlasov (MKV) type.

We will sometimes use the notation $\boldsymbol{\mu}^* = \boldsymbol{\mu}^{\alpha^*}$ for the optimal distribution in the MFC. Note that the objective function in the MFC setting can be written in terms of the objective function in the MFG as:

$$J^{MFC}(\alpha) = J^{MFG}(\alpha; \boldsymbol{\mu}^{\alpha}),$$

where $\mu_n^{\alpha} = \mathcal{L}(X_n^{\alpha})$ for all $n \geqslant 0$. However, in general,

$$J^{MFC}(\alpha^*) = J^{MFG}(\alpha^*; \boldsymbol{\mu}^*) \neq J^{MFG}(\hat{\alpha}; \hat{\boldsymbol{\mu}}).$$

In these two problems, the equilibrium control $\hat{\alpha}$ or the optimal control α^* usually depend on time due to the dependence of p and f on the mean field flow, which evolves with time.

Although these are the usual formulations of MFG and MFC problems, in order to draw connections with reinforcement learning more directly, we turn our attention to formulations in which the control is independent of time. That is naturally the case in some applications, and, roughly speaking, it is also in the spirit of an individual player trying to optimally join a crowd of players already in the long-time asymptotic equilibrium. This will be made more precise in the following section.

3.2 Asymptotic formulations

We consider the following MFG problem: Find $(\hat{\alpha}, \hat{\mu})$ where $\hat{\alpha} : \mathcal{X} \to \mathcal{A}$ and $\hat{\mu} \in \Delta^{|\mathcal{X}|}$, such that the following two conditions hold:

1. $\hat{\alpha}$ is the minimizer of

$$\alpha \mapsto J^{AMFG}(\alpha; \hat{\mu}) = \mathbb{E}\left[\sum_{n=0}^{+\infty} \gamma^n f(X_n^{\alpha,\hat{\mu}}, \alpha(X_n^{\alpha,\hat{\mu}}), \hat{\mu})\right],$$

where the process $X^{\alpha,\hat{\mu}}$ follows the transitions:

$$X_{n+1}^{\alpha,\hat{\mu}} \sim p\left(\cdot | X_n^{\alpha,\hat{\mu}}, \alpha(X_n^{\alpha,\hat{\mu}}), \hat{\mu}\right)$$

with initial distribution $X_0^{\alpha,\hat{\mu}} \sim \mu_0$;

2. $\hat{\mu} = \lim_{n\to+\infty} \mathcal{L}(X_n^{\hat{\alpha},\hat{\mu}})$.

We stress that in this problem the control is a function of the state only and does not depend on time, as p and f depend only on the limiting distribution but not on time. Intuitively, this problem corresponds to the situation in which an infinitesimal player wants to join a crowd of players who are already in the asymptotic regime (as time goes to infinity). This stationary distribution is a Nash equilibrium if the new player joining the crowd has no interest in deviating from this asymptotic behavior.

We also consider the following MFC problem: Find α^* such that the following condition holds: α^* is the minimizer of

$$\alpha \mapsto J^{AMFC}(\alpha) = \mathbb{E}\left[\sum_{n=0}^{+\infty} \gamma^n f(X_n^\alpha, \alpha(X_n^\alpha), \mu^\alpha)\right],$$

where the process X^α follows the transitions

$$X_{n+1}^\alpha \sim p\left(\cdot|X_n^\alpha, \alpha(X_n^\alpha), \mu^\alpha\right)$$

with initial distribution $X_0^\alpha \sim \mu_0$, and with the notation $\mu^\alpha = \lim_{n\to+\infty} \mathcal{L}(X_n^\alpha)$.

We will sometimes use the shorthand notation $\mu^* = \mu^{\alpha^*}$ for the optimal distribution in the MFC setting. Here too, the control is independent of time, and p and f depend only on the limiting distribution. Intuitively, this problem can be viewed as the one posed to a central planner who wants to find the optimal stationary distribution such that the cost for the society is minimal when a new agent joins the crowd.

Note that in this formulation again, the objective function in the MFC setting can be written in terms of the objective function in the MFG as:

$$J^{AMFC}(\alpha) = J^{AMFG}(\alpha; \mu^\alpha),$$

with the notation $\mu^\alpha = \lim_{n\to+\infty} \mathcal{L}(X_n^\alpha)$.

Remark 1 *Although the AMFG and AMFC problems in this section are defined using an initial distribution μ_0 for the state process, one expects that under suitable conditions, ergodicity in particular, the optimal controls $\hat{\alpha}$ and α^* are independent of this initial distribution.*

3.3 Stationary formulations

Another formulation with controls independent of time consists in looking at the situation in which the new agent joining the crowd starts with a position drawn according to the ergodic distribution of the equilibrium control or the optimal control. This type of problems has been considered e.g. in [47], [70], and can be described as follows.

The stationary MFG problem is to find $(\hat{\alpha}, \hat{\mu})$ where $\hat{\alpha} : \mathcal{X} \to \mathcal{A}$ and $\hat{\mu} \in \Delta^{|\mathcal{X}|}$, such that the following two conditions hold:

1. $\hat{\alpha}$ is the minimizer of

$$\alpha \mapsto J^{SMFG}(\alpha; \hat{\mu}) = \mathbb{E}\left[\sum_{n=0}^{+\infty} \gamma^n f(X_n^{\alpha,\hat{\mu}}, \alpha(X_n^{\alpha,\hat{\mu}}), \hat{\mu})\right],$$

where the process $X^{\alpha,\hat{\mu}}$ follows the SDE

$$X_{n+1}^{\alpha,\hat{\mu}} \sim p\left(\cdot | X_n^{\alpha,\hat{\mu}}, \alpha(X_n^{\alpha,\hat{\mu}}), \hat{\mu}\right),$$

and starts with distribution $X_0^{\alpha,\hat{\mu}} \sim \hat{\mu}$;

2. The process $X^{\hat{\alpha},\hat{\mu}}$ admits $\hat{\mu}$ as invariant distribution (so $\hat{\mu} = \mathcal{L}(X_n^{\hat{\alpha},\hat{\mu}})$ for all $n \geqslant 0$).

The key difference with the Asymptotic MFG formulation is that here the process starts with the invariant distribution $\hat{\mu}$. The control is a function of the state only and does not depend of time, and p and f depend only on this stationary distribution.

The stationary MFC problem is defined as follows: Find α^* such that the following condition holds: α^* is the minimizer of

$$\alpha \mapsto J^{SMFC}(\alpha) = \mathbb{E}\left[\sum_{n=0}^{+\infty} \gamma^n f(X_n^\alpha, \alpha(X_n^\alpha), \mu^\alpha)\right],$$

where the process X^α follows the MKV dynamics

$$X_{n+1}^\alpha \sim p\left(\cdot | X_n^\alpha, \alpha(X_n^\alpha), \mu^\alpha\right),$$

with initial distribution $X_0^\alpha \sim \mu^\alpha$, and such that μ^α is the invariant distribution of X^α (assuming it exists).

To conclude, let us mention that there is yet another formulation, in which the solution is stationary but depends on the initial distribution, see [12, Chapter 7].

3.4 Connecting the three formulations

Denoting by $\hat{\alpha}^{MFG}, \hat{\alpha}^{AMFG}$, and $\hat{\alpha}^{SMFG}$, the MFG equilibrium strategies respectively in the non-asymptotic, asymptotic, and stationary formulations, we expect

$$\begin{cases} \hat{\alpha}_n^{MFG}(x) \to \hat{\alpha}^{AMFG}(x), \qquad \forall x, \qquad \text{as} \qquad n \to +\infty, \\ \hat{\alpha}^{AMFG}(x) = \hat{\alpha}^{SMFG}(x), \qquad \forall x. \end{cases} \tag{3.1}$$

Similarly denoting by $\alpha^{*MFC}, \alpha^{*AMFC}$, and α^{*SMFC}, the MFC optimal controls respectively in the non-asymptotic, asymptotic, and stationary formulations, we expect

$$\begin{cases} \alpha_n^{*MFC}(x) \to \alpha^{*AMFC}(x), \qquad \forall x, \qquad \text{as} \qquad n \to +\infty, \\ \alpha^{*AMFC}(x) = \alpha^{*SMFC}(x), \qquad \forall x. \end{cases} \tag{3.2}$$

In fact, we have the following result.

Theorem 2 *Consider the set of admissible controls to be defined as the set of controls α such that the process $(X_n^\alpha)_{n\geqslant 0}$ is an irreducible and aperiodic Markov process on the finite space X. If a solution for the asymptotic MFG (resp. MFC) exists, then it is equal to the solution of the corresponding stationary MFG (resp. MFC) and vice versa.*

Proof: Let us consider the pair $(\hat{\alpha}^{AMFG}, \hat{\mu}^{AMFG})$ solution of an asymptotic MFG. The optimal control $\hat{\alpha}^{AMFG}$ is an optimizer over the set of admissible controls such that the process $(X_n^\alpha)_{n\geqslant 0}$ is an irreducible Markov process and admits a limiting distribution which is then the unique invariant distribution using the control $\hat{\alpha}^{AMFG}$. Note that the control $\hat{\alpha}^{AMFG}$ doesn't depend on the initial distribution μ_0 and consequently $\hat{\mu}^{AMFG}$ doesn't either. Therefore, $(\hat{\alpha}^{AMFG}, \hat{\mu}^{AMFG})$ is the solution of the AMFG starting from $\hat{\mu}^{AMFG}$, which is the corresponding stationary MFG problem. Thus, we deduce the desired relation $\hat{\alpha}^{AMFG} = \hat{\alpha}^{SMFG}$. A similar argument for MFC problems applies and shows that $\alpha^{*AMFC} = \alpha^{*SMFC}$. ■

Remark 2 *In terms of practical applications, the asymptotic formulation (AMFG and AMFC) seems to be the most appropriate, and if one is interested in the optimal controls, Theorem 2 shows that solving the asymptotic games also gives the solutions to the corresponding stationary games. Additionally,* (3.1) *and* (3.2) *indicate that it also gives the long time solutions to the corresponding time-dependent games. Developing Q-learning algorithms for solving time-dependent finite horizon games is addressed in Chapter 5.*

3.5 A Linear Quadratic Example

In this section, we provide explicit solutions for MFG, AMFG, SMFG, MFC, AMFC, and SMFC, in the case of continuous time, continuous space Linear-Quadratic stochastic differential games. We verify that (3.1) and (3.2), and therefore, Theorem 2, are satisfied in that case as well. In Section 4.3, discrete approximations of these games will also serve as benchmarks for our algorithm described in Chapter 4.

Let $(\Omega, \mathcal{F}, \mathbb{F} = (\mathcal{F}_t)_{t \geqslant 0}, \mathbb{P})$ be a filtered probability space, where the filtration supports a 1-dimensional Brownian motion $W = (W_t)_{t \geqslant 0}$ and an initial condition $\mu_0 \in L^2(\Omega, \mathcal{F}_0, \mathbb{P}; \mathbb{R})$. We consider the following model, in which the mean-field interactions are through the first moment. The running cost and the drift are defined as follows

$$f(x, \alpha, \mu) = \frac{1}{2}\alpha^2 + c_1 \left(x - c_2 m\right)^2 + c_3 \left(x - c_4\right)^2 + c_5 m^2, \qquad b(x, \alpha, \mu) = \alpha, \tag{3.3}$$

where $m = \int_{\mathbb{R}} x \mu(x) dx$. Here the parameters $c_2, c_4 \in \mathbb{R}$ and $c_1, c_3, c_5 \in \mathbb{R}_+$ are constant such that $c_1 + c_3 - c_1 c_2 \neq 0$. In this model the drift is simply the control, while the running cost can be understood as follows: the first term is a quadratic cost for controlling the diffusion, which penalizes high velocity, the second term incorporates mean field interactions and encourages the agents to be close to $c_2 m$ (if $c_2 = 1$, this has a mean-reverting effect), the third term creates an incentive for each agent to be close to the target position c_4, and the fourth term penalizes the population when its mean m is far away from zero. We thus obtain a complex combination of various effects, which can be balanced depending on the choice of parameters. A control α is admissible if the infinitesimal generator of the corresponding process has spectral gap implying exponentially ergodicity. To conclude we assume constant volatility σ.

3.5.1 Solution for non-asymptotic MFG

We present the solution for the following MFG problem

1. Fix $\boldsymbol{m} = (m_t)_{t\geqslant 0} \subset \mathbb{R}$ and solve the stochastic control problem:

$$\begin{aligned}
&\min_{\boldsymbol{\alpha}} J^{\boldsymbol{m}}(\boldsymbol{\alpha})\\
&= \min_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^\infty e^{-\beta t} f(X_t^{\boldsymbol{\alpha}}, \alpha_t, m_t)dt\right] =\\
&= \min_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^{+\infty} e^{-\beta t}\left(\frac{1}{2}\alpha_t^2 + c_1\left(X_t^{\boldsymbol{\alpha}} - c_2 m_t\right)^2 + c_3\left(X_t^{\boldsymbol{\alpha}} - c_4\right)^2 + c_5 m_t^2\right)dt\right],
\end{aligned}$$

subject to

$$\begin{aligned}
dX_t^{\boldsymbol{\alpha}} &= \alpha_t dt + \sigma dW_t,\\
X_0^{\boldsymbol{\alpha}} &\sim \mu_0.
\end{aligned}$$

2. Find the fixed point, $\hat{\boldsymbol{m}} = (\hat{m}_t)_{t\geqslant 0}$, such that $\mathbb{E}\left[X_t^{\hat{\boldsymbol{\alpha}}}\right] = \hat{m}_t$ for all $t \geqslant 0$.

This problem can be solved by two equivalent approaches: PDE and FBSDEs. Both approaches start by solving the problem defined by a finite horizon T. Then, the solution to the infinite horizon problem is obtained by taking the limit T goes to infinity. Let $V^{\boldsymbol{m}^T,T}(t,x)$ be the optimal value function for the finite horizon problem conditioned on $X_0 = x$, i.e.

$$\begin{aligned}
V^{\boldsymbol{m}^T,T}(t,x) &= \inf_{\boldsymbol{\alpha}} J^{\boldsymbol{m},x}(\boldsymbol{\alpha}) = \inf_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_t^T e^{-\beta s} f(X_s^{\boldsymbol{\alpha}}, \alpha_s, m_s^T)ds \middle| X_0^{\boldsymbol{\alpha}} = x\right],\\
V^{\boldsymbol{m}^T,T}(T,x) &= 0.
\end{aligned}$$

where $\boldsymbol{m}^T = \{m_t^T\}_{0\leqslant t\leqslant T} \subset \mathbb{R}$. Let's consider the following ansatz with its derivatives

$$\begin{aligned}
V^{\boldsymbol{m}^T,T}(t,x) &= \Gamma_2^T(t)x^2 + \Gamma_1^T(t)x + \Gamma_0^T(t),\\
\partial_t V^{\boldsymbol{m}^T,T}(t,x) &= \dot{\Gamma}_2^T(t)x^2 + \dot{\Gamma}_1^T(t)x + \dot{\Gamma}_0^T(t),\\
\partial_x V^{\boldsymbol{m}^T,T}(t,x) &= 2\Gamma_2^T(t)x + \Gamma_1^T(t),\\
\partial_{xx} V^{\boldsymbol{m}^T,T}(t,x) &= 2\Gamma_2^T(t).
\end{aligned} \tag{3.4}$$

Then, the HJB equation for the value function reads:

$$\begin{aligned}
&\partial_t V^{\boldsymbol{m}^T,T} - \beta V^{\boldsymbol{m}^T,T} + \inf_{\alpha}\{\mathcal{A}^X V^{\boldsymbol{m}^T,T} + f(x,\alpha,m^T)\} \\
&= \partial_t V^{\boldsymbol{m}^T,T} - \beta V^{\boldsymbol{m}^T,T} + \inf_{\alpha}\Big\{\alpha\partial_x V^{\boldsymbol{m}^T,T} + \frac{1}{2}\sigma^2\partial_{xx}V^{\boldsymbol{m}^T,T} + \frac{1}{2}\alpha^2 + c_1(x-c_2m^T)^2 \\
&\quad + c_3(x-c_4)^2 + c_5(m^T)^2\Big\} \\
&= \partial_t V^{\boldsymbol{m}^T,T} - \beta V^{\boldsymbol{m}^T,T} - \partial_x V^{\boldsymbol{m}^T,T^2} + \frac{1}{2}\sigma^2\partial_{xx}V^{\boldsymbol{m}^T,T} + \frac{1}{2}\partial_x V^{\boldsymbol{m}^T,T^2} + c_1(x-c_2m^T)^2 \\
&\quad + c_3(x-c_4)^2 + c_5(m^T)^2 \\
&= \partial_t V^{\boldsymbol{m}^T,T} - \beta V^{\boldsymbol{m}^T,T} - \frac{1}{2}\partial_x V^{\boldsymbol{m}^T,T^2} + \frac{1}{2}\sigma^2\partial_{xx}V^{\boldsymbol{m}^T,T} + c_1(x-c_2m^T)^2 + c_3(x-c_4)^2 \\
&\quad + c_5(m^T)^2 = 0,
\end{aligned}$$

where in the third line we evaluated the infimum at $\hat{\alpha}^T = -V_x^{\boldsymbol{m}^T,T}$. The following ODEs system is obtained by replacing the ansatz and its derivatives in the HJB equation:

$$\begin{cases}
\dot{\Gamma}_2^T - 2(\Gamma_2^T)^2 - \beta\Gamma_2^T + c_1 + c_3 = 0, & \Gamma_2^T(T) = 0, \\
\dot{\Gamma}_1^T = (2\Gamma_2^T + \beta)\Gamma_1^T + 2c_1c_2m^T + 2c_3c_4, & \Gamma_1^T(T) = 0, \\
\dot{\Gamma}_0^T = \beta\Gamma_0^T + \frac{1}{2}(\Gamma_1^T)^2 - \sigma^2\Gamma_2^T - c_3{c_4}^2 - (c_1{c_2}^2 + c_5)(m^T)^2, & \Gamma_0^T(T) = 0, \\
\dot{m}^T = -2\Gamma_2^T m^T - \Gamma_1^T, & m^T(0) = \mathbb{E}[\mu_0] = m_0,
\end{cases} \tag{3.5}$$

where the last equation is obtained by considering the expectation of $X_t^{\boldsymbol{\alpha}}$ after replacing $\hat{\alpha}^T = -\partial_x V^{\boldsymbol{m}^T,T} = -(\Gamma_2^T x + \Gamma_1^T)$. The first equation is a Riccati equation. In particular, the solution Γ_2^T converges to $\hat{\Gamma}_2 = \frac{-\beta+\sqrt{\beta^2+8(c_1+c_3)}}{4}$ as T goes to infinity. The second and fourth ODEs are coupled and they can be written in matrix notation as

$$\dot{\overbrace{\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}}} = \begin{bmatrix} -2\Gamma_2^T & -1 \\ 2c_1c_2 & 2\Gamma_2^T + \beta \end{bmatrix}\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix} + \begin{pmatrix} 0 \\ 2c_3c_4 \end{pmatrix}, \quad \begin{pmatrix} m^T(0) \\ \Gamma_1^T(T) \end{pmatrix} = \begin{pmatrix} m_0 \\ 0 \end{pmatrix}. \tag{3.6}$$

We start by solving the homogeneous equation, i.e.

$$\dot{\overbrace{\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}}} = K_t^T \begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix} := \begin{bmatrix} -2\Gamma_2^T & -1 \\ 2c_1c_2 & 2\Gamma_2^T + \beta \end{bmatrix} \begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}, \quad \begin{pmatrix} m^T(0) \\ \Gamma_1^T(T) \end{pmatrix} = \begin{pmatrix} m_0 \\ 0 \end{pmatrix}. \tag{3.7}$$

We introduce the propagator P^T, i.e.

$$\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix} = P_t^T \begin{pmatrix} m^T(0) \\ \Gamma_1^T(0) \end{pmatrix}. \tag{3.8}$$

By deriving $\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}$ and expressing the initial conditions in terms of the inverse of P^T and $\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}$, we obtain

$$\dot{\overbrace{\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}}} = \dot{P}_t^T \begin{pmatrix} m^T(0) \\ \Gamma_1^T(0) \end{pmatrix} = \dot{P}_t^T (P_t^T)^{-1} \begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}. \tag{3.9}$$

By comparing the last system with (3.7), we obtain

$$\begin{cases} \dot{P}_t^T & = K_t^T P_t^T \\ P_0^T & = \mathbb{I}_2 \end{cases} \tag{3.10}$$

where $\mathbb{I}_2$ is the identity matrix in dimension 2. The solution is given by $P_t^T = e^{\int_0^t K_s^T ds} := e^{L_t^T}$. In particular, the exponent is equal to

$$L_t^T = \int_0^t K_s^T ds = \begin{bmatrix} -2\int_0^t \Gamma_2^T(s)ds & -t \\ 2c_1c_2t & 2\int_0^t \Gamma_2^T(s)ds + \beta t \end{bmatrix} = \begin{bmatrix} g_t^T & d_t \\ b_t & a_t^T \end{bmatrix}. \tag{3.11}$$

We evaluate the exponential $P^T(t) = e^{L_t^T}$ by using the Taylor's expansion and diagonalizing the matrix L_t^T. The eigenvalues/eigenvectors of L_t^T are given by

$$\lambda_{1\backslash 2,t}^T := \frac{a_t^T + g_t^T \pm \sqrt{(a_t^T - g_t^T)^2 + 4b_td_t}}{2}, \quad v_{1,t}^T := \begin{pmatrix} d_t \\ \lambda_{1,t}^T - g_t^T \end{pmatrix}, \quad v_{2,t}^T := \begin{pmatrix} d_t \\ \lambda_{2,t}^T - g_t^T \end{pmatrix}. \tag{3.12}$$

P_t is obtained by

$$\begin{aligned}
P_t^T &= \begin{pmatrix} p_t^T(1,1) & p_t^T(1,2) \\ p_t^T(2,1) & p_t^T(2,2) \end{pmatrix} \\
&= e^{L_t^T} = \sum_{k=0}^{\infty} \begin{bmatrix} v_{1,t}^T & v_{2,t}^T \end{bmatrix} \frac{\begin{pmatrix} \lambda_{1,t}^T & 0 \\ 0 & \lambda_{2,t}^T \end{pmatrix}^k}{k!} \begin{bmatrix} v_{1,t}^T & v_{2,t}^T \end{bmatrix}^{-1} \\
&:= S_t^T \sum_{k=0}^{\infty} \frac{D_t^{T^k}}{k!} (S_t^T)^{-1} \\
&= S_t^T \begin{pmatrix} e^{\lambda_{1,t}^T} & 0 \\ 0 & e^{\lambda_{2,t}^T} \end{pmatrix} (S_t^T)^{-1} \\
&= \frac{1}{d_t(\lambda_{2,t}^T - \lambda_{1,t}^T)} \\
&\times \begin{pmatrix} d_t e^{\lambda_{1,t}^T}(\lambda_{2,t}^T - g_t^T) + d_t e^{\lambda_{2,t}^T}(g_t^T - \lambda_{1,t}^T) & d_t^2(e^{\lambda_{2,t}^T} - e^{\lambda_{1,t}^T}) \\ (\lambda_{1,t}^T - g_t^T)(\lambda_{2,t}^T - g_t^T)(e^{\lambda_{1,t}^T} - e^{\lambda_{2,t}^T}) & d_t e^{\lambda_{2,t}^T}(\lambda_{2,t}^T - g_t^T) + d_t e^{\lambda_{1,t}^T}(g_t^T - \lambda_{1,t}^T) \end{pmatrix}.
\end{aligned} \tag{3.13}$$

In order to solve the non homogeneous case, we introduce an extra term $\begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix}$, i.e.

$$\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix} = P_t^T \begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix}. \tag{3.14}$$

By deriving $\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}$, we obtain

$$\begin{aligned}\overset{\cdot}{\overgroup{\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}}} =& \dot{P}_t^T \begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix} + P_t^T \overset{\cdot}{\overgroup{\begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix}}} = K_t^T P_t^T \begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix} + P_t^T \overset{\cdot}{\overgroup{\begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix}}} \\ =& K_t^T \begin{pmatrix} m_t^T \\ \Gamma_1^T \end{pmatrix} + P_t^T \overset{\cdot}{\overgroup{\begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix}}}. \end{aligned} \tag{3.15}$$

By comparing (3.6) with (3.15), we obtain

$$\overset{\cdot}{\overgroup{\begin{pmatrix} h_1^T \\ h_2^T \end{pmatrix}}} = (P_t^T)^{-1} \begin{pmatrix} 0 \\ 2c_4c_4 \end{pmatrix} = \frac{1}{|P_t^T|} \begin{pmatrix} p_t^T(2,2) & -p_t^T(1,2) \\ -p_t^T(2,1) & p_t^T(1,1) \end{pmatrix} \begin{pmatrix} 0 \\ 2c_3c_4 \end{pmatrix}. \tag{3.16}$$

By integration we obtain

$$\begin{aligned} h_1^T(t) &= h_1^T(0) - 2c_3c_4 \int_0^t \frac{p_s^T(1,2)}{|P_s^T|} ds, \\ h_2^T(t) &= h_2^T(0) + 2c_3c_4 \int_0^t \frac{p_s^T(1,1)}{|P_s^T|} ds, \end{aligned} \tag{3.17}$$

where $h_1^T(0) = m_0$ and $h_2^T(0) = \Gamma_1^T(0)$.

We use the terminal condition $\Gamma_1^T(T) = 0$ to obtain an evaluation of $h_2^T(0) = \Gamma_1^T(0)$ in terms of P_T^T and m_0, i.e.

$$\begin{aligned} \Gamma_1^T(T) &= p_T^T(2,1) h_1^T(T) + p_T^T(2,2) h_2^T(T) = 0, \\ \Gamma_1^T(T) &= p_T^T(2,1) \left(m_0 - 2c_3c_4 \int_0^T \frac{p_s^T(1,2)}{|P_s^T|} ds \right) \\ &\quad + p_T^T(2,2) \left(\Gamma_1^T(0) + 2c_3c_4 \int_0^T \frac{p_s^T(1,1)}{|P_s^T|} ds \right) = 0, \\ \Gamma_1^T(0) &= -\frac{p_T^T(2,1)}{p_T^T(2,2)} \left(m_0 - 2c_3c_4 \int_0^T \frac{p_s^T(1,2)}{|P_s^T|} ds \right) - 2c_3c_4 \int_0^T \frac{p_s^T(1,1)}{|P_s^T|} ds. \end{aligned} \tag{3.18}$$

In order to evaluate the limit of $\Gamma_1^T(0)$ as T goes to infinity, we analyze the different terms separately. First, we evaluate the following limit:

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T \Gamma_2^T(s) ds = \lim_{T\to\infty} \Gamma_2^T(s_1) = \hat{\Gamma}_2, \quad s_1 \in [0,T], \tag{3.19}$$

where we applied the mean value integral theorem and $\hat{\Gamma}_2 = \frac{-\beta+\sqrt{\beta^2+8(c_1+c_3)}}{4}$ is the limit of the solution of the Riccati equation obtained previously, i.e. $\hat{\Gamma}_2 = \lim_{T\to\infty} \Gamma_2^T(s)$. We recall that

$$\lambda_{2,T}^T - \lambda_{1,T}^T = \sqrt{(a_T^T - g_T^T)^2 + 4b_T^T d_T} = T\sqrt{\left(\frac{4}{T}\int_0^T \Gamma_2^T(s)ds + \beta\right)^2 - 8c_1c_2} > 0$$

which goes to infinity as T goes to ∞ when the term under square root is well defined. We observe that

$$\begin{aligned}
\hat{g}_t &:= \lim_{T\to\infty} g_t^T = \lim_{T\to\infty} -2\int_0^t \Gamma_2^T(s)ds = -2\hat{\Gamma}_2 t := gt,\\
b_t &= 2c_1c_2t,\\
\hat{a}_t &:= \lim_{T\to\infty} a_t^T = \lim_{T\to\infty} 2\int_0^t \Gamma_2^T(s)ds + \beta t = 2\hat{\Gamma}_2 t + \beta t,\\
d_t &= -t,\\
\hat{\lambda}_{1\backslash 2,t} &:= \lim_{T\to\infty} \lambda_{1\backslash 2,t}^T = \frac{\hat{a}_t + \hat{g}_t \pm \sqrt{(\hat{a}_t - \hat{g}_t)^2 + 4b_td_t}}{2}\\
&= t\frac{\beta \pm \sqrt{(4\hat{\Gamma}_2 + \beta)^2 - 8c_1c_2}}{2} := t\lambda_{1\backslash 2},\\
\hat{P}_t &:= \lim_{T\to\infty} P_t^T = \frac{1}{d_t(\hat{\lambda}_{2,t} - \hat{\lambda}_{1,t})}\\
&\times \begin{pmatrix} d_te^{\hat{\lambda}_{1,t}}(\hat{\lambda}_{2,t} - \hat{g}_t) + d_te^{\hat{\lambda}_{2,t}}(\hat{g}_t - \hat{\lambda}_{1,t}) & d_t^2(e^{\hat{\lambda}_{2,t}} - e^{\hat{\lambda}_{1,t}}) \\ (\hat{\lambda}_{1,t} - \hat{g}_t)(\hat{\lambda}_{2,t} - \hat{g}_t)(e^{\hat{\lambda}_{1,t}} - e^{\hat{\lambda}_{2,t}}) & d_te^{\hat{\lambda}_{2,t}}(\hat{\lambda}_{2,t} - \hat{g}_t) + d_te^{\hat{\lambda}_{1,t}}(\hat{g}_t - \hat{\lambda}_{1,t}) \end{pmatrix}.
\end{aligned} \tag{3.20}$$

To evaluate $\hat{\Gamma}_1(0) = \lim_{T\to\infty} \Gamma_1^T(0)$, we study the limit of the remaining terms:

$$
\begin{aligned}
\lim_{T\mapsto\infty} -\frac{p_T^T(2,1)}{p_T^T(2,2)} &= \lim_{T\mapsto\infty} \frac{(\lambda_{1,T}^T - g_T^T)(\lambda_{2,T}^T - g_T^T)(e^{\lambda_{2,T}^T} - e^{\lambda_{1,T}^T})}{d_T e^{\lambda_{2,T}^T}(\lambda_{2,T}^T - g_T^T) + d_T e^{\lambda_{1,T}^T}(g_T^T - \lambda_{1,T}^T)} \\
&= \lim_{T\mapsto\infty} \frac{1}{\frac{d_T}{(\lambda_{1,T}^T - g_T^T)(1-e^{\lambda_{1,T}^T - \lambda_{2,T}^T})} + \frac{d_T}{(\lambda_{2,T}^T - g_T^T)(1-e^{\lambda_{2,T}^T - \lambda_{1,T}^T})}} \\
&= -(\lambda_1 - g) \\
&= -(\lambda_1 + 2\hat{\Gamma}_2), \\
\lim_{T\mapsto\infty} \int_0^T \frac{p_s^T(1,2)}{|P_s^T|} ds &= \lim_{T\mapsto\infty} \int_0^T \frac{d_s(e^{\lambda_{2,s}^T} - e^{\lambda_{1,s}^T})}{(\lambda_{2,s}^T - \lambda_{1,s}^T)(e^{\lambda_{1,s}^T + \lambda_{2,s}^T})} ds \\
&= \frac{1}{\lambda_2 - \lambda_1}\left(\frac{1}{\lambda_2} - \frac{1}{\lambda_1}\right) \\
\lim_{T\mapsto\infty} \int_0^T \frac{p_s^T(1,1)}{|P_s^T|} ds &= \lim_{T\mapsto\infty} \int_0^T \frac{1}{e^{\lambda_{1,s}^T + \lambda_{2,s}^T}} \left(e^{\lambda_{1,s}^T} \frac{\lambda_{2,s}^T - g_s^T}{\lambda_{2,s}^T - \lambda_{1,s}^T} + e^{\lambda_{2,s}^T} \frac{g_s^T - \lambda_{1,s}^T}{\lambda_{2,s}^T - \lambda_{1,s}^T} \right) ds \\
&= \frac{\lambda_2 - g}{\lambda_2(\lambda_2 - \lambda_1)} + \frac{g - \lambda_1}{\lambda_1(\lambda_2 - \lambda_1)}.
\end{aligned}
\tag{3.21}
$$

Finally, the value of $\hat{\Gamma}_1(0)$ is given by

$$
\hat{\Gamma}_1(0) = -(\lambda_1 - g)m_0 - 2\frac{c_3 c_4}{\lambda_2}. \tag{3.22}
$$

Given $\hat{\Gamma}_1(0)$, we evaluate the limit as T goes to ∞ of (3.17), i.e.

$$
\begin{aligned}
h_1(t) := \lim_{T\mapsto\infty} h_1^T(t) &= m_0 - 2c_3c_4 \lim_{T\mapsto\infty} \int_0^t \frac{p_s^T(1,2)}{|P_s^T|} ds \\
&= m_0 + 2\frac{c_3c_4}{\lambda_2 - \lambda_1}\left(\frac{1}{\lambda_2}e^{-t\lambda_2} - \frac{1}{\lambda_1}e^{-t\lambda_1} + \frac{1}{\lambda_1} - \frac{1}{\lambda_2}\right), \\
h_2(t) := \lim_{T\mapsto\infty} h_2^T(t) &= \lim_{T\mapsto\infty}\left(\Gamma_1^T(0) + 2c_3c_4\int_0^t \frac{p_s^T(1,1)}{|P_s^T|} ds\right) \\
&= \hat{\Gamma}_1(0) + 2\frac{c_3c_4}{\lambda_2 - \lambda_1}\left(\frac{\lambda_2 - g}{\lambda_2}(1 - e^{-t\lambda_2}) + \frac{g - \lambda_1}{\lambda_1}(1 - e^{-t\lambda_1})\right).
\end{aligned}
\tag{3.23}
$$

We can conclude that

$$\begin{aligned}
\hat{m}_t &= \lim_{T\to\infty} m_t^T \\
&= \hat{p}_t(1,1)h_1(t) + \hat{p}_t(1,2)h_2(t) \\
&= \left(m_0 + 2\frac{c_3c_4}{\lambda_2-\lambda_1}\left(\frac{1}{\lambda_1}-\frac{1}{\lambda_2}\right)\right)e^{t\lambda_1} + 2\frac{c_3c_4}{\lambda_2-\lambda_1}\left(\frac{1}{\lambda_2}-\frac{1}{\lambda_1}\right), \\
\hat{\Gamma}_1(t) &= \lim_{T\to\infty}\Gamma_1^T(t) \\
&= \hat{p}_t(2,1)h_1(t) + \hat{p}_t(2,2)h_2(t) \\
&= m_0(g-\lambda_1)e^{t\lambda_1} + 2\frac{c_3c_4}{\lambda_2-\lambda_1}\left(\frac{\lambda_2-g}{\lambda_2}-\frac{\lambda_1-g}{\lambda_1}\right).
\end{aligned} \tag{3.24}$$

Finally, the third ODE in (3.5) can be solved by plugging in the solution of the previous ones and integrating. Since our interest is into the evolution of the mean and the control function, we omit these calculations, but we recall that:

$$\hat{\alpha}_t = -(\hat{\Gamma}_2 x + \hat{\Gamma}_1(t)), \quad \hat{\Gamma}_2 = \frac{-\beta+\sqrt{\beta^2+8(c_1+c_3)}}{4}, \tag{3.25}$$

and we observe that

$$\lim_{t\to\infty}\hat{\alpha}_t = -(\hat{\Gamma}_2 x + \hat{\Gamma}_1), \quad \hat{\Gamma}_1 = -\frac{4c_1c_2\hat{\Gamma}_2}{\lambda_2} = \frac{c_3c_4\hat{\Gamma}_2}{2(c_1+c_3-c_1c_2)}. \tag{3.26}$$

3.5.2 Solution for Asymptotic MFG

The asymptotic version of the problem presented above is given by:

1. Fix $m \in \mathbb{R}$ and solve the stochastic control problem:

$$\begin{aligned}
\min_{\boldsymbol{\alpha}} J^m(\boldsymbol{\alpha}) &= \min_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^\infty e^{-\beta t} f(X_t^{\boldsymbol{\alpha}}, \alpha_t, m)dt\right] \\
&= \min_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^\infty e^{-\beta t}\left(\frac{1}{2}\alpha_t^2 + c_1\left(X_t^{\boldsymbol{\alpha}} - c_2 m\right)^2 + c_3\left(X_t^{\boldsymbol{\alpha}} - c_4\right)^2 + c_5 m^2\right)dt\right],
\end{aligned}$$

subject to: $\quad dX_t^{\boldsymbol{\alpha}} = \alpha_t dt + \sigma dW_t, \quad X_0^{\boldsymbol{\alpha}} \sim \mu_0.$

2. Find the fixed point, $\hat{m}$, such that $\hat{m} = \lim_{t\to+\infty}\mathbb{E}\left[X_t^{\hat{\alpha},\hat{m}}\right]$.

Let $V^m(x)$ be the optimal value function given $m \in \mathbb{R}$ and conditioned on $X_0 = x$, i.e.

$$\begin{aligned} V^m(x) &= \inf_{\boldsymbol{\alpha}} J^{m,x}(\boldsymbol{\alpha}) \\ &= \inf_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^{+\infty} e^{-\beta t}\left(\frac{1}{2}\alpha_t^2 + c_1\left(X_t^{\boldsymbol{\alpha}} - c_2 m\right)^2 + c_3\left(X_t^{\boldsymbol{\alpha}} - c_4\right)^2 + c_5 m^2\right)\Big| X_0^{\boldsymbol{\alpha}} = x\right]. \end{aligned}$$

We consider the following ansatz with its derivatives with respect to x:

$$\begin{aligned} V^m(x) &= \Gamma_2 x^2 + \Gamma_1 x + \Gamma_0, \\ \dot{V}^m(x) &= 2\Gamma_2 x + \Gamma_1, \\ \ddot{V}^m(x) &= 2\Gamma_2. \end{aligned}$$

Let's consider the HJB equation

$$\begin{aligned} &\beta V^m(x) - \inf_{\alpha}\{\mathcal{A}^X V^m(x) + f(x,\alpha,m)\} \\ =&\beta V^m(x) - \inf_{\alpha}\left\{\alpha \dot{V}(x) + \frac{1}{2}\sigma^2 \ddot{V}^m(x) + \frac{1}{2}\alpha^2 + c_1(x - c_2 m)^2 + c_3(x-c_4)^2 + c_5 m^2\right\} \\ =&\beta V^m(x) - \left\{-(\dot{V}^m)^2(x) + \frac{1}{2}\sigma^2\ddot{V}^m(x) + \frac{1}{2}(\dot{V}^m)^2(x) + c_1(x-c_2 m)^2 + c_3(x-c_4)^2\right. \\ &\quad \left. + c_5 m^2\right\} \\ =&\beta V^m(x) + \frac{1}{2}(\dot{V}^m)^2(x) - \frac{1}{2}\sigma^2\ddot{V}^m(x) - c_1(x-c_2 m)^2 - c_3(x-c_4)^2 - c_5 m^2 = 0, \end{aligned}$$

where in the third line we evaluated the infimum at $\hat{\alpha}(x) = -\dot{V}^m(x)$. Replacing the ansatz and its derivatives in the HJB equation, it follows that

$$\begin{aligned} &\left(\beta\Gamma_2 + 2\Gamma_2^2 - c_1 - c_3\right)x^2 + (\beta\Gamma_1 + 2\Gamma_2\Gamma_1 + 2c_1c_2m + 2c_3c_4)x \\ &\qquad\qquad + \beta\Gamma_0 + \frac{1}{2}\Gamma_1^2 - \sigma^2\Gamma_2 - (c_1{c_2}^2 + c_5)m^2 - c_3{c_4}^2 = 0. \end{aligned}$$

An easy computation gives the values

$$\begin{aligned} \Gamma_2 &= \frac{-\beta + \sqrt{\beta^2 + 8(c_1+c_3)}}{4}, \\ \Gamma_1 &= -\frac{2c_1c_2m + 2c_3c_4}{\beta + 2\Gamma_2}, \\ \Gamma_0 &= \frac{c_5m^2 + c_3{c_4}^2 + c_1{c_2}^2m^2 + \sigma^2\Gamma_2 - \frac{1}{2}\Gamma_1^2}{\beta}. \end{aligned}$$

By plugging the control $\hat{\alpha}(x) = -(2\Gamma_2 x + \Gamma_1)$ into the dynamics of X_t and taking the expected value, we obtain an ODE for m_t

$$\dot{m}_t = -(2\Gamma_2 m_t + \Gamma_1). \tag{3.27}$$

The solution of (3.27) is used to derive m as follows

$$\begin{aligned} m &= \lim_{t\mapsto\infty} m_t = \lim_{t\mapsto\infty} -\frac{\Gamma_1}{2\Gamma_2} + \left(m_0 + \frac{\Gamma_1}{\Gamma_2}\right) e^{-2\Gamma_2 t} = -\frac{\Gamma_1}{2\Gamma_2} = \frac{2c_1c_2m + 2c_3c_4}{2\Gamma_2(\beta + 2\Gamma_2)}, \\ m &= \frac{c_3c_4}{\Gamma_2(\beta + 2\Gamma_2) - c_1c_2} \end{aligned} \tag{3.28}$$

To summarize, we derived that $\hat{\alpha}(x) = -(2\Gamma_2 x + \Gamma_1)$ with $\Gamma_2 = \hat{\Gamma}_2$ and $\Gamma_1 = \hat{\Gamma}_1$ obtained in (3.26). In other words, we have checked that

$$\lim_{t\to\infty} \hat{\alpha}_t^{MFG}(x) = \hat{\alpha}^{AMFG}(x), \quad \forall x,$$

that is the first part of (3.1) for this LQ MFG.

3.5.3 Solution for stationary MFG

The only difference with the derivation above in the case of asymptotic MFG is that m_t should be a constant which, from (3.27), should satisfy $2\Gamma_2 m + \Gamma_1 = 0$. Therefore, m takes the same value as in (3.28), and we deduce

$$\hat{\alpha}^{SMFG}(x) = \hat{\alpha}^{AMFG}(x), \quad \forall x,$$

that is the second part of (3.1) for this LQ MFG.

3.5.4 Solution for non-asymptotic MFC

We present the solution for the following non-asymptotic MFC problem

$$\begin{aligned}\min_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}) &= \min_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^{\infty} e^{-\beta t} f(X_t^{\boldsymbol{\alpha}}, \alpha_t, \mathbb{E}\left[X_t^{\boldsymbol{\alpha}}\right])dt\right] \\ &= \min_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^{+\infty} e^{-\beta t}\left(\frac{1}{2}\alpha_t^2 + c_1\left(X_t^{\boldsymbol{\alpha}} - c_2\mathbb{E}\left[X_t^{\boldsymbol{\alpha}}\right]\right)^2 + c_3\left(X_t^{\boldsymbol{\alpha}} - c_4\right)^2\right.\right. \\ &\qquad \left.\left. + c_5\mathbb{E}\left[X_t^{\boldsymbol{\alpha}}\right]^2\right) dt\right],\end{aligned}$$

subject to:

$$dX_t^{\boldsymbol{\alpha}} = \alpha_t dt + \sigma dW_t, \quad X_0^{\boldsymbol{\alpha}} \sim \mu_0.$$

Note that here the mean $\mathbb{E}\left[X_t^{\boldsymbol{\alpha}}\right]$ of the population changes instantaneously when $\boldsymbol{\alpha}$ changes.

This problem can be solved by two equivalent approaches: PDE and FBSDEs. Both approaches start by solving the problem defined by a finite horizon T. Then, the solution to the infinite horizon problem is obtained by taking the limit for T goes to infinity. Let $V^T(t,x)$ be the optimal value function for the finite horizon problem conditioned on $X_0 = x$, i.e.

$$V^T(t,x) = \inf_{\boldsymbol{\alpha}} J^{\boldsymbol{m}^{\boldsymbol{\alpha}},x}(\boldsymbol{\alpha}) = \inf_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_t^T e^{-\beta s} f(X_s^{\boldsymbol{\alpha}}, \alpha_s, m_s^{\boldsymbol{\alpha}}) ds \middle| X_0^{\boldsymbol{\alpha}} = x\right], \quad V^T(T,x) = 0.$$

Let's consider the following ansatz with its derivatives

$$\begin{aligned} V^T(t,x) &= \Gamma_2^T(t)x^2 + \Gamma_1^T(t)x + \Gamma_0^T(t), \quad V^T(T,x) = 0, \\ \partial_t V^T(t,x) &= \dot{\Gamma}_2^T(t)x^2 + \dot{\Gamma}_1^T(t)x + \dot{\Gamma}_0^T(t), \\ \partial_x V^T(t,x) &= 2\Gamma_2^T(t)x + \Gamma_1^T(t), \\ \partial_{xx} V^T(t,x) &= 2\Gamma_2^T(t), \end{aligned} \tag{3.29}$$

Starting by the MFC-HJB equation (4.12) given in [12], we extended it to the asymptotic case as follows

$$\beta V^T - V_t^T - H\left(t,x,\boldsymbol{\mu},\alpha\right) - \int_{\mathbb{R}} \frac{\delta H}{\delta \mu}\left(t,h,\boldsymbol{\mu},-\partial_x V^T\right)(x)\mu_t(h)dh = 0,$$

where $m_t = \int_{\mathbb{R}} y\mu_t(dy)$ and $\alpha^* = -\partial_x V^T$. We have:

$$\begin{aligned}
H\left(t,x,\boldsymbol{\mu},\alpha\right) &:= \inf_{\alpha}\left\{\mathcal{A}^X V^T + f\left(t,x,\alpha,\boldsymbol{\mu}\right)\right\}\\
&= \inf_{\alpha}\left\{\alpha\partial_x V^T + \frac{1}{2}\sigma^2\partial_{xx}V^T + \frac{1}{2}\alpha^2 + c_1(x-c_2m_t)^2 + c_3(x-c_4)^2 + c_5{m_t}^2\right\}\\
&= -\frac{1}{2}(\partial_x V^T)^2 + \frac{1}{2}\sigma^2\partial_{xx}V^T + c_1(x-c_2m_t)^2 + c_3(x-c_4)^2 + c_5{m_t}^2,
\end{aligned}$$

$$\begin{aligned}
\frac{\delta H\left(t,h,\boldsymbol{\mu},\alpha\right)}{\delta\mu} &= \frac{\delta}{\delta\mu}\left(c_1(h-c_2m_t)^2 + c_5{m_t}^2\right)(x)\\
&= \frac{\delta}{\delta\mu}\left(c_1\left(h - c_2\int_{\mathbb{R}} y\mu_t(dy)\right)^2 + c_5\left(\left(\int_{\mathbb{R}} y\mu_t(dy)\right)^2\right)\right)(x)\\
&= -2c_1c_2x\left(h - c_2\int_{\mathbb{R}} y\mu_t(dy))\right) + 2c_5x\int_{\mathbb{R}} y\mu_t(dy)\\
&= -2c_1c_2x(h-c_2m_t) + 2c_5xm_t,
\end{aligned}$$

$$\int_{\mathbb{R}} \frac{\delta H}{\delta\mu}\left(t,h,\boldsymbol{\mu},-\partial_x V^T\right)(x)\mu_t(h)dh = -2c_1c_2x(m_t-c_2m_t) + 2c_5xm_t,$$

and finally

$$\begin{aligned}
\beta V^T - \partial_t V^T + \frac{1}{2}(\partial_x^T)^2 - \frac{1}{2}\sigma^2\partial_{xx}V^T - c_1(x-c_2m_t)^2 - c_3(x-c_4)^2&\\
- c_5{m_t}^2 + 2c_1c_2x(m_t-c_2m_t) - 2c_5xm_t = 0.&
\end{aligned}$$

The following system of ODEs is obtained by replacing the ansatz and its derivatives in the MFC-HJB:

$$\begin{cases}
\dot{\Gamma}_2^T - 2(\Gamma_2^T)^2 - \beta\Gamma_2^T + c_1 + c_3 = 0, & \Gamma_2^T(T) = 0,\\
\dot{\Gamma}_1^T = (2\Gamma_2^T + \beta)\Gamma_1^T + (2c_1c_2(2-c_2) - 2c_5)m_t^T + 2c_3c_4, & \Gamma_1^T(T) = 0,\\
\dot{\Gamma}_0^T = \beta\Gamma_0^T + \frac{1}{2}(\Gamma_1^T)^2 - \sigma^2\Gamma_2^T - c_3{c_4}^2 - (c_1{c_2}^2 + c_5)(m_t^T)^2, & \Gamma_0^T(T) = 0,\\
\dot{m}_t^T = -2\Gamma_2^T m^T - \Gamma_1^T, & m^T(0) = \mathbb{E}\left[X_0^{\boldsymbol{\alpha}}\right] = m_0,
\end{cases} \tag{3.30}$$

where the last equation is obtained by considering the expectation of $X_t^{\boldsymbol{\alpha}}$ after replacing $\alpha^*(x) = -\partial_x V^T(x) = -(\Gamma_2^T x + \Gamma_1^T)$. The first equation is a Riccati equation. In particular, the solution Γ_2^T converges to $\Gamma_2^* = \frac{-\beta+\sqrt{\beta^2+8(c_1+c_3)}}{4}$ as T goes to infinity. The second and fourth ODEs are coupled and they can be written in matrix notation as

$$\overset{\cdot}{\overbrace{\begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix}}} = \begin{bmatrix} -2\Gamma_2^T & -1 \\ (2c_1c_2(2-c_2)-2c_5) & 2\Gamma_2^T+\beta \end{bmatrix} \begin{pmatrix} m^T \\ \Gamma_1^T \end{pmatrix} + \begin{pmatrix} 0 \\ 2c_3c_4 \end{pmatrix}, \quad \begin{pmatrix} m^T(0) \\ \Gamma_1^T(T) \end{pmatrix} = \begin{pmatrix} m_0 \\ 0 \end{pmatrix}. \tag{3.31}$$

By similar calculations to the non-asymptotic MFG case, the following solutions can be obtained

$$\begin{aligned} m_t^* &= \lim_{T\to\infty} m_t^T = p_t^*(1,1)h_1(t) + p_t^*(1,2)h_2(t) \\ &= \left(m_0 + 2\frac{c_3c_4}{\lambda_2-\lambda_1}\left(\frac{1}{\lambda_1}-\frac{1}{\lambda_2}\right)\right)e^{t\lambda_1} + 2\frac{c_3c_4}{\lambda_2-\lambda_1}\left(\frac{1}{\lambda_2}-\frac{1}{\lambda_1}\right), \\ \Gamma_1^*(t) &= \lim_{T\to\infty} \Gamma_1^T(t) = p_t^*(2,1)h_1(t) + p_t^*(2,2)h_2(t) \\ &= m_0(g-\lambda_1)e^{t\lambda_1} + 2\frac{c_3c_4}{\lambda_2-\lambda_1}\left(\frac{\lambda_2-g}{\lambda_2}-\frac{\lambda_1-g}{\lambda_1}\right), \end{aligned} \tag{3.32}$$

where

$$\begin{aligned} g &:= -2\Gamma_2^*, \\ b &:= 2(c_1c_2(2-c_2)-c_5), \\ a &:= 2\Gamma_2^*+\beta, \\ d &:= -1, \\ \lambda_{1\backslash 2} &:= \frac{a+g\pm\sqrt{(a-g)^2+4bd}}{2} = t\frac{\beta\pm\sqrt{(4\Gamma_2^*+\beta)^2-8(c_1c_2(2-c_2)-c_5)}}{2}. \end{aligned} \tag{3.33}$$

As in the MFG case, the third ODE in (3.30) can be solved by plugging in the solution of the previous ones and integrating. Since our interest is into the evolution of the mean and the control function, we omit the calculation for this ODE.

3.5.5 Solution for Asymptotic MFC

The asymptotic version of the problem presented above is given by:

$$\begin{aligned}\min_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}) &= \inf_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^{\infty} e^{-\beta t} f(X_t^{\boldsymbol{\alpha}}, \alpha_t, m^{\boldsymbol{\alpha}})dt\right]\\ &= \inf_{\boldsymbol{\alpha}} \mathbb{E}\left[\int_0^{+\infty} e^{-\beta t}\left(\frac{1}{2}\alpha_t^2 + c_1\left(X_t^{\boldsymbol{\alpha}} - c_2 m^{\boldsymbol{\alpha}}\right)^2 + c_3\left(X_t^{\boldsymbol{\alpha}} - c_4\right)^2 + c_5 (m^{\boldsymbol{\alpha}})^2\right)dt\right],\end{aligned}$$

subject to: $\quad dX_t^{\boldsymbol{\alpha}} = \alpha_t dt + \sigma dW_t, \quad X_0^{\boldsymbol{\alpha}} \sim \mu_0,$

where $m^{\boldsymbol{\alpha}} = \lim_{t\to+\infty} \mathbb{E}\left[X_t^{\alpha}\right].$

Let $V(x)$ be the optimal value function conditioned on $X_0^{\boldsymbol{\alpha}} = x$, i.e.

$$\begin{aligned}V(x) &= \inf_{\boldsymbol{\alpha}} J^x(\boldsymbol{\alpha})\\ &= \inf_{\boldsymbol{\alpha}} \mathbb{E}^x\left[\int_0^{+\infty} e^{-\beta t}\left(\frac{1}{2}\alpha_t^2 + c_1\left(X_t^{\boldsymbol{\alpha}} - c_2 m^{\boldsymbol{\alpha}}\right)^2 + c_3\left(X_t^{\boldsymbol{\alpha}} - c_4\right)^2 + c_5 (m^{\boldsymbol{\alpha}})^2\right)dt\right],\end{aligned}$$

where $\mathbb{E}^x[\cdot] = \mathbb{E}[\cdot|X_0^{\boldsymbol{\alpha}} = x]$.

We consider the following ansatz with its derivative

$$\begin{aligned}V(x) &= \Gamma_2 x^2 + \Gamma_1 x + \Gamma_0,\\ \dot{V}(x) &= 2\Gamma_2 x + \Gamma_1,\\ \ddot{V}(x) &= 2\Gamma_2.\end{aligned}$$

Starting by the MFC-HJB equation (4.12) given in [12], we extended it to the asymptotic case as follows

$$\beta V(x) - H\left(x, \mu^{\boldsymbol{\alpha}}, \alpha\right) - \int_{\mathbb{R}} \frac{\delta H}{\delta \mu}\left(h, \mu^{\boldsymbol{\alpha}}, -\dot{V}(h)\right)(x)\mu^{\boldsymbol{\alpha}}(h)dh = 0,$$

where $m^{\boldsymbol{\alpha}} = \int_{\mathbb{R}} y\mu^{\boldsymbol{\alpha}}(dy)$. We have:

$$\begin{aligned}
H\left(x, \mu^{\boldsymbol{\alpha}}, \alpha\right) &:= \inf_{\alpha} \left\{ \mathcal{A}^X V(x) + f\left(x, \alpha, \mu^{\boldsymbol{\alpha}}\right) \right\} \\
&= \inf_{\alpha} \left\{ \alpha \dot{V}(x) + \frac{1}{2}\sigma^2 \ddot{V}(x) + \frac{1}{2}\alpha^2 + c_1(x - c_2 m^{\boldsymbol{\alpha}})^2 + c_3(x - c_4)^2 + c_5 (m^{\boldsymbol{\alpha}})^2 \right\} \\
&= -\frac{1}{2}\dot{V}(x)^2 + \frac{1}{2}\sigma^2 \ddot{V}(x) + c_1(x - c_2 m^{\boldsymbol{\alpha}})^2 + c_3(x - c_4)^2 + c_5 (m^{\boldsymbol{\alpha}})^2,
\end{aligned}$$

$$\begin{aligned}
\frac{\delta H\left(h, \mu^{\boldsymbol{\alpha}}, \alpha\right)}{\delta \mu} &= \frac{\delta}{\delta \mu}\left(c_1(h - c_2 m^{\boldsymbol{\alpha}})^2 + c_5 (m^{\boldsymbol{\alpha}})^2\right)(x) \\
&= \frac{\delta}{\delta \mu}\left(c_1 \left(h - c_2 \int_{\mathbb{R}} y\mu^{\boldsymbol{\alpha}}(dy)\right)^2 + c_5 \left(\left(\int_{\mathbb{R}} y\mu^{\boldsymbol{\alpha}}(dy)\right)^2\right)\right)(x) \\
&= -2c_1 c_2 x \left(h - c_2 \int_{\mathbb{R}} y\mu^{\boldsymbol{\alpha}}(dy))\right) + 2c_5 x \int_{\mathbb{R}} y\mu^{\boldsymbol{\alpha}}(dy) \\
&= -2c_1 c_2 x(h - c_2 m^{\boldsymbol{\alpha}}) + 2c_5 x m^{\boldsymbol{\alpha}},
\end{aligned}$$

$$\int_{\mathbb{R}} \frac{\delta H}{\delta \mu}\left(h, \mu^{\boldsymbol{\alpha}}, -\dot{V}(h)\right)(x)\mu^{\boldsymbol{\alpha}}(h)dh = -2c_1 c_2 x(m^{\boldsymbol{\alpha}} - c_2 m^{\boldsymbol{\alpha}}) + 2c_5 x m^{\boldsymbol{\alpha}},$$

and finally the HJB equation becomes:

$$\begin{aligned}
\beta V(x) + \frac{1}{2}\dot{V}(x)^2 - \frac{1}{2}\sigma^2 \ddot{V}(x) - c_1(x - c_2 m^{\boldsymbol{\alpha}})^2 - c_3(x - c_4)^2 \\
- c_5 (m^{\boldsymbol{\alpha}})^2 + 2c_1 c_2 x(m^{\boldsymbol{\alpha}} - c_2 m^{\boldsymbol{\alpha}}) - 2c_5 x m^{\boldsymbol{\alpha}} = 0.
\end{aligned}$$

A system of ODEs is obtained by replacing the ansatz and its derivatives in the MFC-HJB and cancelling terms in x^2, and x and constant:

$$\begin{aligned}
\left(\beta\Gamma_2 + 2\Gamma_2^2 - c_1 - c_3\right)x^2 + \left(\beta\Gamma_1 + 2\Gamma_2\Gamma_1 + 2c_1 c_2 m^{\boldsymbol{\alpha}}(2 - c_2) + 2c_3 c_4 - 2c_5 m^{\boldsymbol{\alpha}}\right)x \\
+ \beta\Gamma_0 + \frac{1}{2}\Gamma_1^2 - \sigma^2\Gamma_2 - ({c_1 c_2}^2 + c_5)(m^{\boldsymbol{\alpha}})^2 - {c_3 c_4}^2 = 0.
\end{aligned}$$

An easy computation gives the values

$$\begin{aligned}
\Gamma_2 &= \frac{-\beta + \sqrt{\beta^2 + 8(c_1 + c_3)}}{4}, \\
\Gamma_1 &= \frac{2c_5 m^{\boldsymbol{\alpha}} - 2c_1 c_2 m^{\boldsymbol{\alpha}}(2 - c_2) - 2c_3 c_4}{\beta + 2\Gamma_2}, \\
\Gamma_0 &= \frac{c_5 (m^{\boldsymbol{\alpha}})^2 + {c_3 c_4}^2 + {c_1 c_2}^2 (m^{\boldsymbol{\alpha}})^2 + \sigma^2\Gamma_2 - \frac{1}{2}\Gamma_1^2}{\beta}.
\end{aligned}$$

By plugging the control $\alpha^*(x) = -(2\Gamma_2 x + \Gamma_1)$ into the dynamics of $X_t^{\boldsymbol{\alpha}}$ and taking the expected value, we obtain an ODE for $m_t^{\boldsymbol{\alpha}}$

$$\dot{m}_t^{\boldsymbol{\alpha}} = -(2\Gamma_2 m_t^{\boldsymbol{\alpha}} + \Gamma_1). \tag{3.34}$$

The solution of (3.34) is used to derive m as follows

$$\begin{aligned} m^{\boldsymbol{\alpha}} &= \lim_{t\mapsto\infty} m_t^{\boldsymbol{\alpha}} = \lim_{t\mapsto\infty} \left(-\frac{\Gamma_1}{2\Gamma_2} + \left(m_0 + \frac{\Gamma_1}{\Gamma_2} \right) e^{-2\Gamma_2 t} \right) \\ &= -\frac{\Gamma_1}{2\Gamma_2} = -\frac{2c_5 m^{\boldsymbol{\alpha}} - 2c_1 c_2 m^{\boldsymbol{\alpha}}(2-c_2) - 2c_3 c_4}{2\Gamma_2(\beta + 2\Gamma_2)} \\ m^{\boldsymbol{\alpha}} &= \frac{c_3 c_4}{\Gamma_2(\beta + 2\Gamma_2) + c_5 - c_1 c_2 (2 - c_2)} \end{aligned} \tag{3.35}$$

We remark that the values of $m_t^{\boldsymbol{\alpha}}$ and $\Gamma_1(t)$ obtained in the non-asymptotic case converge to m^α and Γ_1 respectively as t goes to ∞. Therefore, we have obtained that

$$\lim_{t\to\infty} \alpha_t^{*MFC}(x) = \alpha^{*AMFG}(x), \quad \forall x,$$

that is the first part of (3.2) for this LQ MFC problem.

3.5.6 Solution for stationary MFC

The only difference with the derivation above in the case of asymptotic MFC is that m_t^α should be a constant which, from (3.34), should satisfy $2\Gamma_2 m^\alpha + \Gamma_1 = 0$. Therefore, m^α takes the same value as in (3.35), and we deduce

$$\alpha^{*SMFG}(x) = \alpha^{*AMFG}(x), \quad \forall x,$$

that is the second part of (3.2) for this LQ MFC problem .

Chapter 4

Unified Reinforcement Q-Learning for Asymptotic Mean Field Game and Control Problems

In this Chapter, we discuss the results presented in our paper [6]. We introduce a new Reinforcement Learning (RL) algorithm to solve the infinite horizon asymptotic Mean Field Game (MFG) and Mean Field Control (MFC) problems discussed in Chapter 3. Our approach can be described as a unified two-timescale Mean Field Q-learning: The *same* algorithm can learn either the MFG or the MFC solution by simply tuning the ratio of two learning parameters.

The design of the algorithm is derived by the new definition of a MKV MDP: the environment not only receives an action by the agent as in a classical MDP, but also estimates a distribution of the state in order to take into account the mean field feature of the problem. Extending the classical Q-learning algorithm to this framework requires to draw a connection between MFG, MFC, Q-learning and Borkar's two timescale approach [15, 16]. The stochastic approximation of the Q-function presented in Chapter

2 is coupled with a stochastic estimation of the population distribution. While this procedure is sufficient for the MFG case, the MFC framework requires the definition of a new Q-function, that we called the MKV Q-function (see Definition 1), together with the optimal new Bellman equation (see Theorem 4).

In Section 4.1.4, we recast the infinite horizon Asymptotic MFG and MFC problems introduced in Section 3.5 as a two-timescale problem of Borkar's type [15, 16] which provides convergence results. The algorithm is in discrete time and space and it is presented in Section 4.2. In Section 4.3, we show numerical results with comparison to the benchmark case of discrete time and space approximations for continuous time and space linear-quadratic problems for which we have explicit formulas derived in Section 3.5.

4.1 A unified view of learning for MFG and MFC

The definitions of MFG and MFC reveal that the two formulations are very similar and both involve an optimization and a distribution. This leads to the idea of designing an iterative procedure which would update the value function and the distribution. However, in the MFG, the distribution is frozen during the optimization and then a fixed point condition is enforced, whereas in the MFC problem the distribution is directly linked to the control, which implies that it should change instantaneously when the control function is modified. Hence, to compute the solutions using an iterative algorithm, the updates should be done differently for each problem: intuitively, in a MFG, the value function should be updated in an inner loop and the distribution in an outer loop, whereas it should be the converse for MFC. More generally, we can update both functions in turn but at different rates. Then, to compute the MFG solution, the distribution should be updated at a lower rate than the value function. For MFC, it should be the converse. In the rest of this Chapter, we formalize these ideas.

4.1.1 Action-value function in the classical Q-learning setup

As introduced in Chapter 2, Q-learning [73] is one of the most popular methods in RL. Instead of looking at the value function V as in a PDE approach for optimal control, this method is based on the action-value function, also called Q-function, which takes as inputs not only a state x but also an action a. Intuitively, in a standard (non mean-field) MDP, this function quantifies the optimal cost-to-go of an agent starting at x, using action a for the first step and then acting optimally afterwards. In other words, the value of (x, a) is the the cost of using a when in state x, plus the minimal cost possible after that, i.e. the cost induced by using the optimal control; see e.g. [71, Chapter 3] for more details. The definition of the optimal Q-function, denoted by Q^*, is similar to (2.1), up to a change of sign since we consider a cost f and a minimization problem instead of a reward r and a maximisation problem, namely,

$$Q^*(x,a) = \min_{\alpha} \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n f(X_n, \alpha(X_n)) \,\middle|\, X_0 = x, A_0 = a\right].$$

The equivalent of Theorem 1 is provided by the Bellman equation:

$$Q^*(x,a) = f(x,a) + \gamma \sum_{x' \in \mathcal{X}} p(x'|x,a) \min_{a'} Q^*(x',a'), \qquad (x,a) \in \mathcal{X} \times \mathcal{A}.$$

Consequently, Corollary 1 is rewritten as:

$$V^*(x) = \min_{a} Q^*(x,a), \qquad x \in \mathcal{X}.$$

By adopting the optimal action-value function, one can directly recover the optimal control, given by $\arg\min_{a \in \mathcal{A}} Q^*(x,a)$. This allows to design model-free methods.

4.1.2 Action-value function for Asymptotic MFG

In the context of Asymptotic MFG introduced in Section 3.2, we can view the problem faced by an infinitesimal agent among the crowd as an MDP *parameterized* by the population distribution. Hence, given a population distribution μ, standard RL techniques can be applied to compute the Q-function of an infinitesimal agent against this given μ.

Then, the optimal Q-function is defined, for a given μ, by

$$Q^*_\mu(x,a) = \min_\alpha \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu) \,\middle|\, X_0 = x, A_0 = a\right], \tag{4.1}$$

where the cost function $f(x,a,\mu)$ depends on the fixed μ as well as the transition probabilities $p(x'|x,a,\mu)$. Since μ is fixed, as in the classical case, one obtains the the Bellman equation:

$$Q^*_\mu(x,a) = f(x,a,\mu) + \gamma \sum_{x' \in \mathcal{X}} p(x'|x,a,\mu) \min_{a'} Q^*_\mu(x',a'), \qquad (x,a) \in \mathcal{X} \times \mathcal{A}. \tag{4.2}$$

This function characterizes the optimal cost-to-go for an agent starting at state x, using action a for the first step, and then acting optimally for the rest of the time steps, while the population distribution is given by μ (for every time step). Note that $\min_a Q^*_\mu(x,a) = \min_\alpha J^{AMFG}(\alpha;\mu)$ in the notation of Section 3.2.

4.1.3 Action-value function for Asymptotic MFC

For MFC, it is not obvious how to use the same Q-function because, as noticed earlier, the distribution appearing in the definition of MFC is directly linked to the control and not fixed a priori. One possibility is to look at MFC as an MDP on the space of distributions and then to introduce a Q-function which takes a distribution as an input [30, 44, 45, 64].

We take a different route and introduce a new modified Q- function as follows. For an admissible control $\alpha(x)$, we define the MKV- dynamics $p(x'|x,a,\mu^\alpha)$ so that μ^α is the limiting distribution of the associated process (X_n^α). We define the control $\tilde{\alpha}$ by

$$\tilde{\alpha}(x') = \begin{cases} a & \text{if} \quad x' = x, \\ \alpha(x) & \text{for} \quad x' \neq x. \end{cases} \tag{4.3}$$

Remark that $\tilde{\alpha}$ depends on x and a which we omit for the sake of a lighter notation.

Definition 1 (New MKV Q-function) *The MKV Q-function for the asymptotic MFC problem discussed in Section 3.2 is given by*

$$Q^\alpha(x,a) = f(x,a,\mu^{\tilde{\alpha}}) + \mathbb{E}\left[\sum_{n=1}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x, A_0 = a\right]. \tag{4.4}$$

Note that, compared with the Q_μ-function used for MFG, our MFC modified Q-function involves the differences $\Delta_\mu f := f(x,a,\tilde{\mu}) - f(x,a,\mu)$ and $\Delta_\mu p := p(\cdot|x,a,\tilde{\mu}) - p(\cdot|x,a,\mu)$ which play the role of derivatives with respect to the probability distribution in the classical continuous time and space Mean Field Control problems.

In the following steps, we extend the classical results presented in Chapter 2 to the new MKV Q-function (4.4). After introducing the new Bellman equation for the MKV Q-function in Lemma 3, the policy improvement theorem is extended to the new MKV MDP in Theorem 3. Lemma 4 (Corollary 2) verifies the connections among the (optimal) MKV state value function and the corresponing (optimal) state-action value function. Finally, the new Bellman equation for the optimal MKV Q-function is proved in Theorem 4.

Lemma 3 (Bellman eq'n for MKV Q function) *The function $Q^\alpha : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ satisfies the Bellman equation given by*

$$Q^\alpha(x,a) = f(x,a,\mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[Q^\alpha(X_1, \alpha(X_1)) \,\Big|\, X_0 = x, A_0 = a\right], \tag{4.5}$$

Proof:

$$
\begin{aligned}
Q^\alpha(x,a) \overset{\text{(TP)}}{=} & f(x,a,\mu^{\tilde{\alpha}}) \\
& + \gamma \mathbb{E}\left[\mathbb{E}\left[\sum_{n=1}^{\infty} \gamma^{n-1} f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x, A_0 = \alpha(x), X_1 \right] \,\Big|\, X_0 = x, A_0 = a \right] \\
\overset{\text{(MP)}}{=} & f(x,a,\mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[\mathbb{E}\left[\sum_{n=1}^{\infty} \gamma^{n-1} f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_1 \right] \,\Big|\, X_0 = x, A_0 = a \right] \\
= & f(x,a,\mu^{\tilde{\alpha}}) \\
& + \gamma \mathbb{E}\left[f(X_1, \alpha(X_1), \mu^\alpha) + \gamma \mathbb{E}\left[\sum_{n=2}^{\infty} \gamma^{n-2} f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_1 \right] \,\Big|\, X_0 = x, A_0 = a \right] \\
= & f(x,a,\mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[Q^\alpha(X_1, \alpha(X_1)) \,\Big|\, X_0 = x, A_0 = a \right],
\end{aligned}
$$

where *TP* and *MP* stand for tower and Markov property respectively. The last step is justified by observing that the population distribution $\mu^{\tilde{\alpha}}$ based on the modification of α given the pair $(x, \alpha(x))$ is equal to μ^α itself. ■

Lemma 4 *The state value function $V^\alpha : \mathcal{X} \to \mathbb{R}$ is linked to the action-value function $Q^\alpha : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ by*

$$
V^\alpha(x) = Q^\alpha(x, \alpha(x)). \tag{4.6}
$$

Proof:

$$
\begin{aligned}
V^\alpha(x) &= f(x, \alpha(x), \mu^\alpha) + \mathbb{E}\left[\sum_{n=1}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x, A_0 = \alpha(x) \right] \\
&= f(x, \alpha(x), \mu^{\tilde{\alpha}}) + \mathbb{E}\left[\sum_{n=1}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x, A_0 = \alpha(x) \right] \\
&\overset{(4.4)}{=} Q^\alpha(x, \alpha(x))
\end{aligned}
$$

where we used that the modification of α given the pair $(x, \alpha(x))$ is equal to α itself and consequently $\mu^\alpha = \mu^{\tilde{\alpha}}$. ■

Theorem 3 (New Policy improvement for the MKV MDP) *Let $\tilde{\alpha}$ be a policy derived by α*

$$\tilde{\alpha}(s) = \begin{cases} \alpha(s), & \text{for } s \neq x, \\ a, & \text{for } s = x. \end{cases}$$

such that

$$Q^{\alpha}(x, \tilde{\alpha}(x)) < V^{\alpha}(x). \tag{4.7}$$

Then,

$$V^{\tilde{\alpha}}(x') < V^{\alpha}(x') \quad \forall x' \in \mathcal{X}. \tag{4.8}$$

Proof: **Step 1** Show that $V^{\alpha}(x) < V^{\tilde{\alpha}}(x)$.

We observe that

$$\begin{aligned}
&V^{\alpha}(x) > Q^{\alpha}(x, \tilde{\alpha}(x)) \\
&\quad \overset{(4.5)}{=} f(x, \tilde{\alpha}(x), \mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[Q^{\alpha}(X_1, \alpha(X_1)) \,\middle|\, X_0 = x, A_0 = \tilde{\alpha}(x) \right] \\
&\quad \overset{(4.6)}{=} f(x, \tilde{\alpha}(x), \mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[V^{\alpha}(X_1) \,\middle|\, X_0 = x, A_0 = \tilde{\alpha}(x) \right] \\
&\quad \overset{(4.7)}{\geqslant} f(x, \tilde{\alpha}(x), \mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[Q^{\alpha}(X_1, \tilde{\alpha}(X_1)) \,\middle|\, X_0 = x, A_0 = \tilde{\alpha}(x) \right] \\
&\quad \overset{(4.5)}{=} f(x, \tilde{\alpha}(x), \mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[f(X_1, \tilde{\alpha}(X_1), \mu^{\tilde{\alpha}}) + \gamma Q^{\alpha}(X_{t_2}, \alpha(X_{t_2})) \,\middle|\, X_0 = x, A_0 = \tilde{\alpha}(x) \right] \\
&\quad \vdots \\
&\quad \geqslant \mathbb{E}\left[\sum_{n=0}^{k} \gamma^n f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) + \gamma^{k+1} V^{\alpha}(X_{k+1}) \,\middle|\, X_0 = x \right]
\end{aligned}$$

Considering the limit as $k \to \infty$, it follows that

$$V^{\alpha}(x) > \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) \,\middle|\, X_0 = x \right] = V^{\tilde{\alpha}}(x)$$

Step 2 Show that $V^{\alpha}(x') > V^{\tilde{\alpha}}(x') \quad \forall x' \in \mathcal{X} \backslash \{x\}$.

Let define $\tau_x = \min\{n : X_n = x\}$. Then

$$\begin{aligned}
V^\alpha(x') &= \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x'\right] \\
&= \mathbb{E}\left[\sum_{n=0}^{\tau_x - 1} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) + \sum_{n=\tau_x}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x'\right] \\
&= \mathbb{E}\left[\sum_{n=0}^{\tau_x - 1} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x'\right] + \mathbb{E}\left[\sum_{n=\tau_x}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x'\right] \\
&:= T_1 + T_2
\end{aligned}$$

We start analyzing the first term observing that $X_n \neq x$ and $\alpha(X_n) = \tilde{\alpha}(X_n)$ for all $n \leqslant \tau_x - 1$. Then,

$$T_1 = \mathbb{E}\left[\sum_{n=0}^{\tau_x - 1} \gamma^n f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) \,\Big|\, X_0 = x'\right]$$

The analyses of the term T_2 is based on the tower property (TP), the Markov property (MP) and Step 1 (S1). It follows that

$$\begin{aligned}
T_2 &\overset{\text{(TP)}}{=} \mathbb{E}\left[\mathbb{E}\left[\sum_{n=\tau_x}^{\infty} \gamma^n f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_0 = x', X_1, \ldots, X_{\tau_x}\right] \,\Big|\, X_0 = x'\right] \\
&\overset{\text{(MP)}}{=} \mathbb{E}\left[\gamma^{\tau_x} \mathbb{E}\left[\sum_{n=\tau_x}^{\infty} \gamma^{n-\tau_x} f(X_n, \alpha(X_n), \mu^\alpha) \,\Big|\, X_{\tau_x}\right] \,\Big|\, X_0 = x'\right] \\
&= \mathbb{E}\left[\gamma^{\tau_x} V^\alpha(X_{\tau_x}) \,\Big|\, X_0 = x'\right] \\
&\overset{\text{(S1)}}{>} \mathbb{E}\left[\gamma^{\tau_x} V^{\tilde{\alpha}}(X_{\tau_x}) \,\Big|\, X_0 = x'\right]
\end{aligned}$$

Combining the analyses of T_1 and T_2, it follows that

$$\begin{aligned}
V^\alpha(x') &= T_1 + T_2 > \\
&> \mathbb{E}\left[\sum_{n=0}^{\tau_x - 1} \gamma^n f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) \,\Big|\, X_0 = x'\right] + \mathbb{E}\left[\gamma^{\tau_x} V^{\tilde{\alpha}}(X_{\tau_x}) \,\Big|\, X_0 = x'\right] \\
&= \mathbb{E}\left[\sum_{n=0}^{\tau_x - 1} \gamma^n f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) + \gamma^{\tau_x} \sum_{n=\tau_x}^{\infty} \gamma^{n-\tau_x} f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) \,\Big|\, X_0 = x'\right] \\
&= \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n f(X_n, \tilde{\alpha}(X_n), \mu^{\tilde{\alpha}}) \,\Big|\, X_0 = x'\right] \\
&= V^{\tilde{\alpha}}(x')
\end{aligned}$$

■

Corollary 2 *Let $V^* : \mathcal{X} \mapsto \mathcal{R}$ be the optimal state value function defined as $V^*(x) = \min_\alpha V^\alpha(x)$. Then,*

$$V^*(x) = \min_a \min_\alpha Q^\alpha(x,a), \tag{4.9}$$

Remark 3 *The optimal value function $V^*(x) = \min_a Q^*(x,a)$ is equal to $J^{AMFC}(\alpha^*)$ in the notation of Section 3.2).*

Proof: Let $\mathcal{X} = \{x_1, \dots, x_n\}$ and $\mathcal{A} = \{a_0, \dots, a_m\}$ be the state and action spaces.

Step 1 Let α^0 be an initial policy and define α^1 as follows

$$\alpha^1(x) = \begin{cases} \arg\min_a Q^{\alpha^0}(x,a), & \text{if } x = x_1, \\ \alpha_0(x), & \text{o.w.} \end{cases}$$

Then,

$$Q^{\alpha^0}(x_1, \alpha^1(x_1)) \leqslant V^{\alpha^0}(x_1) \overset{(4.8)}{\Longrightarrow} V^{\alpha^1}(x) \leqslant V^{\alpha^0}(x), \quad \forall x$$

Step 2 Consider α^2 defined as follows

$$\begin{aligned} \alpha^2(x) &= \begin{cases} \arg\min_a Q^{\alpha^1}(x,a), & \text{if } x = x_2, \\ \alpha_1(x), & \text{o.w.} \end{cases} \\ &= \begin{cases} \arg\min_a Q^{\alpha^1}(x,a), & \text{if } x = x_2, \\ \arg\min_a Q^{\alpha^0}(x,a), & \text{if } x = x_1, \\ \alpha_0(x), & \text{o.w.} \end{cases} \end{aligned}$$

Then,

$$Q^{\alpha^1}(x_2, \alpha^2(x_2)) \leqslant V^{\alpha^1}(x_1) \overset{(4.8)}{\Longrightarrow} V^{\alpha^2}(x) \leqslant V^{\alpha^1}(x) \leqslant V^{\alpha^0}(x), \quad \forall x$$

Step $\boldsymbol{n}$ Consider α^n defined as follows

$$\begin{aligned}\alpha^n(x) &= \begin{cases} \arg\min_a Q^{\alpha^{n-1}}(x,a), & \text{if } x = x_n, \\ \alpha_{n-1}(x), & \text{o.w.} \end{cases} \\ &= \arg\min_a Q^{\alpha^{k-1}}(x,a), \qquad \text{if } x = x_k, \text{ for } k = 1, \dots, n,\end{aligned}$$

Then,

$$Q^{\alpha^{n-1}}(x_n, \alpha^n(x_n)) \leqslant V^{\alpha^{n-1}}(x_n) \overset{(4.8)}{\Longrightarrow} V^{\alpha^n}(x) \leqslant V^{\alpha^{n-1}}(x) \leqslant V^{\alpha^0}(x), \quad \forall x$$

Step $\boldsymbol{N}$ Since the state and action spaces are finite, the policy can be improved only a finite number of times. In other words, $\exists N > 0$ such that

$$\alpha^N(x) = \arg\min_a Q^{\alpha^N}(x,a), \quad \forall x \in \mathcal{X}$$

and

$$V^{\alpha^N}(x) = Q^{\alpha^N}(x, \alpha^N(x)) = \min_a Q^{\alpha^N}(x,a), \quad \forall x \in \mathcal{X}.$$

Can α^N be still suboptimal? No, by extending Bellman and Dreyfus's Optimality Theorem (1962), [10]. ■

Theorem 4 (New Bellman equation for the optimal MKV Q-function) *The optimal $Q^*(x,a) = \min_\alpha Q^\alpha(x,a)$ satisfies the Bellman equation*

$$Q^*(x,a) = f(x,a,\tilde{\mu}^*) + \gamma \sum_{x' \in \mathcal{X}} p(x'|x,a,\tilde{\mu}^*) \min_{a'} Q^*(x',a'), \qquad (x,a) \in \mathcal{X} \times \mathcal{A}, \quad (4.10)$$

where the optimal control α^* is given by $\alpha^*(x) = \arg\min_a Q^*(x,a)$, the modification $\tilde{\alpha}^*(x)$ is based on the pair (x,a) and $\tilde{\mu}^* := \mu^{\tilde{\alpha}^*}$.

Proof:

$$\begin{aligned}
\text{RHS} &= f(x,a,\mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[\min_{a'} Q^*(X_1,a') \,\middle|\, X_0 = x, A_0 = a\right] \\
&\overset{(4.9)}{=} f(x,a,\mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[V^*(X_1) \,\middle|\, X_0 = x, A_0 = a\right] \\
&\overset{(4.6)}{=} f(x,a,\mu^{\tilde{\alpha}}) + \gamma \mathbb{E}\left[Q^{\alpha^*}(X_1,\alpha^*(X_1)) \,\middle|\, X_0 = x, A_0 = a\right] \\
&\overset{(4.5)}{=} Q^{\alpha^*}(x,a) = Q^*(x,a),
\end{aligned}$$

where the last step is due to what shown in the proof of equation (4.9), i.e. the same policy α^* optimizes V^α and Q^α. ■

4.1.4 Unification through a two timescale approach

The goal is now to design a learning procedure which can approximate, for either MFG or MFC, not only Q but also the corresponding μ. For MFG, the usual fixed point iterations are on the distribution and at each iteration, the best response against this distribution (which can be deduced from the corresponding Q table) is computed. For MFC, the iterations are on the control (here again, it can be deduced from the Q table) and the distribution corresponding to this control is computed at each iteration. Instead of completely freezing the distribution (resp. the control) in the first case (resp. the second case), we can imagine that letting it evolve at a slow rate would still lead to the same limit. In other words, the definitions of MFG and MFC seem to lie at the two opposite sides of a spectrum.

Based on this viewpoint, we consider the following iterative procedure, where both variables (Q and μ) are updated at each iteration but with different rates. Starting from an initial guess $(Q_0, \mu_0) \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{A}|} \times \Delta^{|\mathcal{X}|}$, define iteratively for $k = 0, 1, \dots$:

$$\begin{cases}
\mu_{k+1} = \mu_k + \rho_k^{\mu} \mathcal{P}(Q_k, \mu_k), & (4.11\text{a}) \\
Q_{k+1} = Q_k + \rho_k^{Q} \mathcal{T}(Q_k, \mu_k), & (4.11\text{b})
\end{cases}$$

where

$$\begin{cases}\mathcal{P}(Q,\mu)(x) = (\mu P^{Q,\mu})(x) - \mu(x), \qquad x \in \mathcal{X}, \\ \mathcal{T}(Q,\mu)(x,a) = f(x,a,\mu) + \gamma \sum_{x'} p(x'|x,a,\mu) \min_{a'} Q(x',a') - Q(x,a),\ (x,a) \in \mathcal{X} \times \mathcal{A},\end{cases}$$

and

$$P^{Q,\mu}(x,x') = p(x'|x, \arg\min_a Q(x,a), \mu), \qquad (\mu P^{Q,\mu})(x) = \sum_{x_0} \mu(x_0) P^{Q,\mu}(x_0,x),$$

$P^{Q,\mu}$ is the transition matrix when the population distribution is μ and the agent uses the optimal control according to Q. The learning rates ρ^μ_k and ρ^Q_k are assumed to satisfy usual Robbins-Monro type conditions, namely: $\sum_k \rho^\mu_k = \sum_k \rho^Q_k = +\infty$ and $\sum_k |\rho^\mu_k|^2 = \sum_k |\rho^Q_k|^2 < +\infty$.

If $\rho^\mu_k < \rho^Q_k$, the approximate Q-function evolves faster, while it is the converse if $\rho^\mu_k > \rho^Q_k$. This suggests that these two regimes should converge to different limit points. These ideas have been studied by Borkar [15, 16] in connection with reinforcement learning methods under the name of two timescales approach. More precisely, from Borkar [16, Chapter 6, Theorem 2], we expect to have the following two situations. We assume that the operators $\mathcal{T}$ and $\mathcal{P}$ are Lipschitz continuous, which, as explained in Section 4.1.6, can be obtained from the Lipschitz continuity of f and p in the model, as well as a slight modification of $\mathcal{P}$ to regularize the minimizer.

- **Two timescale approach for MFG**.

  If $\rho^\mu_k/\rho^Q_k \to 0$ as $k \to +\infty$, the system (4.11a)–(4.11b) tracks the ODE system

  $$\begin{cases}\dot{\mu}_t = \mathcal{P}(Q_t,\mu_t), \\ \dot{Q}_t = \frac{1}{\epsilon}\mathcal{T}(Q_t,\mu_t),\end{cases}$$

  where ρ^μ_k/ρ^Q_k is thought of being of order $\epsilon \ll 1$. We consider, for any fixed μ, the ODE

  $$\dot{Q}_t = \frac{1}{\epsilon}\mathcal{T}(Q_t,\mu),$$

and we assume it has a globally asymptotically stable equilibrium Q_μ. In particular, $\mathcal{T}(Q_\mu, \mu) = 0$, meaning by (4.2) that Q_μ is the value function of an infinitesimal agent facing the crowd distribution μ. We further assume that Q_μ is Lipschitz continuous with respect to μ. Convergence to Q_μ can be obtained following standard arguments for Q-learning (see, *e.g.*, [16, Section 10.3]) and the Lipschitz continuity of Q_μ can be guaranteed through Lipschitz continuity of f, p and the minimizer in (4.1). Then the first ODE becomes

$$\dot{\mu}_t = \mathcal{P}(Q_{\mu_t}, \mu_t).$$

Assuming it has a globally asymptotically stable equilibrium μ_∞, this distribution satisfies

$$\mathcal{P}(Q_{\mu_\infty}, \mu_\infty) = 0.$$

This condition implies that μ_∞ and the associated control function given by $\hat{\alpha}(x) = \arg\min_a Q_{\mu_\infty}(x, a)$ form a Nash equilibrium. From [16, Chapter 6, Theorem 2], the system (4.11a)–(4.11b) with discrete time updates also converges to this Nash equilibrium when $\rho^\mu_k / \rho^Q_k \to 0$ as $k \to +\infty$.

- **Two timescale approach for MFC**.

  If $\rho^Q_k / \rho^\mu_k \to 0$ as $k \to +\infty$, the system (4.11a)–(4.11b) tracks the ODE system

$$\begin{cases} \dot{\mu}_t = \dfrac{1}{\epsilon}\mathcal{P}(Q_t, \mu_t), \\ \dot{Q}_t = \mathcal{T}(Q_t, \mu_t), \end{cases}$$

  where ρ^Q_k / ρ^μ_k is thought of being of order $\epsilon \ll 1$. We consider, for any fixed Q, the ODE

$$\dot{\mu}_t = \frac{1}{\epsilon}\mathcal{P}(Q, \mu_t),$$

  and we assume it has a globally asymptotically stable equilibrium μ_Q. In particular, $\mathcal{P}(Q, \mu_Q) = 0$, meaning that μ_Q is the asymptotic distribution of a population in which every agent uses the control $\alpha(x) = \arg\min_a Q(x,a)$. We further assume that μ_Q is Lipschitz continuous with respect to Q. Then the second ODE becomes

$$\dot{Q}_t(x,a) = \mathcal{T}(Q_t(x,a), \widetilde{\mu}_{Q_t}),$$

where $\widetilde{\mu}_{Q_t}$ is defined by (4.3) at (x,a) for $\alpha(\cdot) = \arg\min_{a'} Q_t(\cdot, a')$. This is consistent with the update of Q and what the algorithm proposed in Section 4.2 does. Assuming this ODE has a globally asymptotically stable equilibrium Q_∞, this Q-table satisfies

$$\mathcal{T}(Q_\infty, \widetilde{\mu}_{Q_\infty}) = 0.$$

This last condition means that $Q_\infty = Q^*$ satisfies the MFC Bellman equation (4.10), and that the control $\alpha^*(x) = \arg\min_a Q_\infty(x,a)$ is an MFC optimum for the asymptotic formulation and the induced optimal distribution is μ_{Q_∞}. From [16, Chapter 6, Theorem 2], the system (4.11a)–(4.11b) with discrete time updates also converges to this social optimum when $\rho^Q_k/\rho^\mu_k \to 0$ as $k \to +\infty$.

4.1.5 Stochastic approximation

The above (deterministic) algorithm relies on the operators $\mathcal{P}$, $\mathcal{T}$ which, in many practical situations are not known, for instance because the agent does not know for sure the dynamics or the reward function. In such situations, the agent can only rely on random samples (more details are provided in the next section). The algorithm can be modified to account for such stochastic approximations. Indeed, let us assume that, for any Q, μ, x, a, the agent can know the value $f(x,a,\mu)$ and can sample a realization of the random variable

$$X'_{x,a,\mu} \sim p(\cdot|x,a,\mu).$$

Then, she can compute the realization of the following random variables $\check{\mathcal{T}}_{Q,\mu,x,a}$ and $\check{\mathcal{P}}_{Q,\mu,x,a}$ taking values respectively in $\mathbb{R}$ and $\Delta^{|\mathcal{X}|}$:

$$\check{\mathcal{T}}_{Q,\mu,x,a} = f(x,a,\mu) + \gamma \min_{a'} Q(X'_{x,a,\mu}, a') - Q(x,a),$$

and

$$\check{\mathcal{P}}_{Q,\mu,x,a}(x'') = \mathbf{1}_{\{X'_{x,a,\mu}=x''\}} - \mu(x''), \qquad \forall x'' \in \mathcal{X}.$$

Observe that

$$\mathbb{E}[\check{\mathcal{T}}_{Q,\mu,x,a}] = \sum_{x'} p(x'|x,a,\mu) \Big[f(x,a,\mu) + \gamma \min_{a'} Q(x',a') - Q(x,a) \Big] = \mathcal{T}(Q,\mu)(x,a), \tag{4.14}$$

and

$$\mathbb{E}[\check{\mathcal{P}}_{Q,\mu,x,a}(x'')] = \sum_{x'} p(x'|x,a,\mu) \left(\mathbf{1}_{\{x'=x''\}} - \mu(x'') \right) = p(x''|x,a,\mu) - \mu(x'').$$

If the starting point x comes from a random variable $X \sim \mu$ and if a is chosen to be an optimal action at X according to a given table Q, i.e., $a \in \arg\min_{\mathcal{A}} Q(X,\cdot)$, then we obtain

$$\begin{aligned}
\mathbb{E}[\check{\mathcal{P}}_{Q,\mu,X,\arg\min_a Q(X,a)}(x'')] &= \sum_x \mu(x) \sum_{x'} p(x'|x, \arg\min_a Q(x,a), \mu) \left(\mathbf{1}_{\{x'=x''\}} - \mu(x'') \right) \\
&= \sum_x \mu(x) \left(p(x''|x, \arg\min_a Q(x,a), \mu) - \mu(x'') \right) \\
&= (\mu P^{Q,\mu})(x'') - \mu(x'') \\
&= \mathcal{P}(Q,\mu)(x'').
\end{aligned} \tag{4.15}$$

We can thus replace the deterministic updates (4.11a)–(4.11b) by the following stochastic ones, starting from some initial Q_0, μ_0: for $k = 0, 1, \dots,$

$$\begin{cases} \mu_{k+1}(x) = \mu_k(x) + \rho_k^\mu \check{\mathcal{P}}_{Q_k, \mu_k, X_k, \arg\min_a Q(X_k, a)}(x) & (4.16a) \\ \qquad = \mu_k(x) + \rho_k^\mu \mathcal{P}(Q_k, \mu_k)(x) + \mathbf{P}^k(x), \qquad \forall x \in \mathcal{X} & \\ Q_{k+1}(x, a) = Q_k(x, a) + \rho_k^Q \check{\mathcal{T}}_{Q_k, \mu_k, x, a} & (4.16b) \\ \qquad = Q_k + \rho_k^Q \mathcal{T}(Q_k, \mu_k)(x, a) + \mathbf{T}^k(x, a), \qquad \forall (x, a) \in \mathcal{X} \times \mathcal{A}, & \\ X_k \sim \mu_k, & \end{cases}$$

where we introduced the notation:

$$\mathbf{P}^k(x) = \rho_k^\mu \Big(\check{\mathcal{P}}_{Q_k, \mu_k, X_k, \arg\min_a Q_k(X_k, a)}(x) - \mathcal{P}(Q_k, \mu_k)(x) \Big), \qquad \forall x,$$

and

$$\mathbf{T}^k(x, a) = \rho_k^Q \Big(\check{\mathcal{T}}_{Q_k, \mu_k, x, a} - \mathcal{T}(Q_k, \mu_k)(x, a) \Big), \qquad \forall (x, a),$$

with X_k sampled from μ_k. Note that $\mathbf{T}^k$ and $\mathbf{P}^k$ are martingales by the above remarks, see (4.14)–(4.15). Hence under suitable conditions, we expect convergence to hold by classical stochastic approximation results [16].

However, the procedure (4.16a)–(4.16b) is synchronous (it updates all the coefficients of the Q-table and the distribution at each iteration k) and it requires having access to a generative model, i.e., to a simulator which can provide samples of transitions drawn according to $p(\cdot|x, a, \mu_k)$ for arbitrary state x. In the next section, we propose a procedure which works even with a more restricted setting, which uses episodes: In each episode, the learner is constrained to follow the trajectory sampled by the environment without choosing arbitrarily its state.

4.1.6 Lipschitz property of the 2 scale operators

Generic setting

We modify the original operators using the softmin operator on $\mathbb{R}^{|\mathcal{A}|}$ defined as:

$$\text{soft-min}(z) = \left(\frac{e^{-z_i}}{\sum_j e^{-z_j}}\right)_{i=1,\dots,|\mathcal{A}|} \in \Delta^{|\mathcal{A}|}, \qquad z \in \mathbb{R}^{|\mathcal{A}|}.$$

Intuitively, it gives a probability distribution on the indices $i = 1,\dots,|\mathcal{A}|$ which has higher values on indices whose corresponding values are closer to be a minimum. In particular, the elements of $\min\{i = 1,\dots,|\mathcal{A}| : z_i = \arg\min_j z_j\}$ have equal weight and this weight is the largest among $\left(\frac{e^{-z_i}}{\sum_j e^{-z_j}}\right)_{i=1,\dots,|\mathcal{A}|}$. We recall that the function soft-min is Lipschitz continuous for the 2-norm. Denoting by L_s its Lipschitz constant, it means that

$$\|\text{soft-min}(z) - \text{soft-min}(z')\|_2 \leqslant L_s\|z - z'\|_2, \qquad z, z' \in \mathbb{R}^{|\mathcal{A}|}.$$

Moreover, since $|\mathcal{A}|$ is finite, all the norms on $\mathbb{R}^{|\mathcal{A}|}$ are equivalent so there exists a positive constant $c_{2,\infty}$ such that

$$\|\text{soft-min}(z) - \text{soft-min}(z')\|_\infty \leqslant L_s c_{2,\infty}\|z - z'\|_\infty, \qquad z, z' \in \mathbb{R}^{|\mathcal{A}|}.$$

To alleviate the notation, we will write $Q(x) := (Q(x,a))_{a\in\mathcal{A}}$ for any $Q \in \mathbb{R}^{|\mathcal{X}|\times|\mathcal{A}|}$. We also introduce a more general version $\underline{p}$ of the transition kernel p, which can take as an input a probability over actions instead of a single action: for $x, x' \in \mathcal{X}, \nu \in \Delta^{|\mathcal{A}|}, \mu \in \Delta^{|\mathcal{X}|}$,

$$\underline{p}(x'|x,\nu,\mu) = \sum_a \nu(a)p(x'|x,a,\mu).$$

Intuitively, this is the probability for a agent at x to move to x' when the population distribution is μ and the agent picks a random action following the distribution ν.

We now consider the following iterative procedure, which is a slight modification of (4.11a)–(4.11b). Here again, both variables (Q and μ) are updated at each iteration but with different rates. Starting from an initial guess $(Q_0, \mu_0) \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{A}|} \times \Delta^{|\mathcal{X}|}$, define iteratively for $k = 0, 1, \ldots$:

$$\begin{cases} \mu_{k+1} = \mu_k + \rho_k^{\mu} \underline{\mathcal{P}}(Q_k, \mu_k), & (4.17a) \\ Q_{k+1} = Q_k + \rho_k^{Q} \mathcal{T}(Q_k, \mu_k), & (4.17b) \end{cases}$$

where

$$\begin{cases} \mathcal{T}(Q,\mu)(x,a) = f(x,a,\mu) + \gamma \sum_{x'} p(x'|x,a,\mu) \min_{a'} Q(x',a') - Q(x,a), \ (x,a) \in \mathcal{X} \times \mathcal{A}, \\ \underline{\mathcal{P}}(Q,\mu)(x) = (\mu \underline{P}^{Q,\mu})(x) - \mu(x), \qquad x \in \mathcal{X}, \end{cases}$$

with

$$\underline{P}^{Q,\mu}(x,x') = \underline{p}(x'|x, \text{soft-min}\, Q(x), \mu), \qquad \text{and} \qquad (\mu \underline{P}^{Q,\mu})(x) = \sum_{x_0} \mu(x_0) \underline{P}^{Q,\mu}(x_0, x),$$

is the transition matrix when the population distribution is μ and the agent uses an approximately optimal randomized control according to the soft-min of Q.

Lemma 5 *Assume that f is Lipschitz continuous with respect to μ and that $\underline{p}$ is Lipschitz continuous with respect to ν and μ. Then*

- *the operator $\mathcal{T}$ is Lipschitz continuous w.r.t. μ (with a Lipschitz constant possibly depending on $\|Q\|_\infty$), and Lipschitz continuous in Q (uniformly in μ);*
- *the operator $\underline{\mathcal{P}}$ is Lipschitz continuous in both variables.*

If p is independent of μ, then both $\mathcal{T}$ and $\underline{\mathcal{P}}$ are Lipschitz continuous.

Proof: Let us denote by L_p and L_f the Lipschitz constants of p and f respectively. Let $(Q,\mu), (Q',\mu') \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{A}|} \times \Delta^{|\mathcal{X}|}$. We first consider $\mathcal{T}$. We have

$$\begin{aligned} \|\mathcal{T}(Q,\mu) - \mathcal{T}(Q',\mu)\|_\infty &\leqslant \gamma \sum_{x'} \max_{x,a} p(x'|x,a,\mu) \left| \min_{a'} Q(x',a') - \min_{a'} Q'(x',a') \right| + \|Q - Q'\|_\infty \\ &\leqslant (\gamma + 1) \|Q - Q'\|_\infty . \end{aligned}$$

Moreover,

$$\begin{aligned}\|\mathcal{T}(Q,\mu)-\mathcal{T}(Q,\mu')\|_\infty &\leqslant |f(x,a,\mu)-f(x,a,\mu')| \\ &\quad + \gamma\sum_{x'}|p(x'|x,a,\mu)-p(x'|x,a,\mu')|\,|\min_{a'} Q(x',a')| \\ &\leqslant (L_f+\gamma L_p\|Q\|_\infty)|\mathcal{X}|\|\mu-\mu'\|_\infty,\end{aligned}$$

where L_f and L_p are respectively the Lipschitz constants of f and p with respect to μ. If p is independent of μ, we obtain

$$\|\mathcal{T}(Q,\mu)-\mathcal{T}(Q,\mu')\|_\infty \leqslant L_f\|\mu-\mu'\|_\infty.$$

We then show that the operator $\underline{\mathcal{P}}$ is Lipschitz continuous. We have

$$\begin{aligned}&\|\underline{\mathcal{P}}(Q,\mu)-\underline{\mathcal{P}}(Q,\mu')\|_\infty \\ &\leqslant \|\mu\underline{P}^{Q,\mu}-\mu'\underline{P}^{Q,\mu'}\|_\infty+\|\mu-\mu'\|_\infty \\ &\leqslant \left\|\sum_x \Big(\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu)\mu(x)-\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu')\mu'(x)\Big)\right\|_\infty \\ &\qquad +\|\mu-\mu'\|_\infty.\end{aligned}$$

For the first term, we note that, for every $x\in\mathcal{X}$,

$$\begin{aligned}&\left\|\Big(\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu)\mu(x)-\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu')\mu'(x)\Big)\right\|_\infty \\ &\leqslant \left\|\Big(\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu)-\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu')\Big)\mu(x)\right\|_\infty \\ &\qquad+\left\|\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu')\Big(\mu(x)-\mu'(x)\Big)\right\|_\infty \\ &\leqslant (L_p+1)\left\|\mu-\mu'\right\|_\infty,\end{aligned}$$

where we used the fact that discrete probability measures are non-negative and bounded by 1.

Moreover, we have

$$\begin{aligned}\|\underline{\mathcal{P}}(Q,\mu)-\underline{\mathcal{P}}(Q',\mu)\|_\infty &\leqslant \|\mu(\underline{P}^{Q,\mu}-\underline{P}^{Q',\mu'})\|_\infty \\ &\leqslant \sum_x \|\underline{p}(\cdot|x,\text{soft-min}\,Q(x),\mu)-\underline{p}(\cdot|x,\text{soft-min}\,Q'(x),\mu)\|_\infty \\ &\leqslant \sum_x L_p\|\,\text{soft-min}\,Q(x)-\text{soft-min}\,Q'(x)\|_\infty \\ &\leqslant |\mathcal{X}|\,L_p\,L_s\,c_{2,\infty}\,\|Q-Q'\|_\infty,\end{aligned}$$

which concludes the proof. ■

Application to a discrete model for the LQ problem

Recall that the continuous linear-quadratic model we consider is defined by (3.3). Here, we propose a finite space MDP which approximates the dynamics of a typical agent in this continuous LQ model. We consider that the action space is given by $\mathcal{A}=\{a_0=-1,a_1=-1+\Delta_.,\dots,a_{N_\mathcal{A}}=1-\Delta_.,a_{N_\mathcal{A}}=1\}$ and the state space by $\mathcal{X}=\{x_0=x_c-2,x_1=x_c-2-\Delta_.,\dots,x_{N_\mathcal{X}-1}=x_c+2-\Delta_.,x_{N_\mathcal{X}}=x_c+2\}$, where x_c is the center of the state space. The step size for the discretization of the spaces $\mathcal{X}$ and $\mathcal{A}$ is given by $\Delta_.=\sqrt{\Delta t}=10^{-1}$.

Consider the transition probability:

$$\begin{aligned}p(x,x',a,\mu)=&\mathbb{P}(Z^{x+a,\Delta t}\in[x'-\Delta_./2,x'+\Delta_./2])\\ =&\Phi_{x+a,\sigma^2\Delta t}(x'+\Delta_./2)-\Phi_{x+a,\sigma^2\Delta t}(x'-\Delta_./2),\end{aligned}$$

where $Z\sim\mathcal{N}(x+a,\sigma^2\Delta t)$ and $\Phi_{x+a,\sigma^2\Delta t}$ is the cumulative distribution function of the $\mathcal{N}(x+a,\sigma^2\Delta t)$ distribution. Moreover, consider that the one-step cost function is given by $f(x,a,\mu)\Delta t$ with

$$f(x,a,\mu)=\frac{1}{2}a^2+c_1\left(x-c_2\sum_{\xi\in S}\mu(\xi)\right)^2+c_3\,(x-c_4)^2+c_5\left(\sum_{\xi\in S}\mu(\xi)\right)^2,\qquad b(x,a,\mu)=a,$$

For simplicity, we write $\bar{\mu}=\sum_{\xi\in S}\mu(\xi)$.

Lemma 6 *In this model, f is Lipschitz continuous with respect to μ and $\underline{p}$ is Lipschitz continuous with respect to ν and μ*

Proof:

We start with f. For the μ component, we have:

$$\begin{aligned}
|f(x,a,\mu) - f(x,a,\mu')| &\leqslant c\left|(x - c_2\bar{\mu})^2 - (x - c_2\bar{\mu}')^2\right| + c\left|(\bar{\mu})^2 - (\bar{\mu}')^2\right| \\
&\leqslant c\,(\bar{\mu}' - \bar{\mu}) \cdot (2x + (\bar{\mu}' - \bar{\mu})) + c(\bar{\mu} - \bar{\mu}')(\bar{\mu} + \bar{\mu}') \\
&\leqslant c \max_{x \in S} \|x\|_\infty \, (\bar{\mu}' - \bar{\mu}) \\
&\leqslant c \max_{x \in S} \|x\|_\infty \sum_{x \in S} (\mu'(x) - \mu(x)) \\
&\leqslant c \max_{x \in S} \|x\|_\infty \, |S| \, \|\mu' - \mu\|_\infty,
\end{aligned}$$

where $c > 0$ is a constant depending only on the parameters of the model and whose value may change from line to line.

Then we consider $\underline{p}$. It is independent of μ in this model. For the action component, we have:

$$
\begin{aligned}
&|\underline{p}(x,x',\nu,\mu) - \underline{p}(x,x',\nu',\mu)| \\
&= \Bigg| \sum_a \nu(a) \Big(\Phi_{x+a,\sigma^2\Delta t}(x' + \Delta_./2) - \Phi_{x+a,\sigma^2\Delta t}(x' - \Delta_./2) \Big) \\
&\qquad - \sum_{a'} \nu'(a') \Big(\Phi_{x+a',\sigma^2\Delta t}(x' + \Delta_./2) - \Phi_{x+a',\sigma^2\Delta t}(x' - \Delta_./2) \Big) \Bigg| \\
&= \left| \sum_a \left(\nu(a)\Phi_{x+a,\sigma^2\Delta t}(x' + \Delta_./2) - \nu'(a)\Phi_{x+a,\sigma^2\Delta t}(x' + \Delta_./2) \right) \right| \\
&\qquad + \left| \sum_a \Big(\nu(a)\Phi_{x+a,\sigma^2\Delta t}(x' - \Delta_./2) \Big) - \nu'(a)\Phi_{x+a,\sigma^2\Delta t}(x' - \Delta_./2) \Big) \right| \\
&= \int_{-\infty}^{x'+\Delta_./2} \frac{1}{\sigma\sqrt{2\pi\Delta t}} \left| \sum_a (\nu(a) - \nu'(a)) e^{-\frac{(y-(x+a))^2}{2\sigma^2\Delta t}} \right| dy \\
&\qquad + \int_{-\infty}^{x'-\Delta_./2} \frac{1}{\sigma\sqrt{2\pi\Delta t}} \left| \sum_a (\nu(a) - \nu'(a)) e^{-\frac{(y-(x+a))^2}{2\sigma^2\Delta t}} \right| dy \\
&\leqslant c\|\nu - \nu'\|_\infty,
\end{aligned}
$$

where c is a constant depending only on the model (and in particular on the state space, the action space and Δt).

■

4.2 Reinforcement Learning Algorithm

As recalled in Chapter 2, RL studies the algorithms to solve a Markov decision process (MDP) based on trials and errors. An MDP can be described through the interactions of an agent with an environment. At each time n, the agent observes its current state $X_n \in \mathcal{X}$ and chooses an action $A_n \in \mathcal{A}$. Due to the agent's action, the environment provides the new state of the agent X_{n+1} and incurs a cost f_{n+1}. The goal of the agent is to find an optimal strategy (or policy) π^* which assigns to each state an action in order to minimize the aggregated discounted costs. The idea is then to design methods which allow the agent to learn (an approximation of) π^* by making repeated use of the environment's outputs but without knowing how the environment produces the new state and the associated cost. A detailed overview of this field can be found in [71] (although RL methods are often presented with reward maximization objectives, we consider cost minimization problems for the sake of consistency with the MFG literature).

As presented in Section 4.1.1, the optimal strategy can be derived from the optimal action-value function. However Q^* is a priori unknown. In order to learn Q^* by trials and errors, an approximate version Q of the table Q^* is constructed through an iterative procedure. At each step, an action is taken, which leads to a cost and to a new state. On the one hand, it is interesting to act efficiently in order to avoid high costs, and on the other hand it is important to improve the quality of the table Q by trying actions and states which have not been visited many times so far. This is the so-called exploitation–exploration trade-off. The trade-off between exploration of the unknown environment and exploitation of the currently available information can be taken care of by an ϵ-greedy policy based on Q. The algorithm chooses the action that minimizes the immediate cost with probability $1-\epsilon$, and a random action otherwise, as in (2.2) with an $\arg\min$.

4.2.1 U2-MF-QL : Unified Two Timescales Mean Field Q-learning

In order to apply the RL paradigm to mean field problems, the first step consists in defining the connection between these two frameworks. In a MFG (resp. a MFC) the goal of a typical agent is to find the pair $(\hat{\alpha}, \hat{\mu})$ (resp. (α^*, μ^*)) where $\hat{\alpha} : \mathcal{X} \mapsto \mathcal{A}$ (resp. $\alpha^* : \mathcal{X} \mapsto \mathcal{A}$) represents the equilibrium (resp. optimal) strategy which assigns at each state the equilibrium (resp. optimal) action in order to minimize the aggregated discounted costs and $\hat{\mu}$ (resp. μ^*) is the ergodic distribution of the population at equilibrium (resp.

optimum). The traditional definition of an MDP based on the agent–environment pair is augmented with the distribution of the population. In this new framework, the agent corresponds to the representative player of the mean field problem.

We now define the type of environment to which the agent is assumed to have access. A key difference with prior works on RL for mean field problems is that we do not assume that agent can witness the evolution of the population's distribution. Instead, the environment estimates the distribution of the population by exploiting the symmetry property of the problem. Indeed, when the system is at equilibrium the law of the representative player matches the distribution of the population. As showed in the diagram of Figure 4.1, at each time n, the agent observes its current state $X_n \in \mathcal{X}$ and then chooses an action $A_n \in \mathcal{A}$. An approximation of the distribution μ_n is computed by the environment based on the observed states of the representative player. Provided with the choice of the action and the estimate of the distribution, the environment generates the new state of the agent X_{n+1} and assigns a cost f_{n+1}.

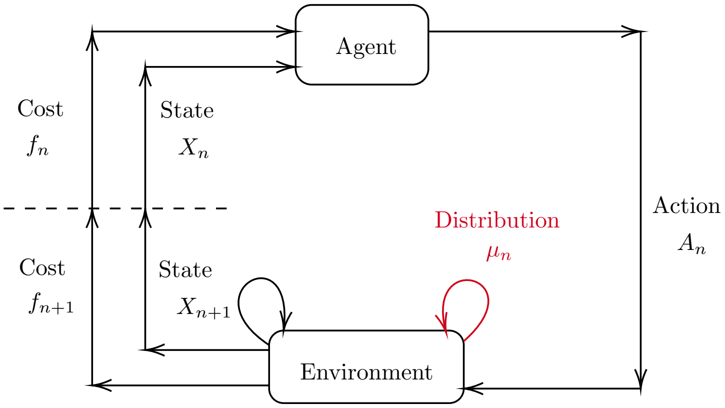

Figure 4.1: MDP with Mean Field interactions: Interaction of the representative agent with the environment. When the current state of the representative agent is X_n, given an action A_n, the environment produces an estimate of the distribution μ_n, the new state X_{n+1} and incurs a cost f_{n+1} calculated by starting from the current state of the environment X_n and using the transition controlled by A_n and parameterized by μ_n.

The algorithm is designed to solve infinite horizon problems through an online approach, i.e. interacting with the environment. The learning procedure is based on splitting the infinite horizon in successive episodes in order to promote the exploration of the environment. The first episode is initialized based on the initial distribution of the representative player. Within a given episode, the agent updates her strategy at each learning step aiming to optimize the expected aggregated cost based on the current estimate of the distribution of the population μ_n. Changes in the representative player's strategy have an effect on the population requiring to update μ_n accordingly. After an assigned number of steps T, the episode is terminated. A new episode is initialized based on the current version of the environment represented by the estimate of the population obtained at the last time point of the previous episode. One may think at the initialization step as a change in the choice of the representative player who provides the data flow. As the number of episodes increases, one expects the distribution of the representative player to converge to the limiting distribution. Within a given learning step, the environment computes an estimate of μ_n based on the current state of the agent X_n, provides the next state X_{n+1} and assigns the cost f_{n+1} given the triple (X_n, A_n, μ_n). In other words, the environment consists of the dynamics of the agent and the cost structure. The case of our interest corresponds to the one in which the dynamics of the agent and the cost structure are unknown. In this way, introducing the RL paradigm is equivalent to define a data driven approach to solve mean field models which may scale their applicability to real world problems.

In contrast with standard Q-learning, since in the mean field framework the cost function also depends on the distribution of the population, the goal here consists in learning the optimal strategy along with the corresponding ergodic distribution of the population, i.e. $(\hat{\alpha}, \hat{\mu})$ in the MFG setting and (α^*, μ^*) in the MFC setting. Based on the intuition provided in Sections 4.1.4, 4.1.5 related to the two timescale approach, we propose Algorithm 2. At each step, we update the Q-table at the observed state-action pair $Q(X_n, A_n)$. With a different learning rate, the estimate of the distribution is updated based on the operator $\boldsymbol{\delta} : \mathcal{X} \mapsto \Delta^{|\mathcal{X}|}$ which maps the next observed state $X_{n+1} \in \mathcal{X}$ to the corresponding one-hot vector measure. To be specific, we identify the simplex $\Delta^{|\mathcal{X}|}$ with the subset $\left\{[\mu(x_i)]_{i=0,\dots,|\mathcal{X}|-1} \,:\, \mu(x_i) \in [0,1] \,\text{and}\, \sum_i \mu(x_i) = 1\right\}$ of $\mathbb{R}^{|\mathcal{X}|}$. Then $\boldsymbol{\delta}$ is the function which associates to each element of $\mathcal{X} = \{x_0, \dots, x_{|\mathcal{X}|-1}\}$ the corresponding element of the canonical basis $(e_0, \dots, e_{|\mathcal{X}|-1})$ of $\mathbb{R}^{|\mathcal{X}|}$, i.e., for each $i = 0, \dots, |\mathcal{X}| - 1$, $\boldsymbol{\delta}(x_i) = e_i$, which is an element of $\Delta^{|\mathcal{X}|}$ by the above identification. In order to learn the limiting distribution of the population through successive learning episodes, an estimate μ_{n_i} is computed for each step n_i based on the sample $X^k_{n_i}$ collected from episodes $k = 1, 2, \dots$. This approach attempts to minimize the correlation of the sampled states. The update rule presented in algorithm 2 allocates more weight on the most recent samples allowing to forget progressively the initial sample that were obtained by a distribution far from the limiting one. At convergence, one may expect each μ_{n_i} to be an estimate of the limiting distribution.

The algorithm returns both an approximation μ^k_T of the distribution and an approximation Q^k of the Q-function, from which an approximation of the optimal control can be recovered as $x \mapsto \arg\min_{a \in \mathcal{A}} Q^k(x, a)$.

Algorithm 2 Unified Two Timescales Mean Field Q-learning - Tabular version

```
Require: $T$ : number of time steps in a learning episode,
    $\mathcal{X} = \{x_0, \dots, x_{|\mathcal{X}|-1}\}$ : finite state space,
    $\mathcal{A} = \{a_0, \dots, a_{|\mathcal{A}|-1}\}$ : finite action space,
    $\mu_0$ : initial distribution of the representative player,
    $\epsilon$ : parameter related to the $\epsilon-$greedy policy,
    $tol_\mu$, $tol_Q$ : break rule tolerances.
Initialization: episode $k = 0$, $Q^k(x,a) = 0$ for all $(x,a) \in \mathcal{X} \times \mathcal{A}$, $\mu_n^k = \left[\frac{1}{|\mathcal{X}|}, \dots, \frac{1}{|\mathcal{X}|}\right]$ for $n = 0, \dots, T$
repeat
    Episode $k = k + 1$
    Initialization: Sample $X_0^k \sim \mu_T^{k-1}$ and set $Q^k \equiv Q^{k-1}$
    for $n \leftarrow 0$ to $T - 1$ do
        Update $\mu$:
            $$\mu_n^k = \mu_n^{k-1} + \rho_k^\mu(\boldsymbol{\delta}(X_n^k) - \mu_n^{k-1}) \quad (4.18)$$
        where $\boldsymbol{\delta}(X_n^k) = \left[\mathbf{1}_{x_0}(X_n^k), \dots, \mathbf{1}_{x_{|\tilde{\mathcal{X}}|-1}}(X_n^k)\right]$
        Choose action $A_n^k$ using the $\epsilon$-greedy policy derived from $Q^k(X_n^k, \cdot)$
        Observe cost $f_{n+1} = f(X_n^k, A_n^k, \mu_n^k)$ and state $X_{n+1}^k$ provided by the environment
        Update $Q$:
            $$Q^k(X_n^k, A_n^k) = = Q^k(X_n^k, A_n^k) + \rho^Q_{k,n,X_n^k,A_n^k}[f_{n+1} + \gamma \min_{a' \in \mathcal{A}} Q^k(X_{n+1}^k, a') - Q^k(X_n^k, A_n^k)] \quad (4.19)$$
    end for
until $\delta(\mu_T^{k-1}, \mu_T^k) \leqslant tol_\mu$ and $\|Q^k - Q^{k-1}\|_{1,1} < tol_Q$
return $(\mu^k, Q^k)$
```

The Unified Two Timescales Mean Field Q-learning (U2-MF-QL) algorithm represents a unified approach to solve mean field problems. On the one hand, by choosing the learning rate for the distribution of the population slower than the one for the Q-table, we obtain the solution to the MFG problem. Similarly to the scheme presented in Section 4.1.5, the iterations in Q perceive the quantity μ as quasi-static mimicking the freezing of the flow of measures characteristic in the solving scheme of a MFG. In particular, the update rule (4.19) of the algorithm represents a stochastic version of the Bellman equation (4.2) for the optimal Q-function Q^*_μ. On the other hand, by choosing the learning rate for the mean-field term faster than the one for the Q-table, we obtain the solution to the MFC problem. Indeed, this choice of the parameters guarantees that the distribution changes instantaneously for each variation of the control function (Q-table) replicating the structure of the MFC problem. Due to the choice $\rho^Q << \rho^\mu$, the Q function and the corresponding control function $\alpha(x) = \arg\min_{a'} Q(x, a')$ behave as frozen. The update rule (4.18) for the distribution is based on the pair (x, a) visited at each time step implying the learning of $\mu^{\tilde{\alpha}}$ as described in Section 4.1.3. The quantity $\mu^{\tilde{\alpha}}$ is passed as input to the running cost defining the update rule (4.19) as a stochastic estimate of the Bellman equation for the new optimal MKV Q-function given by equation (4.10).

4.2.2 Application to continuous problems

Although it is presented in a setting with finite state and action spaces, the application of the algorithm U2-MF-QL can be extended to continuous problems. Such adaptation requires truncation and discretization procedures to time, state and action spaces which should be calibrated based on the specific problem.

In practice, the learning episode will correspond to a uniform discretization $\tau = \{t_n\}_{n\in\{0,\dots,|\tau|-1\}}$ of a time interval $[0, T]$ with T large enough. The environment will provide the new state and reward at these discrete times. We assume that T is large enough to reach the ergodic regime. The continuous state space will be represented as the disjoint union of equally sized neighbors. Each of them will be identified by its centroid and it will correspond to a row of the Q table. Likewise, actions will be provided to the environment in a finite set $\mathcal{A} = \{a_0, \ldots, a_{|\mathcal{A}|-1}\} \subset \mathbb{R}^k$, and the distribution μ will be estimated on the set of centroids $\mathcal{X} = \{x_0, \ldots, x_{|\mathcal{X}|-1}\} \subset \mathbb{R}^k$ identifying $\mu(x_i)$ as the probability of the neighbor centered in x_i. Then Algorithm 2 is ran on those spaces.

We will use the benchmark linear-quadratic models given in continuous time and space for which we have explicit formulas given in Section 3.5. In that case, we use an Euler discretization. We do not address here the error of approximation since the purpose of this comparison with a benchmark is mainly for illustration.

4.3 Numerical experiments

We present the results obtained by applying the U2-MF-QL algorithm to the mean field problems discussed in Section 3.5. These results show how the algorithm successfully learns the MFG solution or the MFC solution based on simply tuning the learning rates. Moreover, this shows that the algorithm manages to solve problems defined on continuous time and continuous state, action spaces even though it is conceived for discrete problems. Such applications require to apply truncation and discretization procedures to time, state and actions which should be calibrated on a problem base.

We consider the problem defined by the choice of parameters: $c_1 = 0.25$, $c_2 = 1.5$, $c_3 = 0.50$, $c_4 = 0.6$, $c_5 = 5$, discount parameter $\beta = 1$ and volatility $\sigma = 0.3$. The infinite time horizon is truncated at time $T = 20$. The continuous time is discretized using step $\Delta t = 10^{-2}$. Recall that γ in the discrete time setting corresponds to $e^{-\beta \Delta t}$ in the continuous time setting. The action space is given by $\mathcal{A} = \{a_0 = -1, \ldots, a_{N_\mathcal{A}} = 1\}$ and the state space by $\mathcal{X} = \{x_0 = -2 + x_c, \ldots, x_{N_\mathcal{X}} = 2 + x_c\}$, where x_c is the center of the state space. The step size for the discretization of the spaces $\mathcal{X}$ and $\mathcal{A}$ is given by $\Delta_{\cdot} = \sqrt{\Delta t} = 10^{-1}$. The state space $\mathcal{X}$ and the action space $\mathcal{A}$ have been chosen large enough to make sure that the state is within the boundary most of the time. In practice, this would have to be calibrated in a model-free way through experiments. In this example, for the numerical experiments, we used the knowledge of the model. In particular, we choose $x_c = 0.5$ for both examples. Note that if the problem under consideration is posed on finite spaces, this issue does not occur since the domain is fixed. The exploitation-exploration trade off is tackled on each episode using an $\epsilon-$greedy policy, see (2.2). In particular, the value of ϵ is fixed to 0.15.

We present the following results for both the MFG and MFC benchmark examples:

1. learning rates analyses;
2. learning of the controls and the ergodic distribution;
3. empirical error analyses;
4. empirical analyses of the stopping criteria.

4.3.1 Learning rates analyses

It is important to observe that even if in the MFC case the choice of ρ^{μ}_{k} below does not satisfy the classical Robbins-Monro summability condition recalled in Section 4.1.4, the numerical convergence of the algorithm is obtained suggesting that these requirements may be relaxed in this framework. Failing in satisfying these conditions generates a noisy approximation of the distribution μ in the MFC problem. However, averaging over the last 10k episodes allows to minimize such noise as showed in the Figures below. Based on the theoretical results given in [35], we define the learning rates appearing in Algorithm 2 as follows:

$$\rho^{Q}_{k,n,x,a} = \frac{1}{(1+\#|(x,a,k,n)|)^{\omega^Q}}, \qquad \rho^{\mu}_{k} = \frac{1}{(1+k)^{\omega^\mu}}, \tag{4.20}$$

where $\#|(x,a,k,n)|$ is the number of times that the algorithm visited state x and performed action a until episode k and time t_n. The exponent ω^Q can take values in $(\frac{1}{2},1)$. The value of ω^μ is chosen depending on the value of ω^Q and the cooperative or non-cooperative nature of the problem we want to solve. The algorithm is run over 80×10^3 episodes over the interval $[0,T]$.

Figures 4.2, 4.3, 4.4, 4.5: comparison of the learning rates. The solution of the MFG benchmark is reached based on the choice $(\omega^Q,\omega^\mu) = (0.55, 0.85)$, such that $\rho^\mu < \rho^Q$. As pointed out in section 4.1.4, by satisfying this relation the Q-function evolves faster than the estimation of the distribution mimicking the solving scheme of a MFG. On the other hand, the solution of the MFC benchmark can be obtained by opting for the pair of parameters $(\omega^Q,\omega^\mu) = (0.65, 0.15)$ such that $\rho^\mu > \rho^Q$. In figures 4.2, 4.3, 4.4, 4.5, we suppose that $\#|(x,a,k,1)| = k$. The $x-$axis refers to the episode. The $y-$axis represents the rate evaluated at episode k.

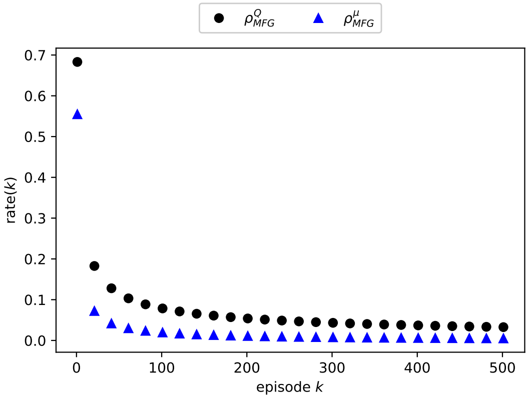

Figure 4.2: MFG: learning rates over the first 500 episodes

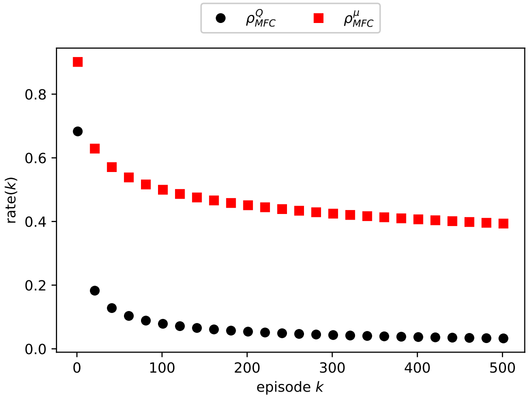

Figure 4.3: MFC: learning rates over the first 500 episodes

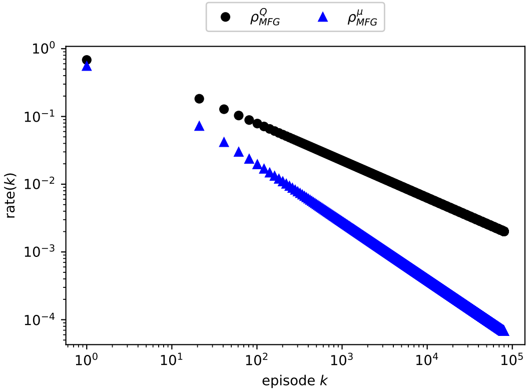

Figure 4.4: MFG: learning rates over 80×10^3 episodes

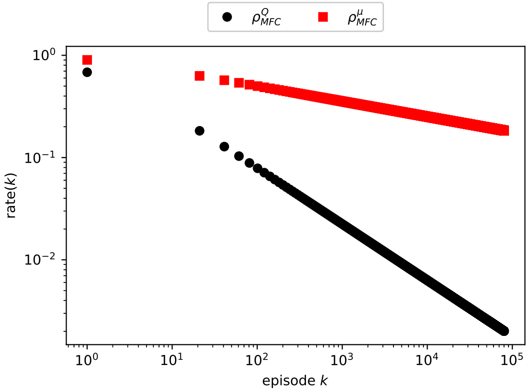

Figure 4.5: MFC: learning rates over 80×10^3 episodes

Figures 4.6, 4.7, 4.8, 4.9: Empirical check of the two timescale conditions. The U2-MF-QL algorithm is based on an asynchronous QL approach which makes use of different learning rates for each $Q(x, a)$ based on the number of visits to the relative state-action pair. An empirical check of the two timescale conditions presented in section 4.1.4 is presented in the following plots. The number of visits to each state depends on their proximity to the mean of the ergodic distribution. As a proof of concept, the learning rates for two different states in the MFG and MFC frameworks are analyzed after 80×10^3 learning epochs. The plots on the left are relative to the state on the left bound of $\mathcal{X}$, while the plots on the right are relative to the closest state to the theoretical mean. Each plot shows the value of the learning rates ρ^μ_k and $\rho^Q_{k,n,x,a}$ together with the counter of visits to each pair (x, a). The two timescale conditions are satisfied in each plot. The number of visits changes from order 10^2 for the state on the border of $\mathcal{X}$ to order 10^7 for the closest state to the ergodic mean. The $x-$axis refers to the action. The left $y-$axis represents the learning rate. The right $y-$axis represents the counter of visits for each state-action pair.

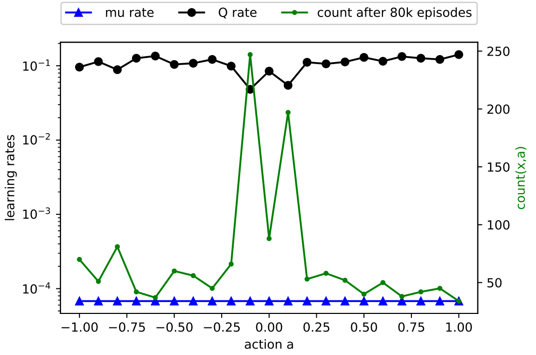

Figure 4.6: MFG: comparison learning rates for state $x = -1.50$

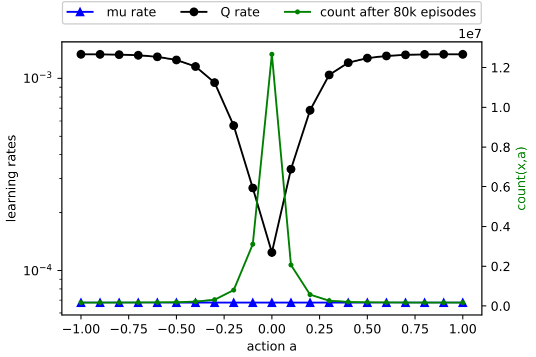

Figure 4.7: MFG: comparison learning rates for state $x = 0.80$

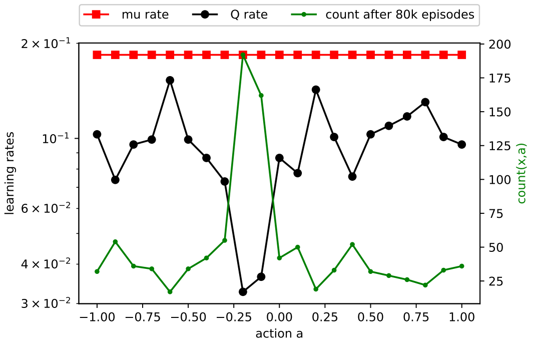

Figure 4.8: MFC: comparison learning rates for state $x = -1.50$

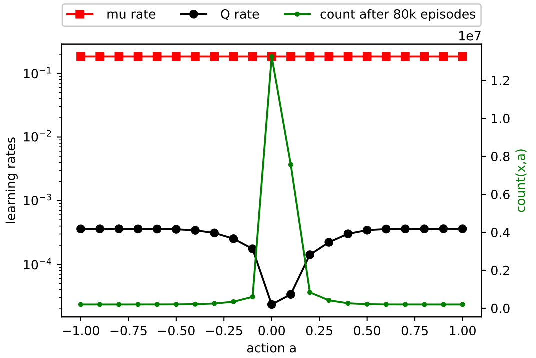

Figure 4.9: MFC: comparison learning rates for state $x = 0.10$

4.3.2 Learning of the controls and the ergodic distribution

Figures 4.10, 4.11, 4.12, 4.13, 4.14, 4.15: controls, distributions and value functions learned by the algorithm. The controls and the distribution learned by the algorithm are compared with the theoretical solution obtained in Section 3.5. As presented in Sections 4.1.4, 4.1.5, the control $\alpha(x)$ is obtained as the $\arg\min_a Q(x,a)$. Similarly, the value function $V(x)$ can be recovered as $\min_a Q(x,a)$. The $x-$axis represents the state variable x. In Figures 4.10, 4.11, 4.12, 4.13, 4.14, 4.15, the left $y-$axis relates to the action $\alpha(x)$. The right $y-$axis refers to the probability mass $\mu(x)$. The red (resp. blue) line shows the theoretical control function for the MFG (resp. MFC) problem. The black dots are the controls learned by the algorithm. Note that the peak of the distribution μ is not located at the same point x for MFG and MFC. Note that the peak of the distribution μ is not located at the same point x for MFG and MFC. In Figures 4.10, 4.12, the $y-$axis corresponds to the value function $V(x)$. The continuous lines refer to the theoretical solution. The black dots are the numerical approximation recovered by the Q-function. We observe that the algorithm converges to different solutions based on the choice of the pair (ω^Q, ω^μ). On the left, the choice $(\omega^Q, \omega^\mu) = (0.55, 0.85)$ produces the approximation of the solution of the MFG. On the right, the set of parameters $(\omega^Q, \omega^\mu) = (0.65, 0.15)$ lets the algorithm learn the solution of the MFC problem. In Figures 4.10 , 4.11 the learned controls and the learned ergodic distribution is averaged over 10 runs. In Figures 4.12 , 4.13 the learned controls and the learned distribution μ_T is averaged over 10 runs and the last 10^4 episodes.

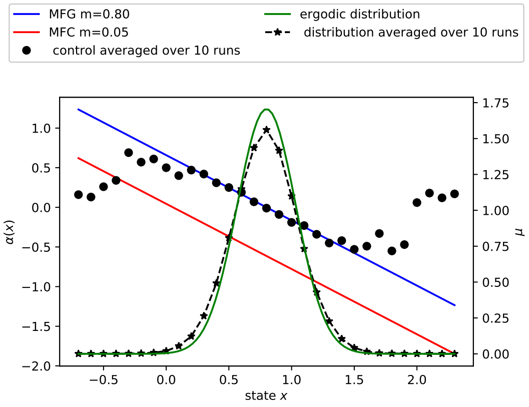

Figure 4.10: MFG: results averaged over 10 runs

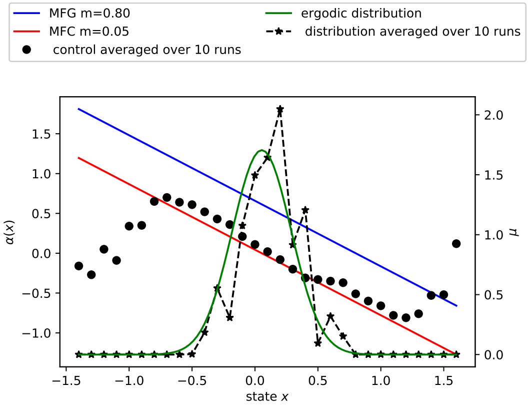

Figure 4.11: MFC: results averaged over 10 runs

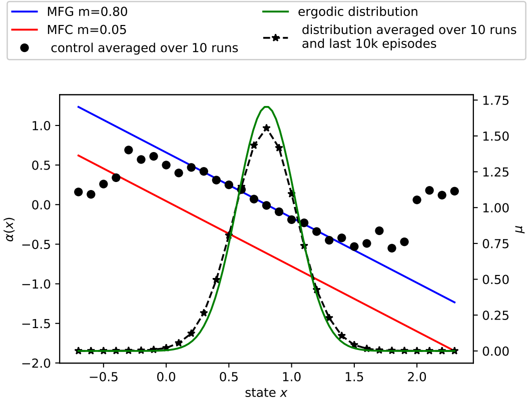

Figure 4.12: MFG: results averaged over 10 runs and last $10k$ episodes

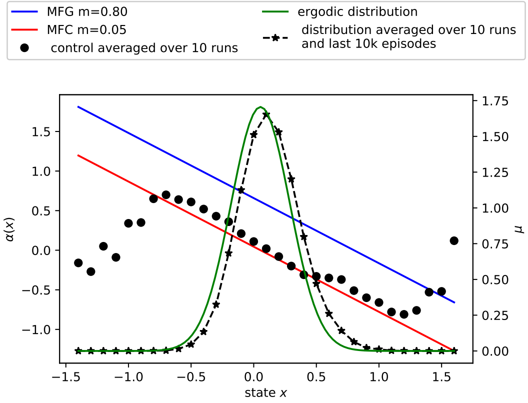

Figure 4.13: MFC: results averaged over 10 runs and last $10k$ episodes

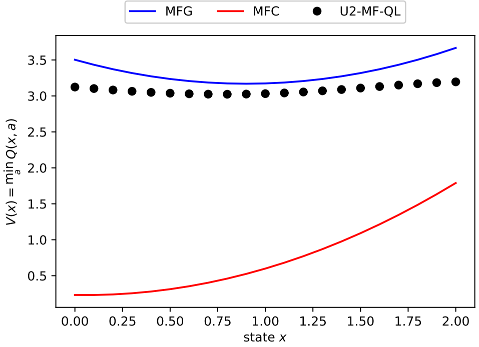

Figure 4.14: MFG: value function

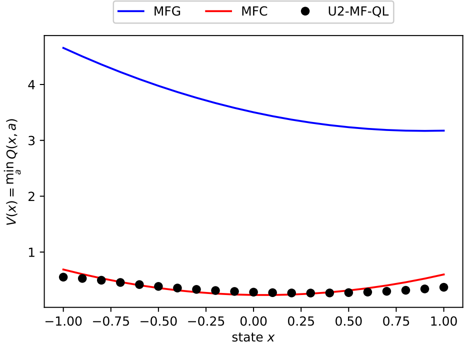

Figure 4.15: MFC: value function

4.3.3 Empirical error analyses

Figures 4.16, 4.17: MSE error on the control. A metric used to evaluate the numerical results consists in the mean squared error (MSE) of the controls learned by episode k with respect to the theoretical solution presented in Section 3.5. In particular, this metric considers the states $x \in \mathcal{X}$ where the ergodic distribution $\hat{\mu}$ is mostly concentrated. Let $\mathcal{C}_{MFG} \subset \mathcal{X}$ be centered in $\hat{m}$ s.t. $\hat{\mu}(\mathcal{C}_{MFG}) = 0.99$, then the mean squared error by episode k for run i and its average over all runs are defined as

$$\mathrm{MSE}_\alpha(i,k) = \frac{1}{|\mathcal{C}_{MFG}|}\sum_{j=0}^{|\mathcal{C}_{MFG}|-1}(\alpha^{i,k}(x_j) - \hat{\alpha}(x_j))^2, \quad \mathrm{MSE}_\alpha(k) = \frac{1}{\#runs}\sum_{i=0}^{\#runs}\mathrm{MSE}_\alpha(i,k).$$

The $x-$axis represents the number of episodes used for learning. The $y-$axis represents the mean squared error averaged over 10 runs (solid line) and its standard deviation (shaded region).

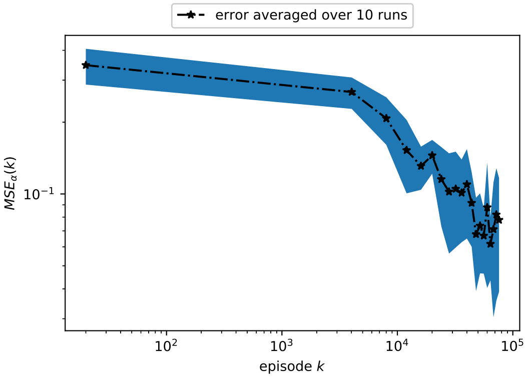

Figure 4.16: MFG: squared root of $\mathrm{MSE}_\alpha(k)$

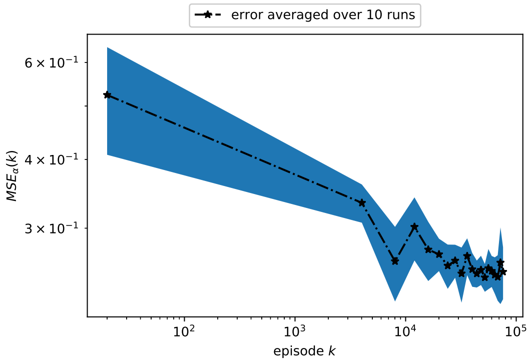

Figure 4.17: MFC: squared root of $\mathrm{MSE}_\alpha(k)$

Figures 4.18, 4.19: MSE on the ergodic mean. A metric used to evaluate the numerical results consists in the squared error of the ergodic mean learned by episode k compared with its theoretical value obtained in Section 3.5 averaged over the total numbers of runs, i.e.

$$\mathrm{MSE}_m(k) = \frac{1}{\#runs}\sum_{i=0}^{\#runs}(m_T^{i,k} - \hat{m})^2.$$

The $x-$axis represents the number of episodes used for learning. The $y-$axis represents the error averaged over 10 runs (solid line) and its standard deviation (shaded region). For the MFG, the error in the approximation of the ergodic mean reduces both in mean and standard deviation by increasing the number of episodes. For the MFC case, an oscillating behavior is observed. The choice of $\omega_\mu = 0.15$ in the learning rates defined in 4.20 allows to quicker adjustment of the mean by allocating more weights on the most recent sample. In this way, the algorithm mimics the nature of the MFC problem at the expense of a slower and more oscillating convergence.

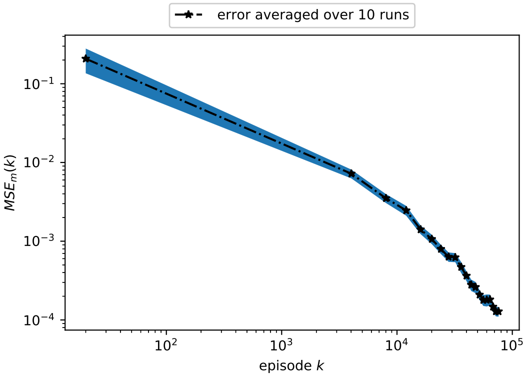

Figure 4.18: MFG: mean sqared error on $\hat{m}$

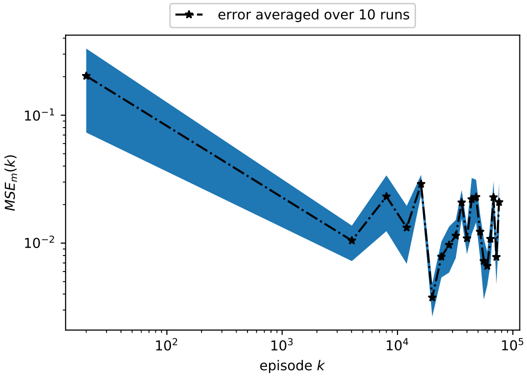

Figure 4.19: MFC: mean sqared error on $\hat{m}$

4.3.4 Empirical analyses of the stopping criteria

Figures 4.20, 4.22, 4.21, 4.23: stopping criteria. The goal of the the U2-MF-QL is to obtain a good approximation of the optimal controls and the ergodic distribution. As presented in algorithm 2, the stopping criteria is based on the analyses of the progresses in learning the optimal Q function and the ergodic distibution. The total variation and the $1,1$-norm between the start and the end of each episode is evaluated for the distribution and the Q-table respectively as follows

$$\delta(\mu_T^{k-1}, \mu_T^k) = \sum_{x_i \in \mathcal{X}} \left|\mu_T^k(x_i) - \mu_T^{k-1}(x_i)\right|, \qquad \|Q^k - Q^{k-1}\|_{1,1} = \sum_{i,j} \left|Q_{i,j}^k - Q_{i,j}^{k-1}\right|.$$

The algorithm stops when the increments are not significant anymore based on a threshold given as input. The value of the threshold depends on the user's needs and it may be calibrated by a trial and error approach. The remaining plots show how these quantities decrease as the number of episodes increase. The $x-$axis represents the number of episodes used for learning. The $y-$axis represents the value of the total variation.

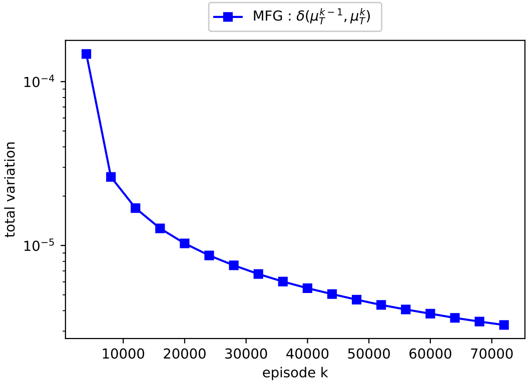

Figure 4.20: MFG: total variation on μ

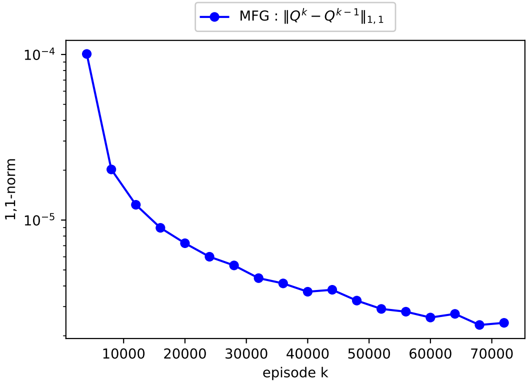

Figure 4.21: MFG: total variation on Q

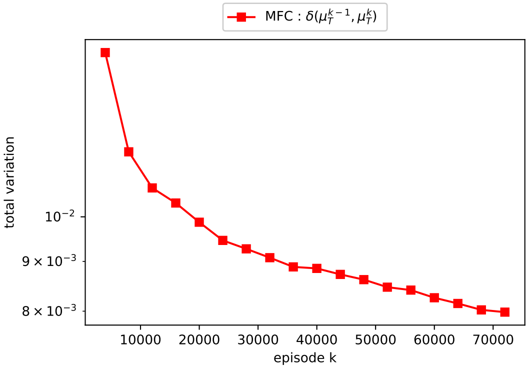

Figure 4.22: MFC: total variation on μ

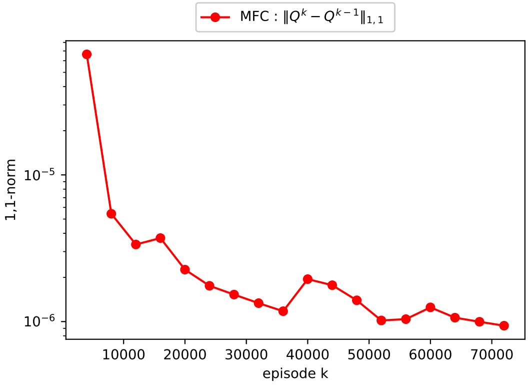

Figure 4.23: MFC: total variation on Q

Chapter 5

Mean Field Reinforcement Learning for Finite Horizon Problems, with Applications to Economics

Mean field games with interactions through the controls, sometimes called “extended”, occur when the dynamics or the cost function of a typical player explicitly depends on the empirical measure of the *controls* of the other players, and not just on their respective states. Such games were first introduced by Gomes et al. [39, 41] and their investigation quickly garnered interest.

Interaction through the controls' distribution is particularly relevant in economics and finance, see *e.g.* [50, 40, 26, 31, 43, 21] and [23] for a recent survey. Some aspects of the PDE approach and the probabilistic approach to such games have been treated respectively in [13, 14, 54] and in [26]. As in many fields, linear-quadratic models are particularly appealing due to their tractability, see *e.g.* [4, 42] for applications to energy production.

In Chapter 4, we proposed a unified two timescale Q-learning algorithm to solve both MFG and MFC problems in an infinite horizon stationary regime. The key idea is to iteratively update estimates of the distribution and the Q-function with different learning rates. Suitably choosing these learning rates enables the algorithm to learn the solution of the MFG or the one of the MFC. A slow updating of the distribution of the state leads to the Nash equilibrium of the competitive MFG and the algorithm learns the corresponding optimal strategy. A rapid updating of the distribution leads to learning of the optimal control of the corresponding cooperative MFC. Moreover, in contrast with other approaches, our algorithm does not require the environment to output the population distribution which means that a single agent can learn the solution of mean field problems.

In this Chapter, we present the new approach introduced in our paper [7] to extend this algorithm in two directions: finite horizon setting, and "extended" mean field problems which involve the distribution of controls as well. That demonstrates the flexibility of our two timescale algorithm and broadens the range of applications.

The rest of the Chapter is organized as follows. In Section 5.1, we introduce the framework of finite horizon mean field games and mean field control problems. In Section 5.2, we present the main ideas behind the two timescale approach in this context. Based on this perspective, we introduce in Section 5.3 a reinforcement learning algorithm to solve MFC and MFG problems. We then illustrate this method on two examples: a mean field accumulation problem in Section 5.4 and an optimal execution problem for a mean field of traders in Section 5.5.

Notation. For a random variable X, $\mathcal{L}(X)$ denotes its law. d and k are two positive integers corresponding respectively to the state and the action dimensions. Unless otherwise specified, ν will be used to denote a state-action distribution, and its first and second marginals will respectively be denoted by μ and θ.

5.1 Finite horizon mean field problems

In this section, we introduce the framework of mean field games and mean field control problems on a finite horizon in a discrete time and discrete space framework. For the link with finite player games, see *e.g.* [24].

5.1.1 Mean field games

Let $\mathcal{X}$ and $\mathcal{A}$ be finite sets corresponding to spaces of states and actions respectively. We denote by $\Delta^{|\mathcal{X}|}$ the simplex in dimension $|\mathcal{X}|$, which we identify with the space of probability measures on $\mathcal{X}$. $\Delta^{|\mathcal{X}\times\mathcal{A}|}$ is defined similarly on the product space $\mathcal{X}\times\mathcal{A}$. The state follows a random evolution in which X_{n+1} is determined as a function of the current state X_n, the action α_n, the state-action population distribution ν_n at time n, and some noise. We introduce the transition probability function:

$$p(x'|x,a,\nu), \qquad (x,x',a,\nu)\in\mathcal{X}\times\mathcal{X}\times\mathcal{A}\times\Delta^{|\mathcal{X}\times\mathcal{A}|},$$

which gives the probability to jump to state x' when being at state x and using action a and when the state-action population distribution is ν. For simplicity, we consider the homogeneous case where this function does not depend on time, which corresponds, in the continuous formulation, to the case where both b and σ are time-independent. Restoring this time-dependence if needed is a straightforward procedure.

Let $f:\mathcal{X}\times\mathcal{A}\times\Delta^{|\mathcal{X}\times\mathcal{A}|}\to\mathbb{R}$ be a running cost function. We interpret $f(x,a,\nu)$ as the one-step cost, at any given time step, incurred to a representative agent who is at state x and uses action a while the state-action population distribution is ν. Let $g:\mathcal{X}\times\Delta^{|\mathcal{X}|}\to\mathbb{R}$ be the terminal cost function depending on the pair (x,μ), where μ is the first marginal distribution of ν corresponding to the state population distribution. For the sake of simplicity and without loss of generality, we consider the case where the function f and g are time-independent.

An extended mean field game equilibrium is defined as a pair $(\hat{\alpha}, \hat{\nu})$ where $\hat{\alpha} : \{0, \dots, T\} \times \mathcal{X} \to \mathcal{A}$ and $\hat{\boldsymbol{\nu}} = (\hat{\nu}_n)_{n \in \{0,\dots,T\}} \in (\Delta^{|\mathcal{X}\times\mathcal{A}|})^{T+1}$ is a flow of probability distributions on $\mathcal{X} \times \mathcal{A}$, such that the following two conditions hold:

1. $\hat{\alpha}$ is the minimizer of

$$\alpha \to J^{MFG}(\alpha, \hat{\nu}) = \mathbb{E}\left[\sum_{n=0}^{T-1} f(X_n^{\alpha,\hat{\nu}}, \alpha_n(X_n^{\alpha,\hat{\nu}}), \hat{\nu}_n) + g(X_T^{\alpha,\hat{\nu}}, \hat{\mu}_T)\right],$$

   where $\alpha_n(\cdot) := \alpha(n, \cdot)$ and $\hat{\mu}_T$ is the first marginal of $\hat{\nu}_T$ corresponding to the terminal state distribution. The process $X^{\alpha,\hat{\nu}}$ has a given initial distribution $\mu_0 \in \Delta^{|\mathcal{X}|}$ and follows the dynamics

$$\mathbb{P}(X_{n+1}^{\alpha,\hat{\nu}} = x' | X_n^{\alpha,\hat{\nu}} = x, \alpha_n = a, \nu_n = \hat{\nu}_n) = p(x'|x, a, \hat{\nu}_n).$$

2. $\hat{\nu}_n = \mathcal{L}(X_n^{\hat{\alpha},\hat{\nu}}, \hat{\alpha}_n)$ for all $n \in \{0, \dots, T-1\}$.

5.1.2 Mean field control

In contrast with the MFG problem – which corresponds to a Nash equilibrium, the mean field control (MFC) problem is an optimization problem. It can be interpreted as the problem posed to a social planner trying to find the optimal behavior of a population so as to minimize a social cost (*i.e.*, a cost averaged over the whole population). It is an optimal control problem for a McKean-Vlasov dynamics: Find α^* which minimizes

$$\alpha \to J^{MFC}(\alpha) = \mathbb{E}\left[\sum_{n=0}^{T-1} f(X_n^{\alpha}, \alpha_n(X_n^{\alpha}), \nu_n^{\alpha})dt + g(X_T^{\alpha}, \mu_T^{\alpha})\right],$$

where ν_n^{α} is a shorthand notation for $\mathcal{L}(X_n^{\alpha}, \alpha_n(X_n^{\alpha}))$ and μ_T^{α} is its first marginal at terminal time T. The process X^{α} has initial distribution μ_0 and dynamics

$$\mathbb{P}(X_{n+1}^{\alpha} = x' | X_n^{\alpha} = x, \alpha_n = a, \nu_n = \nu_n^{\alpha}) = p(x'|x, a, \nu_n^{\alpha}).$$

The dynamics of X involves the law of this process, hence the terminology McKean-Vlasov dynamics [60]. To alleviate notation we will sometimes write $\nu^* = \nu^{\alpha^*}$ for the law of the optimally controlled process.

Remark 4 *Although the two problems look similar, they in general have different solutions, i.e.,* $\hat{\alpha} \neq \alpha^*$ *and* $\hat{\nu} \neq \nu^*$*, even when the functions in the cost and the dynamics are the same, see e.g. [25].*

Remark 5 *Although the mean field paradigm is the same, the special case where the interactions are only through the state distribution (i.e., the first marginal of* ν*) has attracted more interest in the literature than the present general setup. However interactions through the distribution of controls appears in many applications, particularly in economics and finance as already mentioned in the introduction. See next sections for some examples.*

Remark 6 *Although the reinforcement learning literature typically focuses on infinite horizon discounted problems, we focus here on finite horizon problems. This will cause some numerical difficulties but is crucial for many applications.*

5.2 Two timescale approach

5.2.1 State-value function

In the MFG framework, given a state-action population distribution sequence $\boldsymbol{\nu} = (\nu_n)_{n\in\{0,\dots,T\}}$ and a deterministic policy $\boldsymbol{\alpha} = (\alpha_n)_{n\in\{0,\dots,T\}}$, the state-value function of an infinitesimal player at a given time step n is

$$V^{\alpha}_{n,\nu}(x) = \mathbb{E}\left[\sum_{n'=n}^{T-1} f(X^{\alpha,\nu}_{n'}, \alpha_{n'}(X^{\alpha,\nu}_{n'}), \nu_{n'}) + g(X^{\alpha,\nu}_T, \mu_T)\middle| X^{\alpha,\nu}_n = x\right].$$

Note that the J^{MFG} and the V functions are related by:

$$J^{MFG}_{\nu}(\alpha) = \mathbb{E}_{X_0\sim\mu_0}[V^{\alpha}_{0,\nu}(X_0)].$$

In the MFC case, since the dynamics is of MKV type, the value function is the value function of the social planner and it takes the distribution ν as input, see *e.g.* [57, 68, 30, 64, 44, 33]. However, when the population is already evolving according to the sequence of distributions ν^α generated by a control α, the cost-to-go of an infinitesimal agent starting at position x at time n and using control α too is simply a function of its position and is given by

$$V_n^\alpha(x) = \mathbb{E}\left[\sum_{n'=n}^{T-1} f(X_{n'}^\alpha, \alpha_{n'}(X_{n'}^\alpha), \nu_{n'}^\alpha) + g(X_T^\alpha, \mu_T^\alpha)\Big| X_n^\alpha = x\right].$$

5.2.2 Action-value function

As explained in the previous chapters, the state value function is useful as far as the value of the game or control problem is concerned. However, it does not provide any information about the equilibrium or optimal control $\hat{\alpha}$ or α^*. For this reason, one can introduce the state-action value function, also called Q-function, which takes as inputs not only a state x but also an action a.

Before moving on to the mean-field setup, let us recall that the definition of the optimal Q-function for a classical MDP in the finite horizon framework is given by:

$$\begin{cases} Q_T^*(x,a) = g(x), \qquad (x,a) \in \mathcal{X} \times \mathcal{A}, \\ Q_n^*(x,a) = \min_\alpha \mathbb{E}\left[\sum_{n'=n}^{T-1} f(X_{n'}, \alpha_{n'}(X_{n'})) + g(X_T)\,\Big|\, X_n = x, A_n = a\right], \\ \qquad\qquad n < T, \quad (x,a) \in \mathcal{X} \times \mathcal{A}. \end{cases}$$

Using dynamic programming, it can be shown that $(Q_n^*)_n$ is the solution of the Bellman equation:

$$\begin{cases} Q_T^*(x,a) = g(x), \qquad (x,a) \in \mathcal{X} \times \mathcal{A}, \\ Q_n^*(x,a) = f(x,a) + \sum_{x' \in \mathcal{X}} p(x'|x,a) \min_{a'} Q_{n+1}^*(x',a'), \qquad n < T, \quad (x,a) \in \mathcal{X} \times \mathcal{A}. \end{cases}$$

The corresponding optimal value function $(V_n^*)_n$ is given by:

$$V_n^*(x) = \min_a Q_n^*(x,a), \qquad n \leqslant T, \quad x \in \mathcal{X}.$$

As mentioned above, one of the main advantages of computing the action-value function instead of the value function is that from the former, one can directly recover the optimal control at time n, given by $\arg\min_{a\in\mathcal{A}} Q_n^*(x,a)$. This is particularly important in order to design model-free methods, as we will see in the next section.

The above approach can be adapted to solve MFG by noticing that, when the population behavior is given, the problem posed to a single representative agent is a standard MDP. It can thus be tackled using a Q-function which implicitly depends on the population distribution: given $\boldsymbol{\nu} = (\nu_{t_n})_{n=0,\dots,T}$

$$\begin{cases} Q^*_{T,\boldsymbol{\nu}}(x,a) = g(x,\mu_T), \qquad (x,a) \in \mathcal{X}\times\mathcal{A}, \\ Q^*_{n,\boldsymbol{\nu}}(x,a) = f(x,a,\nu_n) \\ \qquad\qquad + \sum_{x'\in\mathcal{X}} p(x'|x,a,\nu_n) \min_{a'} Q^*_{n+1,\boldsymbol{\nu}}(x',a'), \qquad n < T, \quad (x,a) \in \mathcal{X}\times\mathcal{A}. \end{cases}$$

This function characterizes, at each time step n, the optimal cost-to-go for an agent starting at time n at state x, using action a for the first step, and then acting optimally for the rest of the time steps, while the population evolution is given by $\boldsymbol{\nu} = (\nu_n)_n$. However, to find the Nash equilibrium, it is not sufficient to compute the Q-function for an arbitrary sequence of distributions ν: we want to find $Q^*_{\nu^*}$ where ν^* is the population evolution generated by the optimal control computed from $Q^*_{\nu^*}$. In the sequel, we will directly aim at the Q-function $Q^*_{\nu^*}$ via a two timescale approach.

In the MFC problem the population distribution is not fixed while each player optimizes because all the agents cooperate to choose a distribution which is optimal from the point of view of the whole society. As a consequence, the optimization problem can not be recast as a standard MDP. However we will show below that it is still possible to compute the social optimum using a modified Q-function (not involving explicitly the population distribution). This major difficulty is treated in detail in the context of infinite horizon in Section 4.1.3.

5.2.3 Unification through a two timescale approach

A simple approach to compute the MFG solution is to iteratively update the state-action value function, Q, and the population distribution, ν: Starting with an initial guess $\nu^{(0)}$, repeat for $k = 0, 1, \dots,$

1. Solve the backward equation for $Q^{(k+1)} = Q^*_{\nu^{(k)}}$, which characterizes the optimal state-action value function of a typical player if the population behavior is given by $\nu^{(k)}$:
$$\begin{cases} Q_T^{(k+1)}(x,a) = g(x, \mu_T^{(k)}), \qquad (x,a) \in \mathcal{X} \times \mathcal{A}, \\ Q_n^{(k+1)}(x,a) = f(x,a,\nu_n^{(k)}) \\ \qquad + \sum_{x' \in \mathcal{X}} p(x'|x,a,\nu_n^{(k)}) \min_{a'} Q_{n+1}^{(k+1)}(x',a'), \quad n < T, \quad (x,a) \in \mathcal{X} \times \mathcal{A}. \end{cases} \tag{5.1}$$

2. Solve the forward equation for $\mu^{(k+1)}$ (resp. $\nu^{(k+1)}$), which characterizes the evolution of the population state distribution (resp. state-action distribution) if everyone uses the optimal control $\alpha_n^{(k+1)}(x) = \arg\min_a Q_n^{(k+1)}(x,a)$ coming from the above Q-function (assuming this control is uniquely defined for simplicity):
$$\begin{cases} \mu_0^{(k+1)}(x) = \mu_0(x), \qquad x \in \mathcal{X}, \\ \nu_0^{(k+1)}(x,a) = \mu_0(x)\mathbf{1}_{a=\alpha_n^{(k+1)}(x)}, \qquad (x,a) \in \mathcal{X} \times \mathcal{A}, \\ \mu_{n+1}^{(k+1)}(x) = \sum_{x' \in \mathcal{X}} \mu_n^{(k+1)}(x') p(x|x', \alpha_n^{(k+1)}(x'), \nu_n^{(k+1)}), \qquad 0 \leqslant n < T, \quad x \in \mathcal{X}, \\ \nu_{n+1}^{(k+1)}(x,a) = \mu_{n+1}^{(k+1)}(x)\mathbf{1}_{a=\alpha_{n+1}^{(k+1)}(x)}, \qquad 0 \leqslant n < T, \quad (x,a) \in \mathcal{X} \times \mathcal{A}. \end{cases} \tag{5.2}$$

Here the evolution of the joint state-action population distribution is simply the product of the state distribution and a Dirac mass:

$$\nu_n^{(k+1)} = \mu_n^{(k+1)} \otimes \delta_{\alpha_n^{(k+1)}}.$$

This is because we assumed that the optimal control is given by a deterministic function from $\mathcal{X}$ to $\mathcal{A}$. If we were using randomized control, the Dirac mass would need to be replaced by the distribution of controls.

To alleviate notation, let us introduce the operators $\widetilde{\mathcal{T}} : (\Delta^{|\mathcal{X}\times\mathcal{A}|})^{T+1} \to (\mathbb{R}^{|\mathcal{X}\times\mathcal{A}|})^{T+1}$ and $\widetilde{\mathcal{P}} : (\mathbb{R}^{|\mathcal{X}\times\mathcal{A}|})^{T+1} \to (\Delta^{|\mathcal{X}\times\mathcal{A}|})^{T+1}$ such that: (5.1) and (5.2) rewrite

$$Q^{(k+1)} = \widetilde{\mathcal{T}}(\nu^{(k)}), \qquad \nu^{(k+1)} = \widetilde{\mathcal{P}}(Q^{(k+1)}).$$

If this iteration procedure converges, we have $Q^{(k+1)} \to Q^{(\infty)}, \nu^{(k+1)} \to \nu^{(\infty)}$ as $k \to +\infty$ for some $Q^{(\infty)}, \nu^{(\infty)}$ satisfying

$$Q^{(\infty)} = \widetilde{\mathcal{T}}(\nu^{(\infty)}), \qquad \nu^{(\infty)} = \widetilde{\mathcal{P}}(Q^{(\infty)}),$$

which implies that $\nu^{(\infty)}$ is the state-action equilibrium distribution of the MFG solution, and the associated equilibrium control is given by: $\alpha_n^{(\infty)}(x) = \arg\min_a Q_n^{(\infty)}(x,a)$ for each n.

However, this procedure fails to converge in many MFG by lack of strict contraction property. To remedy this issue, a simple twist is to introduce some kind of damping. Building on this idea, we introduce the following iterative procedure, where $(\rho_Q^{(k)})_{k\geqslant 0}$ and $(\rho_\nu^{(k)})_{k\geqslant 0}$ are two sequences of learning rates:

$$Q^{(k+1)} = (1-\rho_Q^{(k)})Q^{(k)} + \rho_Q^{(k)}\widetilde{\mathcal{T}}(\nu^{(k)}), \qquad \nu^{(k+1)} = (1-\rho_\nu^{(k)})\nu^{(k)} + \rho_\nu^{(k)}\widetilde{\mathcal{P}}(Q^{(k+1)}).$$

For the sake of brevity, let us introduce the operators $\mathcal{T} : (\mathbb{R}^{|\mathcal{X}\times\mathcal{A}|})^{T+1} \times (\Delta^{|\mathcal{X}\times\mathcal{A}|})^{T+1} \to (\mathbb{R}^{|\mathcal{X}\times\mathcal{A}|})^{T+1}$ and $\mathcal{P} : (\mathbb{R}^{|\mathcal{X}\times\mathcal{A}|})^{T+1} \times (\Delta^{|\mathcal{X}\times\mathcal{A}|})^{T+1} \to (\Delta^{|\mathcal{X}\times\mathcal{A}|})^{T+1}$

$$\mathcal{T}(Q,\nu) = \widetilde{\mathcal{T}}(\nu) - Q, \qquad \mathcal{P}(Q,\nu) = \widetilde{\mathcal{P}}(Q) - \nu.$$

Then the above iterations can be written as

$$Q^{(k+1)} = Q^{(k)} + \rho_Q^{(k)} \mathcal{T}(Q^{(k)}, \nu^{(k)}), \qquad \nu^{(k+1)} = \nu^{(k)} + \rho_\nu^{(k)} \mathcal{P}(Q^{(k+1)}, \nu^{(k)}). \tag{5.3}$$

If $\rho_\nu^{(k)} < \rho_Q^{(k)}$, the Q-function is updated at a faster rate, while it is the converse if $\rho_\nu^{(k)} > \rho_Q^{(k)}$. We can thus intuitively guess that these two regimes should converge to different limits. Similar ideas have been studied by Borkar [15, 16] in the so-called two timescales approach. The key insight comes from rewriting the (discrete time) iterations in continuous time as a pair of ODEs. From [16, Chapter 6, Theorem 2], we expect to have the following two situations:

- If $\rho_\nu^{(k)} < \rho_Q^{(k)}$, the system (5.3) tracks the ODE system

  $$\begin{cases} \dot{Q}^{(t)} = \dfrac{1}{\epsilon} \mathcal{T}(Q^{(t)}, \nu^{(t)}), \\ \dot{\nu}^{(t)} = \mathcal{P}(Q^{(t)}, \nu^{(t)}), \end{cases}$$

  where $\rho_\nu^{(k)} / \rho_Q^{(k)}$ is thought of being of order $\epsilon \ll 1$. Hence, for any fixed $\tilde{\nu}$, the solution of

  $$\dot{Q}^{(t)} = \frac{1}{\epsilon} \mathcal{T}(Q^{(t)}, \tilde{\nu}),$$

  is expected to converge as $\epsilon \to 0$ to a $Q^{\tilde{\nu}}$ such that $\mathcal{T}(Q^{\tilde{\nu}}, \tilde{\nu}) = 0$. This condition can be interpreted as the fact that $Q^{\tilde{\nu}} = (Q_n^{\tilde{\nu}})_{n=0,\ldots,T}$ is the state-action value function of an infinitesimal agent facing the crowd distribution sequence $\tilde{\nu} = (\tilde{\nu}_n)_{n=0,\ldots,T}$. Then the second ODE becomes

  $$\dot{\nu}^{(t)} = \mathcal{P}(Q^{\nu^{(t)}}, \nu^{(t)}),$$

  which is expected to converge as $t \to +\infty$ to a $\nu^{(\infty)}$ satisfying

  $$\mathcal{P}(Q^{\nu^{(\infty)}}, \nu^{(\infty)}) = 0.$$

  This condition means that $\nu^{(\infty)}$ and the associated control given by $\hat{\alpha}_n(x) = \arg\min_a Q_n^{\nu^{(\infty)}}(x, a)$ form a Nash equilibrium.

- If $\rho_\nu^{(k)} > \rho_Q^{(k)}$, the system (5.3) tracks the ODE system

  $$\begin{cases} \dot{Q}^{(t)} = \mathcal{T}(Q^{(t)}, \nu^{(t)}), \\ \dot{\nu}^{(t)} = \frac{1}{\epsilon}\mathcal{P}(Q^{(t)}, \nu^{(t)}), \end{cases}$$

  where $\rho_Q^{(k)}/\rho_\nu^{(k)}$ is thought of being of order $\epsilon \ll 1$. Here, for any fixed $\tilde{Q}$, the solution of

  $$\dot{\nu}^{(t)} = \frac{1}{\epsilon}\mathcal{P}(\tilde{Q}, \nu^{(t)}),$$

  is expected to converge as $\epsilon \to 0$ to a $\nu^{\tilde{Q}}$ such that $\mathcal{P}(\tilde{Q}, \nu^{\tilde{Q}}) = 0$, meaning that $\nu^{\tilde{Q}} = (\nu_n^{\tilde{Q}})_{n=0,\dots,T}$ is the distribution evolution of a population in which every agent uses control $\tilde{\alpha}_n(x) = \arg\min_a \tilde{Q}_n(x,a)$ at time n. In fact, the definitions of $\tilde{\alpha}_n$ and $\nu^{\tilde{Q}}$ need to be *modified* to take into account the first action (x,a). The details of this crucial step for handling MFC were discussed in Section 4.1.3 .

  Then the first ODE becomes

  $$\dot{Q}^{(t)} = \frac{1}{\epsilon}\mathcal{T}(Q^{(t)}, \nu^{Q^{(t)}}),$$

  which is expected to converge as $t \to +\infty$ to a $Q^{(\infty)}$ such that

  $$\mathcal{T}(Q^{(\infty)}, \nu^{Q^{(\infty)}}) = 0.$$

  This condition (in the *modified* MFC setup) means that the control $\hat{\alpha}_n(x) = \arg\min_a Q_n^{(\infty)}(x,a)$ is a MFC optimum and the induced optimal distribution is $\nu^{Q^{(\infty)}}$.

The above iterative procedure is purely deterministic and allows us to understand the rationale behind the two timescale approach. However, in practice we rarely have access to the operators $\mathcal{T}$ and $\mathcal{P}$. Instead, we will consider that we only have access to noisy versions and we use intuition from stochastic approximation to design an algorithm.

Instead of assuming that we know the dynamics or the cost functions, we will simply assume that the learning agent can interact with an environment from which she can sample stochastic transitions as discussed in Section 4.2.1.

5.3 Reinforcement Learning Algorithm

In this Section we propose an extension of the Unified Two Timescales Mean Field Q-learning (U2-MF-QL) algorithm discussed in Section 4.2.1 to tackle MFG and MFC problems in the finite horizon framework. After introducing the algorithm, we will discuss how to adapt the learning rates schedule to the new framework and how to apply the algorithm to continuous problems.

5.3.1 U2-MF-QL-FH: an extension for the finite horizon framework

The U2-MF-QL algorithm represents a unified approach to solve asymptotic Mean Field Games and Mean Field Control problems based on the relationship between two learning rates relative to the update rules of the Q table and the distribution of the population μ respectively. Based on the intuition presented in Section 5.2, a choice of learning rates (ρ^Q, ρ^μ) such that $\rho^Q > \rho^\mu$ allows the algorithm to solve a MFG problem. The estimation of Q is updated at a faster pace with respect to the distribution which behaves as quasi-static mimicking the freezing of the flow of measures characteristic of the solving scheme discussed in Section 5.1.1. On the other hand, learning rates satisfying $\rho^Q < \rho^\mu$ allow the algorithm to update instantaneously the control function (Q table) at any change of the distribution reproducing the MFC framework. Under suitable assumptions, one may expect the asymptotic problems to be characterized by controls that are independent of time. In this case, the learning goals reduce to a control function valid for every time point and the asymptotic distribution of the states of the population. The finite horizon framework presented in Sections 5.1.1 and 5.1.2 differs from the asymptotic case discussed in Chapter 4 in several ways other than the restriction on the finite time interval $[0, T]$. First, the mean field interaction is through the joint distribution of states and actions of the population rather than the marginal distribution of the states. Further, both the control rule and the mean field distribution are generally time dependent. Due to these differences, the $2-$dimensional matrix Q in U2-MF-QL is replaced by a $3-$dimensional matrix $\boldsymbol{Q} := (Q_n(\cdot,\cdot))_{n=0,\ldots,T} = (Q(\cdot,\cdot,n))_{n=0,\ldots,T}$ in the finite horizon version of the algorithm (U2-MF-QL-FH). The extra dimension is introduced to learn a time dependent control function.

The Unified Two Timescales Mean Field Q-learning for Finite Horizon problems (U2-MF-QL-FH) is designed to solve problems with finite state and action spaces in finite and discrete time.

Algorithm 3 Unified Two Timescales Mean Field Q-learning - Finite Horizon

```
Require: $T$ : number of time steps,
    $\mathcal{X} = \{x_0, \dots, x_{|\mathcal{X}|-1}\}$, $\mathcal{A} = \{a_0, \dots, a_{|\mathcal{A}|-1}\}$ : finite state and action spaces,
    $\mu_0$ : initial distribution of the representative player,
    $\epsilon$ : factor related to the $\epsilon-$greedy policy,
    $tol_\nu, tol_Q$ : break rule tolerances.
Initialization: episode $k = 0$
    $Q_n^k(\cdot,\cdot) := Q^k(\cdot,\cdot,n) = 0$ for all $(x,a) \in \mathcal{X} \times \mathcal{A}$, for $n = 0, \dots, T$,
    $\nu_n^k = \frac{1}{|\mathcal{X}\times\mathcal{A}|} J_{|\mathcal{X}\times\mathcal{A}|}$ for $n = 0, \dots, T$ where $J_{d\times m}$ is an $d \times m$ unit matrix
repeat
    $k = k + 1$
    Observe $X_0^k \sim \mu_0$
    for $n \leftarrow 0$ to $T - 1$ do
        Choose action $A_n^k$ using the $\epsilon$-greedy policy derived from $Q_n^{k-1}(X_n^k, \cdot)$
        Update $\nu$:
        $\nu_n^k = \nu_n^{k-1} + \rho_k^\nu(\boldsymbol{\delta}(X_n^k, A_n^k) - \nu_n^{k-1})$
        where $\boldsymbol{\delta}(X_n^k, A_n^k) = \big(\mathbf{1}_{x,a}(X_n^k, A_n^k)\big)_{x\in\mathcal{X}, a\in\mathcal{A}}$
        Observe cost $f_{n+1} = f(X_n^k, A_n^k, \nu_n^k)$ and state $X_{n+1}^k$ provided by the environment
        Update $Q_n$:
        $Q_n^k(x,a) := \begin{cases} Q_n^{k-1}(x,a) + \rho_{x,a,k}^{Q_n}[\mathcal{B} - Q_n^{k-1}(x,a)] & \text{if } (X_n^k, A_n^k) = (x,a) \\ Q_n^{k-1}(x,a) & \text{o.w.} \end{cases}$
        where
        $\mathcal{B} := \begin{cases} f_{n+1} + \gamma \min_{a'\in\mathcal{A}} Q_{n+1}^{k-1}(X_{n+1}^k, a'), & \text{if } n < T \\ f_{n+1}, & \text{o.w.} \end{cases}$
    end for
until $||\nu_n^k - \nu_n^{k-1}||_1 \leqslant tol_\nu$ and $\|Q_n^k - Q_n^{k-1}\|_{1,1} < tol_Q$ for all $n = 0, \dots, T$
```

The same algorithm can be applied to MFG and MFC problems where the the interaction with the population is through the marginal distribution of the states $\mu \in \mathcal{P}(\mathcal{X})$ or the law of the controls $\theta \in \mathcal{P}(\mathcal{A})$. In these cases the estimation of the flow of marginal distributions is obtained through the vectors $(\mu_n)_{n=0,\dots,T}$ (resp. $(\theta_n)_{n=0,\dots,T}$) defined on the space $\mathcal{X}$ (resp. $\mathcal{A}$). The initialization is given by $\mu_n^0 = \left[\frac{1}{|\mathcal{X}|}, \dots, \frac{1}{|\mathcal{X}|}\right]$ $\left(\text{resp. } \theta_n^0 = \left[\frac{1}{|\mathcal{A}|}, \dots, \frac{1}{|\mathcal{A}|}\right]\right)$ for $n = 0, \dots, T$. The update rule at episode k is given by $\mu_n^k = \mu_n^{k-1} + \rho_k^\mu(\boldsymbol{\delta}(X_n) - \mu_n^{k-1})$ $\left(\text{resp. } \theta_n^k = \theta_n^{k-1} + \rho_k^\theta(\boldsymbol{\delta}(A_n) - \theta_n^{k-1}\right)$ where $\boldsymbol{\delta}(X_n) = \left[\mathbf{1}_{x_0}(X_n), \dots, \mathbf{1}_{x_{|\mathcal{X}|-1}}(X_n)\right]$ $\left(\text{resp. } \boldsymbol{\delta}(A_n) = \left[\mathbf{1}_{a_0}(A_n), \dots, \mathbf{1}_{a_{|\mathcal{A}|-1}}(A_n)\right]\right)$ for $n = 0, \dots, T$.

5.3.2 Learning rates

The algorithm 3 is based on two stochastic approximation rules for the distribution $\boldsymbol{\nu}$ and the 3$-$dim matrix $\boldsymbol{Q}$. The design of the learning is discussed widely in the literature, in a general context by [15] and [16], and with focus in reinforcement learning by [17] and [35]. Based on experimental evidences, we define the learning rates appearing in Algorithm 3 as follows:

$$\rho_{x,a,k}^{Q_n} = \frac{1}{(1 + T\#|(x, a, n, k)|)^{\omega^Q}}, \qquad \rho_k^\nu = \frac{1}{(1 + k)^{\omega^\nu}}, \tag{5.6}$$

where $\#|(x, a, n, k)|$ is counting the number of visits of the pair (x, a) at a given time step n until episode k. Differently from the asymptotic version of the algorithm presented in Section 4.3.1 for which each pair (x, a) has a unique counter for all time points, in the finite horizon formulation a distinct counter $\#|(x, a, n, k)|$ is defined for each time point n. This choice of learning rates allows to update each matrix Q_n in an asynchronous way. The exponent ω^Q can take values in $(\frac{1}{2}, 1]$. As presented in Section 5.2.3, the pair (ω^Q, ω^ν) is chosen depending on the particular problem to solve. In a competitive framework (MFG), these parameters have to be searched in the set of values for which the condition

$\rho^Q > \rho^\nu$ is satisfied at each iteration. On the other hand, a good choice for the cooperative case (MFC) should satisfy the condition $\rho^Q < \rho^\nu$.

5.3.3 Application to continuous problems

Although it is presented in a setting with finite state and action spaces, the application of the algorithm U2-MF-QL-FH can be extended to continuous problems. Such adaptation requires truncation and discretization procedures to time, state and action spaces which should be calibrated based on the specific problem.

In practice, a continuous time interval $[0, T]$ would be replaced by a uniform discretization $\tau = \{t_n\}_{n \in \{0,\dots,N_T\}}$. The environment would provide the new state and reward at these discrete times. The continuous state would be projected on a finite set $\mathcal{X} = \{x_0, \dots, x_{|\mathcal{X}|-1}\} \subset \mathbb{R}^d$. Likewise, actions will be provided to the environment in a finite set $\mathcal{A} = \{a_0, \dots, a_{|\mathcal{A}|-1}\} \subset \mathbb{R}^k$, where the projected distribution ν would be estimated. Then Algorithm 3 is ran on those spaces.

In the problems presented in Section 5.5, we will use the benchmark linear-quadratic models given in continuous time and space for which we present explicit formulas. In that case, we use an Euler discretization of the dynamics followed by a projection on $\mathcal{X}$. We do not address here the error of approximation since the purpose of this comparison with a benchmark is mainly for illustration.

5.4 A mean field accumulation problem

5.4.1 Description of the problem

A further application of mean field theory to economics is given by the mean field capital accumulation problem by Huang in [50]. In this paper, the author studies an extension of the the classical one-agent modeling of optimal stochastic growth to an infinite population of symmetric agents. We introduce the model following the author's presentation.

At discrete time $t \in \mathbb{Z}_+$, the wealth of the representative agent is represented by a process $X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}}$ characterized by the dynamics

$$X_{t+1}^{\boldsymbol{\alpha},\boldsymbol{\theta}} = G\left(\int a d\theta_t(a), W_t\right)\alpha_t \tag{5.7}$$

where $\boldsymbol{\alpha} = (\alpha_t)_{0 \leqslant t \leqslant T}$ is the controlled variable denoting the agent's investment for production, $G\left(\int a d\theta_t(a), W_t\right)$ is the production function, $\boldsymbol{\theta} = (\theta_t)_{0 \leqslant t \leqslant T}$ is the mean field term represented by the law of the investment level of the population, $\int a d\theta_t(a)$ is its mean, and $\boldsymbol{W} = (W_t)_{0 \leqslant t \leqslant T}$ is a random disturbance. At each time t, the control α_t can only take values in $[0, X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}}]$ so that $Supp(\theta_t) \subseteq [0, X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}}]$, implying that borrowing is not allowed. The wealth remaining after investment is all consumed, *i.e.* the consumption variable c_t is equal to $c_t = X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}} - \alpha_t$. The model is based on the following assumptions:

(A1) $\boldsymbol{W}$ is a random noise source with support D_W. The initial state X_{t_0} is a positive random variable independent of $\boldsymbol{W}$ with mean m_0;

(A2) The function $G : [0, \infty) \times D_W \mapsto [0, \infty)$ is continuous. If $w \in D_W$ is fixed, $G(z, w)$ is a decreasing function of z;

(A3) $\mathbb{E}G(0, W) < \infty$ and $\mathbb{E}G(z, W) > 0$ for each $z \in [0, \infty)$.

The multiplicative factor G in the dynamics of the wealth process $X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}}$ shows the direct dependence of the wealth on both the individual investment and the population aggregated investment. Further, assumption (A2) relates to the negative mean field impact explained as the loss in production efficiency when the aggregated investment increases. An example for the function G is given by $G(z, w) = \frac{\beta w}{1+\delta z^\eta}$, where β, δ, η are non negative parameters. Let $\boldsymbol{W}$ be a positive random noise with mean equal to 1. Then $D_W \subset [0, \infty)$ and (A2) - (A3) are satisfied.

The goal of the agent is to optimize the expected aggregated discounted utility of consumption given by

$$J(\boldsymbol{\alpha}, \boldsymbol{\theta}) = \mathbb{E} \sum_{t=0}^{T} \rho^t v(c_t) = \mathbb{E} \sum_{t=0}^{T} \rho^t v(X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}} - \alpha_t), \tag{5.8}$$

where $\rho \in (0, 1]$ is the discount factor. In particular, the author of [50] analyses the case of a Hyperbolic Absolute Risk Aversion (HARA) utility function defined as

$$v(c_t) = v(X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}} - \alpha_t) := \frac{1}{\gamma}(X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}} - \alpha_t)^\gamma, \tag{5.9}$$

where $\gamma \in (0, 1)$.

5.4.2 Solution of the MFG

In a competitive game setting, the resulting mean field game problem has solution given by Theorem 3 of Section 3.2 and Theorem 6 of Section 4 in [50]. Let denote the functions $\Phi(z)$, $\phi(z)$ and $\Psi(z)$ as follows

$$\Phi(z) = \rho \mathbb{E} G^\gamma(z, W), \quad \phi(z) = \Phi(z)^{\frac{1}{\gamma - 1}}, \quad \Psi(z) = \mathbb{E} G(z, W).$$

Let suppose that the mean field interaction is through $(z_t)_{t=0,\ldots,T}$ the first moment of the flow of measures $\boldsymbol{\theta} = (\theta_t)_{t=0,\ldots,T}$. The relative value function is defined as

$$V^{\boldsymbol{\theta}}(t, x) = \sup_{\boldsymbol{\alpha}} \mathbb{E} \left[\sum_{s=t}^{T} \rho^s v(X_s^{\boldsymbol{\alpha},\boldsymbol{\theta}} - \alpha_s) | X_t^{\boldsymbol{\alpha},\boldsymbol{\theta}} = x \right].$$

The value function is equal to $V^{\boldsymbol{\theta}}(t,x) = \frac{1}{\gamma} D_t^{\gamma-1} x^{\gamma}$, where D_t can be obtained using the recursive formula

$$D_t = \frac{\phi(z_t) D_{t+1}}{1+\phi(z_t) D_{t+1}}, \qquad D_T = 1.$$

The optimal control w.r.t. $\boldsymbol{\theta}$ is given by

$$\hat{\alpha}_t(x) = \frac{x}{1+\phi(z_t) D_{t+1}}, \quad t \leqslant T-1, \qquad \hat{\alpha}_T = 0.$$

The equivalent of the Nash equilibrium in the mean field limit is obtained by solving the fixed point equation

$$(\Lambda_0, \dots, \Lambda_{T-1})(z_0, \dots, z_{T-1}) = (z_0, \dots, z_{T-1}),$$

where

$$\begin{cases} \Lambda_0(z_0,\dots,z_{T-1}) := \frac{1+\phi(z_{T-1})+\dots+\phi(z_{T-1})\dots\phi(z_1)}{1+\phi(z_{T-1})+\dots+\phi(z_{T-1})\dots\phi(z_0)} m_0, \\ \\ \Lambda_k(z_0,\dots,z_{T-1}) := \\ := \frac{1+\phi(z_{T-1})+\dots+\phi(z_{T-1})\dots\phi(z_{k+1})}{1+\phi(z_{T-1})+\dots+\phi(z_{T-1})\dots\phi(z_0)} \Psi(z_{k-1})\dots\Psi(z_0) m_0, \quad \text{for } 1 \leqslant k \leqslant T-2, \\ \\ \Lambda_{T-1}(z_0,\dots,z_{T-1}) := \\ := \frac{1}{1+\phi(z_{T-1})+\dots+\phi(z_{T-1})\dots\phi(z_0)} \Psi(z_{T-2})\dots\Psi(z_0) m_0, \quad \text{for } k = T-1. \end{cases}$$

Example 5.4.1 *A simple example is proposed in Section 3.3 of [50]. Let T be equal to 2 and (z_0, z_1) be given. The solution is defined by*

$$D_0 = \frac{\phi(z_1)\phi(z_0)}{1+\phi(z_1)+\phi(z_1)\phi(z_0)}, \quad D_1 = \frac{\phi(z_1)}{1+\phi(z_1)}, \quad D_2 = 1,$$

with controls

$$\hat{\alpha}_0(x) = \frac{(1+\phi(z_1))x}{1+\phi(z_1)+\phi(z_1)\phi(z_0)}, \quad \hat{\alpha}_1(x) = \frac{x}{1+\phi(z_1)+\phi(z_1)\phi(z_0)}, \quad \hat{\alpha}_2(x) = 0.$$

5.4.3 Solution of the MFC

We now turn our attention to the cooperative setting. For this problem, we are not aware of any explicit solution for the social optimum. Instead, we employ the numerical method proposed in [27] and use the result as a benchmark. We recall how this method works in our context. The initial problem is to minimize over α:

$$J(\boldsymbol{\alpha}) = \mathbb{E}\sum_{t=0}^{T}\rho^t v(c_t) = \mathbb{E}\sum_{t=0}^{T}\rho^t v(X_t^{\boldsymbol{\alpha}} - \alpha_t),$$

subject to: $X_0^{\boldsymbol{\alpha}}$ has a fixed distribution and

$$X_{t+1}^{\boldsymbol{\alpha}} = G(\mathbb{E}[\alpha_t], W_t)\alpha_t, \quad t > 0.$$

This problem is approximated by the following one. We fix an architecture of neural network with input in $\mathbb{R}^2$ and output in $\mathbb{R}$. Such neural networks are going to play the role of the control function, in a Markovian feedback form. The inputs are the time and space variables, and the output is the value of the control. Then the goal is to minimize over parameters ω of neural networks with this architecture the following function:

$$\widetilde{J}^N(\omega) = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T}\rho^t v(c_t^i)\right] = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T}\rho^t v(X_t^{i,\varphi_\omega} - \varphi_\omega(t, X_t^{i,\varphi_\omega}))\right],$$

subject to: $X_0^{i,\varphi_\omega}, i = 1, \ldots, N$ are i.i.d. with fixed distribution and

$$X_t^{i,\varphi_\omega} = G\left(\frac{1}{N}\sum_{j=1}^{N}\varphi_\omega(t, X_t^{j,\varphi_\omega}), W_t^i\right)\varphi_\omega(t, X_t^{i,\varphi_\omega}), \quad t > 0, i = 1, \ldots, N.$$

Notice that the parameters ω are used to compute the X_t^{i,φ_ω} for every i and every t. The mean of the control $\mathbb{E}[\alpha_t]$ is replaced by an empirical average over N samples. For this problem, an approximate minimizer is computed by running stochastic gradient descent (SGD for short) or one of its variants. At iteration k, we have a candidate ω_k for the parameters of the neural network. We randomly pick initial positions $\underline{X}_0 := (X_0^{i,\varphi_{\omega_k}})_{i=1,\ldots,N}$ and noises $\underline{\boldsymbol{W}} := (W_t^i)_{t=1,\ldots,T,i=1,\ldots,N}$. Based on this, we simulate trajectories $(X_t^{i,\varphi_{\omega_k}})_{t,i}$

and compute the associated cost, namely the term inside the expectation in the definition of $\widetilde{J}^N(\omega)$:

$$L(\omega_k; \underline{X}_0, \underline{\boldsymbol{W}}) := \frac{1}{N}\sum_{i=1}^{N}\sum_{t=0}^{T}\rho^t v(X_t^{i,\varphi_{\omega_k}} - \varphi_{\omega_k}(t, X_t^{i,\varphi_{\omega_k}})).$$

Using backpropagation, the gradient $\nabla_\omega L(\omega_k; \underline{X}_0, \underline{\boldsymbol{W}})$ of this cost with respect to ω is computed, and it is used to update the parameters. We thus obtain ω_{k+1} defined by:

$$\omega_{k+1} = \omega_k - \eta_k \nabla_\omega L(\omega_k; \underline{X}_0, \underline{\boldsymbol{W}}),$$

where $\eta_k > 0$ is the learning rate used at iteration k. In our implementation for the numerical results presented below, instead of the plain SGD algorithm we used Adam optimizer [53].

5.4.4 Numerical results

In this section, numerical results of the application of the U2-MF-QL-FH algorithm to the mean field capital accumulation problem are presented. The interaction with the population is through the law of the controls. The algorithm 3 was adapted to this case as discussed in Section 5.3.1.

The problem analyzed is a specific case of the Example 5.4.1. For more details we refer to [50, Sections 6.3 and 7, Example 18].

The production function is defined as follows

$$G(z, W) = g(z)W, \qquad g(z) = \frac{1}{\rho\mathbb{E}[W^\gamma]}\frac{C}{1 + (C-1)z^3}, \tag{5.10}$$

where W has support $D_W = \{0.9, 1.3\}$ with corresponding probabilities $[0.75, 0.25]$, C is equal to 3, the discount factor ρ is equal to 0.95 and the parameter γ of the utility function defined in equation (5.9) is equal to 0.2. The distribution of $X_0^{\alpha,\theta}$ is uniform in $[0, 1]$.

This problem is characterized by discrete time and continuous state and action spaces. In order to apply the U2-MF-QL-FH algorithm, these spaces are truncated and discretized as discussed in Section 5.3.3. They have been chosen large enough to make sure that the state is within the boundary most of the time. In practice, this would have to be calibrated in a model-free way through experiments. In this example, for the numerical experiments, we used the knowledge of the model.

The action space is given by $\mathcal{A} = \{a_0 = 0, \dots, a_{|\mathcal{A}|-1} = 4\}$ and the state space by $\mathcal{X} = \{x_0 = 0, \dots, x_{|\mathcal{X}|-1} = 4\}$. The step size for the discretization of the state and action spaces is given by 0.05.

The algorithm 3 is adapted to this particular example. Since borrowing is not allowed, the set of admissible action at state x is given by $\mathcal{A}(x) = \{a \in \mathcal{A} \text{ if } a \leqslant x\} \subseteq \mathcal{A}$. The exploitation-exploration trade off is tackled on each episode using an $\epsilon-$greedy policy. Supposed that the agent is in state x, the algorithm chooses a random action in $\mathcal{A}(x)$ with probability ϵ and the action in $\mathcal{A}(x)$ which results optimal based on the current estimation with probability $1 - \epsilon$. In our example, the value of epsilon is fixed to 0.15.

The following numerical results show how the U2-MF-QL-FH algorithm is able to learn an approximation of the control function and the mean field term in the MFG and MFC cases depending on the choice of the parameters $(\omega^Q, \omega^\theta)$.

Learning of the controls

Figures 5.1, 5.2, 5.3, 5.4,5.5, 5.6: controls learned by the algorithm. The controls learned by the U2-MF-QL-FH algorithm are compared with the benchmark solutions. Each plot corresponds to a different time point $t \in \{0, 1, 2\}$. The $x-$axis represents the state variable x. The $y-$axis relates to the action $\alpha_t(x)$. The blue (resp. green) markers show the benchmark control function for the MFG (resp. MFC) problem.

The red markers are the controls learned by the algorithm. The plots show how the algorithm converges to different solutions based on the choice of the pair $(\omega^Q, \omega^\theta)$. On the left, the choice $(\omega^Q, \omega^\theta) = (0.55, 0.85)$ produces the approximation of the solution of the MFG. On the right, the set of parameters $(\omega^Q, \omega^\theta) = (0.7, 0.05)$ lets the algorithm learn the solution of the MFC problem. The results presented in the Figures are averaged over 10 runs.

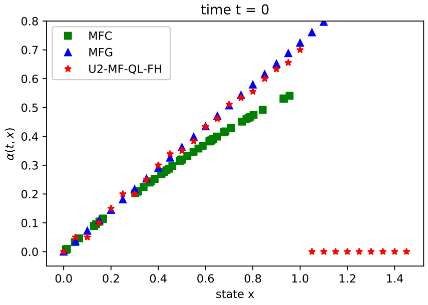

Figure 5.1: Learned Controls for MFG at time 0.

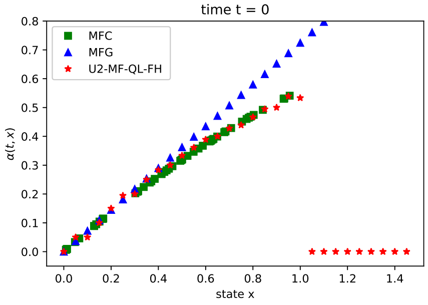

Figure 5.2: Learned Controls for MFC at time 0.

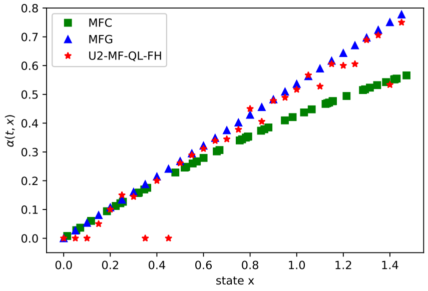

Figure 5.3: Learned Controls for MFG at time 1.

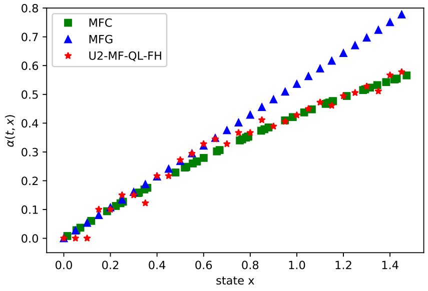

Figure 5.4: Learned Controls for MFC at time 1.

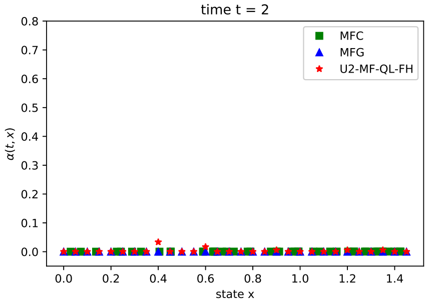

Figure 5.5: Learned Controls for MFG at time 2.

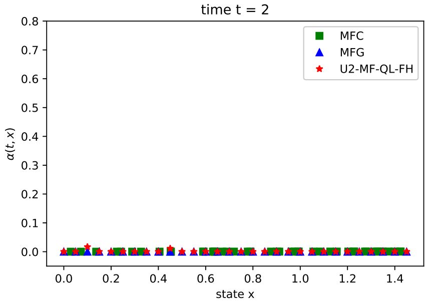

Figure 5.6: Learned Controls for MFC at time 2.

Learning of the mean field

Figures 5.7, 5.8, 5.9, 5.10, 5.11, 5.12: $\mathbb{E}\left[\alpha_t\right]$ learned by the algorithm. The estimation of the first moment of the distribution of the controls evolves with respect the number of learning episodes. The estimated quantity is compared with the benchmarks presented in Sections 5.4.2 and 5.4.3. Each plot corresponds to a different time point $t \in \{0, 1, 2\}$. The $x-$axis represents the learning episode k. The $y-$axis relates to the estimate of the first moment of the mean field $\mathbb{E}\left[\alpha_t^k\right]$ obtained by episode k. The blue (resp. green) line shows the benchmark solution for the MFG (resp. MFC) problem. The red dots are the estimates learned by the algorithm. On the left, the algorithm reaches the solution of the MFG based on the parameters $(\omega^Q, \omega^\theta) = (0.55, 0.85)$. On the right, the values $(\omega^Q, \omega^\theta) = (0.7, 0.05)$ allows the algorithm to converge to the solution of the MFC problem. The results presented in the Figures are averaged over 10 runs.

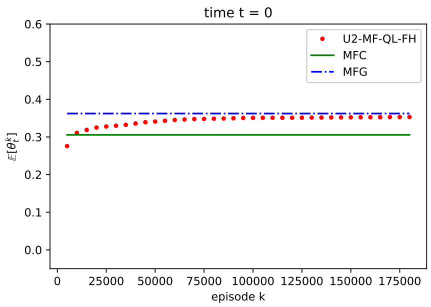

Figure 5.7: Learned control's mean for MFG at time 0.

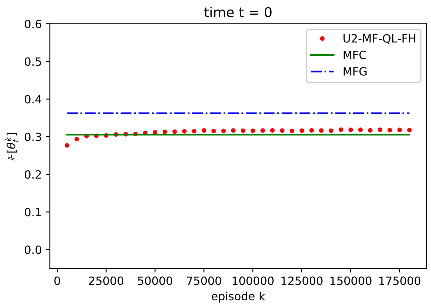

Figure 5.8: Learned control's mean for MFC at time 0.

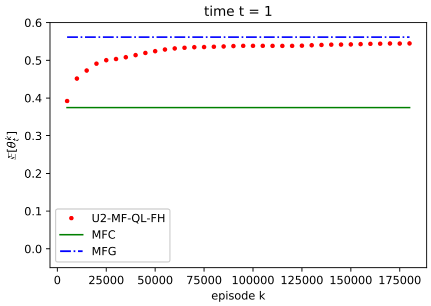

Figure 5.9: Learned control's mean for MFG at time 1.

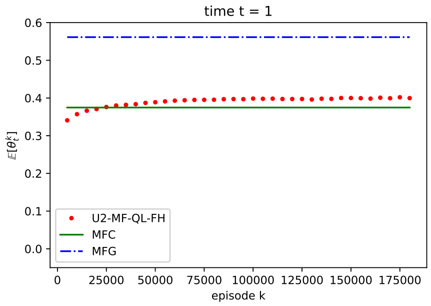

Figure 5.10: Learned control's mean for MFC at time 1.

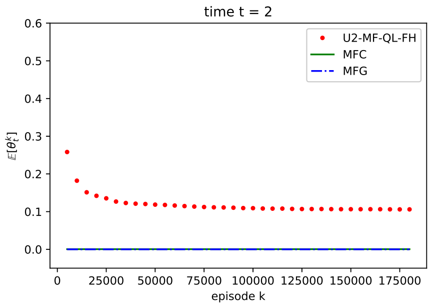

Figure 5.11: Learned control's mean for MFG at time 2.

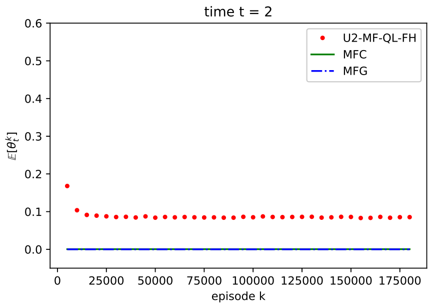

Figure 5.12: Learned control's mean for MFC at time 2.

5.5 A mean field execution problem

We now consider the *Price Impact Model* as an example of application to finance originally studied by Carmona and Lacker in [26], and presented in the book of Carmona and Delarue [24, Vol I, Sections 1.3.2 and 4.7.1]. This model addresses the question of optimal execution in the context of high frequency trading when a large group of traders want to buy or sell shares before a given time horizon T (*e.g.*, one day). The price of the stock is influenced by the actions of the traders: if they buy, the price goes up, whereas if they sell, the price goes down. This effect is stronger if a significant proportion of traders buy or sell at the same time. Incorporating such a price impact naturally leads to a problem with mean field interactions through the traders' actions.

Approaching this problem as a mean field game, the inventory of the representative trader is modeled by a stochastic process $(X_t)_{0\leqslant t\leqslant T}$ such that

$$dX_t = \alpha_t dt + \sigma dW_t, \quad t \in [0,T],$$

where α_t corresponds to the trading rate and W is a standard Brownian motion. The noise term σdW_t models a random stream of demand that a broker may receive from her clients. The price of the asset $(S_t)_{0\leqslant t\leqslant T}$ is influenced by the trading strategies of all the traders through the mean of the law of the controls $(\theta_t = \mathcal{L}(\alpha_t))_{0\leqslant t\leqslant T}$ as follows:

$$dS_t = \gamma\left(\int_{\mathbb{R}} a d\theta_t(a)\right)dt + \sigma_0 dW_t^0, \quad t \in [0,T],$$

where γ and σ_0 are constants and the Brownian motion W^0 is independent from W. The amount of cash held by the trader at time t is denoted by the process $(K_t)_{0\leqslant t\leqslant T}$. The dynamic of K is modeled by

$$dK_t = -[\alpha_t S_t + c_\alpha(\alpha_t)]dt,$$

where the function $\alpha \mapsto c_\alpha(\alpha)$ is a non-negative convex function satisfying $c_\alpha(0) = 0$, representing the cost for trading at rate α. The wealth V_t of the trader at time t is defined as the sum of the cash held by the trader and the value of the inventory with respect to the price S_t:

$$V_t = K_t + X_t S_t.$$

Applying the self-financing condition of Black-Scholes' theory, the changes over time of the wealth V are given by the equation:

$$\begin{aligned} dV_t &= dK_t + X_t dS_t + S_t dX_t \\ &= \Big[- c_\alpha(\alpha_t) + \gamma X_t \int_{\mathbb{R}} a d\theta_t(a) \Big] dt + \sigma S_t dW_t + \sigma_0 X_t dW_t^0. \end{aligned} \tag{5.11}$$

We assume that the trader is subject to a running liquidation constraint modeled by a function c_X of the shares they hold, and to a terminal liquidation constraint at maturity T represented by a scalar function g. Thus, the cost function is defined by:

$$J(\alpha) = \mathbb{E}\Big[\int_0^T c_X(X_t) dt + g(X_T) - V_T \Big],$$

where the terminal wealth V_T is taken into account with a negative sign as the cost function is to be minimized. From equation (5.11), it follows that

$$J(\alpha) = \mathbb{E}\Big[\int_0^T f(t, X_t, \theta_t, \alpha_t) dt + g(X_T) \Big],$$

where the running cost is defined by

$$f(t, x, \theta, \alpha) = c_\alpha(\alpha) + c_X(x) - \gamma x \int_{\mathbb{R}} a d\theta(a),$$

for $0 \leqslant t \leqslant T$, $x \in \mathbb{R}^d$, $\theta \in \mathcal{P}(\mathbb{A})$ and $\alpha \in \mathbb{A} = \mathbb{R}$. We assume that the functions c_X and g are quadratic and that the function c_α is strongly convex in the sense that its second derivative is bounded away from 0. See [26] for other technical assumptions. Such a particular case is known as the Almgren-Chriss linear price impact model. Thus, the control is chosen to minimize:

$$J(\alpha) = \mathbb{E}\left[\int_0^T \left(\frac{c_\alpha}{2} {\alpha_t}^2 + \frac{c_X}{2} X_t^2 - \gamma X_t \int_{\mathbb{R}} a d\theta_t(a) \right) dt + \frac{c_g}{2} X_T^2 \right],$$

over $\alpha \in \mathbb{A}$. To summarize, the running cost consists of three components. The first term represents the cost for trading at rate α. The second term takes into consideration the running liquidation constraint in order to penalize unwanted inventories. The third term defines the actual price impact. Finally, the terminal cost represents the terminal liquidation constraint.

5.5.1 The MFG trader problem

Referring to Section 5.1.1, the MFG problem is solved by first solving a standard stochastic control problem where the flow of distribution of control is given and then, solving a fixed point problem ensuring that this flow of distribution is identical to the flow of distributions of the optimal control. We adopt here the FBSDE approach where the backward variable represents the derivative of the value function. In other words, the optimal control is obtained by minimizing the Hamiltonian

$$H(x, \alpha, \theta, y) = \left(\frac{c_\alpha}{2}\alpha^2 + \frac{c_X}{2}x^2 - \gamma x \int_{\mathbb{R}} a d\theta(a)\right) + \alpha y, \tag{5.12}$$

to obtain

$$\hat{\alpha}_t = -\frac{1}{c_\alpha} Y_t, \tag{5.13}$$

where (X, Y) solves the FBSDE system obtained via the Pontryagin approach:

$$\begin{cases} dX_t = -\dfrac{1}{c_\alpha} Y_t dt + \sigma dW_t, \qquad X_0 \sim \mu_0 \\ dY_t = -\left(c_X X_t + \dfrac{\gamma}{c_\alpha}\mathbb{E}[Y_t]\right) dt + Z_t dW_t, \quad Y_T = c_g X_T. \end{cases} \tag{5.14}$$

Solution of the MFG problem

The solution of the mean field game case is discussed in details in [24, Vol I, Sections 1.3.2 and 4.7.1]. In a nutshell, one takes expectation in (5.14) to obtain a system of forward-backward ODEs for the mean of X_t denoted by $\bar{x}_t$ and the mean of Y_t denoted by $\bar{y}_t$. This system is solved using the ansatz $\bar{y}_t = \bar{\eta}_t \bar{x}_t + \bar{\chi}_t$. The coefficient function $\bar{\eta}_t$ satisfies a Riccati equation which admits the solution:

$$\bar{\eta}_t = \frac{-C(e^{(\delta^+-\delta^-)(T-t)}-1) - c_g(\delta^+ e^{(\delta^+-\delta^-)(T-t)} - \delta^-)}{(\delta^- e^{(\delta^+-\delta^-)(T-t)} - \delta^+) - c_g B(e^{(\delta^+-\delta^-)(T-t)} - 1)},$$

for $t \in [0, T]$, where $B = 1/c_\alpha$, $C = c_X$, $\delta^\pm = -D \pm \sqrt{R}$, with $D = -\gamma/(2c_\alpha)$, $R = D^2 + BC$ and $\bar{x}_0 = \mathbb{E}[X_0]$. Additionally, we found $\bar{\chi}_t = 0$, and

$$\bar{x}_t = \bar{x}_0 e^{-\int_0^t \frac{\bar{\eta}_s}{c_\alpha} ds}.$$

The FBSDE system (5.14) is solved by replacing $\mathbb{E}[Y_t]$ with the explicit expression for $\bar{y}_t = \bar{\eta}_t \bar{x}_t + \bar{\chi}_t$, and using the ansatz $Y_t = \eta_t X_t + \chi_t$. One finds the following explicit formulas for the coefficient functions η_t and χ_t:

$$\eta_t = -c_\alpha \sqrt{c_X/c_\alpha} \frac{c_\alpha \sqrt{c_X/c_\alpha} - c_g - (c_\alpha \sqrt{c_X/c_\alpha} + c_g) e^{2\sqrt{c_X/c_\alpha}(T-t)}}{c_\alpha \sqrt{c_X/c_\alpha} - c_g + (c_\alpha \sqrt{c_X/c_\alpha} + c_g) e^{2\sqrt{c_X/c_\alpha}(T-t)}},$$

$$\chi_t = (\bar{\eta}_t - \eta_t)\bar{x}_t.$$

Finally, the optimal control (5.13) is given by $\hat{\alpha}_t = \hat{\alpha}(t, X_t)$ where

$$\hat{\alpha}(t, x) = -\frac{1}{c_\alpha}\left(\eta_t x + (\bar{\eta}_t - \eta_t)\bar{x}_t\right). \tag{5.15}$$

5.5.2 The MFC trader problem

In the case of mean field control (*i.e.*, control of McKean-Vlasov dynamics), following [1, Theorem 3.2] and [58, Section 5.3.2], we find that the optimal control is given by

$$\alpha_t^* = -\frac{1}{c_\alpha}\left(Y_t - \gamma \mathbb{E}[X_t]\right), \tag{5.16}$$

which differs from the equilibrium control (5.13) from the MFG solution because the optimality condition in the MFC case involves the derivative of the Hamiltonian (5.12) with respect to the distribution of controls. More precisely, we have

$$0 = \partial_\alpha H(X_t, \alpha_t, \theta_t, Y_t) + \tilde{\mathbb{E}}\left[\partial_\theta H(\tilde{X}_t, \tilde{\alpha}_t, \tilde{\theta}_t, \tilde{Y}_t)(\alpha_t)\right] = c_\alpha \alpha_t + Y_t - \gamma \mathbb{E}[X_t].$$

Then, the corresponding FBSDE system becomes

$$\begin{cases} dX_t = -\dfrac{1}{c_\alpha}\left(Y_t - \gamma\mathbb{E}[X_t]\right)dt + \sigma dW_t, \quad X_0 \sim \mu_0 \\ dY_t = -\left(c_X X_t + \dfrac{\gamma}{c_\alpha}\mathbb{E}[Y_t] - \dfrac{\gamma^2}{c_\alpha}\mathbb{E}[X_t]\right)dt + Z_t dW_t, \quad Y_T = c_g X_T. \end{cases} \tag{5.17}$$

As a consequence, the two FBSDE systems (5.14) and (5.17) respectively for MFG and MFC differ.

Solution of the MFC problem

The approach to obtain the solution of the MFC problem is similar to what was presented in Section 5.5.1 for the MFG, but taking into consideration the extra terms due to the derivative of the Hamiltonian with respect to the distribution of controls.

First, taking expectation in (5.17), one obtains the following system of forward-backward ODEs:

$$\begin{cases} \dot{\bar{x}}_t = -\dfrac{1}{c_\alpha}\left(\bar{y}_t - \gamma\bar{x}_t\right), \quad \bar{x}_0 = x_0, \\ \dot{\bar{y}}_t = -\left(c_X\bar{x}_t + \dfrac{\gamma}{c_\alpha}\bar{y}_t - \dfrac{\gamma^2}{c_\alpha}\bar{x}_t\right), \quad \bar{y}_T = c_g\bar{x}_T. \end{cases} \tag{5.18}$$

Using the ansatz $\bar{y}_t = \bar{\phi}_t\bar{x}_t + \bar{\psi}_t$, we deduce that the coefficient functions $\bar{\phi}_t$ and $\bar{\psi}_t$ must satisfy

$$\begin{cases} \dot{\bar{\phi}}_t + 2\dfrac{\gamma}{c_\alpha}\bar{\phi}_t - \dfrac{1}{c_\alpha}\bar{\phi}_t^2 + c_X - \dfrac{\gamma^2}{c_\alpha}, \quad \bar{\phi}_T = c_g, \\ \dot{\bar{\psi}}_t + \dfrac{1}{c_\alpha}(\gamma - \bar{\phi}_t)\bar{\psi}_t = 0, \quad \bar{\psi}_T = 0. \end{cases} \tag{5.19}$$

From the second equation we get $\bar{\psi}_t = 0$ for all $t \in [0, T]$, and solving the Riccati equation for $\bar{\phi}_t$, we obtain:

$$\bar{\phi}_t = -\frac{1}{R}\frac{(c_2 + Rc_g)c_1 e^{(T-t)(c_2-c_1)} - c_2(c_1 + Rc_g)}{(c_2 + Rc_g)e^{(T-t)(c_2-c_1)} - (c_1 + Rc_g)}, \tag{5.20}$$

where $c_{1/2} = \frac{-a\pm\sqrt{a^2-4b}}{2}$ are the roots of $c^2 + ac + b = 0$, with $a = 2\gamma R$, $b = R(\gamma^2 R - c_X)$, and $R = 1/c_\alpha$.

Using $\bar{y}_t = \bar{\phi}_t \bar{x}_t$ in the first equation of (5.18), we obtain a first-order linear equation for $\bar{x}_t$ which admits the solution

$$\bar{x}_t = \bar{x}_0 e^{-\frac{1}{c_\alpha}\left(\int_0^t \bar{\phi}_s ds - \gamma t\right)}. \tag{5.21}$$

The solution of the McKean-Vlasov FBSDE system (5.17) is obtained using the ansatz $Y_t = \phi_t X_t + \psi_t$. Observe that the drift terms in the equations for Y_t in the systems (5.14) and (5.17) have the same linear component $-c_X X_t$. Due to this similarity, the slope coefficient functions η_t and ϕ_t are identical;

$$\eta_t = \phi_t, \quad \text{for all} \quad t \in [0, T].$$

However, the function $\psi_t = (\bar{\phi}_t - \phi_t)\bar{x}_t$ differs from χ_t in the MFG case due to the new formulations of $\bar{\phi}_t$ and $\bar{x}_t$ given in (5.20) and (5.21). Finally, the optimal control (5.16) is given by $\alpha^*_t = \alpha^*(t, X_t)$ where

$$\alpha^*(t, x) = -\frac{1}{c_\alpha}\left(\phi_t x + (\bar{\phi}_t - \phi_t - \gamma)\bar{x}_t\right). \tag{5.22}$$

5.5.3 Numerical results

In this section, numerical results of the application of the U2-MF-QL-FH algorithm to the trader problem are discussed. As in the case of the mean field capital accumulation problem, the interaction with the population is through the law of the controls. The algorithm 3 is adapted to this case as discussed in Section 5.3.1.

We consider the problem defined by the choice of parameters: $c_\alpha = 1$, $c_x = 2$, $\gamma = 1.75$, and $c_g = 0.3$. The time horizon is equal to $T = 1$. The distribution of the inventory process at initial time X_{t_0} is Gaussian with mean 0.5 and standard deviation 0.3. The volatility of the process X_t is given by $\sigma = 0.5$.

This problem is characterized by continuous time and continuous state and action spaces.

In order to solve this problem using the U2-MF-QL-FH algorithm, truncation and discretization techniques together with a projector operator are applied. The time interval $[0, T]$ is uniformly discretized as $\tau = \{t_0, \dots, t_{N_T} = T\}$ with $\Delta t = 1/16$. The state and action spaces are truncated and discretized as discussed in Section 5.3.3. The truncation parameters are chosen large enough to make sure that the state is within the boundary most of the time. This may induce a different truncation in the MFG and the MFC version of each problem.

In the MFG (resp. MFC), the action space is given by $\mathcal{A} = \{a_0 = -2.5, \dots, a_{|\mathcal{A}|-1} = 1\}$ (resp. $\mathcal{A} = \{a_0 = -0.25, \dots, a_{|\mathcal{A}|-1} = 5\}$) and the state space by $\mathcal{X} = \{x_0 = -1.5, \dots, x_{|\mathcal{X}|-1} = 1.75\}$ (resp. $\mathcal{X} = \{x_0 = -0.75, \dots, x_{|\mathcal{X}|-1} = 4\}$). The step size for the discretization of the spaces $\mathcal{A}$, and $\mathcal{X}$ is given by $\Delta_a = \Delta_x = \sqrt{\Delta t} = 1/4$. The exploitation-exploration trade off is tackled on each episode using an $\epsilon-$greedy policy. Suppose the agent is in state x, the algorithm picks the action that is optimal based on the current estimates with probability $1 - \epsilon$ and a random action in $\mathcal{A}$ with probability ϵ. In particular, the value of ϵ is fixed to 0.1.

The following numerical results show how the U2-MF-QL-FH algorithm is able to learn an approximation of the control function and the mean field term in the MFG and MFC cases depending on the choice of the parameters $(\omega^Q, \omega^\theta)$.

Learning of the controls

Figures 5.13, 5.14, 5.15, 5.17, 5.18: controls learned by the algorithm. The controls learned by the U2-MF-QL-FH algorithm are compared with the theoretical solutions. Each plot corresponds to a different time point $t \in \{0, 0.5, 1\}$. The layout is the same applied for the mean field capital accumulation problem in Section 5.4.4. On the left, the choice $(\omega^Q, \omega^\theta) = (0.55, 0.85)$ produces the approximation of the solution of the MFG.

On the right, the values of the parameters $(\omega^Q, \omega^\theta) = (0.65, 0.15)$ lets the algorithm to approach the solution of the MFC problem. The accuracy of the approximation is better at initial times and degrades towards the final horizon showing an higher complexity of the tuning of the algorithm to this problem. The results presented in the Figures are averaged over 10 runs.

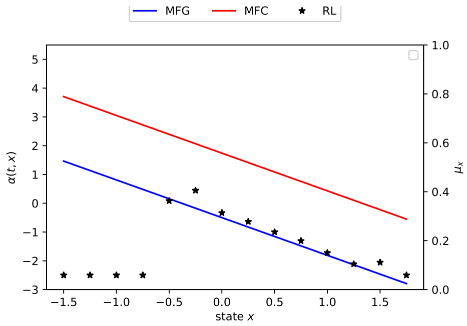

Figure 5.13: Learned Controls for MFG at time 0.

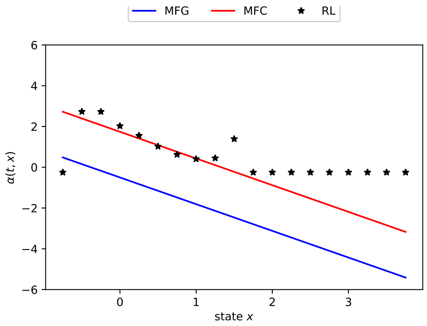

Figure 5.14: Learned Controls for MFC at time 0.

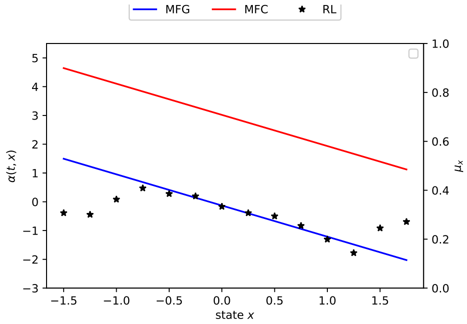

Figure 5.15: Learned Controls for MFG at time 7/16.

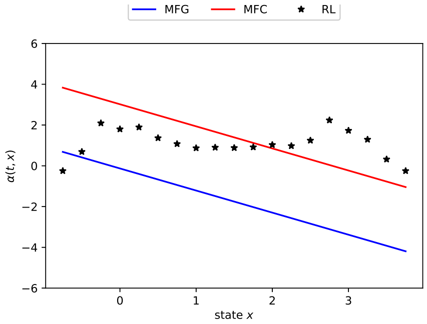

Figure 5.16: Learned Controls for MFC at time 7/16.

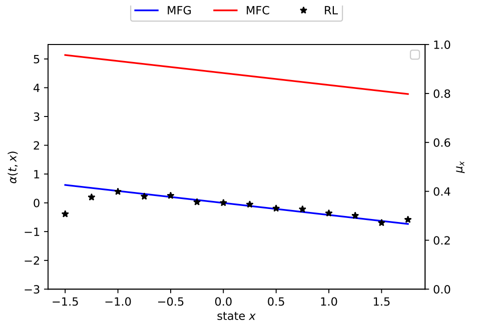

Figure 5.17: Learned Controls for MFG at time 15/16.

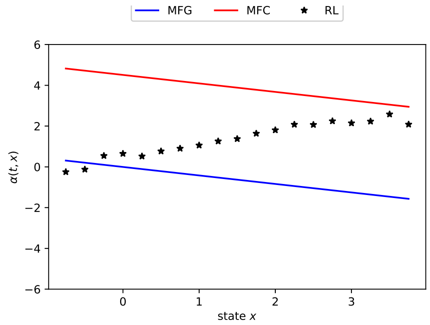

Figure 5.18: Learned Controls for MFC at time 15/16.

Learning of the mean field

Figures 5.19, 5.20, 5.21, 5.22, 5.23, 5.24: $\mathbb{E}[\theta_t]$ learned by the algorithm. The estimation of the first moment of the distribution of the controls evolves with respect to the number of learning episodes. Each plot corresponds to a different time point $t \in \{0, 0.5, 1\}$. The layout is the same described in Section 5.4.4. On the left, the solution of the MFG is obtained choosing $(\omega^Q, \omega^\theta) = (0.55, 0.85)$. On the right, the MFC solution is approached by the set of parameters $(\omega^Q, \omega^\theta) = (0.65, 0.15)$. The results presented in the Figures are averaged over 10 runs.

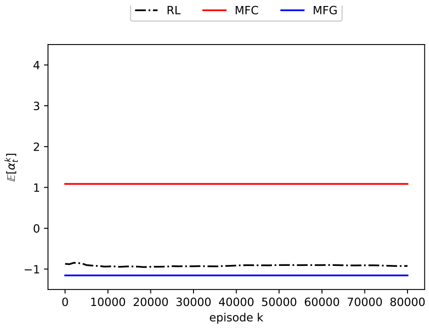

Figure 5.19: Learned control's mean for MFG at time 0.

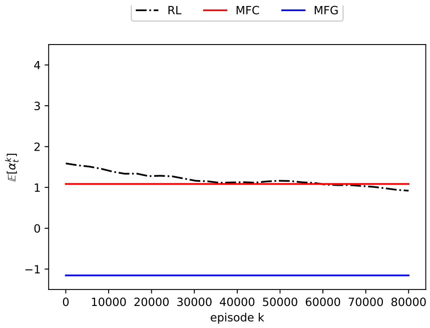

Figure 5.20: Learned control's mean for MFC at time 0.

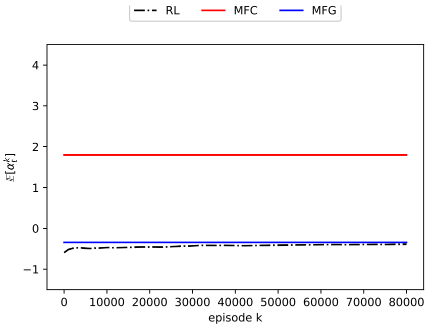

Figure 5.21: Learned control's mean for MFG at time 7/16.

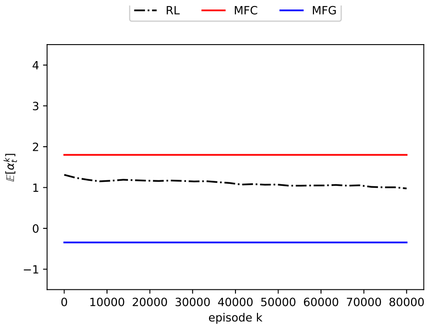

Figure 5.22: Learned control's mean for MFC at time 7/16.

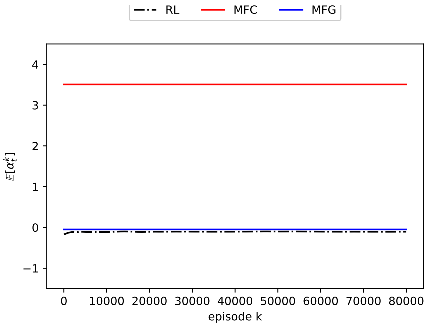

Figure 5.23: Learned control's mean for MFG at time 15/16.

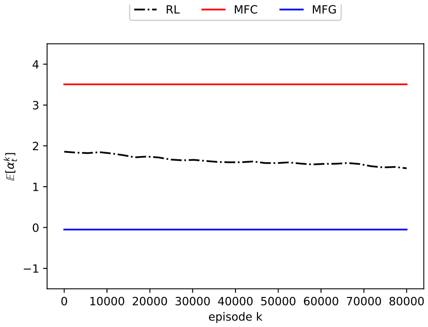

Figure 5.24: Learned control's mean for MFC at time 15/16.

Chapter 6

Deep Reinforcement Learning for Mean Field Games and Mean Field Control Problems with Continuous Space

In the previous chapters, we discuss a Q-learning based solving scheme for mean field problems in the case of finite spaces. We show how this method can be adapted to solve some of the continuous problems arising from real world applications. However, this extension requires a calibration of the algorithm tailored on the specific problem which may not be trivial. In order to avoid this challenging step, we are working on an approach which is designed specifically for problems defined on continuous spaces. Algorithms to solve classical MDPs with continuous spaces have been extensively studied (we refer to Chapter 7 of [74] for an exhaustive overview).

In our on-going work [9], we propose a Unified three-scale Mean Field Actor-Critic (U3-MF-AC) algorithm able to solve mean field games and mean field control problems in the same fashion as our Q-learning method.

We start presenting the classical Actor-Critic (AC) approach following the presentation given in [74]. We introduce our new AC based algorithm to tackle mean field problems. We conclude by showing numerical results on the asymptotic linear quadratic problem discussed in Section 3.5 and an infinite horizon variation of the capital accumulation problem discussed in Section 5.4.

6.1 Actor-Critic

Actor critic algorithms are popular model free methods in RL to solve classical MDPs in case of continuous spaces. This approach is characterized by two main parts: the actor, corresponding to the policy followed by the agent, and the critic, represented by the value function and acting as an evaluation of the actor. Even if the optimal policy is expected to be deterministic, the critic is a randomized strategy in order to allow exploration of the unknown environment. As more knowledge of the environment is collected and the algorithm converges to the solution of the MDP, the variance of the policy vanishes defining a deterministic policy.

In order to handle continuous spaces, the actor and the critic are approximated by parametric functions (e.g. neural networks). Let $\pi_\psi : \mathcal{S} \times \mathcal{A} \times \Psi \to [0, 1]$ be a representation of a randomized policy within the parameter space $\Psi \subset \mathbb{R}^{D_\Psi}$. Let $V_\theta^{\pi_\psi} : \mathcal{S} \times \Theta \to \mathbb{R}$ be the corresponding evaluation of the value function within the parameter space $\Theta \subset \mathbb{R}^{D_\Theta}$. Diagram 6.1 shows how actor critic algorithms are able to solve an MDP without any knowledge of the environment.

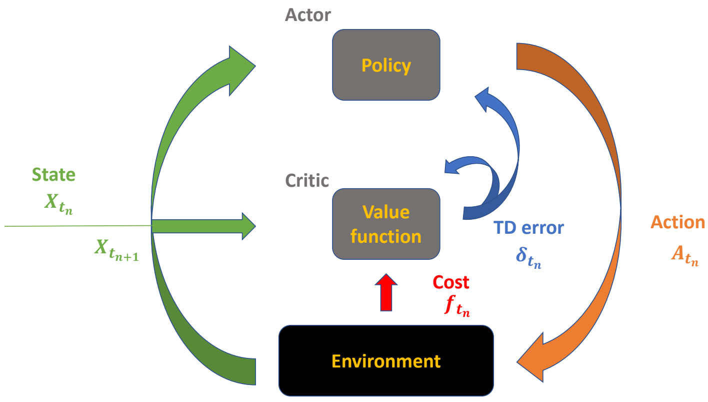

Figure 6.1: Diagram inspired from [71]

At each learning episode k and step t_n, the agent picks an action based on the strategy π_ψ (the actor) and her state $X^k_{t_n}$. Due to this action, the environment generates a cost $f^k_{t_n}$ and a new state $X^k_{t_{n+1}}$. The pair $(f^k_{t_n}, X^k_{t_{n+1}})$ acts as input of the Temporal Difference (TD) error defined for a given choice of parameters $\theta \in \Theta$ as follows

$$\delta^k_{t_n}(\theta) = f^k_{t_n} + \gamma V^{\pi_\psi}(X^k_{t_{n+1}}) - V^{\pi_\psi}_\theta(X^k_{t_n}),$$

where the first evaluation of the value function is considered independent of the parameter θ explaining why the subscript is omitted. The TD error is derived from the Bellman equation of the value function in the same fashion as the update rule for the Q function discussed in Section 2.1. Learning of the value function is obtained by minimizing the squared TD error through a stochastic gradient-descent step. The resulting update rule is given by

$$\theta' = \theta - \frac{1}{2}\rho^V_{k,t_n} \boldsymbol{\nabla}_\theta \left(\delta^k_{t_n}(\theta)\right)^2 = \theta + \rho^V_{k,t_n}\delta^k_{t_n}(\theta)\boldsymbol{\nabla}_\theta V_\theta(X^k_{t_n}),$$

where ρ^V_{k,t_n} represents the learning weight at episode k and step t_n.

The solution of the classical MDP is represented by the optimal policy π^* for which the corresponding value function is minimal. Consequently, the update rule for the policy parameters π_ψ is obtained by a stochastic gradient-descent step on the quantity $V_\theta^{\pi_\psi}$, i.e.

$$\psi^{'} = \psi + \rho^{\pi}_{k,t_n} \boldsymbol{\nabla}_\psi V_\theta^{\pi_\psi}(X^k_{t_n}),$$

where ρ^{π}_{k,t_n} defines the learning weight.

Intuitively, one may expect that the knowledge of the model is required to evaluate the gradient $\boldsymbol{\nabla}_\psi V_\theta^{\pi_\psi}(X^k_{t_n})$ given that the transitional probabilities depend on π_ψ. However, the Policy Gradient Theorem shows that this quantity is independent of the dynamics of the model allowing to design model free approaches (see [74] for more details). The resulting update rule for the classical actor critic algorithm is given by

$$\psi^{'} = \psi + \rho^{\pi}_{t_n} \delta_{t_n}(\theta) \boldsymbol{\nabla}_\psi \log \pi_\psi(X_{t_n}, A_{t_n}).$$

The actor-critic algorithm has been cast as a two-scale stochastic approximation procedure by Borkar and Konda in [17]. The learning weights of the critic ρ^V are chosen larger than the corresponding rates of the actor ρ^π. Intuitively, this situation is equivalent to having two nested loops. At each step of the outer loop (slower) corresponds a choice of a policy and several steps of the inner loop (faster) in which the corresponding value function is estimated. Once the estimation procedure is completed, its value is passed to the outer loop and a new policy is chosen based on it. Due to the stochastic gradient descent step, the new policy is such that the corresponding value function is not greater in value. Theoretically, the procedure completes when the gradient reaches value zero. In practice, the algorithm stops when the change of the value function is lower than a given tolerance.

6.2 U3-MF-AC: Unified Three Timescales Mean Field Actor Critic

In order to solve asymptotic mean field problems, we propose algorithm 4, a new AC based method which is derived as a three-scale stochastic approximation procedure. The actor and the critic are approximated by two Neural Networks (NNs). The two learning rules introduced above are combined with an extra update rule which allows the learning of the distribution of the population. The current version is tailored to problems in which the interaction is through the first moment of the limiting distribution.

In the same fashion of algorithm 2, an estimate of the first moment of the limiting distribution of the population is computed through successive learning episodes. At each step t_n, an estimate $m_{t_n}^k$ is updated based on a sample $X_{t_n}^k$ collected from episodes $k = 1, 2, \dots$. One may expect each m_{t_n} to converge to an approximation of the first moment of the limiting distribution. The update rule is given by

$$m_{t_n}^k = m_{t_n}^{k-1} + \rho_k^m (X_{t_n}^k - m_{t_n}^k),$$

where ρ_k^m corresponds to the learning weight.

We show through our preliminary results how a different choice of the learning rates ρ^m allows the algorithm to solve a MFG or a MFC problem representing a unified approach to mean field problems in continuous spaces.

Algorithm 4 U3-MF-AC: Unified Three-scale Mean Field Actor-Critic

```
Require: $\mathcal{X} = \mathbb{R}, \mathcal{A} = \mathbb{R}$ : state and action spaces, $tol_V$, $tol_\pi$, $tol_m$: break rule tolerances,
    $\tau = \{t_0 = 0, \dots, t_{|\tau|-1} = T\}$ with $t_i \leqslant t_{i+1}$ : time discretization where $T >> 0$,
    $\rho^V, \rho^\pi, (\rho^m_k)_{k \geqslant 0}$ learning rates for the critic, actor and mean field term respectively.
Initialization: Create two NNs within the parameter sets $\Theta \subset \mathbb{R}^{D_\Theta}$ and $\Psi \subset \mathbb{R}^{D_\Psi}$ :
    Critic $V_{\theta_0^k} : \mathcal{X} \mapsto \mathcal{R}$, approximation of the value function given $\theta_0^k \in \Theta$,
    Actor $\pi_{\psi_0^k} : \mathcal{X} \mapsto \mathcal{P}(\mathcal{A})$, approximation of the optimal policy given $\psi_0^k \in \Psi$,
    Mean field first moment $m_{t_n}^k = 0$ for $n = 0, \dots, |\tau| - 1$ and episode $k = 0$.
repeat
    Update episode index k = k + 1; Observe $X_{t_0}^k$ provided by the environment
    for $n \leftarrow 0$ to $|\tau| - 1$ do
        Sample action $A_{t_n}^k$ from $\pi_{\psi_n^k}(X_{t_n}^k)$
        Observe: cost $f_{t_n}^k = f(X_{t_n}^k, A_{t_n}^k, m_{t_n}^k)$, state $X_{t_{n+1}}^k$ provided by the environment
        Compute: TD error: $\delta_n^k = f_{t_n}^k + \gamma V_{\theta_n^k}^{\pi_{\psi_n^k}}(X_{t_{n+1}}^k) - V_{\theta_n^k}^{\pi_{\psi_n^k}}(X_{t_n}^k)$
        Update Critic : $\theta_{n+1}^k = \theta_n^k + \rho^V \delta_n^k \boldsymbol{\nabla}_\theta V_{\theta_n^k}^{\pi_{\psi_n^k}}(X_{t_n}^k)$
        Update Actor : $\psi_{n+1}^k = \psi_n^k + \rho^\pi \delta_n^k \boldsymbol{\nabla}_\psi \log \pi_{\psi_n^k}(X_{t_n}^k, A_{t_n}^k)$
        Update Mean Field : $m_{t_n}^k = m_{t_n}^{k-1} + \rho_k^m (X_{t_n}^k - m_{t_n}^k)$
    end for
until $\left|\boldsymbol{\nabla}_\theta V_{\theta_n^k}^{\pi_{\psi_n^k}}(X_{t_n}^k)\right| < tol_V$, $\left|\boldsymbol{\nabla}_\psi \log \pi_{\psi_n^k}(X_{t_n}^k, A_{t_n}^k)\right| < tol_\pi$ and $\|m_T^{k-1} - m_T^k\|_2 < tol_m$
```

The extension of this procedure to problems depending on the full distribution is a work in progress. An approach under investigation consists to choose a parameterization for the distribution and calibrate it based on the data flow.

6.3 Numerical experiments

In this section, we present some preliminary numerical results obtained by applying the U3-MF-AC algorithm to two examples: the linear quadratic model discussed in Section 3.5 and an infinite horizon version of the capital accumulation problem presented in Section 5.4. In both cases, the actor is designed as a Gaussian exploration policy. In particular, the state of the model is given as input to a feed-forward NN with an hidden layer of 64 nodes and Exponential Linear Unit (ELU) as activation function. The network returns two outputs: one representing the mean of the Gaussian policy and the other which is passed to a softmax operator. The resulting value corresponds to the standard deviation of the policy. On the other hand, the value function is defined as a feed-forward NN with an hidden layer of 128 nodes and ELU activation function. The learning rates for the actor and critic are fixed to the values $\rho^{\pi} = 5 \times 10^{-6}$ and $\rho^{V} = 10^{-5}$. The stochastic gradient descent steps are executed using the Adam optimizer [53]. The learning rate for the mean field term is defined as $\rho^{m}_{k} = \frac{1}{(1+k)^{\omega}}$ where ω is chosen depending on the problem.

6.3.1 A linear quadratic example

In Section 3.5, we presented the solution of a linear quadratic model in its asymptotic MFG and MFC versions. In Section 4.3, we showed how the U2-MF-QL algorithm is able to solve both problems depending on the choice of the learning rates. Since the example is defined in continuous spaces, the algorithm requires a calibration based on truncation and discretization techniques presented in Section 4.2.2. In general this procedure should be performed based on a trial and error approach which may not be trivial.

In this section, we show the results obtained by applying the U3-MF-AC algorithm to this example for the choice of parameters $c_1 = 0.25$, $c_2 = 1.5$, $c_3 = 0.5$, $c_4 = 0.6$, discount parameter $\beta = 1$ and volatility $\sigma = 0.3$. The infinite horizon is truncated at time $T = 20$. The continuous time is discretized using step $\Delta_t = 10^{-2}$.

In Figures 6.2, 6.3, 6.4, 6.5, we show how the U3-MF-AC algorithm is able to learn the optimal control function and the first moment of the limiting distribution in the MFG and MFC frameworks fixing the parameter in the learning rates of the mean field term equals to $\omega^{MFG} = 0.7$ and $\omega^{MFC} = 0.05$ respectively. Indeed, a slow update of this term mimics the solving scheme of a MFG. On the other hand, a fast update allows the distribution to evolve accordingly to the policy aligning the algorithm to the MFC framework. Differently from the U2-MF-QL approach, this method does not require any calibration representing a simplified and flexible approach to solve mean field problems with continuous spaces.

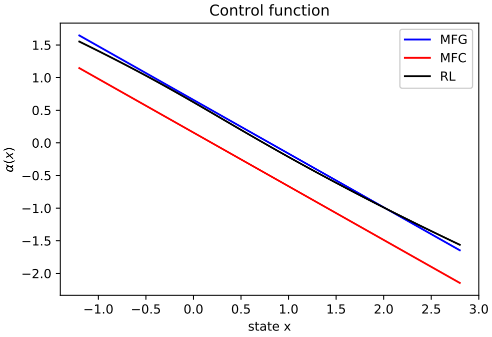

Figure 6.2: MFG: results after 15k learning episodes

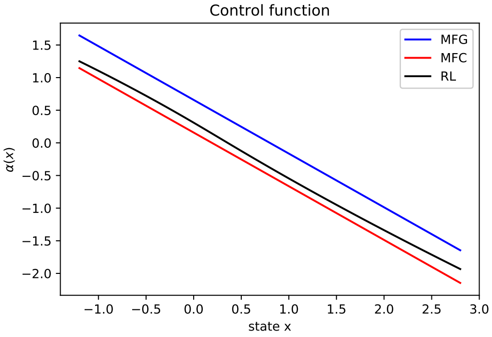

Figure 6.3: MFC: results after 15k learning episodes

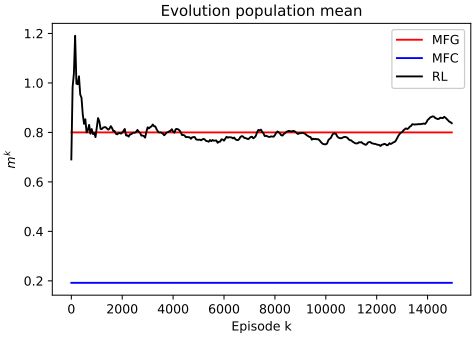

Figure 6.4: MFG: learning evolution of m_t through 15k episodes

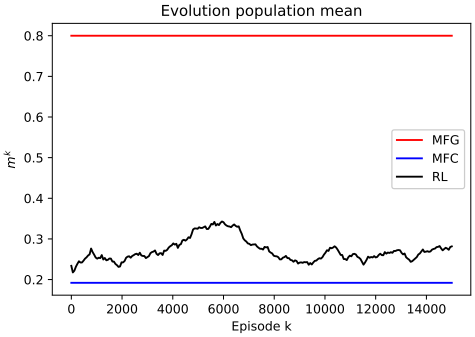

Figure 6.5: MFC: learning evolution of m_t through 15k episodes

6.3.2 A mean field accumulation problem

In this section, we show the results obtained by applying the U3-MF-AC to an infinite horizon version of the mean field accumulation problem from [50] discussed in Section 5.4.

The time horizon is equal to $T = \infty$. The remaining parameters are fixed as follows: discount factor $\rho = 0.95$, utility factor $\gamma = 0.2$, constant $C = 2$, and the noise W has support $D_W = \{0.9, 1.3\}$ and corresponding probabilities $\{0.75, 0.25\}$.

In Figures 6.6 and 6.7, we show how the algorithm is able to learn the control function and the first moment of the learning distribution given the parameter of the limiting rate for the mean field term equals to $\omega = 0.85$. We train it using 35×10^3 episodes of 100 steps. Differently from the previous example, this problem is not of the linear quadratic family, the noise is multiplicative and the admissible control set is not static but changes depending on the state of the model. These differences show the flexibility of the algorithm to different kinds of problem.

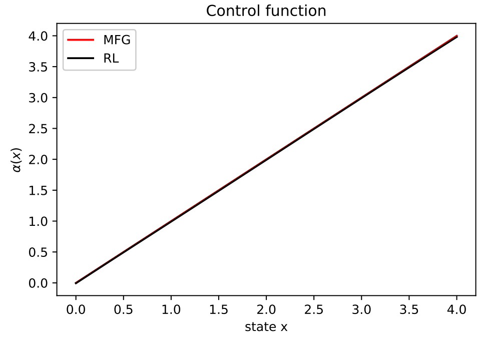

Figure 6.6: MFG: learning the controls through 35k episodes

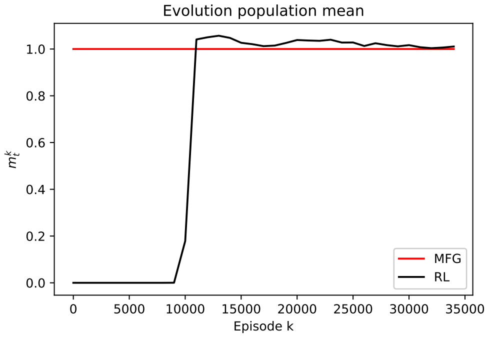

Figure 6.7: MFG: learning evolution of m_t through 35k episodes

These preliminary results encourage a deeper investigation of this procedure. In particular, we are working on extensions able to deal with the full distribution of the population and finite horizon problems together with a theoretical analyses of this method.

Chapter 7

Conclusions

In this manuscript, we presented our contribution to the field of differential games by proposing a data driven solving approach. Our method is derived by merging mean field theory, reinforcement learning and multi scale stochastic approximation.

In order to achieve this goal, we propose in [6] a new definition of asymptotic mean field games and mean field control problems which facilitates a connection with the reinforcement learning framework. We unify the two problems through a two timescale perspective and present a Q-learning based solving scheme. In order to obtain this method, we introduce a new Bellman equation for a modified Q-function that is tailored to the MFC framework. The algorithm is tested on an infinite horizon linear quadratic example.

This approach is extended in [7] to the case of interaction through the distribution of controls and finite horizon. We have illustrated the second algorithm with two examples: an optimal investment problem with HARA utility function, and an optimal liquidation problem. Differently from the others, the first problem is not of the linear quadratic family, presents multiplicative noise and is characterized by a dynamic admissible control set. These core differences show the flexibility of our approach.

The main ingredients of these algorithms are the learning rates for the Q-matrix and for the distribution of the states (controls) of the population. Their relative decay with respect to the number of episodes is the key quantity to stir the algorithm towards learning the optimal controls for MFG or MFC problem. Roughly speaking, updating the Q-matrix faster (resp. slower) than the distribution of states (controls) leads to the MFG (resp. MFC) solution. Convergence follows by applying Borkar's results as shown in [6] in the case of infinite horizon problems. Choosing these rates in an optimal way remains the main challenge in specific applications. In particular, we expect that allowing these rates to depend on the time steps could lead to improved results. This aspect is left for future investigations.

The algorithms presented here are the context of finite space via the Q-matrix even though the proposed examples are originally in continuous space and then discretized. Dealing directly with a continuous space is the topic of the ongoing work on deep reinforcement learning for mean filed problems [9].

The area of reinforcement learning for mean field problems is extremely rich with a huge potential for applications in various disciplines. It is in its infancy, and we hope that the results and explanations presented here will be helpful to newcomers interested in this direction of research.

Bibliography

[1] B. Acciaio, J. Backhoff-Veraguas, and R. Carmona, *Extended mean field control problems: stochastic maximum principle and transport perspective*, *SIAM J. Control Optim., to appear* (2018).

[2] Y. Achdou and M. Laurière, *Mean field games and applications: Numerical aspects*, in *Mean Field Games*, vol. 2281 of *C.I.M.E. Foundation Subseries*. Springer International Publishing, 2020.

[3] A. Al-Aradi, A. Correia, D. Naiff, G. Jardim, and Y. Saporito, *Solving nonlinear and high-dimensional partial differential equations via deep learning*, *arXiv preprint arXiv:1811.08782* (2018).

[4] C. Alasseur, I. Ben Tahar, and A. Matoussi, *An extended mean field game for storage in smart grids*, *Forthcoming in Journal of Optimization Theory and Applications* (2019).

[5] B. Anahtarci, C. D. Kariksiz, and N. Saldi, *Q-learning in regularized mean-field games*, *arXiv preprint arXiv:2003.12151* (2020).

[6] A. Angiuli, J.-P. Fouque, and M. Laurière, *Unified reinforcement q-learning for mean field game and control problems*, to appear in *Mathematics of Control, Signals, and Systems (MCSS)* (2021). https://arxiv.org/pdf/2006.13912.pdf.

[7] A. Angiuli, J.-P. Fouque, and M. Lauriere, *Reinforcement learning for mean field games, with applications to economics*, in *Machine Learning in Financial Markets: A Guide to Contemporary Practice* (A. Capponi and C.-A. Lehalle, eds.). Cambridge University Press, Cambridge, to appear, 2021. https://arxiv.org/pdf/2106.13755.pdf.

[8] A. Angiuli, C. V. Graves, H. Li, J.-F. Chassagneux, F. Delarue, and R. Carmona, *Cemracs 2017: numerical probabilistic approach to mfg*, *ESAIM: Proceedings and Surveys* **65** (2019) 84–113. https://www.esaim-proc.org/articles/proc/pdf/2019/01/proc196505.pdf.

[9] A. Angiuli and R. Hu, “Deep reinforcement learning for mean field games and mean field control problems in continuous spaces.” In preparation, 2021.

[10] R. E. Bellman and S. E. Dreyfus, *Applied dynamic programming*, vol. 2050. Princeton university press, 2015.

[11] A. Bensoussan, J. Frehse, P. Yam, *et. al.*, *Mean field games and mean field type control theory*, vol. 101. Springer, 2013.

[12] A. Bensoussan, J. Frehse, and S. C. P. Yam, *Mean field games and mean field type control theory.* Springer Briefs in Mathematics. Springer, New York, 2013.

[13] C. Bertucci, J.-M. Lasry, and P.-L. Lions, *Some remarks on mean field games*, *Comm. Partial Differential Equations* **44** (2019), no. 3 205–227.

[14] F. J. Bonnans, S. Hadikhanloo, and L. Pfeiffer, "Schauder estimates for a class of potential mean field games of controls." arXiv:1902.05461, 2019.

[15] V. S. Borkar, *Stochastic approximation with two time scales*, *Systems & Control Letters* **29** (1997), no. 5 291–294.

[16] V. S. Borkar, *Stochastic approximation.* Cambridge University Press, Cambridge; Hindustan Book Agency, New Delhi, 2008. A dynamical systems viewpoint.

[17] V. S. Borkar and V. R. Konda, *The actor-critic algorithm as multi-time-scale stochastic approximation*, *Sadhana* **22** (1997), no. 4 525–543.

[18] H. Cao, X. Guo, and M. Laurière, *Connecting GANs and MFGs*, *arXiv preprint arXiv:2002.04112* (2020).

[19] P. Cardaliaguet, F. Delarue, J.-M. Lasry, and P.-L. Lions, *The master equation and the convergence problem in mean field games.* Princeton University Press, 2019.

[20] P. Cardaliaguet and S. Hadikhanloo, *Learning in mean field games: the fictitious play*, *ESAIM: Control, Optimisation and Calculus of Variations* **23** (2017), no. 2.

[21] P. Cardaliaguet and C.-A. Lehalle, *Mean field game of controls and an application to trade crowding*, *Math. Financ. Econ.* **12** (2018), no. 3 335–363.

[22] R. Carmona, , and M. Laurière, "Deep learning for Mean Field Games, with applications to finance." In preparation., 2021.

[23] R. Carmona, *Applications of mean field games to economic theory*, *Proc. AMS Short Course, arXiv preprint arXiv:2012.05237* (2020).

[24] R. Carmona and F. Delarue, *Probabilistic Theory of Mean Field Games with Applications I-II.* Springer, 2018.

[25] R. Carmona, C. V. Graves, and Z. Tan, *Price of anarchy for mean field games*, in *CEMRACS 2017—numerical methods for stochastic models: control, uncertainty quantification, mean-field*, vol. 65 of *ESAIM Proc. Surveys*, pp. 349–383. EDP Sci., Les Ulis, 2019.

[26] R. Carmona and D. Lacker, *A probabilistic weak formulation of mean field games and applications*, *Ann. Appl. Probab.* **25** (2015), no. 3 1189–1231.

[27] 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*, *arXiv preprint arXiv:1908.01613* (2019).

[28] R. Carmona and M. Laurière, *Convergence analysis of machine learning algorithms for the numerical solution of mean field control and games i: The ergodic case*, *SIAM Journal on Numerical Analysis* **59** (2021), no. 3 1455–1485.

[29] R. Carmona, M. Laurière, and Z. Tan, "Linear-quadratic mean-field reinforcement learning: Convergence of policy gradient methods." Preprint, 2019.

[30] R. Carmona, M. Laurière, and Z. Tan, "Model-free mean-field reinforcement learning: Mean-field MDP and mean-field Q-learning." Preprint, 2019.

[31] P. Chan and R. Sircar, *Bertrand and Cournot mean field games*, *Appl. Math. Optim.* **71** (2015), no. 3 533–569.

[32] A. Charpentier, R. Elie, and C. Remlinger, *Reinforcement learning in economics and finance*, *arXiv preprint arXiv:2003.10014* (2020).

[33] M. F. Djete, D. Possamaï, and X. Tan, *Mckean-vlasov optimal control: the dynamic programming principle*, *arXiv preprint arXiv:1907.08860* (2019).

[34] R. Elie, J. Perolat, M. Laurière, M. Geist, and O. Pietquin, *On the convergence of model free learning in mean field games*, in *in proc. of AAAI*, 2020.

[35] E. Even-Dar and Y. Mansour, *Learning rates for q-learning*, *Journal of machine learning Research* **5** (2003), no. Dec 1–25.

[36] J.-P. Fouque and Z. Zhang, *Deep learning methods for mean field control problems with delay*, *Frontiers in Applied Mathematics and Statistics* **6(11)** (2020).

[37] Z. Fu, Z. Yang, Y. Chen, and Z. Wang, *Actor-critic provably finds nash equilibria of linear-quadratic mean-field games*, *arXiv preprint arXiv:1910.07498* (2019).

[38] M. Germain, J. Mikael, and X. Warin, *Numerical resolution of mckean-vlasov fbsdes using neural networks*, *arXiv preprint arXiv:1909.12678* (2019).

[39] D. A. Gomes, S. Patrizi, and V. Voskanyan, *On the existence of classical solutions for stationary extended mean field games*, *Nonlinear Anal.* **99** (2014) 49–79.

[40] D. A. Gomes and J. a. Saúde, *Mean field games models—a brief survey*, *Dyn. Games Appl.* **4** (2014), no. 2 110–154.

[41] D. A. Gomes and V. K. Voskanyan, *Extended deterministic mean-field games*, *SIAM J. Control Optim.* **54** (2016), no. 2 1030–1055.

[42] P. J. Graber, *Linear quadratic mean field type control and mean field games with common noise, with application to production of an exhaustible resource*, *Appl. Math. Optim.* **74** (2016), no. 3 459–486.

[43] P. J. Graber and A. Bensoussan, *Existence and uniqueness of solutions for Bertrand and Cournot mean field games*, *Appl. Math. Optim.* **77** (2018), no. 1 47–71.

[44] H. Gu, X. Guo, X. Wei, and R. Xu, *Dynamic programming principles for learning mfcs*, *arXiv preprint arXiv:1911.07314* (2019).

[45] H. Gu, X. Guo, X. Wei, and R. Xu, *Mean-field controls with Q-learning for cooperative MARL: Convergence and complexity analysis*, *arXiv preprint arXiv:2002.04131* (2020).

[46] H. Gu, X. Guo, X. Wei, and R. Xu, *Q-learning for mean-field controls*, *arXiv preprint arXiv:2002.04131* (2020).

[47] X. Guo, A. Hu, R. Xu, and J. Zhang, *Learning mean-field games*, in *Advances in Neural Information Processing Systems*, pp. 4966–4976, 2019.

[48] S. Hadikhanloo, *Learning in anonymous nonatomic games with applications to first-order mean field games*, *arXiv preprint arXiv:1704.00378* (2017).

[49] S. Hadikhanloo and F. J. Silva, *Finite mean field games: fictitious play and convergence to a first order continuous mean field game*, *Journal de Mathématiques Pures et Appliquées (9)* **132** (2019).

[50] M. Huang, *A mean field capital accumulation game with HARA utility*, *Dynamic Games and Applications* **3** (2013), no. 4 446–472.

[51] M. Huang, P. E. Caines, and R. P. Malhamé, *Large-population cost-coupled LQG problems with nonuniform agents: individual-mass behavior and decentralized ϵ-Nash equilibria*, *IEEE Trans. Automat. Control* **52** (2007), no. 9 1560–1571.

[52] M. Huang, R. P. Malhamé, and P. E. Caines, *Large population stochastic dynamic games: closed-loop McKean-Vlasov systems and the Nash certainty equivalence principle*, *Commun. Inf. Syst.* **6** (2006), no. 3 221–251.

[53] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, *arXiv preprint arXiv:1412.6980* (2014).

[54] Z. Kobeissi, "On classical solutions to the mean field game system of controls." arXiv:1904.11292, 2019.

[55] J.-M. Lasry and P.-L. Lions, *Mean field games*, *Jpn. J. Math.* **2** (2007), no. 1 229–260.

[56] M. Laurière, "On numerical methods for mean field games and mean field type control." Proc. AMS Short Course, 2020.

[57] M. Laurière and O. Pironneau, *Dynamic programming for mean-field type control*, *C. R. Math. Acad. Sci. Paris* **352** (2014), no. 9 707–713.

[58] M. Laurière and L. Tangpi, *Convergence of large population games to mean field games with interaction through the controls*, *arXiv preprint arXiv:2004.08351* (2020).

[59] A. T. Lin, S. W. Fung, W. Li, L. Nurbekyan, and S. J. Osher, *Apac-net: Alternating the population and agent control via two neural networks to solve high-dimensional stochastic mean field games*, *arXiv preprint arXiv:2002.10113* (2020).

[60] H. P. McKean Jr, *A class of markov processes associated with nonlinear parabolic equations*, *Proceedings of the National Academy of Sciences of the United States of America* **56** (1966), no. 6 1907.

[61] D. Mguni, J. Jennings, and E. M. de Cote, *Decentralised learning in systems with many, many strategic agents*, in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[62] M. Min and R. Hu, *Signatured deep fictitious play for mean field games with common noise*, *arXiv preprint arXiv:2106.03272* (2021).

[63] R. K. Mishra, D. Vasal, and S. Vishwanath, *Model-free reinforcement learning for non-stationary mean field games*, in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 1032–1037, IEEE, 2020.

[64] M. Motte and H. Pham, *Mean-field markov decision processes with common noise and open-loop controls*, *arXiv preprint arXiv:1912.07883* (2019).

[65] J. Pérolat, S. Perrin, R. Elie, M. Laurière, G. Piliouras, M. Geist, K. Tuyls, and O. Pietquin, *Scaling up Mean Field Games with Online Mirror Descent*, 2021.

[66] S. Perrin, M. Laurière, J. Pérolat, M. Geist, R. Élie, and O. Pietquin, *Mean field games flock! the reinforcement learning way*, *Accepted to IJCAI'21 (arXiv preprint arXiv:2105.07933)* (2021).

[67] S. Perrin, J. Pérolat, M. Laurière, M. Geist, R. Elie, and O. Pietquin, "Fictitious Play for Mean Field Games: Continuous Time Analysis and Applications." In preparation, 2020.

[68] H. Pham and X. Wei, *Discrete time mckean–vlasov control problem: a dynamic programming approach*, *Applied Mathematics & Optimization* **74** (2016), no. 3 487–506.

[69] L. Ruthotto, S. J. Osher, W. Li, L. Nurbekyan, and S. W. Fung, *A machine learning framework for solving high-dimensional mean field game and mean field control problems*, *Proceedings of the National Academy of Sciences* **117** (2020), no. 17 9183–9193.

[70] J. Subramanian and A. Mahajan, *Reinforcement learning in stationary mean-field games*, in *Proceedings. 18th International Conference on Autonomous Agents and Multiagent Systems*, 2019.

[71] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.

[72] W. Wang, J. Han, Z. Yang, and Z. Wang, *Global convergence of policy gradient for linear-quadratic mean-field control/game in continuous time*, *arXiv preprint arXiv:2008.06845* (2020).

[73] C. J. C. H. Watkins, *Learning from delayed rewards.* PhD thesis, King's College, Cambridge, 1989.

[74] M. A. Wiering and M. Van Otterlo, *Reinforcement learning*, *Adaptation, learning, and optimization* **12** (2012), no. 3.

[75] Q. Xie, Z. Yang, Z. Wang, and A. Minca, *Provable fictitious play for general mean-field games*, *arXiv preprint arXiv:2010.04211* (2020).

[76] J. Yang, X. Ye, R. Trivedi, H. Xu, and H. Zha, *Deep mean field games for learning optimal behavior policy of large populations*, in *International Conference on Learning Representations*, 2018.

[77] Y. Yang, R. Luo, M. Li, M. Zhou, W. Zhang, and J. Wang, *Mean field multi-agent reinforcement learning*, in *International Conference on Machine Learning*, pp. 5567–5576, 2018.